ACADÉMIE D'AIX-MARSEILLE
UNIVERSITÉ D'AVIGNON ET DES PAYS DE VAUCLUSE

THÈSE

présentée à l'Université d'Avignon et des Pays de Vaucluse
pour obtenir le diplôme de DOCTORAT

SPÉCIALITÉ : Informatique

École Doctorale 536 « Agrosciences et Sciences »
Laboratoire d'Informatique (EA 4128)

Vers une représentation du contexte thématique en Recherche d'Information

par
Romain DEVEAUD

Soutenue publiquement le 29 novembre 2013 devant un jury composé de :

M^{me}	Josiane MOTHE	Professeur, IRIT, Toulouse	Présidente du jury
M.	Jian-Yun NIE	Professeur, RALI, Montréal	Rapporteur
M.	Philippe MULHEM	Chargé de recherche CNRS, LIG, Grenoble	Rapporteur
M.	Jacques SAVOY	Professeur, IIUN, Neuchâtel	Examinateur
M.	Jaap KAMPS	Associate professor, ILLC, Amsterdam	Examinateur
M.	Benjamin PIWOWARSKI	Chargé de recherche CNRS, LIP6, Paris	Examinateur
M.	Eric SANJUAN	Maître de conférences, LIA, Avignon	Co-Directeur de thèse
M.	Patrice BELLOT	Professeur, LSIS, Marseille	Directeur de thèse

---

Laboratoire d'Informatique d'Avignon

Remerciements

Je commencerai par remercier mon co-directeur de thèse, Eric Sanjuan pour son immense implication tout au long de cette thèse dans tous les aspects liés de près ou de loin à ma recherche. Je le remercie notamment pour m'avoir soutenu, pour avoir su me faire confiance au bon moment et pour avoir laissé libre cours à mes intuitions, alors que ses propres intuitions allaient à l'encontre des miennes. Il a su m'insuffler une vision de la recherche qui lui est très personnelle, complémentaire avec celle que j'ai acquis par moi-même, et je lui en suis extrêmement reconnaissant.

Je voudrais également remercier mon autre directeur de thèse, Patrice Bellot pour son soutien et son expérience. Ses conseils avisés auront toujours été précieux, notamment pendant les périodes de rédaction intensive d'articles.

Je tiens à remercier vivement Jian-Yun Nie et Philippe Mulhem de me faire l'honneur de rapporter cette thèse, ainsi que pour les discussions enrichissantes et détendues que nous avons pu avoir au cours de diverses conférences où nous nous sommes rencontrés. Mes remerciements vont également à Josianne Mothe, Jacques Savoy, Jaap Kamps et Benjamin Piwowarski qui ont accepté d'examiner cette thèse.

Plus personnellement, je tiens à remercier ma famille, mes amis et bien sûr Joséphine pour ses encouragements, sa tendresse et son amour. Merci à mes parents et à mon grand-père qui ont constitué un point stable auquel j'ai pu me raccrocher dans les moments difficiles. Merci à Marion qui a eu le courage et la patience de relire cette thèse, Solène et Yoann pour la musique et Mickey, Matthieu pour les bières et les phrases non finies, Kevin, Julien, Marjorie, Olivier, Mathieu, Milène, Patoche et Dal, Ouag, Camille, Nico, Arnaud, Annabelle, Vincent, Vincent Highlander, Raphaëlle et Arnaud, Alessandro, Cyril, Jade, Killian, Hugo, Raphaël, Bassam, Young-Min, Stéphane, Benjamin, Greg, Manu et tant d'autres personnes qui ont croisé ma route et avec qui j'ai partagé un moment au cours de ces trois dernières années.

Je voudrai également adresser des remerciements particuliers à Florian Boudin et Ludovic Bonnefoy. Merci à Florian de m'avoir fait profiter de son amitié, de son humour et de ses conseils, ainsi que de m'avoir accordé son temps lors de nos collaborations. Un énorme merci à Ludo, pour toutes ces choses qui font que sans lui cette thèse n'aurait pas été la même.

Résumé

Quand des humains cherchent des informations au sein de bases de connaissances ou de collections de documents, ils utilisent un système de recherche d'information (SRI) faisant office d'interface. Les utilisateurs doivent alors transmettre au SRI une représentation de leur besoin d'information afin que celui-ci puisse chercher des documents contenant des informations pertinentes. De nos jours, la représentation du besoin d'information est constituée d'un petit ensemble de mots-clés plus souvent connu sous la dénomination de « requête ». Or, quelques mots peuvent ne pas être suffisants pour représenter précisément et efficacement l'état cognitif complet d'un humain par rapport à son besoin d'information initial. Sans une certaine forme de contexte thématique complémentaire, le SRI peut ne pas renvoyer certains documents pertinents exprimant des concepts n'étant pas explicitement évoqués dans la requête.

Dans cette thèse, nous explorons et proposons différentes méthodes statistiques, automatiques et non supervisées pour la représentation du contexte thématique de la requête. Plus spécifiquement, nous cherchons à identifier les différents concepts implicites d'une requête formulée par un utilisateur sans qu'aucune action de sa part ne soit nécessaire. Nous expérimentons pour cela l'utilisation et la combinaison de différentes sources d'information générales représentant les grands types d'information auxquels nous sommes confrontés quotidiennement sur internet. Nous tirons également parti d'algorithmes de modélisation thématique probabiliste (tels que l'allocation de Dirichlet latente) dans le cadre d'un retour de pertinence simulé. Nous proposons par ailleurs une méthode permettant d'estimer conjointement le nombre de concepts implicites d'une requête ainsi que l'ensemble de documents pseudo-pertinent le plus approprié afin de modéliser ces concepts. Nous évaluons nos approches en utilisant quatre collections de test TREC de grande taille. En annexes, nous proposons également une approche de contextualisation de messages courts exploitant des méthodes de recherche d'information et de résumé automatique.

Mots-clés Recherche d'information, contextualisation, concepts implicites, modélisation thématique probabiliste, sources d'information, retour de pertinence simulé, modèles de pertinence, TREC.

Abstract

When searching for information within knowledge bases or document collections, humans use an information retrieval system (IRS). So that it can retrieve documents containing relevant information, users have to provide the IRS with a representation of their information need. Nowadays, this representation of the information need is composed of a small set of keywords often referred to as the « query ». A few words may however not be sufficient to accurately and effectively represent the complete cognitive state of a human with respect to her initial information need. A query may not contain sufficient information if the user is searching for some topic in which she is not confident at all. Hence, without some kind of context, the IRS could simply miss some nuances or details that the user did not – or could not – provide in query.

In this thesis, we explore and propose various statistic, automatic and unsupervised methods for representing the topical context of the query. More specifically, we aim to identify the latent concepts of a query without involving the user in the process nor requiring explicit feedback. We experiment using and combining several general information sources representing the main types of information we deal with on a daily basis while browsing the Web. We also leverage probabilistic topic models (such as Latent Dirichlet Allocation) in a pseudo-relevance feedback setting. Besides, we propose a method allowing to jointly estimate the number of latent concepts of a query and the set of pseudo-relevant feedback documents which is the most suitable to model these concepts. We evaluate our approaches using four main large TREC test collections. In the appendix of this thesis, we also propose an approach for contextualizing short messages which leverages both information retrieval and automatic summarization techniques.

Keywords Information retrieval, contextualization, latent concepts, probabilistic topic modeling, information sources, pseudo-relevance feedback, relevance models, TREC.

Table des matières

Chapitre 1

Introduction

Sommaire

1.1 La Recherche d'Information

Nous vivons dans une société moderne et ultra-connectée, dans laquelle un an équivaut au développement d'une nouvelle génération de smartphones ou de tablettes tactiles. Plus de 5 milliards de téléphones mobiles sont actuellement en utilisation, pour lesquels plus d'un milliard ½ sont des smartphones. Selon les prédictions, le trafic internet global devrait même provenir à 30% des appareils mobiles d'ici la fin de l'année 2014 [1]. Dans ce contexte où l'accès à internet est quasi-permanent, accéder rapidement et surtout efficacement à l'information est un défi majeur. Chaque jour, plus de 5 milliards de requêtes sont soumises au moteur de recherche de Google [2], qui totalise environ 67% du trafic global des moteurs de recherche [3] avec Microsoft (Bing) et Yahoo ! comme premiers concurrents. Ces différents acteurs du Web sont à présent entrés dans la culture collective et il est devenu naturel de poser nos questions aux moteurs de recherche sous forme de mots-clés, tout en attendant une liste de documents ordonnés dont une grande partie devraient contenir la ou les réponses.

Nous sommes entourés par des quantités astronomiques d'informations présentes sous la forme de pages web, de documents vidéo, audio, mais également de journaux ou de publicités. Ces connaissances sont majoritairement compilées par les moteurs de

1. http://readwrite.com/2013/05/29/huge-potential-only-15-of-global-internet-traffic-is-mobile
2. http://www.statisticbrain.com/google-searches
3. http://www.comscore.com/Insights/Press_Releases/2013/8/comScore_Releases_July_2013_U.S._Search_Engine_Rankings

recherche commerciaux cités précédemment, qui sont devenus des points d'entrée du Web. Nos appareils électroniques connectés à internet font office de terminaux nous reliant à une « infosphère », abstraite, représentant les informations accumulées, indexées et accessibles. Du point de vue de l'humain, rechercher de l'information revient à formuler son besoin d'information le plus précisément possible sous forme de mots-clés afin que le moteur de recherche puisse « comprendre » ce besoin et proposer à l'utilisateur une liste de documents ou de réponses. Du point de vue du système, le défi de la recherche d'information réside justement dans la compréhension du besoin d'information qui n'est exprimé que sous la forme d'une requête composée d'un nombre réduit de mots. Idéalement, pour proposer une réponse parfaite à l'utilisateur, le système devrait pouvoir connaître ses pensées, son niveau d'éducation par rapport au besoin d'information [4] ou encore ses connaissances dans des thématiques connexes.

D'une façon très générale, un système de Recherche d'Information (RI) prend en entrée une requête formulée par un utilisateur puis va récupérer des données au sein d'une collection préalablement indexée. Historiquement, la RI fait principalement référence à la recherche documentaire : les données récupérées sont des documents entiers qui contiennent des informations que le système a jugées comme pertinentes par rapport à la requête (Harman, 2011). Le système cherche les documents qui contiennent les mots-clés, afin de fournir à l'utilisateur une liste de documents ordonnés en fonction de leur pertinence estimée par rapport à la requête. De nos jours, la RI n'est néanmoins plus réduite à cette recherche documentaire et se rapproche de l'accès à l'information en général. Parmi ces différentes aspects, nous pouvons citer, entre autres, la recherche de passages (uniquement certaines parties des documents) (Kaszkiel et Zobel, 1997; Fuhr et al., 2008), la génération de mini-phrases décrivant les documents dans la liste de résultats (ou *snippets*) (Huang et al., 2008) ou le résumé multi-documents orienté par une requête (Boudin et Torres Moreno, 2007). Bien que tentant de s'abstraire des notions de « document » pour se diriger vers des notions (plus abstraites) d'« information », les travaux traitant les aspects précédents se basent très principalement sur des documents. La recherche précise et efficace de documents pertinents reste ainsi une pièce centrale de la RI en général. Nous nous penchons dans cette thèse sur des problématiques de contextualisation thématique dans le but d'améliorer la recherche documentaire.

1.2 Problématiques

Nous partons du constat qu'un besoin d'information complet peut être trop complexe pour être exprimé en quelques mots, ou encore que 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

4. Un scientifique cherchant des informations sur son champ de recherche est par exemple moins susceptible d'avoir besoin d'une introduction à son domaine, contrairement à un étudiant ou même à un enfant.

sur une thématique pour laquelle il n'a pas de connaissances. Des mots très spécifiques à la thématique de la recherche peuvent par exemple manquer. Cette différence de vocabulaire (*vocabulary mismatch*) a été identifiée très tôt par Furnas et al. (1987) comme étant un problème majeur touchant les systèmes automatiques interagissant avec des humains. 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, et récupérer automatiquement des documents pertinents à la requête peut ainsi être difficile. Ce contexte peut prendre la forme d'un modèle des intérêts de l'utilisateur basé sur son historique personnel (ou ses interactions sociales) (Finkelstein et al., 2002; White et al., 2010), ou peut être composé d'éléments extraits de documents similaires représentant les thèmes de la recherche (White et al., 2009; Kaptein et Kamps, 2011). Nous nous concentrons dans cette thèse sur cette seconde catégorie d'approches.

Les méthodes traditionnelles que l'on peut voir dans la littérature forment généralement cet ensemble de documents liés aux thématiques de la recherche en utilisant la requête originale. Le retour de pertinence classique (Koenemann et Belkin, 1996) présente quant à lui les N premiers documents à l'utilisateur, qui peut alors indiquer quels documents sont pertinents par rapport à son besoin d'information, permettant alors au système d'avoir une représentation plus précise du contexte thématique. Dans un monde où les recherches sur le Web ne prennent que quelques millisecondes, les utilisateurs sont parfois hésitants ou réticents à l'idée de donner ce retour de pertinence. Cette étape peut en effet être longue et fastidieuse, et elle ne correspond pas aux standards fixés par les moteurs de recherche Web commerciaux actuels. Il est ainsi nécessaire de proposer des méthodes automatiques capables d'achever une précision comparable à la méthode manuelle. L'automatisation du retour de pertinence est connu sous le nom de retour de pertinence simulé (ou *pseudo relevance feedback*). Il est dit « simulé » car le système fait l'hypothèse que les N premiers documents renvoyés par le système sont pertinents, ce qui n'est pas forcément le cas : ils sont considérés comme *pseudo-pertinents*.

Cet ensemble de documents peut ainsi être vu comme une représentation concrète de la notion abstraite du contexte thématique de la requête : tous ces documents traitent de différents sujets et portent sur des thématiques variées qui ont toutes un lien plus ou moins important avec la requête. Certaines thématiques peuvent par exemple avoir un lien avec la requête dont le poids est nul, il s'agît alors de thématiques non-pertinentes. Nous pouvons alors imaginer de façon abstraite le contexte thématique de la requête comme une couverture complète de toutes les informations liées à la requête et au besoin d'information dont elle résulte, où ces informations peuvent avoir des poids différents. Nous introduisons dans cette thèse plusieurs méthodes permettant de modéliser ce contexte thématique en se basant sur les modèles de pertinence (Lavrenko et Croft, 2001; Zhai et Lafferty, 2001). Nous cherchons plus précisément à répondre à trois questions de recherche principales :

– est-il possible d'améliorer la représentation du contexte thématique d'une requête en utilisant plusieurs sources d'information de natures différentes ? Des premiers travaux (Diaz et Metzler, 2006) suggèrent qu'une combinaison de sources externes permet d'améliorer les performances de recherche documentaire, mais les sources utilisées sont peu diverses. Idéalement, nous voudrions simuler l'« infosphère »

qui nous entoure en utilisant un plusieurs sources d'information de natures très différentes mais complémentaires,
- est-il possible de quantifier et de modéliser avec précision les concepts implicites d'une requête ? Étant donné que la quantité d'informations accessibles augmente de façon spectaculaire chaque jour, nous voudrions nous affranchir des ontologies et autres ressources structurées proposant des hiérarchies de concepts manuellement définis (utilisées par exemple dans les débuts du Web sémantique (Vallet et al., 2005)),
- quels sont les effets de tels concepts sur les performances d'un système de RI ? Sont-ils efficaces et représentent-ils efficacement le contexte thématique de la requête ?

Alors que ces questions semblent délimitées, nous les traitons comme un tout dans cette thèse et explorons l'influence de chacun des composants que nous étudions indépendamment des autres.

1.3 Plan de la thèse

Nous présentons notre travail de la manière suivante. Le chapitre 2 constitue une introduction aux notions de Recherche d'Information et aux méthodes d'expérimentations utilisées de nos jours pour évaluer et comparer des systèmes de RI. Nous proposons une revue historique des évolutions de la culture expérimentale en RI puis présentons les jeux de données que nous avons utilisés tout au long de cette thèse. La suite de ce document est séparée en trois parties consacrées au développement et à l'évaluation d'approches permettant de modéliser le contexte thématique de la requête et de récupérer des documents. Le chapitre 3 présente une méthode originale d'estimation des modèles de pertinence permettant de prendre en compte des séquences de mots sans supervision. Nous explorons l'influence de plusieurs sources d'informations et conduisons un grand nombre d'expériences visant à déterminer la qualité et la robustesse de ces modèles. Ce chapitre est une extension du travail que nous avons publié dans (Deveaud et al., 2013b). Dans le chapitre 4, nous introduisons une approche entièrement non-supervisée et reposant sur un faible nombre de paramètres quantifiant et modélisant les concepts implicites d'une requête. Nous utilisons pour cela un algorithme de modélisation thématique *sur* des documents pseudo-pertinents et évaluons la cohérence des concepts générés. Nous discutons également de la corrélation de notre méthode par rapport à un algorithme de modélisation thématique hiérarchique de l'état-de-l'art. Nous nous attardons dans le chapitre 5 à évaluer l'apport des concepts modélisés par la méthode précédente pour la recherche documentaire. Nous introduisons un nouveau modèle de pertinence et explorons ses performances dans divers cas et en faisant également varier ses paramètres. Les chapitres 4 et 5 sont des versions étendues de travaux publiés dans (Deveaud et al., 2013c) et (Deveaud et al., 2013a). Pour finir, le chapitre 6 clôt cette thèse et récapitule nos principales observations, tout en proposant plusieurs pistes d'améliorations et de poursuite des travaux de recherche.

Chapitre 2

Méthodologie expérimentale

Sommaire

Dans le chapitre précédent, nous avons donné un aperçu des différentes questions et hypothèses qui sous-tendent cette thèse. Le domaine de la Recherche d'Information (RI) se construit depuis maintenant plusieurs décennies sur une culture de la validation d'hypothèse par l'expérimentation. Au centre de ces évaluations se trouvent les notions de pertinence et de mesures d'évaluation, indispensables à la compréhension du comportement d'un système de RI. Dans ce chapitre, nous détaillons ces notions et dressons un portrait des différentes campagnes d'évaluation en activité. Nous décrivons enfin les collections de test et les sources d'information que nous avons utilisées tout au long de cette thèse.

2.1 Pertinence

> Pertinence :
> a) qualité de ce qui est pertinent, logique, parfaitement approprié,
> —— Dictionnaire Larousse (2012).
>
> b) la capacité (d'un système de recherche d'information) à récupérer des éléments qui satisfont les besoins d'un utilisateur.
> —— Dictionnaire Merriam-Webster (2012).

L'objectif principal de la Recherche d'Information (RI) est de récupérer tous les documents pertinents, et en même temps de récupérer aussi peu de documents non pertinents que possible (Rijsbergen, 1979). La notion de pertinence est ainsi centrale dans l'évaluation des systèmes de RI mais, paradoxalement, c'est une notion *subjective*. Ce paradoxe vient du fait qu'un système de RI cherche à estimer *objectivement* (Borlund, 2003; Hjørland, 2010) la pertinence de documents par rapport à une requête, alors que cette pertinence est en réalité connue seulement de l'utilisateur qui soumet la requête. Différents utilisateurs peuvent ainsi avoir des opinions différentes sur la pertinence (ou sur la non-pertinence) de certains documents pour une même requête (Rijsbergen, 1979; Schamber et al., 1990). Schamber et al. (1990) et Harter (1992) évoquent notamment une notion de pertinence « situationnelle », ou contextuelle, et l'apparente à un concept cognitif multidimensionnel dépendant très principalement du besoin d'information de l'utilisateur ainsi que de sa façon d'appréhender les informations qui lui sont présentées. De nombreuses études se sont penchées sur l'évaluation interactive des systèmes de RI (Kelly, 2009), prenant en compte les changements de comportement des utilisateurs au cours de leurs recherches afin de modéliser cette pertinence situationnelle. Comme nous pouvons le voir sur la figure 2.1, plusieurs degrés d'interactivité avec l'utilisateur ont été étudiés, allant de l'étude orientée purement utilisateur à l'étude orientée système. Les expériences menées dans cette thèse se situent à gauche de cette figure. Plus spécifiquement, nous utilisons des requêtes et des collections de documents dédiées à la recherche et accessibles sur demande (et parfois payantes). La pertinence des documents par rapport aux requêtes (pertinence thématique) est jugée de façon « objective » par des assesseurs. Ces jugements objectifs permettent ainsi de quantifier la quantité d'information contenue dans un document par rapport à une requête. Nous nous concentrons précisément dans cette thèse sur des problématiques de modélisation et d'identification des différentes informations pouvant être liées à une recherche (Kaptein et Kamps, 2011), ce que ce protocole d'évaluation orienté système nous permet précisément d'effectuer.

La notion d'« information » est elle-même très proche de la notion de pertinence et, dans le cadre de l'évaluation des systèmes de RI, ces deux notions tendent à être similaires. Bates (2006) définit plus précisément deux types différents d'informations qui, tout comme la pertinence, peuvent être objectifs ou subjectifs. Cette double définition a notamment été la source de nombreux débats (Bates, 2008, 2011; Hjørland, 2009) portant principalement sur le fait qu'une information ne pourrait être que subjective, à l'opposé des deux définitions avancées précédemment. Dans le cadre d'une tâche de

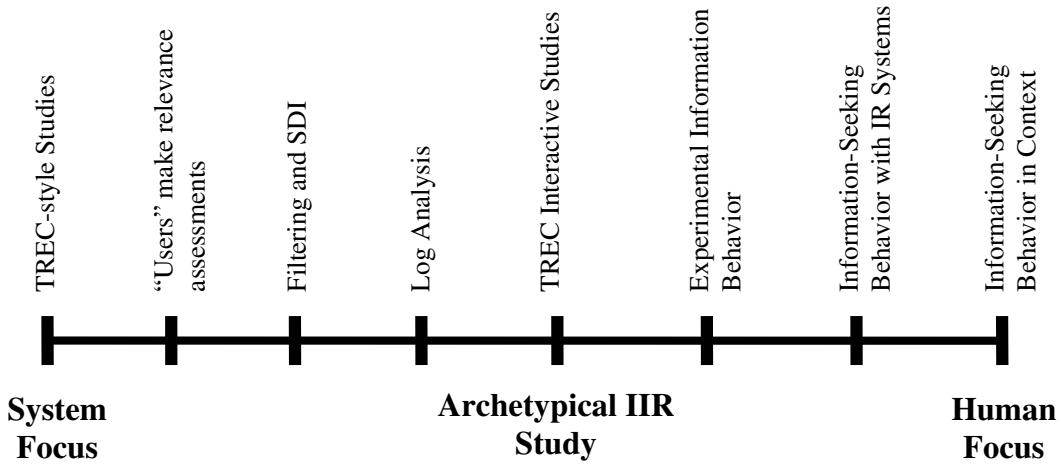

FIGURE 2.1 – *Graphique issu de l'article de Kelly (2009) représentant toutes les sous-tâches (et les sous-types d'évaluation).*

RI, l'information est subjective, ou conditionnée par la requête. Néanmoins si l'on se place au niveau du document, on peut quantifier objectivement la quantité d'information (liée à la requête) qu'il contient. Dans ce contexte, la pertinence s'applique à un document (qui représente une unité textuelle, telle qu'une page web, un passage, une phrase, ...) et définit à quel point le document contient des informations importantes par rapport à une requête. Dans les premières évaluations de TREC (voir section 2.2.1), un document pouvait ainsi être considéré comme pertinent dans son ensemble, même si une seule de ses phrases était pertinente (Harman, 2011).

Les notions de pertinence et d'information sont étroitement liées et se confondent parfois lors de l'évaluation de systèmes de Recherche d'Information. Nous détaillons dans la prochaine section les méthodes expérimentales employées de nos jours dans les différentes campagnes d'évaluation ainsi que leurs origines.

2.2 Évaluation

L'évaluation est une tradition qui est au centre de la Recherche d'Information en tant que domaine de recherche depuis maintenant plus de vingt ans[1]. L'arrivée des campagnes d'évaluation, des expériences reproductibles et des données communes et partagées a influencé la culture expérimentale d'une génération de chercheurs. Nous détaillons dans les sections suivantes le paradigme commun à ces campagnes d'évaluation, le lien avec les jugements de pertinence, composant essentiel de l'évaluation, ainsi que les mesures d'évaluation standard utilisées dans cette thèse.

1. La première édition de TREC a eu lieu en 1992.

2.2.1 Paradigme d'évaluation de Cranfield

Évaluer et comparer les performances de différents systèmes d'indexation automatique et de recherche d'information est une problématique qui a émergé dans les années 1950 (Robertson, 2008), et dont le premier cadre expérimental a été défini dans les années 1960 avec les expériences menées à Cranfield (Cleverdon, 1962; Cleverdon et al., 1962). La collection de documents était alors très réduite, et les requêtes étaient générées directement à partir des documents. Chaque document était alors jugé pertinent ou non par rapport à chaque requête. Avec ces informations à disposition, il était alors possible de comparer les résultats fournis par un système automatique avec la référence complète ainsi annotée.

Ce « paradigme d'évaluation de Cranfield » (Voorhees, 2002) a gagné en popularité principalement pour les possibilités de reproductibilité qu'il offre, et a perduré au sein de la communauté de Recherche d'Information. De nombreuses campagnes d'évaluation telles que TREC [2], INEX [3], CLEF [4], FIRE [5] et NTCIR [6] se sont mises en place à l'initiative de quelques chercheurs telles que Donna Harman et Ellen Voorhees (Harman, 1992a; Harman et Voorhees, 2006), avec pour but de construire des collections comprenant des requêtes, des documents et des jugements de pertinence, pouvant ainsi être réutilisées par d'autres chercheurs. Nous avons vu dans la section précédente que la pertinence d'un document par rapport à une requête pouvait dépendre de plusieurs paramètres, dont l'utilisateur (Kamps et al., 2009). La mise en place d'une méthodologie pour le développement d'environnements d'évaluation initiée à Cranfield permet notamment d'essayer de s'abstraire de ces différences individuelles. Tague-Sutcliffe (1996) définit six éléments qui composent un processus de recherche d'information et qui se retrouvent dans ces campagnes :

- un ensemble de documents à renvoyer (ou plus communément « collection cible », ou encore simplement « collection »),
- un algorithme de recherche documentaire,
- un besoin d'information d'un utilisateur,
- une expression de ce besoin d'information (généralement sous la forme de mots-clés, autrement dit la « requête »),
- une liste de documents renvoyés, et
- des jugements de pertinence.

Un système de Recherche d'Information prend donc en compte une représentation d'un besoin d'information en entrée, et produit une liste de documents ordonnée par ordre décroissant en fonction de leur pertinence estimée. L'évaluation d'un tel système reflète dans ce cas à quel point il a la capacité de satisfaire l'utilisateur courant, ainsi que tous les utilisateurs passés et à venir (avec des besoins d'information, et donc des requêtes, variés). Dans la littérature, on nomme généralement « collection de test » l'ensemble formé par la collection de documents, les requêtes et les jugements de pertinence.

2. http://trec.nist.gov/
3. http://inex.mmci.uni-saarland.de/
4. http://www.clef-initiative.eu/
5. http://www.isical.ac.in/~clia/
6. http://research.nii.ac.jp/ntcir/index-en.html

2.2.2 Jugements de pertinence et échantillonnage

La création de jugements de pertinence était réalisable pour une collection de documents limitée comme celle employée pour les premières expériences de Cranfield (1 400 documents et 225 requêtes), mais ce n'est clairement pas un scénario réaliste dès que le nombre de documents à juger augmente trop (742 611 documents et 100 requêtes dans la première collection de TREC (Harman, 1992b), ce qui reviendrait à juger plus de 74 millions de paires requête-document). La solution adoptée à l'époque par les organisateurs de TREC, et qui est toujours utilisée de nos jours dans différentes campagnes d'évaluation, a été de sélectionner les N premiers documents renvoyés par les systèmes des participants et de ne construire des jugements de pertinence que pour ces documents-ci (Harman, 1992b). Cette méthode d'échantillonnage (ou *pooling*) avait été initialement introduite par Jones et al. (1975), et des études ont montré (dans le cadre de TREC) son efficacité à produire des jugements de pertinence statistiquement équivalents à des jugements complets (Zobel, 1998). De plus, les systèmes ne contribuant pas à l'échantillon peuvent eux-aussi être évalués équitablement (Zobel, 1998). L'intervention humaine étant limitée, les coûts de création d'une collection de test sont donc très largement diminués, tout en permettant à d'autres chercheurs de disposer des mêmes données pour évaluer leurs algorithmes et pour vérifier si leur découvertes se généralisent (Jones, 1981).

Par ailleurs, l'avènement de la sous-traitance de masses (ou *crowdsourcing*) (Howe, 2008), notamment à travers le service Mechanical Turk d'Amazon [7], donne une alternative pour la collecte de grands nombres de jugements de pertinence à faible coût (Alonso et al., 2008; Alonso et Mizzaro, 2009). Ce service permet aux chercheurs (entre autres) de définir des tâches, idéalement simples et rapides, qui peuvent être exécutées par des internautes (aussi appelés *Turkers*) en échange de micro-paiements. Kazai et al. (2011) montrent que les jugements de pertinence produits par les *Turkers* peuvent permettre de reproduire les classements de systèmes de RI obtenus avec des jugements obtenus de manière classique, sous réserve de concevoir des tâches et des interfaces de façon à ne pas biaiser le jugement des *Turkers*. Bien que cette alternative semble viable, nous utilisons dans cette thèse les méthodes traditionnelles pour des raisons de reproductibilité, de comparaison et également de coût. De plus, à l'heure où nous écrivons cette thèse, l'utilisation de services de *crowdsourcing* (et plus particulièrement Mechanical Turk) va à l'encontre du droit européen sur la propriété intellectuelle (Sagot et al., 2011).

2.2.3 Mesures d'évaluation

Rechercher de l'information peut impliquer d'effectuer plusieurs tâches différentes, où chacune modélise le comportement de l'utilisateur de façon différente. Dans cette thèse, nous nous intéressons au cas d'un utilisateur voulant acquérir des informations sur un sujet ou une thématique précise. Ce type bien particulier de tâche porte souvent la dénomination de *ad hoc* dans la littérature. On peut imaginer d'autres cas où

7. https://www.mturk.com

un utilisateur souhaite retrouver une page web spécifique ou une entrée dans une encyclopédie (Balog et al., 2009) (recherche de page principale), ou encore un utilisateur cherchant des personnes expertes sur un sujet (Bailey et al., 2007) (recherche d'entités). Sans chercher précisément un objet, certains utilisateurs peuvent vouloir des réponses à leurs questions (Voorhees et Tice, 1999; Moriceau et al., 2009) (question-réponse), ou encore comprendre la teneur de certains messages courts et ambigus (SanJuan et al., 2012) (contextualisation).

À chacune de ces tâches correspondent des mesures d'évaluation précises permettant de quantifier la qualité d'un système répondant à ces problèmes. Par exemple, dans le cas où l'on évalue un système devant renvoyer une page principale sachant une requête, une mesure convenable devrait récompenser les systèmes qui placent la bonne page en première position et pénaliser les systèmes qui ne le font pas. Les tâches de recherche d'information évoluent avec le temps et avec les changements de comportement des utilisateurs. La définition et l'analyse de nouvelles mesures permettent de faire face à ces nouveaux comportements ainsi qu'aux nouvelles tâches et problématiques émergentes (telles que la recherche d'entités par exemple). Néanmoins, à l'origine, les systèmes de RI renvoyaient des listes de documents (non ordonnées) et pouvaient être ainsi évalués en calculant des mesures de précision et de rappel (van Rijsbergen, 1979; Manning et al., 2008).

	Document pertinent	Document non-pertinent
Document renvoyé	vrai positif (vp)	faux positif (fp)
Document non renvoyé	faux négatif (fn)	vrai négatif (vn)

TABLE 2.1 – *Matrice de confusion.*

Ces mesures ont largement été utilisées en classification et peuvent être aisément transposées à la RI[8]. Soit un ensemble de documents renvoyés en réponse à une requête utilisateur, la fraction des documents renvoyés qui sont également pertinents est exprimée par la précision ; le rappel représente quant à lui la fraction des documents pertinents qui sont effectivement renvoyés par le système. En utilisant les notations introduite par la matrice de confusion présentée dans le tableau 2.1, on peut exprimer la précision par :

$$\text{precision} = \frac{vp}{vp + fp} = P(\text{pertinent}|\text{renvoyé}) \tag{2.1}$$

et le rappel par :

$$\text{rappel} = \frac{vp}{vp + fn} = P(\text{renvoyé}|\text{pertinent}) \tag{2.2}$$

8. Il est à noter que ce n'est pas vrai pour toutes les mesures utilisées en classification. Même si une tâche *ad hoc* de RI peut être vue comme une tâche de classification à deux classes (pertinent contre non-pertinent), les documents non-pertinents par rapport à une requête sont très largement majoritaires (> 99%) au sein d'une collection de documents de taille importante. En utilisant une mesure d'*accuracy* ($= \frac{vp + vn}{vp + fp + vn + fn}$, suivant la notation introduite dans le tableau 2.1), il serait facile d'atteindre d'excellents résultats avec un système prédisant qu'un document est non-pertinent à chaque fois.

Ces deux mesures sont complémentaires mais ne reflètent pas forcément exactement les attentes qu'un utilisateur peut avoir d'un moteur de recherche. Par exemple, renvoyer l'intégralité des documents de la collection permettra d'obtenir un rappel de 1, mais la précision sera à ce moment très faible. Dans le cas d'une recherche sur le Web impliquant des dizaines de milliards de documents [9], on sera plutôt intéressé par la capacité du système à ramener un forte proportion de documents pertinents au sein d'une liste réduite (de 10 documents par exemple), tandis qu'une recherche experte (de brevets par exemple) va nécessiter un système étant capable de renvoyer *tous* les documents pertinents peu importe leur rang (Lupu, 2013).

Dans notre cas, l'utilisateur se voit présenter une liste *ordonnée* de documents, où le premier document est implicitement le plus pertinent (selon le système de RI). Il est ainsi commun de calculer certaines mesures d'évaluation (comme la précision) sur les k premiers documents. Ainsi la précision à 10 documents, P@10, représente la précision de l'ensemble formé par les 10 premiers documents renvoyés par le système de RI.

Une des mesures très largement employée est la moyenne des précisions aux rangs où se trouvent les documents pertinents par rapport à une requête (abrégée en AP pour *average precision*). Plus formellement, soit $D = \{d_1, ..., d_n\}$ une liste de documents renvoyés pour une requête Q donnée, la précision moyenne de cette liste se définit comme :

$$AP = \frac{1}{|R|} \sum_{k=1}^{n} P@k \times rel(d_k) \tag{2.3}$$

où $|R|$ représente le nombre total de documents jugés pertinents, n est le nombre de documents renvoyés par le système. $rel(d_k)$ est un indicateur de la pertinence du document au rang k ; il est égal à 1 si d_k est pertinent, à 0 sinon.

La mesure MAP (*mean average precision*) s'est imposée au sein de la communauté TREC comme une mesure standard permettant d'évaluer d'un coup les performances d'un système sur une tâche donnée. Celle-ci utilise les AP calculées pour un ensemble de requêtes, traditionnellement 50, et en fait la moyenne :

$$MAP = \frac{1}{|Q|} \sum_{i=1}^{|Q|} AP(Q_i) \tag{2.4}$$

avec $|Q|$ étant le nombre de requêtes. Les mesures AP et MAP sont connues pour leur stabilité et leur fort *pouvoir discriminatif* par rapport à d'autres mesures (Buckley et Voorhees, 2000; Sakai, 2006) ; en d'autres mots, elles sont plus efficaces pour identifier les différences entre deux systèmes de RI différents et les départager. La mesure MAP a été la mesure d'évaluation officielle pour de nombreuses tâches de TREC, y compris celles dont nous utilisons les collections (voir section 2.3) ; c'est logiquement la mesure d'évaluation principale de cette thèse. Nous utilisons comme mesures secondaires la précision et la nDCG à 10 ou 20 documents, ce qui correspond à l'affichage d'une ou deux pages de résultats dans le cadre d'un moteur de recherche standard. L'acronyme

9. http://www.worldwidewebsize.com/

nDCG, pour *normalized discounted cumulative gain*, est utilisé pour désigner une mesure prenant en compte des jugements de pertinence gradués (Järvelin et Kekäläinen, 2002) et pénalisant les documents qui apparaissent trop bas dans la liste renvoyée par le système. Certains documents peuvent en effet avoir des niveaux de pertinence plus importants que d'autres ; on pourrait par exemple imaginer une échelle de pertinence comprenant 0 : non-pertinent, 1 : pertinent, 2 : central, 3 : vital. De plus, les utilisateurs explorent naturellement les documents renvoyés en premier par le système avant de descendre dans la liste, des documents pertinents apparaissant trop bas sont donc susceptibles de ne pas être vus. L'objectif des mesures graduées est ainsi de pénaliser un système qui classerait un document vital après un document qui ne serait que pertinent, tout en favorisant les systèmes qui renvoient les documents pertinents dans les tous premiers rangs. Ce type de mesures est notamment populaire pour évaluer les performances des moteurs de recherche sur le Web (Chapelle et al., 2009).

Récemment, la nécessité d'évaluer la diversité d'une liste de documents est apparue, tout en prenant en compte leur pertinence, avec pour but de répondre notamment au problème des requêtes ambiguës ou traitant de thèmes larges. Un exemple classique est la requête « java » : elle peut être synonyme de café en anglais, ou peut faire référence à une danse ou au célèbre langage de programmation. Dans ce cas précis, un système de RI privilégiant la diversité devrait proposer à l'utilisateur des documents traitant de ces différents sujets dans les premiers classements, et non uniquement des documents traitant du langage de programmation (même s'ils sont tous pertinents). Chaque requête est donc représentée par un ensemble de « sous-thématiques » inconnues du système avant la recherche documentaire. La pertinence des documents est alors jugée pour chacune des sous-thématiques comme si elles étaient des besoins d'information à part entière. La première mesure d'évaluation à prendre en compte ces sous-thématiques a été α-nDCG (Clarke et al., 2008, 2011a), puis Agrawal et al. (2009) ont introduit une famille de mesures *intent-aware* (IA) intégrant naturellement ces jugements multiples. Comme nous le verrons dans la section 2.3 ci-dessous, la collection de test de la tâche Web de TREC (2009-2012) est une des seules à ce jour à proposer des jugements permettant des calculer des performances de diversité d'un système de RI. Quand nous utilisons cette collection dans cette thèse, nous reportons également les performances de diversité en fonction des mesures officielles utilisées dans cette tâche, à savoir ERR-IA@20 [10] et α-nDCG@20.

2.3 Collections de documents

Nous décrivons dans cette section les différentes collections de test que nous utilisons pour évaluer les contributions de cette thèse. Nous avons utilisé Indri [11], un système d'indexation et de recherche d'information libre et open-source, pour indexer ces collections et chercher les documents. Les mêmes paramètres ont été utilisés dans tous

10. Cette mesure est une version *intent-aware* de la mesure ERR (Chapelle et al., 2009), déjà populaire pour évaluer la recherche de pages Web. Elle est ainsi complémentaire à la mesure α-nDCG.
11. http://www.lemurproject.org/indri

les cas : les mots ont été légèrement racinisés par l'algorithme standard de Krovetz (1993), et les mots outils présents dans la liste fournie avec Indri ont été supprimés. Toutes ces collections contiennent des documents en langue anglaise. Durant toutes nos expérimentations nous n'utilisons que la formulation par mots-clés des requêtes (aussi connue sous la dénomination de *title queries*), et nous ignorons les formulations développées pouvant être composées de plusieurs phrases (ou *description* ou *narrative queries*).

Nous détaillons ici les caractéristiques de ces collections ainsi que le contexte dans lequel elles ont été créées et les tâches auxquelles elles sont dédiées. Les tableaux 2.2 et 2.3 proposent quant à eux quelques statistiques sur ces collections.

Nom	# documents	taille de l'index	# mots uniques	# total de mots	μ
WT10g	1 692 096	9,2 Go	5 437 563	1 043 993 839	617
Robust04	528 155	2 Go	675 713	253 367 449	480
GOV2	25 205 179	202 Go	39 286 722	23 623 611 729	937
ClueWeb09-B	50 220 423	583 Go	87 330 765	40 416 831 010	805

TABLE 2.2 – *Résumé des collections de test de TREC utilisées pour nos évaluations. μ indique la longueur moyenne des documents, en nombre de mots.*

			Par requête		
Nom	**requêtes utilisées**	**# docs. pertinents**	**moyen**	**min.**	**max.**
WT10g	451-550	5 980	59,8	1	519
Robust04	301-450, 601-700	17 412	69,65	3	448
GOV2	701-850	26 917	179,45	4	617
ClueWeb09-B	50-150	14 842	98,95	1	314

TABLE 2.3 – *Statistiques sur les requêtes et les documents jugés pertinents pour les collections utilisées dans cette thèse.*

2.3.1 TREC Web 2000-2001

La première collection (WT10g) a servi de support à la tâche de recherche Web de TREC pour les années 2000 et 2001 (Hawking, 2000). Il s'agit d'un ensemble de pages Web récupérées en 2000 ne contenant que des documents en langue anglaise. Nous utilisons ici les requêtes des années 2000 et 2001 (50 pour chaque année). Ce sont des requêtes soumises par des utilisateurs réels, extraites à partir des historiques de requêtes du moteur de recherche eXcite (Hawking, 2000; Bailey et al., 2003). Les jugements de pertinence ont été construits en suivant la méthodologie traditionnelle de TREC (section 2.2), à savoir en jugeant des échantillons de documents renvoyés par les participants de la tâche.

2.3.2 TREC Robust 2004

La seconde collection de documents est celle de la tâche Robust de TREC 2004 (Robust04), et est composée d'articles de presse provenant de divers journaux dont le *Financial Times*, le *Federal Register*, le *Los Angeles Times* et le *Foreign Broadcast Information Service* (service de diffusion d'information étrangère), couvrant une période allant de 1989 à 1996. Ces différents corpus journalistiques se trouvent sur les disques 4 et 5 de TREC (en enlevant la partie Congressional Record). Sur les 250 requêtes de cette collection, 200 ont été reprises des précédentes tâches *ad hoc* de TREC car elles avaient été jugées difficiles (Voorhees, 2005) (i.e. ce sont des requêtes pour lesquelles les systèmes participant à TREC n'ont pas eu de bons résultats). Les 50 dernières requêtes ont quant à elles été développées spécifiquement pour la tâche. Cette collection est spécifiquement reconnue pour sa stabilité et la bonne qualité de ses jugements.

2.3.3 TREC Terabyte 2004-2006

La troisième collection tire son nom (GOV2) des noms de domaines des sites internet du gouvernement américain, qui contiennent tous un `.gov` dans leur URL. Elle est constituée d'une large partie de ce domaine et contient le texte HTML des pages web, ainsi que les transcriptions de documents PDF, Word et postscript (Clarke et al., 2004). Elle a été utilisée dans les tâches TREC Terabyte, Million Query et Relevance Feedback, mais nous utilisons dans cette thèse uniquement les requêtes de la tâche Terabyte. Celle-ci commença en 2004 et finit en 2006, et fut la première à utiliser une collection d'une taille aussi importante (d'où son nom) (Büttcher et al., 2006) : son but était de développer une méthodologie d'évaluation pour les collections de documents de l'échelle du Téraoctet. Nous utilisons dans cette thèse toutes les requêtes des trois années.

2.3.4 TREC Web 2010-2011

Pour finir, le ClueWeb09 [12] est la deuxième plus grande collection de pages web mise à disposition de la communauté de RI à l'heure où nous écrivons cette thèse (après le ClueWeb12, dont la distribution a commencé en janvier 2013). Cette collection a été utilisée dans plusieurs tâches de TREC comme par exemple les tâches Web (de 2009 à 2012) (Clarke et al., 2011b), 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 en anglais, tandis que l'intégralité de la collection contient plus d'un milliard de documents, dont 500 millions en anglais. La méthodologie d'évaluation pour les requêtes de la tâche Web de 2009 a suivi un processus inhabituel se basant sur des groupements de documents superficiels et de tailles réduites (Carterette et al., 2006). Les jugements de pertinence produits pour ces requêtes (possédant les identifiants allant de 1 à 50) ne sont donc pas compatibles avec les traditionnelles mesures *ad hoc*, c'est

12. http://boston.lti.cs.cmu.edu/clueweb09/

pourquoi nous considérons dans cette thèse uniquement les requêtes des années 2010 et 2011.

2.4 Sources d'information

En plus des collections de test standards que nous avons présentées dans la section précédente, nous utilisons différentes sources d'information dans le cadre de nos expérimentations. Nous employons aussi le terme de « collection externe » ou de « ressource » pour nous référer à ces sources d'information, en opposition à la collection de documents dans laquelle le système de RI va chercher des documents, ou « collection cible ». Nous nous en servons afin de récupérer des informations supplémentaires ou complémentaires sur le contexte thématique de la requête, nous considérons donc ici des sources de grande taille et contenant des documents de natures différentes. De la même façon que pour les collections de documents, nous détaillons leurs caractéristiques et le contexte de leur acquisition.

Ressource	# documents	taille de l'index	# mots uniques	# total de mots	μ
NYT	1 855 658	11 Go	1 086 233	1 378 897 246	743
Wiki	3 406 520	12 Go	7 419 901	1 143 840 781	336
GW	4 111 240	12 Go	1 288 389	1 397 727 483	340
Web	29 038 220	336 Go	33 314 740	22 814 465 842	786

TABLE 2.4 – *Récapitulatif des quatre sources d'information générales utilisées. μ représente la longueur moyenne des documents.*

2.4.1 Wikipédia

Notre première source d'information est Wikipédia, l'encyclopédie en ligne. Les avantages pouvant être apportés par une telle ressource sont multiples et sont principalement liés au fait que les articles sont entièrement construits manuellement. Derrière chaque phrase se trouve de la connaissance que des humains ont pris le temps d'acquérir, de retranscrire et surtout de corriger afin d'en améliorer en permanence la qualité (Medelyan et al., 2009). Cette qualité a été exploitée de façon abondante par de nombreuses équipes de chercheurs travaillant dans divers domaines liés au Traitement Automatique des Langues (TAL), et elle l'est toujours aujourd'hui. Notre objectif dans cette section n'est pas de faire une revue systématique de tous les champs de recherche impliquant Wikipédia, mais plutôt de détailler quelques travaux importants principalement liés à la Recherche d'Information.

Les utilisations de Wikipédia pour l'accès à l'information sont très variées. Des études rapportent des améliorations de performances dans le cadre d'une recherche d'articles de presse ou de pages web en utilisant de l'expansion de requêtes avec des mots liés provenant de Wikipédia (Li et al., 2007; Xu et al., 2009; Meij et de Rijke, 2010).

Koolen et al. (2009) rapportent également des résultats avec une méthode similaire pour la recherche de livres. La recherche documentaire cross-lingue (Nie, 2010) a également beaucoup bénéficié de la nature multilingue de l'encyclopédie, le but étant de proposer à l'utilisateur des documents pertinents exprimés dans une langue différente de la requête. De nombreuses approches se basent sur des corpus comparables afin d'effectuer les traductions, et l'utilisation de versions de Wikipédia en différentes langues permet notamment de s'affranchir des problèmes de différences de domaines (Potthast et al., 2008; Roth et Klakow, 2010).

En lieu d'améliorer ou de traduire la requête ou les documents, Hu et al. (2009) ont utilisé Wikipédia afin de deviner l'intention qu'avait l'utilisateur en soumettant sa requête au système de RI. Plus spécifiquement, ils se sont concentrés sur l'identification de voyages (requêtes pointant vers des lieux ou des destinations), de personnes et de recherche d'emplois. Wikipédia a été aussi utilisée dans des tâches de résumé automatique (Svore et al., 2007), mais c'est pour des problèmes de recherche d'entités (Zaragoza et al., 2007; Bron et al., 2010), d'extraction d'attributs (Wu et Weld, 2007, 2010) et d'extraction de relations (Strube et Ponzetto, 2006; Suchanek et al., 2007; Yan et al., 2009) qu'elle aura été le plus efficace.

Dans cette thèse, nous ne recourons pas aux méthodes d'extraction d'information et n'exploitons que le texte des articles Wikipédia, sans explicitement tenir compte des attributs ou des relations entre les pages. Les données que nous utilisons correspondent à une capture complète de la version anglaise de l'encyclopédie en ligne datant du mois de janvier 2012 qui contient 3 406 520 documents (les pages de redirection et de désambiguïsation ont été supprimées). Ces documents ont été indexés avec Indri, avec les mêmes paramètres que ceux utilisés pour les collections de documents (section 2.3).

2.4.2 New York Times

Notre seconde source d'information est constituée d'articles de presse du New York Times qui ont été publiés entre 1987 et 2007. Cette collection est basée sur le corpus Linguistic Data Consortium (LDC) (Sandhaus, 2008) du NYT [13] dont nous avons supprimé les différentes balises et annotations afin de ne garder que le texte produit par les journalistes. Elle est constituée de 1 855 658 documents.

2.4.3 GigaWord

Comme troisième source d'information, nous utilisons le corpus GigaWord anglais distribué également par le LDC [14] (Graff et Cieri, 2003). Il est constitué de 4 111 240 dépêches d'actualités collectées à partir de quatre sources internationales distinctes : le service anglais de l'Agence France Presse, le service anglais de l'Associated Press Worldstream, le service de dépêches du New York Times et le service anglais de la Xinhua News Agency. Les dépêches étant par nature des articles courts et factuels traitant

13. http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2008T19
14. http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2003T05

de sujets très précis, les documents du GigaWord sont en moyenne plus de deux fois moins longs que ceux du New York Times.

2.4.4 Web

Enfin, nous avons considéré une quatrième source plus générale et de taille beaucoup plus importante : un ensemble de pages web. Le ClueWeb09 étant connu pour contenir un grand nombre de pages considérées comme *spam*, nous avons utilisé une liste standard fournie par Cormack et al. (2011) qui liste tous les documents de la collection ainsi que le pourcentage de *spam* qu'ils contiennent [15]. La notion de *spam* est floue et reste difficile à définir précisément (Gyongyi et Garcia-Molina, 2005). Ici une page Web est considérée comme *spam* lorsqu'elle est de faible qualité et sans valeur, avec de très faibles chances d'être considérée comme pertinente pour n'importe quelle requête pour laquelle elle pourrait être renvoyée. Nous avons ainsi supprimé du ClueWeb09-B (voir section 2.3.4) tous les documents ayant une probabilité d'être un *spam* supérieure à 30%, suivant les recommandations des auteurs (Cormack et al., 2011). Le corpus résultant de cette opération est composé de 29 038 220 pages web.

15. http://plg.uwaterloo.ca/~gvcormac/clueweb09spam/

Chapitre 3

Estimation du contexte thématique par de multiples sources d'informations

Sommaire

3.1 Introduction

Une approche automatique très largement utilisée en Recherche d'Information pour désambiguïser la requête et identifier ses thématiques principales est le retour de pertinence simulé. Le retour de pertinence classique (Koenemann et Belkin, 1996) consiste à

présenter un ensemble de N documents à l'utilisateur et d'attendre ses retours sur leur pertinence. Le système peut alors récupérer automatiquement de nouveaux documents similaires (i.e. liés) à ceux que l'utilisateur a marqué comme pertinents (Rocchio, 1971). Le retour de pertinence simulé fait l'hypothèse que les N documents sont pertinents et s'affranchit ainsi de l'implication de l'utilisateur. Un des défis est ainsi de choisir un ensemble de documents *pseudo-pertinents* d'une taille appropriée pour optimiser les performances du système de RI : trop de documents contiennent trop d'informations pas assez ciblées sur la requête, et trop peu de documents mènent à un manque d'informations.

Après avoir formé un ensemble de documents pseudo-pertinents, l'idée principale des approches faisant du retour de pertinence simulé est d'extraire des caractéristiques ou des indices sur leur contexte thématique. En partant du principe que ces documents sont tous pertinents par rapport à la requête, tous les thèmes abordés devraient l'être aussi. Même si ce n'est évidemment pas le cas en pratique, des études ont montré que supposer que les 10 premiers documents renvoyés sont pertinents était équivalent à utiliser uniquement les documents réellement pertinents au sein de ces 10 documents (He et Ounis, 2009). Les indices sur le contexte thématique sont très généralement des mots ou des multi-mots [1] extraits directement des documents pseudo-pertinents (Lavrenko et Croft, 2001; Zhai et Lafferty, 2001; Metzler et Croft, 2007). Ces mots sont généralement des synonymes ou des concepts liés, et permettent ainsi de compléter les informations fournies par l'utilisateur à travers la requête avec des informations complémentaires qui fournissent une meilleure description du contexte thématique. Une nouvelle recherche est alors automatiquement effectuée en utilisant ces nouvelles connaissances, permettant ainsi de récupérer et de proposer à l'utilisateur des documents liés au contexte thématique estimé.

Nous explorons dans ce chapitre une direction légèrement opposée et expérimentons une méthode qui réduit l'importance des documents non-liés au contexte thématique. Cela nous permet notamment d'intégrer naturellement plusieurs sources d'information différentes (voir section 2.4) pour estimer le contexte thématique, et ainsi de réduire l'importance des documents en fonction de leur divergence par rapport à différentes sources. Cette combinaison de divergences constitue la deuxième contribution de ce chapitre. Nous nous plaçons dans le cadre théorique des modèles de langue pour la Recherche d'Information. Nous commençons ce chapitre par effectuer une revue des modèles de pertinence (*relevance models*), méthode état-de-l'art pour l'estimation du contexte thématique. Nous faisons ensuite une revue des différents travaux combinant plusieurs sources d'informations externes pour améliorer les performances de systèmes de RI, nous détaillons leurs forces et leurs faiblesses afin de mettre en perspective nos contributions. Nous reportons alors les résultats de nos expériences sur les collections de test présentées en section 2.3 et proposons une discussion sur les performances de notre approche avant de conclure ce chapitre.

1. Séquences de plusieurs mots ayant un sens différent de celui des mêmes mots pris séparément. Par exemple, « la grande muraille » fait très probablement référence à la muraille de Chine tandis que les mots « la », « grande » et « muraille » sont ambigus et ne font référence à rien de particulier.

3.2 Recherche documentaire par modèles de langue

3.2.1 Vraisemblance de la requête

Nous prenons dans cette thèse une approche par modèles de langue pour la Recherche d'Information, où les documents sont classés en se basant sur la probabilité qu'ils soient pertinents par rapport à la requête (propre aux modèles probabilistes). Cette probabilité s'exprime formellement par $P(D|Q)$ et estime la pertinence du document D conditionnée par la requête Q. En appliquant le théorème de Bayes, on a :

$$P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)} \tag{3.1}$$

La probabilité *a priori* $P(Q)$ de la requête est constante pour tous les documents et n'affecte donc pas leur classement, nous l'ignorons simplement :

$$P(D|Q) \propto P(Q|D)P(D) \tag{3.2}$$

Dans ce qui suit, nous faisons l'hypothèse que la probabilité *a priori* $P(D)$ d'un document est uniforme pour tous les documents, ce qui revient à les classer uniquement en se basant sur la probabilité $P(Q|D)$ que le document D génère la requête Q. C'est aussi l'approche standard pour la RI par modèle de langue, plus souvent connue sous la dénomination de vraisemblance de la requête (Ponte et Croft, 1998; Hiemstra, 2001) (*query likelihood*), que nous abrévions en QL (nous gardons les notations introduites tout au long de cette thèse). La probabilité *a priori* $P(D)$ peut néanmoins être utile pour favoriser certains documents présentant certaines caractéristiques. Pour cette raison, et pour garder une certaine consistance avec les notations utilisées dans la littérature, nous gardons la probabilité $P(D)$ dans notre modèle mais la traitons comme une constante.

Soit θ_D un modèle de langue unigramme tel que $\theta_D = \{P(w|D)\}_{w \in \mathcal{V}}$, avec $P(w|D)$ étant la probabilité du mot w dans un document D, et $\mathcal{V}$ étant le vocabulaire de la collection. Nous traitons à partir de maintenant chaque document D comme un échantillon du modèle de langue multinomial θ_D. Le modèle de vraisemblance de la requête traditionnel fait l'hypothèse que l'ordre des mots n'a pas d'importance, notre modèle devient alors :

$$P(D|Q) \propto P(D)P(Q|\theta_D) \propto P(D) \prod_{w \in Q} P(w|\theta_D)^{tf(w,Q)} \tag{3.3}$$

Ici, $tf(w, Q)$ est la fréquence du mot w dans la requête Q. Les requêtes contenant en pratique rarement plusieurs fois le même mot, ce terme est parfois omis dans la littérature.

Nous généralisons la fréquence $tf(w, Q)$ en la probabilité $P(w|\theta_Q)$; cette généralisation permet d'introduire le modèle de la requête θ_Q analogue au modèle du document θ_D et d'ouvrir la voie à une estimation plus fine de la pondération des mots de la requête. Les probabilités calculées peuvent être très proches de zéro lorsque l'on considère des larges collections de documents. Afin de prévenir les problèmes de calcul et

d'éviter des approximations non désirées, nous passons à l'échelle logarithmique et calculons la *log-vraisemblance* d'un document par rapport à la requête :

$$\log P(D|Q) \propto \log P(D) + \sum_{w \in Q} P(w|\theta_Q) \log P(w|\theta_D) \tag{3.4}$$

Ce modèle revient à classer les documents par leur divergence de Kullback-Leibler (Zhai et Lafferty, 2001) par rapport à la requête, et nous permet d'incorporer naturellement des informations issues du retour de pertinence grâce à des estimations du modèle de la requête θ_Q.

Un problème bien connu des modèles de recherche d'information est la prise en compte des probabilités nulles par le lissage. En effet si un mot w n'apparaît pas dans le document D la probabilité $P(w|\theta_D)$ est égale à 0. Les probabilités étant multipliées entre elles, l'absence d'un seul mot donnerait une probabilité $P(D|Q) = 0$. Nous réglons ce problème en lissant le modèle de langue du document en utilisant le lissage de Dirichlet. D'autres techniques de lissage sont souvent utilisées [2] mais il a été montré que le lissage de Dirichlet était particulièrement efficace pour les requêtes courtes formées de mots-clés (Zhai et Lafferty, 2004), ce qui est notre cas pour les expériences menées dans cette thèse. La probabilité lissée s'écrit plus formellement comme :

$$P(w|\theta_D) = \frac{c(w, D) + \mu \cdot P(w|\mathcal{C})}{|D| + \mu} \tag{3.5}$$

avec $c(w, D)$ étant le nombre de fois que w apparaît dans le document D, $|D|$ étant la longueur du document et $P(w|\mathcal{C}) = \frac{c(w, \mathcal{C})}{|\mathcal{C}|}$ ($\mathcal{C}$ étant la collection de documents entière). Nous fixons la valeur du paramètre du lissage de Dirichlet μ à 1500, une valeur ayant démontré son efficacité (Zhai et Lafferty, 2004), et nous ne le changeons à aucun moment au cours de nos expériences.

3.2.2 Modèles de pertinence

Le but des modèles de pertinence est d'améliorer la représentation de la requête en sélectionnant des mots ou des termes à partir d'un ensemble de documents initialement renvoyés (Lavrenko et Croft, 2001). La concentration des documents pertinents étant généralement plus élevée dans les premiers rangs de la liste ordonnée de documents, l'ensemble à partir duquel les termes vont être extrait est constitué des N documents les mieux classés. Les modèles de pertinence obtiennent généralement de meilleurs résultats lorsqu'ils sont combinés avec le modèle original de la requête (ou estimation par maximum de vraisemblance). Soit $\tilde{\theta}_Q$ cette estimation de la requête par maximum de vraisemblance et $\hat{\theta}_Q$ un modèle de pertinence, le nouveau modèle de la requête est donné par :

$$P(w|\theta_Q) = \lambda\ P(w|\tilde{\theta}_Q) + (1 - \lambda) P(w|\hat{\theta}_Q) \tag{3.6}$$

2. Comme par exemple l'interpolation linéaire, ou lissage de Jelinek-Mercer.

où $\lambda \in [0,1]$ est un paramètre fixé librement qui contrôle l'influence de la requête originale par rapport au modèle de pertinence. Ce paramètre permet notamment d'éviter des problèmes de glissement de thématique (ou *topic drift*) lorsque des mots éloignés ou n'appartenant pas au contexte thématique de la requête sont ajoutés au modèle de pertinence. Il est à noter que fixer $\lambda = 1$ revient au modèle de vraisemblance de la requête détaillé dans la section précédente (QL). Une des variantes les plus robustes des modèles de pertinence est calculée comme suit :

$$P(w|\hat{\theta}_Q) \propto \sum_{\theta_D \in \Theta} P(\theta_D) P(w|\theta_D) \prod_{t \in Q} P(t|\theta_D) \tag{3.7}$$

où Θ est un ensemble de modèles de documents pseudo-pertinents renvoyés par une première passe d'un algorithme état-de-l'art de recherche documentaire. Comme nous l'avons fait précédemment dans la section 3.2.1, nous faisons l'hypothèse que $P(\theta_D)$ est uniforme. Le modèle de pertinence ainsi estimé est donc une somme sur les différents modèles des documents pseudo-pertinents pondérés par le score de vraisemblance de la requête. L'estimation finale exprimée par l'équation (3.6) est souvent nommée RM3 dans la littérature.

Nous présentons dans le tableau 3.1 un exemple de modèle de pertinence construit pour une requête standard de TREC : « hubble telescope achievements ». Alors que les mots de la requête sont très fortement pondérés, on peut voir que d'autres mots très liés au contexte thématique de la requête sont ajoutés. Les mots « space », « universe » et « NASA » sont par exemple très liés à cette recherche dans laquelle l'utilisateur aimerait connaître les réussites du télescope spatial Hubble développé par la NASA ; celui-ci permet notamment de mesurer le taux d'expansion de l'univers. Le mot « ultraviolet » est aussi très intéressant car il correspond à un des spectromètres qui permet notamment d'étudier les compositions des nuages de gaz ou des atmosphères planétaires. Ces différents mots ainsi que les poids qui leur sont associés permettent ainsi d'avoir une représentation plus complète de la requête.

Dans la prochaine section, nous effectuons une revue des différentes approches utilisant différentes sources d'informations externes pour améliorer la RI.

3.3 Utilisation de sources d'information externes pour la Recherche d'Information

L'utilisation de données externes a été considérablement étudiée dans le cadre du retour de pertinence simulé. Les premières approches ont tenté d'utiliser des ressources structurées comme l'ontologie WordNet (Miller, 1995) permettant par exemple de trouver des synonymes ou de désambiguïser certains mots de la requête. Les premiers travaux d'enrichissement de requête à l'aide de WordNet ont été menés par Voorhees (1993, 1994) sur des requêtes courtes de TREC. Une heuristique exploitant les relations de synonymie et d'hyperonymie mises à disposition par WordNet permettait d'ajouter différents mots liés à la requête en pondérant différemment les relations lexicales. De la même façon, Liu et al. (2004) ont proposé une approche permettant d'ajouter

w	$P(w\|\hat{\theta}_Q)$
telescope	0.0567695577
space	0.0419802250
hubble	0.0380013632
shuttle	0.0217168275
light	0.0217168275
universe	0.0208438799
NASA	0.0195159071
mirror	0.0190680336
earth	0.0172484003
ultraviolet	0.0158515806

TABLE 3.1 – *Dix mots de plus fortes probabilités du modèle de pertinence estimé pour la requête « hubble telescope achievements » (issue de la tâche Robust de TREC 2004, requête 303) en utilisant les 10 premiers documents pseudo-pertinents renvoyés en utilisant la vraisemblance de la requête.*

également à la requête des mots issus de définitions et des sous-phrases à partir de l'ontologie. D'un autre côté, Mandala et al. (1999) présentent dans leur travail une méthode d'enrichissement qui combine des caractéristiques extraites de WordNet et de deux thésaurus spécifiques créés manuellement à partir de la collection de documents. Le premier a pour but d'identifier les relations sémantiques entre deux mots en calculant ses co-occurrences. Le second se concentre sur la pondération de paires de mots liés par leur relation syntaxique.

Les approches plus récentes utilisant des sources d'information externes ont remplacé WordNet par DBpedia [3] et Wikipédia, qui ont l'avantage de proposer des informations contrôlées et validées par une large communauté, tout en étant mises à jour en permanence. Dans leur étude, Li et al. (2007), ajoutent des mots provenant d'articles Wikipédia dont la catégorie est majoritaire au sein d'un ensemble de 20 articles renvoyés pour chaque requête. L'utilisation de Wikipédia comme source pour l'estimation de modèles de pertinence et pour l'expansion a aussi été expérimentée par Meij et de Rijke (2010) qui ont appris un classifieur binaire [4] afin d'identifier les articles Wikipédia liés aux requêtes. Les articles catégorisés comme liés à la requête servaient ensuite de source pour l'extraction de mots utilisés pour enrichir la requête.

D'autres types de sources d'information telles que des articles journalistiques ou même le Web ont également été utilisées dans ce but (Bendersky et al., 2012; Diaz et Metzler, 2006). Ces deux dernières études font partie des rares approches à avoir proposé des méthodes permettant de combiner des sources d'information, mais leurs évaluations sont sujettes à quelques problèmes. Dans la première, Bendersky et al. (2012) utilisent les textes de liens hypertexte et d'en-têtes, qui sont des unités textuelles très pe-

3. http://dbpedia.org/

4. Nous n'entrons pas dans le détail des différentes méthodes de classification dans cette thèse. Concernant l'étude de Meij et de Rijke (2010), le classifieur utilisé était une machine à vecteur de support (SVM).

tites qui sont moins susceptibles de contenir des informations complètes sur le contexte thématique. Les auteurs utilisent également Wikipédia mais ne rapportent aucun résultat de sa contribution au sein de la combinaison de sources. Diaz et Metzler (2006) ont mené des expériences avec des sources d'informations plus larges et plus générales que celles qu'ont utilisé Bendersky et al. (2012). Ils présentent une approche qui combine différents modèles de pertinence estimant le modèle de la requête en utilisant un large ensemble d'articles journalistiques et deux corpus Web comme sources externes. Néanmoins, ils ne combinent jamais les trois modèles ensemble et ne reportent les résultats de recherche documentaire qu'avec les combinaisons de toutes les paires de sources d'information.

À notre connaissance, cette dernière approche est la plus proche de celle que nous présentons dans ce chapitre, à l'exception près que nous considérons des n-grammes au lieu d'unigrammes pour l'estimation du modèle de la requête. Nous menons également des expériences avec des sources d'information encore non utilisées précédemment en RI, telles que le New York Times (voir présentation en section 2.4). Dans la prochaine section, nous présentons une méthode calculant des divergences à partir de sources d'information afin d'estimer le contexte thématique. La méthode réduit ensuite le score des documents en fonction de ces divergences.

3.4 Divergence à partir de sources d'information

3.4.1 Contribution

Le but de la contribution de ce chapitre est de pouvoir modéliser avec précision le contexte thématique d'une requête en utilisant des sources d'information externes. Nous utilisons la divergence de Kullback-Leibler pour mesurer le gain d'information ou la dissimilarité entre une ressource donnée $\mathcal{R}$ et un document D. La divergence de Kullback-Leibler permet de mesurer une dissimilarité entre deux distributions de probabilités. Plus formellement, nous reprenons la notation introduite dans la section précédente et définissons $\theta_{\mathcal{R}}$ comme étant le modèle de langue de la ressource $\mathcal{R}$ et θ_D comme étant le modèle de langue du document D. La divergence de Kullback-Leibler entre ces deux distributions multinomiales se note :

$$
\begin{aligned}
KL(\theta_{\mathcal{R}}||\theta_D) &= \sum_{t \in V} P(t|\theta_{\mathcal{R}}) \log \frac{P(t|\theta_{\mathcal{R}})}{P(t|\theta_D)} && (3.8)\\
&= \sum_{t \in V} P(t|\theta_{\mathcal{R}}) \log P(t|\theta_{\mathcal{R}}) - \sum_{t \in V} P(t|\theta_{\mathcal{R}}) \log P(t|\theta_D) && (3.9)\\
&\propto -\sum_{t \in V} P(t|\theta_{\mathcal{R}}) \log P(t|\theta_D) && (3.10)
\end{aligned}
$$

où t est un terme appartenant au vocabulaire V. Le premier terme de l'équation (3.9) est l'entropie de la ressource $\mathcal{R}$ et n'affecte donc pas l'estimation du contexte thématique : ce nombre est le même pour tous les documents. Nous simplifions alors cette équation et calculons donc la divergence de Kullback-Leibler à partir de l'équation (3.10). Le

modèle $\theta_{\mathcal{R}}$ est quant à lui estimé par retour de pertinence simulé. Soit Θ un ensemble de modèles constitué à partir des N documents issus de la ressource $\mathcal{R}$ les mieux classés par leur vraisemblance à la requête (modèle de classement de documents QL), le modèle de pertinence thématique de la ressource est estimé par :

$$P(t|\hat{\theta}_{\mathcal{R}}) \propto \sum_{\theta_D \in \Theta} P(\theta_D) H_{\Theta}(t) \prod_{w \in Q} P(w|\theta_D) \tag{3.11}$$

On peut voir très facilement le parallèle avec l'estimation du modèle de la requête dans le cadre d'un modèle de pertinence (voir équation (3.7)). Dans cette nouvelle méthode d'estimation, nous remplaçons simplement la probabilité $P(w|\theta_D)$ d'occurrence du mot w dans le document D par l'entropie $H_{\Theta}(t)$ du terme [5] t dans l'ensemble Θ de documents pseudo-pertinents issus de la ressource. Nous réduisons ainsi la ressource $\mathcal{R}$ au contexte thématique de la requête déterminé par l'ensemble de documents pseudo-pertinents. Le modèle $\hat{\theta}_{\mathcal{R}}$ ainsi calculé peut être vu comme une interprétation ou une désambiguïsation de la requête par la ressource $\mathcal{R}$.

Quant à elle, l'entropie du terme t est calculée selon :

$$H_{\Theta}(t) = -\sum_{w \in t} P(w|\Theta) \log P(w|\Theta) \tag{3.12}$$

Cette entropie donne ainsi une mesure de la quantité d'information contenue dans l'ensemble Θ de documents. Un des avantages d'utiliser l'entropie est de pouvoir la calculer en considérant des n-grammes et non pas seulement des unigrammes comme dans les approches traditionnelles de modèles de pertinence. Nous modifions l'ensemble V utilisé dans l'équation (3.10) afin qu'il ne représente pas seulement l'ensemble des unigrammes, mais également des bi et trigrammes. Nous parcourons ainsi les documents de l'ensemble Θ en considérant toutes les séquences de 1, 2 ou 3 mots. Dans ce chapitre, une terme fait ainsi référence à une séquence de plusieurs mots.

Nous suivons l'équation (3.10) pour calculer la divergence d'information entre la ressource $\mathcal{R}$ et un document D :

$$KL(\hat{\theta}_R||\theta_D) = -\sum_{t \in V} P(t|\hat{\theta}_{\mathcal{R}}) \log P(t|\theta_D) \tag{3.13}$$

Le score final d'un document D par rapport à une requête utilisateur $Q = w_1, ..., w_n$ est déterminé par la combinaison linéaire du modèle standard de vraisemblance de la requête et des divergences à partir de plusieurs ressources. Nous l'écrivons formellement comme :

$$s(Q,D) = \lambda \log P(Q|\theta_D) - (1-\lambda) \sum_{\mathcal{R} \in \mathcal{S}} \varphi_{\mathcal{R}} \cdot KL(\hat{\theta}_{\mathcal{R}}||\theta_D) \tag{3.14}$$

$$= \lambda \log P(Q|\theta_D) + (1-\lambda) \sum_{\mathcal{R} \in \mathcal{S}} \varphi_{\mathcal{R}} \sum_{t \in V} P(t|\hat{\theta}_{\mathcal{R}}) \log P(t|\theta_D) \tag{3.15}$$

où $\mathcal{S}$ est un ensemble de sources d'information et $P(Q|\theta_D)$ est le modèle standard de vraisemblance de la requête avec lissage de Dirichlet. $\varphi_{\mathcal{R}} \in [0,1]$ représente le poids

5. Par « terme », nous entendons ici un mot ou une séquence de plusieurs mots.

donné à la ressource $\mathcal{R}$; il peut aussi être vu comme une probabilité *a priori* sur $\mathcal{R}$. Nous détaillons dans la prochaine section comment nous déterminons ce poids. Nous pouvons directement voir que la fonction de classement ci-dessus s'apparente à la fonction classique des modèles de pertinence (RM3) détaillés en section 3.2.2 qui, après développement, se présente sous la forme suivante :

$$s_{RM3}(Q,D) = \lambda \ \log P(Q|\theta_D) + (1-\lambda) \sum_{w \in V} P(w|\hat{\theta}_Q) \log P(w|\theta_D) \tag{3.16}$$

où $P(w|\hat{\theta}_Q)$ est le modèle de la requête estimé par retour de pertinence simulé.

L'équation (3.14) permet ainsi de réduire le score d'un document possédant une trop grande divergence informative par rapport à des modèles thématiques estimés sur différentes sources d'information ; plus grande est la divergence, plus faible est le score du document. La combinaison de ces sources d'information agit intuitivement comme une généralisation du contexte thématique : les différents modèles estimés possèdent chacun des caractéristiques différentes par rapport à ce contexte et apportent chacun leur spécificité. Wikipédia contient des informations segmentées par articles rédigés dans un style encyclopédique, tandis que le corpus du New York Times contient des informations vraisemblablement plus factuelles et souvent centrées sur des sujets ou des évènements très spécifiques. Ainsi, augmenter le nombre de sources d'information devrait finalement améliorer la représentation du besoin d'information de l'utilisateur, tout en ne reposant que sur sa requête. Nous choisissons ici d'utiliser le modèle standard de vraisemblance de la requête pour des raisons de reproductibilité et de facilité de comparaison, mais il pourrait être entièrement remplacé par n'importe quel autre modèle de Recherche d'Information état-de-l'art (comme par exemple les modèles MRF-IR (Metzler et Croft, 2005) ou BM25 (Robertson et Walker, 1994)). Nous attribuons à notre méthode l'acronyme DfRes, pour *Divergence from Resources*.

3.4.2 Systèmes de base et comparaison

Nous comparons la méthode DfRes avec trois systèmes de base afin de mesurer son impact sur les performances de recherche documentaire par rapport à l'état-de-l'art. Le premier de ces systèmes est la vraisemblance de la requête (noté QL). Le second est RM3 (voir section 3.2.2) qui, comme nous l'avons vu précédemment est une méthode état-de-l'art d'enrichissement de requête. Finalement, nous comparons notre approche avec la méthode qui lui est la plus proche : la combinaison de modèles de pertinence étudiée par Diaz et Metzler (2006). Cette méthode (à laquelle nous référons sous l'acronyme MoRM pour *Mixture of Relevance Models*) est une généralisation du modèle RM3 qui permet de prendre en compte plusieurs sources de documents pseudo-pertinents. Plus formellement, le modèle de la requête s'estime par la formule :

$$P(w|\hat{\theta}_Q) \propto \sum_{\mathcal{R} \in \mathcal{S}} k_{\mathcal{R}} \sum_{\theta_D \in \Theta} P(\theta_D) P(w|\theta_D) \prod_{t \in Q} P(t|\theta_D) \tag{3.17}$$

Alors que la méthode RM3 fait du retour de pertinence en utilisant la collection cible, la méthode QL n'utilise aucune information additionnelle. Les méthodes MoRM et DfRes

combinent la collection cible avec les quatre sources d'information générales présentées en section 2.4 : Wikipédia (wiki), le New York Times (nyt), le GigaWord (gigaword) et le ClueWeb09-B sans spam (web).

3.5 Expérimentations et résultats

Cette section détaille la méthodologie que nous utilisons pour nos expérimentations ainsi que les résultats que nous obtenons. Nous évaluons les différentes approches présentées précédemment en utilisant les quatre collections de test de TREC détaillées en section 2.3 : Robust04, WT10g, GOV2 et ClueWeb09-B.

3.5.1 Protocole expérimental

Les méthodes RM3, MoRM et DfRes dépendent toutes les trois de trois paramètres : λ qui contrôle le poids donné à la requête originale, k qui représente le nombre de termes utilisés pour estimer le contexte thématique et N qui est le nombre de documents pseudo-pertinents à partir desquels les termes sont extraits. Nous estimons le paramètre λ par validation croisée : nous formons un ensemble d'entraînement avec toutes les requêtes sauf une qui sera testée. Nous observons ensuite quelle valeur de λ donne les meilleurs résultats pour toutes les requêtes de l'ensemble d'entraînement puis appliquons cette valeur à la requête restante. Nous rapportons ensuite les résultats des performances moyennées.

Concernant les paramètres N et k, une étude menée par He et Ounis (2009) a montré qu'effectuer du retour de pertinence simulé avec les 10 premiers documents pseudo-pertinents était aussi efficace que d'effectuer du retour de pertinence simulé avec uniquement les documents pertinents présents dans ce top-10 (et ce sur plusieurs collections de test). Ils ont également montré qu'il n'y avait aucune différence statistiquement observable entre ces deux méthodes. Nous concluons donc qu'utiliser 10 documents semble être efficace, nous fixons $N = 10$ pour ces expériences. De la même façon, nous fixons le nombre de termes utilisés $k = 20$. Malgré le fait que les résultats rapportés dans le tableau 3.2 correspondent spécifiquement à ces réglages de paramètres, nous explorons dans la section suivante l'influence des paramètres N et k sur les performances de notre modèle. À partir de ce point, lorsque nous discutons des résultats obtenus en utilisant des sources d'information prises individuellement pour notre méthode DfRes, nous utilisons la notation DfRes-$\mathcal{R}$ où $\mathcal{R} \in \{Wiki, NYT, Gigaword, Web\}$.

Un dernier paramètre utilisé par les méthodes que nous évaluons est le poids attribué à chaque ressource. Pour la méthode MoRM, nous suivons exactement le procédé décrit par Diaz et Metzler (2006) pour fixer le paramètre $k_{\mathcal{R}}$. Ce paramètre contrôle l'importance de la ressource $\mathcal{R}$ et est analogue à notre paramètre $\varphi_{\mathcal{R}}$ (il n'a aucun lien avec le nombre k de termes). Ce procédé est plus spécifiquement déterminé par validation croisée en testant les valeurs de $k_{\mathcal{R}}$ dans l'ensemble $\{0,0; 0,1; 0,2; ...; 1,0\}$. Nous utilisons le même partitionnement des requêtes que pour le paramètre λ pour effectuer

la validation croisée (i.e. une requête contre toutes les autres). Concernant le paramètre $\varphi_{\mathcal{R}}$ utilisé par notre approche DfRes dans l'équation 3.14, nous l'estimons de façon plus fine. Pour toutes les requêtes sauf une, nous regardons quelle ressource $\mathcal{R}$ maximise les performances de recherche documentaire pour DfRes-$\mathcal{R}$. Le paramètre $\varphi_{\mathcal{R}}$ représente alors le nombre de fois où la ressource $\mathcal{R}$ est meilleure que toutes les autres, divisé par le nombre total de requêtes. Plus formellement :

$$\varphi_{\mathcal{R}} = \frac{\sum_{i=1}^{M} max_{AP}(Q_i, \mathcal{R})}{M} \tag{3.18}$$

où $max_{MAP}(Q_i, \mathcal{R}) = 1$ si DfRes-$\mathcal{R}$ obtient une MAP [6] plus importante que DfRes avec les autres ressources, 0 sinon. Dans cette formule, M correspond au nombre de requêtes utilisées et Q_i est la i^e requête. Nous gardons le même principe que la validation croisée et nous apprenons ce paramètre sur toutes les requêtes, sauf une, et nous le testons sur cette dernière, afin d'éviter des problèmes évidents posés par l'utilisation de données de test pour l'apprentissage.

3.5.2 Résultats

Les résultats des différentes approches que nous expérimentons dans ce chapitre sont présentés dans le tableau 3.2. Bien qu'il puisse paraître synthétique, il est le résultat de 320 *runs* [7], soit 192 000 requêtes évaluées.

	QL		RM3		MoRM		DfRes	
	MAP	P@20	MAP	P@20	MAP	P@20	MAP	P@20
WT10g	0,2026	0,2429	0,2035	0,2449	$0,2339^{\alpha,\beta}$	$0,2833^{\alpha,\beta}$	$0,2463^{\alpha,\beta}$	$0,2954^{\alpha,\beta}$
Robust04	0,2461	0,3528	$0,2727^{\alpha}$	0,3677	$0,2869^{\alpha,\beta}$	$0,3799^{\alpha,\beta}$	$0,3147^{\alpha,\beta,\gamma}$	$0,4024^{\alpha,\beta,\gamma}$
GOV2	0,2911	0,5145	0,2877	0,5074	$0,3083^{\alpha,\beta}$	$0,5409^{\alpha,\beta}$	$0,3257^{\alpha,\beta,\gamma}$	$0,5638^{\alpha,\beta,\gamma}$
ClueWeb09-B	0,1007	0,2347	0,1007	0,2260	0,1045	0,2250	$0,1140^{\alpha,\beta,\gamma}$	$0,2770^{\alpha,\beta,\gamma}$

TABLE 3.2 – *Résultats de recherche documentaire reportés en terme de précision moyenne (MAP) et de précision à 20 documents pour les approches QL, RM3, MoRM et DfRes. Nous utilisons le test apparié de Student (t-test) pour déterminer les différences significatives avec les systèmes de base. α, β et γ indiquent respectivement des améliorations significatives par rapport à QL, RM3 et MoRM, avec $p < 0,05$.*

L'observation principale que l'on peut faire des résultats de recherche *ad hoc* présentés dans ce tableau est que l'utilisation d'une combinaison de sources d'information est toujours plus performante que l'utilisation de la collection cible. Les chiffres que nous reportons sont différents de ceux présentés par Diaz et Metzler (2006), néanmoins nous n'avons pas pu reproduire les expériences à l'identique car les auteurs n'ont pas détaillé tous leurs paramètres d'indexation. Ces variations impactent toutefois peu les résultats relatifs : la méthode MoRM est plus efficace que la méthode RM3 qui est plus efficace que QL.

6. Pour plus de détails sur la mesure MAP, voir la section 2.2.3.
7. Par *run*, nous désignons les résultats (sous forme d'une liste de documents la plupart du temps) renvoyés pour un ensemble de requêtes prédéfini.

On voit dans le tableau 3.2 que la méthode DfRes obtient les meilleurs résultats, et ce pour toutes les collections de test. Il est néanmoins plus difficile d'observer des améliorations significatives par rapport à MoRM sur les deux collections web WT10g et ClueWeb09-B. Nous conduisons une analyse plus détaillée dans les prochaines sections, mais nous pouvons d'ores et déjà conjecturer que cette difficulté est due à la nature même de ces collections. Les documents web peuvent en effet être très hétérogènes et bruités, ce qui peut gêner une bonne sélection des termes utilisés pour estimer le contexte thématique. Pour les collections Robust04 et GOV2, les améliorations que DfRes apporte par rapport à MoRM sont très importantes et sont toutes significatives. Contrairement aux collections web précédentes, les articles journalistiques et les entrées du domaine `.gov` américain sont chacun ciblés sur des sujets précis.

Les différentes sources d'information utilisées sont de natures très différentes, et contribuent donc à estimer le contexte thématique avec des importances différentes. Comme nous l'avons détaillé dans la section précédente, nous utilisons une méthode permettant d'apprendre le poids de chaque ressource pour chaque requête, en utilisant les autres requêtes du jeu de test comme ensemble d'apprentissage. À titre d'indication, nous reportons dans le tableau 3.3 les moyennes de ces poids appris pour les différentes sources d'information utilisées sur les quatre collections de test.

	nyt	**wiki**	**gigaword**	**web**	**WT10g**	**Robust04**	**GOV2**	**ClueWeb09-B**
WT10g	0,303	0,162	0,121	**0,313**	0,101	-	-	-
Robust04	**0,309**	0,076	0,281	0,149	-	0,185	-	-
GOV2	0,213	0,121	0,179	0,219	-	-	**0,261**	-
ClueWeb09-B	0,195	0,215	0,127	**0,351**	-	-	-	0,108

TABLE 3.3 – *Moyennes des poids φ_R appris pour les quatre collections. Les nombres en gras correspondent aux plus forts poids par collection.*

Une observation que l'on peut faire de ce tableau (et qui va se confirmer dans les prochaines sections) est l'importance du New York Times comme source d'information. On voit en effet qu'elle a un poids très important pour les collections WT10g et Robust04, et que même si elle a une importance réduite pour GOV2 et ClueWeb09-B les poids restent relativement élevés comparés aux autres. Un résultat surprenant et qui sera récurrent tout au long de cette thèse est l'inefficacité de Wikipédia comme source d'information dans le cadre d'une recherche documentaire utilisant des modèles de pertinence. On aurait en effet initialement pu penser qu'utiliser Wikipédia pourrait donner de bons résultats de par sa structure encyclopédique où les sujets sont cloisonnés par article. Il semble néanmoins que le vocabulaire est moins varié que dans des sources comme le NYT, où les journalistes professionnels sont entraînés à user de synonymes et de métaphores pour éviter les répétitions, ce qui permet à nos modèles de construire une estimation plus complète du contexte thématique. Les chiffres présentés dans le tableau 3.3 montrent que les ensembles bruts de pages web (WT10g et ClueWeb09-B) sont trop bruités pour pouvoir être des sources efficaces. Il est néanmoins très intéressant de voir le poids de la ressource **web** par rapport à ClueWeb09-B [8],

8. Il s'agît en effet de la même ressource, à l'exception près que la ressource **web** a été nettoyée d'envi-

qui montre qu'une phase de nettoyage de ces documents bruités peut être extrêmement bénéfique à la formation d'une source d'information complète et hétérogène. Elle est assez logiquement la source la plus importante pour les deux collections Web, tandis que le NYT et le GigaWord sont eux aussi assez logiquement les meilleures sources pour la collection Robust04. Ces observations sont assez intuitives (des sources journalistiques sont plus efficaces pour améliorer une recherche d'articles dans une base de journaux), mais nous trouvons qu'il est important que ces intuitions se valident dans notre cas.

3.5.3 Qualité du contexte thématique estimé

Dans cette section, nous analysons les résultats plus en détail et faisons varier la valeur du paramètre λ contrôlant l'influence du contexte thématique par rapport à la requête originale. Ainsi, suivant la formule détaillée dans l'équation (3.14), $\lambda = 1$ signifie que toute l'importance est donnée à la requête originale, tandis que $\lambda = 0$ signifie que toute l'importance est donnée au contexte thématique estimé. Nous rapportons dans les figures 3.1, 3.2, 3.3 et 3.4 les performances de DfRes utilisant toutes les sources d'information individuellement, ainsi que leur combinaison (sous la dénomination « tout »).

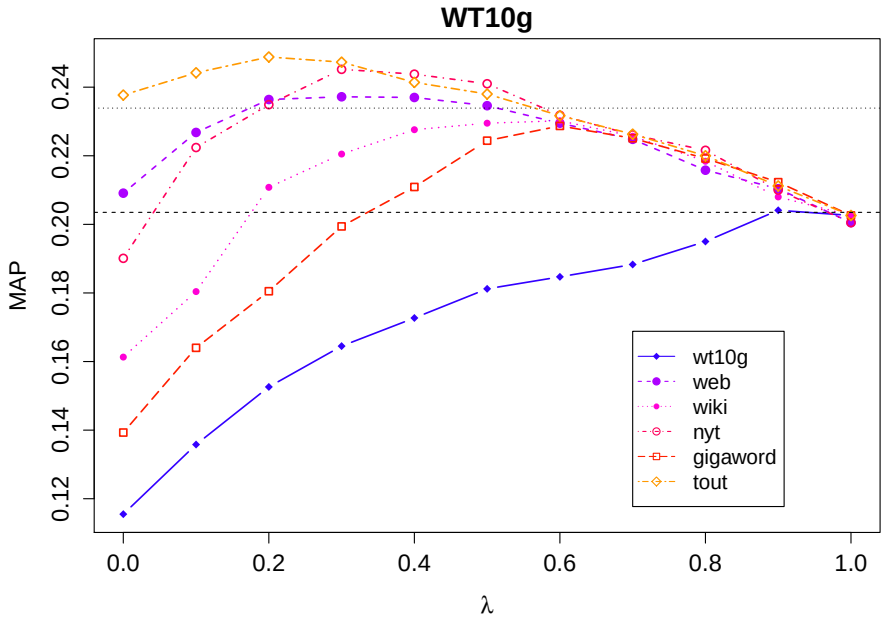

FIGURE 3.1 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection WT10g. La méthode DfRes dont les résultats sont reportés dans le tableau 3.2 est représentée par la courbe « tout », tandis que les autres courbes correspondent à la méthode DfRes utilisant une seule source d'information à la fois. Les systèmes de bases sont reportés pour référence : les tirets représentent RM3 et la ligne pointillée représente MoRM.*

Nous voyons sur ces figures que ces sources d'information agissent différemment suivant la collection utilisée. Pour la collection WT10g (figure 3.1), il est très surprenant

ron 50% des documents considérés comme spam (avec un taux supérieur à 70%). Voir 2.4.

de voir que DfRes-NYT obtient de très bons résultats, presque aussi bons que DfRes utilisant toutes les ressources. En effet, étant donné que WT10g est une collection de pages web, nous aurions pensé que DfRes-Web ou DfRes-WT10g auraient obtenu de meilleurs résultats. Un autre résultat remarquable est l'inefficacité de WT10g comme source d'enrichissement : les performances de DfRes-WT10g sont les moins bonnes pour toutes les valeurs de λ. Ceci est néanmoins cohérent avec les résultats reportés précédemment par Diaz et Metzler (2006), qui ont montré que les sources d'articles journalistiques sont plus efficaces que les sources de pages web dans le cas de recherches sur la collection WT10g. Si nous supprimons WT10g de l'ensemble des ressources utilisées pour DfRes-tout, les performances sont améliorées avec une MAP de 0,2501. Nous voyons que DfRes-Gigaword et DfRes-Wiki obtiennent globalement de pauvres résultats, mais leur absence dans la combinaison de ressources affecte négativement les performances. Enfin, un résultat remarquable de la combinaison de sources d'information est l'apparente robustesse du contexte estimé. En effet, nous pouvons voir que fixer $\lambda = 0$ dégrade peu les résultats : nous observons même que les performances sont au-dessus des systèmes de base (pour lesquels les valeurs de λ ont été apprises) et de tous les autres DfRes-$\mathcal{R}$ avec $\lambda = 0$. De plus, les performances sont supérieures à celles obtenues en utilisant uniquement la requête originale ($\lambda = 1$).

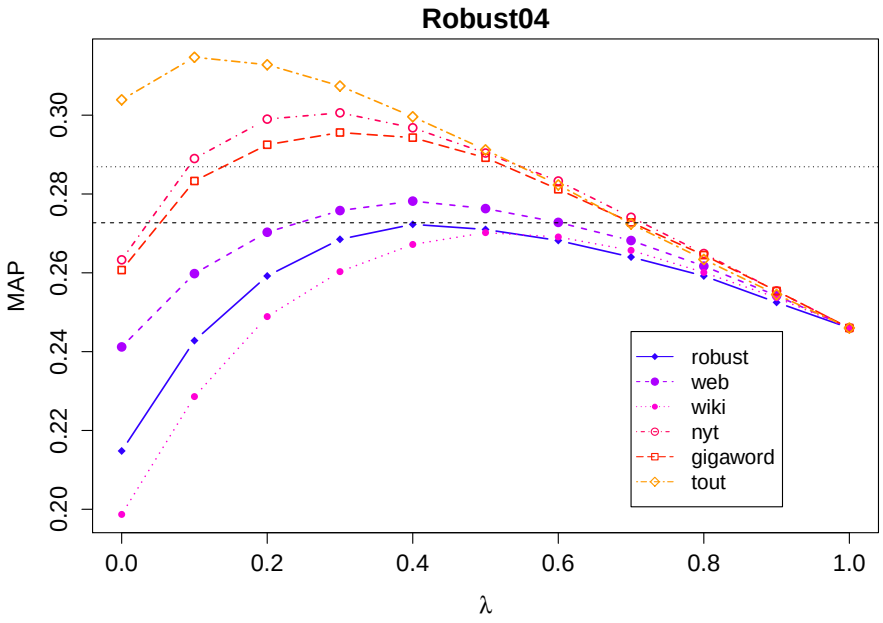

FIGURE 3.2 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection Robust04. La légende est identique à celle de la figure 3.1.*

Les résultats sont plus cohérents sur la collection Robust04 (figure 3.2). L'utilisation du NYT et du Gigaword permet d'obtenir de très bons résultats, alors que la combinaison de toutes les ressources obtient systématiquement les meilleurs résultats. Il est important de noter que le NYT n'est pas inclus dans les documents composant la collection Robust04, c'est donc réellement une source « externe » qui donne de bons résultats. De la même façon, les services de dépêches d'où sont issus les documents du Gigaword

ne se recoupent pas avec les journaux présents dans Robust04. Par contre, le NYT, le Gigaword et la collection Robust04 couvrent à quelques détails près les même périodes, ce qui explique les bons résultats des deux sources d'information journalistiques pour une tâche de recherche d'articles. Il est malgré tout surprenant de voir les mauvais résultats obtenus en utilisant uniquement Robust04 comme source d'information, mais ceci est cohérent avec le but initial de la tâche Robust de TREC qui était de reprendre les requêtes pour lesquelles les systèmes de RI obtenaient de mauvais résultats dans le cas d'expansion ou de reformulation de requête (Voorhees, 2005). Nous remarquons également que les systèmes de base obtiennent des résultats mitigés sur cette collection, contrairement à DfRes utilisant toutes les ressources. Nous notons que la combinaison de toutes les ressources avec $\lambda = 0$ obtient de meilleures performances que toutes les autres ressources utilisées individuellement, ce qui confirme le comportement observé sur la collection WT10g.

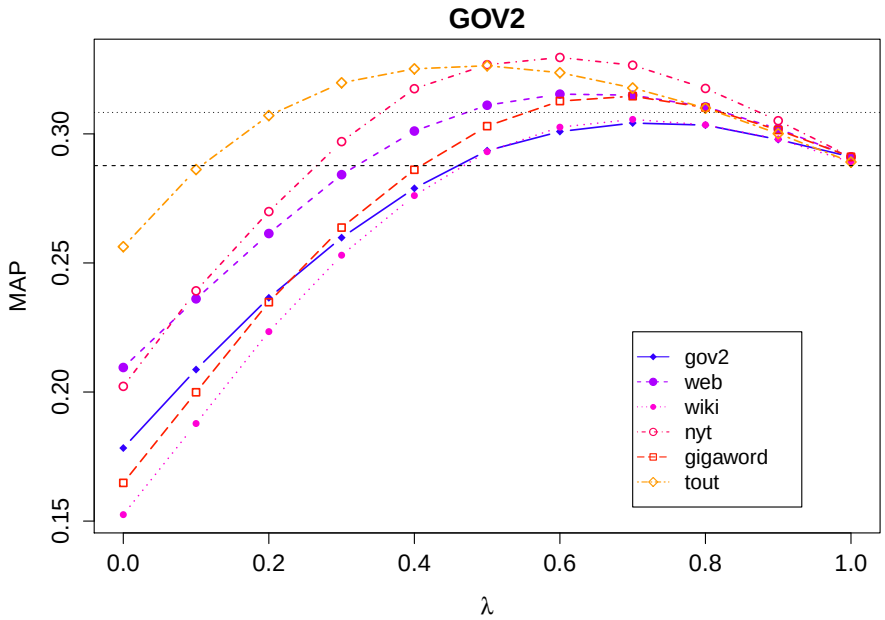

FIGURE 3.3 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection GOV2. La légende est identique à celle de la figure 3.1.*

Ces observations sont malgré tout très différentes pour les deux collections restantes : GOV2 et ClueWeb09-B. Nous pouvons voir que les courbes présentées dans les figures 3.3 et 3.4 sont relativement similaires, et si l'ordre des ressources donnant les meilleurs résultats n'est pas le même (sauf pour la combinaisons de toutes les ressources), les formes des courbes en fonction du paramètre λ sont presque identiques. Ces deux collections sont très proches de par leur nature (ensembles de pages web) et surtout par leur taille : GOV2 contient 25 millions de documents et ClueWeb09-B en contient 50 millions. GOV2 se différencie principalement par le fait qu'elle contient des transcriptions de documents (PDF, Word et postscript) en sus des pages web.

La principale opposition que nous pouvons observer par rapport aux deux collections précédentes est la baisse de performance quand on fait tendre λ vers 0. Les valeurs

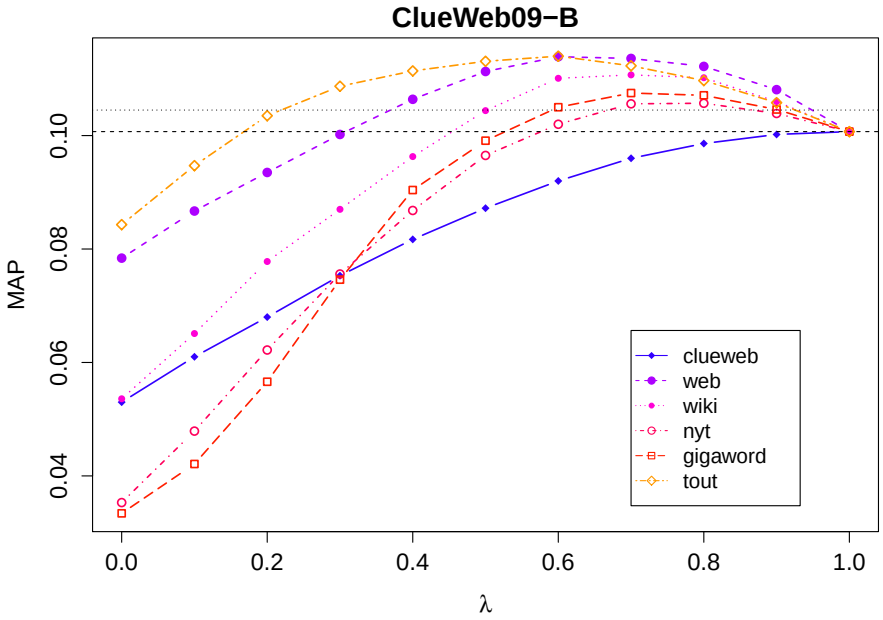

FIGURE 3.4 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection ClueWeb09-B. La légende est identique à celle de la figure 3.1.*

optimales de λ se situent vers 0,4-0,5 pour la collection GOV2 en utilisant toutes les ressources, et plutôt vers 0,6-0,7 en utilisant les ressources individuellement. De la même façon, ces valeurs se situent vers 0,6 pour la collection ClueWeb09-B et 0,7-0,8 pour les ressources individuelles. Ainsi, si le contexte thématique estimé semblait être de bonne qualité pour les collections WT10g et Robust04, il semble qu'il soit moins efficace pour des collections de grande taille.

Nous pensons néanmoins que ces résultats sont un biais inhérent à l'évaluation automatique des performances de systèmes de RI sur des collections composées d'un très grand nombre de documents. Comme nous l'avons vu dans la section 2.2, la pertinence des documents par rapport à une requête est jugée manuellement par des assesseurs. Il est ainsi logiquement plus facile de produire des jugements de pertinence complets pour des collections de l'ordre du million de documents, où le nombre de documents pertinents par requête est réduit, que pour des collections de plusieurs dizaines de millions. Notre approche reformule la requête en ajoutant un grand nombre de mots, le plus souvent des synonymes, des hyperonymes ou des mots conceptuellement liés au thème de la requête. De nombreux documents potentiellement non jugés peuvent alors être récupérés par notre méthode, et sont considérés comme non-pertinents par défaut (et ce même s'ils contiennent des informations pertinentes).

Cette hypothèse est confirmée par les courbes de la figure 3.5. Nous voyons en effet très bien que pour $\lambda = 0$, il y a environ 10% moins de documents jugés pour la collection GOV2 et 5% moins pour le Clueweb09-B. Ces chiffres étant absolus, la différence relative est encore plus importante : -28% pour GOV2 entre $\lambda = 1$ et $\lambda = 0$ et -35% pour ClueWeb09-B. Les nombres de documents renvoyés et jugés pour les deux

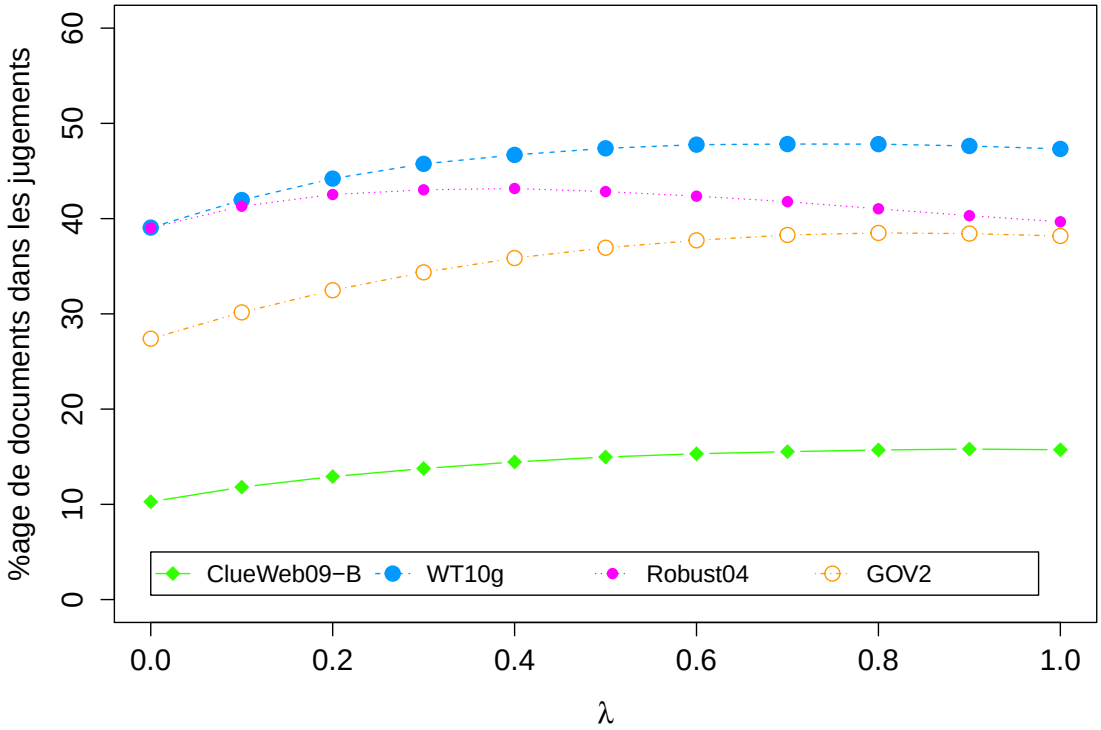

FIGURE 3.5 – *Pourcentage de documents (parmi les 1000 premiers renvoyés par le système) ayant été jugés par des assesseurs, pertinents ou non. Nous considérons ici les runs DfRes-tout sur toutes les collections, et faisons varier le paramètre λ.*

autres collections sont beaucoup plus importants, et nous voyons même que les pourcentages varient très peu pour Robust04. Loin de remettre en cause la validité et la justification de ces larges collections, nous pointons le fait qu'il semble normal d'observer des comportements différents entre les collections WT10g/Robust04 et les collections GOV2/ClueWeb09-B. De plus, le fait que nos méthodes aient des comportements similaires au sein de ces « groupes » de collections justifie la généralisation de notre approche.

3.5.4 Influence du nombre de termes et du nombre de documents

En plus du paramètre λ, nous explorons dans cette section l'influence du nombre k de termes ainsi que du nombre N de documents pseudo-pertinents utilisés pour estimer le contexte thématique sur les performances de recherche documentaire. Nous ne reportons pas les résultats du changement du nombre de documents pseudo-pertinents car ils n'ont rien de notable. En effet, les performances restent presque constantes pour toutes les ressources lorsque N varie. Les changements en précision moyenne (MAP) sont de $\pm$ 5% de $N = 2$ à $N = 20$, suivant la ressource. Cette stabilité démontre que la méthode DfRes est peu sensible au nombre de documents pseudo-pertinents utili-

sés. Des résultats complémentaires sur la variation du nombre de documents pseudo-pertinents et sur la sélection de « bons » documents sont disponibles dans l'étude de He et Ounis (2009).

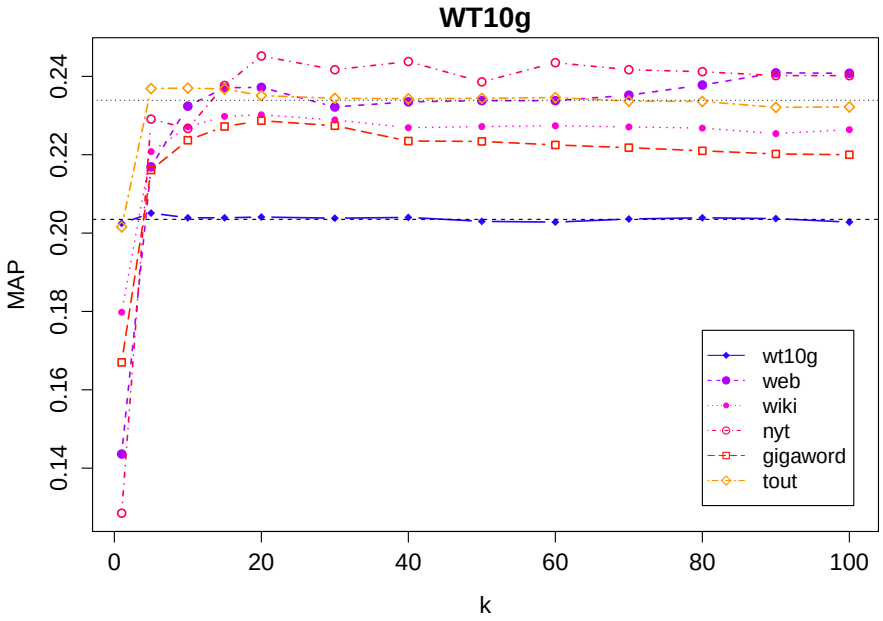

FIGURE 3.6 – *Performances (exprimées en MAP) en fonction du nombre de termes k utilisés pour estimer le modèle de langue de la source d'information sur la collection WT10g. La légende est identique à celle des figures précédentes (3.1 et suivantes).*

Nous avons également mené des expériences en variant le nombre de termes utilisés pour estimer le modèle de chaque ressource. Alors que nous pouvions penser qu'augmenter le nombre de termes aurait amélioré la granularité du modèle et aurait peut-être pu capturer des indices contextuels plus fins et plus nombreux, nous voyons par exemple dans la figure 3.7 que, pour la collection Robust04, l'utilisation de 100 termes n'apporte pas de grandes différences par rapport à une utilisation de 20 termes. Nous pouvons même voir sur la figure 3.6 qu'utiliser 5 ou 10 termes permet d'obtenir les meilleurs résultats de DfRes pour la collection WT10g. Nous voyons même que, pour cette collection, l'ajout d'un grand nombre de termes dégrade les performances notamment pour DfRes-Gigaword ou DfRes-Wiki. Il n'y a néanmoins pas de différences significatives pour les différentes valeurs de k choisies, pour toutes les méthodes reportées ici.

Ces observations sont assez similaires pour la collection GOV2 (figure 3.8), où l'on voit que les performances se stabilisent dès que l'on dépasse 15 ou 20 termes. Il est assez surprenant de voir, encore une fois, les performances obtenues par DfRes-NYT qui obtient les meilleurs résultats pour $15 \leq k \leq 100$. Même si la méthode DfRes combinant toutes les ressources n'obtient pas toujours les meilleurs résultats, on peut voir qu'elle est robuste dans les cas où très peu de termes sont ajoutés : pour 5 termes, elle obtiendra toujours de meilleurs résultats que les autres méthodes, et parfois même plus que MoRM utilisant 20 termes.

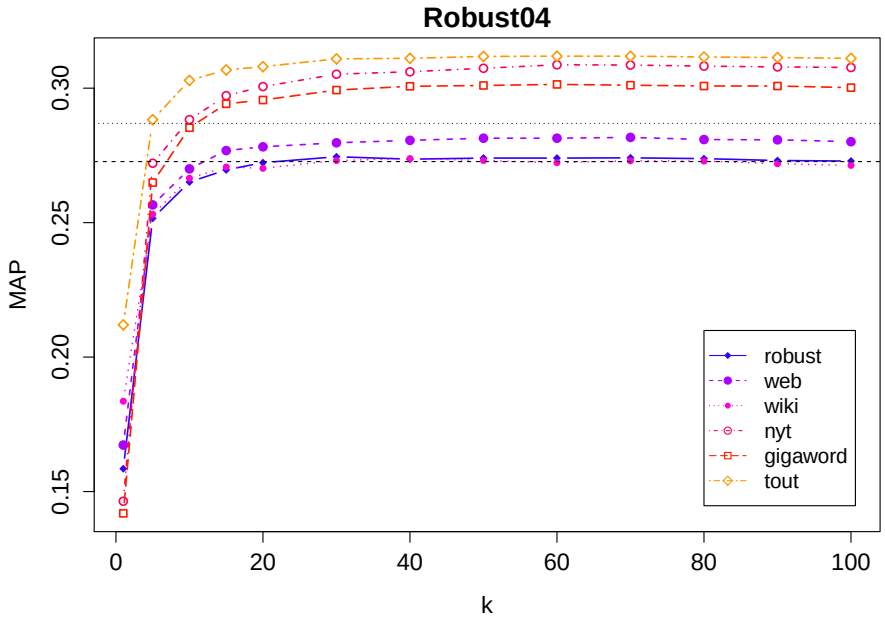

FIGURE 3.7 – *Performances (exprimées en MAP) en fonction du nombre de termes k utilisés pour estimer le modèle de langue de la source d'information sur la collection Robust04. La légende est identique à celle des figures précédentes (3.1 et suivantes).*

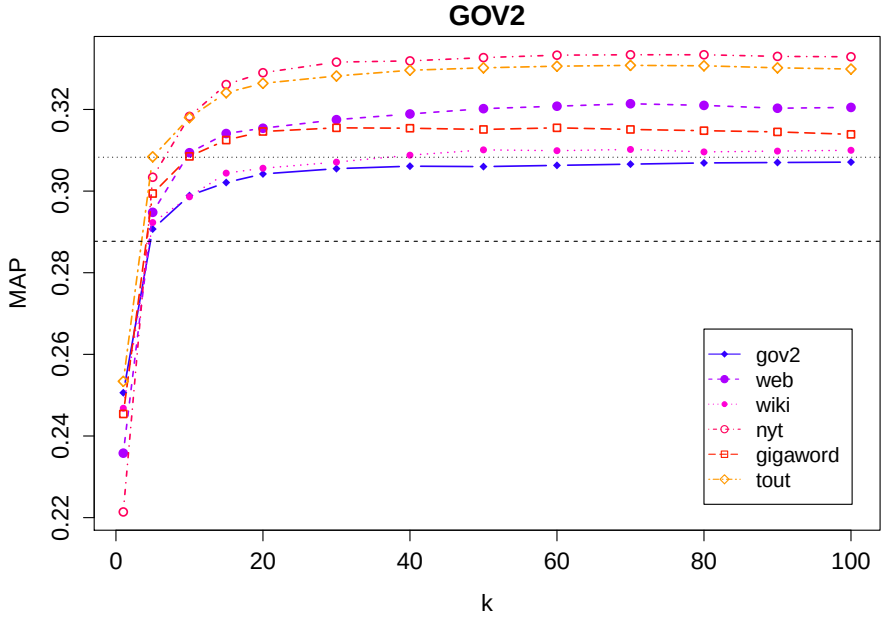

FIGURE 3.8 – *Performances (exprimées en MAP) en fonction du nombre de termes k utilisés pour estimer le modèle de langue de la source d'information sur la collection GOV2. La légende est identique à celle des figures précédentes (3.1 et suivantes).*

Les résultats obtenus pour la collection ClueWeb09-B (figure 3.9) sont malgré tout très différents des précédents. Tout d'abord, la courbe bleue dénotant les performances

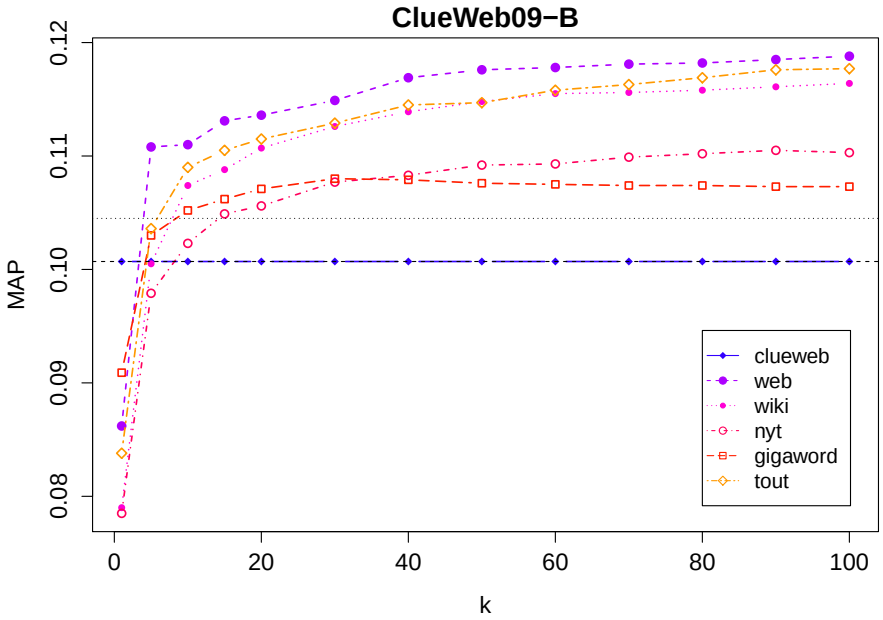

FIGURE 3.9 – *Performances (exprimées en MAP) en fonction du nombre de termes k utilisés pour estimer le modèle de langue de la source d'information sur la collection ClueWeb09-B. La légende est identique à celle des figures précédentes (3.1 et suivantes).*

de DfRes-ClueWeb est horizontale et n'a pas la même forme que les autres : pour toutes les requêtes, l'algorithme de validation croisée a estimé que fixer $\lambda = 1$ permettait d'obtenir les meilleures performances, et c'est effectivement ce que l'on observait déjà sur la figure 3.4. Ainsi, peu importe le nombre de termes ajoutés, la fonction de score ne tient compte que de la requête originale. Ensuite, nous voyons sur la figure 3.9 que, contrairement aux résultats reportés sur les précédentes collections, les performances augmentent de façon régulière au fur et à mesure que le nombre de termes augmente. La méthode DfRes-Web obtient les meilleurs résultats, même si les améliorations semblent se stabiliser vers $k = 80$. La ressource Web étant composée de pages Web contenant un faible taux de *spam*, cela peut expliquer ses bonnes performances.

Ces différents résultats nous montrent globalement qu'ajouter un grand nombre de termes ne dégrade pas les performances, comme on pourrait s'y attendre. En effet le risque en ajoutant des mots supplémentaires est de dériver de la thématique initiale de la requête, et ainsi de récupérer des documents non pertinents. La pondération à base d'entropie utilisée pour estimer le contexte thématique semble être efficace. Il pourrait être intéressant de voir comment se comporterait DfRes si l'on prenait tous les mots contenus dans les documents pseudo-pertinents. Ce genre d'approche pourrait poser des problèmes de comparaison entre les ressources (documents différents, nombre de mots différents), nous gardons donc cette piste d'amélioration pour des travaux futurs.

3.5.5 Robustesse du contexte thématique

Un des problèmes connus des approches faisant de l'enrichissement de requêtes est le potentiel glissement de thématique. En ajoutant des mots, on prend le risque de trop étendre le contexte thématique et de renvoyer des documents peu ou pas du tout pertinents par rapport à la requête. Nous présentons dans cette section une analyse de la robustesse de notre approche DfRes par rapport à une approche QL qui n'effectue aucun enrichissement.

Nous avons examiné, pour chaque requête, le changement absolu en précision moyenne (AP par requête) que DfRes apporte. Plus spécifiquement, nous calculons pour chaque requête Q :

$$\Delta_{AP}(Q) = AP_{DfRes}(Q) - AP_{QL}(Q) \tag{3.19}$$

Lorsque le résultat est négatif, c'est que l'estimation du contexte thématique que nous proposons a eu des conséquences négative sur les performances de recherche documentaire, lorsqu'il est positif c'est que cette estimation a eu été bénéfique. La figure 3.10 condense les valeurs de Δ_{AP} pour toutes les requêtes des quatre collections étudiées dans cette thèse. Chaque valeur correspond à une requête, une valeur positive indiquant que DfRes améliore la pertinence de la liste de documents renvoyés pour la requête concernée par rapport à QL, et une valeur négative indiquant que DfRes la dégrade.

Nous observons sur cette figure que, globalement, DfRes améliore plus de requêtes qu'elle n'en dégrade. Il y a néanmoins des cas, comme pour le WT10g par exemple, où les dégradations sont substantielles avec par exemple -0,583 de AP sur la requête n°485 « gps clock ». Les termes identifiés comme représentant le contexte thématique sont globalement orientés vers les appareils GPS et les satellites plutôt que sur les horloges ou le temps en général.

D'un autre côté, le nombre de requêtes dégradées est moins important pour les collections Robust04 et GOV2. Les gains de performance sont aussi élevés (+0,425 pour Robust04 et +0,363 pour GOV2) comparés aux plus fortes dégradations (-0,22 pour Robust04 et -0,159 pour GOV2). Pour la collection Robust04, la requête la plus dégradée par DfRes est la n°619 « winnie mandela scandal ». Alors que de nombreux termes concernent Nelson Mandela, l'apartheid et le gouvernement sud-africain, très peu concernent son ex-femme et les comportements criminels dont elle a été accusée et rendue coupable. Ce dernier cas est un exemple parfait de glissement thématique causé par un enrichissement de requête. Alors que l'utilisateur cherche des informations précises sur un sujet clairement identifié, l'enrichissement généralise un peu trop et se concentre sur un thème connexe, très proche, mais hors-thématique. C'est exactement ce comportement que l'on retrouve pour la requête n°831 « dulles airport security » qui obtient la plus forte dégradation sur la collection GOV2. Les termes ajoutés concernent tous des thématiques liées aux aéroports en général (passagers, compagnies aériennes...) ou à Washington, D.C. [9], mais il y en a très peu qui réfèrent à des notions de sécurité.

9. Dulles est un des aéroports de la ville de Washington, D.C.

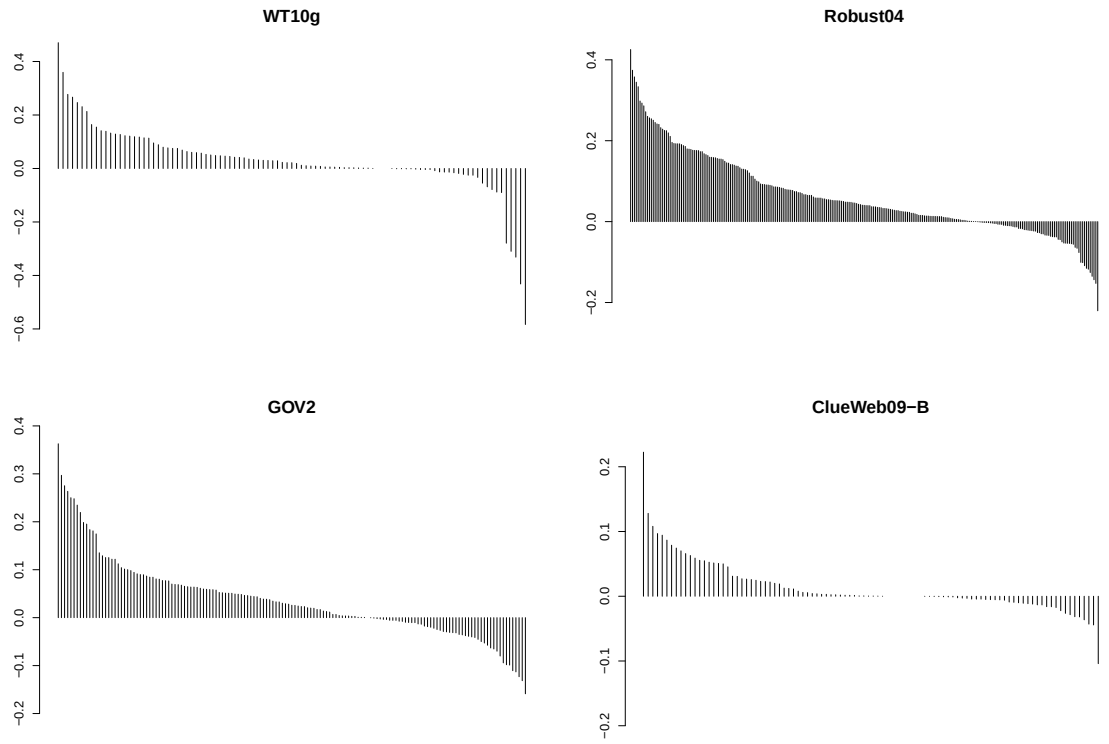

FIGURE *3.10* – *Robustesse de l'approche DfRes présentée dans ce chapitre par rapport au modèle standard de vraisemblance de la requête (QL). Chaque barre représente une requête ; elles sont ordonnées par ordre croissant suivant la AP améliorée.*

Enfin, les requêtes du ClueWeb09-B semblent être peu dégradées, mais elles ne sont pas autant améliorées que pour les autres collections. La plus haute amélioration se situe à +0,223 en AP tandis que la plus forte dégradation est de -0,104. Concernant cette dernière, il s'agît de la requête n°147 « tangible personal property tax ». Il est directement évident que cette requête est déjà très spécialisée par l'utilisateur et qu'il est difficile de l'améliorer en apportant des mots précisant ce contexte. L'approche DfRes identifie principalement des mots liés à la finance et aux taxes mais ignore complètement l'aspect « personnel » de la requête.

Pour conclure, nous avons vu dans cette section que notre méthode améliore globalement plus de requêtes qu'elle n'en dégrade, et que les améliorations sont plus importantes que les dégradations. Il y a néanmoins des cas où certaines requêtes sont très largement dégradées, principalement dans le cas où l'utilisateur cherche des informations déjà précises où le contexte thématique est déjà compris dans la requête. Dans ces cas là, l'estimation du contexte échoue car elle est trop générale.

Dans la prochaine section, nous discutons les résultats que nous avons présentés précédemment et mettons en perspective ce que nous avons appris du comportement de notre méthode.

3.5.6 Discussion

Les résultats reportés dans les sections précédentes ne sont pas surprenants et étayent les principes de *polyreprésentation* (Ingwersen, 1994) et de *redondance intentionnelle* (Jones, 1990). Ceux-ci affirment que combiner des représentations du besoin d'information structurellement et cognitivement différentes permet d'améliorer les chances de trouver des documents pertinents. Plus spécifiquement, la requête formulée par l'utilisateur agit comme une première représentation parfois incomplète. Les différentes estimations du contexte thématique que nous proposons en utilisant différentes sources d'information agissent comme autant de représentations différentes, souvent redondantes car liées aux mêmes thématiques, mais généralement suffisamment complètes pour améliorer significativement les résultats de recherche documentaire. Étant donné que nous utilisons plusieurs sources d'information de natures très différentes, allant des articles journalistiques aux pages web, la méthode DfRes tire parti de cette variété pour améliorer l'estimation globale et combinée du contexte thématique.

Nous avons également vu que, dans certains cas, la méthode DfRes obtenait de très bons résultats pour $\lambda = 0$, parfois largement supérieurs à ceux obtenus avec $\lambda = 1$. Ceci suggère que, dans certains cas, l'estimation du contexte thématique réalisée par DfRes est bien meilleure que celle induite par les mots-clés de la requête initialement formulée par l'utilisateur. Ces conclusions sont néanmoins à prendre avec précaution, étant donné que nous avons uniquement observé ce comportement sur deux des quatre collections de test étudiées. D'un autre côté, il se peut que nous n'ayons pas pu faire ces observations à cause de la grande taille de ces collections qui empêche d'avoir des jugements de pertinence exhaustifs. La très grande similarité des courbes présentées dans les sections précédentes serait un argument en faveur de cette hypothèse.

Nous observons aussi dans les précédentes sections que le New York Times est une ressource qui donne de très bons résultats lorsqu'elle est utilisée seule, malgré le fait qu'elle soit la plus petite de toutes. Pour les collections WT10g, Robust04 et GOV2, DfRes-NYT obtient les meilleurs résultats par rapport à toutes les autres ressources utilisées individuellement. Ceci peut être dû au fait que les articles du NYT ont été écrits par des journalistes professionnels, et contiennent ainsi beaucoup de synonymes afin d'éviter les répétitions. Dans notre cas, cela se traduit par un grand nombre de mots décrivant le contexte thématique et permettant ainsi de récupérer plus de documents liés à la requête même si celle-ci ne contient pas ces mots. De la même façon, les mauvaises performances de DfRes-Wiki peuvent être dues à la segmentation encyclopédique des articles. Chaque article Wikipédia couvre un concept ou une entité spécifique et détaille toutes les informations y référant. Il est ainsi probable qu'une partie des informations traitées soient trop spécifiques ou annexes par rapport à la requête et ne soient donc pas pertinentes. De plus, une requête se référant quant à elle à un nombre réduit de concepts thématiques, 10 articles Wikipédia paraît être un nombre déjà élevé pour estimer un contexte thématique ciblé et peut mener à des phénomènes de glissement thématique. Nous verrons notamment dans le chapitre suivant que, dans la plupart des cas, 2 ou 3 articles Wikipédia contiennent suffisamment d'informations pour identifier les concepts thématiques liés à une requête.

Une des originalités de la méthode DfRes est qu'elle peut automatiquement prendre en compte des n-grammes sans aucune supervision (comme par exemple fixer *a priori* la longueur n des n-grammes). En pratique, nous avons identifié qu'il y avait en moyenne 1,19 mots par terme utilisé, mais la plupart du temps des articles tels que « the » sont ajoutés à des mots qui étaient déjà sélectionnés. Par exemple, dans le cas où DfRes identifie les mots « nativity » et « scene » comme candidats importants, le terme « the nativity scene » est lui aussi considéré comme important.

3.6 Conclusions et perspectives

Nous avons présenté dans ce chapitre DfRes, une méthode permettant d'estimer le contexte thématique d'une requête utilisateur à l'aide de plusieurs sources d'informations, puis de l'enrichir. Nous avons montré que la méthode DfRes permettait d'obtenir de bons résultats de recherche documentaire en estimant un contexte thématique de qualité. Dans de nombreux cas, les résultats des requêtes sont grandement améliorés car aucun des mots de la requête n'était présent dans les documents pertinents, ce qui pouvait mener à des résultats nuls. Utiliser des sources d'information externes, et qui plus est les combiner, apporte une diversité de représentation thématique et de vocabulaire permettant de récupérer des documents liés et pertinents.

Néanmoins, nous avons vu que, dans certains cas, DfRes pouvait dégrader substantiellement les résultats par rapport à une approche simple utilisant uniquement les mots de la requête. Dans un scénario idéal, un nouveau modèle de Recherche d'Information ne doit dégrader les résultats d'aucune requête par rapport à un autre modèle, et nous pensons qu'il y a plusieurs pistes d'améliorations pour arriver à gommer ce manque de robustesse. Nous pourrions par exemple ajouter une étape d'apprentissage supervisé, où nous apprendrions différentes caractéristiques propres aux requêtes qui seront difficiles à améliorer ou qui seront dégradées si on choisit de faire de l'enrichissement. Les résultats obtenus dans cette thèse pourraient être un point de départ, et on pourrait ainsi imaginer un système choisissant de ne pas enrichir la requête originale dans le cas où il prévoit que celui-ci risque de dégrader la pertinence des résultats. Ce type d'approche répond de plus à des problématiques actuelles des moteurs de recherche commerciaux (Wang et al., 2012) et peut s'inscrire dans le cadre de nombreuses approches développées dans ce but (Collins-Thompson, 2009).

Une autre piste d'amélioration est le traitement des n-grammes redondants : les n-grammes sélectionnés par DfRes dont les mots qui les composent ont déjà été sélectionnés. Idéalement, nous pourrions imaginer de supprimer les unigrammes déjà sélectionnés, laissant ainsi plus de place pour d'autres n-grammes. Un problème potentiel est, encore une fois, le glissement thématique qui pourrait être introduit par une sélection trop large de n-grammes pouvant être hors-thématique. Encore une fois, une couche d'apprentissage supervisé similaire à celle proposée par Cao et al. (2008) pourrait être une solution dans ce cas-là.

Dans l'ensemble, les mots et termes que nous utilisons pour estimer le contexte thé-

matique de la requête font référence à un ou plusieurs concepts informatifs liés à cette requête. L'approche présentée dans ce chapitre ne peut pas définir une démarcation fixe entre ces concepts, ni ne peut pondérer les mots en fonction de l'adéquation des concepts aux thématiques de la requête. Dans la suite de cette thèse, nous nous concentrons à définir des approches permettant de modéliser les concepts implicites d'une requête, puis à étendre les modèles de pertinence traditionnels pour incorporer ces notions conceptuelles.

Chapitre 4

Modélisation des concepts implicites d'une requête

Sommaire

4.1 Introduction

Dans le chapitre précédent, nous avons introduit une méthode permettant d'estimer le contexte thématique à l'aide d'un ensemble de n mots censés représenter au mieux ce contexte. Cette méthode rentre plus spécifiquement dans la famille des approches « conceptuelles » pour la Recherche d'Information, où l'on cherche à développer un système capable d'enrichir la requête avec les mots les plus représentatifs des concepts associés à celle-ci. Ce type d'approche a reçu beaucoup d'attention au cours de ces dernières années (Chang et al., 2006; Bai et al., 2007; Metzler et Croft, 2007; Bendersky et al.,

2011; Egozi et al., 2011). L'idée générale est ainsi d'étendre les requêtes avec des ensembles de mots ou de multi-mots extraits de documents pseudo-pertinents, avec pour but d'obtenir ainsi une couverture conceptuelle importante. 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. 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.

La recherche par facettes (ou *Faceted Search*) (Tunkelang, 2009) a été une tentative pour prendre en compte cette vision ensembliste du besoin d'information. Dans cette approche, la collection de documents est entièrement classifiée selon un nombre prédéfini de dimensions, résultant en une taxonomie. Ainsi les utilisateurs peuvent naviguer au sein de cette hiérarchie de facettes afin de préciser leur domaine de recherche. Cette technique est notamment très utilisée dans les sites de commerce électronique tels qu'Amazon [1] où l'utilisateur peut restreindre la portée de sa requête à des catégories ou des sous-catégories fixes (Livres, Films, ...). Mais si ce type d'approches fonctionne bien dans le cadre de scénarios de recherche où les domaines sont fixés, la recherche en domaine ouvert où le contenu évolue à grande vitesse (Web, réseaux sociaux, ...) est bien plus problématique. La construction de ces taxonomies requiert un temps de calcul important et leur mise à jour peut représenter un défi.

L'objectif du travail présenté dans ce chapitre est de représenter avec précision les concepts sous-jacents associés à une requête, améliorant indirectement les informations contextuelles liées à la recherche documentaire. De plus, nous ne nous reposons pas sur des ontologies ou des taxonomies n'ayant pas de possibilité d'évolution, mais détectons ces concepts directement au sein des documents. 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 pseudo-pertinents initialement récupérés par le système. Tout comme dans le chapitre précédent, ces documents pseudo-pertinents 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 pseudo-pertinents nécessaires.

L'exemple présenté dans le Tableau 4.1 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

1. http ://www.amazon.com/

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
	...		...		...		...
$\hat{\delta}_0 = 0,434$		$\hat{\delta}_1 = 0,254$		$\hat{\delta}_2 = 0,021$		$\hat{\delta}_3 = 0,291$	

TABLE 4.1 – *Concepts identifiés pour la requête « dinosaurs » (topic 14 de la Web Track de TREC) par l'approche présentée dans ce chapitre. 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.*

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 pseudo-pertinents. 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* paraît 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 pseudo-pertinents 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.

Le travail présenté dans ce chapitre 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. Néanmoins nous nous concentrons ici spécifiquement sur l'approche de modélisation de ces concepts et sur l'évaluation de leur qualité. Le chapitre suivant couvre quant à lui les problèmes liés à la recherche d'information utilisant ces concepts.

La suite de ce chapitre est organisée comme suit. La section 4.2 présente rapidement l'allocation de Dirichlet latente, puis détaille l'approche que nous proposons pour estimer automatiquement le nombre de concepts implicites ainsi que le nombre de documents. Nous proposons différentes évaluations et analyses, et discutons du comportement de notre approche dans la section 4.3. Pour finir, la section 4.4 conclut ce chapitre et offre quelques perspectives introduisant le prochain chapitre.

4.2 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 pseudo-pertinents les mieux classés par rapport à une requête Q lors d'une première étape de recherche. Le modèle 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 pseudo-pertinents 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 pseudo-pertinents doit être choisie pour s'assurer que les concepts extraits sont effectivement liés à la requête ? En d'autres 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 certains 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.

4.2.1 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 en estimant deux distributions de probabilités : une distribution des thèmes sur les documents, et une distribution des mots sur les thèmes. Les documents de la collection sont ainsi modélisés comme des ensembles de K thèmes qui sont eux-mêmes des distributions multinomiales sur W.

Plus spécifiquement, 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 termes, $\theta_{D,k}$ 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. $P(w|k)$). La figure 4.1 détaille la notation graphique du modèle LDA et la relation entre les différents paramètres, les distributions apprises (i.e. variables latentes) et les variables observées.

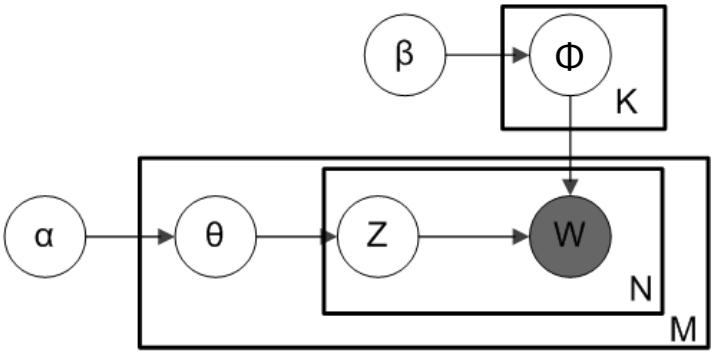

FIGURE 4.1 – *Représentation graphique (en plates) de LDA selon Blei et al. (2003). La variable observable est représentée par un cercle gris, tandis que les autres variables sont latentes.*

Dans la suite de cette thèse, nous notons $P_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)$ [2] la probabilité qu'un mot w appartienne à un thème k, sachant un modèle LDA appris avec K thèmes latents sur M documents. De la même façon, nous notons $P_{TM}(k|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)$ la probabilité qu'un thème k apparaisse dans un document D.

Il a été montré que le modèle LDA était trop complexe pour qu'une solution exacte soit calculable. Différentes méthodes d'approximation ont été développées, les plus célèbres étant l'algorithme d'inférence variationnelle présenté dans les travaux initiaux de Blei et al. (2003) et l'échantillonnage de Gibbs appliqué pour la première fois au modèle LDA par Griffiths et Steyvers (2004). Tout au long de cette thèse et pour toutes les expérimentations que nous menons dans ce chapitre et dans le suivant, nous utilisons l'algorithme d'inférence variationnelle implémenté et mis à disposition librement par le Pr. Blei [3].

4.2.2 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

2. TM pour *Topic Model*, la version anglaise de « modélisation thématique ».
3. http://www.cs.princeton.edu/~blei/lda-c

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 argmax$[n]$ qui produit les n arguments obtenant les plus fortes valeurs pour une fonction donnée. Nous pouvons ainsi obtenir l'ensemble $\mathcal{W}_k$ des n mots qui ont les plus fortes probabilités dans le concept k :

$$\mathcal{W}_k = \underset{w}{\text{argmax}}[n]\ P_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \tag{4.1}$$

Dans ce travail, nous fixons le nombre de mots appartenant à un concept à $n = 10$. En effet, représenter un tel concept par ses dix mots les plus probables et une pratique commune qui apporte habituellement suffisamment de détail pour exprimer le sujet du concept, et de distinguer les concepts entre eux (Newman et al., 2010).

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 pseudo-pertinents $\mathcal{R}_Q$, 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 toutes 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 :

$$\hat{K} = \underset{K}{\text{argmax}}\ \frac{1}{K(K-1)} \sum_{i=1}^{K} \sum_{j=i+1}^{K} D(k_i||k_j) \tag{4.2}$$

où K est le nombre de concepts donné en paramètre pour apprendre le modèle LDA. 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) [4]. Nous utilisons la version symétrique de la

4. Rechercher des documents en les classant par leur divergence de Kullback-Leibler par rapport au modèle de la requête est d'ailleurs une généralisation du modèle classique de recherche par modèle de langue détaillé dans le chapitre précédent (vraisemblance de la requête).

divergence de Kullback-Leibler afin d'éviter des problèmes évidents lors du calcul de divergences entre toutes les paires de concepts :

$$D(k_i||k_j) = \sum_{w \in \mathcal{W}_{inter}} P_{TM}(w|k_i, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \log \frac{P_{TM}(w|k_i, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)}{P_{TM}(w|k_j, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)} + \sum_{w \in \mathcal{W}_{inter}} P_{TM}(w|k_j, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \log \frac{P_{TM}(w|k_j, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)}{P_{TM}(w|k_i, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)} \tag{4.3}$$

où $\mathcal{W}_{inter} = \mathcal{W}_{k_i} \cap \mathcal{W}_{k_j}$. La sortie finale de cette première étape est le nombre estimé de concepts implicites $\hat{K}$, et indirectement l'ensemble de concepts $\mathcal{T}_{\hat{K}}$ qui lui est associé. Nous définissons cet ensemble de concepts comme étant un *modèle conceptuel* de la requête.

4.2.3 Combien de documents pseudo-pertinents ?

Un problème récurrent avec les approches à base de retour de pertinence simulé est que des documents non pertinents peuvent être inclus dans les documents pseudo-pertinents. 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. De nombreuses approches ont tenté de proposer une solution au problème de la sélection de bons documents candidats pour le retour de pertinence simulé. Ces approches sont très variées, allant de l'estimation d'un modèle génératif estimant conjointement les mots et les documents à utiliser pour l'expansion (Tao et Zhai, 2006) jusqu'à l'apprentissage de classifieurs robustes prédisant si un document va être un candidat efficace pour faire partie de l'ensemble des documents pseudo-pertinents (He et Ounis, 2009; Keikha et al., 2011).

Ne travaillant pas directement sur les mots mais au niveau des concepts, nous prenons ici une approche différente : au lieu de sélectionner des documents en fonction de leur pertinence ou de leur qualité estimée, nous nous attachons à sélectionner le *modèle conceptuel* le plus représentatif de la requête. Étant donné qu'un *modèle conceptuel* est appris à partir d'un ensemble fixe de documents pseudo-pertinents, on peut apprendre plusieurs modèles sur différents ensembles puis estimer leur qualité. Nous faisons malgré tout une hypothèse forte sur ces documents pseudo-pertinents : l'ordre dans lequel ils ont été renvoyés par une première passe d'un système de RI état-de-l'art est important. 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 pseudo-pertinents 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 pseudo-pertinents doit donc être automatique pour chaque requête. Dans ce but, nous comparons les modèles conceptuels générés à partir de différents nombres m de documents pseudo-pertinents. 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 pseudo-pertinents discutent de concepts similaires ou liés, peu importe le nombre de documents.

Il est important de rappeler que ce que nous appelons *modèle conceptuel* est un modèle thématique appris par LDA avec un nombre de concepts automatiquement estimé par la méthode présentée en section 4.2.2. Potentiellement, augmenter le nombre m de documents pseudo-pertinents devrait également faire augmenter nombre de concepts implicite $\hat{K}$: en effet, ajouter des documents reviendrait à ajouter des concepts. Ainsi, des concepts apparaissant dans différents modèles appris sur différents ensembles de documents pseudo-pertinents sont certainement liés à la requête, tandis que des concepts bruités ont peu de chances d'apparaître à chaque fois. Nous avons vu dans le chapitre précédent qu'un faible nombre de documents pseudo-pertinents contiennent un grand nombre d'informations sur le contexte thématique de la requête. Ces informations sont concentrées et sont généralement similaires, puisque liées à la requête. Ainsi, si on ajoute un nouveau document et qu'apparaît alors un concept qui n'était auparavant présent dans aucun autre document, le nouveau document contient vraisemblablement des informations n'étant pas liées aux thématiques de la requête. Finalement, le meilleur modèle conceptuel est celui qui contient les concepts *les plus similaires* par rapport aux autres modèles.

Différents modèles conceptuels sont ainsi appris sur les m premiers documents pseudo-pertinents, et nous faisons varier m. Nous estimons la similarité entre deux modèles conceptuels en calculant les similarités entre toutes les paires de concepts des deux modèles conceptuels. Seulement, deux modèles différents sont générés à partir d'ensembles de documents pseudo-pertinents différents : ils ne partagent pas le même vocabulaire ni les mêmes documents, leurs espaces probabilistes sont entièrement différents. Les distributions de probabilités apprises par LDA ne sont donc pas comparables et ne peuvent pas être utilisées de la même façon que dans la section 4.2.2. Le calcul de similarité entre deux modèles conceptuels ne peut donc se faire qu'en prenant en compte les mots des concepts. Les concepts sont ramenés à de simples sacs de mots sans pondération, et nous utilisons une mesure de similarité basée sur la fréquence inverse des mots dans les documents de la collection cible :

$$sim(\mathcal{T}_{\Theta_m}^{\hat{K}(m)}, \mathcal{T}_{\Theta_n}^{\hat{K}(n)}) = \frac{1}{\eta} \sum_{k \in \mathcal{T}_{\Theta_m}^{\hat{K}(m)}} \sum_{k' \in \mathcal{T}_{\Theta_n}^{\hat{K}(n)}} \frac{|k \cap k'|}{|k|} \sum_{w \in W} \log \frac{N}{df_w} \tag{4.4}$$

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. $\mathcal{T}_{\Theta_m}^{\hat{K}}$ et $\mathcal{T}_{\Theta_n}^{\hat{K}}$ sont les modèles conceptuels appris sur les ensembles de documents pseudo-pertinents Θ_m et Θ_n, respectivement constitués de m et n documents. Il est à noter que les nombres $\hat{K}(m)$ et $\hat{K}(n)$ peuvent être (et sont souvent) différents, mais ce n'est pas un problème. Le facteur η permet d'effectuer une normalisation et donné par $\eta = \hat{K}(m) \times (\hat{K}(n) - 1)$.

Le but initial de cette mesure basée sur la fréquence inverse des mots était la détection de la nouveauté (i.e. minimisation de la similarité) entre deux phrases (Metzler et al., 2005), 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 $\hat{M}$ premiers documents pseudo-pertinents sont utilisés pour modéliser les concepts, où :

$$\hat{M} = \underset{m}{\operatorname{argmax}} \sum_{n} sim(\mathcal{T}_{\Theta_m}^{\hat{K}(m)}, \mathcal{T}_{\Theta_n}^{\hat{K}(n)}) \tag{4.5}$$

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 sont donc peu sensibles aux problèmes de complexité algorithmique. Nous proposons une expérience traitant de ce problème dans la section 4.3.5.

4.2.4 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 :

$$\delta_k = \sum_{D \in \Theta} P(Q|D) P_{TM}(k|D, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}}) \tag{4.6}$$

où Θ est l'ensemble des $\hat{M}$ documents pseudo-pertinents et $P(Q|D)$ est la probabilité donnée par la vraisemblance de la requête [5].

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 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

5. Pour plus de détails, revenir à la section 3.2.1.

donné par :

$$P'_{TM}(w|k, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}}) = \frac{P_{TM}(w|k, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}})}{\sum_{w' \in \mathcal{W}_k} P_{TM}(w'|k, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}})} \tag{4.7}$$

où $\mathcal{W}_k$ est l'ensemble des mots du concept k tel que défini dans la section 4.2.2. 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.

4.3 Expériences et analyses

Nous présentons dans cette section les différentes expériences que nous avons menées afin d'évaluer la qualité des concepts générés et de valider nos hypothèses. Comme nous l'avons vu dans le chapitre précédent, nous pouvons utiliser différentes sources d'information afin de récupérer des documents pseudo-pertinents, et notamment des sources externes. Nous commençons dans cette section par proposer quelques analyses reposant uniquement sur la collection cible comme source d'information, afin de rester dans le cadre standard des approches de retour de pertinence. Nous proposons néanmoins par la suite des analyses complémentaires pour des méthodes utilisant les différentes sources d'information externes présentées en section 2.4.

4.3.1 Analyse des nombres de concepts et de documents pseudo-pertinents estimés

Les méthodes que nous présentons dans ce chapitre permettent de générer différents modèles conceptuels comprenant des concepts différents, à partir d'ensembles variables de documents pseudo-pertinents. Dans la section précédente, nous avons fait l'hypothèse (réaliste) que, plus on augmente le nombre de documents, plus on va augmenter le nombre de concepts. Ainsi, si notre méthode d'estimation du nombre de concepts présentée en section 4.2.2 est efficace, le nombre estimé devrait augmenter de la même façon qu'augmente le nombre de documents.

Nous présentons dans cette section une première analyse du nombre de concepts estimé par notre méthode en fonction du nombre de documents. Plus spécifiquement, nous estimons la valeur $\hat{K}$ pour chacune des requêtes de nos quatre collections de test et pour chaque ensemble de documents pseudo-pertinents (allant du premier aux 20 premiers), et nous comptons le nombre de requêtes obtenant les mêmes valeurs. Les documents pseudo-pertinents proviennent tous de la collection cible.

Les résultats présentés dans la figure 4.2 montrent une corrélation très claire entre le nombre de concepts et le nombre de documents. Pour toutes les collections, la grande majorité des requêtes peuvent être reliées à un nombre variable mais faible de concepts (entre 2 et 5), qui sont eux-mêmes modélisés à partir d'un ensemble réduit de documents pseudo-pertinents (entre 2 et 8). On observe également un effet de dispersion du

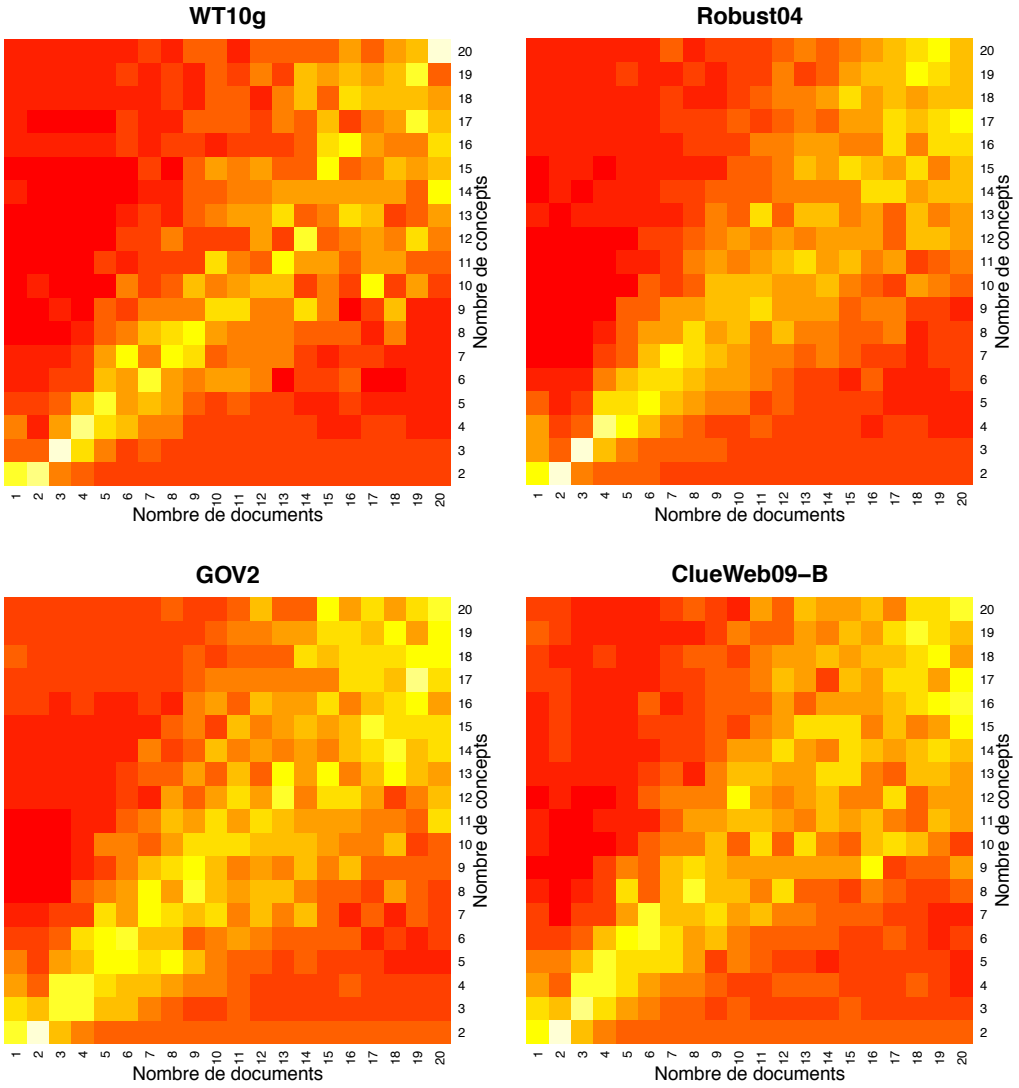

FIGURE 4.2 – *Nombre de requêtes pour lesquelles $\hat{K}$ concepts en fonction de différents nombres de documents pseudo-pertinents. Un carré jaune tirant vers le blanc indique un grand nombre de requêtes tandis qu'un carré rouge indique qu'aucune requête n'est associée aux valeurs correspondantes.*

nombre de concepts estimé quand le nombre de documents vient à augmenter. Cet effet de dispersion, opposé à l'effet de cohésion observé quand le nombre de documents pseudo-pertinents $m \leq 10$, nous apporte beaucoup d'informations sur les concepts contenus dans les premiers documents pseudo-pertinents. Les systèmes de RI étant optimisés pour maximiser (entre autres) la précision à faible rang, il est bien connu que les tout premiers documents renvoyés ont une plus forte tendance à être pertinents que les autres. C'est cette tendance que l'on observe pour $m \leq 10$: les documents traitent plus ou moins tous des mêmes thématiques, et logiquement les mêmes concepts peuvent y être rattachés. À l'opposé, augmenter le nombre de documents pseudo-pertinents aug-

mente les chances d'intégrer des documents traitant d'informations plus larges ou de thématiques connexes, ou encore des documents non pertinents. C'est ce qui est représenté par ce cône de dispersion. La partie basse du cône est relative aux cas où de nombreux documents pseudo-pertinents traitent de sujets très similaires (un nombre de documents m haut et un nombre de concepts $\hat{K}$ bas), tandis que la partie haute est relative aux cas où de très nombreux sujets sont abordés par peu de documents (un m bas et un $\hat{K}$ haut), avec une plus forte propension au bruit. Cet effet peut également être expliqué par le fait que certaines requêtes sont ciblées sur des besoins d'information précis pour lesquels peu de documents peuvent être très liés, le reste des documents pseudo-pertinents renvoyés sont ainsi beaucoup plus larges et ne contiennent pas forcément des concepts centrés sur la requête.

Il est également intéressant de voir que le cône de dispersion est beaucoup plus important pour les collections GOV2 et ClueWeb09-B. Comme nous l'avons déjà vu dans le chapitre précédent, il semble que ces deux collections de taille importante se comportent différemment des deux autres, plus petites. Ici, la taille de ces collections joue un rôle sur la récupération de documents pseudo-pertinents et on peut voir que de nombreux concepts sont traités par peu de documents. La collection Robust04 est, quant à elle, entièrement à l'opposé. On peut en effet voir une très forte corrélation entre le nombre de concepts et le nombre de documents, avec presque un nouveau concept pour chaque document. Pour rappel, la collection Robust04 est constituée d'articles journalistiques qui contiennent généralement du texte centré sur des sujets d'actualités bien précis.

Bien que nous montrons que notre méthode permet d'estimer avec précision un nombre de concepts cohérent par rapport à différents nombres de documents pseudo-pertinents et à la nature de différentes collections, nous ne savons pas si ce nombre est réellement représentatif d'un « bon » nombre de concepts liés à la requête. Nous proposons dans la section suivante une deuxième expérimentation visant à explorer la qualité de l'estimation de ce nombre.

4.3.2 Corrélation du nombre de concepts estimé avec une modélisation thématique hiérarchique

À ce point là de l'évaluation, nous n'avons aucun moyen de dire si notre méthode permet effectivement d'identifier les « bons » concepts implicites de la requête. L'approche que nous proposons est entièrement non-supervisée, c'est à dire qu'elle apprend à partir des données et non à partir d'un ensemble d'entraînement étiqueté. Pour chaque requête, nous ne connaissons donc pas *a priori* le nombre de concepts, et nous n'avons *a posteriori* qu'une estimation dont on ne connait pas la précision.

Une première solution pour évaluer la qualité de la modélisation aurait été de faire un travail d'étiquetage manuel pour chaque requête individuellement. Il aurait fallu ainsi identifier et comprendre les informations liées à la requête, en extraire des concepts, puis les comparer avec ceux générés par notre méthode. Les concepts que nous générons sont des sacs de mots possédant des liens thématiques, mais ces liens se révèlent

à travers l'interprétation du cerveau humain. Cette évaluation aurait ainsi été très subjective et aurait demandé un investissement conséquent afin d'être menée à bien.

Nous avons donc décidé de comparer le nombre de concepts identifié par notre méthode avec celui identifié par une méthode de modélisation thématique hiérarchique. Plus précisément, nous utilisons les processus de Dirichlet hiérarchiques (HDP pour *Hierarchical Dirichlet Processes*) (Teh et al., 2006), un algorithme généralisant LDA et permettant d'attribuer des poids aux concepts modélisés. Nous avons donc récupéré le texte des documents pseudo-pertinents utilisés pour identifier les concepts de chaque requête, et nous avons construit les modèles thématiques hiérarchiques correspondants. Un des attraits du modèle HDP souvent avancé pour justifier son utilisation est le fait qu'il soit non-paramétrique, c'est-à-dire qu'il ne nécessite pas qu'on lui précise un nombre de concepts en paramètre. Or, ce paramètre est toujours nécessaire pour pouvoir définir la dimension de la loi de Dirichlet régissant la distribution des mots sur les concepts. Le modèle HDP est ainsi en réalité paramétrique mais nous ne considérons que les x concepts de plus forts poids (au-dessus d'un certain seuil). Afin de pouvoir s'affranchir du paramètre définissant le nombre de concept, le modèle HDP a donc tout de même besoin d'un autre paramètre déterminant à partir de quelle valeur un concept n'est plus assez important pour être considéré dans le modèle. Dans cette expérience, nous notons ce seuil t et nous fixons empiriquement $t = 0,05$. Ainsi, nous ignorons tous les concepts dont le poids estimé par le modèle HDP est inférieur à 0,05.

1	2	3	4		20
2	6	9	5	...	16

(a) *Nombre de concepts estimé par notre méthode.*

1	2	3	4		20
4	4	4,7	4,1	...	5,2

(b) *Nombre de concepts estimé par HDP.*

FIGURE 4.3 – *Exemple des nombres de concepts estimés par notre méthode et par HDP, pour différents ensembles de documents pseudo-pertinents (allant de 1 à 20). Ce sont les vraies valeurs obtenues pour la requête n°550 de la collection WT10g : « volcanoes made ». La corrélation, en utilisant le coefficient de corrélation de Kendall, est de $\tau = 0,514$.*

De cette façon, nous avons une méthode automatique permettant d'identifier et de quantifier des concepts que nous pouvons comparer avec notre propre approche. Pour chaque requête de chaque collection, nous avons donc mesuré la corrélation qu'il y avait entre les nombres de concepts estimés par les deux méthodes. La figure 4.3 donne un exemple de cette méthodologie, où le taux de corrélation de Kendall est égal à $\tau = 0,514$ ce qui est assez haut pour ce test. Les nombres de concepts estimés par la méthode HDP ne sont pas des nombres entiers car son initialisation est aléatoire, les résultats peuvent donc changer entre deux exécutions de la même modélisation. Afin de pallier ce problème nous construisons 10 modèles HDP pour chaque ensemble de documents

pseudo-pertinents, puis nous prenons la moyenne des nombres de concepts.

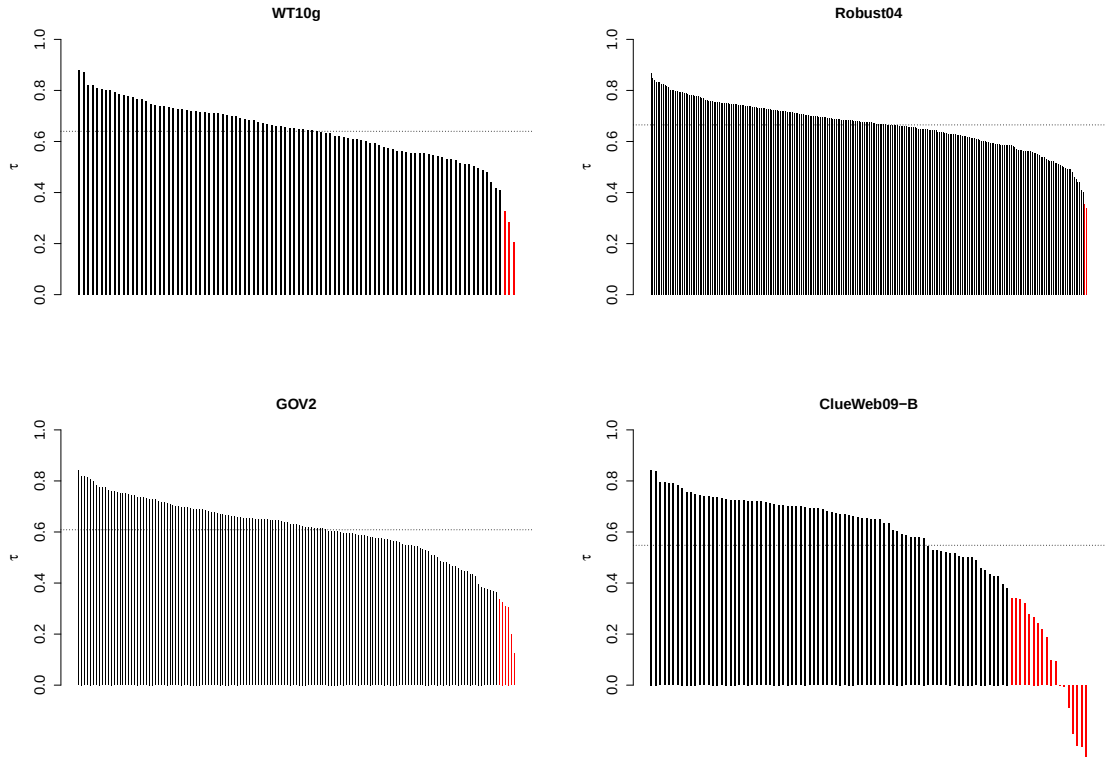

FIGURE 4.4 – *Coefficient de corrélation de Kendall τ entre le nombre de concepts estimé par la méthode présentée en section 4.2.2 et des processus de Dirichlet hiérarchiques (avec un seuil $t = 0,05$), pour chaque requête de chaque collection. Les corrélations représentées par des barres noires sont statistiquement significatives (niveau de confiance de 95%), tandis que les barres rouges indiquent qu'il n'y a pas de corrélation statistiquement significative. Les lignes pointillées représentent les corrélations moyennes. Les requêtes sont ordonnées par leur corrélation décroissante.*

Nous reportons sur la figure 4.4 les résultats des taux de corrélation de Kendall pour chaque requête prise individuellement, et ce pour les quatre collections de test. Parallèlement, nous reportons également les corrélations moyennes dans le tableau 4.2.

Nous pouvons tout d'abord voir qu'il y a très peu de requêtes pour lesquelles les corrélations ne sont pas significatives. Nous remarquons encore une fois que la collection ClueWeb09-B est un peu à part par rapport aux autres collections, avec des taux de corrélation très bas pour 18% des requêtes. Les taux de corrélations obtenus pour ses autres requêtes sont malgré tout proches de ceux obtenus pour les autres collections.

Ces résultats sont globalement très bons et confirment que notre méthode est capable d'estimer un nombre de concepts réaliste et corrélé avec celui issu du modèle HDP. Mais malgré ces corrélations, il y a néanmoins de grandes différences dans les nombres estimés. La méthode HDP modélise par exemple toujours entre 4 et 6 concepts, peu importe le nombre de documents pseudo-pertinents utilisés. La plupart du temps l'utilisation d'un seul document entraîne une modélisation proche de 4 concepts tan-

dis que l'utilisation de 20 documents entraîne une modélisation proche de 6 concepts. Nous avons tenté de modifier le seuil t mais cela ne change guère ces résultats. La méthode HDP semble ainsi identifier des concepts très « plats » ayant peu de consistance lorsqu'on utilise un faible nombre de documents, et des concepts trop généralistes lorsqu'on utilise un plus grand nombre de documents. En revanche, comme nous l'avons vu dans la section précédente et comme le laisse entrevoir l'exemple de la figure 4.3, notre méthode s'adapte à la quantité d'information exprimée dans les documents, et le nombre de concepts augmente de façon linéaire au fur et à mesure qu'on en rajoute.

	ρ	τ
WT10g	0,763	0,640
Robust04	0,782	0,665
GOV2	0,754	0,609
ClueWeb09-B	0,657	0,548

TABLE 4.2 – *Corrélations exprimées en fonction du taux ρ de Pearson et du taux τ de Kendall.*

4.3.3 Cohérence sémantique des concepts implicites de la requête

Nous continuons ici notre série d'expériences et étudions la cohérence sémantique des concepts modélisés. Les algorithmes de modélisation thématique tels que LDA sont généralement appliqués sur des larges collections contenant plusieurs dizaines de milliers de documents, à partir desquelles on cherche habituellement à modéliser plusieurs centaines de thèmes. Dans notre cas, nous agissons sur des ensembles très réduits où les informations contenues dans les documents sont centrées sur les thématiques de la requête. Nous pouvons donc nous poser la question de la qualité des concepts que nous générons. De plus, notre approche repose sur deux paramètres importants : le nombre de concepts et le nombre de documents pseudo-pertinents. Nous avons vu précédemment que les variations de ces paramètres permettaient de capturer plus ou moins d'information et de modéliser des concepts plus ou moins fins et précis, mais quel est l'impact sur leur cohérence sémantique ?

La première étape vers une évaluation de la cohérence sémantique d'un ensemble de termes a été de mesurer la similarité entre termes dans des domaines restreints (Gliozzo et al., 2007), et aura été une première base pour le développement de plusieurs mesures d'évaluation de cohérence des concepts générés par les modèles thématiques (Newman et al., 2010). Plus précisément, le calcul du score PMI (*Pointwise Mutual Information*) de toutes les paires de mots composant le concept en utilisant Wikipédia comme corpus de référence a permis d'obtenir les meilleurs résultats, montrant que ce score est le plus discriminant parmi tous ceux que les auteurs ont évalués. Les expériences qui ont mené à ces conclusions n'étaient néanmoins constitués que d'articles journalistiques et de livres. Formellement, le calcul du score PMI entre deux mots s'écrit :

$$PMI(w, w') = log \frac{P(w, w')}{P(w)P(w')} \tag{4.8}$$

où les probabilités sont calculées au sein d'un corpus de référence, ici Wikipédia. Dans notre cas, nous utilisons la même version de Wikipédia que celle utilisée comme source d'information dans le chapitre précédent.

Plus récemment, Stevens et al. (2012) ont appliqué (entre autres) une version agrégée de cette mesure afin d'évaluer la cohérence de grands nombres de modèles thématiques. Nous utilisons directement cette méthode afin d'évaluer la cohérence des concepts implicites de la requête. Plus spécifiquement, la cohérence d'un modèle conceptuel $\mathcal{T}_\Theta^K$ composé de K concepts identifiés au sein d'un ensemble Θ de documents pseudo-pertinents est donnée par :

$$c(\mathcal{T}_\Theta^K) = \frac{1}{K} \sum_{i=1}^{K} \sum_{(w,w') \in k_i} \log \frac{P(w,w') + \epsilon}{P(w)P(w')} \tag{4.9}$$

Cette métrique est en réalité la cohérence moyenne des différents concepts du modèle conceptuel. Nous suivons la formulation de Stevens et al. (2012) et ajoutons un paramètre ϵ à la formule du score PMI, et nous le fixons à $\epsilon = 1$.

D'après l'équation (4.9), les concepts les plus cohérents devraient être composés de mots rares qui apparaissent rarement dans le corpus de référence, mais qui co-occurrent souvent. Nous avons donc mesuré cette cohérence sémantique pour tous les modèles conceptuels générés pour les différentes requêtes de nos quatre collections. Nous avons pris des valeurs remarquables pour les nombres de concepts $K \in \{3, 5, 10, 15, 20\}$ et de documents pseudo-pertinents $N \in \{5, 10, 20, 30, 40, 50\}$. Nous reportons les résultats de cette expérience dans la figure 4.5.

Nous pouvons observer que les concepts très cohérents sont identifiés dans les 5 et 10 premiers documents pseudo-pertinents pour la collection WT10g, ce qui indique que des documents très similaires sont renvoyés dans les premiers rangs. La cohérence semble s'atténuer au fur et à mesure que l'on rajoute des documents. Sachant que les documents les plus pertinents sont renvoyés dans les premiers rangs, nous pourrions ainsi logiquement en conclure que les concepts les plus cohérents apparaissent dans les documents les plus probablement pertinents. Seulement, nous observons des comportements totalement différents sur les autres collections.

Sur la collection Robust04, les résultats sont très particuliers. On voit que la cohérence des modèles conceptuels contenant peu de concepts (3 et 5) décroit de façon monotone quand le nombre de documents augmente, tandis que c'est le phénomène inverse qui se produit pour les modèles conceptuels composés d'un plus grand nombre de concepts (20, 15 dans une moindre mesure). La cohérence d'un modèle de 20 concepts calculé sur 5 documents est ainsi très réduite. Nous voyons en explorant le contenu des documents pseudo-pertinents que ceux de la collection Robust04 sont en moyenne deux fois plus petits que les pages web de la collection WT10g. Ainsi, quantité de documents égale, les modèles conceptuels calculés pour les requêtes de la collection WT10g sont basés sur un plus grand nombre de mots, et sont potentiellement plus fins. Néanmoins, les articles de la collection Robust04 permet de modéliser des concepts de qualité, même lorsqu'on augmente les nombres de documents et de concepts utili-

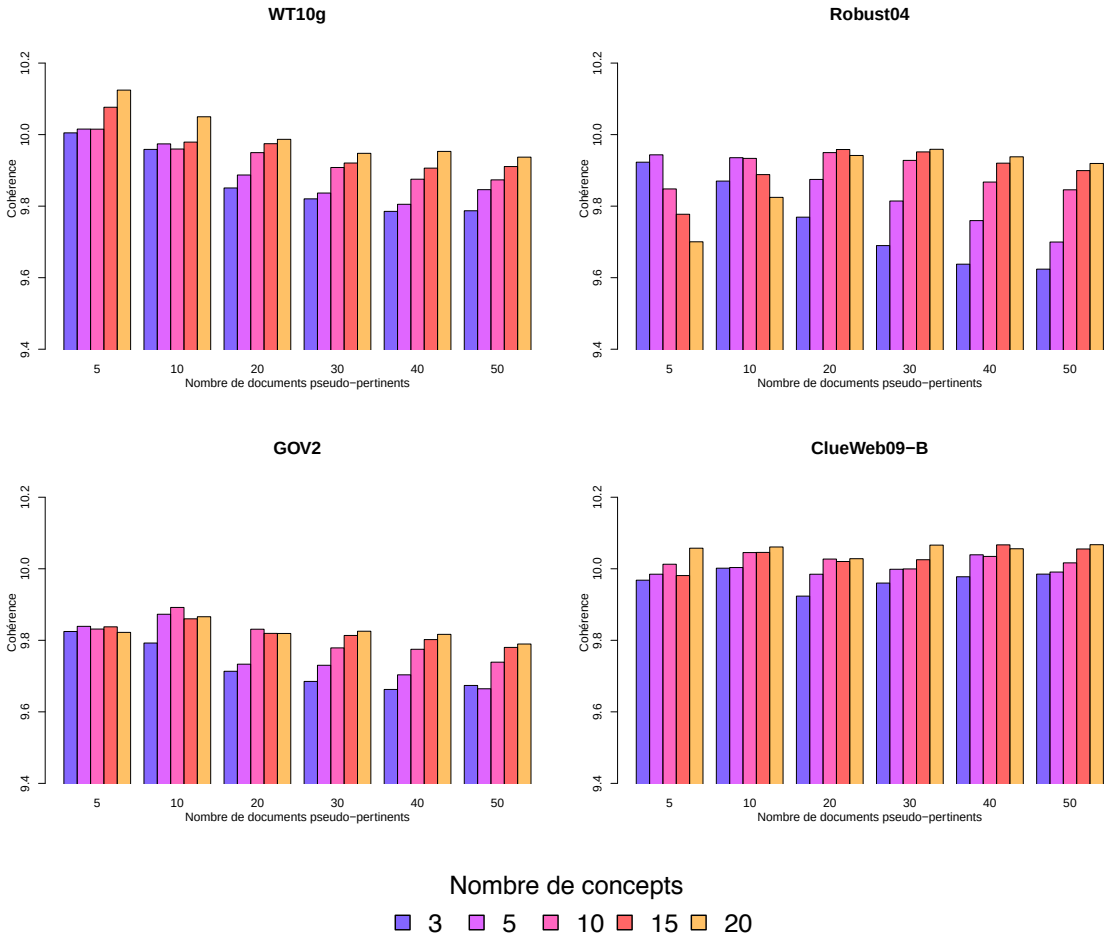

FIGURE 4.5 – *Cohérence sémantique des modèles conceptuels pour différents nombres de concepts K, en fonction du nombre N de documents pseudo-pertinents. Les valeurs de cohérence sont obtenues en faisant la moyenne sur toutes les requêtes. Les échelles de valeurs sont identiques pour les quatre collections.*

sés pour apprendre le modèle. Ce n'est par exemple pas le cas du WT10g, pour lequel ajouter plus d'articles va toujours dégrader la qualité des concepts.

Pour les collections GOV2 et ClueWeb09, les variations de cohérence semblent moins importantes. GOV2 modélise les concepts les moins cohérents, mais nous pensons que ceci est dû à la différence entre la collection et le corpus de référence utilisé pour calculer la cohérence. En effet GOV2 est composé de documents web issus du domaine gouvernemental américain (ainsi que de documents PDF, postscript, ...) dont le vocabulaire est loin de correspondre à celui employé dans les pages Wikipédia. Ainsi, les mesures de co-occurrence de mots donnent des résultats inférieurs pour cette collection. Inversement, le ClueWeb09-B obtient les plus hauts scores de cohérence en valeur absolue par rapport aux autres collections. Varier le nombre de documents pseudo-pertinents semble avoir peu d'effet sur la cohérence des concepts modélisés pour cette collection, alors que celle-ci décroit un peu pour un grand nombre de documents et un

faible nombre de concepts sur GOV2.

Globalement, les modèles conceptuels les plus stables contiennent un nombre raisonnable de concepts (entre 10 et 15), et ne sont pas sensibles au nombre de documents utilisés pour la modélisation. Il semble tout de même que les plus cohérents soient d'une façon générale ceux qui possèdent le nombre maximum de concepts. Ceci peut s'expliquer par le fait qu'utiliser plus de concepts permet d'introduire plus de mots, et ainsi stabiliser la mesure de cohérence. Celle-ci étant une simple moyenne de la cohérence de tous les concepts d'un modèle, elle peut être sensible à la variabilité, lorsque certains concepts sont très peu cohérents par exemple. Utiliser un grand nombre de concepts permet ainsi de réduire cet impact. Nous laissons néanmoins l'étude de ces variations pour des travaux futurs.

Le lecteur attentif remarquera que les scores de cohérence que nous obtenons sont bien plus élevés que ceux reportés dans la littérature (Stevens et al., 2012), qui tournent autour de 4 et 5 (alors que les nôtres varient entre 9,5 et 10). Notre approche capture des concepts qui sont centrés autour d'un besoin d'information très spécifique, souvent avec un vocabulaire limité, qui favorise la détection de co-occurrences de mots, ce qui peut expliquer ces scores plus élevés. D'un autre côté, les scores précédemment reportés sont ceux de modèles appris sur des collections entières avec un nombre de concepts réduits. Ces concepts sont donc logiquement plus généraux et d'un plus haut niveau que ceux modélisés par notre méthode, qui sont très spécifiques et liés à des besoins d'information précis. Des concepts généraux sont susceptibles d'être composés de mots apparaissant peu ensemble, ce qui mène à des scores de cohérence moins importants que ceux que l'on peut obtenir avec des concepts ciblés.

4.3.4 Sources d'information pour l'identification de concepts

L'approche présentée dans ce chapitre nécessite une source d'information à partir de laquelle les concepts peuvent être extraits. Dans toutes les expériences précédentes, nous n'avons considéré comme source que la collection cible, comme dans les approches traditionnelles de retour de pertinence. Néanmoins n'importe quelle source d'information externe peut être utilisée sans changement dans les algorithmes ou la procédure. Dans cette section, nous explorons la modélisation de concepts implicites en utilisant les différentes sources d'information présentées en section 2.4 et que nous retrouvons tout au long de cette thèse. Celle-ci ont notamment l'avantage d'être 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 des documents de chaque source sur l'identification des concepts.

Nous avons donc appris des modèles conceptuels sur les quatre sources d'information externes (Wikipédia, NYT, Gigaword, Web) et la collection cible pour toutes les requêtes de nos collections. Nous reportons ici, pour chaque collection, le nombre de requêtes en fonction du nombre $\hat{K}$ de concepts et du nombre $\hat{M}$ de documents pseudo-pertinents utilisés pour chacune des sources d'information.

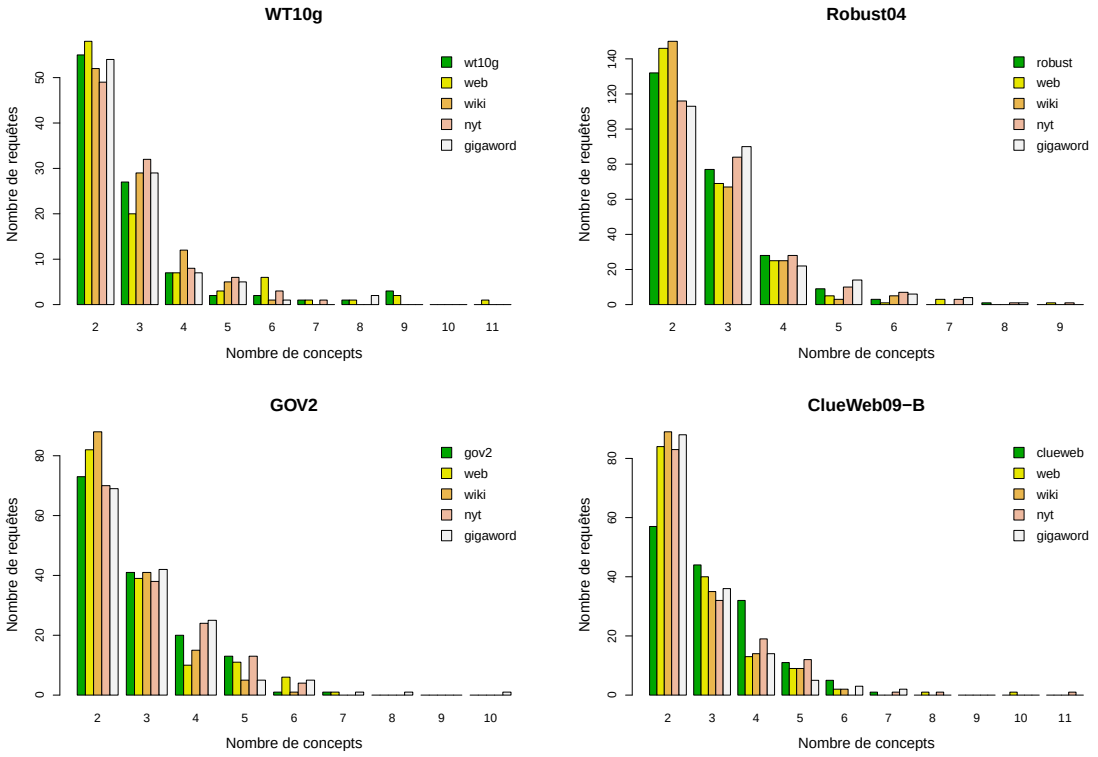

FIGURE 4.6 – *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{K}$ de concepts implicites (section 4.2.2).*

La figure 4.6 présente des histogrammes traçant le nombre de requêtes en fonction du nombre de concepts implicites estimé et du nombre de documents pseudo-pertinents, et ce pour les deux collections. On voit que le comportement est relativement identique sur les quatre collections. Entre deux et trois concepts sont identifiés pour la grande majorité des requêtes. On peut noter toutefois une tendance des collections cibles utilisées comme sources d'information (les barres vertes) à identifier un plus grand nombre de concepts en moyenne. Ceci est d'autant plus vrai sur le ClueWeb09-B, où la source Clueweb modélise deux ou trois concepts à une fréquence presque identique. C'est un résultat intéressant, surtout quand on le compare à la ressource Web qui, nous le rappelons, est une version allégée sans documents spammés de la ressource Clueweb. Il semble donc que les documents bruités contiennent globalement plus de concepts qui sont probablement peu pertinents. Utiliser Wikipédia permet de modéliser dans presque tous les cas un nombre plus réduit de concepts. Comme nous l'avons déjà développé dans le chapitre précédent, nous pensons que ce comportement est dû à la segmentation encyclopédique de ses articles, où les concepts sont très clairement délimités.

Nous remarquons également sur les figures 4.7, 4.7, 4.9 et 4.10 que les concepts implicites sont généralement identifiés au sein d'un nombre assez réduit de documents. Pour la grande majorité des requêtes, entre 1 et 5 documents suffisent pour identifier

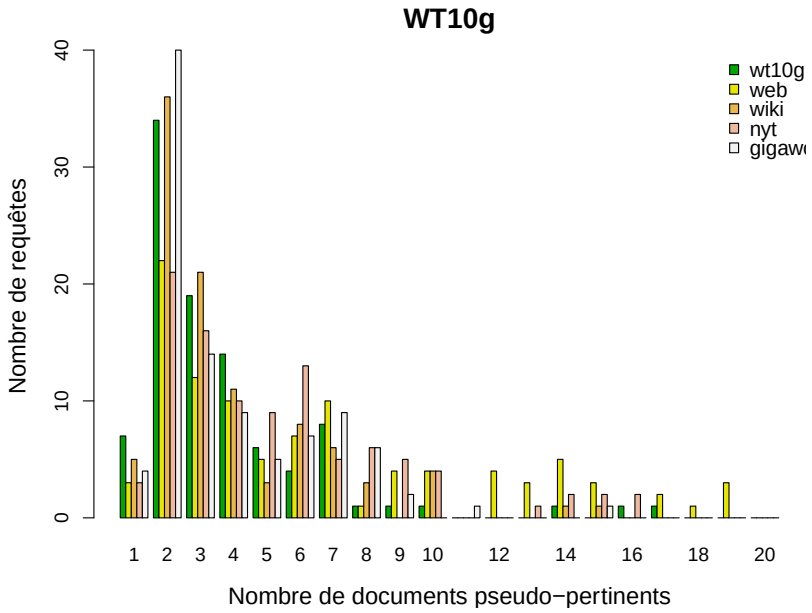

FIGURE 4.7 – *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{M}$ de documents pseudo-pertinents (section 4.2.3) pour la collection WT10g.*

les concepts, et moins de 10% ont recours à plus de 10 documents. Ce comportement est récurrent pour toutes les collections et confirme la justesse de notre méthode. He et Ounis (2009) précisent en effet dans leur étude que l'information contenue dans les 10 premiers documents pseudo-pertinents n'était pas significativement différente de l'information contenue dans les documents uniquement jugés comme pertinents parmi ces 10. Des différences apparaissaient néanmoins si l'on considérait un nombre plus important de documents. Ici, notre méthode semble automatiquement favoriser des nombres réduits de documents afin de capturer l'information pertinente, tout en évitant de modéliser des concepts potentiellement bruités et non pertinents.

Il est également intéressant de noter la différence entre le nombre de documents pseudo-pertinents 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. C'est d'ailleurs la ressource Wikipédia qui utilise le plus fréquemment 2 documents pseudo-pertinents sur toutes les collections sauf le WT10g, pour lequel c'est le Gigaword. Comme nous l'avons déjà précisé précédemment, 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 (beaucoup de sous-sujets abordés) soit coupé en plusieurs autres traitant chacun d'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 $p < 0,01$ obtenue par un test de permutations). Ces corrélations sont également cohérentes avec les observations faites en section 4.3.1 où l'on voyait un lien très clair entre le nombre de concepts et le nombre de documents utilisés pour les estimer.

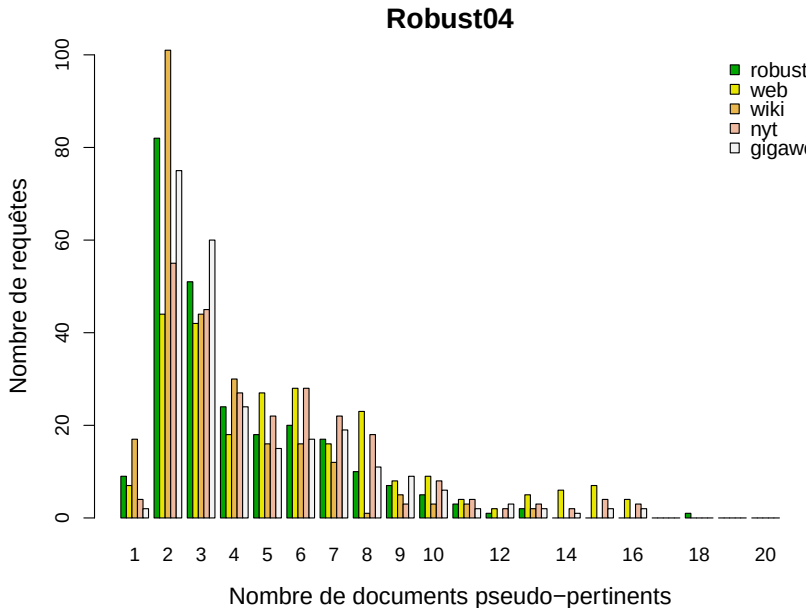

FIGURE 4.8 – *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{M}$ de documents pseudo-pertinents (section 4.2.3) pour la collection Robust04.*

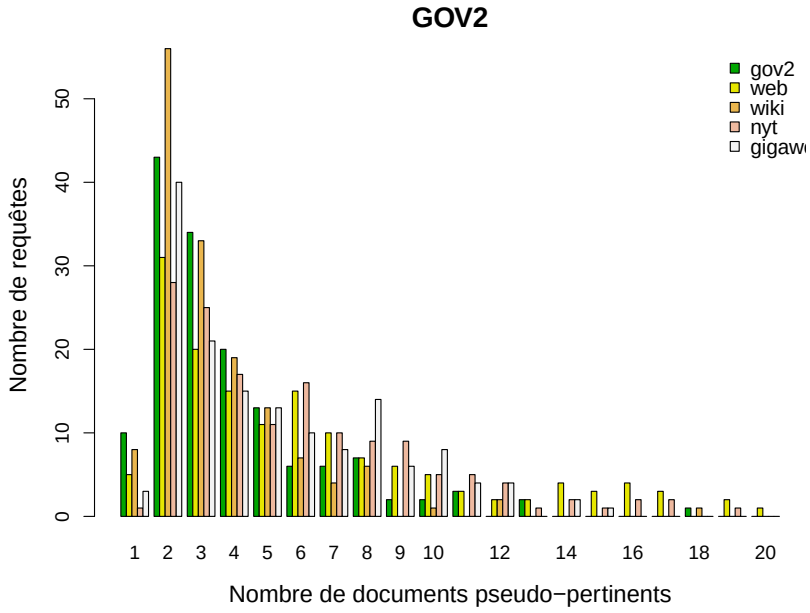

FIGURE 4.9 – *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{M}$ de documents pseudo-pertinents (section 4.2.3) pour la collection GOV2.*

À l'opposé, le nombre de documents pseudo-pertinents utilisés par la ressource Web est plus étalé et se concentre moins dans les 2 ou 3 premiers. Les pages Web sont en effet par nature très hétérogènes, contiennent potentiellement des publicités, peuvent être des blogs traitant de multiples sujets... Notre méthode est malgré tout

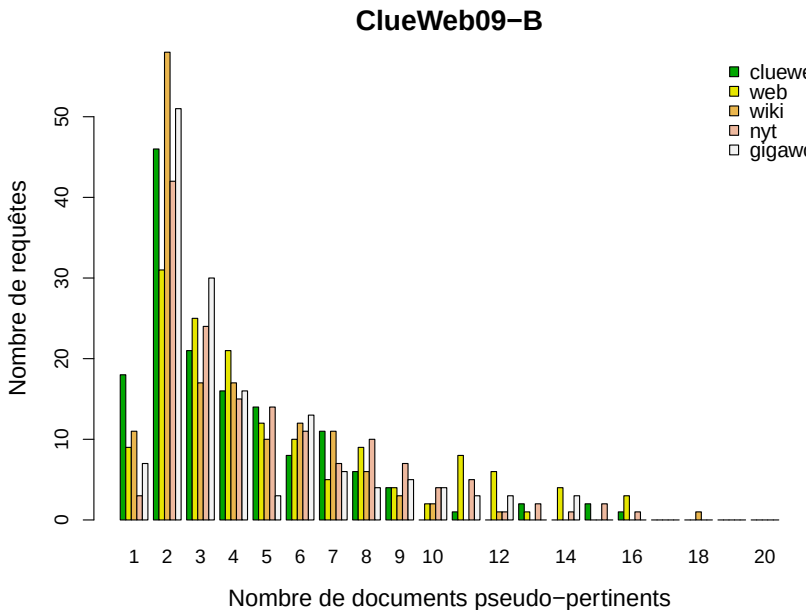

FIGURE 4.10 – *Histogrammes présentant le nombre requêtes en fonction du nombre $\hat{M}$ de documents pseudo-pertinents (section 4.2.3) pour la collection ClueWeb09-B.*

robuste à cette hétérogénéité et permet de s'adapter automatiquement aux différentes ressources, en utilisant un plus grand nombre de documents dans le cas de la ressource Web par exemple. 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. Une corrélation entre le nombre de concepts et le nombre de documents 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.

4.3.5 Temps d'exécution

Comme nous l'avons déjà évoqué dans la section 4.2.3, la méthode de modélisation de concepts implicites d'une requête que nous présentons dans ce chapitre fait appel de très nombreuses fois à LDA, qui est connu pour sa complexité algorithmique et son temps de calcul. Nous proposons ici une petite expérience permettant d'avoir une idée du temps de calcul nécessaire pour effectuer cette modélisation. Nous avons mesuré le temps que prend celle-ci pour toutes les requêtes des collections WT10g et Robust04 en utilisant les collections cibles comme sources d'information. Nous reportons les moyennes de ces temps sur les différentes requêtes dans la figure 4.11, en fonction du nombre de concepts et du nombre de documents pseudo-pertinents utilisés.

Nous pouvons tirer deux conclusions évidentes de ces expériences, la première étant que le nombre de concepts et le nombre de documents utilisés dans la modélisation ont une influence significative sur le temps d'exécution. On voit sur les deux graphiques

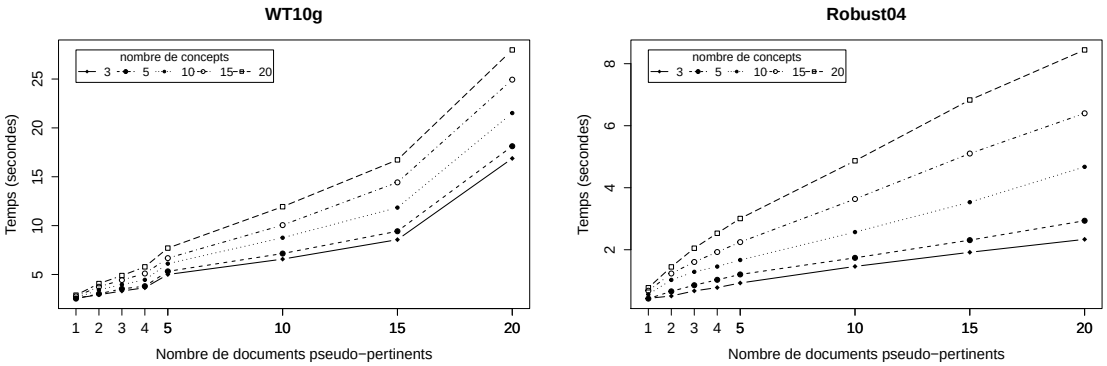

FIGURE 4.11 – *Temps d'exécution (en secondes) en fonction du nombre de documents pseudo-pertinents pour différents nombres de concepts.*

qu'utiliser un petit nombre de concepts prendra toujours moins de temps, tandis qu'utiliser un plus grand nombre de concepts en prendra toujours plus. De la même façon, le temps d'exécution augmente presque linéairement pour la collection Robust04 lorsqu'on augmente le nombre de documents. Concernant la collection WT10g, on remarque une plus forte augmentation lorsque l'on passe de 15 à 20 documents, mais cela peut être dû à une quantité de texte plus importante nécessitant ainsi plus de calculs.

Cette hypothèse nous mène à la seconde conclusion : la taille des documents utilisés a un impact considérable sur le temps d'exécution. Nous avions en effet précisé dans la section 4.3.3 que les documents pseudo-pertinents issus de la collection Robust04 étaient en moyenne deux fois plus petits que ceux de la collection WT10g. En observant la figure 4.11, nous remarquons que la modélisation est environ 2,5 fois plus lente pour les requêtes de la collection WT10g que pour celles de la collection Robust04, ce qui nous laisse penser que la principale raison de cette grande différence est la taille des documents considérés.

Pour finir, même si effectuer une telle modélisation en temps réel dans le contexte d'un moteur de recherche interactif semble compliqué par rapport au temps d'exécution pouvant dépasser les 30 secondes dans les cas extrêmes, de grands progrès peuvent être réalisés dans cette direction. Tout d'abord, nous n'avons pas tiré parti de la parallélisation sur de multiples processeurs, le calcul concurrentiel permettrait ainsi de grandement réduire ces temps d'exécution. De plus, nous nous basons sur la version originale de l'algorithme LDA connu pour sa complexité algorithmique, utiliser des logiciels tels que gensim [6] (Řehůřek et Sojka, 2010) et tirer parti de la puissance de l'échantillonnage de Gibbs serait une deuxième piste. Enfin, nous avons vu dans la section précédente que pour plus de 90% des requêtes, notre méthode utilise 10 documents pseudo-pertinents ou moins pour la modélisation des concepts. Une dernière piste d'amélioration du temps d'exécution serait ainsi la limitation à 10 ou 15 documents pseudo-pertinents.

6. http://radimrehurek.com/gensim

4.4 Conclusions et perspectives

Nous avons présenté dans ce chapitre 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 approprié de documents pseudo-pertinents sont automatiquement estimés au moment de l'exécution de la requête, sans apprentissage supervisé ni étiquetage préalable.

Alors que la méthode que nous proposons pour estimer le nombre de concepts implicites semble corrélée avec une modélisation thématique hiérarchique, nous avons vu que notre méthode tirait notamment parti de la quantité d'information présente dans les documents et augmentait en conséquence le nombre de concepts identifiés. Les modèles conceptuels ainsi générés restent également sémantiquement cohérents même lorsqu'on augmente le nombre de documents pseudo-pertinents. Tout comme dans le chapitre précédent, nous avons pu observer que les concepts issus de différentes sources d'information possèdent des caractéristiques très différentes. Tandis que les informations conceptuelles semblent être diluées parmi plusieurs documents sur le Web, elles sont au contraire très concentrées dans un petit nombre de documents quand il s'agit de Wikipédia. De la même façon, les concepts identifiés dans les documents encyclopédiques sont peu nombreux car ces derniers sont très ciblés sur des sujets spécifiques. Notre méthode présente néanmoins de nombreuses limitations que nous prévoyons d'étudier dans de futurs travaux, telle qu'une stratégie de repli lors de l'identification de concepts non pertinents.

Cette approche de modélisation de concepts liés à une requête pourrait notamment 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. Dans le prochain chapitre, nous prenons le parti d'introduire une méthode automatique et laissons l'aspect interactif comme une facette pouvant être explorée dans des travaux futurs. Plus spécifiquement, nous proposons une méthode originale permettant d'améliorer l'estimation de modèles de pertinence en utilisant les concepts ainsi modélisés.

Chapitre 5

Modèles de pertinence conceptuels

Sommaire

5.1 Introduction

Représenter les documents comme un ensemble de « concepts » ou de « thèmes » a toujours été un objectif et un défi pour les chercheurs travaillant dans les champs de recherche liés au traitement automatique du texte. L'utilisation de ressources telles que WordNet (Miller, 1995) ou plus récemment DBpedia (Lehmann et al., 2013) a permis notamment d'associer des concepts précis et complets à des mots ou des séquences de mots. Comme nous l'avons vu précédemment, un problème inhérent aux ontologies ou aux taxonomies est leur coût de construction et leur faible capacité d'évolution.

D'un autre côté, les algorithmes de modélisation thématique peuvent apprendre des relations thématiques entre les mots d'un ensemble de documents, en se basant sur l'hypothèse que ces documents traitent d'un nombre fini de concepts. L'apprentissage des différentes thématiques traitées par une collection de documents peut aider à extraire des informations sémantiques de haut niveau, et ainsi aider les humains

à comprendre le sens des documents et quelles informations ils couvrent réellement. L'indexation sémantique latente (ou *Latent Semantic Indexing*, LSI) (Deerwester et al., 1990), l'analyse sémantique latente probabiliste (ou *probabilistic Latent Semantic Analysis*, pLSA) (Hofmann, 2001) et l'allocation latente de Dirichlet [1] (ou *Latent Dirichlet Allocation*, LDA) (Blei et al., 2003) sont les approches les plus célèbres qui ont abordé ce problème au fil des ans. Les concepts et les thématiques produits par ces méthodes sont généralement attirants et de bonne qualité, et sont souvent corrélés avec les concepts humains (Chang et al., 2009). C'est une des raisons de l'utilisation intensive des algorithmes de modélisation thématique (et particulièrement LDA) au sein des recherches menées actuellement dans les domaines liés au Traitement Automatique des Langues (TAL).

Un des principaux problèmes de la Recherche d'Information *ad hoc* que nous tentons d'adresser dans cette thèse est la difficulté qu'ont les utilisateurs à traduire un besoin d'information potentiellement complexe en une requête formée de mots-clés. Alors que nous avons présenté dans le chapitre précédent une approche permettant de modéliser avec précision les concepts implicites d'une requête, nous explorons dans ce chapitre leur apport pour la recherche documentaire. Nous présentons ainsi une approche tirant parti des algorithmes de modélisation thématique et des modèles de pertinence (Lavrenko et Croft, 2001), où l'on va enrichir la requête avec les concepts modélisés *à partir* des documents pseudo-pertinents.

De nombreuses études se sont concentrées sur l'amélioration de la qualité des méthodes de classement des documents en utilisant des algorithmes de modélisation thématique, et plus particulièrement des méthodes probabilistes. L'approche proposée par Wei et Croft (2006) a été la première à tirer parti des concepts identifiés par LDA pour améliorer l'estimation des modèles de langue des documents, et a obtenu de bons résultats expérimentaux. 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. Suivant ce travail novateur, plusieurs études ont exploré l'utilisation de pLSA et de LDA dans différents cadres expérimentaux de RI (Park et Ramamohanarao, 2009; Yi et Allan, 2009; Andrzejewski et Buttler, 2011; Lu et al., 2011). Parmi ces études, certaines se basent sur les conclusions de Wei et Croft (2006) et proposent d'autres méthodes de lissage des modèles de langue ou d'appariement requête-document (Lu et al., 2011). D'autres approches essayent quant à elles d'enrichir directement la requête avec les mots qui appartiennent à ces concepts pseudo-pertinents (Park et Ramamohanarao, 2009; Yi et Allan, 2009; Andrzejewski et Buttler, 2011). L'idée d'utiliser des documents pseudo-pertinents a été explorée par Andrzejewski et Buttler (2011), où des concepts spécifiques à la requête sont extraits des deux premiers documents renvoyés par retour de pertinence simulé en utilisant 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 pseudo-pertinents. Globalement, les résultats reportés pour toutes ces méthodes

1. Voir section 4.2.1.

suggèrent que les mots et les distributions de probabilité apprises par les modèles thématiques probabilistes sont efficaces dans le cadre d'un enrichissement de requête.

Le principal inconvénient de ces approches est que les concepts sont appris sur la collection cible entière avant l'étape de récupération de documents. La représentation conceptuelle de la collection est ainsi statique pour toutes les requêtes, et est composée d'un nombre prédéfini de concepts. Suivant la requête et suivant sa spécificité, les concepts peuvent ainsi être trop généraux ou trop ciblés pour représenter précisément les concepts implicites réels de la requête. Plus récemment, Ye et al. (2011) ont proposé une méthode qui utilise elle aussi LDA et apprend les concepts directement sur les documents pseudo-pertinents. Tandis que cette approche est une première étape vers la modélisation de concepts liés à la requête, elle manque de lien avec la théorie des modèles de pertinence et ne cherche qu'à identifier le « meilleur » concept avant d'enrichir la requête avec les mots les plus probables de ce dernier.

Dans ce chapitre, nous abordons ces problèmes et proposons des solutions à travers les contributions suivantes :

- nous introduisons les modèles de pertinence conceptuels (que nous abrégeons en TDRM, pour *Topic-Driven Relevance Models*), une approche intégrant les concepts appris par modélisation thématique au sein des modèles de pertinence traditionnels (Lavrenko et Croft, 2001; Zhai et Lafferty, 2001). Ces concepts sont appris *uniquement à partir* des documents pseudo-pertinents, au lieu de la collection entière,
- nous intégrons l'approche présentée dans le chapitre précédent afin d'obtenir des modèles de pertinence conceptuels adaptatifs, pour lesquels les nombres de concepts et le nombre de documents pseudo-pertinents utilisés peuvent varier,
- nous adaptons également l'approche présentée dans le chapitre 3 afin d'intégrer les concepts modélisés à partir de plusieurs sources d'information.

Comme dans les chapitres précédents, nous évaluons les différentes méthodes en utilisant nos quatre collections de test habituelles et présentons les résultats en section 5.3. Pour finir, nous clôturons ce chapitre par une conclusion et des perspectives en section 5.4.

5.2 Modèles de pertinence conceptuels

5.2.1 Modèles de pertinence

Nous avons introduit précédemment les modèles de pertinence dans notre premier chapitre de contribution (voir la section 3.2.2), nous ne reviendrons donc pas sur les détails mais sur leur principe général, qui nous permettra de mettre en perspective les contributions de ce chapitre-ci.

Pour rappel, les modèles de pertinence sont un enrichissement du modèle de la requête destiné à dépasser les limites d'un petit nombre de mots-clés entrés par l'utilisateur. Un ensemble Θ de documents pseudo-pertinents est formé à partir de la requête

originale et sert de base à l'estimation de ce modèle de pertinence. Celui-ci est finalement interpolé avec la requête originale, et son influence dans la nouvelle estimation de la requête est contrôlée par un paramètre λ. Ainsi, soit $\tilde{\theta}_Q$ le modèle original de la requête et $\hat{\theta}_Q$ le modèle de pertinence estimé à partir de l'ensemble Θ, l'estimation enrichie de la requête se note :

$$P(w|\theta_Q) = \lambda\ P(w|\tilde{\theta}_Q) + (1-\lambda)P(w|\hat{\theta}_Q) \tag{5.1}$$

Les détails de l'estimation du modèle de pertinence $\hat{\theta}_Q$ sont disponibles en section 3.2.2 pour la méthode RM3, ainsi qu'en section 3.4 pour notre première contribution.

5.2.2 Modèle thématique de la requête

L'estimation du modèle de pertinence $\hat{\theta}_Q$ constitue la première contribution de ce chapitre. Nous proposons de modéliser de façon explicite les concepts implicites liés à un besoin d'information, et de les utiliser pour améliorer la représentation de la requête. Le point important qui différencie principalement notre approche des autres études utilisant des algorithmes de modélisation thématique pour la Recherche d'Information est que nous modélisons les concepts directement *à partir* des documents pseudo-pertinents. Alors qu'une modélisation thématique globale effectuée sur une collection entière peut manquer de précision, l'avantage d'utiliser uniquement un faible ensemble de documents déjà liés thématiquement à la requête permet d'extraire des concepts fortement liés à la requête plutôt que des concepts vagues ou trop généraux. De plus, comme nous l'avons vu dans le chapitre précédent, le faible nombre de documents utilisés nous permet de réaliser cette modélisation à la volée au moment de l'exécution de la requête. Nous ne sommes donc pas dépendants d'ontologies ou autres ressources structurées, les concepts étant directement identifiés au sein du texte des documents pseudo-pertinents.

Nous reprenons les mêmes notations déjà utilisées tout au long de cette thèse, et considérons Θ comme étant l'ensemble de documents pseudo-pertinents à partir desquels les concepts implicites vont être extraits. L'algorithme de RI utilisé pour obtenir ces documents peut être de n'importe quelle sorte, le point important est que Θ soit une collection réduite qui contient les documents les mieux classés par un processus automatique de recherche documentaire. Nous pouvons également voir cette étape comme une réduction de la collection au domaine thématique de la requête.

Au lieu de voir Θ comme un ensemble de modèles de langue de documents qui contiennent (avec une certaine probabilité) des informations thématiques au sujet de la requête, nous prenons une approche de modélisation thématique probabiliste. Nous nous concentrons spécifiquement sur l'allocation latente de Dirichlet (LDA, présentée en section 4.2.1), puisque c'est actuellement l'algorithme le plus utilisé et le plus représenté. Néanmoins, tout comme notre approche n'est aucunement dépendante de l'algorithme de RI utilisé pour récupérer les documents pseudo-pertinents, nous pensons que nous pourrions utiliser d'autres algorithmes de modélisation thématique tels

que pLSA et obtenir des résultats comparables. Nous vérifierons cette hypothèse dans de futurs travaux.

Nous l'avons déjà vu plus en détails dans le précédent chapitre, LDA apprend deux distributions de probabilités multinomiales : une distribution $\boldsymbol{\theta}$ des concepts sur les documents et une distribution $\boldsymbol{\phi}$ des mots sur les concepts. Nous nous référons aux notations introduites dans la section 4.2.1 pour la suite de cette section. Dans ce cadre spécifique, nous calculons l'estimation conceptuelle du modèle de pertinence en utilisant l'équation suivante :

$$P(w|\hat{\theta}_Q) \propto \sum_{\theta_D \in \Theta} P(\theta_D) P_{TM}(w|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \prod_{t \in Q} P(t|\theta_D) \tag{5.2}$$

où $P_{TM}(w|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)$ est la probabilité que le mot w apparaisse dans le document D en utilisant les distributions multinomiales précédemment apprises. Soit $\mathcal{T}_\Theta^K$ un modèle conceptuel composé de K concepts appris sur un ensemble Θ de documents pseudo-pertinents, cette probabilité est donnée par :

$$P_{TM}(w|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) = \sum_{k \in \mathcal{T}_\Theta^K} P'_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \cdot P_{TM}(k|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \tag{5.3}$$

Où la probabilité $P'_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K)$ est directement issue de l'équation (4.7). Nous pouvons ainsi réécrire l'équation (5.2) comme suit :

$$\begin{aligned} P(w|\hat{\theta}_Q) &\propto \sum_{\theta_D \in \Theta} P(\theta_D) \sum_{k \in \mathcal{T}_\Theta^K} P_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) P_{TM}(k|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \prod_{t \in Q} P(t|\theta_D) \\ &\propto \sum_{\theta_D \in \Theta} P(\theta_D) P(w|\theta_D) \sum_{k \in \mathcal{T}_\Theta^K} P_{TM}(w|k, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) \delta_{k,D} \end{aligned} \tag{5.4}$$

La dernière simplification peut être faite suivant l'équation (4.6), définie dans le chapitre précédent, qui donne le poids global d'un concept. Ici nous l'adaptons légèrement afin qu'elle reflète le poids d'un concept dans le document D. On a ainsi :

$$\delta_{k,D} = P_{TM}(k|D, \boldsymbol{\theta}_M, \boldsymbol{\phi}_K) P(Q|D) \tag{5.5}$$

Il est important de noter que ce poids $\delta_{k,D}$ n'est pas uniquement issu des distributions apprises par LDA, mais qu'il intègre aussi la probabilité que le document D soit lié à la requête Q. Ainsi, nous nous assurons que les concepts sont réellement importants par rapport à la requête, et non pas uniquement représentatifs au sein de l'ensemble de documents pseudo-pertinents. Dans ces modèles de pertinence conceptuels, les mots sont ainsi pondérés en fonction de leur probabilité d'apparition dans chaque document pseudo-pertinent et de leur probabilité d'appartenance à chaque concept modélisé. L'importance de chaque concept vis-à-vis de la requête est également présente au sein du modèle.

Dans la suite de cette thèse, nous nous référons à cette approche générale par l'acronyme TDRM pour *Topic-Driven Relevance Models*.

5.2.3 Modèles de pertinence conceptuels adaptatifs

Les modèles de pertinence conceptuels que nous venons d'introduire sont dépendants de deux paramètres principaux : le nombre de documents pseudo-pertinents utilisés pour identifier les concepts, et le nombre de concepts à identifier. Nous avions précisément proposé dans le chapitre précédent une méthode permettant d'estimer conjointement ces deux paramètres de façon à sélectionner le modèle conceptuel présentant les concepts les plus disjoints et les moins bruités. Nous considérons donc ici des modèles de pertinence conceptuels adaptatifs, où le nombre de concepts et l'ensemble de documents pseudo-pertinents sont estimés en fonction de la requête et au moment de son exécution, au lieu d'être simplement fixés aux mêmes valeurs pour toutes les requêtes.

La formulation de ces modèles adaptatifs est très similaire à celle présentée dans l'équation générale (5.2), avec quelques modifications mineures :

$$P(w|\hat{\theta}_Q) \propto \sum_{\theta_D \in \Theta} P(\theta_D) P_{TM}(w|D, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}}) \prod_{t \in Q} P(t|\theta_D) \tag{5.6}$$

où $|\Theta| = \hat{M}$. Ainsi, le modèle conceptuel $\mathcal{T}_{\Theta}^{\hat{K}}$ est appris suivant les méthodes présentées dans le chapitre 4. Dans les sections suivantes, nous nous référons à cette approche par l'acronyme ATDRM pour *Adaptive Topic-Driven Relevance Models*.

5.2.4 Combinaison de modèles de pertinence conceptuels

La première contribution de cette thèse, présentée dans la section 3.4, portait sur une combinaison de différentes sources d'information pour améliorer l'estimation du contexte thématique de la requête. Cette contribution comprenait également l'introduction d'un modèle de pertinence prenant en compte l'entropie des termes considérés au sein des documents pseudo-pertinents. De la même façon, nous avons étudié dans le chapitre précédent (et plus spécifiquement dans la section 4.3.4) les différences entre les concepts modélisés sur les sources d'information déjà utilisées précédemment. Nous continuons ici dans la même direction et considérons les concepts modélisés à partir des ressources détaillées en section 2.4, et les combinons afin d'arriver à une meilleur estimation des modèles de pertinence conceptuels. Nous reprenons une formulation très équivalente à celle introduite dans la section 3.4, ce qui donne une combinaison de la forme suivante :

$$P(w|\hat{\theta}_Q) \propto \sum_{\mathcal{R} \in \mathcal{S}} \varphi_{\mathcal{R}} \sum_{\theta_D \in \Theta} P(\theta_D) P_{TM}(w|D, \boldsymbol{\theta}_{\hat{M}}, \boldsymbol{\phi}_{\hat{K}}) \prod_{t \in Q} P(t|\theta_D) \tag{5.7}$$

Le paramètre $\varphi_{\mathcal{R}}$ contrôlant le poids de chaque ressource $\mathcal{R}$ appartenant à l'ensemble $\mathcal{S}$ est calculé selon l'équation (3.18). Comme nous pouvons le voir dans l'équation (5.7), nous ne considérons qu'une combinaison de ATDRMs dans cette thèse, mais cette approche générale de combinaison pourrait naturellement s'appliquer à n'importe quel TDRM. Pour rester constant avec la notation introduite par Diaz et Metzler (2006) (MoRM), nous définissons l'acronyme MoATDRM pour *Mixture of Adaptive Topic-Driven Relevance Models*.

5.3 Évaluation

5.3.1 Protocole expérimental

Nous présentons dans cette section une évaluation des performances des différentes approches que nous avons introduites. Comme nous l'avons déjà fait tout au long de cette thèse, nous explorons les effets des modèles que nous proposons sur quatre collections de test. Le protocole expérimental est identique à celui détaillé dans la section 3.5. Le paramètre λ contrôlant l'influence de la requête originale par rapport aux concepts implicites est estimé par validation croisée. Les modèles de pertinence conceptuels sont basés sur les modèles de pertinence, nous prenons donc logiquement l'implémentation RM3 comme système de base, pour lequel des détails sont proposés en section 3.2.2. Le nombre total de termes utilisés dans la méthode RM3 reste le même que dans notre première implémentation et nous le fixons à 20. Le nombre de mots utilisés dans chaque concept est limité à 10. Augmenter le nombre de concepts à modéliser permet d'augmenter indirectement le nombre de mots utilisés, nous explorons les effets engendrés dans cette section.

5.3.2 Recherche conceptuelle de documents

Nous examinons dans cette section les résultats de recherche documentaire pour l'approche générale TDRM introduite en section 5.2.2. Nous fixons ici les paramètres K et M, respectivement le nombre de concepts modélisés au sein des documents pseudo-pertinents et le nombre de documents utilisés pour cette modélisation. Nous reportons les performances dans les figures ci-dessous en terme de précision moyenne (MAP), avec $K \in \{3, 5, 10, 15, 20\}$ et $M \in \{5, 10, 20, 30, 40, 50\}$.

Les performances obtenues pour les requêtes de la collection WT10g sont détaillées dans la figure 5.1. Nous avions observé dans la section 4.3.3 du précédent chapitre que les modèles conceptuels les plus cohérents étaient modélisés en utilisant 5 documents pseudo-pertinents et 20 concepts pour cette collection. Nous observons ici que c'est également cette combinaison de paramètres qui obtient les meilleurs résultats. Jusqu'à 10 documents utilisés, le nombre de concepts a peu d'importance, sauf l'utilisation de 3 concepts sur 10 documents qui obtient de très mauvais résultats comparés aux autres. Toutes les combinaisons de paramètres obtiennent des résultats très supérieurs au système de base RM3, sauf lorsqu'on utilise uniquement 3 concepts et un grand nombre de documents pseudo-pertinents. La quantité de texte devient alors trop importante pour qu'un faible nombre concepts puissent modéliser avec précision le contexte thématique de la requête. Globalement, de meilleurs résultats sont obtenus en utilisant un plus grand nombre de concepts. Les résultats plus faibles sont obtenus avec un plus petit nombre de concepts, ce qui explique les performances relativement faibles de l'approche adaptative ATDRM. Nous avons en effet vu dans le chapitre précédent que notre méthode avait tendance à modéliser un petit nombre de concepts pour un petit nombre de documents pseudo-pertinents. Or, il semble que la pondération des mots mise en

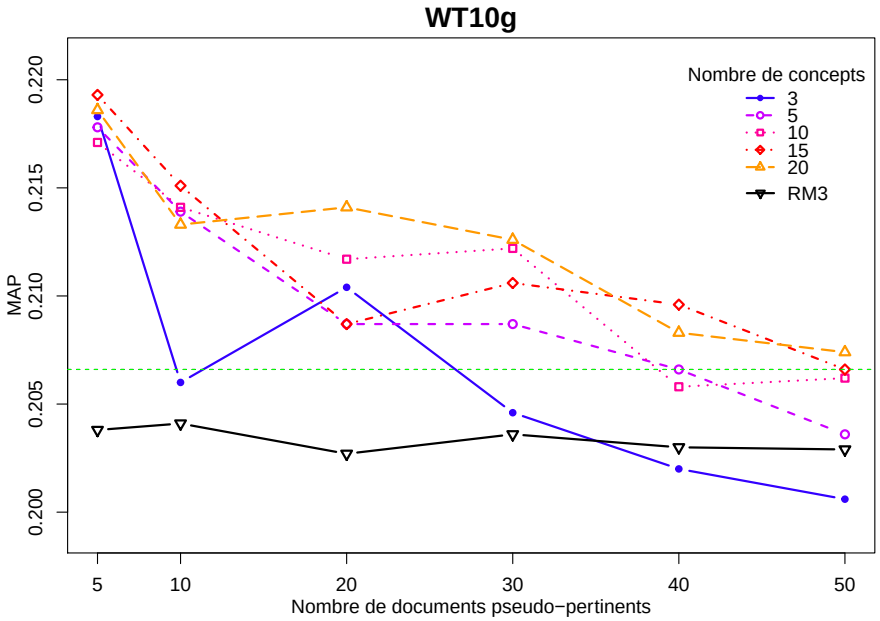

FIGURE 5.1 – *Performances de l'approche TDRM en terme de précision moyenne (MAP) pour la collection WT10g. Chaque ligne représente un différent nombre K de concepts, et les performances sont exprimées en fonction du nombre M de documents pseudo-pertinents. La ligne noire, solide, représente le système de base RM3. La ligne verte, pointillée, représente l'approche adaptative AT-DRM.*

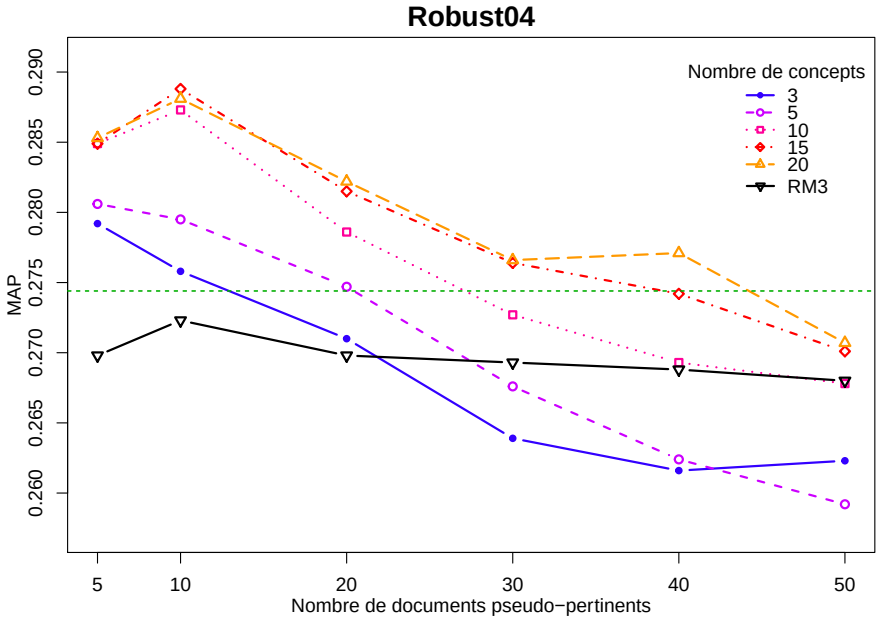

FIGURE 5.2 – *Performances de l'approche TDRM en terme de précision moyenne (MAP) pour la collection WT10g. La légende est identique à celle de la figure 5.1.*

place par nos modèles de pertinence conceptuels permet de discriminer efficacement les mots liés conceptuellement à la requête des autres.

Les résultats sont très similaires pour la collection Robust04, comme nous pouvons le voir sur la figure 5.2. En effet, les meilleurs résultats sont obtenus en modélisant un grand nombre de concepts, alors que les performances diminuent lorsqu'on augmente le nombre de documents pseudo-pertinents. De même que pour la collection WT10g, les modèles de pertinence conceptuels utilisant peu de concepts finissent par être moins performants que le système de base RM3 après 30 documents utilisés. Il est encore une fois très intéressant de noter qu'utiliser un grand nombre de concepts pour la modélisation semble « gommer » le bruit introduit par un grand nombre de documents potentiellement non-pertinents. L'approche ATDRM obtient ici aussi des résultats mitigés, même si RM3 n'obtient jamais de meilleurs résultats.

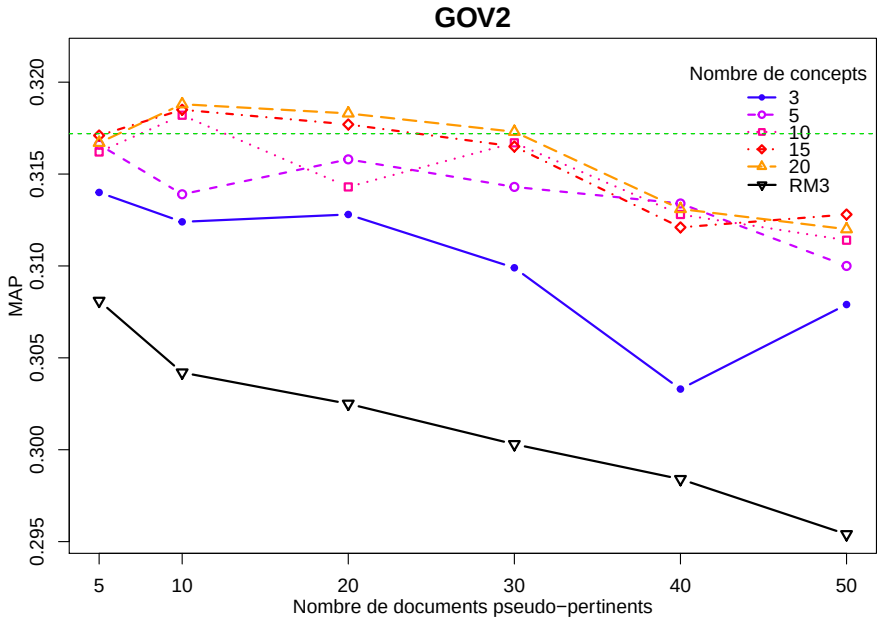

FIGURE 5.3 – *Performances de l'approche TDRM en terme de précision moyenne (MAP) pour la collection GOV2. La légende est identique à celle de la figure 5.1.*

La figure 5.3 représentant les résultats de l'approche TDRM sur la collection GOV2 affiche elle-aussi une forme très similaire aux précédentes, avec de très bonnes performances de ATDRM. Encore une fois, utiliser un plus grand nombre de concepts pour effectuer la modélisation thématique permet d'extraire un vocabulaire riche et varié, au sein d'un modèle donnant de l'importance aux mots thématiquement importants. Une autre remarque importante est la forte sensibilité de RM3 au nombre de documents pseudo-pertinents sur cette collection. Les documents classés dans les premiers rangs ont ainsi une plus forte tendance à être pertinents. Or, malgré cette tendance, nous voyons que la courbe représentant l'approche TDRM avec $K = 20$ décroit bien moins rapidement que RM3 entre 10 et 30 documents pseudo-pertinents, ce qui renforce notre hypothèse d'une pondération discriminante des mots au sein du modèle.

Pour finir, les résultats obtenus sur la très large et très spammée collection ClueWeb09-B sont complètement différents des précédents. Au lieu d'une décroissance, nous observons sur la figure 5.4 que les résultats augmentent au fur et à mesure que l'on augmente le nombre de documents pseudo-pertinents. Peu de documents pertinents sont présents dans les tout premiers rangs, ce qui mène à la formation de concepts peu pertinents lorsqu'on utilise un petit nombre de documents. Des concepts pertinents peuvent encore être trouvés après avoir parcouru plus d'une quarantaine de documents. Les résultats se stabilisent néanmoins entre 40 et 50 documents. Comme pour les collections précédentes, les meilleurs résultats sont encore une fois obtenus en utilisant un grand nombre de concepts. Une observation importante sur les collections Robust04, GOV2 et ClueWeb09-B est la très forte similarité entre les résultats obtenus pour $K = 15$ et $K = 20$. Ainsi, un ajout de cinq concepts (soit 50 mots) perturbe de façon presque imperceptible les modèles de pertinence. Cela démontre une fois de plus la robustesse de la pondération des modèles de pertinence conceptuels et de leur capacité à ne favoriser que les concepts liés à la requête. Nous explorons dans la section suivante l'impact du nombre de concepts sur les mots uniques intégrés aux modèles.

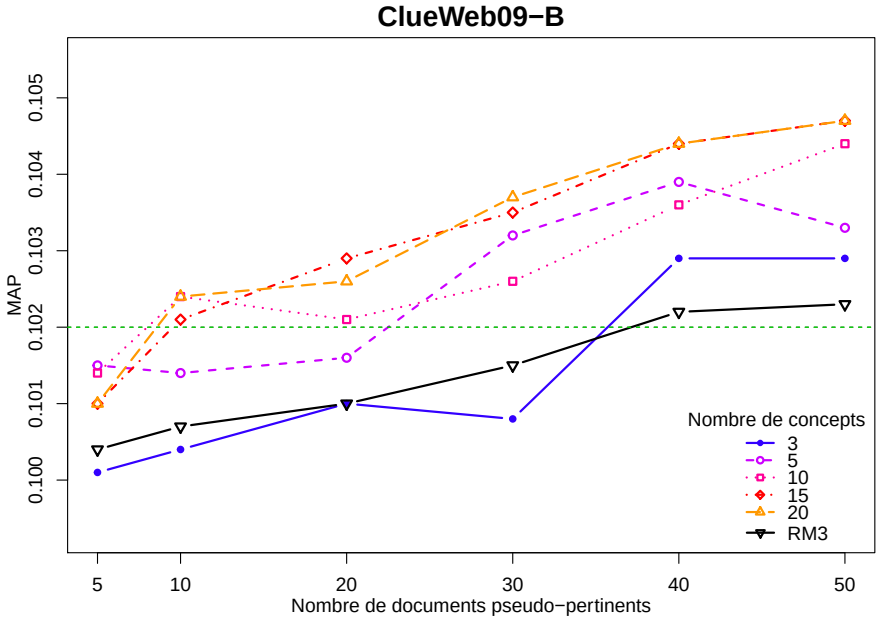

FIGURE 5.4 – *Performances de l'approche TDRM en terme de précision moyenne (MAP) pour la collection ClueWeb09-B. La légende est identique à celle de la figure 5.1.*

D'un autre côté, une piste d'amélioration de l'approche ATDRM serait l'utilisation des scores de cohérence sémantique afin de choisir le « bon » modèle conceptuel à utiliser pour l'estimation du modèle de pertinence conceptuel. Néanmoins nous avons vu dans le chapitre précédent que la cohérence évolue différemment selon les collections (i.e. selon les requêtes), et il est difficile de définir un modèle unique. Nous avons mené des premières expériences sur les collection WT10g et Robust04 où nous sélectionnons, pour chaque requête, le modèle conceptuel le plus cohérent parmi un ensemble de modèles appris en faisant varier les paramètres K et M. Ce modèle est alors utilisé directe-

ment dans le TDRM. Les premiers résultats sont prometteurs pour la collection WT10g et permettent au modèle ATDRM d'obtenir une MAP de 0,2112 (contre 0,2066 dans sa version actuelle), mais ils sont moins impressionnants pour la collection Robust04 avec une MAP de 0,2726 (contre 0,2744 dans sa version actuelle). Ces améliorations sortent néanmoins du cadre de cette thèse, et nous les laissons pour des travaux futurs.

5.3.3 Influence du nombre de mots composant les concepts

Nous venons de voir qu'utiliser un grand nombre de concepts permettait d'obtenir globalement les meilleurs résultats pour l'approche TDRM, peu importe le nombre de documents pseudo-pertinents utilisés. Seulement, nous fixons *a priori* le nombre de mots composant un concept à 10 : augmenter le nombre de concepts revient donc à augmenter le nombre de mots utilisés pour l'estimation du modèle de pertinence. Comme nous l'avions vu plus tôt dans cette thèse dans la section 3.5.4, le nombre de mots utilisés au sein d'un modèle de pertinence joue un rôle significatif jusqu'à environ 20 mots, mais les performances restent stables lorsqu'on en rajoute. Le risque d'ajouter des mots non pertinents ou non liés au contexte thématique de la requête est également grand, c'est pourquoi les approches traditionnelles se contentent généralement d'un faible nombre de mots (20 est un nombre qui revient souvent dans la littérature).

Dans notre approche conceptuelle des modèles de pertinence, nous voyons que le nombre de mots semble avoir un impact très positif sur les performances, au lieu d'un impact neutre ou négatif comme précédemment observé. Les documents pseudo-pertinents étant par nature liés à la requête et abordant tous des thématiques similaires, le vocabulaire est redondant et certains mots peuvent se retrouver dans plusieurs concepts. Nous comptons ici pour chaque requête le nombre de mots uniques présents dans les concepts identifiés pour celle-ci, puis nous reportons la moyenne des nombres de mots uniques dans la figure 5.5.

Nous observons que, pour les quatre collections de test, les tendances sont très similaires et conformes à notre hypothèse. En effet, plus on utilise de concepts, plus le vocabulaire utilisé pour estimer les modèles de pertinence conceptuels est varié. Il est intéressant de noter qu'il semble exister une limite pour 3 ou 5 concepts utilisés. Les modèles conceptuels composés de 3 concepts comprennent en moyenne entre 20 et 22 mots uniques, et ce pour toutes les collections et n'importe quel nombre de documents pseudo-pertinents. De la même façon, 5 concepts permettent de récolter en moyenne entre 30 et 35 mots uniques. Le nombre de documents pseudo-pertinents a un très faible impact pour ces modèles composés d'un petit nombre de concepts, mais on peut observer une baisse légère du nombre de mots uniques au fur et à mesure qu'on augmente le nombre de documents. Cette tendance est également observable sur la collection Robust04 avec 10 concepts : le nombre de mots uniques augmente jusqu'à 20 documents puis chute.

On peut ainsi voir les concepts comme des « emplacements » dans lesquels des mots conceptuellement liés à la requête doivent être placés : augmenter le nombre de documents permet d'augmenter la variabilité du vocabulaire au risque d'intégrer des mots

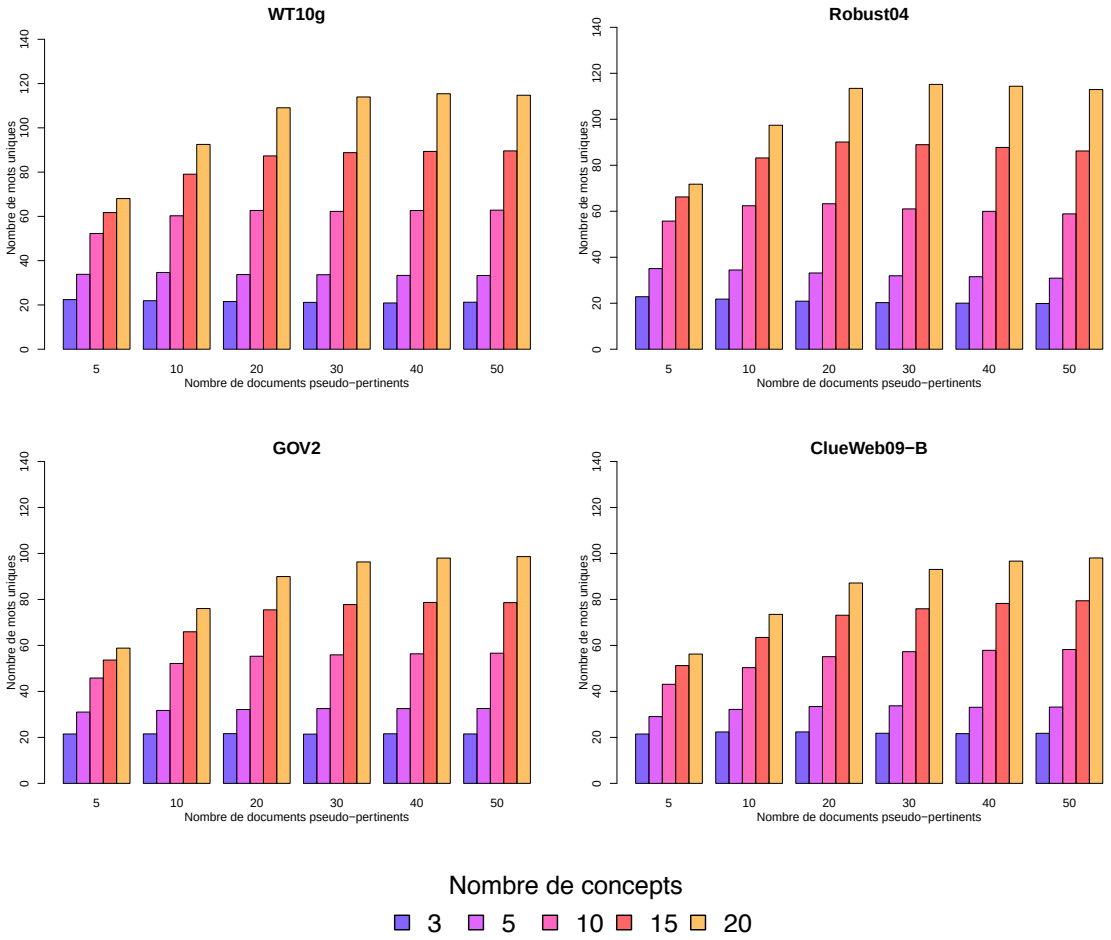

FIGURE 5.5 – *Moyennes des nombre de mots uniques utilisés dans les concepts modélisés pour les quatre collections. Les échelles de valeurs sont identiques pour les quatre collections.*

peu importants. Lorsque ces emplacements sont limités (peu de concepts), l'algorithme de modélisation thématique va favoriser la redondance des mots au détriment de leur diversité.

Nous avons vu dans la section précédente que cette diversité pouvait être très bénéfique, principalement grâce à une pondération efficace qui permet de limiter l'influence de mots marginaux sans les écarter totalement du modèle. C'est ce comportement que nous pouvons encore une fois observer sur la figure 5.5, avec une augmentation constante du nombre de mots uniques en fonction du nombre de documents pour des modèles de 15 et 20 concepts. Une petite baisse peut malgré tout être observée sur la collection Robust04. C'est également sur cette collection que les modèles de 20 concepts comprennent le plus grand nombre de mots uniques. Cette collection étant composée d'articles journalistiques rédigés par des journalistes professionnels, évitant la redondance et favorisant les synonymes, les concepts sont plus divers que dans les autres collections (même si les résultats sont très similaires pour la collection WT10g). Une grande différence peut être observée avec les collections GOV2 et ClueWeb09-B

qui, au maximum, utilisent en moyenne 20 mots uniques de moins que les collections Robust04 et WT10g.

Contrairement aux observations faites en section 3.5.4 où un grand nombre de mots avait une très faible influence sur les performances, nous voyons ici qu'un grand nombre de concepts variés permet toujours d'obtenir de meilleurs résultats. La pondération de nos modèles de pertinence conceptuels s'en trouve validée, puisqu'elle permet de correctement discriminer les mots conceptuellement liés à la requête d'autres mots potentiellement annexes. Nous continuons notre exploration des performances des modèles conceptuels dans la section suivante, qui présente les performances atteintes par l'approche MoATDRM.

5.3.4 Résultats de combinaison de modèles

Dans cette section, analogue à la section 3.5.2, nous présentons les performances du modèle combinant différents modèles de pertinence conceptuels estimés à partir de différentes sources d'informations. Comme nous l'avons précisé en section 5.2.4, il s'agit d'une combinaison de modèles adaptatifs : évaluer toutes les combinaisons en variant les paramètres K et M aurait mené à une explosion du nombre de *runs*. Tirer des conclusions significatives et sérieuses aurait constitué un réel défi. Nous nous basons donc sur les résultats obtenus en section 5.3.2 pour les TDRM et les ATDRM utilisant la collection cible comme source de documents pseudo-pertinents, et nous généralisons nos observations. De plus, nous observons dans cette section les résultats des différents ATDRM-$\mathcal{R}$ où $\mathcal{R}$ peut également être la collection cible (ou alors $\mathcal{R} \in \{Wiki, NYT, Gigaword, Web\}$).

	QL		RM3		MoRM		MoATDRM	
	MAP	P@20	MAP	P@20	MAP	P@20	MAP	P@20
WT10g	0,2026	0,2429	0,2035	0,2449	$0,2339^{\alpha,\beta}$	$0,2833^{\alpha,\beta}$	$0,2499^{\alpha,\beta,\gamma}$	$0,2874^{\alpha,\beta}$
Robust04	0,2461	0,3528	$0,2727^{\alpha}$	0,3677	$0,2869^{\alpha,\beta}$	$0,3799^{\alpha,\beta}$	$0,3124^{\alpha,\beta,\gamma}$	$0,4086^{\alpha,\beta,\gamma}$
GOV2	0,2911	0,5145	0,2877	0,5074	$0,3083^{\alpha,\beta}$	$0,5409^{\alpha,\beta}$	$0,3262^{\alpha,\beta,\gamma}$	$0,5765^{\alpha,\beta,\gamma}$
ClueWeb09-B	0,1007	0,2347	0,1007	0,2260	0,1045	0,2250	$0,1175^{\alpha,\beta,\gamma}$	$0,2806^{\alpha,\beta,\gamma}$

TABLE 5.1 – *Résultats de recherche documentaire reportés en terme de précision moyenne (MAP) et de précision à 20 documents pour les approches QL, RM3, MoRM et MoATDRM. Nous utilisons le test apparié de Student (t-test) pour déterminer les différences significatives avec les systèmes de base. α, β et γ indiquent respectivement des améliorations significatives par rapport à QL, RM3 et MoRM, avec $p < 0,05$.*

Les résultats présentés dans le tableau 5.1 représentent les performances de l'approche MoATDRM par rapport à la vraisemblance de la requête (QL, *query likelihood*), aux modèles de pertinence (RM3) et à la combinaison de modèles de pertinence de Diaz et Metzler (2006). Des améliorations significatives par rapport à MoRM peuvent être observées pour toutes les collections et les deux métriques, à l'exception de la précision à 20 documents pour la collection WT10g. Ces résultats sont très satisfaisants, surtout sachant que les modèles combinés sont des ATDRM, pour lesquels nous avons vu dans les sections précédentes que les performances pouvaient être grandement améliorées.

Celles-ci sont du même ordre que la méthode DfRes, même si étant légèrement supérieures pour les collections WT10g, GOV2 et ClueWeb09-B.

	nyt	wiki	gigaword	web	WT10g	Robust04	GOV2	ClueWeb09-B
WT10g	**0,343**	0,090	0,160	0,313	0,089	-	-	-
Robust04	0,273	0,100	**0,309**	0,116	-	0,201	-	-
GOV2	0,247	0,188	0,187	0,093	-	-	**0,280**	-
ClueWeb09-B	0,142	0,173	0,202	**0,369**	-	-	-	0,113

TABLE 5.2 – *Moyennes des poids $\varphi_{\mathcal{R}}$ appris pour les quatre collections. Les chiffres en gras correspondent aux plus forts poids par collection. Ce tableau est analogue à celui présenté dans la section 3.5.*

Pour cette combinaison aussi, l'apprentissage des poids $\varphi_{\mathcal{R}}$ joue un rôle essentiel dans les bonnes performances obtenues, et nous donne aussi une vision de l'utilité de chaque source d'information. Nous reportons ainsi les poids moyens appris pour les différentes collections et sources d'information utilisées dans le tableau 5.2.

Nous pouvons noter que peu de tendances globales se dégagent de ce tableau : l'importance des sources d'information varie différemment suivant la collection et le contexte de recherche (recherche web ou recherche d'articles, large collection ouverte ou collection de taille réduite). Nous remarquons néanmoins encore une fois que Wikipédia semble ne pas être une source de choix pour l'extraction de concepts : les informations conceptuelles sont concentrées dans un très petit nombre de documents, où il n'y a pas de redondance. Hors, nous avons vu précédemment que la redondance d'information pouvait être très bénéfique dans le cas des modèles de pertinence conceptuels. Une dizaine de documents pseudo-pertinents issus d'une source de pages Web vont typiquement tous traiter des mêmes informations, des mêmes concepts, mais à des niveaux différents et en variant le vocabulaire utilisé. Dans le cas de Wikipédia, la structure encyclopédique de cette ressource fait qu'un article correspond à un concept précis ou à une entité, et une dizaine d'articles traiteront d'autant de concepts qui peuvent être très éloignés de la requête qui a servi à les récupérer. Ainsi, la source Web est importante pour les collections WT10g et ClueWeb09-B, tandis que le NYT a des poids très importants dans presque tous les cas. Les sources journalistiques semblent globalement être très utiles (voir les poids de la source Gigaword) malgré le fait qu'elles soient ciblées sur des évènements ou des sujets précis.

C'est ce que nous voyons précisément sur la figure 5.6 qui représente les différents ATDRM-$\mathcal{R}$ ainsi que l'approche MoATDRM pour la collection WT10g. Ayant un poids $\varphi_{\mathcal{R}}$ très similaire à celui de la source Web, la source NYT obtient d'excellents résultats lorsqu'elle est utilisée seule. Ces résultats sont même équivalents à ceux de l'approche MoATDRM pour $\lambda \geq 0,4$. Un résultat très surprenant est la grosse amélioration des résultats de ATDRM-Gigaword lorsque l'on fait augmenter la valeur de λ. Pour $\lambda = 0,6$, c'est même la deuxième approche utilisant une seule source d'information donnant de meilleurs résultats. Or les résultats sont particulièrement mauvais lorsque $\lambda = 0$. Seuls, les concepts modélisés à partir du corpus Gigaword ne sont pas suffisamment précis et contiennent un vocabulaire trop spécifique n'étant pas présent dans les documents

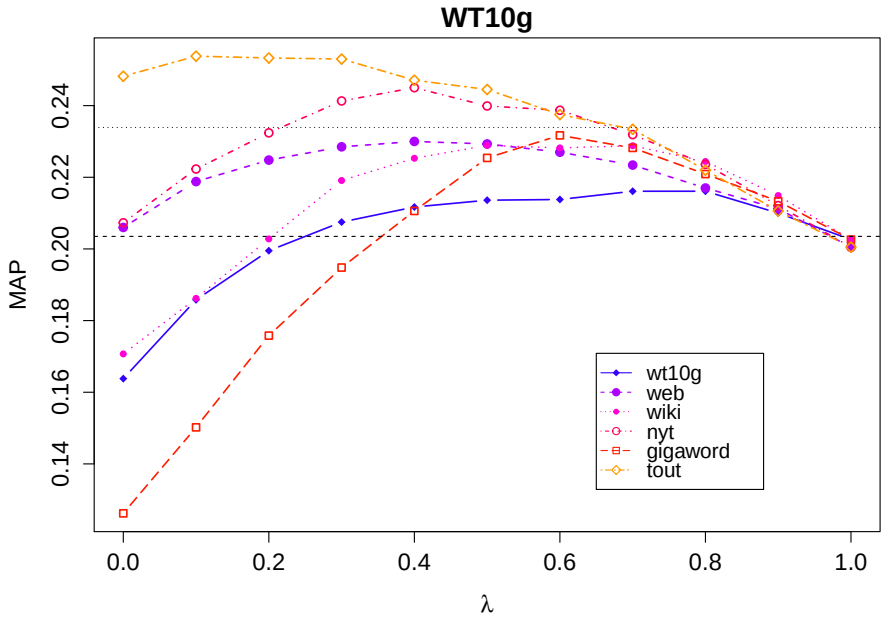

FIGURE 5.6 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection WT10g. La méthode MoATDRM est représentée par la courbe « tout », tandis que les autres courbes correspondent à des ATDRMs utilisant une seule source d'information à la fois. Les systèmes de bases sont reportés pour référence : les tirets représentent RM3 et la ligne pointillée représente MoRM.*

jugés pertinents mais, combinés à la requête originale, ils permettent de récupérer un plus grand nombre de documents pertinents que d'autres sources de tailles plus importantes. Nous pouvons également voir que l'approche MoATDRM avec $\lambda = 0$ obtient de meilleurs résultats que toutes les autres approches (systèmes de base compris), et obtient même presque les meilleurs résultats globaux. En effet, seule cette même approche avec $0,1 \leq \lambda \leq 0,3$ obtient des résultats plus élevés, mais sans différence statistiquement significative. Cette combinaison de concepts issus de différentes sources d'information est donc de très bonne qualité, puisqu'elle permet d'obtenir une excellente estimation du contexte thématique de la requête tout en se passant de celle-ci. De plus, il est très intéressant de voir que cette combinaison est basée sur des concepts qui, seuls, ne donnent pas de très bons résultats. Pour $\lambda = 0$, toutes les approches ATDRM-$\mathcal{R}$ obtiennent de mauvais résultats ou arrivent péniblement à dépasser RM3.

Nous pouvons également faire ces mêmes observations pour la collection Robust04, dont les résultats sont reportés dans la figure 5.7. Ici encore, les approches ATDRM-$\mathcal{R}$ sont très largement en dessous des systèmes de base pour $\lambda = 0$, et pourtant la combinaison MoATDRM obtient de très bons résultats pour cette même valeur de λ. Elle est uniquement dépassée pour $0,1 \leq \lambda \leq 0,2$. On voit pour cette collection que la nature des sources d'information utilisées joue un rôle très important, comme en attestent les très bonnes performances des ressources NYT et Gigaword [2]. Les sources Wikipédia et Web obtiennent de mauvais résultats pour toutes les valeurs de λ ce qui ex-

2. Par ailleurs, ces observations sont similaires à celles faites dans la section 3.5.3.

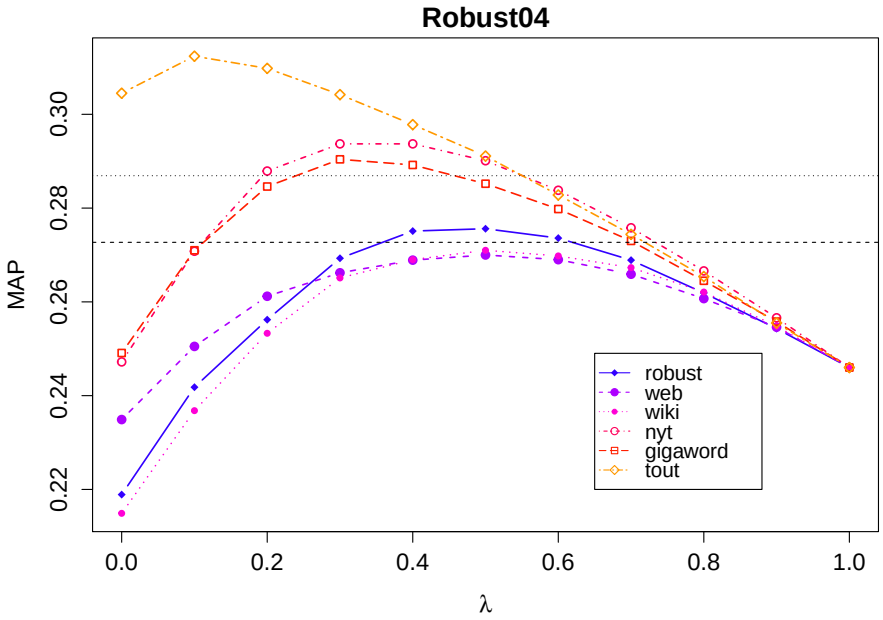

FIGURE 5.7 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection Robust04. La légende est identique à celle de la figure 5.6.*

plique leurs poids relativement bas pour cette collection. Néanmoins malgré ces contre-performances, elles apportent leur contribution à la combinaison. Pour 25 requêtes sur 250, ATDRM-Wiki obtient de meilleurs résultats que les autres ATDRM, contre 29 requêtes pour ATDRM-Web.

Tout comme nous l'avions mentionné dans le premier chapitre, les collections WT10g et Robust04 semblent très adaptées à des approches effectuant de l'enrichissement, peu importe la méthode utilisée. En dehors des fortes améliorations de résultats, nous avons montré à plusieurs reprises que nous étions capables de modéliser le contexte thématique de la requête de manière très efficace. Cette modélisation est suffisamment précise pour nous permettre de nous passer de la requête originale. Néanmoins, nous ne pouvons pas faire les mêmes observations pour les deux plus larges collections : GOV2 et ClueWeb09-B. Nous reportons les résultats dans les figures 5.8 et 5.9 respectivement. La première observation qui ressort en effet de ces figures, comparées aux deux précédentes, est que l'approche MoATDRM n'obtient plus de meilleurs résultats que les systèmes de base pour $\lambda = 0$. Un trop grand nombre de documents compose ces collections, et la requête originale reste indispensable pour pouvoir récupérer au moins une fraction des documents jugés pertinents[3]. Par ailleurs, la forme parabolique des résultats de l'approche MoATDRM pour nos deux dernières collections est centrée sur des valeurs de λ proche de 0,5, ce qui atteste de l'importance quasi-équivalente de la requête originale et des modèles de pertinence.

Les résultats de MoATDRM restent malgré tout très bons et dans presque tous les

3. Voir la figure 3.5 dans le chapitre 3 pour plus de détails sur le cas des documents potentiellement pertinents mais non jugés dans les très larges collections.

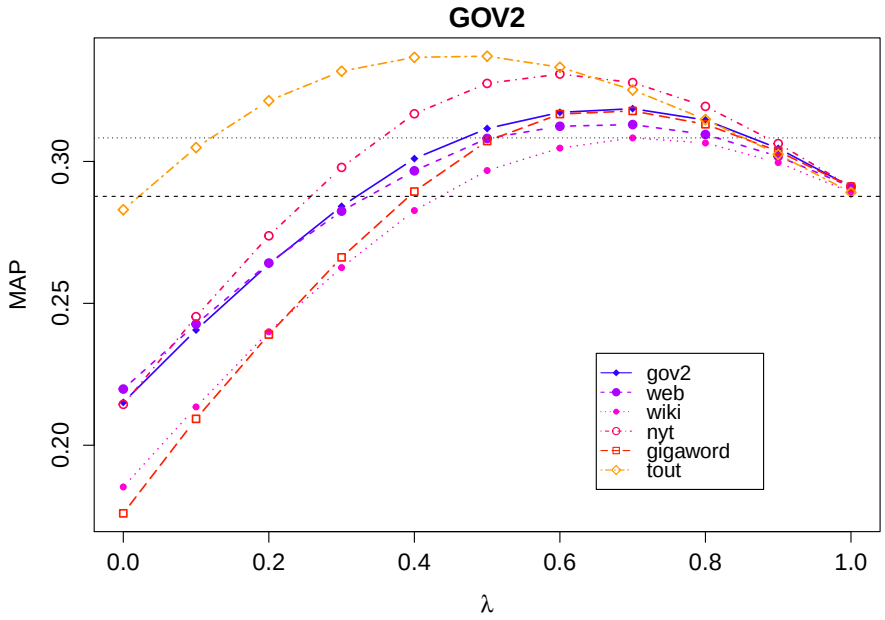

FIGURE 5.8 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection GOV2. La légende est identique à celle de la figure 5.6.*

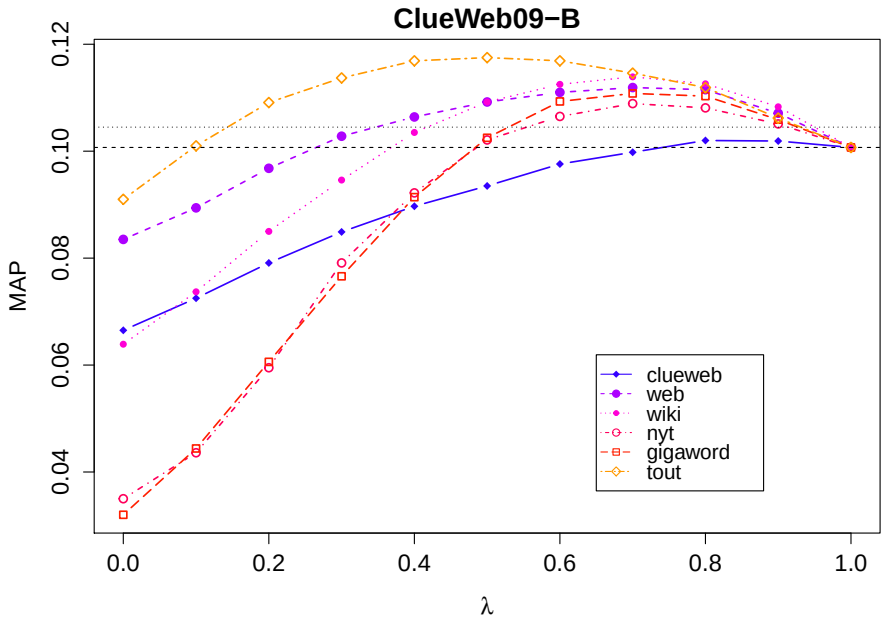

FIGURE 5.9 – *Performances (exprimées en MAP) en fonction du paramètre λ sur la collection Clueweb09-B. La légende est identique à celle de la figure 5.6.*

cas supérieurs aux autres approches basées sur les ATDRM. Il est d'ailleurs important de noter que l'utilisation de modèles de pertinence conceptuels a permis d'augmenter les résultats de la combinaison par rapport à l'approche DfRes, même si ces améliorations ne sont pas significatives lorsque l'on fixe le paramètre λ par validation croisée.

Globalement, les résultats obtenus pour l'approche ATDRM sont moins bons quand on utilise la collection cible plutôt qu'une source d'information externe. Le seul cas particulier concerne la collection GOV2, pour laquelle nous avions déjà remarqué dans la section précédente que l'approche ATDRM obtenait de bons résultats par rapport aux différentes combinaisons de paramètres fonctionnant bien pour les autres collections. L'apport de sources d'information externes semble donc bien réel malgré les problèmes de différences de vocabulaire entre deux collections de documents.

5.4 Conclusions et perspectives

Nous avons présenté dans ce chapitre une nouvelle approche générale permettant d'intégrer des concepts identifiés à l'aide d'un algorithme de modélisation thématique au sein de modèles de pertinence. Tout comme les modèles de pertinence traditionnels, les modèles de pertinence conceptuels estimés à partir de différentes sources d'information peuvent être combinés afin d'obtenir de meilleurs résultats de recherche documentaire. Nous avons également intégré l'approche présentée dans le chapitre précédent afin de sélectionner automatiquement, pour chaque requête, un ensemble de concepts à utiliser au sein des modèles de pertinence. Bien que les résultats n'aient pas été aussi bons que les meilleures combinaisons de paramètres (nombre de concepts et nombre de documents pseudo-pertinents utilisés pour les identifier) pour la majorité des collections de test, cette approche adaptative a constamment donné de meilleurs résultats que les modèles de pertinence traditionnels.

De plus, des expériences préliminaires sembleraient nous indiquer que prendre en compte la cohérence sémantique des modèles conceptuels pourrait être une voie intéressante à explorer. En effet, nous avons pu observer que la cohérence sémantique était un bon indicateur de performance dans le cadre d'une tâche de recherche Web, mais pas dans le cadre d'une recherche d'articles journalistiques. Nous avons vu par ailleurs que les meilleurs résultats étaient globalement obtenus en modélisant un grand nombre de concepts sur un nombre réduit de documents pseudo-pertinents. Nous pensons donc définir un modèle pouvant combiner l'approche présentée dans le chapitre 4 avec différentes mesures de cohérence sémantique telles que définies par Stevens et al. (2012) ou Newman et al. (2010).

Une extension évidente de ce travail serait également l'utilisation d'autres algorithmes de modélisation thématique probabilistes tels que pLSA (Hofmann, 2001). En effet, LDA et pLSA sont bâtis sur les mêmes principes, estimant que la distribution des mots sur les concepts est régie par une loi de probabilités multinomiale, tout comme la distribution des concepts sur les documents. Il pourrait ainsi être intéressant de comparer les concepts modélisés à partir de documents pseudo-pertinents en utilisant ces deux algorithmes, voir leur influence sur la cohérence sémantique des concepts et leur effet sur les performances de recherche documentaire. De même, utiliser d'autres méthodes d'estimation de LDA, comme l'échantillonnage de Gibbs (Griffiths et Steyvers, 2004), pourrait être une piste d'exploration complémentaire afin de valider notre proposition. L'utilisation de processus de Dirichlet hiérarchiques (Teh et al., 2006), équi-

valents à un LDA pour lesquels les poids des concepts sont déterminés automatiquement, pourrait également être une perspective de travail. Nous risquerions néanmoins de nous heurter à des problèmes de fixation adaptative du seuil contrôlant le poids minimum associé aux concepts. Nous avions également observé dans la section 4.3.2 que cette méthode apprenait en général un faible nombre de concepts sur les documents pseudo-pertinents (entre 4 et 6), alors que les meilleurs résultats de recherche documentaire sont obtenus avec 15 ou 20 concepts. Ces problématiques sortant néanmoins du cadre de cette thèse, nous proposons de les aborder dans de futurs travaux.

Chapitre 6

Conclusion

Sommaire

Est-il possible de représenter de façon entièrement automatique les thématiques liées au besoin d'information d'un utilisateur, exprimé uniquement par une requête ? C'est la question principale qui a motivé le développement des méthodes que nous avons présentées dans cette thèse. Nous avons proposé plusieurs approches performantes permettant d'extraire des mots liés thématiquement à la requête et de les pondérer de façon à refléter l'importance de l'information qu'ils transmettent. Un fil conducteur que l'on aura retrouvé tout au long de cette thèse aura été l'utilisation de sources d'information externes dans nos approches, ainsi que leur combinaison.

Alors que, intuitivement, on pourrait penser que l'utilisation de collections externes est défavorable aux méthodes faisant de l'enrichissement ou de la réécriture de requête (principalement à cause des différences de vocabulaire avec la collection cible), nous avons observé que les différentes sources d'information que nous avons utilisées étaient globalement plus performantes que les collections cibles. Ces bonnes performances tiennent tout d'abord de la qualité des sources que nous avons utilisées. Les corpus NYT (Sandhaus, 2008) et Gigaword (Graff et Cieri, 2003) distribués par le Linguistic Data Consortium [1] sont des sources très bien formées, sans déchets, où les documents ont été rédigés par des journalistes professionnels. Bien que l'usage de l'encyclopédie Wikipédia ait obtenu des résultats mitigés, c'est une vaste source de connaissance enrichie, modifiée et modérée collectivement par un ensemble de contributeurs. Pour finir, la ressource Web que nous avons utilisée a été nettoyée de ses documents *spammés*, et nous avons pu voir qu'elle était très efficace pour plusieurs de nos approches. Au delà de toutes ces qualités, la diversité de ces sources d'information aura joué un rôle primordial afin de représenter le contexte thématique de la requête le plus précisément possible. En projetant la requête dans cinq espaces thématiques différents (quatre

1. http://www.ldc.upenn.edu/

sources externes et la collection cible) et de tailles relativement importantes, nos approches ont pu tirer parti de toutes les informations à leur disposition, qu'elles soient aussi bien uniques que redondantes. Nous avons également touché les limites des très larges collections de test, qui ont tendance à être difficilement réutilisables dans les cas extrêmes où l'on ignore entièrement la requête originale. Une évaluation manuelle complémentaire, assurément coûteuse, pourrait être une solution à ce problème. La suite de cette conclusion reprend les résultats principaux que nous avons mis en valeur dans cette thèse, puis offre des perspectives d'évolution et de travaux futurs.

6.1 Résultats

Dans le chapitre 3, nous avons proposé une variante des modèles de pertinence traditionnels utilisant l'entropie de termes au sein de l'ensemble de documents pseudo-pertinents. Le score final d'un document revenait au final à un calcul de divergence entre son modèle de langue et le modèle de langue des documents pseudo-pertinents. Le deuxième aspect de ce chapitre était l'utilisation de plusieurs sources d'information pour calculer cette divergence. Ainsi, un document était correctement classé s'il ne divergeait d'aucune des sources de documents pseudo-pertinents. Ce processus de contextualisation de la requête, visant à estimer automatiquement au plus près les informations thématiques liées au besoin d'information ayant mené l'utilisateur à formuler cette même requête, était donc réalisé par l'extraction de mots représentatifs de ce contexte et pondérés à l'aide du modèle de pertinence. Les résultats de recherche documentaire nous ont montré que cette approche était très efficace pour tous les scénarios de recherche que nous avons simulés. Dans l'ensemble, l'utilisation d'une source d'information seule améliore peu ou pas les performances, tandis que la combinaison dépasse toutes les autres méthodes issues de l'état-de-l'art. L'observation principale que nous retiendrons de ce chapitre est la très bonne qualité du contexte estimé, au moins pour les deux collections de tailles réduites.

Nous nous sommes attaqués dans le chapitre 4 au problème d'une modélisation des concepts implicites de la requête. Nous avons proposé une méthode entièrement automatique et non-supervisée tirant parti de l'allocation latente de Dirichlet (LDA), un algorithme de modélisation thématique. La LDA est célèbre pour ses modélisations précises sur de larges collections, mais nous l'avons ici appliquée à des ensembles réduits de documents pseudo-pertinents afin qu'elle modélise uniquement les concepts liés à la requête et non des concepts généraux. Nous estimons automatiquement le nombre de concepts et le nombre de documents pseudo-pertinents à utiliser afin de modéliser les concepts les plus informatifs et les moins bruités. Nous n'avons pas mené d'évaluation des performances de recherche documentaire dans ce chapitre, mais nous avons étudié la qualité des concepts ainsi générés. Des expériences menées dans ce chapitre, il ressort que les concepts que nous générons sont dans l'ensemble très cohérents sémantiquement comparés à des concepts modélisés sur des collections entières. Cette observation est très intuitive : une requête agissant comme un « ciblage » de l'information, il est logique que les concepts informatifs qui lui sont directement liés soient eux aussi très

ciblés et donc très cohérents. Nous sommes néanmoins à notre connaissance les premiers à confirmer cette intuition par des expériences. En appliquant cette approche à nos différentes sources d'information, nous avons pu mettre en évidence les différences de concentrations conceptuelles au sein des documents pseudo-pertinents. Finalement, nous avons montré que notre méthode d'estimation du nombre de concepts implicites d'une requête était corrélée, pour une grande majorité des requêtes, avec une méthode de modélisation thématique hiérarchique.

Nous avons finalement clôt la présentation des contributions de cette thèse avec le chapitre 5 qui présentait les modèles de pertinence conceptuels, une évolution thématique des modèles de pertinence traditionnels. Au lieu de réaliser l'estimation des modèles de pertinence à l'aide des simples mots contenus dans les documents pseudo-pertinents, nous avons tiré profit des idées déployées dans le chapitre 4 et avons utilisé les concepts implicites de la requête. Nous avons plus spécifiquement défini trois types de modèles de pertinence conceptuels : un modèle classique, un modèle adaptatif reprenant les méthodes d'estimation du nombre de concepts et du nombre de documents pseudo-pertinents (chapitre 4) et un modèle mixte combinant les modèles estimés sur différentes sources d'information. Nous avons montré que les meilleurs résultats de recherche documentaire étaient obtenus en utilisant un grand nombre de concepts et un nombre modéré de documents pseudo-pertinents (généralement entre 5 et 10). Ce faible nombre de documents permet de limiter les chances de considérer des documents non pertinents, tandis qu'un grand nombre de concepts permet de capturer un très grand nombre de mots. Nous avons ainsi pu conclure de ces observations que nos modèles de pertinence conceptuels étaient robustes et pondéraient correctement les différents mots afin de valoriser ceux qui sont importants et de déprécier ceux qui ne sont pas liés au contexte thématique de la requête. Dans ce chapitre-ci aussi, la combinaison de plusieurs modèles estimés à partir de sources d'information différentes a donné les meilleurs résultats, améliorant encore la qualité du contexte estimé par rapport à la méthode présentée dans le chapitre 3.

6.2 Perspectives

Nous avons rencontré plusieurs problèmes tout au long de cette thèse et certains sont restés non résolus. Plus précisément, nous avons vu que les modèles que nous avons introduits peuvent parfois manquer de robustesse et peuvent dégrader les performances des requêtes au lieu de les améliorer. Un des avantages de nos méthodes est qu'elles sont entièrement automatiques et non-supervisées, et ne requièrent donc aucune phase d'entraînement préalable. Nous pensons néanmoins qu'implémenter une stratégie de repli permettant d'estimer le risque associé à l'enrichissement d'une requête pourrait être grandement bénéfique à nos modèles. Nous avons à notre disposition un grand nombre de requêtes et les améliorations apportées par rapport aux systèmes de base pour chacune d'elles. Nous pourrions alors apprendre un classifieur supervisé pouvant décider de l'approche à choisir. Ce type de technique a déjà été étudié avec succès au niveau des mots, avec un classifieur décidant si un mot pouvait être uti-

lisé pour reformuler la requête ou non (Cao et al., 2008). Nous pourrions par exemple appliquer cette approche directement au niveau des concepts, ou encore modifier la pondération des mots en fonction de la décision et de la confiance du classifieur.

Nous avons ensuite vu que la méthode présentée dans le chapitre 4 visant à sélectionner automatiquement le modèle conceptuel représentant au mieux les concepts implicites de la requête ne permettait pas forcément d'obtenir les meilleurs résultats de recherche documentaire. Néanmoins de premières expériences exploratoires nous encouragent à penser qu'utiliser la cohérence sémantique de ces modèles conceptuels pourrait être un critère important dans le choix d'un modèle efficace. En effet, la cohérence sémantique des concepts est une caractéristique majeure définissant leurs performances dans le cadre d'une recherche de pages Web, même si cela est beaucoup moins évident dans le cadre d'une recherche d'articles journalistiques. Nous prévoyons néanmoins d'étendre l'utilisation des ces mesures de cohérence sémantique (Newman et al., 2010) afin d'identifier des traits communs aux concepts « utiles » pour la Recherche d'Information. Ici aussi, il pourrait être envisageable d'appliquer une couche d'apprentissage supervisé pouvant discriminer les concepts efficaces des autres.

Ces mesures de cohérence sémantiques pourraient finalement être adaptées pour prédire directement la difficulté des requêtes. De nombreuses études ont été menées afin de pouvoir estimer *a priori* les performances d'un système de Recherche d'Information par rapport à une requête, le mémoire de thèse de Hauff (2010) est un bon point d'entrée pour appréhender ce domaine. Il pourrait alors être envisageable de modéliser les différents concepts implicites d'une requête et de mesurer leur cohérence afin d'avoir des indications sur la nature de la requête. Un ensemble de concepts implicites très cohérents pourrait ainsi signifier que la requête fait référence à des informations bien définies qui seront vraisemblablement aisées à trouver. D'un autre côté, des concepts peu cohérents pourraient renvoyer à une requête mal définie, hautement ambigüe ou (très) mal orthographiée.

Annexes

Annexe A

Contextualisation automatique de Tweets à partir de Wikipédia

A.1 Introduction

La grande démocratisation de l'accès à internet et l'avènement des smartphones ont changé le paysage virtuel et la nature des échanges entre les personnes. L'information n'attend plus forcément d'être trouvée par quelqu'un ayant un besoin précis, elle vient directement à nous. Au centre de ce phénomène, les réseaux sociaux sont un média privilégié pour la diffusion de contenu à grande échelle (Bakshy et al., 2012). Les utilisateurs sont reliés par des connections de natures diverses (professionnelles, personnelles, publicitaires...) et s'échangent des informations en temps réel sur le monde qui les entoure. Twitter fait partie de ces réseaux sociaux et favorise des échanges de messages très courts. Quand il se connecte à Twitter, l'utilisateur doit répondre à la question « Quoi de neuf ? ». La réponse à cette question doit faire moins de 140 caractères et est appelée un *Tweet*. De par sa taille, un Tweet est naturellement ambigu et souvent sous-spécifié, ce qui peut rendre la compréhension compliquée pour une personne ne possédant pas le contexte approprié. Ce contexte peut être formé de phrases récupérées sur le Web (ou toute autre source) et réunies afin d'éclairer les lecteurs d'un Tweet sur sa nature et sur les concepts informatifs mis en jeu.

Nous plaçons notre étude dans le cadre d'un scénario mobile où un utilisateur va lire des Tweets (ou autres messages courts) sur son smartphone. Le contexte d'un Tweet doit donc être court afin de pouvoir être affiché de façon pratique sur un écran de téléphone. La tâche *Tweet Contextualization* d'INEX [1] propose un cadre expérimental permettant d'évaluer la contextualisation de Tweets réalisée à l'aide de phrases issues de Wikipédia. La collection de test est composée d'un ensemble statique d'articles Wikipédia, de Tweets et de phrases contextuelles de référence sélectionnées par les organisateurs.

1. https://inex.mmci.uni-saarland.de/

Notre approche de la contextualisation met en jeu successivement des techniques de Recherche d'Information (RI) et de résumé automatique. Tout d'abord, nous cherchons à améliorer la compréhension du Tweet en récupérant des articles Wikipédia liés à celui-ci. Ces derniers sont susceptibles de contenir des passages informatifs pour la construction du contexte du Tweet. Ensuite, nous considérons la formation du contexte comme une tâche de résumé automatique multi-documents, où il s'agit de résumer les articles Wikipédia retournés. Nous présentons dans cette annexe le modèle de RI puis l'approche de résumé automatique qui constituent notre système de contextualisation, puis nous évaluons notre approche en utilisant l'ensemble de données issu de la tâche *Tweet Contextualization* d'INEX 2012 (SanJuan et al., 2012).

A.2 Travaux précédents

Le problème de contextualisation de messages courts est émergent et se situe aux confluents de la Recherche d'Information ciblée et du résumé automatique. La tâche *Tweet Contextualization* de la campagne d'évaluation INEX 2012 est la première à proposer un cadre d'évaluation formel pour ce type de problématique et a été suivie par de nombreux participants. Différents travaux ont également considéré les Tweets comme sources d'informations récentes (Sankaranarayanan et al., 2009), et ont tenté des approches de recommandation (Chen et al., 2010). Allant dans le même sens, une nouvelle tâche de *Temporal Summarization* va faire son apparition à TREC pour l'année 2013. Le but sera ici de produire des résumés évoquant des grands événements (ouragans, élections...) et d'ordonner les différentes phrases chronologiquement.

Au cours de la dernière décennie, de nombreux chercheurs se sont penchés sur la problématique du résumé automatique. La quasi-totalité des approches proposées recourent à des méthodes d'extraction où il s'agit d'identifier les unités textuelles, le plus souvent des phrases, les plus importantes des documents. Les phrases les plus pertinentes sont ensuite assemblées pour générer le résumé.

De nombreuses méthodes ont été utilisées pour évaluer l'importance des phrases, e.g. (Barzilay et al., 1997; Radev et al., 2004). Parmi elles, les méthodes basées sur les modèles de graphes (Mihalcea, 2004) donnent de bons résultats. L'idée est de représenter le texte sous la forme d'un graphe d'unités textuelles (phrases) inter-connectées par des relations de similarité. Des algorithmes d'ordonnancement tels que PAGERANK (Page et al., 1999) sont ensuite utilisés pour sélectionner les phrases les plus centrales dans le graphe.

Le résumé automatique orienté (Dang, 2005) est probablement la tâche qui se rapproche le plus de la contextualisation automatique. Il s'agit de générer un résumé répondant à un besoin utilisateur exprimé sous la forme d'une requête. Une grande partie des approches proposées reposent sur des méthodes de résumé automatique existantes et y ajoutent divers critères de pertinence par rapport à la requête, e.g. (Boudin et al., 2008). Parmi les différentes méthodes utilisées pour estimer la pertinence des phrases, plusieurs modèles issus de la RI donnent de bons résultats (Wei et al., 2008).

A.3 Recherche de phrases candidates contextuelles issues de Wikipédia

Dans le cadre de la tâche Tweet Contextualization d'INEX, le contexte d'un Tweet est défini par un texte composé de 500 mots au maximum et dont les phrases sont issues de Wikipédia. La Figure A.1 illustre la méthodologie que nous utilisons. Cette section détaille le processus de sélection des phrases candidates qui pourront composer le contexte.

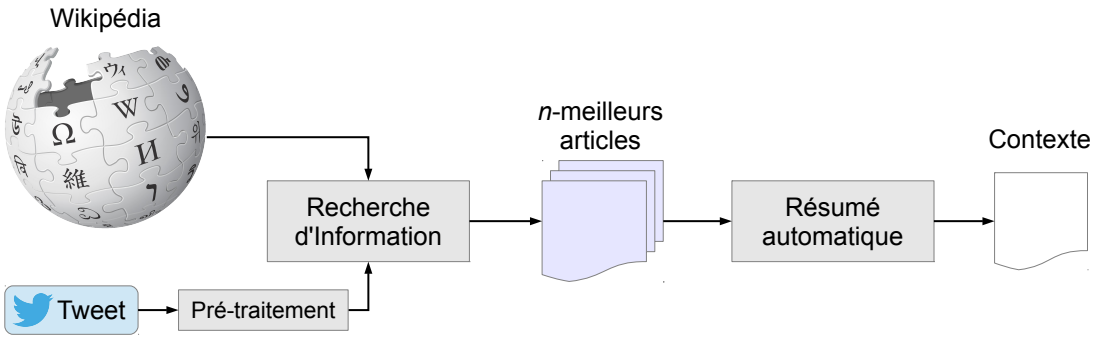

FIGURE A.1 – *Méthodologie de contextualisation d'un Tweet à partir de Wikipédia.*

A.3.1 Interprétation des #HashTags et formatage des Tweets

La première étape que nous effectuons consiste à appliquer un ensemble de prétraitements aux Tweets. Il s'agit de formater le contenu de ces derniers en vue de l'étape de recherche d'information.

Le symbole #, appelé *hashtag*, est utilisé pour signaler des mots-clés ou des sujets dans un Tweet. Il a été créé par les utilisateurs de Twitter comme un moyen permettant de catégoriser leurs messages. Les utilisateurs emploient les *hashtags* avant un mot-clé ou une phrase pertinente (sans espace) de leurs Tweets. Ils agissent comme un moyen de catégorisation et d'étiquetage et sont ainsi des marqueurs d'informations importantes directement fournis par l'auteur. Il semble donc logique de privilégier leur utilisation dans le cadre d'une récupération d'articles Wikipédia liés à un Tweet.

La principale difficulté avec l'utilisation des *hashtags* vient du fait qu'ils sont pour la plupart composés de plusieurs mots concaténés. La figure A.2 illustre ce problème avec le hashtag #WhitneyHouston. Dans ce cas précis, il ne serait pas possible pour un système de recherche d'information classique de renvoyer des documents liés à Whitney Houston étant donné que ces deux mots n'apparaissent pas dans le Tweet.

Pour résoudre ce problème, nous avons utilisé un algorithme de segmentation automatique de mots basé sur celui présenté dans le chapitre « Natural Language Corpus Data » du livre « Beautiful Data » (Segaran et Hammerbacher, 2009). Nous calculons le

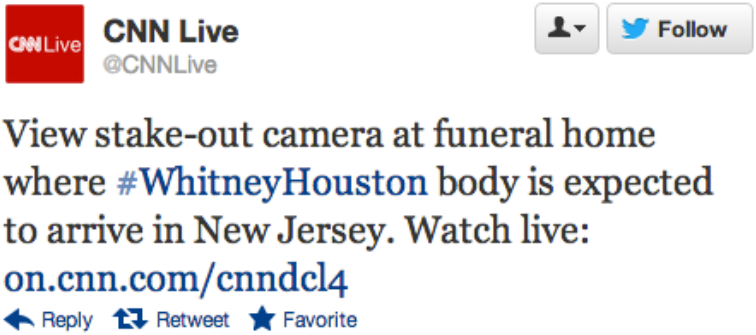

FIGURE A.2 – *Exemple d'un Tweet issu de la collection INEX Tweet Contextualization pour l'année 2012.*

découpage le plus probable d'un *hashtag* à l'aide des probabilités d'apparition d'unigrammes et de bigrammes au sein du corpus Bing N-Gram [2]. Ainsi, chaque *hashtag* présent dans le Tweet initial est remplacé par sa version découpée.

Twitter étant un réseau social, l'interaction entre les utilisateurs est au centre de son fonctionnement. Ainsi, un Tweet peut contenir différentes mentions destinées à d'autres personnes, comme par exemple une réponse ou un retweet. Un Tweet réponse commence par un @ suivi du pseudonyme d'un (ou plusieurs) utilisateur(s). Cela permet notamment de créer une discussion spontanée entre plusieurs personnes. Quant au retweet, il consiste à reposter le Tweet d'une autre personne. Parfois les utilisateurs tapent *RT* au début d'un Tweet pour indiquer qu'ils repostent le contenu d'un autre utilisateur. Ce n'est pas une commande ou une fonction officielle de Twitter, mais cela signifie qu'ils citent le Tweet d'un autre utilisateur. Néanmoins ces différentes mentions n'apportent rien au contenu informatif du Tweet, nous les supprimons donc simplement. Les mots outils sont également supprimés en utilisant la liste standard INQUERY fournie avec le système de recherche d'information Indri [3]. La sortie finale de cette étape de formatage est un Tweet nettoyé, sans mots-outils, ni *hashtags* collés, ni mentions inutiles.

A.3.2 Recherche d'articles Wikipédia

La sélection d'articles Wikipédia apportant des informations contextuelles par rapport à un Tweet est une étape cruciale pour trouver les phrases qui vont former le contexte. Nous présentons dans cette section les différentes méthodes de recherche documentaire que nous utilisons dans nos expériences.

2. http://web-ngram.research.microsoft.com/info/
3. http://www.lemurproject.org/

Modèle de base

L'une des approches standard de la recherche d'information par modèle de langue se fait avec un modèle de vraisemblance de la requête. Ce modèle mesure la probabilité que la requête puisse être générée à partir d'un document donné, ainsi les documents sont ordonnés en se basant sur cette probabilité. Soit θ_D le modèle de langue estimé en se basant sur un document D, le score d'appariement entre D et une requête $\mathcal{T}$ est défini par la probabilité conditionnelle suivante :

$$P(\mathcal{T}|\theta_D) = \prod_{t \in \mathcal{T}} f_T(t, D) \tag{A.1}$$

Un des points importants dans le paramétrage des approches par modèle de langue est le lissage des probabilités nulles. Dans ce travail, θ_D est lissé en utilisant le lissage de Dirichlet (Zhai et Lafferty, 2004), on a donc :

$$f_T(t, D) = \prod_{t \in \mathcal{T}} \frac{c(t, D) + \mu \cdot P(t|\mathcal{C})}{|D| + \mu} \tag{A.2}$$

où $c(t, D)$ est le nombre d'occurrences du mot t dans le document D. $\mathcal{C}$ représente la collection de documents et μ est le paramètre du lissage de Dirichlet (nous fixons $\mu = 2500$ tout au long de cette annexe).

Une des limitations évidente de l'approche par unigramme est qu'elle ne tient pas compte des dépendances ou des relations qu'il peut y avoir entre deux termes adjacents dans la requête. Le modèle MRF (Markov Random Field) (Metzler et Croft, 2005) est une généralisation de l'approche par modèle de langue et résoud spécifiquement ce problème. L'intuition derrière ce modèle est que des mots adjacents de la requête sont susceptibles de se retrouver proches dans les documents pertinents. Trois différents types de dépendances sont considérés :

1. l'indépendance des termes de la requête (ce qui revient à un modèle de langue standard prenant en compte uniquement les unigrammes),
2. l'apparition exacte de bigrammes de la requête,
3. et l'apparition de bigrammes de la requête dans un ordre non défini au sein d'une fenêtre de mots.

Le modèle propose deux fonctions supplémentaires pour deux autres types de dépendances qui agissent sur les bigrammes de la requête :

$$f_O(t_i, t_{i+1}, D) = \frac{c(\#1(t_i, t_{i+1}), D) + \mu \cdot \frac{c(\#1(t_i, t_{i+1}), \mathcal{C})}{|\mathcal{C}|}}{|D| + \mu} \tag{A.3}$$

$$f_U(t_i, t_{i+1}, D) = \frac{c(\#uw8(q_i, q_{i+1}), D) + \mu \cdot \frac{c(\#uw8(q_i, q_{i+1}), \mathcal{C})}{|\mathcal{C}|}}{|D| + \mu} \tag{A.4}$$

La fonction $f_O(q_i, q_{i+1}, D)$ considère la correspondance exacte de deux mots adjacents de la requête. Elle est dénotée par l'indice O. La seconde est dénotée par l'indice

U et considère la correspondance non ordonnée de deux mots au sein d'une fenêtre de 8 unités lexicales. Ici, $c(\#1(t_i, t_{i+1}), D)$ est le nombre d'occurrences du bigramme (t_i, t_{i+1}) dans le document D. Comparativement, $c(\#uw8(t_i, t_{i+1}), D)$ est le nombre d'occurences des deux mots de la requête t_i et t_{i+1} au sein d'une fenêtre non ordonnée composée de 8 termes du document D.

Finalement, le score d'un article Wikipédia D par rapport à un Tweet formaté $\mathcal{T}$ est donné par la fonction suivante :

$$\begin{aligned} s_{MRF}(\mathcal{T}, D) \quad = \quad & \lambda_T \prod_{t \in \mathcal{T}} f_T(t, D) \; + \\ & \lambda_O \prod_{i=1}^{|Q|-1} f_O(t_i, t_{i+1}, D) \; + \\ & \lambda_U \prod_{i=1}^{|Q|-1} f_U(t_i, t_{i+1}, D) \end{aligned} \tag{A.5}$$

où λ_T, λ_O et λ_U sont des paramètres libres dont la somme est égale à 1. Dans nos expériences nous fixons ces paramètres en suivant les recommandations des auteurs ($\lambda_T = 0,85$, $\lambda_O = 0,10$ et $\lambda_U = 0,05$).

Intégration de *hashtags*

Les *hashtags* peuvent être considérés comme des étiquettes définies manuellement par les auteurs des Tweets. Ce sont par conséquent des marqueurs évidents d'informations importantes. Ils peuvent également être considérés comme des requêtes courtes, sorte d'abréviation du Tweet. Considérons le Tweet $\mathcal{T}$ suivant :

> « All #Airbus #A380 Jumbo Jets Ordered To Be Inspected For Wing Cracks - Neon Tommy : http ://t.co/SofXXzCN »

Le sujet principal est correctement représenté par un ensemble de *hashtags* $H_\mathcal{T} = \{"airbus", "a380"\}$. Nous pouvons ainsi le considérer comme une simplification du Tweet ou encore une expression des informations les plus importantes. Un parallèle peut également être fait avec les *topics* de TREC qui sont traditionnellement composés d'une requête courte (2 à 5 mots-clés) et d'une description plus détaillée du besoin d'information (pouvant comprendre plusieurs phrases).

Nous introduisons donc les *hashtags* de façon explicite dans la fonction de score des articles Wikipédia de notre système. Soient un Tweet $\mathcal{T}$ et ses *hashtags* $H_\mathcal{T}$, le score d'un article Wikipédia D est donné par :

$$s(\mathcal{T}, H_\mathcal{T}, D) = \alpha \, s_{MRF}(H_\mathcal{T}, D) + (1 - \alpha) \, s_{MRF}(\mathcal{T}, D) \tag{A.6}$$

Le paramètre α permet de maintenir la balance entre l'influence des *hashtags* seuls et le Tweet entier. Nous nous plaçons dans le cadre d'une contextualisation en temps réel, et la nature très hétérogène des Tweets ne nous semble pas adaptée pour effectuer un apprentissage *a priori* de ce paramètre. De plus, les *hashtags* peuvent avoir une utilité parfois très limitée voire nulle, comme dans l'exemple suivant :

> « U Just Heard “Hard To Believe" by @andydavis on the @mtv Teen Mom 2 Finale go 2 http ://t.co/iwb2JuL8 for info #iheardit on MTV »

Dans ce cas-ci, « I heard it on MTV » est une phrase d'accroche de type publicitaire et n'apporte rien pour la compréhension du Tweet. L'importance des *hashtags* est donc elle aussi contextuelle et dépend de leur pouvoir discriminant. Nous choisissons d'estimer ce pouvoir discriminant en calculant un score de clarté (Cronen-Townsend et Croft, 2002). Ce score est en réalité la divergence de Kullback-Leibler entre le modèle de langue de l'ensemble de *hashtags* et le modèle de langue de la collection $\mathcal{C}$ d'articles Wikipédia :

$$\alpha = \sum_{w \in V} P(w|H_{\mathcal{T}}) \log \frac{P(w|H_{\mathcal{T}})}{P(w|\mathcal{C})} \tag{A.7}$$

où V représente le vocabulaire. Le modèle de langue des *hashtags* est estimé par retour de pertinence simulé :

$$P(w|H_{\mathcal{T}}) = \sum_{D \in R} P(w|D)P(D|H_{\mathcal{T}}) \tag{A.8}$$

Nous utilisons pour cela une approche standard de retour de pertinence simulé. Celle-ci consiste à récupérer l'ensemble R constitué des 5 premiers documents de la collection $\mathcal{C}$ renvoyés pour la requête $H_{\mathcal{T}}$. Dans le modèle des *hashtags*, la probabilité $P(D|H_{\mathcal{T}})$ est estimée en appliquant le théorème de Bayes : $P(D|H_{\mathcal{T}}) = P(H_{\mathcal{T}}|D)P(D)$, où la probabilité $P(D)$ est égale à zéro pour les documents qui ne contiennent aucun mot de la requête. Plus les documents utilisés pour estimer le modèle de langue des *hashtags* sont homogènes, plus la divergence de Kullback-Leibler augmente. Ainsi le paramètre α permet de quantifier à quel point les *hashtags* sont précis et à quel point ils permettent de sélectionner des documents distincts du reste de la collection.

Seuls 23% des Tweets utilisés dans l'évaluation officielle de la tâche *Tweet Contextualization* d'INEX 2012 contiennent des *hashtags*. Lorsqu'il n'y en a pas, nous fixons logiquement $\alpha = 0$ dans l'équation A.6.

A.3.3 Génération des phrases candidates

Pour un Tweet donné, nous sélectionnons les n articles Wikipédia les plus pertinents selon l'équation A.6. Chaque article est découpé en phrases en utilisant la méthode PUNKT de détection de changement de phrases mise en œuvre dans `nltk` [4].

Dans ce travail nous fixons $n = 5$, et toutes les phrases des 5 premiers articles sont considérées comme des phrases candidates. Nous calculons ensuite différentes caractéristiques pour chacune de ces phrases qui nous permettrons de les classer et, ainsi, de former le contexte. Nous détaillons ces caractéristiques dans la section suivante.

4. http://nltk.org/

A.4 Choix des phrases et formation du contexte

Pour pouvoir être compréhensible dans un cas d'utilisation mobile (sur un *smartphone* par exemple), le contexte doit avoir une taille limitée. Les recommandations de la tâche *Tweet Contextualization* d'INEX fixent la taille limite du contexte à 500 mots. Dans cette section, nous présentons la méthode que nous utilisons pour sélectionner les phrases candidates les plus pertinentes et générer le contexte.

A.4.1 Caractéristiques des phrases

Plusieurs caractéristiques entrent en compte lors de la sélection des phrases candidates. Ces dernières peuvent être regroupées en quatre catégories :

1. Importance de la phrase vis-à-vis du document d'où elle provient
2. Pertinence de la phrase par rapport au Tweet (y compris les *hashtags*)
3. Pertinence de la phrase par rapport à une page web dont l'URL est dans le Tweet
4. Pertinence du document d'où provient la phrase par rapport au Tweet

Nous détaillons et justifions dans cette section le calcul des différentes caractéristiques que nous utilisons ensuite pour ordonner les phrases par importance et former le contexte. Nous rappelons quelques notations déjà utilisées dans cette annexe et nous en introduisons de nouvelles dans le tableau suivant :

$\mathcal{T}$	un tweet nettoyé
$H_{\mathcal{T}}$	les *hashtags* du Tweet $\mathcal{T}$
$U_{\mathcal{T}}$	l'URL présente dans le Tweet $\mathcal{T}$
S	une phrase candidate

Les caractéristiques décrites ci-dessous sont largement basées sur le calcul de mesures de recouvrement et de similarité cosine entre une phrase candidate $S = \{m_1, m_2, ..., m_i\}$ et un Tweet $\mathcal{T} = \{m_1, m_2, ..., m_j\}$. Soit $| \bullet |$ le cardinal de l'ensemble $\bullet$, le recouvrement en mots est donné par :

$$recouvrement(\mathcal{T}, S) = \frac{|S \cap \mathcal{T}|}{min(|S|, |\mathcal{T}|)}$$

Aussi, soient $\vec{S}$ et $\vec{\mathcal{T}}$ les représentations vectorielles de S et $\mathcal{T}$, et $||\vec{\bullet}||$ la norme du vecteur $\vec{\bullet}$, la similarité cosine est donnée par :

$$cosine(\mathcal{T}, S) = \frac{\vec{S} \cdot \vec{\mathcal{T}}}{\sqrt{||\vec{S}||\ ||\vec{\mathcal{T}}||}}$$

Les mesures décrites précédemment sont calculées à partir des représentations lexicales nettoyées des phrases et des Tweets. Nous supprimons les mots outils et appliquons le méthode de racinisation (*stemming*) des mots de Porter, et ce uniquement afin

de calculer les différentes caractéristiques. La représentation des Tweets utilisée pour effectuer la recherche d'article Wikipédia n'utilise pas de racinisation.

Importance de la phrase dans le document

L'importance d'une phrase par rapport au document dans lequel elle apparaît est estimée avec la méthode TextRank (Mihalcea, 2004). Chaque document est représenté sous la forme d'un graphe pondéré non dirigé G dans lequel les noeuds V correspondent aux phrases, et les arêtes E sont définies en fonction d'une mesure de similarité. Cette mesure détermine le nombre de mots communs entre les deux phrases, les mots outils ayant été au préalable supprimés et les mots restants *stemmés* avec l'algorithme de Porter. Pour éviter de favoriser les phrases longues, cette valeur est normalisée par les longueurs des phrases. Soit $\text{freq}(m, S)$ la fréquence du mot m dans la phrase S, la similarité entre les phrases S_i et S_j est définie par :

$$Sim(S_i, S_j) = \frac{\sum_{m \in S_i, S_j} \text{freq}(m, S_i) + \text{freq}(m, S_j)}{\log(|S_i|) + \log(|S_j|)}$$

L'importance d'une phrase est évaluée en tenant compte de l'intégralité du graphe. Nous utilisons une adaptation de l'algorithme PAGERANK (Page et al., 1999) qui inclut les poids des arêtes. Le score de chaque sommet V est calculé itérativement jusqu'à la convergence par :

$$p(V_i) = (1 - d) + d \times \sum_{V_j \in \text{voisins}(V_i)} \frac{Sim(S_i, S_j)}{\sum_{V_k \in \text{voisins}(V_i)} Sim(S_k, S_i)} \, p(V_i)$$

où d est un « facteur d'amortissement » (typiquement dans l'intervalle $[0.8, 0.9]$) et $\text{voisins}(V_i)$ représente l'ensemble des nœuds connectés à V_i. Le score de la phrase S correspond au score du nœud qui la représente dans le graphe.

$$c_1 = p(S)$$

Pertinence de la phrase par rapport au Tweet

Intuitivement, les indicateurs de pertinence devraient être les plus importants pour sélectionner des phrases donnant des informations contextuelles par rapport au Tweet. Le recouvrement et la similarité cosine entre un Tweet $\mathcal{T}$ et une phrase candidate S sont les premières caractéristiques que nous avons mis en place.

$$c_2 = recouvrement(\mathcal{T}, S) \qquad c_3 = cosine(\mathcal{T}, S)$$

Tout en gardant la logique de l'utilisation des *hashtags*, nous calculons le recouvrement et la similarité cosine entre chaque phrase et l'ensemble des hashtags du Tweet.

$$c_4 = recouvrement(H_\mathcal{T}, S) \qquad c_5 = cosine(H_\mathcal{T}, S)$$

Pertinence de la phrase par rapport à une page web

Les Tweets contiennent parfois des URLs, liens pointant vers des pages web porteuses d'informations contextuelles. Nous utilisons le même type de mesure que précédemment et nous calculons ainsi le recouvrement et la similarité cosine entre une phrase candidate et le titre $titre(U_\mathcal{T})$ de la page web.

$$c_6 = recouvrement\big(titre(U_\mathcal{T}), S\big) \qquad c_7 = cosine\big(titre(U_\mathcal{T}), S\big)$$

De la même façon, nous calculons ces deux mesures entre le contenu entier $page(U_\mathcal{T})$ de la page web et une phrase candidate.

$$c_8 = recouvrement\big(page(U_\mathcal{T}), S\big) \qquad c_9 = cosine\big(page(U_\mathcal{T}), S\big)$$

Pertinence du document par rapport au Tweet

Les articles Wikipédia à partir desquels les phrases candidates sont extraites ont des importances contextuelles différentes par rapport à un Tweet donné. Ainsi, une phrase provenant d'un article bien classé a plus de chance d'être importante qu'une phrase provenant d'un article mal classé. Pour capturer ce comportement, nous définissons la dernière caractéristique comme étant le score d'un document par rapport à un Tweet et ses *hashtags*, normalisé sur l'ensemble R de tous les documents renvoyés :

$$c_{10} = \frac{s(\mathcal{T}, H_\mathcal{T}, D)}{\sum_{D' \in R} s(\mathcal{T}, H_\mathcal{T}, D')}$$

Score final d'une phrase candidate

Le score d'importance de chaque phrase candidate est obtenu par la combinaison linéaire des scores des critères présentés ci-dessus.

$$score = \sum_x \log(c_x + 1)$$

A.4.2 Génération du contexte

Le contexte d'un Tweet est généré par assemblage des phrases candidates les plus importantes. Il est cependant possible que le contexte ainsi obtenu contienne plusieurs phrases redondantes, ce qui dégrade à la fois sa lisibilité et son contenu informatif. Pour résoudre ce problème, nous ajoutons une étape supplémentaire lors de la génération des contextes.

Nous générons tous les contextes possibles à partir des combinaisons des N phrases ayant les meilleurs scores, en veillant à ce que le nombre total de mots soit optimal (i.e. en dessous du seuil de 500 mots et qu'il soit impossible d'ajouter une autre phrase sans dépasser ce seuil). La valeur N est fixée empiriquement au nombre minimum de phrases de meilleurs scores pour atteindre 500 mots, plus quatre phrases. Le contexte retenu au final est celui possédant le score global le plus élevé, ce score étant calculé comme le produit du score de la diversité du résumé, estimé par le nombre de n-grammes différents, et de la somme des scores des phrases.

Afin d'améliorer la lisibilité du contexte généré, si deux phrases sont extraites à partir d'un même document, l'ordre original du document est conservé.

A.5 Évaluation et discussion

Cette section débute par la description de la collection de test que nous utilisons. Nous présentons ensuite les résultats de notre méthode de contextualisation, et nous analysons l'importance des différentes caractéristiques dans le processus de sélection des phrases candidates.

A.5.1 Cadre expérimental

Nous utilisons la collection de test de la tâche *Tweet Contextualization* d'INEX 2012 pour nos expérimentations ainsi que les différentes données mises à disposition par les organisateurs (SanJuan et al., 2012). La collection de documents Wikipédia est basée sur une capture de la version anglaise de l'encyclopédie en ligne datant de Novembre 2011 et comprend 3 691 092 articles. Nous avons indexé cette collection avec le moteur de recherche libre Indri [5] en supprimant les mots-outils présents dans la liste INQUERY. Une racinisation légère des mots est également appliquée par l'algorithme de Krovetz.

La collection de test comprend au total 1126 Tweets pour lesquels un système doit produire un contexte. Cependant, nous n'utilisons que le sous-ensemble de 63 Tweets pour lesquels des jugements de pertinence ont été réalisés. Ces jugements ont été générés par un processus de groupement des dix premières phrases des contextes de tous les participants qui ont ensuite été jugées manuellement par les organisateurs.

La mesure d'évaluation développée pour cette tâche ne prend pas en compte les exemples négatifs, seules les phrases jugées pertinentes ont été conservées. Les jugements sont donc un ensemble de phrases directement issues de Wikipédia et jugées pertinentes par les organisateurs en fonction de leur importance contextuelle par rapport à un Tweet. Certains Tweets peuvent ainsi avoir un contexte de référence composé d'un grand nombre de phrases, tandis que d'autres peuvent en avoir un nombre très réduit. Ces différences de taille ainsi que le fait qu'une seule référence soit disponible pour chaque Tweet empêchent l'utilisation de la mesure classique ROUGE (Lin, 2004)

5. http://www.lemurproject.org/indri.php

pour l'évaluation des contextes. Les organisateurs ont donc proposé une mesure d'évaluation qui calcule une divergence entre le contexte produit et les phrases jugées pertinentes (SanJuan et al., 2012). Elle peut prendre en compte des unigrammes stricts, des bigrammes ou des bigrammes avec possibilité d'insertion. La mesure principale utilisée pour départager les systèmes est la troisième (« Bigrammes à trous »).

A.5.2 Résultats de contextualisation

Nous reportons dans le tableau A.1 les résultats de contextualisation pour trois méthodes de recherche d'articles Wikipédia présentées dans la section A.3 : l'approche standard par modèle de langue pour la RI (équation A.1, notée **QL**), l'approche **MRF** (équation A.5) et l'approche mixant MRF pour le Tweet et pour ses *hashtags* (équation A.6, notée **MRFH**). Les scores étant calculés en tant que divergences, les scores les plus bas correspondent aux systèmes les plus performants.

	Unigrammes	**Bigrammes**	**Bigrammes à trous**
QL	0,7967	0,8923	0,8940
MRF	0,7883	0,8851	0,8865
MRFH	**0,7872**	**0,8815**	**0,8839**
1^{er} INEX 2012	0,7734	0,8616	0,8623

TABLE *A.1* – *Résultats de contextualisation pour les 3 différents algorithmes de RI et l'ensemble des caractéristiques pour l'attribution des scores.*

Nous remarquons que les résultats sont relativement proches et qu'il n'y a pas de différence significative entre les trois approches. Néanmoins l'approche qui considère les *hashtags* dans la fonction de score des documents obtient les meilleurs résultats (avec $p = 0,17$ pour un t-test entre **QL** et **MRFH**). Les faibles différences observées entre les méthodes sont sans doute dues à la relative similarité entre les modèles de RI, même si l'on voit que l'utilisation de *hashtags* améliore sensiblement les scores. Il est néanmoins difficile de tirer des conclusions définitives étant donné que seuls 23% des Tweets utilisés pour l'évaluation contiennent au moins un *hashtag*. Nous reportons pour information les résultats officiels du meilleur système mais, à l'heure actuelle, leur approche n'est pas connue en détails. Grâce à l'analyse détaillée de l'influence des différentes caractéristiques proposée en section A.5.3, nous avons pu établir que la borne supérieure de notre système était de $0,8824$ ce qui est encore loin du meilleurs score. Cependant, il n'y a pas de différence statistiquement significative entre notre approche **MRFH** et le meilleur système d'INEX 2012.

Nous pensons que cette différence de score est due à deux biais lors de l'évaluation. Le premier se situe lors de la constitution des jugements : pour chaque Tweet, uniquement les dix premières phrases de chaque système sont considérées pour être ensuite jugées manuellement. Or, un des buts de cette tâche étant la lisibilité, les phrases les plus informatives ne se trouvent pas forcément en début de contexte pour pouvoir favoriser la cohérence globale et l'enchaînement des phrases. Le deuxième biais se situe

au sein de la mesure d'évaluation elle-même. En effet, elle ne possède pas de composante visant à pénaliser les phrases non pertinentes. Ainsi, remplir le contexte avec des phrases très diverses permettra toujours d'obtenir des meilleurs scores que de faire attention et de ne pas ajouter de phrases dégradant la cohérence du contexte.

Pour illustrer ces biais, nous présentons dans la figure A.3 un Tweet ainsi que le contexte produit par notre méthode qui a obtenu un score nul lors de notre évaluation. Or, même si ce contexte n'est à l'évidence pas parfait, il apporte tout de même des informations contextuelles sur le Tweet. On peut en effet apprendre que Van Gogh était un peintre et que "The Starry Night" est une de ses compositions, et dont le style transparaît sur d'autres de ses peintures.

> « Very cool ! An interactive animation of van Gogh's "The Starry Night" http ://t.co/ErJCPObh (thanks @juliaxgulia) »
>
> *Vincent van Gogh painted at least 18 paintings of "olive trees", mostly in Saint-Rémy in 1889. The olive tree paintings had special significance for Van Gogh. One painting, "Olive Trees in a Mountainous Landscape (with the Alpilles in the Background)", a complement to "The Starry Night", symbolized the divine. In both "The Starry Night" and his olive tree paintings, Van Gogh used the intense blue of the sky to symbolize the "divine and infinite presence" of Jesus. ...*

FIGURE A.3 – *Les premières phrases d'un contexte produit par notre méthode. La mesure d'évaluation a attribué un score nul à ce contexte.*

Il est à noter que si les résultats obtenus par notre méthode lors de la campagne INEX ne sont pas les meilleurs, notre approche est celle qui apporte le meilleur compromis entre informativité et lisibilité. Néanmoins l'évaluation de lisibilité, qui a été faite manuellement, n'est pas reproductible et le travail présenté dans cette annexe est différent de celui réalisé pour INEX, nous ne pouvons donc pas reporter de résultats.

A.5.3 Importance des différentes caractéristiques

En l'état, les caractéristiques calculées pour chacune des phrases candidates ont toutes la même importance dans le score final attribué à une phrase. Étant donné que les Tweets proposés pour la tâche QA@INEX 2011 n'avaient ni *hashtags* ni URL, nous n'avons pas pu entraîner notre système pour qu'il apprenne les poids de ces caractéristiques. Nous proposons néanmoins une analyse de leur importance sur les Tweets de l'année 2012. Bien évidemment, les chiffres présentés ici ne nous ont pas servi à paramétrer notre système, et les résultats présentés dans la section précédente ne tiennent pas compte de ces poids.

En principe, nous pourrions utiliser n'importe quelle méthode d'apprentissage pour apprendre les poids optimaux. Ici, nous utilisons un modèle de régression logistique. Ainsi, nous calculons toutes les caractéristiques présentées dans la section A.4.1 pour

chacune des phrases extraites et nous les lions à leur pertinence $r \in \{0,1\}$. La variable r peut ainsi être vue comme une mesure de la contribution totale de toutes les caractéristiques utilisées dans le modèle et est habituellement définie comme $r = \bar{w}\bar{x}$. Spécifiquement, $\bar{x}$ est un vecteur de valeurs numériques représentant les caractéristiques, et $\bar{w}$ représente l'ensemble des poids relatifs de chaque caractéristique. Un poids positif signifie que la caractéristique correspondante améliore la probabilité d'obtenir r, un poids négatif signifie qu'elle la dégrade.

Caractéristique	Nom	Valeur	Significativité
$c1$	**TextRank**	$8,996$	$p < 2^{-16}$
$c2$	**Recouvrement Tweet**	$2,496$	$p = 2,38^{-6}$
$c3$	**Cosine Tweet**	$5,849$	$p = 4^{-15}$
$c4$	Recouvrement *hashtags*	$-2,051$	$p = 0,1368$
$c5$	Cosine *hashtags*	$0,671$	$p = 0,3074$
$c6$	Recouvrement titre URL	$1,373$	$p = 0,2719$
$c7$	Cosine titre URL	$0,788$	$p = 0,6287$
$c8$	Recouvrement page URL	$0,543$	$p = 0,4337$
$c9$	**Cosine page URL**	$10,374$	$p = 0,0195$
$c10$	**Score document**	$0,782$	$p < 2^{-16}$

TABLE *A.2* – *Valeurs optimales des poids des caractéristiques calculées pour les phrases candidates.*

Nous pouvons observer dans le tableau A.2 que les caractéristiques les plus significatives pour estimer la pertinence d'une phrase ne sont pas ou peu liées au Tweet. En effet, le TextRank ne concerne que l'importance de la phrase par rapport aux autres phrases du document, et le score du document est un score global. Le Tweet n'intervient dans ces cas qu'au moment de la recherche des articles. Comme on aurait pu s'y attendre, le recouvrement et la similarité cosine entre le Tweet et une phrase sont également des marqueurs de pertinence. Étonnamment, les *hashtags* ont une influence parfois négative et généralement aléatoire, tout comme les titres des pages web pointées par les URLs. Mais comme nous l'avons dit dans la section précédente, les *hashtags* sont très peu nombreux dans les Tweets utilisés pour l'évaluation, ce qui peut expliquer ce comportement aléatoire. Enfin, seule la similarité cosine entre une phrase et le contenu d'une page web semble être faiblement significative.

Globalement, une phrase apporte des informations contextuelles par rapport à un Tweet si elle contient les mêmes mots que celui-ci, si elle apparaît dans un document pertinent, et si elle fait partie des phrases les plus importantes de ce dernier.

A.6 Conclusion

Nous avons présenté dans cette annexe une première approche pour la contextualisation de messages courts. Celle-ci se fait dans le cadre de la tâche Tweet Contextualization d'INEX et utilise Wikipédia comme corpus de référence pour la constitution

des contextes. Les résultats de nos expériences suggèrent que l'utilisation des *hashtags* présents dans les Tweets aide à la recherche d'articles Wikipédia qui contiennent des phrases apportant des informations contextuelles. Nous avons également examiné l'influence de différentes caractéristiques calculées sur les phrases candidates ainsi que leur importance. Il apparaît que pour constituer un contexte, il est préférable de choisir les phrases les plus importantes des articles Wikipédia extraits. Les mesures de similarité entre les phrases et les Tweets sont également des indicateurs fiables, tandis que les *hashtags* semblent ici n'avoir qu'une influence aléatoire.

Une des limitations de notre approche est que le nombre d'articles Wikipédia utilisés pour extraire les phrases candidates est fixé manuellement. Idéalement, une méthode déterminant automatiquement ce nombre en fonction du Tweet permettrait de réduire le bruit et augmenterait indirectement la qualité des contextes générés. Même si l'utilisation du score du document permet de réduire l'effet de cette limitation, nous laissons cette amélioration pour des travaux futurs.

Liste des illustrations

Liste des tableaux

Bibliographie

(Agrawal et al., 2009) R. Agrawal, S. Gollapudi, A. Halverson, et S. Ieong, 2009. Diversifying Search Results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, New York, NY, USA, 5–14. ACM.

(Alonso et Mizzaro, 2009) O. Alonso et S. Mizzaro, 2009. Can we get rid of TREC assessors ? Using Mechanical Turk for relevance assessment. In *Proceedings of the SIGIR 2009 Workshop on the Future of IR Evaluation*, 15–16.

(Alonso et al., 2008) O. Alonso, D. E. Rose, et B. Stewart, 2008. Crowdsourcing for Relevance Evaluation. *SIGIR Forum 42*(2), 9–15.

(AlSumait et al., 2008) L. AlSumait, D. Barbará, et C. Domeniconi, 2008. On-line LDA : Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking. In *Proceedings of the Eighth IEEE International Conference on Data Mining*, ICDM '08, 3–12.

(Andrzejewski et Buttler, 2011) D. Andrzejewski et D. Buttler, 2011. Latent Topic Feedback for Information Retrieval. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, New York, NY, USA, 600–608. ACM.

(Arun et al., 2010) R. Arun, V. Suresh, C. E. Veni Madhavan, et M. N. Narasimha Murthy, 2010. On Finding the Natural Number of Topics with Latent Dirichlet Allocation : Some Observations. In *Proceedings of the 14th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining - Volume Part I*, PAKDD'10, 391–402. Berlin, Heidelberg : Springer-Verlag.

(Bai et al., 2007) J. Bai, J.-Y. Nie, G. Cao, et H. Bouchard, 2007. Using Query Contexts in Information Retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, New York, NY, USA, 15–22. ACM.

(Bailey et al., 2003) P. Bailey, N. Craswell, et D. Hawking, 2003. Engineering a Multi-purpose Test Collection for Web Retrieval Experiments. *Information Processing & Management 39*(6), 853–871.

(Bailey et al., 2007) P. Bailey, A. P. de Vries, N. Craswell, et I. Soboroff, 2007. Overview of the TREC 2007 Enterprise Track. In E. M. Voorhees et L. P. Buckland (Eds.), *TREC*,

Volume Special Publication 500-274. National Institute of Standards and Technology (NIST).

(Bakshy et al., 2012) E. Bakshy, I. Rosenn, C. Marlow, et L. Adamic, 2012. The role of social networks in information diffusion. In *Proceedings of the 21st international conference on World Wide Web*, WWW '12, New York, NY, USA, 519–528. ACM.

(Balog et al., 2009) K. Balog, A. P. de Vries, P. Serdyukov, P. Thomas, et T. Westerveld, 2009. Overview of the TREC 2009 Entity Track. In E. M. Voorhees et L. P. Buckland (Eds.), *TREC*, Volume Special Publication 500-278. National Institute of Standards and Technology (NIST).

(Barzilay et al., 1997) R. Barzilay, M. Elhadad, et al., 1997. Using lexical chains for text summarization. In *Proceedings of the ACL workshop on intelligent scalable text summarization*, Volume 17, 10–17.

(Bates, 2006) M. J. Bates, 2006. Fundamental Forms of Information. *Journal of the American Society for Information Science and Technology 57*(8), 1033–1045.

(Bates, 2008) M. J. Bates, 2008. Hjørland's Critique of Bates' Work on Defining Information. *Journal of the American Society for Information Science and Technology 59*(5), 842–844.

(Bates, 2011) M. J. Bates, 2011. Birger Hjørland's Manichean Misconstruction of Marcia Bates' Work. *Journal of the American Society for Information Science and Technology 62*(10), 2038–2044.

(Bendersky et al., 2011) M. Bendersky, D. Metzler, et W. B. Croft, 2011. Parameterized Concept Weighting in Verbose Queries. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, New York, NY, USA, 605–614. ACM.

(Bendersky et al., 2012) M. Bendersky, D. Metzler, et W. B. Croft, 2012. Effective Query Formulation with Multiple Information Sources. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, New York, NY, USA, 443–452. ACM.

(Blei et al., 2003) D. M. Blei, A. Y. Ng, et M. I. Jordan, 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research 3*, 993–1022.

(Borlund, 2003) P. Borlund, 2003. The concept of relevance in ir. *Journal of the American Society for Information Science and Technology 54*(10), 913–925.

(Boudin et al., 2008) F. Boudin, M. El-Bèze, et J.-M. Torres-Moreno, 2008. A scalable MMR approach to sentence scoring for multi-document update summarization. In *Coling 2008 : Companion volume : Posters*, Manchester, UK, 23–26. Coling 2008 Organizing Committee.

(Boudin et Torres Moreno, 2007) F. Boudin et J. Torres Moreno, 2007. Neo-cortex : A performant user-oriented multi-document summarization system. In A. Gelbukh (Ed.), *Computational Linguistics and Intelligent Text Processing*, Volume 4394 de *Lecture Notes in Computer Science*, 551–562. Springer Berlin Heidelberg.

(Bron et al., 2010) M. Bron, K. Balog, et M. de Rijke, 2010. Ranking Related Entities : Components and Analyses. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, New York, NY, USA, 1079–1088. ACM.

(Buckley et Voorhees, 2000) C. Buckley et E. M. Voorhees, 2000. Evaluating evaluation measure stability. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, New York, NY, USA, 33–40. ACM.

(Büttcher et al., 2006) S. Büttcher, C. L. A. Clarke, et I. Soboroff, 2006. The TREC 2006 Terabyte Track. In E. M. Voorhees et L. P. Buckland (Eds.), *TREC*, Volume Special Publication 500-272. National Institute of Standards and Technology (NIST).

(Cao et al., 2008) G. Cao, J.-Y. Nie, J. Gao, et S. Robertson, 2008. Selecting Good Expansion Terms for Pseudo-relevance Feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, New York, NY, USA, 243–250. ACM.

(Cao et al., 2009) J. Cao, T. Xia, J. Li, Y. Zhang, et S. Tang, 2009. A Density-based Method for Adaptive LDA Model Selection. *Neurocomputing 72*(7-9), 1775–1781.

(Carterette et al., 2006) B. Carterette, J. Allan, et R. Sitaraman, 2006. Minimal Test Collections for Retrieval Evaluation. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, New York, NY, USA, 268–275. ACM.

(Chang et al., 2009) J. Chang, J. Boyd-Graber, C. Wang, S. Gerrish, et D. M. Blei, 2009. Reading Tea Leaves : How Humans Interpret Topic Models. In *NIPS*.

(Chang et al., 2006) Y. Chang, I. Ounis, et M. Kim, 2006. Query reformulation using automatically generated query concepts from a document space. *Information Processing & Management 42*(2).

(Chapelle et al., 2009) O. Chapelle, D. Metlzer, Y. Zhang, et P. Grinspan, 2009. Expected Reciprocal Rank for Graded Relevance. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, New York, NY, USA, 621–630. ACM.

(Chen et al., 2010) J. Chen, R. Nairn, L. Nelson, M. Bernstein, et E. Chi, 2010. Short and tweet : experiments on recommending content from information streams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, 1185–1194.

(Clarke et al., 2011a) C. L. Clarke, N. Craswell, I. Soboroff, et A. Ashkan, 2011a. A Comparative Analysis of Cascade Measures for Novelty and Diversity. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, New York, NY, USA, 75–84. ACM.

(Clarke et al., 2008) C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, et I. MacKinnon, 2008. Novelty and Diversity in Information Retrieval Evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '08, New York, NY, USA, 659–666. ACM.

(Clarke et al., 2004) C. L. A. Clarke, N. Craswell, et I. Soboroff, 2004. Overview of the TREC 2004 Terabyte Track. In E. M. Voorhees et L. P. Buckland (Eds.), *TREC*, Volume Special Publication 500-261. National Institute of Standards and Technology (NIST).

(Clarke et al., 2011b) C. L. A. Clarke, N. Craswell, I. Soboroff, et E. M. Voorhees, 2011b. Overview of the TREC 2011 Web Track. In E. M. Voorhees et L. P. Buckland (Eds.), *TREC*. National Institute of Standards and Technology (NIST).

(Cleverdon, 1962) C. Cleverdon, 1962. *Report on the Testing and Analysis of an Investigation Into Comparative Efficiency of Indexing Systems*. Cranfield, UK : College of Aeronautics.

(Cleverdon et al., 1962) C. Cleverdon, J. Mills, et M. Keen, 1962. *Factors determining the performance of indexing systems*. Cranfield, UK : College of Aeronautics.

(Collins-Thompson, 2009) K. Collins-Thompson, 2009. Reducing the Risk of Query Eexpansion via Robust Constrained Optimization. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, New York, NY, USA, 837–846. ACM.

(Cormack et al., 2011) G. V. Cormack, M. D. Smucker, et C. L. Clarke, 2011. Efficient and Effective Spam Filtering and Re-ranking for Large Web Datasets. *Information Retrieval 14*(5), 441–465.

(Cronen-Townsend et Croft, 2002) S. Cronen-Townsend et W. B. Croft, 2002. Quantifying query ambiguity. In *Proceedings of the second international conference on Human Language Technology Research*, HLT '02, San Francisco, CA, USA, 104–109. Morgan Kaufmann Publishers Inc.

(Dang, 2005) H. Dang, 2005. Overview of duc 2005. In *Proceedings of the Document Understanding Conference*.

(Deerwester et al., 1990) S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, et R. Harshman, 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science 41*(6), 391–407.

(Deveaud et al., 2013a) R. Deveaud, E. SanJuan, et P. Bellot, 2013a. Are Semantically Coherent Topic Models Useful for Information Retrieval ? In *Proceedings of the 51st*

Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers), ACL'13.

(Deveaud et al., 2013b) R. Deveaud, E. SanJuan, et P. Bellot, 2013b. Estimating Topical Context by Diverging from External Resources. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'13. ACM.

(Deveaud et al., 2013c) R. Deveaud, E. SanJuan, et P. Bellot, 2013c. Unsupervised Latent Concept Modeling to Identify Query Facets. In *Proceedings of the Tenth International Conference in the RIAO series*, OAIR'13. CID.

(Diaz et Metzler, 2006) F. Diaz et D. Metzler, 2006. Improving the Estimation of Relevance Models Using Large External Corpora. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, New York, NY, USA, 154–161. ACM.

(Egozi et al., 2011) O. Egozi, S. Markovitch, et E. Gabrilovich, 2011. Concept-Based Information Retrieval Using Explicit Semantic Analysis. *ACM Transactions on Information Systems 29*(2), 8 :1–8 :34.

(Finkelstein et al., 2002) L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, et E. Ruppin, 2002. Placing Search in Context : The Concept Revisited. *ACM Transactions on Information Systems 20*(1), 116–131.

(Fuhr et al., 2008) N. Fuhr, J. Kamps, M. Lalmas, S. Malik, et A. Trotman, 2008. Overview of the INEX 2007 Ad Hoc Track. In N. Fuhr, J. Kamps, M. Lalmas, et A. Trotman (Eds.), *Focused Access to XML Documents*, Volume 4862 de *Lecture Notes in Computer Science*, 1–23. Springer Berlin Heidelberg.

(Furnas et al., 1987) G. W. Furnas, T. K. Landauer, L. M. Gomez, et S. T. Dumais, 1987. The Vocabulary Problem in Human-system Communication. *Communications of the ACM 30*(11), 964–971.

(Gliozzo et al., 2007) A. M. Gliozzo, M. Pennacchiotti, et P. Pantel, 2007. The Domain Restriction Hypothesis : Relating Term Similarity and Semantic Consistency. In *Human Language Technologies : The 2007 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 131–138.

(Graff et Cieri, 2003) D. Graff et C. Cieri, 2003. English Gigaword. *Philadelphia : Linguistic Data Consortium LDC2003T05*.

(Griffiths et Steyvers, 2004) T. L. Griffiths et M. Steyvers, 2004. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America 101 Suppl*.

(Gyongyi et Garcia-Molina, 2005) Z. Gyongyi et H. Garcia-Molina, 2005. Spam : it's not just for inboxes anymore. *Computer 38*(10), 28–34.

(Harman, 1992a) D. Harman, 1992a. Evaluation Issues in Information Retrieval. *Information Processing & Management 28*(4), 439–440.

(Harman, 1992b) D. Harman, 1992b. Overview of the First Text REtrieval Conference (TREC-1). In *TREC*, 1–20.

(Harman, 2011) D. Harman, 2011. *Information Retrieval Evaluation*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers.

(Harman et Voorhees, 2006) D. Harman et E. M. Voorhees, 2006. TREC : An overview. *ARIST 40*(1), 113–155.

(Harter, 1992) S. P. Harter, 1992. Psychological relevance and information science. *Journal of the American Society for Information Science 43*(9), 602–615.

(Hauff, 2010) C. Hauff, 2010. *Predicting the effectiveness of queries and retrieval systems*. Thèse de Doctorat, Enschede. SIKS Dissertation Series No. 2010-05.

(Hawking, 2000) D. Hawking, 2000. Overview of the TREC-9 Web Track. In *Proceedings of the Ninth Text REtrieval Conference (TREC)*.

(He et Ounis, 2009) B. He et I. Ounis, 2009. Finding Good Feedback Documents. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, New York, NY, USA, 2011–2014. ACM.

(Hiemstra, 2001) D. Hiemstra, 2001. *Using language models for information retrieval*. Univ. Twente.

(Hjørland, 2009) B. Hjørland, 2009. The Controversy over the Concept of "Information" : A Rejoinder to Professor Bates. *Journal of the American Society for Information Science and Technology 60*(3), 643–643.

(Hjørland, 2010) B. Hjørland, 2010. The Foundation of the Concept of Relevance. *Journal of the American Society for Information Science and Technology 61*(2), 217–237. English

(Hofmann, 2001) T. Hofmann, 2001. Unsupervised Learning by Probabilistic Latent Semantic Analysis. *Machine Learning 42*, 177–196.

(Howe, 2008) J. Howe, 2008. *Crowdsourcing : Why the Power of the Crowd Is Driving the Future of Business*. Crown Publishing Group.

(Hu et al., 2009) J. Hu, G. Wang, F. Lochovsky, J.-t. Sun, et Z. Chen, 2009. Understanding User's Query Intent with Wikipedia. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, New York, NY, USA, 471–480. ACM.

(Huang et al., 2008) Y. Huang, Z. Liu, et Y. Chen, 2008. Query Biased Snippet Generation in XML Search. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, New York, NY, USA, 315–326. ACM.

(Ingwersen, 1994) P. Ingwersen, 1994. Polyrepresentation of Information Needs and Semantic Entities : Elements of a Cognitive Theory for Information Retrieval Interaction. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, New York, NY, USA, 101–110. Springer-Verlag New York, Inc.

(Järvelin et Kekäläinen, 2002) K. Järvelin et J. Kekäläinen, 2002. Cumulated Gain-based Evaluation of IR Techniques. *ACM Trans. Inf. Syst. 20*(4), 422–446.

(Jones, 1981) K. Jones, 1981. *Information retrieval experiment*. Butterworths.

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

(Jones et al., 1975) K. Jones, C. Van Rijsbergen, B. L. Research, et D. Department, 1975. *Report on the Need for and Provision of an Ideal Information Retrieval Test Collection*. British Library Research and Development reports. Cambridge Univ. Computer Lab.

(Kamps et al., 2009) J. Kamps, M. Lalmas, et B. Larsen, 2009. Evaluation in context. In M. Agosti, J. Borbinha, S. Kapidakis, C. Papatheodorou, et G. Tsakonas (Eds.), *Research and Advanced Technology for Digital Libraries*, Volume 5714 de *Lecture Notes in Computer Science*, 339–351. Springer Berlin Heidelberg.

(Kaptein et Kamps, 2011) R. Kaptein et J. Kamps, 2011. Explicit extraction of topical context. *Journal of the American Society for Information Science and Technology 62*(8), 1548–1563.

(Kaszkiel et Zobel, 1997) M. Kaszkiel et J. Zobel, 1997. Passage Retrieval Revisited. In *Proceedings of the 20Th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '97, New York, NY, USA, 178–185. ACM.

(Kazai et al., 2011) G. Kazai, J. Kamps, M. Koolen, et N. Milic-Frayling, 2011. Crowdsourcing for Book Search Evaluation : Impact of Hit Design on Comparative System Ranking. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, New York, NY, USA, 205–214. ACM.

(Keikha et al., 2011) M. Keikha, J. Seo, W. B. Croft, et F. Crestani, 2011. Predicting Document Effectiveness in Pseudo Relevance Feedback. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, New York, NY, USA, 2061–2064. ACM.

(Kelly, 2009) D. Kelly, 2009. Methods for Evaluating Interactive Information Retrieval Systems with Users. *Foundations and Trends in Information Retrieval 3*(1&#8212 ;2), 1–224.

(Koenemann et Belkin, 1996) J. Koenemann et N. J. Belkin, 1996. A Case for Interaction : A Study of Interactive Information Retrieval Behavior and Effectiveness. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '96, New York, NY, USA, 205–212. ACM.

(Koolen et al., 2009) M. Koolen, G. Kazai, et N. Craswell, 2009. Wikipedia Pages As Entry Points for Book Search. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, New York, NY, USA, 44–53. ACM.

(Krovetz, 1993) R. Krovetz, 1993. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '93, 191–202.

(Lavrenko et Croft, 2001) V. Lavrenko et W. B. Croft, 2001. Relevance Based Language Models. In *Proceedings of the 24Th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, New York, NY, USA, 120–127. ACM.

(Lehmann et al., 2013) J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, et C. Bizer, 2013. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*. Under review.

(Li et al., 2007) Y. Li, W. P. R. Luk, K. S. E. Ho, et F. L. K. Chung, 2007. Improving Weak Ad-hoc Queries Using Wikipedia As External Corpus. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, New York, NY, USA, 797–798. ACM.

(Lin, 2004) C.-Y. Lin, 2004. Rouge : A package for automatic evaluation of summaries. In S. S. Marie-Francine Moens (Ed.), *Text Summarization Branches Out : Proceedings of the ACL-04 Workshop*, Barcelona, Spain, 74–81. Association for Computational Linguistics.

(Liu et al., 2004) S. Liu, F. Liu, C. Yu, et W. Meng, 2004. An Effective Approach to Document Retrieval via Utilizing Wordnet and Recognizing Phrases. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '04, New York, NY, USA, 266–272. ACM.

(Lu et al., 2011) Y. Lu, Q. Mei, et C. Zhai, 2011. Investigating task performance of probabilistic topic models : an empirical study of PLSA and LDA. *Information Retrieval 14*, 178–203.

(Lupu, 2013) M. Lupu, 2013. Patent Retrieval. *Foundations and Trends® in Information Retrieval 7*(1), 1–97.

(Mandala et al., 1999) R. Mandala, T. Tokunaga, et H. Tanaka, 1999. Combining Multiple Evidence from Different Types of Thesaurus for Query Expansion. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, New York, NY, USA, 191–197. ACM.

(Manning et al., 2008) C. Manning, P. Raghavan, et H. Schütze, 2008. *Introduction to Information Retrieval*. An Introduction to Information Retrieval. Cambridge University Press.

(Medelyan et al., 2009) O. Medelyan, D. Milne, C. Legg, et I. H. Witten, 2009. Mining Meaning from Wikipedia. *International Journal of Human-Computer Studies 67*(9), 716–754.

(Meij et de Rijke, 2010) E. Meij et M. de Rijke, 2010. Supervised Query Modeling Using Wikipedia. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, New York, NY, USA, 875–876. ACM.

(Metzler et al., 2005) D. Metzler, Y. Bernstein, W. B. Croft, A. Moffat, et J. Zobel, 2005. Similarity Measures for Tracking Information Flow. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, New York, NY, USA, 517–524. ACM.

(Metzler et Croft, 2005) D. Metzler et W. B. Croft, 2005. A Markov Random Field Model for Term Dependencies. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, New York, NY, USA, 472–479. ACM.

(Metzler et Croft, 2007) D. Metzler et W. B. Croft, 2007. Latent Concept Expansion Using Markov Random Fields. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, New York, NY, USA, 311–318. ACM.

(Mihalcea, 2004) R. Mihalcea, 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of 42st Annual Meeting of the Association for Computational Linguistics*, ACL '04, 170–173.

(Miller, 1995) G. A. Miller, 1995. WordNet : A Lexical Database for English. *Communications of the ACM 38*(11), 39–41.

(Moriceau et al., 2009) V. Moriceau, E. SanJuan, X. Tannier, et P. Bellot, 2009. Overview of the 2009 QA Track : Towards a Common Task for QA, Focused IR and Automatic Summarization Systems. In S. Geva, J. Kamps, et A. Trotman (Eds.), *INEX*, Volume 6203 de *Lecture Notes in Computer Science*, 355–365. Springer.

(Newman et al., 2010) D. Newman, J. H. Lau, K. Grieser, et T. Baldwin, 2010. Automatic Evaluation of Topic Coherence. In *Human Language Technologies : The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Stroudsburg, PA, USA, 100–108. Association for Computational Linguistics.

(Nie, 2010) J.-Y. Nie, 2010. *Cross-Language Information Retrieval*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

(Page et al., 1999) L. Page, S. Brin, R. Motwani, et T. Winograd, 1999. The pagerank citation ranking : bringing order to the web.

(Park et Ramamohanarao, 2009) L. A. Park et K. Ramamohanarao, 2009. The Sensitivity of Latent Dirichlet Allocation for Information Retrieval. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases : Part II*, ECML PKDD '09, Berlin, Heidelberg, 176–188. Springer-Verlag.

(Ponte et Croft, 1998) J. M. Ponte et W. B. Croft, 1998. A Language Modeling Approach to Information Retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, New York, NY, USA, 275–281. ACM.

(Potthast et al., 2008) M. Potthast, B. Stein, et M. Anderka, 2008. A Wikipedia-based Multilingual Retrieval Model. In *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval*, ECIR'08, Berlin, Heidelberg, 522–530. Springer-Verlag.

(Radev et al., 2004) D. Radev, H. Jing, M. Styś, et D. Tam, 2004. Centroid-based summarization of multiple documents. *Information Processing & Management 40*, 919–938. English

(Řehůřek et Sojka, 2010) R. Řehůřek et P. Sojka, 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, 45–50. ELRA.

(Rijsbergen, 1979) C. J. V. Rijsbergen, 1979. *Information Retrieval* (2nd ed.). Newton, MA, USA : Butterworth-Heinemann.

(Robertson, 2008) S. Robertson, 2008. On the history of evaluation in IR. *Journal of Information Science 34*(4), 439–456.

(Robertson et Walker, 1994) S. E. Robertson et S. Walker, 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, New York, NY, USA, 232–241. Springer-Verlag New York, Inc.

(Rocchio, 1971) J. J. Rocchio, 1971. Relevance feedback in information retrieval. In G. Salton (Ed.), *The SMART Retrieval System : Experiments in Automatic Document Processing*, Prentice-Hall Series in Automatic Computation, Chapter 14, 313–323. Prentice-Hall, Englewood Cliffs NJ.

(Roth et Klakow, 2010) B. Roth et D. Klakow, 2010. Combining Wikipedia-based Concept Models for Cross-language Retrieval. In *Proceedings of the First International Information Retrieval Facility Conference on Adbances in Multidisciplinary Retrieval*, IRFC'10, Berlin, Heidelberg, 47–59. Springer-Verlag.

(Sagot et al., 2011) B. Sagot, K. Fort, G. Adda, J. Mariani, et B. Lang, 2011. Un turc mécanique pour les ressources linguistiques : critique de la myriadisation du travail parcellisé. In *TALN'2011 - Traitement Automatique des Langues Naturelles*, Montpellier, France.

(Sakai, 2006) T. Sakai, 2006. Evaluating Evaluation Metrics Based on the Bootstrap. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, New York, NY, USA, 525–532. ACM.

(Sandhaus, 2008) E. Sandhaus, 2008. The New York Times Annotated Corpus. *Philadelphia : Linguistic Data Consortium LDC2008T19*.

(SanJuan et al., 2012) E. SanJuan, V. Moriceau, X. Tannier, P. Bellot, et J. Mothe, 2012. Overview of the INEX 2012 Tweet Contextualization Track. In P. Forner, J. Karlgren, et C. Womser-Hacker (Eds.), *CLEF (Online Working Notes/Labs/Workshop)*.

(Sankaranarayanan et al., 2009) J. Sankaranarayanan, H. Samet, B. E. Teitler, M. D. Lieberman, et J. Sperling, 2009. Twitterstand : news in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, 42–51.

(Schamber et al., 1990) L. Schamber, M. Eisenberg, et M. S. Nilan, 1990. A Re-examination of Relevance : Toward a Dynamic, Situational Definition. *Information Processing & Management 26*(6), 755–776.

(Segaran et Hammerbacher, 2009) T. Segaran et J. Hammerbacher, 2009. *Beautiful Data : The Stories Behind Elegant Data Solutions*. O'Reilly Media.

(Stevens et al., 2012) K. Stevens, P. Kegelmeyer, D. Andrzejewski, et D. Buttler, 2012. Exploring Topic Coherence over Many Models and Many Topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, Stroudsburg, PA, USA, 952–961. Association for Computational Linguistics.

(Stock, 2010) W. G. Stock, 2010. Concepts and semantic relations in information science. *Journal of the American Society for Information Science and Technology 61*(10).

(Strube et Ponzetto, 2006) M. Strube et S. P. Ponzetto, 2006. WikiRelate ! Computing Semantic Relatedness Using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, AAAI'06, 1419–1424. AAAI Press.

(Suchanek et al., 2007) F. M. Suchanek, G. Kasneci, et G. Weikum, 2007. Yago : A Core of Semantic Knowledge. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, New York, NY, USA, 697–706. ACM.

(Svore et al., 2007) K. Svore, L. Vanderwende, et C. Burges, 2007. Enhancing Single-Document Summarization by Combining RankNet and Third-Party Sources. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, 448–457. Association for Computational Linguistics.

(Tague-Sutcliffe, 1996) J. M. Tague-Sutcliffe, 1996. Some Perspectives on the Evaluation of Information Retrieval Systems. *Journal of the American Society for Information Science - Special issue : evaluation of information retrieval systems 47*(1), 1–3.

(Tao et Zhai, 2006) T. Tao et C. Zhai, 2006. Regularized Estimation of Mixture Models for Robust Pseudo-relevance Feedback. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, New York, NY, USA, 162–169. ACM.

(Teh et al., 2006) Y. W. Teh, M. I. Jordan, M. J. Beal, et D. M. Blei, 2006. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association 101*(476), 1566–1581.

(Tunkelang, 2009) D. Tunkelang, 2009. Faceted search. *Synthesis Lectures on Information Concepts, Retrieval, and Services 1*(1), 1–80.

(Vallet et al., 2005) D. Vallet, M. Fernández, et P. Castells, 2005. An Ontology-Based Information Retrieval Model. In A. Gómez-Pérez et J. Euzenat (Eds.), *The Semantic Web : Research and Applications*, Volume 3532 de *Lecture Notes in Computer Science*, 455–470. Springer Berlin Heidelberg.

(van Rijsbergen, 1979) C. J. van Rijsbergen, 1979. *Information Retrieval*. Butterworth.

(Voorhees, 1993) E. M. Voorhees, 1993. Using WordNet to Disambiguate Word Senses for Text Retrieval. In *Proceedings of the 16Th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, New York, NY, USA, 171–180. ACM.

(Voorhees, 1994) E. M. Voorhees, 1994. Query Expansion Using Lexical-semantic Relations. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, New York, NY, USA, 61–69. Springer-Verlag New York, Inc.

(Voorhees, 2002) E. M. Voorhees, 2002. The Philosophy of Information Retrieval Evaluation. In *Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, CLEF '01, London, UK, UK, 355–370. Springer-Verlag.

(Voorhees, 2005) E. M. Voorhees, 2005. The TREC Robust retrieval track. *SIGIR Forum 39*(1), 11–20.

(Voorhees et Tice, 1999) E. M. Voorhees et D. M. Tice, 1999. The TREC-8 Question Answering Track Evaluation. In *TREC*.

(Wang et al., 2012) L. Wang, P. N. Bennett, et K. Collins-Thompson, 2012. Robust Rasking Models via Risk-sensitive Optimization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, New York, NY, USA, 761–770. ACM.

(Wei et al., 2008) F. Wei, W. Li, Q. Lu, et Y. He, 2008. Query-sensitive mutual reinforcement chain and its application in query-oriented multi-document summarization. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, 283–290.

(Wei et Croft, 2006) X. Wei et W. B. Croft, 2006. LDA-based Document Models for Adhoc Retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, New York, NY, USA, 178–185. ACM.

(White et al., 2009) R. W. White, P. Bailey, et L. Chen, 2009. Predicting User Interests from Contextual Information. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, New York, NY, USA, 363–370. ACM.

(White et al., 2010) R. W. White, P. N. Bennett, et S. T. Dumais, 2010. Predicting Short-term Interests Using Activity-based Search Context. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, New York, NY, USA, 1009–1018. ACM.

(Wu et Weld, 2007) F. Wu et D. S. Weld, 2007. Autonomously Semantifying Wikipedia. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, New York, NY, USA, 41–50. ACM.

(Wu et Weld, 2010) F. Wu et D. S. Weld, 2010. Open Information Extraction Using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, Stroudsburg, PA, USA, 118–127. Association for Computational Linguistics.

(Xu et al., 2009) Y. Xu, G. J. Jones, et B. Wang, 2009. Query Dependent Pseudo-relevance Feedback Based on Wikipedia. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, New York, NY, USA, 59–66. ACM.

(Yan et al., 2009) Y. Yan, N. Okazaki, Y. Matsuo, Z. Yang, et M. Ishizuka, 2009. Unsupervised Relation Extraction by Mining Wikipedia Texts Using Information from the Web. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP : Volume 2 - Volume 2*, ACL '09, Stroudsburg, PA, USA, 1021–1029. Association for Computational Linguistics.

(Ye et al., 2011) Z. Ye, J. X. Huang, et H. Lin, 2011. Finding a Good Query-Related Topic for Boosting Pseudo-Relevance Feedback. *JASIST 62*(4), 748–760.

(Yi et Allan, 2009) X. Yi et J. Allan, 2009. A Comparative Study of Utilizing Topic Models for Information Retrieval. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, Berlin, Heidelberg, 29–41. Springer-Verlag.

(Zaragoza et al., 2007) H. Zaragoza, H. Rode, P. Mika, J. Atserias, M. Ciaramita, et G. Attardi, 2007. Ranking Very Many Typed Entities on Wikipedia. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, New York, NY, USA, 1015–1018. ACM.

(Zhai et Lafferty, 2001) C. Zhai et J. Lafferty, 2001. Model-based Feedback in the Language Modeling Approach to Information Retrieval. In *Proceedings of the Tenth International Conference on Information and Knowledge Management*, CIKM '01, New York, NY, USA, 403–410. ACM.

(Zhai et Lafferty, 2004) C. Zhai et J. Lafferty, 2004. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Transactions on Information Systems* *22*(2), 179–214.

(Zobel, 1998) J. Zobel, 1998. How Reliable Are the Results of Large-scale Information Retrieval Experiments ? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, New York, NY, USA, 307–314. ACM.

Bibliographie personnelle

(Bonnefoy et al., 2009) L. Bonnefoy, R. Deveaud, et E. Charton, 2009. Interrogations de moteurs de recherche par des requêtes formulées en langage naturel. In *Actes de la 8e Manifestation des Jeunes Chercheurs francophones en Sciences et Technologies de l'Information et de la Communication*, MajecSTIC'09.

(Bonnefoy et al., 2012) L. Bonnefoy, R. Deveaud, et P. Bellot, 2012. Do Social Information Help Book Search? In *Focused Access to Content, Structure and Context: 11th International Workshop of the Initiative for the Evaluation of XML Retrieval*, INEX'12.

(Bonnefoy et al., 2013) L. Bonnefoy, V. Bouvier, R. Deveaud, et P. Bellot, 2013. Vers une détection en temps réel de documents Web centrés sur une entité donnée. In *Actes de la 10e COnférence en Recherche d'Information et Applications*, CORIA'13, 21-36.

(Deveaud et al., 2010) R. Deveaud, F. Boudin, et P. Bellot, 2010. LIA at INEX 2010 Book Track. In *Comparative Evaluation of Focused Retrieval - 9th International Workshop of the Inititative for the Evaluation of XML Retrieval*, INEX'10, 118-127.

(Deveaud et al., 2011a) R. Deveaud, F. Boudin, E. SanJuan, et P. Bellot, 2011a. Correction de césures et enrichissement de requêtes pour la recherche de livres. In *Actes de la 8e COnférence en Recherche d'Information et Applications*, CORIA'11, 89-96.

(Deveaud et al., 2011b) R. Deveaud, E. SanJuan, et P. Bellot, 2011b. Ajout d'informations contextuelles issues de Wikipédia pour la recherche de passages. In *Actes de la 18e conférence sur le Traitement Automatique des Langues Naturelles*, TALN'11.

(Deveaud et al., 2011c) R. Deveaud, E. SanJuan, et P. Bellot, 2011c. LIA at TREC 2011 Web Track: Experiments on the Combination of Online Resources. In *Proceedings of The Twentieth Text REtrieval Conference*, TREC'11.

(Deveaud et al., 2011d) R. Deveaud, E. SanJuan, et P. Bellot, 2011d. Social Recommendation and External Resources for Book Search. In *Focused Retrieval of Content and Structure - 10th International Workshop of the Inititative for the Evaluation of XML Retrieval*, INEX'11, 68-79.

(Deveaud et Bellot, 2012) R. Deveaud et P. Bellot, 2012. Combinaison de ressource générales pour une contextualisation implicite de requêtes. In *Actes de la 19e conférence sur le Traitement Automatique des Langues Naturelles*, TALN'12, 479-486.

(Deveaud et Boudin, 2012) R. Deveaud et F. Boudin, 2012. LIA/LINA at the INEX 2012 Tweet Contextualization track. In *Focused Access to Content, Structure and Context: 11th International Workshop of the Initiative for the Evaluation of XML Retrieval*, INEX'12.

(Deveaud et al., 2012) R. Deveaud, E. SanJuan, et P. Bellot, 2012. LIA at TREC 2012 Web Track : Unsupervised Search Concepts Identification from General Sources of Information. In *Proceedings of The Twenty-first Text REtrieval Conference*, TREC'12.

(Deveaud et al., 2013) R. Deveaud, L. Bonnefoy, et P. Bellot, 2013. Quantification et identification des concepts implicites d'une requête. In *Actes de la 10e COnférence en Recherche d'Information et Applications*, CORIA'13, 159-174.

(Deveaud et Boudin, 2013a) R. Deveaud et F. Boudin, 2013a. Contextualisation automatique de Tweets à partir de Wikipédia. In *Actes de la 10e COnférence en Recherche d'Information et Applications*, CORIA'13, 125-140.

(Deveaud et Boudin, 2013b) R. Deveaud et F. Boudin, 2013b. Effective Tweet Contextualization with Hashtags Performance Prediction and Multi-Document Summarization. In *Proceedings of the CLEF 2013 Workshop of the Initiative for the Evaluation of XML Retrieval*, INEX'13.

(Deveaud et SanJuan, 2013) R. Deveaud et E. SanJuan, 2013. LIA at the NTCIR-10 INTENT Task. In *Proceedings of The Tenth NTCIR Conference*, NTCIR-10.

(Deveaud et al., 2013a) R. Deveaud, E. SanJuan, et P. Bellot, 2013a. Are Semantically Coherent Topic Models Useful for Information Retrieval? In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL'13, 148-152.

(Deveaud et al., 2013b) R. Deveaud, E. SanJuan, et P. Bellot, 2013b. Estimating Topical Context by Diverging from External Resources. In *Proceedings of the 36th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'13, 1001-1004. ACM.

(Deveaud et al., 2013c) R. Deveaud, E. SanJuan, et P. Bellot, 2013c. Unsupervised Latent Concept Modeling to Identify Query Facets. In *Proceedings of the Tenth International Conference in the RIAO series*, OAIR'13, 93-100. CID.

(Jourlin et al., 2012) P. Jourlin, R. Deveaud, E. SanJuan-Ibekwe, J.-M. Francony, et F. Para, 2012. Design, implementation and experiment of a YeSQL Web Crawler. In *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, OSIR'12.