Reformulation sémantique des requêtes pour la recherche d'information ad hoc sur le Web

Bissan Audeh

NNT : 2014 EMSE 0750

T H È S E

Présentée par

Bissan AUDEH

pour obtenir le grade de

Docteur de l'École Nationale Supérieure des Mines de Saint-Étienne

Spécialité : Informatique

Reformulation sémantique des requêtes pour la recherche d'information ad hoc sur le Web

Soutenue à Saint-Étienne le 09 Septembre 2014

Président :	Eric Gaussier	Professeur, Université Joseph Fourier, Grenoble
Rapporteurs :	Patrice Bellot	Professeur, Aix-Marseille Université CNRS (LSIS), Marseille
	Jacques Savoy	Professeur, Université de Neuchâtel, Suisse
Examinatrice :	Sylvie Calabretto	Professeur, INSA (LIRIS), Lyon
Directeur de thèse :	Olivier Boissier	Professeur, École des Mines de Saint-Étienne
Co-encadrants :	Philippe Beaune	Maître-assistant, École des Mines de Saint-Étienne
	Michel Beigbeder	Maître-assistant, École des Mines de Saint-Étienne

Spécialités doctorales	Responsables :	Spécialités doctorales	Responsables
SCIENCES ET GENIE DES MATERIAUX	K. Wolski Directeur de recherche	MATHEMATIQUES APPLIQUEES	O. Roustant, Maître-assistant
MECANIQUE ET INGENIERIE	S. Drapier, professeur	INFORMATIQUE	O. Boissier, Professeur
GENIE DES PROCEDES	F. Gruy, Maître de recherche	IMAGE, VISION, SIGNAL	JC. Pinoli, Professeur
SCIENCES DE LA TERRE	B. Guy, Directeur de recherche	GENIE INDUSTRIEL	A. Dolgui, Professeur
SCIENCES ET GENIE DE L'ENVIRONNEMENT	D. Graillot, Directeur de recherche	MICROELECTRONIQUE	S. Dauzere Peres, Professeur

EMSE : Enseignants-chercheurs et chercheurs autorisés à diriger des thèses de doctorat (titulaires d'un doctorat d'État ou d'une HDR)

ABSI	Nabil	CR		CMP
AVRIL	Stéphane	PR2	Mécanique et ingénierie	CIS
BALBO	Flavien	PR2		FAYOL
BASSEREAU	Jean-François	PR		SMS
BATTON-HUBERT	Mireille	PR2	Sciences et génie de l'environnement	FAYOL
BERGER DOUCE	Sandrine	PR2		FAYOL
BERNACHE-ASSOLLANT	Didier	PR0	Génie des Procédés	CIS
BIGOT	Jean Pierre	MR(DR2)	Génie des Procédés	SPIN
BILAL	Essaid	DR	Sciences de la Terre	SPIN
BOISSIER	Olivier	PR1	Informatique	FAYOL
BORBELY	Andras	MR(DR2)	Sciences et génie des matériaux	SMS
BOUCHER	Xavier	PR2	Génie Industriel	FAYOL
BRODHAG	Christian	DR	Sciences et génie de l'environnement	FAYOL
BRUCHON	Julien	MA(MDC)	Mécanique et ingénierie	SMS
BURLAT	Patrick	PR2	Génie Industriel	FAYOL
COURNIL	Michel	PR0	Génie des Procédés	DIR
DARRIEULAT	Michel	IGM	Sciences et génie des matériaux	SMS
DAUZERE-PERES	Stéphane	PR1	Génie Industriel	CMP
DEBAYLE	Johan	CR	Image Vision Signal	CIS
DELAFOSSE	David	PR1	Sciences et génie des matériaux	SMS
DESRAYAUD	Christophe	PR2	Mécanique et ingénierie	SMS
DOLGUI	Alexandre	PR0	Génie Industriel	FAYOL
DRAPIER	Sylvain	PR1	Mécanique et ingénierie	SMS
FEILLET	Dominique	PR2	Génie Industriel	CMP
FEVOTTE	Gilles	PR1	Génie des Procédés	SPIN
FRACZKIEWICZ	Anna	DR	Sciences et génie des matériaux	SMS
GARCIA	Daniel	MR(DR2)	Génie des Procédés	SPIN
GERINGER	Jean	MA(MDC)	Sciences et génie des matériaux	CIS
GOEURIOT	Dominique	DR	Sciences et génie des matériaux	SMS
GRAILLOT	Didier	DR	Sciences et génie de l'environnement	SPIN
GROSSEAU	Philippe	DR	Génie des Procédés	SPIN
GRUY	Frédéric	PR1	Génie des Procédés	SPIN
GUY	Bernard	DR	Sciences de la Terre	SPIN
HAN	Woo-Suck	CR	Mécanique et ingénierie	SMS
HERRI	Jean Michel	PR1	Génie des Procédés	SPIN
KERMOUCHE	Guillaume	PR2	Mécanique et Ingénierie	SMS
KLOCKER	Helmut	DR	Sciences et génie des matériaux	SMS
LAFOREST	Valérie	MR(DR2)	Sciences et génie de l'environnement	FAYOL
LERICHE	Rodolphe	CR	Mécanique et ingénierie	FAYOL
LI	Jean-Michel		Microélectronique	CMP
MALLIARAS	Georges	PR1	Microélectronique	CMP
MOLIMARD	Jérôme	PR2	Mécanique et ingénierie	CIS
MONTHEILLET	Frank	DR	Sciences et génie des matériaux	SMS
MOUTTE	Jacques	CR	Génie des Procédés	SPIN
NEUBERT	Gilles			FAYOL
NIKOLOVSKI	Jean-Pierre			CMP
NORTIER	Patrice	PR1		SPIN
PIJOLAT	Christophe	PR0	Génie des Procédés	SPIN
PIJOLAT	Michèle	PR1	Génie des Procédés	SPIN
PINOLI	Jean Charles	PR0	Image Vision Signal	CIS
POURCHEZ	Jérémy	CR	Génie des Procédés	CIS
ROBISSON	Bruno			CMP
ROUSSY	Agnès	MA(MDC)		CMP
ROUSTANT	Olivier	MA(MDC)		FAYOL
ROUX	Christian	PR		CIS
STOLARZ	Jacques	CR	Sciences et génie des matériaux	SMS
TRIA	Assia	Ingénieur de recherche	Microélectronique	CMP
VALDIVIESO	François	MA(MDC)	Sciences et génie des matériaux	SMS
VIRICELLE	Jean Paul	MR(DR2)	Génie des Procédés	SPIN
WOLSKI	Krzystof	DR	Sciences et génie des matériaux	SMS
XIE	Xiaolan	PR1	Génie industriel	CIS
YUGMA	Gallian	CR	Génie industriel	CMP

ENISE : Enseignants-chercheurs et chercheurs autorisés à diriger des thèses de doctorat (titulaires d'un doctorat d'État ou d'une HDR)

BERGHEAU	Jean-Michel	PU	Mécanique et Ingénierie	ENISE
BERTRAND	Philippe	MCF	Génie des procédés	ENISE
DUBUJET	Philippe	PU	Mécanique et Ingénierie	ENISE
FEULVARCH	Eric	MCF	Mécanique et Ingénierie	ENISE
FORTUNIER	Roland	PR	Sciences et Génie des matériaux	ENISE
GUSSAROV	Andrey	Enseignant contractuel	Génie des procédés	ENISE
HAMDI	Hédi	MCF	Mécanique et Ingénierie	ENISE
LYONNET	Patrick	PU	Mécanique et Ingénierie	ENISE
RECH	Joël	PU	Mécanique et Ingénierie	ENISE
SMUROV	Igor	PU	Mécanique et Ingénierie	ENISE
TOSCANO	Rosario	PU	Mécanique et Ingénierie	ENISE
ZAHOUANI	Hassan	PU	Mécanique et Ingénierie	ENISE

Mise à jour : 28/03/2014

À mon père...

Remerciements

J'ai eu la chance pendant ma thèse d'être entourée par des gens extraordinaires, que ce soit au niveau professionnel ou personnel. Grâce à ces personnes, mes années de thèse m'ont permis d'apprendre, d'apprécier, et d'aimer le métier de la recherche. Je voudrais commencer par mon directeur de thèse, Olivier BOISSIER, à qui je sais gré de ses précieux conseils, sa gentillesse remarquable et son soutien indéfectible devant tous les défis que j'ai rencontrés pendant ma thèse. Je souhaite exprimer ma profonde reconnaissance à mes deux encadrants, Michel BEIGBEDER et Philippe BEAUNE, pour leur implication efficace dans mon travail. Ma thèse s'est concrétisée grâce à leur expérience, leur haut niveau scientifique et humain et la confiance qu'ils m'ont accordée. Je ne garderai que de bons souvenirs de cette collaboration. Je tiens également à remercier tous les membres de l'équipe ISCOD, car il était très agréable de travailler avec eux dans les différentes occasions. Je n'oublie pas tous les collègues de l'institut Fayol qui m'ont soutenue et généreusement aidée dans toutes les tâches techniques et administratives demandées par ma thèse.

Sur un plan plus personnel, je souhaite remercier tous mes proches en Syrie, particulièrement mon adorable tante Fatmeh, qui a constamment pensé à moi malgré les événements graves que le pays a affrontés ces dernières années. Mes remerciements vont également à tous mes amis qui m'ont aidée à surmonter les moments difficiles par leurs attentions et leur bonne humeur. Très proche de ma famille, je n'aurai jamais assez de mots pour dire ma gratitude envers mes parents et ma meilleure amie, ma sœur Rasha, pour leur amour et leur encouragements qu'ils m'ont accordés dans tous les moments de ma vie. Ils savaient toujours comment me soutenir même quand ils étaient physiquement loin. Je remercies particulièrement ma mère pour les efforts incomparables qu'elle a fait pour nous. Sa présence à côté de moi et de mes enfants m'a permis de franchir les étapes les plus difficiles. Son aide, son amour et ses prières m'ont été indispensables pour finir cette thèse.

Pour finir, mes remerciements les plus tendres et les plus chaleureux vont à mon mari et à mes enfants. Sans le soutient absolu et concret de mon mari, je n'aurais jamais pu finaliser ma thèse dans des bonnes conditions, je le remercie pour sa patience et pour tous les efforts qu'il a faits pour nous pendant cette période délicate. Mes enfants, Aram et Alma, ont toujours su me motiver malgré leur très jeune âge. Je les remercie, car ils ont été très compréhensifs et courageux, même quand je n'étais pas disponible. Devant leurs sourires et leur amour pur et sans limites, je n'avais pas le droit de baisser les bras.

Résumé

Dans un système de recherche documentaire, mettre en correspondance le besoin d'information de l'utilisateur et les documents pertinents est la problématique de base. Un document sémantiquement pertinent pour une requête n'utilise pas forcément les mêmes termes que cette requête pour exprimer les mêmes concepts, cela est d'autant plus vrai sur le Web.
Dans le cadre d'une solution d'expansion et de reformulation de la requête, nous nous intéressons aux différentes façons d'utiliser la sémantique pour mieux exprimer le besoin d'information de l'utilisateur dans un contexte Web. Nous distinguons deux types de concepts : ceux identifiables dans une ressource sémantique comme une ontologie, et ceux que l'on extrait à partir d'un ensemble de documents de pseudo retour de pertinence. Nous proposons une Approche Sémantique Mixte d'Expansion et de Reformulation (ASMER) qui permet de modéliser l'utilisation de ces deux types de concepts dans une requête automatiquement modifiée. Cette approche considère plusieurs défis liés à la modification automatique des requêtes, notamment le choix sélectif des termes d'expansion, le traitement des entités nommées et la reformulation de la requête finale.

Bien que dans un contexte Web la précision soit le critère d'évaluation le plus adapté, nous avons aussi pris en compte le rappel pour étudier le comportement de notre approche sous plusieurs aspects. Ce choix a suscité une autre problématique liée à l'évaluation du rappel en recherche d'information. En constatant que les mesures précédentes ne répondent pas à nos contraintes, nous avons proposé la mesure MOR (Mesure Orientée Rappel), qui permet d'évaluer le rappel en tenant compte de la précision comme importante mais pas prioritaire dans un contexte dirigé rappel.
En incluant MOR dans notre stratégie de test, nous avons évalué ASMER sur quatre collections Web issues des campagnes d'évaluation INEX et TREC. Nos expériences montrent qu'ASMER améliore la performance en précision par rapport aux requêtes originales. Dans la plupart des cas, notre approche apporte une amélioration statistiquement significative par rapport à l'utilisation des requêtes étendues par une méthode de l'état de l'art de l'expansion de la requête. De plus, ASMER retrouve le premier document pertinent bien avant les autres approches, et il est légèrement meilleur en rappel selon la mesure MOR.

Mots clés : Recherche d'information, reformulation sémantique de la requête, retour de pertinence, ressources sémantiques, évaluation du rappel.

Abstract

Matching users information needs and relevant documents has always been a basic problem in information retrieval. A query and a relevant document don't necessarily use the same terms to express the same concepts, this fact is even more visible on the Web. As a query expansion and reformulation solution, we are interested in the different ways the semantic could be used to translate users information need into a query. We define two types of concepts : those which we can identify in a semantic resource like an ontology, and the ones we extract from the collection of documents via pseudo relevance feedback procedure. We propose a semantic and mixed approach to query expansion and reformulation (ASMER) that allows to integrate these two types of concepts in an automatically modified query. Our approach considers many challenges, especially selective terms expansion, named entity treatment and query reformulation.

Even though the precision is the evaluation criteria the most adapted to a web context, we also considered evaluating the recall to study the behavior of our model from different aspects. This choice led us to handle a different problem related to evaluating the recall in information retrieval. After realizing that actual measures don't satisfy our constraints, we proposed a new recall oriented measure (MOR) which considers the recall as a priority without ignoring the precision. Among other measures, MOR was considered to evaluate our approach ASMER on four web collection from the standard evaluation campaigns Inex and Trec. Our experiments showed that ASMER improves the precision of the non modified original queries. In most cases, our approach achieved statistically significant enhancements when compared to a state of the art query expansion method. In addition, ASMER retrieves the first relevant document in better ranks than the compared approaches, it also has slightly better recall according to the measure MOR.

Keywords : Information retrieval, semantic query reformulation, relevance feedback, semantic resources, recall evaluation.

Table des matières

Chapitre 1

Introduction

Sommaire

1.1 Recherche d'information ad hoc sur le Web

Aujourd'hui, interroger une ressource d'information par une requête composée de quelques mots clés n'est plus une pratique réservée aux spécialistes de certains domaines. Cette pratique est devenue un réflexe qui concerne tout individu ayant un accès à Internet. De plus, le développement rapide des outils de communication et des technologies de stockage, ainsi que la simplicité technique de publier une information sur Internet, ont permis la production d'une masse géante d'informations sur le Web pour laquelle on ne dispose souvent pas de garantie de qualité ou de fiabilité. Pour cela, mettre en correspondance un besoin d'information et un document pertinent est encore plus complexe aujourd'hui même si cela a toujours été la problématique de base de la recherche d'information.
Dans cette thèse, nous nous intéressons à la recherche d'information ad hoc dans un contexte Web, où l'utilisateur interroge le système par un ensemble de termes, et reçoit comme résultat une liste ordonnée de documents d'une manière décroissante selon leur pertinence évaluée par le système. Les caractéristiques principales du contexte Web sont la diversité des sujets des documents, la nature des requêtes (courtes, ambiguës) et l'importance de la précision, c'est-à-dire qu'en général, un utilisateur sur le Web préfère voir les documents qui l'intéressent en tête de la liste des résultats plutôt que trouver plus de documents pertinents dans des rangs avancés de la liste.

Notre travail se focalise sur l'utilisation de la sémantique au niveau de la requête, pour améliorer la performance d'un modèle de recherche d'information non conceptuel à la base. C'est-à-dire que le modèle de recherche pour nous est un modèle de recherche qui utilise les termes, et non pas les concepts, pour indexer les documents et pour mettre en correspondance une requête et un document.

1.2 Défi de la modification automatique des requêtes

Un utilisateur peut mal exprimer son besoin en information, surtout s'il n'est pas un spécialiste du domaine auquel les documents pertinents appartiennent, ce qui est assez fréquent dans le contexte Web. Une requête mal exprimée a peu de chance d'avoir des résultats satisfaisants même avec un très bon modèle de recherche. Modifier les requêtes des utilisateurs pour trouver des meilleurs résultats est le but de beaucoup d'approches d'expansion automatique de la requête. La majorité des approches classiques considère la requête comme un sac de mots, auquel on ajoute les termes d'expansion. Avec de telles requêtes, il est difficile pour un humain de prédire le lien entre les termes de la requête originale, les termes ajoutés par l'approche d'expansion, et l'amélioration ou la dégradation de la qualité des résultats causées par l'ajout de ces nouveaux termes.

Notre problématique principale est d'exprimer par une requête textuelle les concepts à la base du besoin d'information de l'utilisateur pour les modèles de recherche d'information basés sur les termes. Le but de cette étude est de prédire l'effet de la qualité (d'un point de vue humain) de la requête sur la performance d'un modèle de recherche d'information. En plaçant cette problématique dans un contexte Web, les défis deviennent plus grands, car il faut faire face aux requêtes courtes et ambiguës, et à un très grand nombre de documents provenant de domaines divers. Nous formalisons ces défis dans les questions de recherche pour lesquelles nous proposons des réponses dans ce document et que nous synthétisons à la fin de cette thèse (cf. chapitre 7).

1.3 Questions de recherche

Depuis les années soixante, beaucoup d'études ont été proposées sur l'expansion et la reformulation de la requête dans le domaine de la recherche d'information. Beaucoup de problématiques ont été évoquées et analysées pour des contextes divers. Aussitôt, même le mot « sémantique » a fait sa place dans le domaine, où il n'est plus une nouveauté d'évoquer la notion de concepts pour l'expansion de la requête. Mais le développement ne s'arrête pas, et on doit faire face à de nouveaux contextes et à de nouvelles techniques. Ainsi, devant ces nombreuses approches, la première question à se poser est de savoir quelles sont les limites et les lacunes des approches existantes, et quels aspects méritent d'être revisités en considérant les nouveaux besoins d'aujourd'hui.

Question 1 : Pourquoi a-t-on besoin d'une nouvelle approche de reformulation de la requête ?

La problématique précise que nous avons proposée dans la section précédente cache plusieurs questions de recherche bien plus larges. Pour découvrir les concepts du besoin d'information de l'utilisateur, une requête courte composée de deux ou

trois mots peut ne pas suffire. L'utilisation d'autres sources de donnée est donc indispensable. Le choix d'une ressource dépend de plusieurs éléments, comme le contexte, le domaine, et la nature des documents. Une fois les ressources choisies, il faut trouver des méthodes adaptées qui permettent de lier une requête avec les concepts qui lui correspondent dans la ressource. Toutes ces étapes font partie de la question de recherche 2.

Question 2 : Comment connaitre les concepts d'une requête courte de l'utilisateur ?

Vu que nous nous fondons sur un modèle de recherche textuel, il faut transformer les concepts en termes afin de pouvoir les exprimer par une requête textuelle. Le fait de faire cette transformation signifie prendre le risque de la dérive de la requête. Pour cela, il est nécessaire d'indiquer au modèle de recherche que les termes d'expansion d'un terme de la requête sont sémantiquement liés car ils font partie du même concept. Par conséquent, cette étape ne peut pas être générique, elle doit prendre en compte le langage de la requête pour pouvoir utiliser les opérateurs de proximité et de pondération spécifique à chaque modèle de recherche. Ces aspects seront traités par la question de recherche 3.

Question 3 : Comment exprimer un concept dans une requête textuelle ?

Une fois que les nouvelles requêtes sont exécutées, l'évaluation des résultats peut prendre place. Pour être crédible, une évaluation doit prendre en compte la comparaison avec les requêtes originales (avant la reformulation) et au moins une autre méthode de reformulation de la requête réputée dans l'état de l'art. De plus, pour qu'une telle comparaison ait du sens, l'évaluation doit pouvoir traiter le choix des paramètres dont les modèles comparés disposent, mesurer la robustesse de l'approche face aux changements de ces paramètres, et utiliser plusieurs mesures d'évaluation pour comprendre le comportement des approches comparées sous plusieurs angles. Ainsi, même si on est dans un contexte Web où l'évaluation doit être dirigée vers la précision, notre volonté de faire une analyse complète de l'approche nous a fait nous interroger sur la notion de rappel et les différents défauts des mesures existantes, ce qui nous a conduits vers une autre problématique de recherche où nous avons formulée la question 4.

Question 4 : Comment évaluer le rappel ?

Dans ce rapport, la question 1 sera analysée dans les chapitres 2 et 3 de l'état de l'art, alors que les questions 2 et 3 font l'objet du chapitre 4 qui présente notre contribution. La question 4 sera évoquée dans le chapitre 5 et servira à l'évaluation de notre contribution principale dans le chapitre 6.

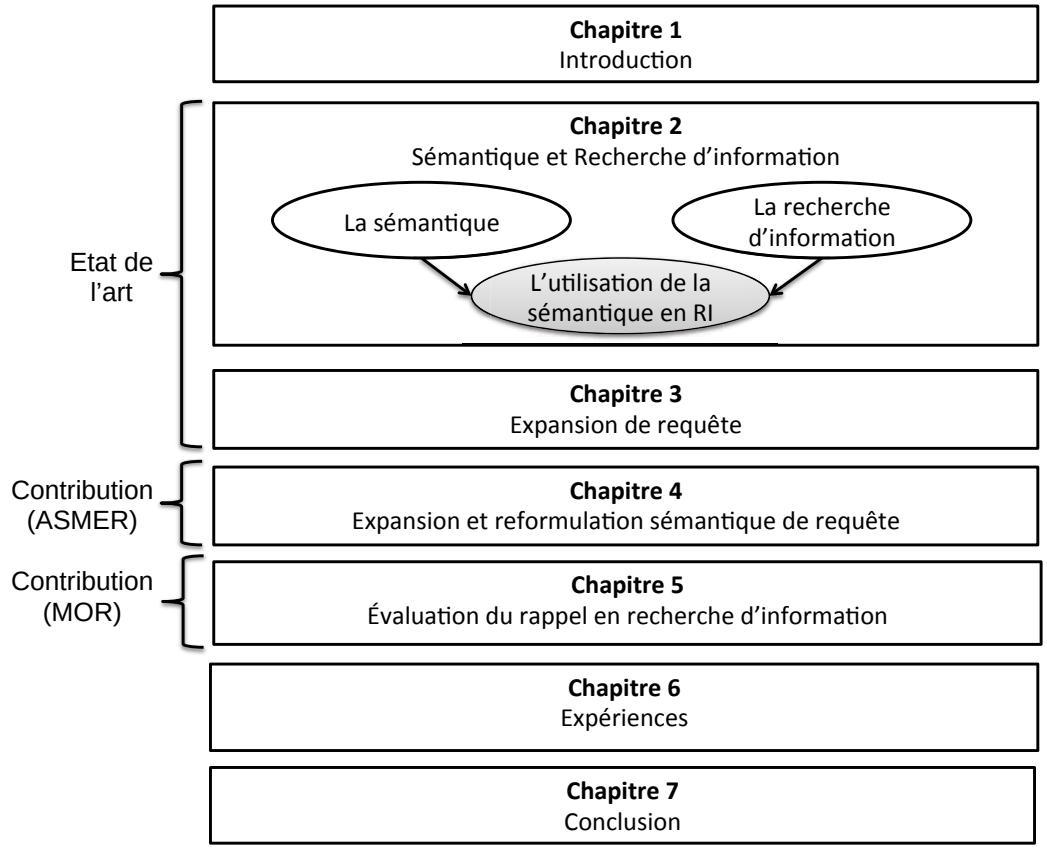


FIGURE 1.1 – La structure de ce mémoire

1.4 Structure du mémoire

Cette thèse s'intéresse principalement à la modélisation sémantique des requêtes Web. La problématique du rappel (question 4) est assez différente de notre sujet principal de thèse, mais elle est essentielle pour la partie expérimentale de notre contribution. Pour cela, la partie qui concerne l'évaluation du rappel (comprenant état de l'art, contribution et expériences) sera présentée dans un seul chapitre qui précède le chapitre expérimental du sujet principal. Ainsi, pour permettre une meilleure lisibilité, la structure de la thèse est présentée par la figure 1.1.

Nous commençons dans le chapitre 2 à présenter la première partie de l'état de l'art, c'est dans ce chapitre que nous faisons le lien entre les deux domaines : la sémantique et la recherche d'information. Le chapitre 3 focalise l'état de l'art sur l'expansion de la requête, où en présentant les questions ouvertes dans ce domaine nous justifions l'intérêt de notre travail. Dans le chapitre 4 nous présentons notre vision sur la modélisation sémantique des requêtes, ce qui conduit à proposer notre Approche Sémantique Mixte d'Expansion et de Reformulation de la requête (ASMER). Avant d'évaluer cette approche, nous évoquons la problématique d'évaluation du rappel, où dans le chapitre 5 nous proposons la mesure MOR (Mesure Orientée Rappel).

Cette nouvelle mesure nous servira, avec d'autres, à faire une analyse de l'approche ASMER de plusieurs points de vue, ce que nous présenterons dans le chapitre d'expériences (Chap. 6). Pour conclure, nous résumons dans le chapitre 7 les idées et les constats principaux présentés dans cette thèse, et nous proposons des réponses aux questions de recherche que nous avons posées dans la section 1.3.

Chapitre 2

Sémantique et recherche d'information

Sommaire

2.1 Objectif du chapitre

Ce chapitre constitue la première partie de l'état de l'art. Il est consacré à une vision générale de deux domaines : la sémantique et la recherche d'information. Le but n'étant pas d'établir un état de l'art exhaustif de chaque aspect, nous essayons de parler principalement des éléments pertinents pour notre étude. L'objectif de ce chapitre consiste donc à bien situer notre travail sur l'expansion et la reformulation sémantique des requêtes, et à clarifier les notions qui nous seront utiles pour la suite

de ce rapport. Nous commençons par définir les notions de sémantique (Sect. 2.2) et du domaine de la recherche d'information (Sect. 2.3). Ces définitions serviront à expliquer les différentes façons d'intégrer la sémantique dans un système de recherche d'information (Sect. 2.4). Nous terminons ce chapitre par un bilan en section 2.5.

2.2 Notion sémantique

La sémantique est un mot d'origine grecque qui signifie l'étude du sens [1]. C'est une notion plutôt philosophique, qui a été employée à propos des systèmes d'information pour mieux rapprocher l'interprétation des choses par des machines avec celle des humains. L'idée de la sémantique est de construire des modèles qui permettent de comprendre, structurer et prédire certaines parties du monde [Hitzler et al., 2009]. L'humanité a commencé tôt à s'intéresser à la modélisation. Nous trouvons de très anciennes études de Platon et d'Aristote (322 av. J.-C.) qui parlent de la classification et de la structuration des objets. Au fil du temps, cette idée a été mieux formalisée, et a commencé à s'intégrer petit à petit dans la littérature de différents domaines. Le livre intitulé « *Semantic information processing* » [Minsky et al., 1968] présente une variété d'applications de la sémantique en recherche d'information et en traitement du langage naturel. L'évolution rapide des technologies, des capacités de stockage et de calcul a fait évoluer ce domaine. On commence à parler de différents types de ressources et de langages sémantiques (Sect. 2.2.1). Dans cette section, nous présentons les parties de la sémantique qui sont en lien avec notre travail, notamment les ressources sémantiques et la désambiguïsation.

2.2.1 Ressources sémantiques

Durant de longues années pendant lesquelles les chercheurs ont essayé d'appliquer la sémantique à divers domaines, plusieurs noms ont été utilisés pour désigner les ressources sémantiques (ontologie, thésaurus, dictionnaire, terminologie, taxonomie, etc.). La signification de chacun de ces mots n'est pas la même selon les études ou les domaines dans lesquels ils sont employés. Il est donc préférable que chaque travail sur la sémantique clarifie précisément son intention dans l'utilisation de ces termes. Dans cette thèse, notre vision des ressources sémantiques est inspirée de la catégorisation des ressources de vocabulaire contrôlé (Fig. 2.1) faite par l'organisation de standardisation [2] ANSI/NISO [2005]. Cette catégorisation place une liste de mots, un cycle de synonymes, une taxonomie et un thésaurus dans un ordre croissant en terme de richesse de types de relations utilisés (complexité). Par contre, cette catégorisation exclut l'ontologie, qui n'est pas considérée comme une ressource de vocabulaire. Nous développons ce classement pour considérer les ressources sémantiques de façon générale (pas uniquement comme un vocabulaire). Cet élargissement (Fig. 2.2) nous permet d'ajouter l'ontologie à la fin de la liste, en tant que ressource la plus riche en relations. Par ailleurs, nous remplaçons le cycle de synonymes par

1. http://www.larousse.fr/
2. ANSI : *American National Standards Institute*.

un dictionnaire, car ce dernier est plus général et peut être utilisé pour créer des cycles de synonymes. De plus, nous ignorons la ressource « liste des mots », car il ne s'agit que d'un ensemble de termes ordonnés selon un certain paramètre (ordre alphabétique, par exemple).

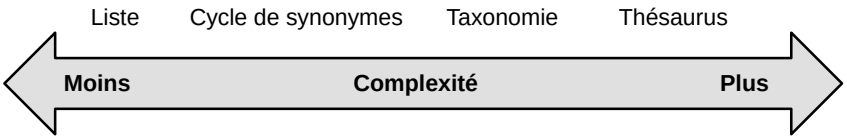


FIGURE 2.1 – Les ressources du vocabulaire contrôlé [ANSI/NISO, 2005]

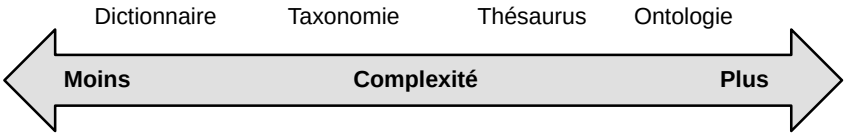


FIGURE 2.2 – Notre vision de la complexité des structures sémantiques

Finalement, nous précisons que l'ordre des ressources dans la figure 2.2, représente également le passage des termes aux concepts. Les éléments de base pour un dictionnaire ou une taxonomie sont les termes, alors que dans un thésaurus ou dans une ontologie nous parlons plutôt de concepts auxquels on associe (éventuellement) des termes. Dans les sections suivantes nous définissons les quatre ressources présentées par la figure 2.2.

2.2.1.1 Un dictionnaire

C'est une ressource qui contient les définitions des termes. Même si de nombreux dictionnaires proposent, en plus de la définition, des synonymes (et éventuellement des antonymes) pour un vocable donné, la structuration de ces ressources y reste peu développée [Bruandet and Chevallet, 2003].

2.2.1.2 Une taxonomie

C'est une structure qui permet de contrôler le vocabulaire par un seul type de relation donnant la possibilité de généraliser ou de préciser un sens. Cette ressource peut être considérée comme une catégorisation pour un domaine précis. La figure 2.3 constitue un exemple d'une taxonomie de bactéries [3] où la relation exprimée est « est un(e) »/« comprend » selon la direction de la relation.

3. Image prise du site de la faculté de médecine de l'université Stellenbosch (http://sumed.stb.sun.ac.za:8001/rid=1HKCPJX2H-2463T4F-3K4/ClassificationofBacteria.cmap).

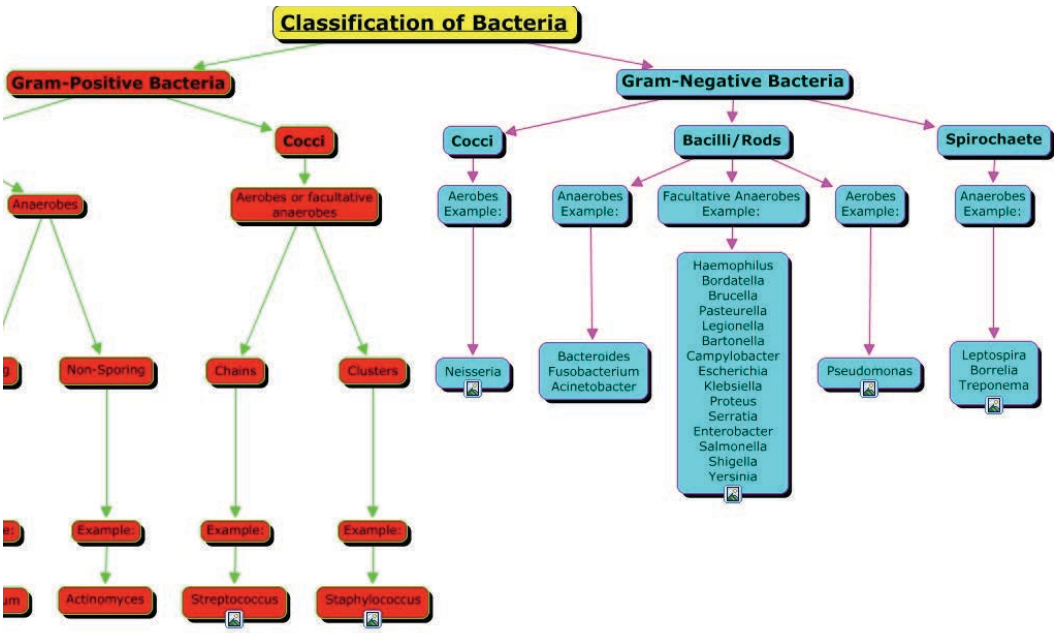


FIGURE 2.3 – Exemple d'une taxonomie de bactéries

2.2.1.3 Un thésaurus

Un thésaurus est une ressource qui représente le vocabulaire par des concepts. Chaque concept a un identifiant, une définition, et contient les termes qui peuvent être employés pour le dénoter. Ainsi, il peut être considéré comme une extension d'un dictionnaire [Bruandet and Chevallet, 2003], car il contient une définition des termes qui appartiennent au même concept, par contre les relations entre ces concepts sont plus riches et plus formalisées par rapport à un dictionnaire. Contrairement à une taxonomie, le thésaurus peut avoir une structure non hiérarchique entre ces éléments. Les relations sémantiques les plus fréquentes dans un thésaurus sont les relations d'équivalence (synonymie/antonymie) et les relations hiérarchiques (hyperonyme/hyponyme) [Kless et al., 2012].

Wordnet [Miller et al., 1990] est l'exemple le plus connu d'un thésaurus, selon notre définition, même si beaucoup d'études le considèrent comme une ontologie (ex. [Wagh and Kolhe, 2011]) ou une taxonomie lexicale (ex. [Jiang and Conrath, 1997]). Dans tous les cas, Wordnet est la ressource lexicale la plus célèbre de la langue anglaise. La figure 2.4 présente une visualisation graphique du mot « book » dans WordNet[4], où les concepts sont appelés « synsets ». Chaque synset contient des termes qui possèdent des relations de synonymie entre eux. Les concepts sont liés par d'autres types de relations sémantiques (antonymie, hyperonymie, etc.). Un mot peut appartenir à un concept ou à plusieurs s'il est polysémique (Sect. 2.2.2).

4. La figure est une capture d'écran de l'outil `http://www.snappywords.com/` qui interroge WordNet et visualise le résultats de façon graphique.

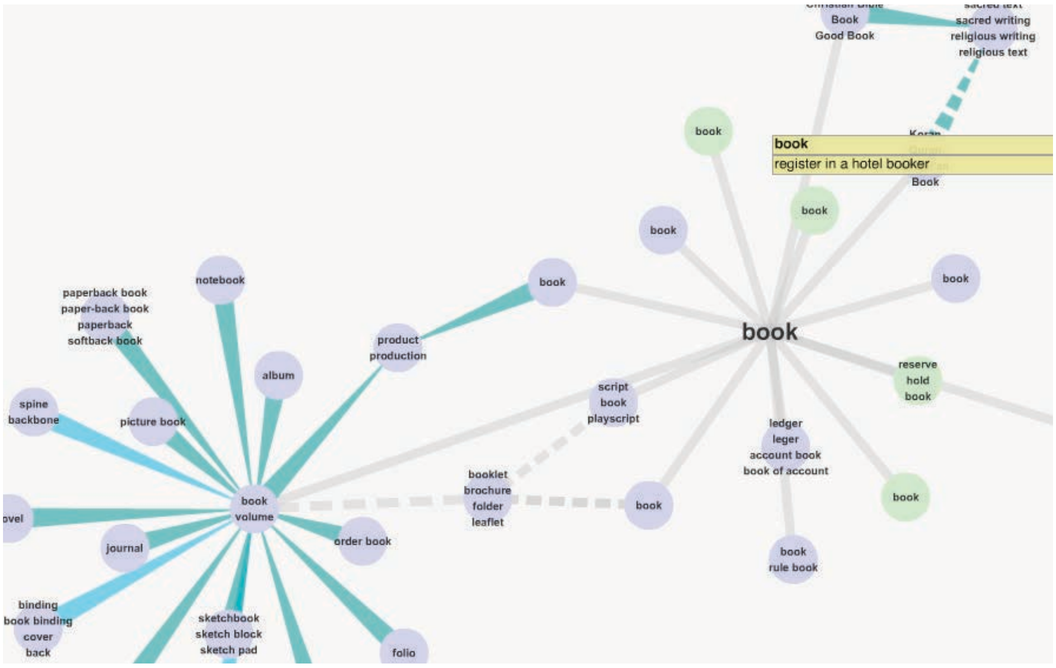


FIGURE 2.4 – Le mot « book » dans le thésaurus anglais WordNet

2.2.1.4 Une ontologie

Une ontologie est une collection de concepts bien définis qui décrivent un domaine spécifique [Van Rees, 2003]. En informatique, cette ressource peut être compréhensible par une machine, les éléments y sont définis de façon formelle [Hitzler et al., 2009]. Les relations entre ces concepts peuvent être différentes d'une ontologie à une autre. Par exemple, les relations existantes dans une ontologie du domaine juridique n'ont pas les mêmes significations que celles d'une ontologie de la génétique. Chaque concept dans une ontologie peut avoir une définition et des propriétés qui le caractérisent. Contrairement à un thésaurus, nous pouvons y trouver des concepts abstraits, c'est-à-dire qui ne sont pas exprimable par un terme du langage humain, mais qui peuvent être compréhensibles par leurs liens avec d'autres concepts. Compte tenu de la variété des relations qui peut être exprimée dans une ontologie, la structure hiérarchique telle qu'on la trouve dans une taxonomie n'est pas forcément garantie.

Avec le niveau élevé de modélisation fourni par les ontologies, des langages de représentation et de spécification ont été proposés pour simplifier la manipulation de ces ressources. Les langages les plus connus sont issus du W3C [5], comme RDF (*Resource Description Framework*), RDF Schema, OWL (*Ontologie Web Language*) et SPARQL.

Alors que les ontologies sont normalement définies pour un domaine spécifique, certaines peuvent être considérées comme générales. Même si ce dernier cas reste

5. http://www.w3.org/

une exception, nous nous concentrons tout de même sur ce type d'ontologie, car c'est le type le plus adapté aux collections de documents divers comme dans le Web, ce qui correspond à notre contexte. La figure 2.5 représente l'ontologie générique Yago [Suchanek et al., 2007] qui est la combinaison de Wordnet et de données extraites de Wikipédia. Chaque élément dans Yago est appelé une entité qui peut être

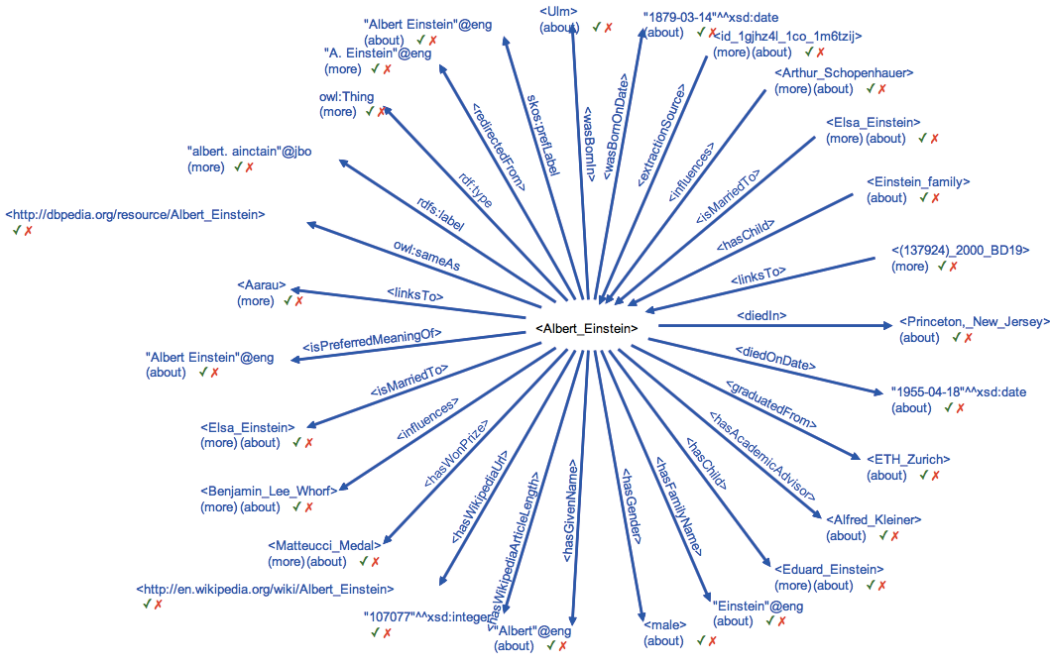


FIGURE 2.5 – Un extrait de l'ontologie YAGO (https://gate.d5.mpi-inf.mpg.de/webyagospotlx/SvgBrowser)

un concept, une relation ou un fait (concept-relation-concept). Un concept dans cette ressource est une classe qui regroupe plusieurs entités ou individus (instances). En règle générale, les classes dans Yago sont les synsets de Wordnet et les individus sont les titres d'articles de Wikipédia. Pour faire les liens entre ces deux ressources, les auteurs ont défini plusieurs règles. Par exemple, les catégories d'une page Wikipédia sont analysées (linguistiquement) pour extraire les concepts correspondants dans WordNet, ce qui génère la relation « type » entre une entité de wikipédia et un synset. Un autre exemple est la relation « rdfs :label », qui existe pour toutes les entités. Cette relation fait référence aux synonymes des synsets correspondants dans WordNet si l'entité est une classe, et représente le lien « *redirect*[6] » si cette entité est un individu.

6. Le lien « *redirect* » est utilisé dans Wikipedia pour diriger l'utilisateur vers la bonne page s'il a utilisé une appellation différente (mais acceptable) de celui du titre de cette page.

2.2.1.5 Comparaison

En nous projetant dans la pyramide des connaissances [Ackoff, 1989] (Fig. 2.6), nous pouvons imaginer qu'une liste de termes est située au niveau des données, car avec une telle liste, aucune information ne peut être extraite. Les dictionnaires et les taxonomies se situent plutôt sur la strate des informations. Ces ressources contiennent des informations concernant une définition ou une certaine hiérarchie entre les termes qui permettent d'obtenir de l'information, en revanche, ces relations ne sont pas assez riches pour autoriser d'en extraire de la connaissance. Cette catégorisation nous amène à placer les ontologies et les thésaurus dans la couche de la connaissance, car avec ces ressources on lie les concepts, et non plus les termes, par une variété de relations sémantiques. Ces relations permettent d'extraire et de calculer des faits, ce qui peut éventuellement produire de la connaissance.

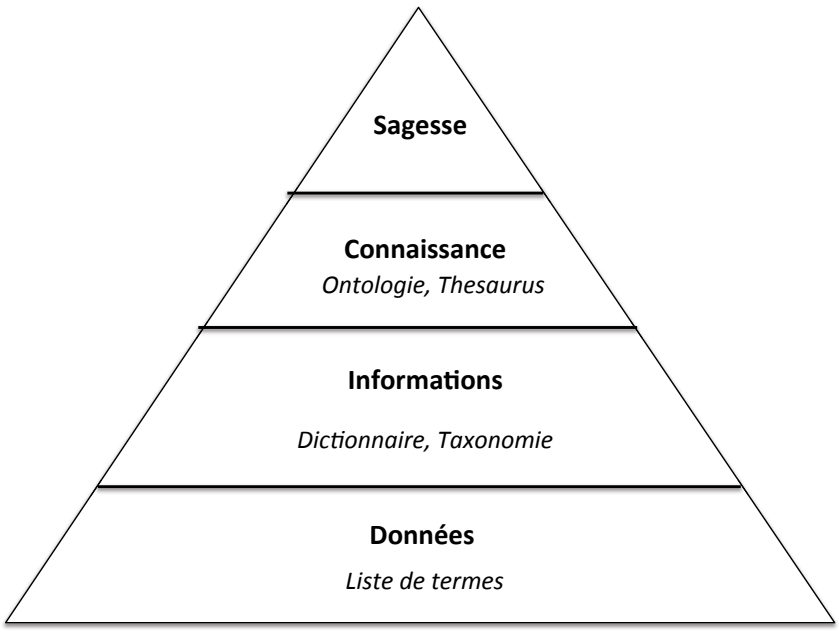


FIGURE 2.6 – Les ressources sémantiques par rapport à la pyramide des connaissances

Nous pensons que ces ressources ne sont pas en concurrence. C'est-à-dire que le niveau élevé de sémantique que l'ontologie apporte ne signifie pas forcément qu'elle est la ressource idéale pour toutes les applications.

2.2.2 Désambiguïsation

Les termes et leurs significations sont liés par une relation de type plusieurs-plusieurs (*many to many*) : un terme peut posséder plusieurs sens (polysémie) et un sens peut être exprimé par plusieurs mots (synonymie). La désambiguïsation est le processus servant à prendre en compte la polysémie, en choisissant le sens de

l'occurrence d'un terme. Ce processus est un problème complexe, largement abordé en traitement du langage naturel. Les méthodes de désambiguïsation peuvent être divisées en trois groupes [Navigli, 2009] : basées sur l'apprentissage supervisé, sur l'apprentissage non supervisé, ou des bases de connaissances. Nous nous intéressons à cette dernière catégorie, car dans notre contribution il est important pour l'étape de l'expansion de lier les termes à une ressource sémantique. L'utilisation des ressources sémantiques pour la désambiguïsation d'un terme est basée sur l'idée de calculer la similarité (is-a relation) ou la proximité sémantique (grâce à d'autres types de relations) entre ce terme et les autres vocables dans son voisinage (le contexte). Les écrits de Pedersen et al. [2007] contiennent une comparaison très compréhensible des différentes méthodes pour mesurer la similarité et la proximité sémantique (Fig. 2.7).

Type	Name	Principle	Pro's	Con's
Path Finding	Path Length	Count of edges between concepts	- Simplicity	- Requires a rich and consistent hierarchy; - no multiple inheritance - WordNet nouns only - IS-A relations only
	Wu & Palmer	Path length to subsumer, scaled by subsumers path to root	- Simplicity	- WordNet nouns only - IS-A relations only
	Leacock & Chodorow	Finds the shortest path between concepts, and log smoothing	- Simplicity - Corrects for depth of hierarchy	- WordNet nouns only - IS-A relations only
	Hirst & St-Onge	Relies on synsets in WordNet	- Measures relatedness of all parts of speech - more than IS-A relations	- WordNet specific - Relies on synsets and relations not available in UMLS
Info. Content	Resnik	Information Content (IC) of the least common subsumer (LCS)	- Uses empirical information from corpora	- Does not use the IC of individual concepts, only that of the LCS - WordNet nouns only - IS-A relations only
	Jiang & Conrath; Lin	Extensions of Resnik; scale LCS by IC of concepts	-Accounts for the IC of individual concepts, only that of the LCS	- WordNet nouns only - IS-A relations only
Context Vector Measures	Patwardhan & Pedersen	Creates context vectors that represent the meaning of concepts derived from co-occurrence statistics of corpora	- Measures relatedness of all parts of speech - No underlying structure required - Uses empirical knowledge implicit in a corpus of data	- Definitions can be short, inconsistent - Computationally intensive

FIGURE 2.7 – Des méthodes de similarité et proximité sémantique Pedersen et al. [2007]

Dans une ressource qui associe à chaque terme l'ensemble des concepts qu'il peut désigner, une méthode possible pour désambiguïser un terme est de calculer la valeur de similarité (ou la proximité sémantique) entre un sens candidat de ce terme et les sens possibles de son voisinage. Une fois ce calcul fait, on peut considérer que le « bon sens » est celui qui maximise cette valeur de similarité/proximité avec le contexte [Navigli, 2009].

2.3 Recherche d'information

La tâche principale d'un système de recherche d'information (SRI) est de sélectionner dans une collection de documents ceux susceptibles de répondre aux besoins en information de l'utilisateur [Boughanem and Savoy, 2008]. La figure 2.8 illustre les étapes classiques d'un processus de recherche d'information dite ad hoc, dans lequel se situe notre travail. Les éléments principaux dans cette opération sont les documents, les utilisateurs et le modèle de recherche au travers duquel l'utilisateur cherche à recevoir un ensemble de documents répondant à son besoin d'information. Les documents et les requêtes sont exprimés, en général, dans le langage naturel. L'interface du SRI permet de saisir les requêtes, ces dernières sont éventuellement racinisées par la même méthode utilisée lors de l'indexation de documents, afin de pouvoir mettre en correspondance le fonds documentaire d'une part, et les besoins d'information exprimés par les requêtes d'autre part. Afin de rapprocher au mieux la pertinence du point de vue de l'utilisateur et de celui du système, une étape de reformulation de la requête (manuelle ou automatique) est souvent utilisée.

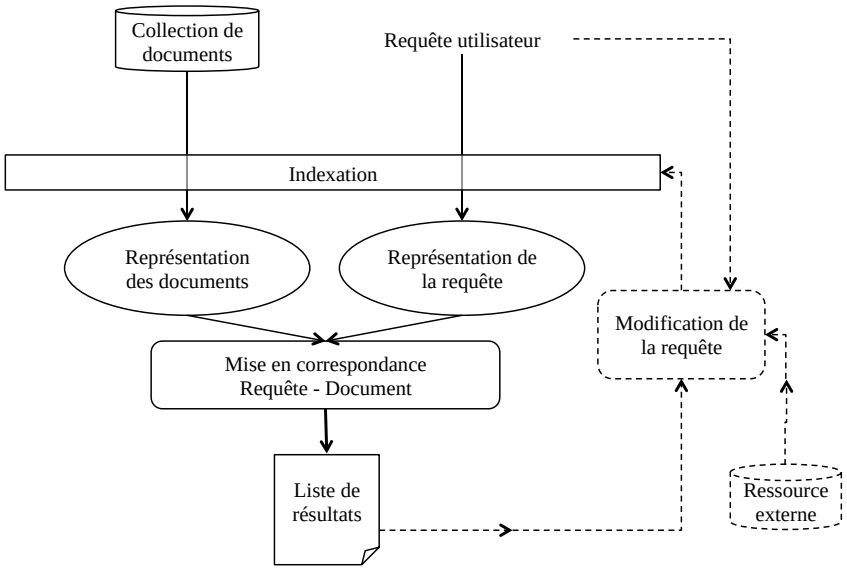


FIGURE 2.8 – Le système de recherche d'information ad hoc

Le terme « Modèle de Recherche d'information » (MRI) est utilisé dans ce rapport pour décrire une approche d'indexation et de mise en correspondance. Un « Système de Recherche d'information » (SRI) est le terme plus global contenant tous les éléments d'un processus de recherche d'information, c'est-à-dire, le MRI et son implémentation, l'interface d'entrée et de sortie, les utilisateurs, les documents et éventuellement une étape de reformulation de la requête.

2.3.1 Du besoin en information à la pertinence

Le besoin en information et la pertinence sont les notions subjectives qui constituent le début et la fin d'un processus de recherche d'information. Deux personnes peuvent exprimer le même besoin d'information par deux requêtes différentes, et ne pas estimer de la même façon la pertinence des documents rendus en réponse à leur demande. Ce fait constitue la difficulté la plus importante pour l'interprétation des requêtes et pour l'évaluation des systèmes dans le domaine de la recherche d'information.

2.3.1.1 Besoin d'information *vs.* requête

Dans la recherche d'information ad hoc, un besoin d'information est exprimé par des mots afin de pouvoir être transmis à un système de recherche d'information. En général, une requête se compose d'une liste de mots-clés. Ces mots-clés peuvent éventuellement être reliés entre eux par des opérateurs définis par un langage adapté au modèle de recherche. Beaucoup de SRI acceptent la pondération de termes, et intègrent la notion de proximité pour s'intéresser au voisinage des termes dans un document. Du côté de l'utilisateur, nous pouvons diviser les requêtes en trois groupes, selon la nature du besoin d'information exprimé par les requêtes :

- Les requêtes ciblées : l'utilisateur cherche une information précise, en général il a une idée bien claire de ce qu'il souhaite obtenir comme résultat. Par exemple, « connaître le nombre d'habitants de Damas en 2007 ».
- Les requêtes exploratoires : dans ce genre de demande, l'utilisateur n'a pas une idée exacte de ce qu'il cherche, mais il connait le contexte dans lequel il souhaite découvrir de nouveaux aspects. Par exemple, la requête large « langage de programmation JAVA » va probablement donner comme résultat des informations sur la syntaxe, l'utilisation, les techniques et d'autres données sur JAVA qui intéressent l'utilisateur, mais il ne pouvait pas les chercher en premier lieu, car il ne les connaissait pas à l'avance.
- Les requêtes par similarité : ici, l'utilisateur a une idée bien définie sur un aspect, et il souhaite avoir le même type d'information sur une autre perspective qu'il ne connaît pas. Par exemple, « trouver les aliments qui ont le même effet excitant que le café ».

2.3.1.2 La pertinence

La pertinence est la correspondance entre un document et une requête, ou encore une mesure d'informativité du document à la requête [Boughanem and Savoy, 2008]. Nous pouvons distinguer deux types de pertinence :

- La pertinence système : elle est déterministe, objective et définie à travers les modèles de RI. Elle est souvent traduite par un score évaluant l'adéquation du contenu des documents vis-à-vis de la requête.
- La pertinence utilisateur : elle est liée à la perception de l'utilisateur sur l'information renvoyée par le système. Elle est subjective et peut évoluer

dans le temps d'une recherche.
Bien que le but de la recherche d'information soit de trouver des documents pertinents au sens de l'utilisateur, il est bien compliqué de mesurer à quel point un SRI va réussir cette tâche.

2.3.2 Méthodes de mise en correspondance

La fonction de correspondance est le moyen utilisé par un MRI pour sélectionner les documents pertinents de la collection pour une requête donnée. La majorité des rapports de recherche d'information cite toujours les modèles dits « classiques » : booléen, vectoriel, probabiliste (dont les modèles de langues). Le nombre de modèles proposés après les classiques est difficile à déterminer. De plus, il est surtout difficile de les comparer, car la motivation derrière chaque modèle et le contexte dans lequel il est évalué sont différents.

Dans le contexte ad hoc où se situe notre travail, nous avons constaté que les modèles de recherche d'information les plus souvent utilisés comme modèles de référence sont le modèle probabiliste *BM25* [Robertson and Walker, 1994] et le modèle de langue par vraisemblance de la requête « *Query Likelyhood* » (*QL*) [Ponte and Croft, 1998]. Le modèle *QL* est l'un des modèles de référence dans nos expériences (Chap. 6). Ce modèle considère que le degré de pertinence d'un document d pour une requête Q est la vraisemblance (probabilité) que le modèle M_d qui a généré ce document soit utilisé pour générer la requête, ce qu'on exprime dans le modèle de langue par $p(Q|M_d)$. Le document dans ce modèle est considéré comme un sac de mot, et M_d est une distribution des mots à partir de laquelle le document a été généré. La probabilité $p(Q|M_d)$ est calculée par l'équation 2.1.

$$p(Q|M_d) = \prod_{q \in Q} p(q|M_d) \tag{2.1}$$

où la méthode la plus fréquente pour calculer $p(q|M_d)$ est de calculer le nombre d'occurrences ($tf_{q,d}$) du terme q dans le document d tout en normalisant par la longueur du document $|d|$. Une explication plus approfondie sur le modèle QL est proposée dans l'annexe A.

2.3.3 Outils de recherche d'information

Plusieurs bibliothèques ont été implémentées pour faciliter le développement des tâches liées à l'indexation et à la mise en correspondance. Ces bibliothèques se présentent souvent comme des systèmes de recherche d'information en proposant une interface pour l'indexation et la recherche. Le modèle de mise en correspondance par défaut dans ces systèmes est souvent une adaptation d'un modèle de recherche classique ou une combinaison de plusieurs idées. Une extension du modèle de vraisemblance de requête, appelée le modèle structuré de langue (annexe. A) constitue le modèle de recherche par défaut du système Indri. Ce système de recherche a été développé par Strohman et al. [2004] au sein du projet Lemur [7]. En plus d'Indri, les

7. http://www.lemurproject.org

systèmes de recherche les plus souvent utilisés dans le domaine de recherche d'information sont Lucene [8] et Terrier [9]. Ces trois systèmes ont été les sujets de plusieurs études de comparaison [Middleton and Baeza-yates, 2007; Armstrong et al., 2008; Turtle et al., 2012]. La majorité de ces études confirme qu'Indri a dépassé Lucene et Terrier en termes de performance (souvent évalués avec le MAP), principalement pour des requêtes courtes. De plus, ce modèle est bien adapté aux études concentrées sur la modification des requêtes, comme notre travail, car il propose un langage flexible et riche. Nous présenterons des exemples des opérateurs de ce modèle dans le chapitre 4.

2.4 Utilisation de la sémantique en recherche d'information

Très tôt dans le domaine de la recherche d'information, il a été clair que la ressemblance syntaxique entre les termes d'une requête et les termes d'un document sur laquelle se base l'évaluation de la pertinence n'était pas suffisante pour répondre au besoin d'information de l'utilisateur. Ce problème a été cité pour la première fois par Swanson [1988] en parlant des études réalisées par John O'Connor sur la relation entre la fréquence des mots et l'indexation de documents, autour des années 1960. Ces études ont montré que parmi 23 documents manuellement indexés sous le mot « toxicity », il n'en existe que 11 contenant le terme « toxicity », alors que plus de la moitié de ces documents réellement pertinents ne contient pas de mots dont l'orthographe est liée au lemme « toxicity ». La littérature de la recherche d'information contient un grand nombre d'études qui ont essayé d'intégrer la sémantique dans un SRI. Egozi et al. [2011] classe ces travaux selon trois paramètres (Fig. 2.9) : la représentation des concepts (explicite/implicite), la technique avec laquelle on transforme les termes en concepts (manuel/automatique) et la phase dans laquelle ces concepts sont utilisés durant le processus de recherche d'information (indexation/requête). Nous ignorons la transformation manuelle des termes en concepts (paramètre 2), car nous nous mettons dans un scénario fréquent en recherche d'information où cette transformation n'est pas faisable manuellement, ni au niveau des documents (des collections de grande taille), ni au niveau de la requête (un utilisateur standard exprime sa requête par des mots clés en langage naturel). Néanmoins, nous sommes conscients qu'il existe des SRI où l'utilisateur choisit les concepts qui correspondent à son texte dans une liste proposée par le système.

Un système de recherche d'information peut impliquer la sémantique au niveau de l'index (Fig. 2.10) ou de la requête (Fig. 2.11). Si la sémantique est intégrée dès la phase d'indexation, toute la procédure de la recherche (la représentation des documents et des requêtes, la mise en correspondance, et éventuellement la reformulation de la requête) sera faite en utilisant des concepts au lieu de mots. Dans le deuxième cas, l'intégration de la sémantique au niveau de la requête, la notion

8. http://lucene.apache.org/core/
9. http://terrier.org

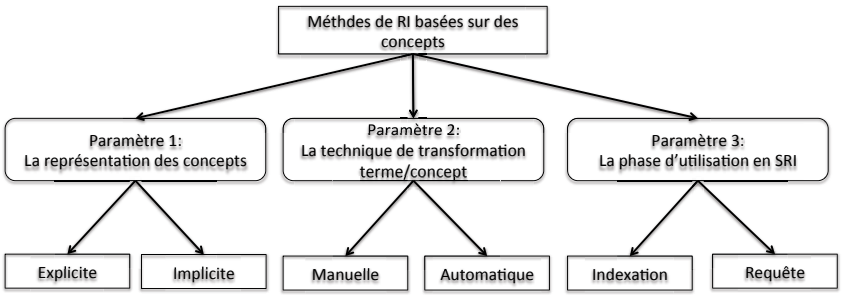


FIGURE 2.9 – Les caractéristiques des modèles basés sur des concepts selon Egozi et al. [2011]

sémantique est utilisée au moment de l'interrogation ; le modèle de recherche par mots-clés n'est donc pas affecté. La couche sémantique dans ce cas sert à identifier les concepts de la requête et à choisir les termes qui les représentent le mieux afin de générer une nouvelle requête textuelle de « meilleure qualité ».

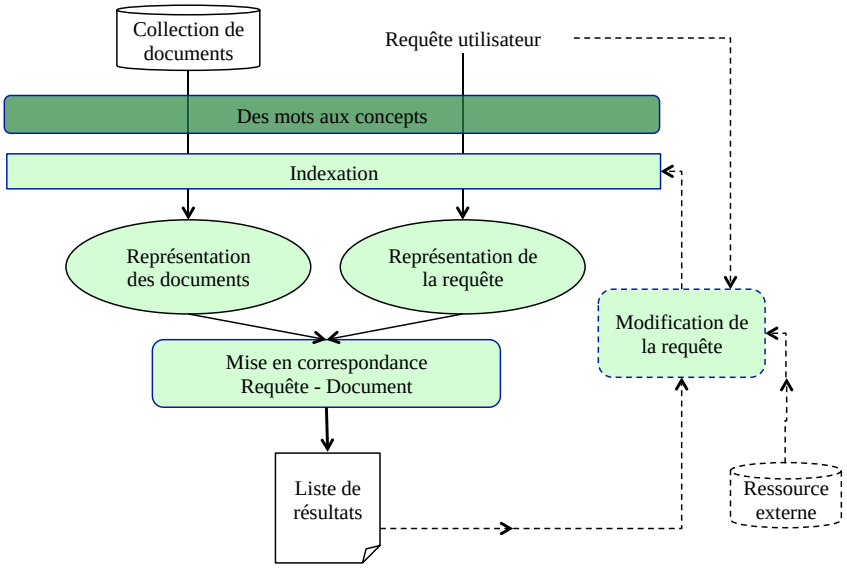


FIGURE 2.10 – L'utilisation de la sémantique au niveau de l'indexation

Les sections suivantes présentent quelques approches utilisées pour intégrer la sémantique dans un SRI.

2.4.1 Sémantique au niveau de l'indexation

Quand la sémantique est utilisée au niveau de l'indexation (Fig. 2.10), nous distinguons l'utilisation des *concepts implicites* découverts par des calculs statistiques sur la collection de documents, ou des *concepts explicites* d'une ressource sémantique indépendante.

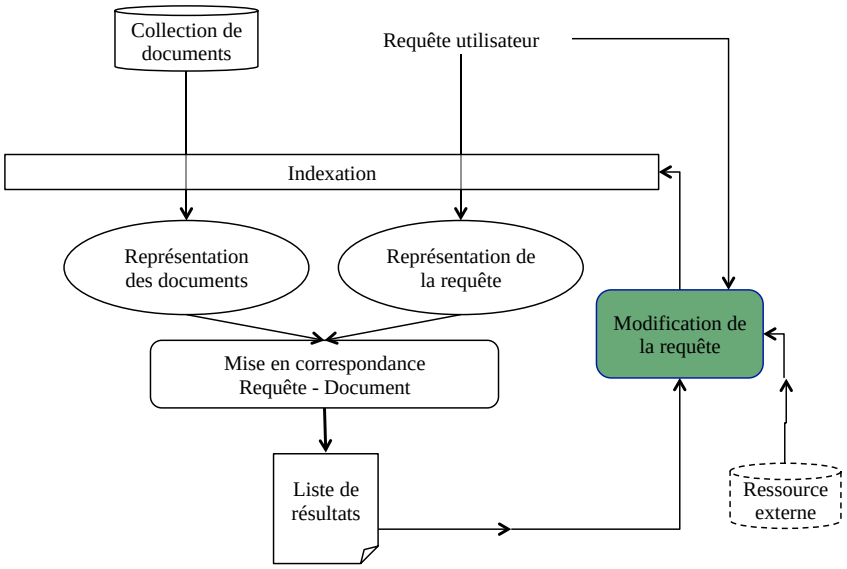


FIGURE 2.11 – L'utilisation de la sémantique au niveau de l'expansion de la requête

2.4.1.1 Indexation par des concepts implicites

L'idée des concepts implicites a commencé avec Koll [1979], qui a représenté les documents et les requêtes dans un espace de concepts, où la mise en correspondance entre une requête et un document a été réalisée par le principe du modèle vectoriel. Dans son système (appelé WEIRD), Koll construisait les concepts en se basant sur les informations de co-occurrence entre les termes dans les documents. Cette idée n'a pas rencontré beaucoup de succès, car les expériences ont été faites sur une petite collection de tests, et le nombre de concepts de base a été choisi manuellement.

Ces défauts ont motivé le travail de Deerwester et al. [1990], qui a proposé l'approche LSI [10] (*Latent Semantic Indexing*). L'idée est basée sur une technique de décomposition des matrices (*Singular Value Decomposition SVD*) pour réduire l'espace des mots dans la collection de documents et parvenir à un espace de « concepts » avec moins de dimensions (le nombre de dimensions conservées, k, est un paramètre de la méthode LSI). Cette transformation de l'espace permet de trouver des liens entre les termes au-delà de la co-occurrence directe : deux termes peuvent être sémantiquement proches même s'ils ne sont jamais apparus dans les mêmes documents. Cette logique a inspiré une partie de notre contribution, pour cela LSI est expliqué avec des exemples démonstratifs en annexe B. Le problème de cette approche est la complexité des calculs de la technique SVD, car une matrice « terme×document » revêt une taille extrêmement grande pour les collections actuelles. La décomposition de ce genre de matrice demande des capacités de calcul importantes. Même si des solutions techniques peuvent exister aujourd'hui pour effectuer cette tâche

10. Dans certains études on appelle cette approche LSA (*Latent Semantic Analysis*), bien que le terme LSA soit plus générique que LSI, nous utilisons l'acronyme LSI dans ce rapport pour être cohérent avec le titre de l'article original [Deerwester et al., 1990].

de façon plus efficiente, la procédure et le coût restent plus élevés par rapport à d'autres techniques. Un autre souci notable avec cette technique est le réglage du nombre de dimensions à garder dans l'espace conceptuel durant le traitement SVD. Plusieurs études ont montré que ce paramètre engendre un effet important sur la performance ; sa valeur idéale varie selon la collection de documents. Kontostathis [2007] a cité certaines de ces études, elle a également réalisé plusieurs expériences en variant le nombre de dimensions sémantiques pour différentes collections. La valeur idéale du paramètre k pour ses expériences a varié entre 75 et 500 selon la collection, en sachant que la performance a été dégradée par rapport au modèle vectoriel pour certaines valeurs de k. D'ailleurs, pour certaines collections, le modèle vectoriel s'est montré meilleur que LSI, quelque soit la valeur de k. La solution selon Kontostathis [2007] a été de combiner les scores de documents obtenus par le modèle vectoriel avec ceux recueillis avec LSI en ne gardant que dix dimensions. Elle a appelé cette méthode EDLSI (*Essential Dimensions of LSI*). Le choix de dix dimensions dans cette approche est arbitraire, et la combinaison avec les scores obtenus avec le modèle vectoriel est discutable, d'autant que les résultats n'ont pas été comparés avec l'utilisation pure d'autres approches connues pour leurs hautes performances.

Un peu après la publication de LSI, Hofmann [1999] a proposé l'approche probabiliste (pLSI). C'est un modèle dans lequel on suppose que la collection de documents a été générée sur la base d'un ensemble de thématiques. Dans ce modèle, le corpus est donc vu comme des couples (terme, identifiant d'un document) regroupés pour former des documents. Le problème principal de pLSI est que le nombre de paramètres du modèle croit linéairement avec la taille de la collection d'apprentissage, et il est difficile d'affecter des paramètres à des nouveaux documents [Gaussier and Yvon, 2011]. Ces défauts de pLSI ont amené à la proposition de l'approche LDA (*Latent Dirichlet Allocation*) Blei et al. [2003] qui a été adaptée par Wei and Croft [2006] pour la tâche de recherche d'information ad hoc. LDA suppose que la génération de chaque document dans la collection a été précédée par le choix du nombre de mots et le choix de la répartition des thèmes de ce document, et que chaque mot a été choisi par la probabilité de son appartenance à ces thèmes. L'idée est donc de prédire, à partir des documents, les thèmes qui ont été utilisés pour générer ces documents. LDA peut être considérée comme une extension de pLSI avec la différence que contrairement à pLSI, LDA utilise un nombre limité de paramètres qui n'a pas forcément un lien direct avec le nombre de documents dans la collection Wei and Croft [2006].

Une autre approche qui a attiré l'attention de la communauté de RI est celle appelée ESA (*Explicit Semantic Analysis*). À la base, cette méthode a été proposée par Gabrilovich, Evgeniy and Markovitch [2007] pour calculer la similarité sémantique entre des mots et des textes. En 2011, Egozi et al. [2011] a adapté ESA aux tâches de recherche d'information. Il faut noter que le mot « explicit » dans le titre met cette approche en opposition avec les modèles constitués avec « latent » précédemment présentés, bien que cela ne change pas son principe basé sur des concepts implicites selon notre définition. La différence majeure entre ESA et les procédures précédentes (LSI, EDLSI, pLSI, LDA) est qu'elle se base sur un autre corpus, typi-

quement le corpus de Wikipédia, au lieu de dépendre du corpus de documents sur lequel la recherche est effectuée. Le principe de cette méthode est de construire un index sur Wikipédia où chaque mot est lié à un vecteur de concepts pondérés et chaque concept correspond au nom d'un article. Avec ce genre d'index, n'importe quel texte (document, requête) peut être transformé en un vecteur de concepts (en choisissant le concept qui a le plus de poids pour chaque terme dans le texte), et la mise en correspondance entre une requête et un document peut se passer dans cet espace de concepts. Le fond de l'approche ESA repose sur l'hypothèse que chaque article de Wikipédia représente un concept unique, ce qui est justifié par les auteurs par l'idée que chaque article se concentre sur un seul concept qu'il aborde en détail. Un regard similaire sur les concepts en tant qu'éléments de Wikipédia est à l'origine de la création de l'ontologie générique YAGO [Suchanek et al., 2007]. Celle-ci n'a pas été prise en compte par Egozi et al. [2011], bien qu'il soit intéressant de comparer l'utilisation de cette ressource avec celle construite par ESA.

Les méthodes précédentes tentent d'extraire des relations sémantiques à partir d'une collection de documents. La notion de concept, correspondant aux concepts générée par ces méthodes, est différente de celle correspondant aux concepts manuellement créés par des experts. L'avantage de ces approches est qu'elles ne nécessitent pas de ressource externe (à l'exception d'ESA), elles sont donc (théoriquement) bien adaptées à la collection de documents qu'on souhaite interroger. Par contre, ces approches demandent une étape d'analyse (comme SVD) qui peut dans certains cas se révéler coûteuse. Une question importante se pose, celle de la gestion d'ajouts ou de modifications de documents qui peut requérir un traitement spécifique pour mettre à jour l'espace des concepts. La même interrogation apparaît concernant l'approche ESA, d'autant plus que cette dernière se base sur la collection dynamique et ouverte Wikipédia.

2.4.1.2 Indexation par des concepts explicites

Quand la ressource sémantique utilisée pour l'indexation est une ontologie selon notre définition (Sect. 2.4, page 18), on parle dans beaucoup de cas d'un contexte de recherche d'information lié à un domaine spécifique (médical, juridique, etc.). Cependant, la recherche d'information liée aux domaines spécifiques n'est pas pertinente avec notre sujet. Nous avons fait le choix de présenter une vision sur l'utilisation des ressources sémantiques génériques (thésaurus général ou ontologie générale) au niveau de l'indexation en recherche d'information.

La première tentative d'utiliser des concepts explicites dans la phase d'indexation et de recherche a eu lieu en 1964, quand Raphael [1964] a proposé un programme pour la recherche d'information sémantique baptisé SIR (*Semantic Information Retrieval system*). SIR est un système de questions-réponses, basé sur la création et l'utilisation d'un modèle formel qui ressemble à ce que l'on appelle maintenant une ontologie. Grâce aux requêtes et aux éléments insérés dans ce système, le modèle de connaissance est enrichi et interrogé si l'entrée est une question.

Le travail dans le monde de l'indexation par des concepts explicites a été marqué

par la naissance du thésaurus Wordnet [Miller et al., 1990] (Sect. 2.2.1.3, page 10), qui est devenu une ressource attirante pour les études en recherche d'information en raison de sa simplicité, sa généralité et sa richesse dans la langue anglaise. Rapidement après sa publication, nous trouvons des écrits qui essayent de l'utiliser pour l'indexation. Voorhees [1993] a tenté de désambiguïser automatiquement un texte en utilisant Wordnet. Les expériences sur un modèle vectoriel ont montré une dégradation de la performance par rapport à un index simple à base de racines des mots. Plusieurs approches entre l'année 1993 et 1996 ont rencontré le même échec en utilisant une ressource sémantique pour indexer un corpus par rapport aux modèles basés sur des mots. Ces études sont citées et bien expliquées par Sanderson [2000]. Plus tard, Woods [1997] a suggéré l'utilisation d'une taxonomie des « *subsumptions* » [11] pour l'indexation. Cette base de connaissances a été construite par des experts avec l'aide de Wordnet. Le travail de Woods [1997] extrait les concepts des documents en se basant sur des analyses lexicales et morphologiques du texte afin de trouver les concepts correspondants dans la ressource sémantique. L'auteur explique le souci issue de l'ambiguïté lors de la transformation d'un texte en concepts, mais il n'a pas traité ce problème automatiquement pour éviter le risque d'une mauvaise désambiguïsation. C'est-à-dire que l'indexation prend en compte tous les sens possibles sans essayer de choisir le « bon sens » selon le contexte. Une utilisation plus exhaustive de Wordnet a été explorée par Gonzalo et al. [1998], qui a essayé d'éviter les problèmes d'ambiguïté connus en pratiquant la désambiguïsation manuellement. Sur une petite collection, les auteurs ont constaté une amélioration par rapport à l'indexation par mots. Peu après, Stokoe et al. [2003] ont créé une base d'information à partir d'une partition du corpus Semcor [12]. Pour chaque mot à sens unique dans cette partition, on extrait les lemmatisations possibles, les mots avec lesquels ils co-occurrent dans le corpus et forment souvent une expression (collocation). Cette base d'information est utilisée pour indexer une collection de documents (Wt10G dans l'expérience de Stokoe et al. [2003]). Si aucune correspondance n'est trouvée, le sens le plus fréquent dans Wordnet est alors employé. Le problème avec cette approche est qu'elle est limitée par la connaissance extraite de Brown1 et de Wordnet, où, par exemple, les entités nommées ne sont pas prises en compte. D'autre part, le modèle de base pour l'évaluation est le simple tf/idf, l'avantage en performance obtenu en comparaison à celui-ci n'est pas significatif compte tenu des capacités des modèles de référence utilisés aujourd'hui.

Des ressources sémantiques génériques autres que Wordnet ont été utilisées pour l'indexation en RI dans des contextes génériques. Nous pouvons par exemple citer le travail de Gauch [2003] basé sur les concepts d'ODP (*Open Directory Project*) dans leur approche de *Key Concepts* [13]. Cette méthode a combiné un modèle de recherche

11. Une subsumption est un raisonnement par lequel on met une idée sous une autre plus générale (`http://fr.wiktionary.org/wiki/subsumption`).

12. Semcor est une collection de 352 documents étiqueté (*tagged*) manuellement par des sens correspondant au synsets de WordNet.

13. A ne pas confondre avec le *Key Concept* de Bendersky and Croft [2008] qui est utilisé pour extraire des expressions à partir des requêtes longues (Chap. 3).

d'information par mots-clés avec un autre à base de concepts extraits de l'ontologie « *Open Directory* ». Le problème avec cette proposition est qu'il faut ajouter les concepts manuellement dans la requête afin de tester l'avantage de l'approche, ce qui pose des questions concernant l'évaluation.

Nous avons présenté un aperçu non exhaustif de certaines approches clés basées sur l'indexation par des ressources sémantiques génériques. La liste des approches est beaucoup plus longue surtout si nous considérons celles fondées sur des ontologies de domaine (médicale, brevet, juridique, etc.) et celles commercialisées sur le Web. Nous estimons que la présentation faite ci-dessus de quelques approches génériques est suffisante pour notre rapport qui est plutôt orienté vers le deuxième type d'intégration de la sémantique : au niveau de la requête.

2.4.2 Sémantique au niveau de la requête

Bien qu'un index sémantique signifie une requête sémantique, il existe des cas où l'on souhaite garder un index par mots-clés, mais en introduisant la sémantique au niveau de la requête. Dans ces occurrences, la requête reformulée est constituée de mots, et le rôle de la sémantique est de bien choisir ces mots. Il faut savoir que la reformulation de la requête a toujours été une pratique naturelle que l'utilisateur peut faire manuellement pour se rapprocher des documents pertinents. Notre cas d'étude concerne l'expansion sémantique des requêtes pour un modèle de recherche basé sur les mots (Fig. 2.11). Le chapitre suivant (Chap. 3) est entièrement consacré à l'état de l'art de ce sujet.

2.5 Bilan et positionnement

Nous avons vu dans ce chapitre les différentes notions dans les domaines de la sémantique et de la recherche d'information. Notre problématique d'expansion et de reformulation sémantique de requêtes est bien liée à ces deux domaines. Comme nous sommes dans un contexte de recherche sur des collections non spécifiques à un domaine, les ressources sémantiques qui nous intéressent sont les thésaurus lexicaux (ex. Wordnet) et les ontologies générales (ex. Yago). Selon notre contexte (Chap. 1), nous travaillons avec des modèles de recherche non conceptuels à la base, où l'indexation est réalisée par termes. Nous employons la sémantique au niveau de la requête à travers l'expansion de la requête, dont l'état de l'art est étudié dans le chapitre 3.

Chapitre 3

Expansion de requête

Sommaire

3.1 Objectif du chapitre

Nous avons parlé dans le chapitre précédent de la nécessité de l'utilisation de la sémantique dans le domaine de la recherche d'information afin de dépasser les limites de la ressemblance lexicale entre les termes d'une requête et ceux des documents. Nous focalisons notre travail sur l'emploi de la sémantique au niveau de l'expansion de la requête dans un cadre de modèles de recherche non conceptuels à la base. Ainsi, ce chapitre propose un état de l'art de l'expansion de la requête, ce qui permet d'introduire et de justifier les questions principales de recherche de

notre travail (exposées dans le chapitre 1). Nous commençons par nous interroger sur l'utilité de modifier une requête originale de l'utilisateur (Sect. 3.2), puis nous présentons dans la section 3.3 notre définition du terme « expansion des requêtes » et notre catégorisation des différentes approches. Selon cette catégorisation, la section 3.4 décrit des méthodes de chaque catégorie. Cette présentation révèle plusieurs interrogations que nous regroupons dans la section 3.5. Par la suite (Sect. 3.6), nous abordons deux aspects importants, mais insuffisamment traités dans les approches d'expansion de la requête. Le lien entre ces aspects, les approches précédentes et nos questions de recherche est introduit dans le bilan de ce chapitre en section 3.7.

3.2 Pourquoi modifie-t-on les requêtes ?

Les requêtes des utilisateurs sont modifiées parce qu'elles expriment le besoin d'information d'une manière pas assez satisfaisante pour un système de recherche d'information. Nous pouvons distinguer deux causes principales à la mauvaise qualité des requêtes des utilisateurs du point de vue du système : le choix des termes et celui des paramètres. Le premier problème est que l'utilisateur n'a pas forcément une idée précise de la collection de documents sur laquelle il exécute sa requête (surtout sur le Web). Il n'a donc pas la capacité de prédire les termes à la fois correspondant à son besoin d'information et apparaissant dans les documents pertinents. Deuxièmement, pour des raisons comme le manque d'expertise ou de temps, un utilisateur standard n'emploie pas les paramètres fournis par le modèle de recherche pour mieux exprimer sa requête (comme les opérateurs de proximité et de pondération) [Jansen et al., 2000], pourtant ces paramètres permettent au système de mieux cibler les documents pertinents. Une approche d'expansion de la requête est donc souvent nécessaire pour que les requêtes des utilisateurs profitent au mieux des capacités fournies par un système de recherche d'information.

3.3 Terminologie

Les méthodes s'intéressant à la modification automatique des requêtes sont nombreuses en recherche d'information. Certaines se focalisent sur l'ajout de nouveaux termes, la pondération de termes existants, ou bien l'extraction des expressions ou des sous-requêtes dans le cas de requêtes longues. Dans la littérature, nous trouvons plusieurs appellations comme l'expansion, l'enrichissement, la reformulation, le raffinement, le retour de pertinence, ou bien les approches d'analyse locales et globales pour la modification automatique de la requête. Ces diverses dénominations sont une source de confusion, car leur signification n'est pas la même d'une étude à l'autre. Par exemple, Zobel [2004] voit que l'expansion de la requête est la procédure qui considère l'utilisation de documents de retour de pertinence pour ajouter des termes à la requête, alors que les méthodes qui utilisent une ressource externe pour choisir ces termes sont appelées des « méthodes basées sur un thésaurus ». Cette vision est à l'opposé de celle de Manning et al. [2008], qui font la différence

entre les méthodes locales de Retour de Pertinence (*RP*) et celles globales qu'ils appellent méthodes d'expansion de la requête. Selon ces auteurs, les procédures de *RP* sont des méthodes locales, car elles sont dépendantes des résultats rendus pour une requête. Par contre, celles reposant sur l'utilisation d'un thésaurus, qu'ils appellent méthodes « d'expansion de la requête », sont globales, car elles sont basées sur un thésaurus généré par toute la collection ou sur une ressource extérieure indépendante. Par ailleurs, dans d'autres écrits, le mot « analyse » est évoqué pour l'expansion de la requête [Xu and Croft, 1996], et l'on parle de méthodes basées sur la collection de documents dans le cas d'une analyse globale, ou sur une partie de documents supposés pertinents dans le cas d'une analyse locale. Dans cette catégorisation, l'utilisation d'une ressource externe est exclus. Plus tard, certaines études catégorisent les méthodes d'un point de vue différent : Grootjen and Weide [2004] distinguent trois classes d'expansion de la requête : extensionnelle qui comprend les approches de *RP* et l'analyse locale, intentionnelle qui se base sur le sens des termes et donc utilise un thésaurus créé à partir de toute la collection ou sur une autre base de connaissance externe, puis un groupe collaboratif où l'on utilise des informations comme les historiques des anciennes recherches faites par l'ensemble des utilisateurs. Dans notre travail, la signification du terme « expansion de la requête » est précisée par la définition 1, qui se distingue de la reformulation de la requête que nous allons présenter plus tard (définition 2, page 41).

Définition 1 *L'expansion de la requête est la procédure qui modifie automatiquement une requête par l'ajout de nouveaux termes (pondérés ou pas) sans considérer une structure spécifique pour intégrer ces nouveaux termes dans la requête.*

Chaque méthode d'expansion de la requête dépend généralement de trois éléments : la ressource utilisée pour chercher les termes candidats pour l'expansion, l'approche adoptée pour choisir, et éventuellement pondérer, les « bons » termes parmi ces candidats, et la façon dont les termes d'expansion sont intégrés dans la requête finale. Nous choisissons de catégoriser l'expansion de la requête selon la provenance des termes d'expansion (Fig. 3.1) : l'expansion basée sur une collection de documents (Sect. 3.4.1), l'utilisation d'une ressource sémantique (Sect. 3.4.2) et l'expansion par des retours implicites, typiquement des historiques de recherche (Sect. 3.4.3).

3.4 Approches d'expansion de la requête

Chaque catégorie exposée dans la figure 3.1 est expliquée dans les sections suivantes en présentant les approches les plus célèbres ou les plus démonstratives de chaque groupe, sans exclure les approches hybrides qui utilisent plusieurs techniques

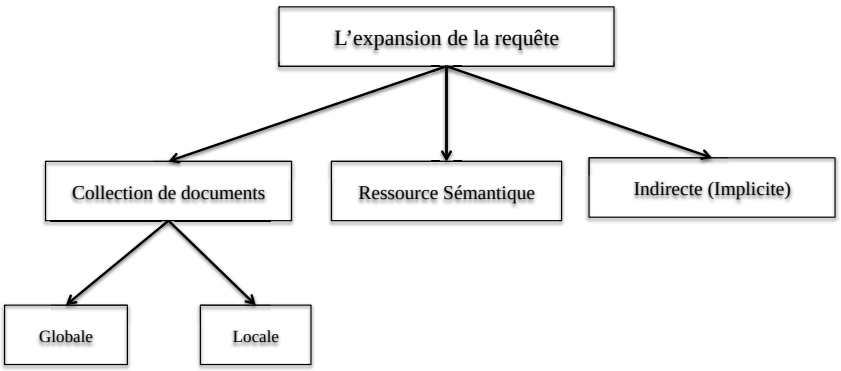


FIGURE 3.1 – La catégorisation des approches d'expansion de la requête

de cette taxonomie. La totalité des méthodes abordées dans ces sections est synthétisée dans le tableau 3.1 (page 37). Par contre, les approches d'expansion qui considèrent également une technique de structuration de la requête sont présentées dans la section sur la reformulation de la requête (Sect. 3.6.1).

3.4.1 Expansion basée sur une collection de documents

Sous cette catégorie se trouvent les approches qui utilisent la collection entière de documents (approches globales) ou un sous-ensemble de cette collection (approches locales) pour mieux exprimer la requête. La motivation de ces méthodes est que les requêtes des utilisateurs n'ont pas suffisamment de contexte pour bien cibler leur besoin d'information. L'utilisation d'un ensemble de documents permet une meilleure estimation du contexte des mots de la requête.

3.4.1.1 Approches globales

Dans ces approches, l'idée est d'utiliser toute la collection de documents pour créer une source d'information qui sera utilisée pour l'expansion de la requête. Ce genre d'approches est souvent évalué avec des collections de tests de taille relativement petite par rapport aux collections actuelles, pour cela, on trouve de moins en moins d'études sur ce genre d'approches, surtout dans un contexte Web. Les premières techniques employées dans ce domaine [Lesk, 1969; Minker et al., 1972] sont basées sur la classification automatique de mots (*term clustering*), où sont ajoutés, pour chaque terme de la requête, tous les termes du même groupe que celui-ci. Ces groupes sont auparavant construits selon les statistiques de co-occurrence de termes dans toute la collection. L'efficacité de cette méthode n'a pas été confirmée par Peat and Willett [1991]. Ces auteurs ont argumenté que les termes qui ont un degré élevé de co-occurrence sont des termes très fréquents dans la collection de documents, et ils sont donc peu discriminants et mauvais pour l'expansion de la requête. Par contre, ces constats ont été fait sur des collections de petite taille, où les auteurs ont utilisé les statistiques des termes et des documents pertinents dans la collection

pour évaluer le pouvoir discriminant des termes. L'effet de la co-occurrence entre les termes sur la performance (en précision et en rappel) d'un modèle de recherche d'information n'a pas été étudié. Plus tard, Qiu and Frei [1993] ont introduit le thésaurus de similarité pour l'expansion des requêtes. Contrairement à la technique de co-occurrence, le thésaurus de similarité est basé sur la façon dont laquelle les termes de la collection sont « indexés » par les documents. Qiu and Frei ont utilisé une méthode probabiliste pour calculer la probabilité qu'un terme soit similaire à une requête dans l'espace du modèle vectoriel. Ils ont constaté que leur méthode a une meilleure performance avec des collections de grande taille (la plus grande collection dans leurs expériences contient moins de douze mille documents) et avec un grand nombre de termes d'expansion (800 termes d'expansion pour la plus grande collection). Ainsi, on peut dire que cette approche n'est pas adaptée à une utilisation sur le Web, surtout que pour beaucoup de modèle de recherche, le temps d'exécution d'une requête est linéaire par rapport au nombre de termes de la requête. De plus, les auteurs déclarent qu'une approche de retour de pertinence a une meilleure performance que leur méthode basée sur toute la collection de document. Jing and Croft [1994] ont proposé la méthode « *phrase finder* », un thésaurus d'association qu'ils génèrent en analysant le texte de chaque paragraphe dans les documents, afin de calculer la fréquence d'occurrence des expressions et des termes. Bien que cette approche ait l'avantage de considérer les expressions, la complexité des calculs est considérable [1], et des questions sur le choix et la pondération de termes et des expressions dans cette approche ne sont pas clairement résolues. Peu d'études sur ce genre d'expansion de la requête sont apparues plus tard, car ce genre de méthode reste très coûteux sur les collections d'aujourd'hui. De plus, sur une collection de documents divers tels que le Web, la similarité sémantique entre les termes calculé sur tous genres de documents confondus, est moins précise que celle qu'on peut calculer sur un sous-ensemble de documents lié au sujet de la requête qu'on souhaite étendre.

3.4.1.2 Approches locales

La façon de choisir les documents dont on extrait l'information d'expansion génère deux techniques possibles : la technique dans laquelle l'utilisateur examine les résultats de sa requête en déclarant au système les documents pertinents et/ou non pertinents, c'est le retour de pertinence, et la technique du retour aveugle de pertinence (ou pseudo retour de pertinence) où l'on suppose que les documents les mieux classés dans la liste de résultats suite à une itération précédente de recherche, sont pertinents. Dans les deux cas, les approches locales d'expansion de la requête nécessitent un ensemble de documents considérés comme pertinents. Comme la majorité d'approches locales, les approches présentées par les paragraphes suivants utilisent la technique de pseudo retour de pertinence.

1. Les auteurs parlent d'un temps de calcul de deux semaines pour une collection de 200 000 documents.

Approches locales en tant qu'extension d'un modèle de recherche Certaines approches locales sont une extension ou une adaptation de la fonction de mise en correspondance. À l'origine, le retour de pertinence a été utilisé avec le modèle vectoriel auquel Rocchio et al. [1965] ont proposé une extension permettant d'étendre la requête originale par les documents de retour de pertinence. L'idée est de rapprocher, dans l'espace des termes, la requête vers les documents pertinents et de l'éloigner des documents non pertinents. Cela est réalisé en combinant le vecteur de la requête avec les vecteurs de documents jugés pertinents. Cette combinaison est basée sur la formule suivante :

$$\overrightarrow{Q}' = \alpha \overrightarrow{Q} + \beta \sum_{\overrightarrow{d} \in R} \frac{\overrightarrow{d}}{|R|} - \gamma \sum_{\overrightarrow{d} \in \bar{R}} \frac{\overrightarrow{d}}{|\bar{R}|} \tag{3.1}$$

Dans la formule 3.1, le vecteur de la nouvelle requête Q' est composée du vecteur de la requête initiale Q auquel on ajoute le centroïde des vecteurs des documents pertinents R et on enlève le centroïde des vecteurs des documents non pertinents $\bar{R}$. Les variables α, β et γ, sont souvent choisies de façon empirique selon l'expérience à réaliser. Grâce à un bon réglage de ces paramètres, Rocchio a constaté une amélioration importante avec l'utilisation de cette méthode.

Dans le cas de modèles probabilistes fondés sur les modèles de langue, les travaux qui influencent le plus les études d'expansion par retour de pertinence sont ceux de Ponte [1998], Lavrenko and Croft [2001] et Zhai and Lafferty [2001]. L'approche de Ponte est une extension naturelle du modèle de langue qu'il a proposé. Dans sa thèse, il suggère une pondération à base du modèle de langue pour les termes qui se trouvent dans les documents de retour de pertinence. Cette pondération est définie dans l'équation 3.2.

$$Score(t) = \sum_{d \in R} \left(\frac{P(t|M_d)}{\frac{tf(t)}{|D|}} \right) \tag{3.2}$$

où R est l'ensemble des documents de retour de pertinence, $tf(t)$ est la fréquence du terme t dans la collection de documents, $|D|$ est la taille de la collection en nombre de tokens et $P(t|M_d)$ est la probabilité du terme t sachant le modèle de document M_d. Cette dernière probabilité est calculée de la même façon que pour les termes d'une requête pour classer les documents dans le modèle de langue de Ponte, sa valeur est la fréquence du terme t dans le document d normalisé par la taille du document d (Sect.2.3.2, page 17). Lavrenko and Croft [2001], de leur côté, se basent sur l'idée de calculer la probabilité qu'un terme t dans un document de retour de pertinence soit généré par le même modèle qui a engendré la requête. Ils proposent deux façons de calculer cette probabilité. La première méthode appelée RM1 [Lv and Zhai, 2009, 2010] suppose une distribution unigramme à partir de laquelle les termes de la requête et du retour de pertinence sont générés : c'est-à-dire que le choix des termes est fait de manière indépendante et identique pour tous les termes. La deuxième méthode (RM2) suppose également une indépendance entre les probabilités des termes de la requête, mais celles-ci sont a priori dépendantes d'une

probabilité du terme t. L'équation 3.3 représente le calcul de la probabilité qu'un terme soit généré par le même modèle qui a généré la requête q selon RM1.

$$p(t|\theta'_Q) \propto \sum_{\theta_d \in \Theta} p(t|\theta_d)p(\theta_d) \prod_{i=1}^{m} p(q_i|\theta_d) \tag{3.3}$$

où Θ est l'ensemble de modèles de documents de retour de pertinence, et m est le nombre de termes dans la requête originale Q. RM1 et RM2 sont souvent utilisés par une interpolation linéaire avec le modèle de vraisemblance de la requête θ_Q comme dans l'article de Lv and Zhai [2009] de la manière suivante :

$$p(t|\theta''_Q) = \alpha p(t|\theta_Q) + (1-\alpha)p(t|\theta'_Q) \tag{3.4}$$

où α est un paramètre libre entre 0 et 1. Certaines études appellent ce modèle RM3 quand $p(t|Q)$ est calculé selon l'équation 3.3, ou RM4 quand la probabilité $p(t|Q)$ suppose une dépendance entre les termes de la requête et le terme t. RM3 est un modèle performant de l'expansion de la requête [Lv and Zhai, 2009], il est utilisé comme modèle de référence par beaucoup d'études qui font l'hypothèse de distribution unigramme de termes [Deveaud et al., 2013; Lv and Zhai, 2010; Zhao and Callan, 2010; Bendersky et al., 2011]. Comme ces études, nous utiliserons RM3 en tant que modèle de référence dans nos expériences. Une autre approche d'expansion par retour de pertinence dans les modèles de langue est celle de Zhai and Lafferty [2001] appelée « *Mixture model* ». Cette approche suppose que chaque terme t dans les documents de retour de pertinence a été généré soit par le modèle qui a généré la requête (*Topic Model*) θ_Q ou par celui qui a généré la collection entière (*Background Model*) θ_c. Ainsi, en considérant R comme la collection de retour de pertinence, et $c(t,d)$ comme le nombre d'occurrences de t dans un document d, la vraisemblance *log-likelyhood* des documents de retour de pertinence peut être calculé par l'équation 3.5.

$$log(p(R|\theta_Q)) = \sum_{d \in R} \sum_{t} c(t,d) log((1-\lambda)p(t|\theta_Q) + \lambda p(t|\theta_c)) \tag{3.5}$$

L'approche cherche à extraire le modèle de la requête qui maximise la valeur présentée par l'équation 3.5. Ce modèle est ensuite utilisé en combinaison avec le modèle de la requête originale pour chercher les documents pertinents.
Une comparaison de ces trois approches de Ponte [1998], Lavrenko and Croft [2001], et Zhai and Lafferty [2001] a été faite par Lv and Zhai [2009]. Ils ont montré qu'au niveau de la précision, le *Mixture Model* et *RM3* ont obtenu les meilleurs résultats sur plusieurs collections de tests. Par contre, le *Mixture Model* a été meilleur au niveau du rappel. Des extensions à ces modèles de base ont été proposées les années suivantes. Une étude intéressante de Lv and Zhai [2010] a proposé la considération des positions des termes dans les documents de retour de pertinence, où en se basant sur la même logique du modèle de pertinence de Lavrenko and Croft [2001] les auteurs prennent également en compte la probabilité qu'un terme fasse partie de la nouvelle requête à chaque position dans chaque document de l'ensemble de retour de pertinence.

Approches locales en tant qu'adaptation d'une technique globale Un autre groupe d'approches a essayé d'appliquer une méthode globale d'expansion de la requête sur l'ensemble de retour de pertinence. L'idée est, d'un côté de profiter de l'efficacité de ces méthodes tout en gagnant un temps de calcul considérable, et de l'autre côté de profiter d'une focalisation plus importante sur le thème de la requête. Sous cette catégorie, nous trouvons, par exemple, le travail de Xu and Croft [1996], qui ont proposé l'utilisation de la technique globale de Jing and Croft [1994], (Sect. 3.4.1.1) sur l'ensemble local des documents de retour de pertinence. Plus récemment, les méthodes d'extraction de concepts à partir de documents comme LSI, ESA, LDA, etc. (Sect. 2.4.1) ont attiré l'attention de chercheurs en expansion de la requête. Bien que l'utilisation de ces méthodes soit à la base dédiée à l'indexation, des études ont essayé de les appliquer sur les documents de retour de pertinence pour étendre la requête. Les exemples de ces travaux sont ceux de He et al. [2009] (recherche d'image) pour LSA, de Luo et al. [2012] pour ESA et de Deveaud et al. [2013] pour LDA. Un travail intéressant de Zhao and Callan [2010] montre l'intérêt de l'utilisation de ce genre de technique (notamment LSA) sur l'ensemble de pseudo retour de pertinence, sauf que les auteurs emploient cette technique parmi d'autres statistiques pour estimer un score pour chaque terme de la requête dans le but de prédire son importance, sans utiliser ces scores pour choisir des termes d'expansion. Xu et al. [2007] utilise LSI sur une matrice de concepts et de documents de retour de pertinence. Ces concepts sont récupérés en faisant le lien entre les termes des documents de retour de pertinence et une ontologie de domaine. Les auteurs constatent une amélioration en utilisant le modèle de recherche BM25 et le modèle vectoriel. Le choix du nombre de dimensions conservées sur les petites matrices générées et la significativité statistique des résultats n'ont pas été précisé dans ce papier. Un travail plus approfondi sur l'utilisation de LSI sur un petit ensemble de documents a été fait par Schütze [1998]. Par contre, les auteurs ne s'intéressent pas à l'expansion de la requête, mais à la classification de termes (*terms clustering*).

3.4.2 Utilisation d'une ressource sémantique

L'utilisation d'une ressource sémantique pour l'expansion de la requête permet d'éviter la complexité élevée des approches basées sur toute la collection de documents et les limites de celles fondées sur le principe de retour de pertinence comme la dépendance de plusieurs paramètres et de la qualité du petit nombre de documents de pseudo retour de pertinence. De plus, la requête étendue avec ces techniques est plus compréhensible par les humains que les méthodes statistiques, car ces dernières peuvent ajouter des mots qui n'ont pas de lien logique (au sens humain) avec les termes de la requête originale.

Les approches proposées dans ce domaine, ont fait l'objet de plusieurs études, comme celles de Bhogal et al. [2007], Wu et al. [2011], et Carpineto and Romano [2012]. Dans le but d'être exhaustives et détaillées, ces études mélangent les approches d'expansion de la requête pour un modèle de recherche textuel avec d'autres basées sur une indexation conceptuelle, ce qui est différent de la direction de notre

étude. Pour éviter la répétition des détails déjà abordés dans les études précédentes, et ne pas éloigner la discussion du contexte Web qui nous intéresse, nous présentons dans cette section uniquement les approches qui utilisent une ressource sémantique générique comme Wordnet ou Yago.

3.4.2.1 Utilisation de Wordnet pour l'expansion de la requête

À notre connaissance, la première étude qui a utilisé Wordnet [Miller, 1995] pour l'expansion de la requête est celle de Voorhees [1994]. Avec l'idée naturelle d'ajouter des synonymes extraits de Wordnet aux termes de la requête, Voorhees a conclu que le succès de cette méthode était lié à deux éléments : la qualité de la requête initiale et celle des synonymes trouvés. Quand la requête initiale exprime déjà correctement le besoin en information, l'ajout de synonymes n'a pas vraiment d'intérêt pour la performance. D'un autre côté, l'ajout automatique (sans désambiguïsation) des synonymes dégrade la performance de la recherche. Par conséquent, Voorhees [1994] a réussi à améliorer les mauvaises requêtes par l'ajout de synonymes choisis manuellement dans Wordnet. Ce constat est similaire à celui de Gonzalo et al. [1998] [2] qui ont, par contre, utilisé Wordnet pour l'indexation de documents, où le choix de synsets de l'index a été fait manuellement.

Le choix manuel des sens par l'utilisateur qui a créé la requête est idéal pour garantir une bonne désambiguïsation. Bien que pertinent dans des environnements de recherche interactive, il est moins supporté par les utilisateurs standard du Web. Navigli and Velardi [2003] ont proposé une étude sur les meilleurs termes d'expansion à choisir une fois la désambiguïsation (le choix du synset dans Wordnet) faite. Alors que cette étude mentionne que l'utilisation de la définition (le glossaire de WordNet) est plus performante pour l'expansion de la requête que les synonymes et les hyperonymes, les résultats ne sont pas concluants à cause des limites des expériences : l'utilisation du moteur de recherche Google n'a pas permis aux auteurs d'évaluer des requêtes de plus de dix termes, et ce choix d'expérience ne dispose pas de listes de jugements comme pour les campagnes d'évaluations, malgré l'utilisation des topics de TREC. Liu et al. [2004] ont proposé une utilisation de Wordnet pour la désambiguïsation et l'expansion, en combinaison avec les informations de retour de pertinence. Les auteurs s'intéressent aux synonymes, hypernymes et définitions des Synsets liés aux termes de la requête. Ce qui est intéressant dans cette étude est qu'ils considèrent l'importance des syntagmes nominaux et des entités nommées (sans explicitement traiter ces dernières). Par contre, cette analyse dépend de l'idée de requêtes booléennes dont l'utilisation avec le modèle BM25 n'est pas clairement établie. Par ailleurs, Shah and Croft [2004] utilisent l'approche de Cronen-Townsend et al. [2002] (cf. section 3.5.2) pour pondérer les termes de la requête par les scores de clarté. Dans cette dernière approche, les termes qui ont un score élevé sont considérés comme précis et ne sont donc pas étendus, ceux avec un score faible sont considérés comme ambigus et ne sont pas étendus non plus. Uniquement les termes de clarté « moyenne » sont étendus par des synonymes de WordNet. La désambi-

2. Ce travail n'est pas cité dans le tableau car il se base sur une indexation conceptuelle

guïsation pour trouver ces synonymes et la méthode pour les ajouter à la requête ne sont pas clairs dans ce papier, de plus les résultats montrent que leur approche n'est pas très efficace sur la précision moyenne mais elle est plus adaptée à un contexte question/réponse. Après ces travaux sur l'utilisation de Wordnet pour l'expansion de la requête, nous trouvons intéressante l'étude de Fang [2008], qui a travaillé sur la pondération de termes d'expansion provenant de Wordnet. Avec des expériences sur six collections de TREC (de taille inférieure à six cent mille documents), Fang a trouvé une amélioration significative lors de l'expansion en utilisant les définitions de synsets. Ce qu'il convient de remarquer est que cette approche a été réalisée à l'aide d'un modèle de recherche *axiomatique*[3] [Fang, 2005], ce qui a permis à l'auteur de pondérer les termes d'expansion d'une façon adaptée au modèle axiomatique, améliorant ainsi les résultats. Un autre travail d'expansion de la requête en utilisant Wordnet a été fait par Zhang et al. [2009]. Les auteurs ont proposé l'ajout simple de tous les synonymes d'un synset auparavant désambiguïsé par leur méthode de désambiguïsation. Bien que les auteurs parlent d'une amélioration de 7 %, nous ne pouvons considérer ces résultats comme valides, car ils ont été obtenus avec un petit nombre de requêtes et d'utilisateurs (5 requête évaluée par 10 utilisateurs) et évalués par une seule mesure (P@10) en utilisant Google comme moteur de recherche.

Cet aperçu de l'état de l'art concernant l'utilisation de Wordnet pour l'expansion de la requête montre que, malgré la simplicité de cette ressource, son utilisation n'est pas évidente sans l'aide d'une autre technique permettant de mieux gérer la désambiguïsation, la pondération des termes et le choix des termes d'expansion. Bien que la plupart de ces problèmes aient été constatés assez tôt dans le domaine, surtout par les travaux de Mandala et al. [1998], Wordnet est resté une ressource attirante pour l'expansion de la requête, car, en tant qu'humains, nous avons l'impression que l'ajout de termes linguistiquement cohérents avec ceux de la requête donnera forcément de meilleurs résultats.

3.4.2.2 Utilisation des ontologies

Il faut bien faire la différence entre la recherche *dans* une ontologie, l'utilisation des ontologies pour l'indexation, et l'emploi des ontologies pour l'expansion des requêtes dans un système de recherche textuel. Plusieurs travaux existent sur l'utilisation des ontologies générales pour la recherche d'information. Hoang and Tjoa [2006] ont réalisé une étude qui présente l'état de l'art de l'utilisation des ontologies en recherche d'information. Leurs travaux se focalisent sur des systèmes de recherche conceptuels à la base, où dans la plupart des cas, l'ontologie utilisée est une ontologie de domaine. Les études concernant l'utilisation des ontologies exclusivement pour l'expansion de la requête sont moins développées et se concentrent uniquement sur l'utilisation de Wordnet, qu'elles considèrent d'ailleurs comme une ontologie.

Bien que plusieurs ontologies générales soient disponibles depuis plusieurs années

3. L'idée d'un modèle de recherche axiomatique est de choisir la fonction d'apparaiment selon des propriétés exprimables mathématiquement qu'elle doit vérifier (axiome), par exemple croissance, convexité, etc.

(comme Kim [4], Tap [Ito et al., 2008], Yago [Suchanek et al., 2007]), leur utilisation pour l'expansion de la requête reste limitée. Nous trouvons par contre des études sur des systèmes de recherche qui combinent la recherche textuelle et conceptuelle sans être liées à un domaine spécifique. Bast et al. [2007] et Pound et al. [2010] se focalisent sur un système de recherche combiné où l'indexation est faite par les concepts de Yago et des mots-clés, l'expansion de la requête n'y est pas abordée. Ngo and Cao [2011] utilisent également Yago parmi d'autres ontologies pour créer une extension conceptuelle du modèle vectoriel et éventuellement étendre les requêtes. Ainsi, ces travaux cherchent à créer un environnement conceptuel pour l'indexation et la recherche à l'aide d'une ontologie globale. Ce qui nous rapproche de ces travaux est la richesse de Yago en entités nommées, un aspect souvent manquant pour les approches d'expansion de la requête. Par contre, notre problématique est bien différente car nous ne nous intéressons pas à un modèle de recherche conceptuel, ni à la recherche spécifique aux entités nommées, mais à l'intégration de la sémantique par l'expansion de la requête pour un modèle de recherche textuel, tout en considérant l'importance des entités nommées.

3.4.3 Utilisation d'une technique indirecte (Implicit Feedback)

Pour modifier une requête, cette technique utilise les informations collectées de façon indirecte à la suite des interactions des utilisateurs dans un processus de recherche d'information. Le plus fréquent est d'utiliser les historiques (*logs*) des sessions de recherche générés par l'ensemble des utilisateurs, ce qui demande un volume important de données. Pour cette raison, nous trouvons ce genre d'approche dans un contexte de recherche sur le Web. L'utilisation du retour de pertinence indirect n'est pas aussi exact que la méthode explicite dans laquelle les utilisateurs jugent la pertinence des documents renvoyés par le système. Cependant, certains travaux sur la recherche Web [White et al., 2002] montrent que cette technique pourrait être un substitut efficace dans des environnements interactifs. Parmi les études intéressantes dans ce domaine, nous trouvons le travail de Cui et al. [2003]. Leur idée est de profiter du lien entre les termes des requêtes et les documents choisis dans les sessions de recherche des utilisateurs. Si un document est souvent ouvert pour la même requête dans plusieurs sessions de recherche, les termes de ce document sont probablement pertinents vis à vis des termes de la requête, et sont donc de bons candidats pour l'expansion de nouvelles requêtes similaires. Aussi, sur les historiques des utilisateurs, Wang and Zhai [2008] proposent l'extraction de motifs (*patterns*) à partir de ces historiques. Ces motifs sont utilisés pour décider si un terme associé à la requête doit être ou non ajouté. D'autres ressources indirectes pour l'expansion ont été explorées en recherche d'information comme l'utilisation des textes d'ancrage (*anchor texts*) [Dang and Croft, 2010] figurant une fois dans les logs de recherche et aussi dans la collection pour enrichir les futures requêtes, ou encore, l'expansion de requête personnalisée à l'aide de données extraites du Web social [Zhou et al., 2012].

4. http://ln.ontotext.com/KIM/screen/ontobrowse.jsp?m=ontology

3.4.4 Synthèse

Nous récapitulons dans le tableau 3.1 les approches d'expansion de la requête mentionnées précédemment. Pour connaître plus d'approches sur l'expansion de la requête, le lecteur intéressé peut consulter le tableau 3 de Carpineto and Romano [2012] qui en contient d'autres, mais avec une vision différente de la notre. Carpineto and Romano [2012] proposent également dans leur papier une comparaison des *MAP* rapportés par plusieurs études d'expansion de la requête. Nous pensons que cette mise en parallèle n'est pas très informative, car les études concernées n'utilisent pas les mêmes modèles de recherche. De plus, nous constatons des différences dans les valeurs de la mesure *MAP*, rapportées par certaines études sur le même modèle de référence et avec la même collection de tests (ce qui peut être expliqué par la différence de paramètres d'indexation comme la lemmatisation et la gestion des mots vides).

Catégorie	Approche	Détails
Collection de documents	*Globale*	
	[Lesk, 1969]	classification des termes
	[Minker et al., 1972]	classification des termes
	[Qiu and Frei, 1993]	thésaurus de similarité
	[Jing and Croft, 1994]	thésaurus d'association
	Locale	
	[Rocchio et al., 1965]	adaptation du modèle vectoriel
	[Ponte, 1998]	adaptation du modèle de langue
	[Lavrenko and Croft, 2001]	adaptation du modèle de langue
	[Zhai and Lafferty, 2001]	adaptation du modèle de langue
	[Xu et al., 2007]	LSI + pseudo retour de pertinence
	[He et al., 2009]	LSA + pseudo retour de pertinence
	[Lv and Zhai, 2010]	analyse locale + proximité
	[Luo et al., 2012]	ESA + pseudo retour de pertinence
Ressource Sémantique	[Voorhees, 1994]	synonymes de WordNet
	[Navigli and Velardi, 2003]	glossaire de WordNet
	[Liu et al., 2004]	WordNet(synonymes, hypernymes et définitions) + pseudo retour de pertinence
	[Shah and Croft, 2004]	WordNet++ pseudo retour de pertinence
	[Fang, 2008]	définitions de WordNet + modèle axiomatique
	[Zhang et al., 2009]	synonymes de WordNet
Indirecte	[Cui et al., 2003]	historiques
	[Wang and Zhai, 2008]	motifs extraits à partir des historiques
	[Dang and Croft, 2010]	texte d'ancrage
	[Zhou et al., 2012]	Web social.

TABLE 3.1 – Quelques approches d'expansion de la requête les plus connues/représentatives de chaque catégorie de la Fig.3.1.

3.5 Discussion : Les défauts et les défis des approches d'expansion de la requête

Le fait de modifier automatiquement la requête de l'utilisateur introduit plusieurs problèmes quelque soit l'approche utilisée pour l'expansion. Nous citons dans cette section les défauts et les défis les plus importants, de notre point de vue, dans le domaine de l'expansion des requêtes.

3.5.1 Risque de dérive de la requête

La dérive de la requête signifie l'altération du but de la recherche [Mitra et al., 1998]. C'est un phénomène qui se produit lors de l'expansion (généralement automatique) de la requête. Les nouveaux termes ajoutés à une requête peuvent pousser les résultats vers un sens différent de celui cherché à la base. Ainsi, après plusieurs modifications automatiques de la requête, les concepts cherchés vont probablement finir par s'éloigner du besoin d'information original. La dérive de la requête est souvent un comportement « mauvais » que l'on souhaite éviter. Néanmoins, certains auteurs, comme Stenmark [2005], proposent de voir positivement son effet dans le cadre de requêtes exploratoires (Sect. 2.3.1.1 page 16) dans lesquelles l'utilisateur peut apprécier la découverte de nouveaux aspects qu'il ne connaissait pas auparavant.

3.5.2 Qualité des termes d'expansion

Pour la plupart des approches d'expansion de la requête, l'ajout de nouveaux termes est systématique pour toutes les requêtes, même si beaucoup de ces approches considèrent l'expansion sélective comme une étape importante dans leurs perspectives. Nous trouvons dans la littérature des méthodes qui mesurent la qualité d'une requête ou celle d'un terme de la requête. Par exemple, le travail de Cronen-Townsend et al. [2002] propose une métrique de clarté, basée sur le calcul de l'entropie entre le modèle de la requête et celui de la collection. Cet article affirme une relation entre cette métrique et l'ambiguïté d'une requête, par contre, les expériences de Zobel [2004] ont montré qu'elle n'est pas efficace pour reconnaitre les requêtes qu'on ne doit pas étendre, même si Zobel [2004] n'explique pas clairement comment cette mesure a été utilisée dans son expérience. Shah and Croft [2004] n'ont pas confirmé non plus l'efficacité (en précision moyenne) de la mesure de clarté sur le choix des termes de la requête qui nécessitent une expansion (Sect. 3.4.2.1). Une approche plus récente proposée par Zhao and Callan [2010] cherche à classer les termes d'une requête par rapport à la probabilité qu'ils soient importants pour la requête. Les auteurs ont utilisé une technique de retour de pertinence accompagné de LSI (Sect. 2.4.1 page 19). L'efficacité de cette technique pour choisir et pondérer des termes d'expansion n'a pas été vérifiée.
Nous signalons que ces méthodes d'évaluation de la qualité, sont dépendantes des statistiques sur la collection de documents, et n'utilisent pas des informations sur

le terme lui même et son lien avec les termes employés par l'utilisateur dans la requête. Le fait de ne pas considérer cet aspect dans les méthodes d'expansion de la requête (surtout les approches locales), rend difficile la compréhension de la requête étendue et l'interprétation du lien entre la requête originale, la requête étendue, et l'amélioration ou la dégradation des résultats de la recherche.

3.5.3 Problématiques des techniques locales

Un défi majeur pour les approches locales est le choix des paramètres, surtout le nombre de documents de retour de pertinence. Plusieurs études se sont intéressées à cette question sans vraiment donner de réponse [Montgomery et al., 2004; Zobel, 2004]. Néanmoins, ces études ont confirmé le souci de stabilité, où un bon réglage des paramètres ne marchera pas forcément sur d'autres collections de tests, et même pas sur deux requêtes différentes sur la même collection de documents. De nos jours, la plupart des approches d'expansion locale fixent le nombre de documents de pseudo retour de pertinence à la suite d'expériences d'apprentissage alors que d'autres utilisent des algorithmes progressifs qui cherchent à maximiser une métrique donnée [Deveaud et al., 2013]. Un autre problème majeur des approches locales concerne l'évaluation. Nous évoquons ce problème en détails dans le chapitre 5 (Sect. 5.2.4).

3.5.4 Comment choisir une approche d'expansion des requêtes ?

Bien qu'en réalité beaucoup d'éléments peuvent affecter le choix d'une approche d'expansion, comme le temps de calcul, le stockage nécessaire, la disponibilité des ressources, etc., d'autres points de fond doivent être également considérés. Principalement, l'avantage d'une approche d'expansion de la requête doit être mesurée en fonction des priorités de l'utilisateur dans un contexte donné. Une bonne approche pour l'expansion des requêtes Web est celle qui améliore la précision, alors que c'est le rappel [5] qu'il faut observer pour une approche d'expansion des requêtes dans un domaine médical. Il serait intéressant que chaque étude communique le compromis rappel/précision de son approche, ce qui permettrait d'étudier l'intérêt de son utilisation dans un contexte ou dans un autre. Malheureusement, la plupart des travaux que nous avons étudiés dans les sections précédentes mesurent la performance des approches en utilisant uniquement des métriques de précision, comme le MAP et la précision à un certain rang dans la liste de résultats.
Un autre critère important lors du choix de la méthode d'expansion est la nature de la collection de documents. Par exemple, une collection dynamique, susceptible d'être modifiée fréquemment par l'ajout, la suppression ou la modification de documents est probablement moins adaptée aux méthodes d'expansion dépendantes de statistiques sur la collection qu'une collection stable.

5. Nous parlerons de la problématique de l'évaluation du rappel dans le chapitre 5.

3.5.5 Expansion de la requête en pratique

Il n'est pas facile de faire le lien entre l'expansion de la requête au sens académique et dans le monde commercial. Lorsque l'on cherche à optimiser certaines mesures statistiques dans le domaine de la recherche, dans la réalité, l'utilisateur s'intéresse plutôt à la rapidité de la réponse, la simplicité d'interface et la compréhensibilité du comportement du moteur de recherche. De nos jours, un utilisateur non spécialiste n'est souvent pas prêt à choisir les concepts de sa requête, envoyer un retour de pertinence, ou même utiliser des opérateurs qui peuvent aider le système à mieux comprendre son besoin d'information.

Les systèmes commerciaux ont bien compris cet enjeu. Google, Yahoo et Bing accompagnent la requête de l'utilisateur dès sa création, en proposant de compléter la requête durant la rédaction des mots-clés dans la case de recherche. Bien que les détails techniques de fonctionnement de ces moteurs de recherche commerciaux soient bien vagues, nous savons au moins que la base de leur approche d'expansion profite des fruits des études sur l'utilisation des historiques pour l'expansion de la requête. De plus, ces systèmes disposent d'une quantité d'information immense sur les comportements des usagers. Ces données permettent à ces moteurs de recherche d'apprendre, en permanence, des règles qui amélioreront les requêtes futures des utilisateurs. Par ailleurs, ces moteurs ont accès au contexte de recherche de l'utilisateur, ce qui leur permet de personnaliser les suggestions. Par exemple, l'abréviation « GM » sera étendue à « General Motors » pour un usager aux États-Unis, alors que Google proposera « Guerre Mondiale » pour un utilisateur en France [6].

3.6 Questions ouvertes

Avec les nombreuses années de recherche dans le domaine de l'expansion de la requête, nous pouvons imaginer que tous les aspects de ce domaine ont été explorés d'une manière ou d'une autre. Néanmoins, deux remarques attirent notre attention dans cette masse importante d'approches : la reformulation de la requête étendue et le traitement des entités nommées. La première concerne l'intégration des termes d'expansion avec les termes originaux pour constituer la requête finale. La plupart des approches n'étudient pas clairement cet aspect. Bien que ce sujet soit lié au modèle de recherche d'information utilisé et aux opérateurs disponibles dans ce modèle, nous trouvons rarement des écrits sur cette question. Souvent, les chercheurs se focalisent plus sur le choix des termes d'expansion et les théories de pondération de ces termes sans pour autant explorer les différentes possibilités de structurer la requête finale. Ce sujet est pourtant abordé par les études sur les requêtes longues pour intégrer une sous-requête extraite des termes originaux avec la requête initiale (ce que nous verrons en détail dans la section 3.6.1).

Le deuxième point insuffisamment traité (de notre point de vue) dans le domaine de l'expansion de la requête est le traitement des entités nommées. Bien que ce

6. Cet exemple est pris du blog de *Paul Haahr* et *Steve Baker* http://googleblog.blogspot.fr/2008/03/making-search-better-in-catalonia.html

sujet ne fût probablement pas prioritaire dans les premières années de la recherche d'information, nous pensons qu'il est un élément clé qui doit être considéré pour la désambiguïsation et l'expansion de la requête, surtout pour les requêtes Web.

Ces deux remarques font l'objet des sections suivantes, elles constituent également la motivation de notre étude.

3.6.1 Reformulation de la requête

Jusqu'à présent, nous avons parlé de l'expansion de la requête selon la définition 1 (page 27). Nous distinguons la reformulation de la requête par la définition 2.

Définition 2 *La reformulation de la requête est la procédure qui permet d'organiser les termes des requêtes dans une structure autre que le sac de mots, en utilisant des fonctions, comme la proximité et la pondération, fournis par le modèle de recherche.*

En explorant les approches d'expansion de la requête, nous remarquons assez rapidement que pour la majorité de ces approches, la requête originale et la requête étendue sont des sacs de mots pondérés ou non, selon la méthode. Plusieurs approches ont été proposées pour prendre en compte les différentes façons de combiner les termes dans une requête, que ce soient les termes originaux d'une requête longues [Metzler and Croft, 2005; Bendersky and Croft, 2008, 2010; Xue et al., 2010; Maxwell and Croft, 2013] ou bien des termes ajoutés à la requête à partir de plusieurs ressources [Bendersky et al., 2011, 2012; Deveaud et al., 2013]. Dans les paragraphes suivants, nous présentons quelques exemples des approches qui prennent en compte une structure autre que le sac de mots entre les termes d'une requête.

Metzler and Croft [2005] : Dans ce travail, les auteurs proposent un modèle qui permet de modéliser la dépendance entre les termes dans un cadre de recherche d'information probabiliste. Leur motivation vient de l'importance des expressions et de la proximité entre les termes d'une requête. Metzler and Croft [2005] construisent un graphe appelé *Random Markov Field* où les nœuds représentent des variables aléatoires du modèle (les termes des requêtes) et les arcs décrivent la dépendance entre ces variables. Trois variantes sont évaluées, l'indépendance complète (Fig.3.2 à gauche) ce qui revient au modèle de langue en utilisant les termes isolés, la dépendance séquentielle (Fig.3.2 milieu) et la dépendance complète (Fig.3.2 à droite), où dans le dernier cas nous supposons une dépendance entre tous les sous-ensembles de termes de la requête. La conclusion des auteurs est que la dépendance séquentielle est utile pour les requêtes longues, les collections homogènes et petites en taille, alors que la dépendance complète est meilleure pour les requêtes courtes sur des collections hétérogènes et grandes, mais aussi que la dépendance séquentielle peut être considérée comme une bonne approximation dans ce cas.

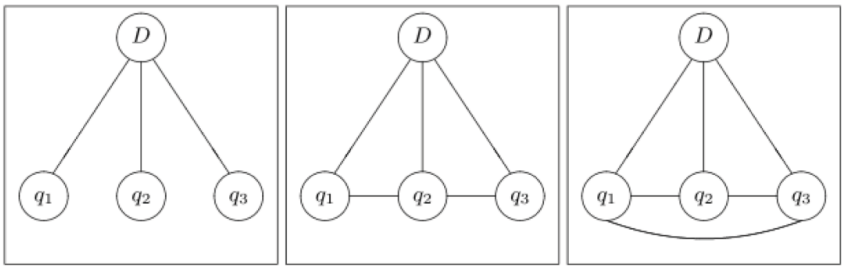


FIGURE 3.2 – L'example de [Metzler and Croft, 2005] pour la modélisation de dépendance sous le modèle de *Random Markov field*

Bendersky and Croft [2008] : Bendersky and Croft [2008] proposent une approche appelée « le concept clé » (*key concept*) qui permet de choisir les syntagmes nominaux qui représentent le mieux les concepts de la requête. Ils identifient d'abord les syntagmes nominaux dans la requête comme des candidats pour être des concepts clés. Avec l'apprentissage supervisé sur ces candidats, ils répartissent ces candidats dans deux classes : Concept Clé (CC) et Non Concept Clé (NCC). La certitude avec laquelle l'algorithme de classification attribue un concept à la classe CC est utilisée comme un poids de ce concept. Ceux qui ont les meilleurs poids sont considérés comme les concepts clés de la requête et sont intégrés avec leur poids dans la requête sac de mots. Le calcul du score initial est ainsi transformé en utilisant cette formule :

$$Score(Q, D) = \lambda p(Q|D) + (1 - \lambda) \sum_{c \in Q} p(c|Q)p(c|D) \tag{3.6}$$

où c est un concept clé découvert dans la requête Q.

Bendersky et al. [2011] : Dans cet article nous trouvons une extension du travail précédent sur la notion de « concepts clés » dans les requêtes longues [Bendersky and Croft, 2008]. Les auteurs distinguent plusieurs types de concepts : les mots isolés de la requête, les expressions qu'ils obtiennent à partir de ces mots isolés et des concepts d'expansion, qu'ils trouvent en appliquant la méthode LCE (cf. section 3.6.2) sur un ensemble de retour de pertinence de 3 collections différentes. Les concepts obtenus sont ensuite utilisés pour formuler une nouvelle requête en utilisant la formule :

$$Score(Q, D) = \sum_{T \in \tau} \sum_{\kappa \in T} \lambda_\kappa f(\kappa, D) \tag{3.7}$$

où τ est l'ensemble de types de concepts, $f(\kappa, D)$ est la fonction de recherche par vraisemblance de la requête qui considère les occurrences du concept κ dans le document D, et λ_κ est la pondération du concept κ. La pondération dans cette formule prend en compte un ensemble de caractéristiques (*features*) du concept basées surtout sur la fréquence du concept dans les trois collections utilisées comme ressource d'expansion.

Deveaud et al. [2013] : Comme Bendersky et al. [2011], Deveaud et al. [2013] s'intéressent à identifier les concepts implicites des requêtes mais sans se restreindre aux requêtes longues. Ils utilisent l'approche de l'allocation latente de Dirichlet (LDA) [Blei et al., 2003] sur un ensemble de retour de pertinence obtenu à partir d'une collection de documents différente de celle qu'on recherche. Les concepts (topics) générés par leur méthode sont des ensembles pondérés contenant des mots pondérés à leur tour. Le tout est combiné avec la requête originale en utilisant la formule :

$$Score(Q,D) = \lambda.P(Q|D) + (1-\lambda). \prod_{k \in T_{\hat{K}(M)}} \hat{\delta_k} \prod_{w \in W_k} \hat{\phi}_{k,w}.P(w|D) \tag{3.8}$$

où W_k est l'ensemble des mots du concept k, $\hat{\phi}_{k,w}$ est le poids du mot w dans le concept k, $\hat{\delta_k}$ est le poids normalisé du concept k, $T_{\hat{K}(M)}$ est le modèle conceptuel construit à partir de la source σ et appartenant à l'ensemble de ressources d'information S. Les auteurs ont évalué l'avantage de combiner quatre collections pour extraire des concepts implicites de natures différentes. Ils ont trouvé que cette combinaison fournit une meilleure performance en MAP que l'utilisation des ressources isolées.

L'idée en commun de ces approches est que la requête n'est plus un sac de mots. Il se constitue des éléments qui regroupent certains mots par la notion de proximité [Metzler and Croft, 2005; Bendersky et al., 2011], ou par la notion de concepts [Bendersky and Croft, 2008; Bendersky et al., 2011; Deveaud et al., 2013]. Par contre, toutes ces approches ne cherchent pas à faire le lien entre ces concepts et les termes originaux de la requête. De plus, même si certaines de ces approches [Metzler and Croft, 2005; Bendersky et al., 2011] cherchent à prendre en compte les expressions, ces expressions ne sont que des combinaisons de mots de la requête, et la considération des entités nommées n'est pas explicitement traitée.

3.6.2 Traitement des entités nommées dans l'expansion de la requête

Les entités nommées ont fait l'objet d'études dans les domaines liés au traitement du langage naturel, pour les reconnaitre dans un texte [Guo et al., 2009], les désambiguïser [Hoffart et al., 2011], ou les classifier [Nadeau and Sekine, 2007]. Dans le domaine de la recherche d'information, les entités nommées sont souvent utilisées pour l'annotation [Kiryakov et al., 2004], l'indexation [Buizza, 2011] ou la recherche [Petkova and Croft, 2007]. Nous utilisons la définition 3 pour préciser notre intention du terme « entité nommée ».

Définition 3 *Une entité nommée est un groupe nominal composé d'un ou plusieurs mots décrivant un objet précis : une personne, un lieu, une entreprise, ou un événement connu.*

Des travaux récents de Guo et al. [2009] ont remarqué l'importance des entités nommées pour les requêtes des utilisateurs, car ils ont constaté que plus de 70% des requêtes Web contiennent au moins une entité nommée. En revanche, ces travaux s'intéressaient plutôt à la classification des entités nommées de la requête sans considérer l'effet de ces éléments sur la performance du modèle de recherche. Dans les approches d'expansion de la requête en recherche d'information, les entités nommées sont souvent traitées comme les autres termes de la requête. Les approches fondées sur la collection de documents utilisent des calculs statistiques sur les termes dans les documents sans un traitement spécifique concernant les entités nommées. Nous rencontrons certaines approches locales qui peuvent trouver des termes d'expansion composés de plusieurs mots, sans pour autant faire la différence par rapport à leur nature (entité nommée ou non). Par exemple, Xu and Croft [2000] ont proposé l'approche de *Local Context Analysis* (LCA), où ils cherchent des noms et des syntagmes nominaux (appelés des concepts) dans les paragraphes (*passages*) des documents de pseudo retour de pertinence. Les concepts trouvés sont ordonnés selon le degré de co-occurrence de chaque concept trouvé avec l'ensemble des termes de la requête, ce qui est considérés comme une pondération du concept dans la requête finale. Ce degré de co-occurrence avec la totalité des termes de la requête est ensuite combiné avec la fréquence documentaire pour construire la pondération des nouveaux « concepts » ajoutés. Ce qui différencie le plus cette méthode est son intérêt aux termes composés de plusieurs mots, même si l'avantage apporté par ce genre de termes sur la performance n'a pas été clairement expliqué. Néanmoins, cette méthode a été utilisée comme une méthode de référence par beaucoup d'études pour sa bonne performance et sa disponibilité dans le système de recherche Inquery.

Plus tard, l'approche *Latent Concept Expansion* (LCE) a été proposée par Metzler and Croft [2007]. Cette approche est plus formelle de celle de LCA. Elle est considérée comme une extension du modèle de recherche d'information par les chaines de Markov. Ce modèle de recherche est un modèle probabiliste construit sur l'idée de considérer une dépendance entre les termes consécutifs de la requête (Sect. 3.6.1). LCE étend cette idée pour considérer également la dépendance entre les termes des documents de retour de pertinence. Ce qui nous intéresse dans ce paragraphe est que contrairement à l'étude précédente, les auteurs ont évalué ce qu'apporte l'utilisation des syntagmes nominaux par rapport à l'usage des mots simples. Ce qui est intéressant est que les auteurs ont constaté que, contrairement à ce qu'on peut imaginer, l'utilisation des syntagmes nominaux choisis dans les documents de retour de pertinence comme termes d'expansion n'apporte pas beaucoup d'avantage en MAP. La raison principale est que les mots isolés choisis pour l'expansion apparaissent dans des expressions que l'approche choisit également comme concept d'expansion. Malgré cette remarque, l'approche LCE de l'expansion de la requête a été considérée comme une approche performante et est utilisée comme modèle de référence dans certains études.

Par ailleurs, les méthodes fondées sur une ressource externe sont confrontées à la difficulté de traitement des entités nommées surtout quand la ressource externe décrit mal ces éléments [Navigli and Velardi, 2003]. Par exemple, l'une des difficultés

de l'utilisation de Wordnet pour l'expansion de la requête [Mandala et al., 1998] est que les entités nommées dans cette ressource sont souvent manquantes, pas à jour, ou n'ont pas (ou très peu) de synonymes dans leurs synsets. Ainsi, Wordnet n'est pas suffisant pour désambiguïser ou étendre les requêtes qui contiennent des entités nommées.

Malgré ces difficultés liées à l'expansion de la requête, d'autres études en recherche d'information confirment l'importance des entités nommées dans les requêtes. Par exemple, certaines recherches sur les requêtes longues estiment qu'une sous-requête contenant une entité nommée est un bon candidat qui doit être considéré pour la reformulation [Huston and Croft, 2010; Kumaran and Carvalho, 2009]. Le travail de Maxwell and Croft [2013] propose un algorithme pour classer des groupes nominaux (donc éventuellement des entités nommées) identifiés dans la requête afin de les utiliser pour construire une nouvelle requête. Récemment, la disponibilité de ressources riches et ouvertes, comme Wikipédia, a permis certains travaux explicitement dédiés à l'étude de l'expansion des entités nommées. Entre autres, Xu et al. [2008] ont utilisé Wikipédia pour extraire des termes sémantiquement proches de ceux de la requête, alors que Brandao et al. [2011] ont étudié des approches basées sur les pages et les Infobox de Wikipédia pour retrouver des expansions des entités nommées. Comme ceux de Xu et al. [2008] et de Brandao et al. [2011], notre travail s'intéresse aussi à l'expansion des entités nommées. Par contre, nous faisons le point sur les méthodes d'intégration des termes d'expansion dans la requête, ce qui n'était pas abordé dans les travaux précédents. De plus, nous étudions l'effet du rôle des entités nommées sur l'expansion.

3.7 Bilan et positionnement

Nous venons de présenter les méthodes d'expansion de la requête, les avantages et les défauts les plus importants dans le domaine, et les aspects de la reformulation de la requête et les entités nommées pour l'expansion de la requête. Dans cette thèse, nous cherchons à rapprocher la sémantique de la recherche d'information via l'expansion de la requête. Sur cette question, nous pouvons voir les approches précédentes de l'expansion de la requête de plusieurs points de vue. La première conception est de considérer comme sémantique toute expansion de la requête, même sans l'utilisation des ontologies. Cette idée est soutenue par l'ambiguïté de la notion de concept : exprimer un concept par une requête textuelle signifie sa transformation en une collection de mots ou d'expression. La relation entre ces mots et expression est le sens, ce qui est encore une notion subjective. Nous pouvons donc considérer que n'importe quelle approche d'expansion de la requête (suffisamment performante) finit par enrichir la requête par des termes appartenant aux concepts de cette requête, et donc toutes les approches d'expansion de la requête sont sémantiques. La deuxième optique est de considérer qu'une approche d'expansion de la requête est sémantique si elle utilise une ressource sémantique (une ontologie ou un thésaurus) durant le processus d'expansion.

De notre point de vue, les ontologies et les thésaurus sont certes des ressources sémantiques intéressantes, mais ce ne sont pas les seules façons pour extraire des concepts. Nous avons constaté en étudiant l'état de l'art que des méthodes statistiques sur une collection de documents peuvent être utilisées pour extraire des concepts dits « cachés ». Avec cette idée, nous considérons que la transformation d'une requête textuelle en une requête sémantique passe par la révélation des concepts (cachés ou explicites) à travers la structure de la requête. C'est-à-dire que la requête qui, à travers sa structure, alimente le modèle de RI avec des informations sur des relations entre les termes (telles que la synonymie, la proximité et la dépendance) est une requête sémantique. Cette idée sera mieux expliquée et défendue dans le chapitre 4.

Chapitre 4

Expansion et reformulation sémantique de requête

Sommaire

4.1 Introduction

Étendre les requêtes n'est pas une idée nouvelle dans le domaine de la recherche d'information comme nous l'avons constaté dans le chapitre 3. Les travaux précédents dans cette discipline sont très nombreux, ils essaient tous de mieux cibler les

documents pertinents. Dans ce travail, nous nous focalisons sur la notion sémantique de l'expansion et de la reformulation de la requête. Notre objectif est de connaître, le lien entre la bonne requête, du point de vue d'un humain, et la requête efficace. Une bonne requête est celle qui contient des mots pertinents relativement au besoin d'information, représentés d'une façon respectant les relations sémantiques entre eux, alors qu'une requête efficace est celle qui obtient de bons résultats avec les métriques d'évaluation. Dans ce chapitre, nous nous occupons de la génération d'une bonne requête, l'évaluation de son efficacité prendra place dans le chapitre des expériences. Nous commençons par présenter notre idée sur la modélisation sémantique de la requête, ce qui fournit la ligne générale suivie par nos approches. Ensuite, nous proposons deux démarches de nature différente pour réaliser cette modélisation. Elles sont présentées en deux étapes : le choix des termes d'expansion (Sect. 4.3) et la reformulation de la requête en utilisant ces termes (Sect. 4.4). La section 4.5 est consacrée à une proposition qui combine les avantages des deux démarches par une Approche Sémantique Mixte d'Expansion et de Reformulation de la requête (ASMER) adaptée au contexte du Web. Nous terminons ce chapitre par une synthèse de nos contributions dans la section 4.6.

4.2 Modélisation sémantique de la requête

La modélisation sémantique des requêtes textuelles est la procédure faisant apparaitre les concepts d'une requête tout en laissant cette dernière applicable à un modèle de recherche classique par mots-clés. Notre but est de générer une requête plus expressive, qui permet au modèle de recherche de mieux repérer et classer les documents pertinents. Une telle procédure de modélisation nécessite des techniques d'expansion et de reformulation de la requête dans un cadre sémantique. Nous avons présenté dans le chapitre 3 plusieurs approches de l'expansion des requêtes, notre objectif est de faire un lien clair entre sémantique et modification automatique des requêtes. Pour commencer, nous clarifions notre vision des termes « expansion sémantique » et « reformulation sémantique » de la requête respectivement par les définitions 4 et 5 [1].

Définition 4 *L'expansion sémantique de la requête correspond à l'identification des concepts de la requête dans le but d'ajouter à cette dernière des termes appartenant à ces concepts.*

1. À partir de ce point, la mention des mots « expansion » et « reformulation » signifie l'aspect sémantique selon les définitions 4 et 5 sauf autre précision.

Définition 5 *La reformulation sémantique de la requête est la procédure qui permet d'exprimer les concepts de la requête par des structures textuelles. Chaque structure représente un concept de la requête et contient des termes sémantiquement proches.*

La définition 4 de l'expansion sémantique de la requête couvre une variété d'approches qui existent déjà dans la littérature [Qiu and Frei, 1993; Grootjen and Weide, 2004; Metzler and Croft, 2007; Zhang et al., 2009; Luo et al., 2012; Deveaud et al., 2013]. Ce qui nous différencie de ces travaux est que nous prêtons une attention particulière aux entités nommées, à la qualité des termes d'expansion, et à la manière avec laquelle on les représente dans la requête modifiée.
D'un côté, nous considérons que les entités nommées possèdent un rôle clé dans la procédure de la recherche d'information, notamment dans un contexte Web, car une fois désambiguïsées, ces termes possèdent une capacité d'expression assez élevée pour exprimer les concepts importants pour l'utilisateur. Par exemple, nous pouvons facilement deviner l'événement, l'objet, la date exacte et l'endroit qui intéressent la requête simple « Titanic 1912 ». Pour cette raison, nous estimons que la mention de ces entités dans une requête est un véritable atout qui mérite une exploration dans le cadre de l'expansion de la requête. D'un autre côté, nous cherchons à représenter la requête étendue par la reformulation sémantique que nous avons précisée par la définition 5. Notre stratégie est de respecter les termes employés par l'utilisateur pour exprimer son besoin d'information, et de montrer le lien entre les termes originaux et les concepts utilisés pour l'expansion. Cela a pour but de créer une requête compréhensible, et donc modulable, par l'utilisateur dans le cas d'un environnement interactif ou exploratoire, où l'utilisateur peut être intéressé par connaitre les alternatives aux termes qu'il a employés dans sa requête. Ainsi, nos approches de modélisation sémantique de la requête respectent les étapes suivantes (Fig. 4.1) : la première étape est la désambiguïsation, qui lie un terme de la requête avec un concept unique. Une fois ce lien établi, ce concept est utilisé pour cibler les termes les plus proches sémantiquement du terme original. Ces termes forment ce qu'on appelle une « formule étendue » que nous précisons par la définition 6.

Définition 6 *Une formule étendue d'un terme de la requête est une structure textuelle qui contient des termes sémantiquement proches du terme original de la requête.*

Enfin, la dernière étape est d'intégrer les formules étendues avec la requête originale pour constituer la requête finale.

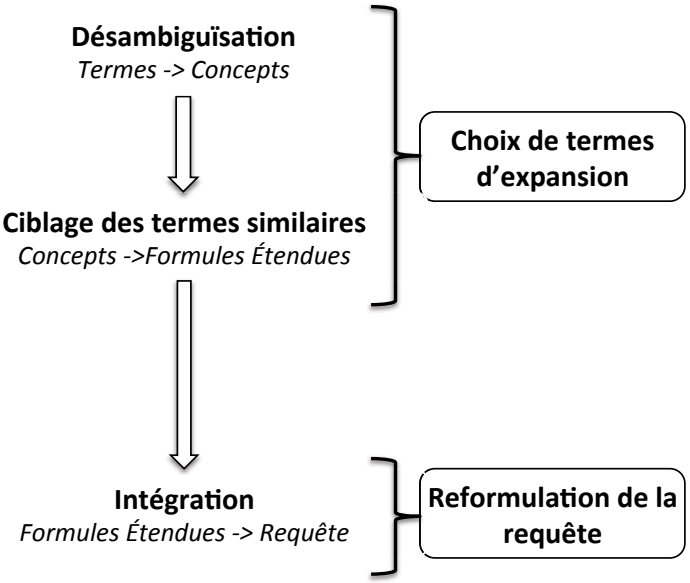


FIGURE 4.1 – Les étapes de nos approches d'expansion et de reformulation sémantique des requêtes

4.3 Choix des termes d'expansion

Pour ajouter de nouveaux termes à une requête, nous pouvons utiliser la collection de documents (entière ou partielle par une technique de retour de pertinence), une ressource sémantique externe ou bien des ressources implicites comme les logs (Sect. 3.4). Nous limitons notre étude à l'utilisation d'une ressource sémantique et d'une technique locale. Nos approches pour trouver les nouveaux termes passent par l'intermédiaire des concepts (Fig. 4.1). Pour commencer, nous considérons l'hypothèse 1.

Hypothèse 1 *Chaque terme utile (i.e. non vide) de la requête originale appartient à un concept.*

Cette hypothèse forte est essentielle pour la modélisation de la requête finale, une fois les formules étendues trouvées. Nous la justifions par le fait qu'en général, un utilisateur ne cherche pas à utiliser un seul mot pour exprimer deux concepts différents, par contre, un concept peut être exprimé par plusieurs mots. Pour cela, même si nous nous basons sur l'hypothèse 1 pour initialiser le nombre de concepts, et pour conserver les liens avec les termes originaux de la requête, notre contribution prendra en compte la détection des cas où plusieurs termes originaux de la requête appartiennent au même concept. Ainsi, le nombre de concepts dans nos requêtes reformulées est égal ou inférieure au nombre de mots utiles dans la requête originale.

Nous faisons la différence entre un concept explicite et un concept implicite de la requête. Nous constatons l'utilisation fréquente des mots « implicite » et « explicite » dans beaucoup d'approches en recherche d'information [Qiu and Frei, 1993; Grootjen and Weide, 2004; Metzler and Croft, 2007; Zhang et al., 2009; Luo et al., 2012; Deveaud et al., 2013; Bendersky et al., 2011; Egozi et al., 2011]. La signification de ces mots est différente selon les approches. Notre utilisation de concepts implicites et explicites est basée sur la ressource employée pour l'expansion de la requête, où les concepts explicites sont découverts avec les ressources sémantiques externes et les concepts implicites sont découverts par une technique locale de l'expansion de la requête. Une explication plus précise de ces deux types de concepts sera donnée par les définitions 7 et 8 dans les sections correspondantes.

Nous proposons dans les sections suivantes des approches pour l'utilisation d'une ressource sémantique externe et pour l'emploi d'une technique de pseudo retour de pertinence pour l'expansion sémantique de la requête.

4.3.1 Avec une ressource sémantique

La recherche de termes d'expansion dans une ressource sémantique se base sur l'identification des concepts explicites de la requête selon la définition 7.

Définition 7 *Un concept explicite est un concept identifiable dans une ressource sémantique.*

Le choix de la ressource sémantique à utiliser pour l'expansion de la requête est crucial et doit être en harmonie avec le domaine de la collection de documents recherchée. Quand nous sommes dans un contexte de recherche Web où le domaine de recherche de la requête ne peut pas être identifié par avance, une ressource sémantique non liée à un domaine spécifique est la mieux adaptée. Le choix le plus naturel dans ce cas est l'utilisation d'un thésaurus comme Wordnet. Une telle ressource couvre un vocabulaire assez important de la langue anglaise, ce qui laisse imaginer une capacité élevée pour trouver des termes intéressants pour enrichir la requête originale. Malgré de nombreux essais dans l'état de l'art d'utilisation de cette ressource pour l'expansion de la requête (Sect. 3.4.2.1, page 33), Wordnet n'a pas rencontré beaucoup de succès, principalement concernant le traitement des entités nommées. L'ontologie Yago a permis de résoudre ce défaut majeur de Wordnet tout en gardant les avantages de cette ressource linguistique. Comme nous l'avons expliqué dans la section 2.2.1.4, Yago est une combinaison de Wordnet et des extraits de Wikipédia. Il lie, par la relation « rdf :type », les concepts correspondant aux articles de Wikipedia avec les synsets de WordNet. À part ce lien, la structure et le contenu (définitions des synsets, relations d'hyperonymie et d'hyponimie) de WordNet sont bien préservés dans Yago, et les articles de Wikipedia sont principalement utilisés

pour les entités nommées[2]. Ainsi, nous pouvons voir Yago comme une ressource combinant des entités nommées et WordNet. C'est la ressource que nous utilisons dans notre approche d'expansion de requêtes basée sur une ressource sémantique.

Avant de pouvoir construire les formules étendues des termes de la requête, il faut lier ces derniers aux concepts de Yago (désambiguïsation). Pour chaque terme q de la requête, un score est calculé pour chaque concept candidat c_i dans l'ontologie. Nous considérons que le concept c_q qui obtient le score le plus élevé est le concept qui correspond au terme q (équ. 4.1).

$$c_q = \text{argmax}_{c_i \in C(q)} S_{dis}(q, c_i) \tag{4.1}$$

où $C(q)$ est l'ensemble des concepts candidats du terme q, et $S_{dis}(q, c_i)$ est le score de désambiguïsation entre le terme q et un concept c_i. A cause de la spécificité des entités nommées, nous avons choisi de distinguer le calcul de leurs scores $S_{dis}(q, c_i)$ (Sect. 4.3.1.1) de celui des autres termes de la requête (Sect. 4.3.1.2) comme nous le présentons dans la figure 4.2. Une fois la désambiguïsation faite, le choix des termes d'expansion peut être appliqué (Sect. 4.3.1.3).

4.3.1.1 Désambiguïsation des entités nommées

Nous avons besoin de lier les entités nommées de la requête à des concepts dans Yago pas seulement pour chercher leurs formules étendues, mais aussi pour qu'elles participent à la désambiguïsation des autres termes de la requête. Bien que nécessaire dans un texte qui a suffisamment de contexte, la désambiguïsation des entités nommées semble moins importante pour les requêtes très courtes du Web. Nous présentons dans cette section l'approche de désambiguïsation *Aida*[3], et nous examinons dans la partie expérimentale (Chap. 6) l'utilité de cette approche par rapport au choix du sens le plus populaire d'une entité nommée.

Aida [Hoffart et al., 2011] utilise *Stanford NER* (*Named Entity Recognition*) pour identifier les entités nommées q_{en} dans un texte, qui est dans notre cas la requête, puis elle applique une combinaison de trois scores : le premier $prior(c, q_{en})$ est la probabilité qu'un concept c soit le sens le plus commun de q_{en}. Cette probabilité est calculée à l'aide de statistiques sur Wikipédia (comme le nombre de liens vers la page correspondant dans Wikipedia). Le deuxième score $sim(ctx(q_{en}), ctx(c))$ est la similarité entre q_{en} et un concept candidat c dans l'ontologie. Cette similarité prend en compte la syntaxe en commun entre le contexte à proximité de l'entité nommée dans la requête $ctx(q_{en})$, et les « phrases clés[4] » $ctx(c)$ du concept candidat dans Yago. Enfin, Aida considère le score $coh(c, \bigcup_{q'_{en}} C(q'_{en}))$ entre le concept candidat c et les concepts candidats des autres entités nommées (q'_{en}) dans le texte. Ces trois

2. Lors de la construction de Yago, en cas de conflit (même concept trouvé en tant qu'article Wikipedia et un synset dans WordNet) les auteurs ont choisi de considérer le synset qui contient ce terme comme un concept dans Yago.

3. http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/aida/

4. Les phrases clé sont récupérées en pré-traitement à partir des articles Wikipédia qui correspondent aux concepts Yago

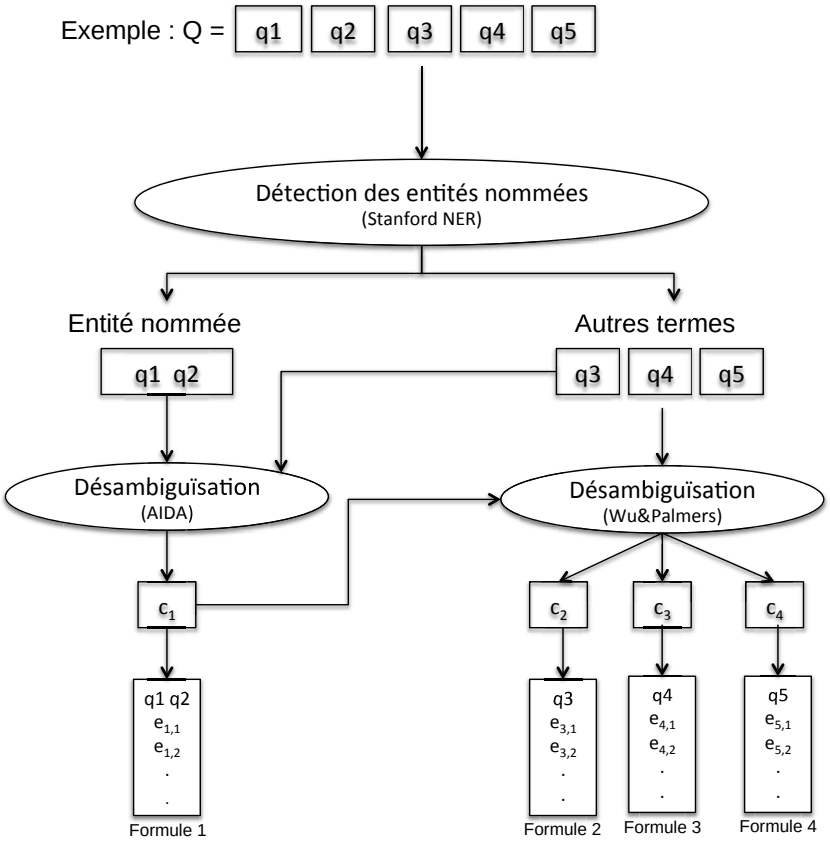


FIGURE 4.2 – Notre approche de construction de formules étendues par des concepts explicites

scores sont combinés d'une manière linéaire pour constituer le score final de chaque concept candidat selon l'équation 4.2.

$$S_{dis}(q_{en}, c) = \alpha.prior(c, q_{en}) + \beta.sim(ctx(q_{en}), ctx(c)) + \gamma.coh(c, \bigcup_{q'_{en}} C(q'_{en})) \quad (4.2)$$

où α, β et γ sont des paramètres libres. Le concept choisi est celui qui obtient le plus grand score (équ. 4.1, page 52).

4.3.1.2 Désambiguïsation des termes non-entités nommées

Pour les termes de la requête qui ne sont pas des entités nommées, nous avons adopté une méthode basée sur la structure de WordNet (Sect. 2.2.2, page 13). Nous rappelons que cette méthode est fondée sur une distance sémantique entre les concepts en utilisant l'équation suivante [Navigli, 2009] :

$$S_{dis}(q, c) = \sum_{q' \in Q : q \neq q'} \max_{c' \in C(q')} Sim(c, c') \quad (4.3)$$

où q est le terme que l'on souhaite désambiguïser, Q est l'ensemble des termes de la requête, $C(q')$ est l'ensemble des concepts auquel le terme q' appartient, ce qui correspond dans Wordnet à l'ensemble des synsets qui contiennent ce terme. $Sim(c, c')$ est la fonction utilisée pour mesurer la similarité entre deux concepts c et c'. Le concept qui obtient le score de désambiguïsation le plus élevé est le concept correspondant au terme q (équ. 4.1, page 52). Plusieurs méthodes existent pour mesurer la similarité entre deux concepts c et c' (Figure 2.7, page 14). Nous choisissons, à la suite de plusieurs expériences de comparaison, une approche basée sur le parcours des arêtes du graphe [Palmer and Wu, 1994]. Ce procédé suppose que la similarité entre deux concepts dépend de la profondeur des nœuds concernés et de leur plus proche ancêtre commun (*Least Common Concept*[5]) par rapport à un nœud racine dans la ressource. Cette mesure de similarité s'applique aux synsets qui se trouvent dans la même taxonomie. Ainsi, la présence d'un verbe ou d'un adjectif dans la requête, qui est moins fréquente que la présence des noms dans les requêtes du Web, est ignorée pendant la désambiguïsation des noms. Pour les verbes et les adjectifs, nous choisissons le synset associé au sens le plus fréquent.
Par ailleurs, dans le cas où le synset choisi ne contient aucun autre terme que celui que nous essayons d'étendre, nous considérons pour l'expansion les termes du synset père (lié par la relation d'hyperonymie[6]). Il est vrai que le passage à un synset plus abstrait conduit à la généralisation du sens de la requête, mais nous le faisons car chaque terme ajouté comme expansion sera contrôlé par rapport à sa spécificité (ce qu'on va voir dans la section 4.3.3.2), et ne sera donc pas adjoint s'il est trop abstrait.

Cette méthode prend aussi en compte les entités nommées de la requête de la manière suivante : une fois que l'entité nommée dans la requête liée à un concept

5. *Least Common Concept* est le premier ascendant commun entre deux noeuds.
6. L'hyperonyme d'un terme est son prédécesseur direct dans la taxonomie IS-A.

Trec Requête	Termes similaires dans Yago
515 what about **Alexander Graham Bell**	"Alexander Gram Bell", "Aleck Bell", "The father of the deaf"
478 **Baltimore**	"City of Baltimore", "Baltimor", "Baltimore riots"
480 **car** traffic report	auto, machine
485 **gps** clock	"Global Positioning System"
490 motorcycle **safety** helmets	guard, "safety device"

TABLE 4.1 – Exemples de variations sémantiques trouvés dans YAGO pour un terme (en gras) de la requête.

dans Yago (selon la section précédente 4.3.1.1), nous prenons la relation « rdf :type » qui lie le concept de cette entité à un synset de WordNet. Le score de similarité avec ce synset est pris en compte pour la désambiguïsation de tous les termes de la requête selon la fonction 4.3.

4.3.1.3 Choix de termes d'expansion

Dans l'ontologie Yago, les relations sémantiques qui lient un concept à son entourage sont nombreuses [7]. Dans le cas de concepts correspondant aux entités nommées, ces relations dépendent de la nature de chaque entité : par exemple, une ville peut être reliée à sa surface ou à son nombre d'habitants, alors que d'autres types de relation sont utilisés dans le cas d'une personne (comme sa date de naissance). Dans le cas de l'expansion de la requête, le choix de la bonne relation sémantique n'est pas une tâche facile. Afin de limiter le risque de dérive de la requête, nous considérons les relations de similarité sémantiques pour lier les termes de la requête à leurs termes d'expansion. Pour cela, nous considérons la relation « rdfs :label » qui existe pour tous les concepts, elle désigne la synonymie pour les concepts d'origine WordNet, et lie une entité nommée à ses différentes variations sémantiquement similaires. Ces variations correspondent au lien « *redirect* » dans Wikipedia (Sect. 2.2.1.4, page 11). Ces termes peuvent donc être des alternatives orthographiques du terme (Baltimore-Baltamore), ou être complètement différents au niveau de la syntaxe, mais sémantiquement liés au terme original (Baltimore-Mobtown). Le tableau 4.1 présente quelques termes (entité nommée ou non) avec leurs formules étendues obtenues par le lien « rdf :label » dans Yago.

4.3.2 Avec une technique locale

L'utilisation d'une ontologie pour trouver des termes d'expansion permet d'obtenir des expressions et des variations impossibles à recueillir par des méthodes locales.

7. Yago contient 72 types de relations sémantiques (`http://www.mpi-inf.mpg.de/yago-naga/yago/statistics.html`).

Par contre, des informations supplémentaires sur le contexte de la requête sont également nécessaires pour connaître d'autres « concepts » cachés derrière les termes de la requête. Les méthodes de pseudo retour de pertinence peuvent être efficaces pour résoudre ce problème.

Nous présentons dans cette partie notre idée de trouver des termes sémantiquement proches des termes de la requête. Alors que la similarité sémantique est bien formalisée dans Yago par la relation (« rdf :label »), trouver ce genre de lien dans un ensemble de documents nécessite de lier les termes de ces documents aux concepts de la requête. Sur l'idée d'exprimer les concepts d'une requête, plusieurs techniques peuvent être employées sur l'ensemble de documents de pseudo retour de pertinence comme LDA, ESA, ou LSA (Sect. 3.4.1.2, page 32). Nous choisissons d'utiliser LSI pour l'expansion de la requête. LSI permet de découvrir les relations de co-occurrence à plusieurs niveaux[8] entre les termes dans les documents. Elle se base sur la décomposition des valeurs singulières (SVD) d'une matrice A (termes $\times$ documents) que nous générons à partir des documents de pseudo retour de pertinence. Dans notre étude, la matrice A contient la fréquence (tf) des termes dans les premiers documents rendus par le modèle de recherche pour la requête originale. L'application de SVD sur la matrice A qui contient m termes et n documents nous donne les trois matrices de l'équation 4.4.

$$A_{\{m,n\}} = U_{\{m,m\}} S_{\{m,n\}} V^T_{\{n,n\}} \tag{4.4}$$

où S est la matrice diagonale qui contient les valeurs singulières de la matrice A. L'idée de LSI est basée sur la possibilité de réduire à k le nombre de dimensions de ces matrices tout en ayant une approximation acceptable de la matrice originale, c'est-à-dire que la matrice A' de l'équation 4.5 peut être considérée comme une alternative acceptable de la matrice originale A.

$$A'_{\{m,n\}} = U_{\{m,k\}} S_{\{k,k\}} V^T_{\{k,n\}} \tag{4.5}$$

Cette pratique est justifiée par le fait que les valeurs de ces matrices sont ordonnées selon leur importance par rapport aux valeurs propres des matrices AA^T et A^TA. La suppression des colonnes à droite de la matrice U ou des lignes du bas de la matrice V^T donnera une bonne approximation des matrices complètes, et permettra d'enlever le bruit provoqué par les dimensions les moins importantes (ce qu'on peut voir en détail dans l'annexe B).
Pour l'expansion de la requête, nous nous intéressons à la matrice $U_{\{m,k\}}$, qui représente les coordonnées de chaque terme des documents de pseudo retour de pertinence dans l'espace de k dimensions générées par la décomposition de la matrice $A_{\{m,n\}}$ et la réduction de dimensionalité (Fig. 4.3). Pour trouver des termes d'expansion pour un terme q de la requête, nous calculons la similarité entre ce terme et les autres termes dans la matrice U en prenant la distance euclidienne entre les vecteurs de

8. Deux termes qui n'apparaissent pas dans le même document, mais souvent avec un troisième terme seront sémantiquement liés selon LSA.

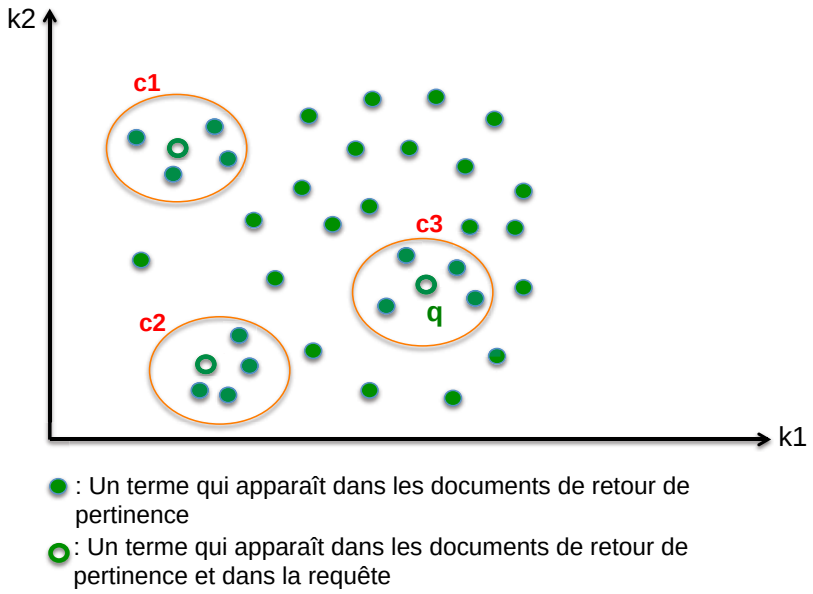


FIGURE 4.3 – Les termes dans l'espace sémantique produit par LSI (démonstration pour 2 dimensions k1 et k2)

ces termes. Nous supposons que les termes les plus similaires sont utilisés pour décrire le même aspect, ils appartiennent donc au même groupe sémantique que nous appellerons « concept implicite » comme précisé par la définition 8 et la figure 4.3. Ainsi, ces termes sont considérés comme une formule étendue du terme q qu'on peut donc intégrer dans la requête finale.

Définition 8 *Un concept implicite de la requête est un ensemble de termes appartenant aux documents de pseudo retour de pertinence et proche d'un terme de la requête dans l'espace sémantique produit par LSI.*

Les étapes de construction des formules étendues par l'approche locale sont illustrées par la figure 4.4 La relation entre un terme de la requête et les termes d'expansion trouvés par LSI est basée sur les informations de la co-occurrence, ils peuvent donc ne pas avoir un lien très évident pour un humain, par contre, le lien entre ces termes et les termes originaux de la requête est bien précis et clair (similarité par la distance euclidienne). La similarité élevée entre ces termes dans l'espace généré par LSI nous permet de considérer l'hypothèse que l'intégration de ces « synonymes sémantiques » peut être bénéfique pour la requête.

Dans certains cas, deux mots de la requête sont fortement liés (l'existence de l'un dans un texte signifie l'existence de l'autre également). Ces deux mots vont avoir

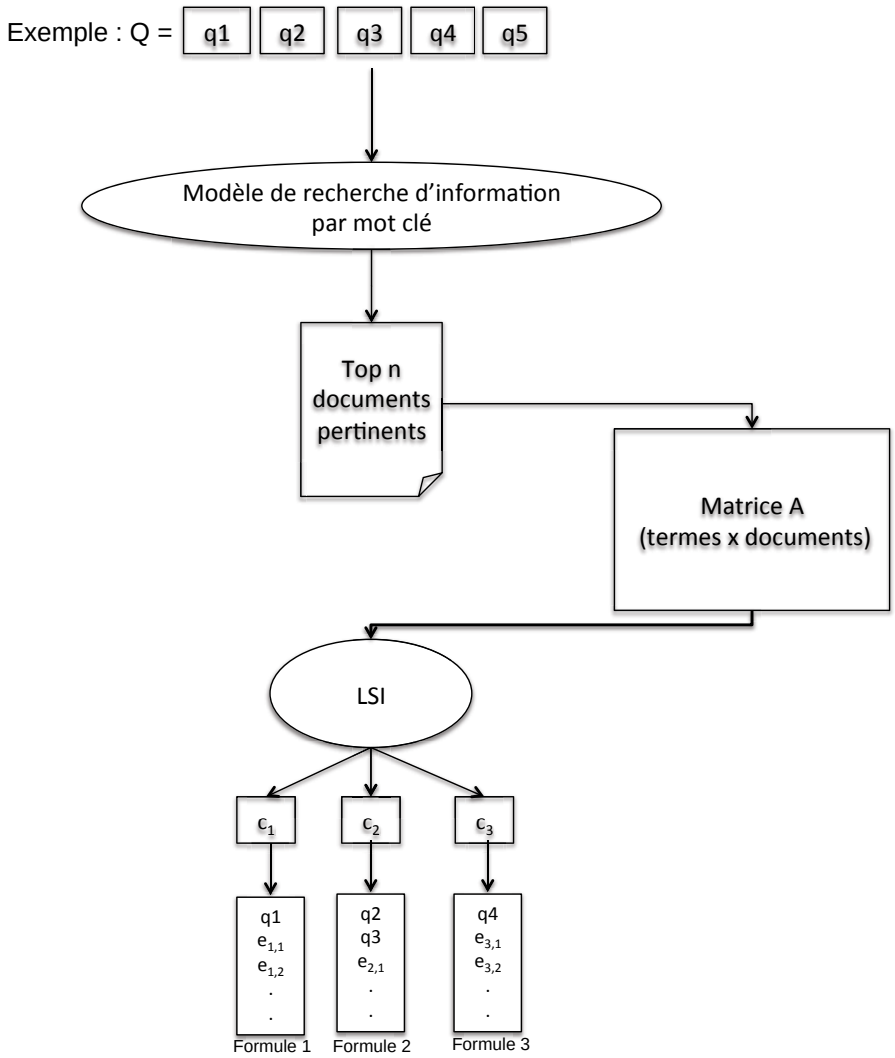


FIGURE 4.4 – Notre approche de construction de formules étendues par des concepts implicites

les mêmes statistiques dans les documents de pseudo retour de pertinence, et nous allons nous retrouver avec des termes d'expansion en commun entre ces deux termes comme nous pouvons le constater dans la figure 4.5. Dans ce cas, nous considérons

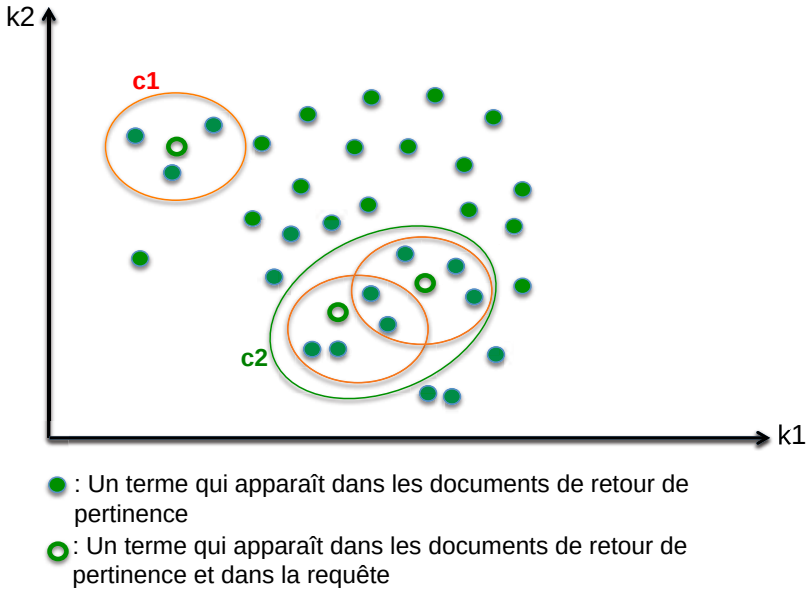


FIGURE 4.5 – L'effet des mots associés sur les concepts implicites générés par LSI

que ces deux termes originaux de la requête font partie du même concept implicite (c2 dans la figure 4.5). Cela sera modélisé par la reformulation de la requête finale par une seule formule étendue contenant les deux mots originaux et leurs termes d'expansion.

4.3.3 Qualité des termes d'expansion

Les approches précédentes génèrent des formules étendues qui correspondent aux termes originaux de la requête. Dans certains cas, les termes de la requête n'ont aucun terme d'expansion. Cela peut arriver avec l'approche de ressource externe si le terme n'existe pas dans la ressource, ou si le concept qui lui correspond n'a aucun lien sémantique de similarité avec d'autres termes. Dans le cas de l'approche locale, l'absence de termes d'expansion se produit si le terme de la requête n'apparaît pas dans les documents de pseudo retour de pertinence, par conséquent, ils n'auraient pas de vecteurs dans la matrice U. Dans ce cas d'absence de termes d'expansion pour un terme de la requête, la formule étendue de ce terme ne contient que le seul terme original. Notre idée est de valoriser les termes originaux employés par l'utilisateur, c'est pour cela que nous considérons qu'un terme original qui n'a pas de termes d'expansion (et qui n'est pas un mot vide) est le représentant unique d'un concept que l'utilisateur souhaite exprimer. Autrement dit, l'échec d'obtention de termes d'expansion pour un terme n'éliminera pas celui-ci de la requête étendue.

En revanche, durant la procédure d'expansion, l'ajout d'un terme à une formule étendue est soumis à un contrôle de qualité, dans le but de limiter le risque de dérive de la requête. Les deux critères que nous retenons pour évaluer un terme d'expansion sont la certitude et la spécificité. Si le terme ne remplit pas le seuil minimal de ces deux critères, nous considérons que l'ajout du terme étudié ne sera pas bénéfique pour la requête, ce qui conduira à son élimination de la formule étendue.

4.3.3.1 La certitude

Ce critère est censé mesurer à quel point nous sommes certains qu'un terme d'expansion va améliorer la requête. Plusieurs possibilités existent dans la littérature comme la clarté de la requête [Cronen-Townsend et al., 2002] ou la nécessité du terme [Zhao and Callan, 2010] que nous avons vues dans la section 3.5.2. Le problème de ces méthodes est que, dans le cas de l'expansion de la requête par une ressource externe, elles nécessitent le lancement de la requête générée pour chaque combinaison possible de termes d'expansion, ce qui est coûteux dans un contexte Web où le temps de recherche est un facteur important pour l'utilisateur. Nous choisissons de calculer la certitude en fonction de la ressource utilisée pour trouver les termes d'expansion au lieu de le faire en nous basant sur les résultats obtenus en intégrant ces termes dans la requête.
La considération d'une relation de similarité dans une ressource sémantique élimine le doute sur la qualité des termes si le concept que nous avons choisi est bien celui correspondant au bon sens du terme. Ainsi, pour l'approche d'expansion par une ressource sémantique, la certitude d'un terme t candidat qu'on souhaite ajouter à la formule étendue du terme q de la requête, correspond à la qualité de la désambiguïsation du terme q (équ. 4.6).

$$Cert(t, q) = \max_{c \in C(q)} S_{dis}(q, c) \tag{4.6}$$

où $S_{dis}(q, c)$ est le score de désambiguïsation du terme q et un concept candidat c selon les équations 4.2 et 4.3 (page 54). Au-dessous d'un certain seuil, nous estimons que la désambiguïsation n'est pas sûre, et qu'il vaut mieux ne pas étendre le terme correspondant.

Concernant une technique locale, la désambiguïsation est faite implicitement par l'utilisation de la requête entière lors de l'itération de pseudo retour de pertinence, nous ne pouvons donc pas considérer la même logique utilisée avec les ressources sémantique pour évaluer la certitude. Par contre, dans ce genre d'approche, l'avantage est que les documents de pseudo retour de pertinence peuvent contenir des informations importantes sur les statistiques des termes. Dans notre cas, l'utilisation de LSI en tant que technique locale d'expansion de la requête, nous rapproche du travail de Zhao and Callan [2010] (expliqué dans la section 3.5.2, page 38) sur les nécessités des termes de la requête. Contrairement à Zhao and Callan [2010] qui mesurent la nécessité des termes originaux de la requête, nous cherchons à estimer la nécessité des termes d'expansion pour savoir s'il faut les ajouter ou non à la requête. Comme nous l'avons mentionné précédemment, avec LSI, nous cherchons les termes

les plus similaires à ceux de la requête, ce qui correspond au critère de synonymie de Zhao and Callan [2010]. Ainsi, pour l'approche locale, nous considérons que plus un terme d'expansion est proche « sémantiquement » du terme correspondant dans la requête, plus nous avons la certitude qu'il est un bon candidat pour l'expansion. Ce que nous exprimons par l'équation 4.7.

$$Cert(t, q) = Dist_{euclide}(\overrightarrow{t}, \overrightarrow{q}) \tag{4.7}$$

où q est un terme de la requête, t est un terme candidat pour la formule étendue de q, $Dist_{euclide}$ est la fonction qui calcule la distance euclidienne entre les vecteurs $\overrightarrow{t}$ et $\overrightarrow{q}$ qui représentent les termes t et q respectivement dans l'espace sémantique produit par LSI. Nous définissons un seuil pour arrêter l'extraction des termes, dont la similarité avec le terme de la requête est en dessous de ce seuil.

4.3.3.2 La spécificité

L'un des avantages de Wordnet est que ses taxonomies, à base des relations d'hyperonymie et d'hyponymie, permettent de voir le niveau de spécificité des termes : les termes qui se trouvent dans des synsets profonds d'une taxonomie de WordNet (par exemple la taxonomie des noms) sont plus spécifiques que d'autres qui se trouvent dans des niveaux plus élevés. Avec le synset qui contient le mot unique « entité » au niveau le plus générique, nous trouvons souvent les entités nommées, qui sont peu nombreuses dans WordNet, dans des niveaux plus profonds, d'où leur haute expressivité. Par contre, les taxonomies des verbes, des adverbes et des adjectifes sont moins riches que celles des noms, et contiennent moins de niveaux. Si le terme est une entité nommée ou s'il n'est pas un nom, nous le considérons comme spécifique quelque soit son niveau dans WordNet. A l'exception de ces cas, un terme candidat à l'expansion est désambiguïsé dans WordNet afin qu'il soit éliminé si son synset est trop générique.

4.3.4 Résumé

Pour trouver des termes d'expansion, notre idée se résume par la recherche des termes qui ont des liens de similarité sémantique avec les termes originaux de la requête. Pour cela, nous cherchons à détecter les concepts explicites identifiables dans une ressource sémantique (Fig. 4.2), et les concepts implicites dévoilés grâce à une technique locale sur les documents de pseudo retour de pertinence (Fig. 4.4). Grâce à ces concepts, nous considérons les termes sémantiquement similaires aux termes de la requête en utilisant la relation « rdfs :label » dans Yago, et la similarité (distance euclidienne) entre les termes de la requête et les termes de pseudo retour de pertinence dans l'espace sémantique produit par LSI. L'ajout de ces termes à la requête étendue dépend de la qualité des termes, qu'on mesure par les critères de certitude et de spécificité. Au final, une formule étendue d'un terme de la requête contient ce terme combiné éventuellement avec ses termes d'expansion retrouvés par son concept explicite ou implicite, et filtrés par les contraintes de spécificité et

de certitude. Nous étudions dans la partie suivante comment générer une requête reformulée avec ces formules étendues.

4.4 Reformulation de la requête

Pour construire une requête sémantique selon la définition 5 (page 49), nous cherchons à faire apparaitre les concepts de la requête et à les représenter d'une manière compréhensible et explicite permettant à l'utilisateur de se repérer par rapport à sa requête originale. Cette idée nécessite un modèle de recherche qui permet une représentation de la requête autre que le sac de mots. Autrement dit, pour pouvoir représenter les concepts dans une requête textuelle, des fonctionnalités comme la proximité, la pondération et la prise en compte de la synonymie sont nécessaires. Le modèle de recherche proposé par Metzler and Croft [2004], appelé le modèle structuré de langue (MSL), est un bon candidat pour appliquer notre idée sur les requêtes sémantiques. En tant qu'extension du modèle de recherche par les réseaux d'inférence (implémenté dans Inquery), MSL a hérité la possibilité d'exprimer des requêtes structurées (Fig. 4.6), tout en profitant de la bonne réputation du modèle de langue par vraisemblance de la requête.

Par la suite, nous présentons un aperçu qui permet d'expliquer l'usage des paramètres de ce modèle. Une explication plus approfondie est proposée dans l'annexe A.

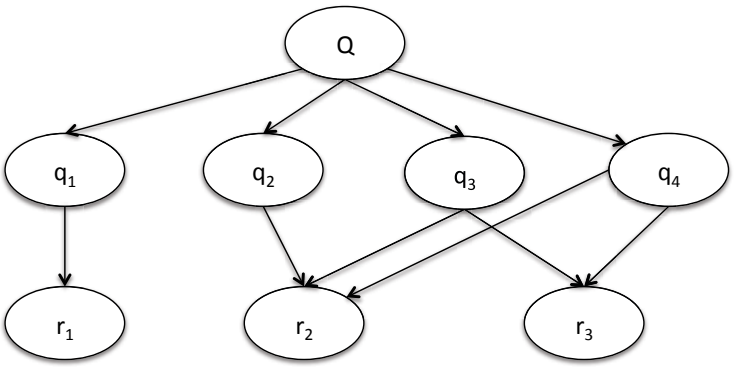


FIGURE 4.6 – Exemple de formules d'expansion par des concepts implicites

Le modèle MSL propose plusieurs façons pour combiner les croyances $b(r_i)$ des nœuds des représentations r_i dans le réseau d'inférence de la requête (Fig. 4.6) afin de calculer le score d'un document. Dans leur papier, Metzler and Croft [2004] argumentent que la meilleure façon de mettre en correspondance une requête q et un document d, selon leur modèle, est de considérer la multiplication des croyances comme exprimé par l'équation 4.8.

$$Score(Q, d) = \prod_{i=1}^{k} b(r_i)^{w_i} \tag{4.8}$$

où w_i est la pondération d'une représentation r_i. Un nœud de représentation dans ce modèle peut être un mot simple de la requête, ou une combinaison de plusieurs

Dans la requête	Signification
#combine(dog train)	$0.5log(b(dog)) + 0.5log(b(train))$
#weight(1.0 dog 0.5 train)	$0.67log(b(dog)) + 0.33log(b(train))$
#wsum(1.0 dog 0.5 dog.(title))	$log(0.67b(dog) + 0.33b(dog.(title)))$
#syn(car automobile)	une occurrence de « car » ou « automobile »
#wsyn(1.0 car 0.5 automobile)	comme #syn, mais l'occurrence de « car » est compté en double
#n(bleu car)	« blue » apparait avant « car » dans une fenêtre de n mots au plus
#uwN	« blue » apparait avant ou après « car » dans une fenêtre de n mots au plus

TABLE 4.2 – Exemples démonstratifs du fonctionnement des opérateurs d'Indri

termes en utilisant la proximité ou une autre méthode de combinaison entre les termes de la requête. La croyance d'une représentation d'un mot simple ($r_i = q$), est calculée selon le modèle de vraisemblance de la requête ($b(q_i|d) = p(q_i|d)$). Ainsi, quand toutes les représentations sont les mots simples de la requête, l'équation 4.8 correspond au modèle de langue par vraisemblance de la requête.
MSL est implémenté par l'outil Indri Strohman et al. [2004] qui fait partie du projet Lemur [9], où une variété d'opérateurs est disponible pour exprimer les différentes façons de combiner les termes d'une requête. Nous reprenons dans le tableau 4.2 les exemples cités dans la page wiki du projet Lemur. En plus de l'opérateur #weight qui correspond à l'équation 4.8, ces exemples comprennent plusieurs opérateurs de proximité et de combinaison qui nous serviront à reformuler la requête par les formules étendues, ce que nous allons voir dans les sections suivantes.

Comme beaucoup d'approches d'expansion de la requête, nous suivons l'idée de combiner d'une manière linéaire, la requête originale (sac de mots) saisie par l'utilisateur, avec la requête formée des termes d'expansion. Ainsi, nous considérons que la pertinence d'un document pour une requête est calculée par l'équation 4.9

$$p(Q|d) = \lambda \prod_{q} p(q|d) + (1 - \lambda) \prod_{i=1}^{k} b(r_i) \tag{4.9}$$

dans cette équation, r_i correspond à une formule étendue qu'on obtient grâce aux concepts explicites ou implicites selon les approches que nous avons proposées dans la section 4.3. Dans les deux cas (concepts explicites ou implicites), nous ne pondérons pas les représentations r de l'équation précédente ($w_i = 1$). Nous considérons que les représentations remplacent les termes originaux de la requête utilisateur, qui ne sont pas pondérés à la base. Autrement dit, nous supposons que tous les concepts d'une requête ont la même importance pour exprimer le

9. http://sourceforge.net/p/lemur/wiki/TheIndriQueryLanguage

besoin d'information, et ce qu'il faut éventuellement pondérer ce sont les termes qui appartiennent à ces concepts.

4.4.1 Exprimer une formule étendue

Les formules étendues que nous obtenons grâce aux concepts implicites et explicites contiennent des termes de nature différente. Alors que les termes obtenus par une ressource sémantique peuvent être des mots simples, des mots composés ou des expressions, ceux recueillis par LSI sont uniquement des mots simples qui n'ont pas une relation de synonymie directe, d'un point de vue humain. Tenir compte de la nature de ces termes est essentiel pour exprimer au mieux les concepts de la requête et donc le besoin d'information de l'utilisateur. Nous nous focalisons sur la proximité entre les mots qui composent ces termes, leurs relations avec d'autres termes au sein du même ensemble, et l'importance de chacun d'entre eux dans cet ensemble. Nous inclurons ces aspects dans le cadre du modèle MSL, à l'aide des opérateurs du tableau 4.2 proposés par le langage de requête d'Indri.

4.4.1.1 La proximité

Les termes composés de plusieurs mots impliquent un lien de proximité entre ces mots. Pour exprimer ce lien, il faut tenir compte de la distance entre les mots du terme et de l'ordre dans lequel ils apparaissent. Pour nos approches d'expansion, l'origine des termes composés de plusieurs mots est la ressource sémantique, où ces termes sont soit des entités nommées soit des expressions. Dans les deux cas, le lien entre les mots dans ces termes est très fort, nous exigeons que ces mots se trouvent dans la plus petite fenêtre possible dans le texte, ce qui peut être exprimé par les opérateurs #1 ou $\#uw1$ dans Indri. Bien que pour certains termes l'ordre des mots n'ait pas d'importance (comme les noms/prénoms des personnes), nous préférons garder le terme d'expansion tel qu'il se trouve dans l'ontologie, car nous supposons que c'est le rôle d'ontologie de fournir les différents alternatifs d'un terme. Pour cela, nous considérons qu'un terme d'expansion composé de plusieurs mots est un bloc qui doit être trouvé tel quel dans les documents. Nous utilisons donc l'opérateur #1 pour exprimer ces termes.

4.4.1.2 La synonymie

Quand les termes d'expansion d'une formule étendue sont des termes sémantiquement similaires dans une ressource sémantique, cela signifie qu'ils décrivent exactement le même objet. Ainsi, nous voulons, dans ce cas, qu'une occurrence de n'importe quel "synonyme" ait le même effet sur la fonction de mise en correspondance que n'importe quel autre synonyme du même terme. Ainsi, nous utiliserons l'opérateur $\#syn$ pour exprimer une formule étendue obtenue par un concept explicite, car nous considérons la similarité sémantique entre les termes d'un concept explicite comme une variation de la synonymie. Ce choix de ne pas pondérer ces

termes (ne pas utiliser l'opérateur #wsyn) est justifier par la section 4.4.1.3.
Par ailleurs, les termes issus des concepts implicites, n'ont pas les mêmes relations d'équivalence avec les termes de la requête que les concepts explicites. Notre utilisation d'une technique locale pour obtenir ces termes signifie que ceux-ci ont un lien de co-occurrence dans la collection de documents recherchée. Pour cela, nous n'utiliserons pas l'opérateur *#syn*, mais nous privilégierons la combinaison des probabilités d'occurrence de ces termes dans les documents, ce qui correspond aux opérateurs *#weight*, *#combine* ou *#wsum* dans le modèle MSL. Alors que *#combine* correspond à l'utilisation de *#weight* sans pondération ($w_i = 1$ dans l'équation 4.9), *#wsum* se diffère par la somme des croyances au lieu de la multiplication. Nos expériences ont montré une faible performance de cet opérateur par rapport à *#weight*, ce qui correspond également aux constats des auteurs du modèle MSL [Metzler and Croft, 2004]. Pour cela, nous utilisons l'opérateur #weignt qui permet de pondérer les termes provenants des concepts implicites comme on va expliquer dans la section suivante (Sect. 4.4.1.3).

4.4.1.3 La pondération

La pondération d'un terme dans une formule étendue représente son importance par rapport aux autres termes dans cette formule. Dans le cas de formules étendues provenant de concepts explicites, nous estimons que la relation de synonymie ne peut pas être graduée. En effet, une fois déclarés en tant que synonymes (ou similaires sémantiquement dans le cas des entités nommées), les termes trouvés ont le même poids dans leur formule étendue, car dire qu'un terme est plus synonyme qu'un autre serait dénué de sens. Par contre, d'un point de vue linguistique, les synonymes ne sont pas tous utilisés de la même manière dans le langage, car les synonymes de certains termes ne sont pas toujours bien connus. Pour cela, nous pourrions considérer les statistiques d'utilisation dans le langage comme un indicateur de l'importance d'un terme par rapport à l'ensemble des termes d'expansion. Nous choisissons de ne pas pondérer les termes d'expansion provenant d'une ressource sémantique, en supposant que la présence peu fréquente d'un terme dans les documents n'indique pas obligatoirement que ce dernier est moins pertinent qu'un autre contenant un synonyme plus récurrent.

Concernant les termes extraits des concepts implicites, nous estimons que l'importance d'un terme dans un concept implicite soit mesurée par sa similarité avec le terme original correspondant dans la requête. Selon notre approche d'expansion locale, le choix des termes est effectué selon leur proximité avec ceux de la requête dans l'espace sémantique. Ainsi, nous partons du principe que si l'utilisateur a choisi un terme pour exprimer un concept implicite dans sa requête, c'est ce terme qui a le plus d'importance pour lui, et en conséquence, il devrait être le plus influent sur le choix des documents pertinents. Les autres termes appartenant au même concept prennent leur importance selon leur similarité avec le terme de l'utilisateur. Pour cette raison, un terme d'expansion d'un concept implicite est pondéré dans la formule étendue selon sa similarité avec le terme original apparaissant dans la requête

et appartenant au même concept. Cette idée de pondération ne nécessite pas davantage de calculs, car les valeurs de similarité entre les termes sont obtenues lors du choix des termes d'expansion

4.4.2 Résumé

Nous avons présenté dans les sections précédentes comment utiliser les notions de proximité, de synonymie et de pondération en prenant en compte la nature et la provenance des termes d'expansion. Nous résumons dans cette section notre approche pour exprimer les formules étendues qu'on obtient par des concepts explicites ou implicites.

Une formule étendue $F_{C_E(q)}$ liée au terme q de la requête et provenant d'une concept explicite (C_E) est exprimé par 4.10,

$$F_{C_E(q)} = \#syn(t) \ : t \in C_E(t) \tag{4.10}$$

où t est un terme d'expansion qui appartient au concept C_E. Par ailleurs, une formule étendue $F_{C_I(q)}$ provenant d'un concept implicite (C_I) est exprimée par l'opérateur $\#weight$ comme dans l'équation 4.10.

$$F_{C_I(q)} = \#weight(w\ t) \ : t \in C_I(q), w = Dist_{euclide}(\overrightarrow{t}, \overrightarrow{q}) \tag{4.11}$$

où t est un terme d'expansion et w est la similarité entre t et q correspondante à la distance euclidiennes entre les vecteurs de ces termes dans l'espace produit par LSI (Sect. 4.3.2).

Pour une présentation plus démonstrative, nous prenons l'exemple de la requête suivante (*Trec 455*) :

```
#combine(Jackie Robinson appear first game)
```

L'utilisation de #combine avec des termes simples non pondérés correspond à la multiplication des probabilités $p(q_i|d)$ selon le modèle de vraisemblance de la requête. L'utilisation des ressources sémantiques donne les formules étendues suivantes, qui combinent les termes d'expansion par l'opérateur de synonymie en respectant la proximité pour les expressions :

```
#syn(#1(Jackie Robinson) #1(Jack Robinson))
#syn(appear look seem)
#syn(first #1(number one) #1(number 1)
game
```

où le sens choisi du mot « game » n'a pas d'alternatifs (par le lien « rdf :label ») dans Yago. Pour cela, sa formule étendue ne contient que lui. Avec l'approche locale, les termes étendus sont les suivantes (nous considérons 7 termes au plus par concept dans cet exemple) :

```
#weight(1.0 Jackie 0.87 Robinson 0.66 player 0.61 league
        0.54 good 0.52 year 0.50 field)
        first
```

```
#weight(1.0 game 0.83 play 0.811 team 0.78 season
        0.75 appear 0.73 ball)
```

Dans cet exemple, l'approche locale a regroupé les mots « Jackie » et « Robinson » dans la même formule étendue, ainsi que les mots « appear » et « game ». Par contre, l'approche n'a pas trouvé le terme « first » dans les documents de pseudo retour de pertinence, ce mot ne correspond donc pas à un concept implicite selon LSI, et considéré comme un représentant unique d'un concept de la requête.

La reformulation de la requête en utilisant ces formules étendues selon l'équation 4.9 donnera la requête suivante pour le cas d'expansion par des concepts explicites :

```
#weight(
  λ1 #combine(Jackie Robinson appear first game)
  λ2 #combine(
       #syn(#1(Jackie Robinson) #1(Jack Robinson))
       #syn(appear look seem)
       #syn(first #1(number one) #1(number 1)
       game
  )  )
```

Pour le cas d'expansion par des concepts implicites, nous obtiendrons la requête finale suivante :

```
#weight(
  λ1 #combine(Jackie Robinson appear first game)
  λ2 #combine(
       #weight(1.0 Jackie 0.87 Robinson 0.66 player 0.61 league
               0.54 good 0.52 year 0.50 field)
       appear
       #weight(1.0 game 0.83 play 0.811 team 0.78 season
               0.75 first 0.73 ball)
  )  )
```

Ainsi, on considère que nos approches génèrent des requêtes qui représentent d'une manière textuelle les concepts correspondant aux termes de l'utilisateur. Même si la similarité sémantique entre les termes obtenus par l'approche locale n'est pas la synonymie directe, la façon dont nous présentons et pondérons ces termes dans la requête reformulée permet de voir le degré de similarité entre chaque terme ajouté et le terme original qui lui correspond dans la requête.

4.5 ASMER :Approche Sémantique Mixte d'Expansion et de Reformulation

Dans cette section, nous proposons une Approche Sémantique Mixte d'Expansion et de Reformulation (ASMER). Celle-ci permet de modéliser une requête par une combinaison de concepts explicites et implicites. Nous explorons l'intégration des termes d'expansion provenant de deux ressources différentes dans la même requête. L'idée est de profiter du contexte apporté par la méthode locale et de la richesse sémantique apporté par une ontologie. Nous choisissons l'utilisation d'une ressource sémantique externe pour les entités nommées exclusivement. Cela signifie concrètement l'utilisation de l'approche Yago présentée dans la section 4.3.1.1 pour détecter les entités nommées dans la requête et pour extraire les différentes termes similaires à ces entités à partir des concepts correspondants dans Yago. Les autres termes de la requête, ceux qui ne sont pas des entités nommées, sont traités par l'approche LSI, où toute la requête est utilisée pour trouver les documents de pseudo retour de pertinence. Nous justifions ces choix d'expansion (ressource externe pour les entités nommées et locale pour les autres termes) par les points suivants :

— Selon notre étude de l'état de l'art sur l'expansion de la requête, nous avons constaté le manque de traitement des entités nommées. Les approches existantes ne prêtent pas une attention particulière à ces éléments. Pourtant, beaucoup d'études montrent leur importance pour les requêtes du Web.

— Contrairement aux termes d'expansion obtenus pour les entités nommées, qui sont souvent précis, un terme d'expansion pour un nom ou un verbe risque d'avoir plusieurs sens (appartenir à plusieurs synsets). C'est-à-dire que la désambiguïsation résout le problème du lien : terme de la requête/sens, mais elle ne garantit pas que le terme d'expansion choisi ne soit pas aussi utilisé dans d'autres sens. Nous pensons que cette possibilité est grande dans WordNet pour les termes qui ne sont pas des entités nommées, mais moins importante dans le cas des entités nommées.

— Même si les entités nommées ne sont pas considérées par l'expansion locale, elles y participent activement en étant une partie de la requête qui trouve les documents de pseudo retour de pertinence. Par ailleurs, les mots de la requête qui ne sont pas des entités nommées participent à la désambiguïsation des entités nommées par l'outil Aida en fournissant le contexte de cette entité.

Nous traitons les différentes questions qui peuvent apparaître en utilisant un tel mélange, durant les étapes de la génération des termes d'expansion et de la reformulation de la requête finale.

4.5.1 Génération des formules étendues

Même en distinguant les entités nommées des autres termes de la requête, on peut se retrouver avec des formules étendues qui contiennent des termes en commun. Pour cela, en plus des seuils de certitude et de spécificité que nous avons évoqués dans la section 4.3.3, nous ajoutons la condition de non redondance. C'est-à-dire que

pour être considéré comme un terme d'expansion, un terme ne doit pas apparaitre dans les formules étendues déjà construites de la requête.

Dans certains cas, c'est un mot de la requête (q_1) qui peut se trouver comme un terme d'expansion d'un autre terme de la requête (q_2). Dans le cas de concepts implicites, nous avons considéré ces termes (q_1 et q_2) comme des termes associés (Sect. 4.3.2), et nous les avons fusionné dans la même formule étendue. L'utilisation de concepts explicites n'apporte pas de changement à ce traitement. Car même si notre stratégie est de valoriser les termes originaux de la requête, nous considérons que la redondance a un effet négatif sur la requête en déséquilibrant l'importance des termes d'une façon non contrôlée. Ainsi, au moment de la prise en compte du mot q_1, si nous trouvons qu'il apparait en tant qu'expansion d'un terme q_2 déjà traité, q_1 ne sera pas exploré et sera éliminé de la requête étendue, il apparaîtra par contre en tant que terme d'expansion de q_2. Il est important de noter que la vérification de l'existence de la réplication de mots se passe d'une façon syntaxique, par contre, le cas des termes composés est bien pris en compte. C'est-à-dire qu'un mot est considéré comme unique dans la requête même s'il apparait dans un terme composé de plusieurs mots, mais il est redondant si nous le trouvons en tant que terme simple dans une formule étendue déjà construite. Par exemple, le mot « machine » sera traité et étendu comme un terme original unique dans la requête, même si cette requête a, dans l'une de ses formules étendues le terme « machine à laver ».

4.5.2 Expression de la requête finale

Pour intégrer des termes d'expansion de nature différente dans une requête finale, la modélisation sur laquelle nous nous sommes basés dans la section 4.4 est parfaitement adaptée au mélange des deux type de concepts, car il dépend des représentations sans mettre de contraintes sur le type de ces représentations. Nous reprenons ici l'équation pour la lier à l'utilisation de deux types de concepts.

$$p(Q|d) = \lambda \prod_{q \in Q} p(q|d) + (1 - \lambda) \prod_{r} b(r|d) \tag{4.12}$$

où on considère que Q est la requête originale de l'utilisateur composée de mots clés q, $p(q|d)$ est calculé selon le modèle de vraisemblance de la requête, et r est une représentation qui combine les termes d'expansions de la manière suivante : s'il s'agit d'une formule étendue d'une entité nommée, r est n'importe quel terme appartenant à cette formule (#syn). Il peut être un mot simple, ou une expression qu'on considère par l'opérateur (#1). Pour les autres termes de la requête qui ne sont pas des entités nommées, r est la combinaison pondérée des termes d'expansion obtenus par LSI, $b(r|d)$ est calculé par l'équation 4.13 dans ce cas.

$$b(r|d) = \prod_{i=1}^{k} p(e_i|d)^{w_i} \tag{4.13}$$

où e_i est un terme qui appartient à la formule étendue qu'on exprime. Cette équation correspond à l'utilisation de l'opérateur #weight en considérant la similarité par

rapport au terme original de cette formule étendue comme une pondération w_i (4.11).

4.5.3 Algorithme final

L'Algorithme 1 illustre les étapes principales de notre approche sémantique mixte d'expansion et de reformulation de la requête. Tout d'abord, nous commençons par enlever les mots vides de la requête avant d'obtenir les termes simples et composés selon StanfordNLP. La requête entière est lancée pour obtenir les documents de pseudo retour de pertinence, que nous utilisons pour générer la matrice U avec LSI (ligne 5). Pour chaque terme de la requête, nous créons une formule étendue qui contient par défaut le terme lui-même. Si le terme est une entité nommée, nous ajoutons à cette formule étendue les autres appellations de l'entité dans l'ontologie YAGO après la désambiguïsation par AIDA. Si le terme n'est pas une entité nommée, nous ajoutons à sa formule étendue les termes les plus proches de lui selon la matrice U. Dans tous les cas, l'ajout d'un terme à une formule étendue ne se fait que si ce terme satisfait les seuils de certitude et de spécificité et la condition de non redondance (lignes 11 et 18). Finalement, les formules étendues sont combinées en utilisant les fonctionnalités du modèle structurel de langue. Le résultat est à son tour combiné avec la requête originale d'une manière linéaire en utilisant le paramètre libre λ pour contrôler l'importance de la requête originale par rapport à la requête reformulée.

4.5.4 Caractéristiques d'ASMER

Bien que notre modèle ne soit pas le premier à mélanger des ressources pour la modification de la requête, il se distingue des autres par les caractéristiques suivantes :

- L'utilisation de ressources et de techniques de nature différente : la considération des concepts explicites et implicites pour les requêtes courtes du Web apporte une vision globale sur le sens de la requête. Ainsi, cette modélisation prend en compte des termes composés provenant d'une ontologie et des termes pondérés issus de la collection de documents. Notre modèle propose donc un mélange de ces termes assez différents à travers la reformulation de la requête.
- La considération spécifique des entités nommées : la prise en compte de ces entités par l'expansion de la requête a été limitée dans l'état de l'art. Notre approche attire l'attention sur l'importance de ces éléments et explore l'avantage de les étendre dans les requêtes du Web.
- La gestion explicite de la dérive de la requête : les termes choisis pour les formules étendues sont contrôlés par des mesures de certitude, de spécificité et de redondance pour éliminer les termes les moins avantageux pour la requête. De plus, notre modélisation permet de détecter le degré d'association entre deux mots de la requête, ce qui conduit à la fusion des formules étendues

Input:
Q : Requête originale
α_1 : Seuil de certitude
α_2 : Seuil de spécificité
Output:
Q' : Requête reformulée

begin
  $Q = FiltreMotsVides(Q)$;
  $T = EnsembleTermes(Q, StanfordNLP)$;
  $D = EnsembleRetourDePertinence(Q)$;
  $U = LanceLSI(Q, D)$;
  $Q' = Q$;
  foreach $q \in T$ **do**
    $F(q) = q$;
    if $q \in Yago$ **then**
      foreach $t \in EtendreOntololgie(Q, Aida, Yago)$ **do**
        if $Certitude(t) > \alpha_1, Spcificit(t) > \alpha_2, !Redondante(t, Q')$ **then**
          $F(q) += (Certitude(t)\ t)$
        end
      end
    end
    else
      foreach $t \in EtendreLSI(U)$ **do**
        if $Certitude(t) > \alpha_1, Spcificit(t) > \alpha_2, !Redondante(t, Q')$ **then**
          $F(q) += t$
        end
      end
    end
    foreach $q \in T$ **do**
      if $q \in Yago$ **then**
        $Q' = Remplace(q, \#syn(F(q)))$;
      end
      else
        $Q' = Remplace(q, \#weight(F(q)))$;
      end
    end
  end
  $Q' = \lambda \#combine(Q) + (1 - \lambda)\#combine(Q')$;
  return Q'
end

Algorithm 1: L'algorithme du modèle ASMER

provenant de termes originaux fortement liés.

— La simplicité d'interprétation et de modification par les utilisateurs : beaucoup d'approches d'expansion de la requête sont très efficaces au niveau des résultats, mais incompréhensibles si elles sont montrées à l'utilisateur. Notre approche est centrée utilisateur. En effet, nous reformulons la requête étendue de façon à ce qu'il puisse faire facilement le lien avec les termes originaux toujours présents dans les formules étendues. Ce qui permet à l'utilisateur de pouvoir gérer l'étude de ces requêtes ou les modifications au niveau de la structure ou des termes dans un environnement interactif.

— La portabilité : dans un cadre de recherche d'information compatible avec une ontologie générale, notre approche de l'étape d'expansion de la requête est compatible avec n'importe quel modèle de recherche, et ne nécessite pas un accès à l'index ou aux statistiques de la collection de documents. Bien que la reformulation ait été présentée pour le modèle structuré de langue dans notre étude, le principe peut être exploitable pour un autre modèle, à condition qu'il fournisse des fonctionnalités et des opérateurs qui permettent d'exprimer la proximité, la synonymie, et la pondération des termes.

4.6 Résumé

Nous avons proposé deux approches d'expansion de la requête. La première approche est basée sur l'ontologie Yago, et la deuxième utilise le principe LSI sur un ensemble de documents de pseudo retour de pertinence. Ces approches cherchent les concepts (explicites ou implicites) de la requête. Une fois ceux-ci identifiés, les deux approches cherchent des relations de similarité entre les termes originaux de la requête et les termes sémantiquement liés selon la ressource utilisée. Dans les deux approches, nous savons exactement à quel terme original appartient une formule étendue. Avec cette information, nous cherchons à construire la version conceptuelle de la requête originale tout en respectant les choix des termes faits par l'utilisateur pour exprimer son propre besoin d'information.

L'approche sémantique mixte ASMER que nous avons proposée par la suite, combine les deux approches précédentes tout en considérant le cas des requêtes Web qui sont courtes, ambiguës et comprennent souvent des entités nommées. À travers cette approche, nous avons modélisé l'intégration des formules étendues de nature et de source différentes toujours en respectant les termes originaux de l'utilisateur.

ASMER se différencie des approches précédentes de l'état de l'art en plusieurs points. Sur l'aspect fondamental, elle est centrée utilisateurs : les termes originaux de l'utilisateur sont valorisés dans la requête reformulée, et la structure de la requête finale lui permet de bien voir et d'interpréter le comportement de la modélisation. Sur les autres aspects, notre approche valorise les entités nommées en leur donnant la priorité lors de l'expansion par rapport aux autres termes de la requête. Ce point signifie également que notre approche considère éventuellement des termes d'expansion composés. Finalement, ASMER explore une représentation de la requête autre

que le simple sac de mots, elle propose une modélisation compréhensible qui permet de mélanger deux types de termes d'expansion dans une seule requête conceptuelle. La performance de cette modélisation et des différents aspects présentés dans ce chapitre sera étudiée dans la partie expérimentale (Chap. 6), mais le problématique d'évaluation du rappel est d'abord étudiée dans le chapitre suivant (Chap. 5).

Chapitre 5

Évaluation du rappel en recherche d'information

Sommaire

5.1 Introduction

Ce chapitre comprend notre contribution concernant l'évaluation du rappel. La motivation de cette partie de la thèse est née lorsque nous avons posé la question suivante : comment savoir si notre approche de reformulation de la requête est performante ? Dans un contexte de recherche sur le Web, la réponse à cette question est l'utilisation des métriques de précision comme le *MAP*. Par contre, au début, notre travail de recherche n'était pas précisément orientée vers le monde du Web, et nous étions intéressés à comprendre le comportement des approches d'expansion à travers plusieurs aspects, y compris le rappel. Nous avons donc constaté que l'évaluation du rappel est une problématique dans certaines applications de recherche d'information où la précision n'est pas la priorité.

Ce que nous proposons dans ce chapitre est une nouvelle mesure qui permet d'évaluer la capacité d'un système à trouver le plus de documents pertinents sans pour autant négliger leur ordre d'apparition. Nous commençons par un aperçu de l'évaluation en recherche d'information, suivi par un état de l'art sur l'évaluation du rappel. Dans la section 5.4 nous proposons la mesure orienté rappel (*MOR*) qui répond aux contraintes que nous avons définies, et qui ne sont pas remplies de façon satisfaisante par les mesures précédentes. Les caractéristiques et l'évaluation de cette mesure sont proposées dans les sections 5.5 et 5.6 respectivement. Nous terminons ce chapitre par un résumé dans la section 5.7.

5.2 Évaluation en recherche d'information

L'évaluation des systèmes de recherche d'information peut être abordée selon deux angles : l'efficience et l'efficacité. L'efficience s'intéresse au temps de réponse et l'espace occupé par le système de recherche lors de l'exécution. Alors que l'efficacité, peut être évaluée par plusieurs mesures, comme la facilité d'utilisation ou la présentation des résultats. D'un point de vue académique, la capacité d'un système à sélectionner des documents pertinents est l'aspect le plus important à évaluer. Plusieurs métriques existent pour mesurer cette capacité, mais pour pouvoir les utiliser, il faut avoir accès aux jugements de pertinence, c'est-à-dire connaitre les documents pertinents pour chaque requête. Dans les sections suivantes, nous parlons de l'évaluation de la pertinence puis de l'évaluation des requêtes. Nous poursuivrons par un peu d'histoire sur les campagnes d'évaluation pour finir par l'évaluation des approches d'expansion de la requête.

5.2.1 Évaluation de la pertinence

En recherche d'information, le même document peut être jugé diversement par deux utilisateurs, ou dans deux conditions différentes. De plus, ce jugement n'est pas nécessairement booléen, un utilisateur peut estimer un document plus pertinent qu'un autre, même si ce document est aussi considéré comme pertinent par le même utilisateur. Ce dernier aspect peut être influencé également par l'ordre dans lequel

l'utilisateur exploite les documents : un document correspondant à une requête peut perdre sa valeur s'il n'apporte pas d'information supplémentaire par rapport à un document pertinent que l'utilisateur a déjà vu. Tous ces éléments font que l'évaluation idéale de la pertinence d'un document doit être faite par l'utilisateur qui a construit la requête durant la même session de recherche, ce qui n'est pas le cas dans la plupart des campagnes d'évaluation comme on va le voir dans la section 5.2.3.

5.2.2 Évaluation des requêtes

Une fois que les jugements de pertinence pour chaque requête sont disponibles, l'évaluation de requêtes peut être faite en utilisant une variété de métriques d'évaluation. Ces métriques d'évaluation sont des mesures objectives basées sur la notion de la pertinence, leur efficacité est donc liée à la qualité des jugements de pertinence. Avec les valeurs d'évaluation des requêtes individuelles pour une métrique donnée, la performance du système est souvent estimée en prenant la moyenne de ces valeurs individuelles pour cette métrique. La plupart des métriques proposées

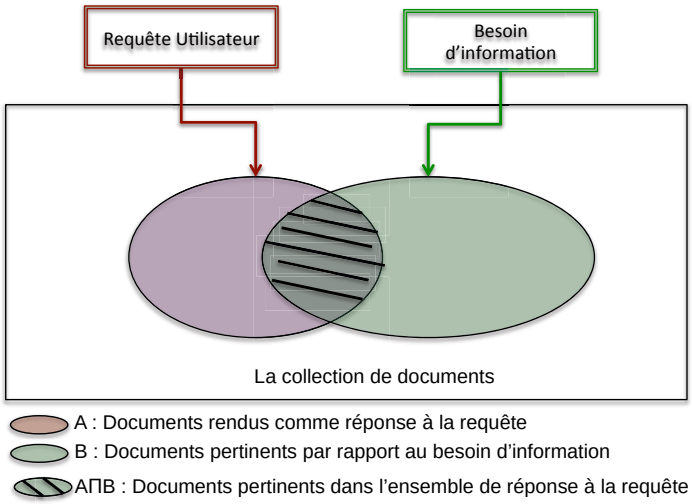


FIGURE 5.1 – Ensembles de documents par rapport au besoin d'information et à la requête

en recherche d'information se basent sur deux notions historiques : la précision et le rappel [Cleverdon et al., 1966; Jones, 1981]. Le calcul direct de ces deux aspects se fait en utilisant les équations 5.1 et 5.2 .

$$précision(q) = \frac{|A \cap B|}{|A|} \tag{5.1}$$

$$rappel(q) = \frac{|A \cap B|}{|B|} \tag{5.2}$$

où A est l'ensemble des documents trouvés par la requête q, et B est l'ensemble de documents pertinents pour cette requête (Fig. 5.1). Dans le cas de la recherche de documents dite ad hoc, les documents dans l'ensemble A sont présentés à l'utilisateur comme une liste de documents ordonnés selon leur pertinence pour la requête. Dans le cas idéal, on trouve tous les documents pertinents au début de la liste de résultats ($précision = rappel = 1$).
La précision et le rappel peuvent être combinés de plusieurs façons. Nous citons ici quelques exemples mais des détails sur d'autres métriques peuvent être trouvés dans la section 5.3.2.

— *AP* : pour une requête q, la précision moyenne, notée AP (*Average Précision*), est la moyenne des valeurs de précision à chaque position d'un document pertinent de la liste de résultats, elle est calculée par l'équation suivante :

$$AP = \sum_{x<|Ren(q)|} \frac{Per(d_x, q) \cdot P_{@x}(q)}{n} \tag{5.3}$$

où $Ren(q)$ est la liste de documents rendus pour la requête q, n est le nombre de documents pertinents dans la collection, $P_{@x}(q)$ est la précision à la position x de la liste de résultats, et $Per(d_x, q)$ est la valeur booléenne de la pertinence entre la requête q et le document d à la position x, elle prend la valeur 1 si le document est pertinent, sinon 0.

— *R-Précision* : Il s'agit de la précision à la position R de la liste de résultats, où R est le nombre de documents pertinents dans la collection pour ce besoin d'information.

— La courbe Rappel-Précision : pour une requête donnée, on calcule la précision à la position de chaque document pertinent trouvé dans la liste de résultats. Par contre, pour pouvoir représenter la précision moyenne de l'ensemble des requêtes, les niveaux de rappel pour lesquels on calcule la précision moyenne doivent être unifiés. Généralement, on retient 11 points de rappel standard, de 0 à 1 par pas de 0,1. Les valeurs de précision non obtenues à partir des valeurs de rappel sont calculées par interpolation comme suit : pour deux points de rappel, i et j, $i < j$, si la précision au point i est inférieure à celle au point j, on dit que la précision interpolée à i égale la précision à j.

5.2.3 Les campagnes d'évaluation

Les mesures d'évaluation permettent de comparer les systèmes de recherche d'information (SRI), à condition que la comparaison soit faite sur le même jeu de données. Plusieurs projets basés sur des corpus d'évaluation existent depuis les années 70. La campagne d'évaluation la plus connue TREC (Text REtrieval Conference) a commencé en 1992 dans le but d'encourager la recherche documentaire basée sur de grandes collections de tests. Pour chaque session de ce genre de campagnes, un ensemble de documents et de requêtes est fourni. Les participants exploitent leurs systèmes de recherche sur les données et renvoient leurs listes ordonnées de documents pour chaque requête à la campagne. Ce principe d'évaluation a été pratiqué

dès 1966, où Cleverdon et son équipe du projet Cranfield ont jugé une collection de 1 400 papiers scientifiques sélectionnés parmi 18 000, pour pouvoir évaluer le rappel et la précision [Cleverdon, 1962; Cleverdon et al., 1966]. Contrairement aux essais de Cleverdon, et à cause des grandes tailles des collections et du coût élevé du jugement humain, dans les campagne d'aujourd'hui, uniquement les K documents les mieux classés des différents SRI participants sont montrés aux évaluateurs qui décident finalement de la pertinence de chaque document. Dans tous les cas, les participants disposent finalement d'une liste de jugements pour chaque requête et peuvent ainsi évaluer les performances de leurs SRI respectifs.

Il existe également d'autres campagnes d'évaluation pour des contextes différents. On note en particulier la campagne INEX (*INitiative for Evaluation of XML Retrieval*) lancée en 2002, et destinée à construire des collections spécifiques pour évaluer les travaux en RI sur des documents XML. On trouve aussi la campagne CLEF (*Conference and Labs of the Evaluation Forum*) qui s'intéresse entre autres aux aspects multi-lingues dans les systèmes de recherche d'information, et la campagne NTCIR (NII Testbeds and Community for Information access Research) qui contient des tâches dédiées à la recherche d'information en langues asiatiques.

De notre point de vue, le problème principal de ces campagnes d'évaluation réside dans l'évaluation de la pertinence dans certains tâches (*tasks*) comme la tâche ad hoc. Cela vient du fait que les individus qui évaluent les documents ne sont pas forcément des experts sur le sujet en question, contrairement à l'ancien projet de Cranfield, où les jugements de pertinence ont été réalisés par la personne qui avait formulé la requête. Ce problème est difficile à résoudre vu le nombre de requêtes et de documents à évaluer. La solution utilisée aujourd'hui pour surmonter ce problème est de formaliser les besoins en information en plusieurs parties (par exemple : titre, narrative et description), fournissant ainsi le plus de détails possible sur l'intention du rédacteur du besoin d'information.

5.2.4 Évaluation et l'expansion de la requête

Les mesures d'évaluation en recherche d'information s'appliquent à n'importe quel système qui rend une liste ordonnée de documents, quelque soit la technique que ce système utilise. Par contre, il y a des cas où ces mesures doivent être appliquées en considérant la spécificité de la technique utilisée par un SRI, c'est le cas des techniques d'expansion de la requête basées sur une approche locale. En 1967, Hall and Weiderman [1967] ont présenté dans leur rapport les problèmes liés à l'évaluation des techniques d'expansion de la requête utilisant le retour de pertinence. Ils distinguent deux effets causés par cette technique « *Ranking Effect* » et « *Feedback Effect* ». Le premier effet est constaté quand les documents jugés par l'utilisateur lors d'un processus de retour de pertinence ne sont pas exclus de la collection avant le lancement de la nouvelle requête étendue. Les termes d'expansion qu'on extrait de ces documents jugés sont déjà intégrés dans la requête. Par conséquent, ces documents vont souvent avoir un meilleur score pour la deuxième itération. Les résultats des expériences qui prennent en compte cet effet pour l'évaluation finale de per-

formance vont avoir une large amélioration par rapport à la requête initiale. Le problème dans ce cas, selon Hall and Weiderman [1967], est que l'utilisateur est intéressé de voir l'effet de son retour de pertinence plutôt que d'avoir de nouveau les mêmes documents qu'il a jugés, ce qu'ils appellent « Feedback Effect ». Pour cela, il est intéressant de mettre de côté (bloquer) les documents jugés pour toute nouvelle itération. Sauf que cette pratique va pénaliser le système dans la deuxième itération : un bon système qui trouve beaucoup de documents pertinents dès la première itération n'en trouvera forcément moins pour la requête reformulée. Ce problème est souvent ignoré ou non précisé dans les travaux d'expansion de la requête que nous avons examinés. Notre position par rapport à ce problème est de ne pas enlever les documents de retour de pertinence de la collection lors de la deuxième itération, surtout que l'utilisation d'une technique de pseudo retour de pertinence élimine le « *Feedback Effect* », car l'utilisateur ne juge et ne voit même pas les documents utilisés pour le retour de pertinence. De plus, cette technique permet d'évaluer toutes les approches d'expansions (locales ou non) dans leur intégralité (comme des boites noires), sans manipuler la collection de documents par rapport à la spécificité des approches locales.

5.3 Problématiques du rappel

Comment évaluer le rappel ? Quand on affirme avoir amélioré le rappel d'un système, quelle mesure faut-il utiliser pour justifier cette amélioration ? Malgré la simplicité et la clarté de ces questions, la réponse n'est pas simple. Alors que la précision est une notion facile à évaluer, car elle ne demande aucune information sur le nombre total de documents pertinents dans la collection, le rappel est une notion qui pose plusieurs problèmes. Il est d'ailleurs impossible de l'évaluer dans les cas réels où on ne connait pas à l'avance le nombre de documents correspondant à sa requête. Contrairement à la précision, l'utilisation de la notion pure du rappel (équ. 5.2), n'est pas une mesure correcte, car le fait de rendre toute la collection de documents va garantir d'obtenir le rappel maximal. De plus, cette mesure seule n'est pas capable de distinguer deux systèmes qui fournissent le même nombre de documents pertinents dans des positions différentes de la liste des résultats.

5.3.1 Contexte

Pour pouvoir étudier le rappel, nous nous intéressons aux situations où l'utilisateur souhaite trouver le maximum de documents pertinents, comme dans les cas de recherche de brevets et de dossiers médicaux. La priorité de l'utilisateur dans ces contextes est le rappel. Pour cette raison, il est prêt en général à évaluer plusieurs pages de résultats pour sa requête, contrairement aux contextes de recherche dirigé précision comme la recherche sur le Web. Ainsi, nous considérons qu'une mesure orientée rappel doit satisfaire les deux besoins suivants :

— **B1** : Pouvoir favoriser un système qui rend plus de documents pertinents.

	Rang doc. pert.	AP	P	Recall	F_1	F'_1	F'_4
Système 1	{1, 2, 3, 4}	1	0,04	1	0,0769	1	1
Système 2	{50, 51, 53, 54}	0,0474	0,04	1	0,0769	0,0917	0,4586
Système 3	{1, 98, 99, 100}	0,2727	0,04	1	0,0769	0,429	0,864
Système 4	{1,54}	0,259	0,02	0,5	0,0385	0,3414	0,4741
Système 5	{1}	0,25	0,01	0,25	0,0192	0,25	0,25

TABLE 5.1 – Les différentes mesures d'évaluation pour cinq exemples

— **B2** : Pour deux systèmes qui rendent le même nombre de documents pertinents, pouvoir favoriser celui qui les donne plus tôt que l'autre dans sa liste de résultats.

Les sections suivantes aborderont les mesures d'évaluation dans l'état de l'art en considérant ce contexte. Pour mieux illustrer le comportement des différentes mesures, nous montrons dans le tableau 5.1 une extension de l'exemple proposé par Magdy and Jones [2010] : supposons avoir cinq systèmes de recherche d'information que nous évaluons par rapport aux réponses rendues par une seule requête q qui rend 100 documents, en sachant que le nombre de documents pertinents dans la collection pour la requête q est 4. Les systèmes sont ordonnés d'une façon décroissante par rapport à un jugement humain : le système 1 est le meilleur pour un utilisateur intéressé au rappel, alors que le système 5 est le pire.

5.3.2 Mesures d'évaluation traditionnelles

Pour considérer le rappel, une extension de la moyenne harmonique de la précision et du rappel peut être utilisée, ou ce que l'on appelle la mesure F_β présentée par l'équation 5.4. Le paramètre β dans cette équation permet d'équilibrer l'importance du rappel par rapport à celui de la précision.

$$F_\beta = \frac{(1+\beta^2)P \cdot R}{\beta^2 \cdot P + R} \tag{5.4}$$

Du tableau 5.1, on peut constater que la mesure F_β (avec β=1) ne distingue pas la performance des systèmes 1, 2 et 3. Elle n'est donc pas capable d'ordonner les systèmes par rapport aux rangs des documents trouvés quand le nombre de documents pertinents trouvés est identique. Pour résoudre ce problème, une extension de cette mesure utilisant la précision moyenne au lieu de la précision pure a été proposée, dans ce cas la mesure est appelée F'_β (équ. 5.5).

$$F'_\beta = \frac{(1+\beta^2)AP \cdot R}{\beta^2 \cdot AP + R} \tag{5.5}$$

En observant le tableau 5.1, on peut constater que l'utilisation de AP au lieu de P a résolu le besoin B2, c'est-à-dire que F' est capable de distinguer les systèmes par rapport au rang des documents pertinents trouvés. Par contre, la mesure F' a largement favorisé les systèmes qui ont trouvé des documents tôt dans la liste

de résultats par rapport aux autres qui ont détecté plus de documents pertinents. Par exemple, le système 4 qui ne rend que deux documents pertinents obtient des scores F_1' et F_4' plus élevés que ceux obtenus par le Système 2 qui a rapporté tous les documents pertinents. Cela s'est produit, car le premier document fourni par le Système 4 est pertinent alors qu'il arrive au rang 50 dans le cas du système 2. La mesure F' n'est donc pas capable de répondre au besoin B1 même en donnant au rappel un poids quatre fois plus élevé qu'à la précision dans l'équation 5.5.

Une autre mesure largement utilisée dans le domaine de la recherche d'information est le *MAP* (*Mean Average Precision*) qui est calculé par l'équation 5.6.

$$MAP = \frac{\sum_q AP}{N_q} \tag{5.6}$$

où N_q est le nombre de requêtes jugées et *AP* (*Average Precision*) est la précision moyenne pour une requête q selon l'équation 5.3. Le tableau 5.1 nous permet de voir clairement que le *MAP*, qui est aussi le *AP* dans notre exemple d'une seule requête, n'est pas la mesure adaptée à notre contexte. Par exemple, selon le *MAP*, le Système 5 est cinq fois meilleur que le Système 2 même si ce dernier a réussi à rendre tous les documents pertinents.

D'autres mesures d'évaluation existent dans le domaine de la recherche d'information. La plupart de ces métriques mesurent, comme le *MAP*, la capacité d'un système à retrouver des documents pertinents tôt dans la liste. Parmi ces alternatives, on peut nommer le *GMAP* (*Geometric Mean Average Precision*) [Robertson, 2006] qui est la moyenne géométrique des AP, et le *MRR* (*Mean Reciprocal Rank*) [Voorhees and Tice, 1999] qui est utilisé lorsque l'utilisateur n'a besoin que d'un seul document, il s'agit de l'inverse du rang du premier document pertinent trouvé. Par ailleurs, certaines mesures considèrent le degré de pertinence des documents trouvés, comme la mesure *NDCG* (*Normalized Discounted Cumulative Gain*) [Järvelin and Kekäläinen, 2002]. *NDCG* récompense ou pénalise le score d'évaluation en prenant compte de le niveau de pertinence d'un document et le rang dans lequel il se trouve dans la liste des résultats. Quand on ne dispose pas de jugements de pertinence gradués, cette mesure a une corrélation très forte avec la précision moyenne [Sakai, 2007], elle n'est donc pas considérée comme une mesure d'évaluation du rappel. Une autre mesure peut paraître adaptée à une recherche orientée rappel est le R-Prec (Sect. 5.2.2), où R est le nombre de documents pertinents dans la collection pour ce besoin d'information. Cette mesure reste une variante de la précision pure à un rang précis dans la liste de résultats, donc elle n'est pas adaptée à un contexte de rappel. Par exemple, avec R qui vaut 4 dans notre exemple précédent, le Système 2 qui a rendu tous les documents pertinents va obtenir un score R-Prec de 0, alors que le R-Prec du système 5 vaut $0,25$ bien que le Système 5 n'ait rendu qu'un seul document pertinent.

5.3.3 Mesures basées sur le Rappel Normalisé

Nous avons mentionné, jusqu'à présent, les mesures traditionnelles en recherche d'information les plus susceptibles d'être adaptées à un contexte orienté rappel. Nous avons illustré, en utilisant des exemples simples, que ces mesures ne sont pas capables de correspondre au jugement humain dans notre contexte. Pour cette raison, plusieurs travaux ont essayé récemment d'étudier ce problème [Zobel and Park, 2009; Magdy and Jones, 2010; Magdy, 2012; Webber, 2010]. On trouve dans la littérature certains papiers remettant en question la notion du rappel elle-même, comme le papier de Zobel and Park [2009] qui critique l'ambiguïté de l'utilisation du rappel à cause de son lien avec plusieurs aspects, comme la « totalité », c'est-à-dire la capacité de trouver la totalité de documents pertinents, et la « persistance » qui indique la volonté de l'utilisateur examiner des documents aux rangs avancés dans la liste de résultats. Nous soulignons que nous nous intéressons à un contexte de totalité, où la priorité de l'utilisateur est de trouver, au mieux, tous les documents pertinents pour sa requête, car cette signification correspond le mieux à la définition originale du rappel, et c'est l'évaluation du rappel dans ce sens qui est le plus problématique. D'autres études proposent de nouvelles mesures pour évaluer le rappel dans le contexte qui nous intéresse. Le travail le plus récent à notre connaissance est le papier de Magdy et Jones Magdy [2012] qui présente la mesure *PRES*. Dans leur étude, les auteurs proposent de reprendre le rappel normalisé (*RNorm*) [Rijsbergen, 1979; Rocchio, 1964] et de l'adapter pour qu'il soit utilisable avec les collections de documents de grande taille. Les deux mesures *RNorm* et *PRES* se basent sur l'idée de définir un meilleur et un pire scénario pour un processus de recherche (Fig. 5.2). Dans le meilleur scénario (S1), tous les documents pertinents sont trouvés au début

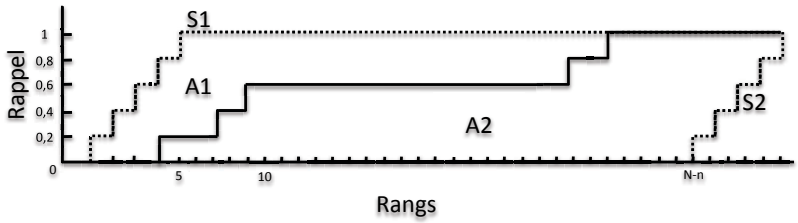


FIGURE 5.2 – Meilleur et pire scénarios de la mesure *RNorm*

de la liste de résultats, dans le pire scénario (S2) ils sont rendus à la fin de la liste à partir de la position $N - n$, où N est le nombre de documents dans la collection et n est le nombre de documents pertinents. Comme présentée par la figure 5.2, la courbe créée par un cas de recherche réel divise la surface entre S1 et S2 en deux parties : A1 et A2. Ainsi, l'idée du rappel normalisé est de calculer le pourcentage de la surface A2 par rapport à la surface totale entre S1 et S2, ce qui se résume par l'équation 5.

$$RNorm = \frac{A2}{A1 + A2} \tag{5.7}$$

	Rang de docs. perts	AP	Recall	F_1	F'_4	*PRES*
Système 1	{1, 2, 3, 4}	1	1	0,0769	1	1
Système 2	{50, 51, 53, 54}	0,0474	1	0,0769	0,0917	0,5
Système 3	{1, 98, 99, 100}	0,2727	1	0,0769	0,429	0,28
Système 4	{1,54}	0,259	0,5	0,0385	0,3414	0,37
Système 5	{1}	0,25	0,25	0,0192	0,25	0,25

TABLE 5.2 – La mesure *PRES* comparée aux mesures basées sur *AP*.

Alors que cette mesure semble un bon candidat pour le contexte de recherche d'information dirigée rappel, son défaut majeur est que l'on a besoin de juger toute la collection de documents par rapport à une requête, ce qui est impossible avec la taille des collections d'aujourd'hui. Pour cette raison, la mesure *PRES* prend en compte un seuil $Nmax$ qui représente le nombre maximum de documents que l'utilisateur a l'intention de juger au lieu de la taille totale de la collection N (Fig. 5.3).

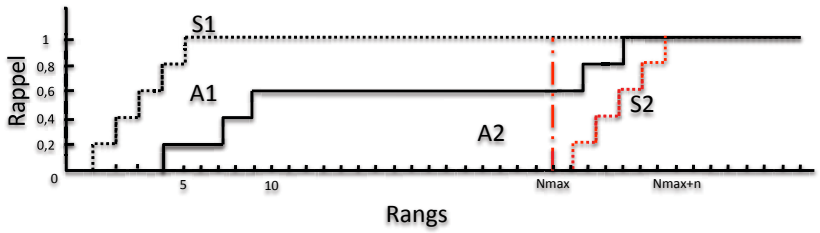


FIGURE 5.3 – Meilleur et pire scénarios de la mesure *PRES*

Cette extension de l'idée ne modifie pas la définition du meilleur scénario pour $RNorm$, alors que le pire scénario sera le fait que le système trouve tous les documents pertinents après la position $Nmax$, ce qui signifie que l'utilisateur ne les verra pas. La mesure *PRES* est exprimée par l'équation suivante :

$$PRES = 1 - \frac{\frac{\sum r_i}{n} - \frac{n+1}{2}}{N_{max}} \tag{5.8}$$

où n est le nombre de documents pertinents, $\frac{\sum r_i}{n}$ est la moyenne des rangs des documents pertinents trouvés avant le rang $Nmax$ si tous les documents pertinents ont été trouvés avant ce rang, sinon $\sum r_i$ est calculé par l'équation 5.9 :

$$\sum r_i = \left(\sum_{1<i<nR} r_i \right) + nR(N_{max} + n) - \frac{nR(nR-1)}{2} \tag{5.9}$$

où R est le rappel à N_{max} [1].

En ajoutant la mesure PRES à notre tableau démonstratif (Table. 5.2), on constate

1. Ce qui revient à dire que nR est le nombre de documents pertinents rendus avant $Nmax$.

que la mesure *PRES* a été capable de corriger les défauts des mesures traditionnelles précédentes. Contrairement à la mesure F', le score *PRES* ne donne pas plus d'importance aux documents repérés tôt dans la liste par rapport au nombre de documents pertinents trouvés, c'est pour cette raison que le Système 4 n'est plus privilégié par rapport au Système 2 malgré le document pertinent qu'il a détecté en début de liste. Par contre, *PRES* pénalise le fait de trouver des documents pertinents à la fin de la liste. Par exemple, *PRES* estime que le Système 3 est moins bon que le Système 4, car le premier a repéré des documents à la fin de la liste, et cela a baissé son score bien qu'il ait trouvé tous les documents pertinents. Ce comportement peut être expliqué par la figure 5.4 où on constate que la surface créée par un système (A) peut être inférieure à celle créée par un autre système (B) qui trouve moins de documents pertinents.

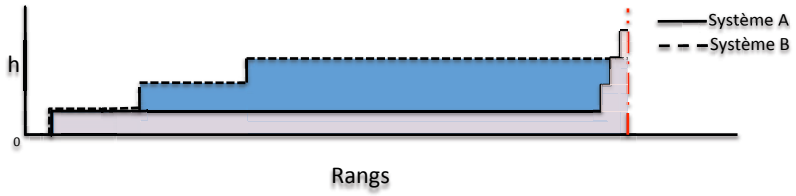


FIGURE 5.4 – La surface crée par les performances des systèmes A et B

PRES est une mesure conçue pour la recherche de brevets. Elle est censée prendre en compte d'un côté le rappel et de l'autre côté l'effort que l'utilisateur va fournir pour obtenir le maximum de résultats. Les auteurs voient ce comportement de *PRES* comme un outil permettant de mesurer l'effort que l'utilisateur doit fournir pendant le processus de recherche : tant que l'utilisateur trouve des documents loin dans la liste, il va continuer de vérifier plus de documents, ce qui signifie plus d'efforts, induisant donc un score faible.

À notre avis, l'hypothèse qui suppose que l'utilisateur préfère trouver moins de documents pertinents plutôt que d'en détecter plus loin dans la liste de résultats est une hypothèse discutable dans un contexte de recherche dirigé rappel. L'effort de l'utilisateur qui est l'élément justifiant le comportement de *PRES* est une notion subjective dépendant fortement de l'utilisateur lui-même, mais aussi du contexte de la recherche. Dans une recherche de haute persistance [Zobel and Park, 2009], surtout avec un système rendant peu de résultats, l'utilisateur est généralement prêt à examiner plusieurs pages de résultats indépendamment des rangs des documents pertinents trouvés, car dans la réalité, il ne connaît pas par avance le nombre total de documents pertinents pour sa requête. De plus, le $Nmax$ est considéré comme le nombre de documents que l'utilisateur souhaite juger. Dans un cas de recherche de faible persistance, l'utilisateur peut choisir une petite valeur de $Nmax$ pour exprimer explicitement le nombre de documents qu'il souhaite voir.

5.3.4 Bilan

Dans cette section, nous avons évoqué le sujet de l'évaluation en recherche d'information, en nous focalisant sur le problème de l'évaluation du rappel. En définissant les besoins à remplir lors d'une évaluation dirigée rappel, et à l'aide d'exemples simples, nous avons constaté que la majorité des mesures de l'état de l'art ne remplissent pas ces besoins, car elles sont fortement influencées par les documents pertinents trouvés en début de liste ou alors pénalisent les documents repérés à la fin de la liste, ce qui, dans les deux cas perturbe leur capacité de détecter les systèmes qui ont un meilleur rappel.

5.4 *MOR* : une Mesure Orientée Rappel

Pour définir une nouvelle mesure, il faut avoir une vision claire du comportement qu'elle doit posséder. Afin d'être le plus proche possible d'un jugement humain sur la préférence d'un système par rapport à un autre dans un contexte dirigé rappel, nous cherchons à construire une mesure d'évaluation qui répond aux contraintes bien définies en fonction de cet objectif. Nous commençons par définir les paramètres qui nous permettent à formaliser ces contraintes et à introduire notre mesure.

5.4.1 Définition des paramètres

Nous avons constaté dans l'état de l'art que la courbe rappel/rang, proposée à l'origine pour le rappel normalisé [Rijsbergen, 1979; Rocchio, 1964] et utilisée par Magdy and Jones [2010], est une bonne base pour une mesure orientée rappel. Nous reprenons donc cette courbe dans la figure 5.5 en remplaçant le rappel par le nombre de documents pertinents pour faciliter le développement de notre mesure.

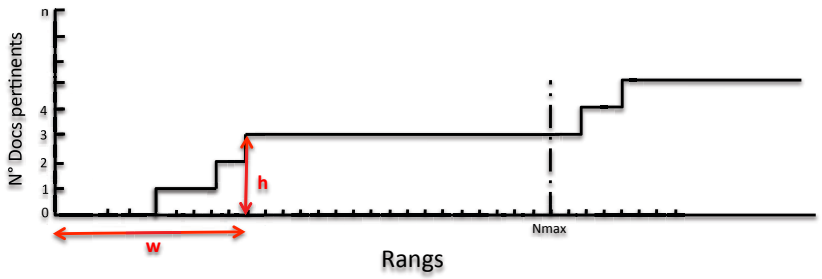


FIGURE 5.5 – Les notions de hauteur et de largeur selon la mesure *ROM*

Cette courbe a l'avantage de donner une vision claire de la propagation des documents pertinents dans la liste de résultats, ce qui permet de bien visualiser le rappel et la précision. Pour bien contrôler ces deux notions dans notre mesure, nous définissons les paramètres suivants :

— La hauteur (h) : est le nombre de documents pertinents trouvés avant le rang $Nmax$.
— La largeur (w) : est le rang auquel nous trouvons le dernier document pertinent avant $Nmax$.
— AP_{N_x} : est la précision moyenne à $Nmax$. En appliquant l'équation 5.3, on peut calculer AP_{N_x} par l'équation 5.10.

$$AP_{N_x} = \sum_{x<min(|Ren(q)|,N_{max})} \frac{Per(d_x, q) \cdot P_{@x}(q)}{n} \tag{5.10}$$

Pour favoriser un système qui pour le même rappel trouve les documents plus tôt dans la liste, w seul ne suffit pas, car pour le même w, deux systèmes peuvent trouver des documents pertinents à des rangs bien différents. Par ailleurs, l'utilisation de la précision moyenne (ou de la précision totale) nous ramène au problème d'être fortement influencé par les documents pertinents trouvés en début de liste. Pour ces raisons, nous considérons qu'une combinaison de w et de AP_{N_x} est indispensable pour juger la précision, alors que pour le rappel, nous estimons qu'il est représenté par le nombre de documents pertinents trouvés, soit h.
Ainsi, la nouvelle mesure doit être une fonction dépendant des trois paramètres h, w, AP_{N_x} (Fig. 5.5) et prenant une valeur réelle dans le domaine $[0, 1]$, comme définie par l'équation 5.11.

$$MOR = f(h, w, AP_{N_x}) : \{0, .., \min(n, Nmax)\} \times \{0, .., Nmax\} \times [0, 1] \rightarrow [0, 1] \tag{5.11}$$

Cette fonction doit prendre en considération qu'un système de recherche d'informations cherche à maximiser h et AP_{N_x} et à minimiser w [2], et également tenir compte des contraintes formelles que nous allons définir par la suite.

5.4.2 Les contraintes formelles

Notre hypothèse est qu'une mesure d'évaluation adaptée à un contexte dirigé rappel doit remplir les contraintes citées dans les sections suivantes.

5.4.2.1 La priorité

Dans la section 5.3.1 (page 80), nous avons défini deux besoins qu'une mesure d'évaluation doit satisfaire pour évaluer le rappel des SRI : donner la priorité au rappel (B1) et ne considérer la précision qu'en cas de rappel équivalent (B2). Nous reprenons ces deux besoins pour formaliser les contraintes de la priorité à l'aide des paramètres h, w et AP_{N_x}. Avec ces trois éléments, nous distinguons trois niveaux de priorité, et ainsi les trois contraintes présentées dans le tableau 5.3. Dans ce tableau, nous précisons que la priorité est le rappel (h), et que pour un h donné, la priorité est ensuite w. AP_{N_x} ne doit influencer la mesure qu'en cas d'impossibilité de juger un système par rapport aux autres selon h et w.

2. Noter que par définition on a $w \geq h$.

Contraintes	Définition
c1	$h > h' \Rightarrow f(h, w, AP_{N_x}) > f(h', w', AP'_{N_x})$
c2	$h = h, w < w' \Rightarrow f(h, w, AP_{N_x}) > f(h', w', AP'_{N_x})$
c3	$h = h, w = w', AP_{N_x} > AP'_{N_x} \Rightarrow f(h, w, AP_{N_x}) > f(h', w', AP'_{N_x})$

TABLE 5.3 – Les contraintes de priorité de *MOR*

Contrainte	Niveau	Meilleur scénario	Pire scénario
c4	Principal	$h = n$	$h = 0$
c5	Pour un h donné	$w = h$	w =Nmax
c6	Pour un h et un w donnés	AP_{N_x}=$AP1$	AP_{N_x}=$AP0$

TABLE 5.4 – Les meilleurs et les pires scénarios de *MOR*

5.4.2.2 Meilleurs et pires scénarios

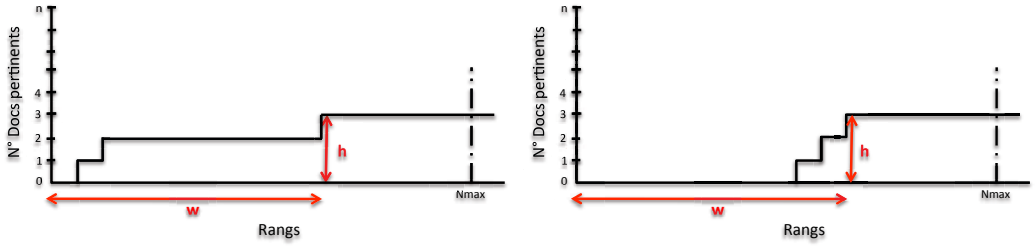
À chaque niveau de priorité, la nouvelle mesure doit être capable de détecter le meilleur et le pire scénario. Nous présentons ces scénarios dans le tableau 5.4. A priori, le meilleur scénario est de trouver tous les documents pertinents et le pire est de n'en découvrir aucun. Pour un h donné, l'idéal est d'obtenir les documents pertinents le plus tôt possible. Dans ce cas, le rang du dernier document pertinent vaut le nombre de documents pertinents trouvés, alors qu'au pire, ce rang se situe à $Nmax$. Pour un h et un w donnés, le meilleur et le pire scénario dépendent de AP_{N_x}. Par contre, rien ne peut garantir qu'AP_{N_x} prendra respectivement la valeur 1 ou 0 pour ces deux scénarios. Pour cela, nous supposons que ces valeurs minimum et maximum sont $AP1$ et $AP0$ dont nous parlons dans la section suivante.

5.4.3 Construction de la mesure

Pour garantir la priorité de h par rapport à w et à AP_{N_x} (contrainte c1), plusieurs alternatives peuvent être utilisées. Vu que h est un entier positif et $< N$, la méthode la plus simple est de considérer le somme de h avec une fonction de w et AP_{N_x} qui prend ces valeurs dans $[0, 1]$, ce que nous présentons dans l'équation 5.12.

$$f(h, w, AP_{N_x}) = \begin{cases} \frac{h+g_h(w,AP_{N_x})}{\min(n,Nmax)+1} & \text{if } h > 0 \\ 0 & \text{if } h = 0 \end{cases} \tag{5.12}$$

où $g_h(w, AP_{N_x})$ est cette fonction de w et AP_{N_x} pour un h donné, nous normalisons pour que la fonction finale ait une valeur entre 0 et 1 conformément à l'équation 5.11. La seule exception pour laquelle cette méthode ne peut pas garantir la priorité de h est quand h vaut 0, ce qui correspond à ne trouver aucun document pertinent. Ainsi, dans cette situation nous forçons la fonction finale à avoir la valeur 0, d'où la deuxième ligne de l'équation 5.12. Cette pratique, en plus de la normalisation, garantira les valeurs 1 et 0 pour le meilleur et le pire scénario respectivement, et



(a) (b)

FIGURE 5.6 – Meilleur (a) et pire (b) scénario pour un h et un w donnés

satisfera ainsi la contrainte c4.
Nous suivons la même stratégie de contrôle de priorité pour définir la fonction $g_h(w, AP_{N_x})$ comme cela est précisé par l'équation 5.13

$$g_h(w, AP_{N_x}) = \frac{(Nmax - w) + g_{hw}(AP_{N_x})}{(Nmax - h) + 1} \tag{5.13}$$

C'est-à-dire, pour garantir la contrainte c2 (w est prioritaire à AP_{N_x}), nous considérons la somme de w qui est un entier positif avec une fonction de AP_{N_x}. Par contre, nous utilisons $(Nmax - w)$ au lieu de w pour inverser l'effet de w, car plus sa valeur est petite plus le système évalué doit être récompensé. Ainsi, la fonction $g_h(w, AP_{N_x})$ est définie par l'équation 5.13 qui est également normalisée pour avoir des valeurs dans $[0, 1]$ (contrainte c5). Ainsi, la prise en compte de la précision moyenne (contrainte c3) passe par la fonction $g_{hw}(AP_{N_x})$. Le but de l'utilisation de cette fonction au lieu de l'emploi simple d'AP_{N_x} (qui est déjà dans le rang $[0, 1]$) est de satisfaire la contrainte c6 : L'utilisation d'AP_{N_x} directement dans $g_h(w, AP_{N_x})$ va perturber les résultats du meilleur et du pire scénario. Pour garantir l'obtention des bonnes valeurs dans ces cas, nous définissons la fonction $g_h(w, AP_{N_x})$ par l'équation 5.14.

$$g_{h,w}(AP_{N_x}) = \begin{cases} \frac{AP_{N_x} - AP0}{AP_{N_x} 1 - AP0} & \text{if } AP0 \neq AP1 \\ AP_{N_x} & \text{if } AP0 = AP1 \end{cases} \tag{5.14}$$

où $AP0$ et $AP1$ sont respectivement le meilleur et le pire scénario pour un h et un w donnés (Fig. 5.6). Notez que, quand h est égale à w, le meilleur et le pire scénario sont identiques. Pour cela, la fonction $g_{h,w}(AP_{N_x})$ prendra la valeur de AP_{N_x} dans ce cas particulier. Par ailleurs, pour calculer $AP0$ et $AP1$, nous utilisons l'équation 5.15 que nous obtenons en appliquant l'équation 5.10 pour le meilleur et le pire scénario.

$$AP0 = \frac{1}{n} \cdot \sum_{i=1}^{h} \frac{i}{w - h + i} \qquad AP1 = \frac{1}{n} \cdot (h - 1 + \frac{h}{w}) \tag{5.15}$$

5.4.4 Équation finale

En considérant les équations 5.12, 5.13 et 5.14, nous obtenons la définition finale de la mesure *MOR*

$$MOR = \begin{cases} \frac{h(Nmax - h + 1) + Nmax - w + \frac{AP_{N_x} - AP0}{AP1 - AP0}}{(\min(n, Nmax) + 1)(Nmax - h + 1)} & \text{if } h > 0 \text{ and } w > h \\ \frac{h(Nmax - h + 1) + Nmax - w + AP_{N_x}}{(\min(n, Nmax) + 1)(Nmax - h + 1)} & \text{if } h > 0 \text{ and } w = h \\ 0 & \text{if } h = 0 \end{cases}$$

où $AP0$ et $AP1$ sont définies par l'équation 5.15.

5.5 Caractéristiques de *MOR*

Avant d'évaluer *MOR*, nous citons dans les sections suivantes certaines de ses caractéristiques, notamment son rôle dans un contexte dirigé rappel, l'effet du paramètre w, son lien avec le tri à plusieurs clés et son comportement dans le cas d'un seul document pertinent.

5.5.1 Le rappel en priorité

Contrairement à la mesure de rappel ordinaire avec laquelle il est très facile d'obtenir le score maximal en rendant toute la collection, il n'existe pas de méthode triviale pour obtenir le score maximal de 1 avec *MOR*. De plus, cette mesure ne dépend pas de la propagation de la surface, qui représente la performance du système dans la courbe rappel/rangs, par rapport à un meilleur et un pire scénario comme c'est le cas de *PRES*, même si ces deux scénarios sont bien distingués par *MOR* (Table. 5.4). Entre ces deux scénarios, *MOR* va ordonner les systèmes selon le nombre de documents pertinents trouvés puis selon leur capacité à rendre les documents pertinents plus tôt dans la liste.
Pour faire la démonstration, nous reprenons les exemples de la section 5.3.1 pour le cas de 5 systèmes de recherche d'information qu'on souhaite classer par rapport à leur rappel pour une seul requête. Nous rappelons que ces systèmes rends 100 documents comme résultat de recherche et que la collection contient quatre documents pertinents. Nous présentons à nouveau dans le tableau 5.5 les valeurs des mesures *AP*, Rappel, F_1, F'_4, *PRES* et nous ajoutons la mesure *MOR*. Nous constatons facilement de ce tableau que *MOR* est la seul mesure qui classe les systèmes d'une façon identique au préférences humaines[3]. Le système 5 est le système le moins performant selon *MOR* malgré le fait qu'il a trouvé un document pertinent au premier

3. Nous rappelons que les systèmes sont présenté dans le tableau 5.5 dans un ordre qui correspond à l'avis d'un utilisateur dans un contexte de rappel.

	Rang de docs.perts	*AP*	Recall	F'_4	*PRES*	*MOR*
Système 1	{1, 2, 3, 4}	1	1	1	1	1
Système 2	{50, 51, 53, 54}	0.0474	1	0.0917	0.5	0.895
Système 3	{1, 98, 99, 100}	0.2727	1	0.429	0.28	0.801
Système 4	{1,54}	0.259	0.5	0.3414	0.37	0.495
Système 5	{1}	0.25	0.25	0.25	0.25	0.398

TABLE 5.5 – *ROM* comparé à *AP*, Recall, F'_4 et *PRES*

rang de la liste de résultats. De plus, contrairement à *PRES*, *MOR* a été capable de mieux placer le système 3, qui a trouvé tous les documents pertinents, par rapport au système 4 qui n'en a trouvé que la moitié.
Nous constatons qu'il est difficile pour un humain de donner une préférence entre les systèmes 2 et le système 3 qui ont trouvé le même nombre de documents pertinents dans les positions présentées dans le tableau 5.5. Si, ces documents ont tous été trouvés en début de la liste de résultats du système 2, par exemple, et tous à la fin de la liste du système 3 un utilisateur pourrait plus facilement décider que le système 2 est mieux, ce qui n'est pas le cas dans notre exemple. Nous remarquons que les valeurs de *MOR* pour ces systèmes (systèmes 2 et le système 3) sont très proches (0.895, 0.801) par rapport aux valeurs données par *PRES* (0.5, 0.28), ce qui nous permet de considérer que *MOR* représente bien l'hésitation d'un humain pour donner une préférence dans ce cas.

5.5.2 Effet du rang de dernier document pertinent

Le fait d'ordonner les systèmes possédant le même rappel par rapport au rang de dernier document pertinent trouvé, avant de prendre en compte l'AP_{N_x}, est un moyen de mesurer la possibilité d'arrêter l'investigation de résultats après avoir obtenu un certain nombre de documents pertinents, plutôt que de stopper après un certain rang, car le rang peut être exprimé en modifiant *Nmax* sans besoin d'être un paramètre indépendant intégré dans la mesure d'évaluation comme c'est le cas pour la mesure *PRES* [4].

5.5.3 *MOR* et le tri à plusieurs clés

Il faut noter que pour une requête donnée, il est possible de trier directement les systèmes de RI par rapport au rappel, puis relativement aux w et AP_{N_x}, ce qui donnera le même résultat qu'en les triant par rapport à leur score *MOR*. Par contre, le fait d'avoir une mesure numérique capable de reproduire ce tri nous permettra de pouvoir moyenner cette mesure sur l'ensemble des requêtes évaluées par les systèmes de RI, ce qui est impossible à réaliser par le tri à plusieurs clés.

4. Notre utilisation de *Nmax* dans *MOR* est uniquement pour inverser l'effet de w.

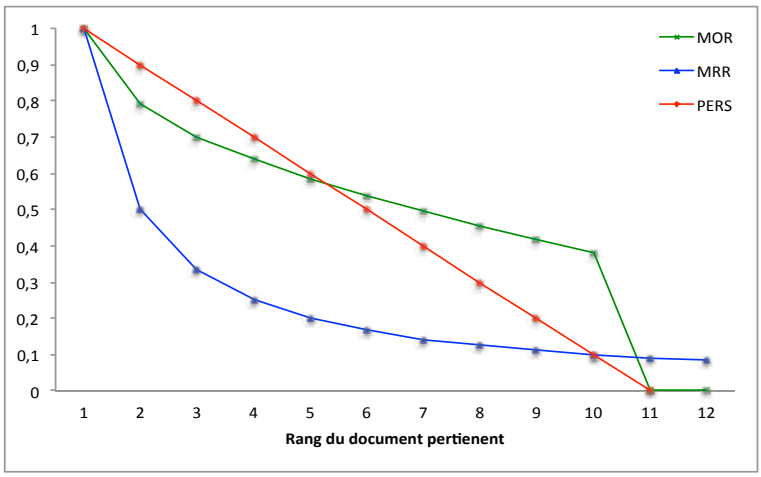


FIGURE 5.7 – La mesure MOR comparée à MRR et $PRES$ dans le cas d'un seul document pertinent pour $Nmax = 10$

5.5.4 Cas d'un seul document pertinent

Dans le cas particulier où il n'existe qu'un seul document pertinent pour la requête, comme dans le contexte de question/réponse, la mesure souvent utilisée est l'inverse du rang du document pertinent trouvé (MRR). Dans ce cas, l'AP_{N_x} dans la formule de MOR deviendra le MRR, car nous ne pouvons trouver qu'un seul document pertinent. La figure 5.7 représente le comportement de MOR comparé à celui de $PRES$ et MRR dans le cas d'un seul document pertinent. De cette figure, nous pouvons voir comment l'utilisation du rappel à $Nmax(h)$ dans l'équation de MOR donnera la valeur 0 à la mesure MOR si le document pertinent a été trouvé après Nmax. Ce comportement est cohérent avec le fait que l'utilisateur ne va jamais voir les documents aux rangs supérieurs à Nmax, c'est aussi la même idée avec $PRES$ pour ce pire scénario. Comme nous pouvons le constater dans la figure, MOR pénalisera moins les systèmes qui trouvent le document pertinent vers la fin de la liste. Notre mesure est conçue pour un contexte où le rappel à $Nmax$ a le plus d'importance, alors que dans le cas d'un seul document pertinent le rappel n'est pas gradué (1 ou 0), ce qui signifie que MOR n'est pas adapté à ce cas. L'utilisateur, dans ce contexte, arrêtera la recherche après avoir trouvé le document pertinent, cela correspond plus à une mesure de comportement linéaire, dans ce cas comme $PRES$.

5.6 Analyse experimentale

L'évaluation d'une mesure d'évaluation n'est pas simple. Logiquement, il faut évaluer à quel point la mesure satisfait les besoins pour lesquels elle a été construite. Dans notre cas, ces besoins ont été formalisés par des contraintes lors de la génération de la mesure, ce qui nous permet de garantir que MOR satisfait bien ces besoins,

Measures	τ
$MOR \leftrightarrow$ Recall	0.619
$MOR \leftrightarrow PRES$	0.205
$MOR \leftrightarrow MAP$	0.093
$PRES \leftrightarrow$ Recall	0.160
$PRES \leftrightarrow MAP$	0.075
Recall $\leftrightarrow MAP$	0.123

TABLE 5.6 – Le coefficient de correlation *Kendall tau pour les différents couples de mesures ($\tau = 1$ signifie un accord parfait)*

c'est-à-dire que pour évaluer un système de recherche d'information, *MOR* donnera la priorité au rappel, puis au positionnement des documents pertinents dans la liste de résultats (w et AP_{N_x}).
Néamoins, il est intéressant de statistiquement étudier le comportement de la mesure sur un ensemble de cas. Ce que nous faisons dans cette section est de voir la ressemblance entre *MOR* et le rappel. Une ressemblance très forte signifie que la mesure n'est pas intéressante car elle n'apportera pas une valeur ajoutée par rapport au rappel. Un écart très important entre les comportements des deux mesures signifie que *MOR* n'est pas une mesure dirigée rappel. Pour cela, nous vérifions que *MOR* soit plus proche du rappel que du *MAP* sans qu'elle soit une duplication du rappel.

5.6.1 Description de l'expérience

Nous nous intéressons aux comparaisons du *MOR* avec le rappel, le *MAP* et *PRES*. Nous réalisons deux expériences sur un ensemble de 88 run [5] du track médical de TREC2012. La première expérience est de calculer la corrélation entre *MOR* et les autres mesures en utilisant le test de Kendall-tau [Kendall, 1938], et la deuxième est de comparer les décisions de ces mesures par rapport au run le plus performant et le plus mauvais.

5.6.2 Corrélation entre *MOR* et les autres mesures

Le test de Kendall-tau permet de mesurer le degré d'accord entre deux classements. Pour cela, pour mesurer la corrélation entre deux mesures, nous calculons le kendall-tau sur les deux listes ordonnées des systèmes que ces deux mesures ont évalué. Les résultats de ces comparaisons sont dans le tableau 5.6. Nous constatons de ce tableau une valeur cohérente avec l'objectif du *MOR*. Sans être complètement identique au rappel, *MOR* reste plus proche du rappel que de la précision. Cela indique que l'utilisation de w et AP_{N_x} a un effet important sur la mesure sans la

5. Un *run* dans la campagne d'évaluation TREC est la liste de résultats obtenue par un système pour un ensemble de requêtes. Les informations importantes dans cette liste sont, pour chaque requête, les documents pertinents et leur scores de pertinence selon le système qu'on évalue

Mesure	meilleur (score)	pire (score)
MOR	83 (0.819)	49 (0.018)
Recall	83 (0.826)	49 (0.012)
PRES	84 (0.734)	49 (0.011)
MAP	30 (0.461)	49 (0.002)

TABLE 5.7 – La meilleure et la pire systèmes selon les mesures d'évaluation

biaiser vers la précision. Ce qui est intéressant dans ce tableau est de voir que notre mesure est plus orientée rappel que *PRES*, qui est légèrement plus proche du rappel que de la précision.

5.6.3 Effet sur le classement

Nous présentons dans le tableau 5.7 les avis des trois mesures (*MOR*, rappel, *MAP* et *PRES*) sur le meilleur et pire systèmes parmi les 88 runs évalués. Nous constatons que toutes les mesures ont choisi le même système comme le moins performant, mais ils ne sont pas d'accord sur le système qui a la meilleure performance. Là aussi, on voit la cohérence entre le rappel et *MOR* sur les décisions des meilleur et pire systèmes. Alors que ce tableau indique que *MOR* n'apporte pas d'information intéressantes par rapport au rappel, l'expérience précédente confirme que les choix des deux mesures ne sont pas toujours identiques.

5.7 Résumé

Nous avons présenté dans ce chapitre la mesure d'évaluation *MOR* qui est adaptée à l'évaluation de systèmes de recherche d'informations dans un contexte de rappel. En utilisant des contraintes formelles inspirées par les besoins de ce contexte, nous avons construit la fonction finale étape par étape. Par conséquent, la mesure proposée correspond entièrement à ces besoins. Bien qu'il soit difficile de mesurer l'efficacité d'une mesure d'évaluation, nous avons présenté deux expériences qui permettent de confirmer que notre mesure correspond à l'objectif d'être dirigée rappel sans pour autant être identique au rappel.

En plus de ces évaluations, il est intéressant de comparer les jugements de la mesures avec les jugements humains dans un contexte rappel. On peut par exemple présenter les résultats de recherche de plusieurs systèmes en indiquant aux utilisateurs le placement des documents pertinents dans ces listes (pour éviter le biais de jugement de pertinence), et demander à ces utilisateurs de classer les systèmes par préférence. Les classement des utilisateurs pourront ainsi être utilisés dans nos deux expériences pour voir la relation entre les mesures automatiques et un classement des systèmes d'un point de vue humain. Par ailleurs, *MOR* fera partie des mesures d'évaluation que nous utilisons, dans le chapitre 6, pour évaluer notre approche sémantique mixte d'expansion et de reformulation des requêtes (Chap. 4).

Chapitre 6

Expériences et évaluations

Sommaire

6.1 Plan d'évaluation

Nous avons proposé dans le chapitre 4 une approche d'expansion et de reformulation sémantique des requêtes web (ASMER) qui génère des requêtes conceptuelles tout en respectant les termes originaux employés par l'utilisateur afin de préserver la compréhensibilité et la capacité de modification par un humain. Notre but était de créer une requête qui exprime mieux le besoin d'information de l'utilisateur, en supposant qu'une telle requête trouvera plus de documents pertinents dans des bonnes positions dans la liste. Notre stratégie d'évaluation est d'étudier la performance moyenne d'ASMER sur un ensemble de requêtes vis-à-vis des requêtes sans expansion et des requêtes étendues par une méthode de l'état de l'art. Après

	Taille de l'index	# documents	# termes uniques	μ_d
Inex2006	2,42G	659.388	3.102.720	405
Inex2009	5,69G	2.666.190	12.598.810	700
Wt10G	9,74G	1.692.096	5.406.526	614
Gov2	202G	25.205.179	39.177.861	930

TABLE 6.1 – Statistiques des collections des tests. μ_d est la moyenne de taille des documents dans la collection.

cette comparaison, nous évaluons la sensibilité d'ASMER aux changements des différents paramètres sur la précision et le rappel. Dans la section 6.4 nous observons le comportement d'ASMER au niveau de la requête, où nous analysons le nombre de requêtes améliorées ou dégradées en utilisant notre approche, et nous faisons un zoom sur les requêtes des cas extrêmes. Finalement, nous analysons l'utilisation d'une seule ressource pour l'expansion de la requête : soit l'ontologie YAGO soit les documents de retour de pertinence exploités avec LSI par rapport à leur combinaison par ASMER. Nous commençons ce chapitre par une description de notre environnement de test.

6.2 Description de l'environnement

Nous présentons dans cette section les données de test utilisées dans nos expériences ainsi que les modèles de références, les mesures d'évaluation et le réglage des paramètres.

6.2.1 Collections et Requêtes de test

Nous avons choisi quatre collections standards de test, issues des campagnes de recherche d'information TREC et INEX. Des statistiques de ces collections sont présentées dans le tableau 6.1 Ces quatre collections sont des collections du Web. La collection Inex2006 contient les articles anglais de Wikipedia de l'année 2005 en format XML. Inex2009 est également la version XML anglaise de Wikipedia mais celle de l'année 2008. La spécificité de cette collection est qu'elle a été annotée par des balises sémantiques dont l'intitulé est issu de l'ontologie Yago. Nous traitons le contenu de ces balises comme appartenant au texte des documents. La collection Wt10g de Trec contient des pages anglaises récupérées du Web en 2000, alors que la collection GOV2 contient des pages récupérées de sites du gouvernement américain sous le domaine .gov dans l'année 2002. Toutes ces collections ont été indexées avec les mêmes paramètres : l'outil Indri (version 5.5), la racinisation de Krovetz, et la liste des mots vides standard d'Inquery (Annexe. C). Les jeux de requêtes utilisés pour nos tests sont listés dans les tableaux 6.2 et 6.3, où nous montrons également le nombre de requêtes qui contiennent des entités nommées et le nombre de mots utiles que contiennent ces requêtes. Nous avons utilisé uniquement la partie « title » des requêtes, aucune information des autres champs (comme le

	requêtes	année(track)	nombe de requêtes jugées	nb des entités nommées
Inex2006	544-677	2008(ad hoc)	70	23
Inex2009	1-115	2009(ad hoc)	68	21
Wt10G	451-550	2000-2001 (Web ad hoc)	98	25
Gov2	701-850	2004-2006 (Terrabyte)	148	47

TABLE 6.2 – Les requêtes de tests

Collection	1 terme	2 à 3 termes	4 termes ou plus
Inex2006	$\approx 3\%$	$\approx 80\%$	$\approx 17\%$
Inex2009	$\approx 2\%$	$\approx 41\%$	$\approx 57\%$
Wt10g	$\approx 17\%$	$\approx 62\%$	$\approx 21\%$
Gov2	$\approx 1\%$	$\approx 72\%$	$\approx 27\%$

TABLE 6.3 – Distribution de requêtes selon le nombre de mots utiles.

narrative et la *description*) n'a été considérée. Lors de l'exécution, les requêtes sont racinisées automatiquement avec la même méthode utilisée pour l'indexation de documents (Krovetz dans notre cas), cela s'applique aussi pour l'itération de retour de pertinence de notre méthode locale (LSI).
Concernant les mots vides, il est indispensable pour notre approche d'enlever ces mots de la requête, pour qu'il ne soient pas étendus. Par contre, pour les concepts explicites, le fait d'enlever les mots vides a un effet négatif sur l'identification des entités nommées, car beaucoup d'acronymes serons perdus par cette opération (par exemple *US* qui est un mot vide mais qui est aussi un acronyme pour les État-Unis). Ainsi, pour l'extraction des concepts explicites, nous passons les requêtes dans leurs état brut sans aucune manipulation, et l'élimination des mots vides concernera uniquement les termes des requêtes qui n'ont pas été identifiés comme des entités nommées dans Yago.

6.2.2 Modèles de référence

Le modèle de recherche que nous utilisons pour nos expériences est le modèle de vraisemblance de requête (Annexe. A). Ce modèle de recherche est utilisé également pour obtenir les documents de retour de pertinence.
Notre premier modèle de référence utilise donc ce modèle de langue avec des requêtes non étendues, ce que nous appelons l'approche QL (*Query Likelyhood*), car ce modèle est utilisé pour l'exécution des requêtes originales. Notre deuxième réfé-

Approche	Description
QL	L'utilisation des requêtes sans expansion avec le modèle de vraisemblance de la requête
RM3	L'expansion par le modèle de pertinence [Lavrenko and Croft, 2001]
Yago	Notre approche de modélisation conceptuelle des requêtes en utilisant YAGO (modèle de recherche : QL)
LSI	Notre approche de modélisation conceptuelle des requêtes en utilisant LSI avec les documents de retour de pertinence (modèle de recherche : QL)
ASMER	Notre approche de modélisation conceptuelle des requêtes en utilisant des concepts explicites et implicites

TABLE 6.4 – Les modèles comparés

rence est le modèle RM3 (*Relevance Model*) [1] d'expansion de la requête par retour de pertinence (Sect. 3.4.1.2, page 30). RM3 est un modèle de l'état de l'art utilisé souvent comme modèle de référence pour les approches d'expansion et de reformulation de la requête, il est reconnu pour sa stabilité et son efficacité [Lv and Zhai, 2009].
Pour bien évaluer l'efficacité du mélange de concepts implicites et explicites à travers ASMER, nous comparons cette approche avec l'utilisation d'un seul type de ressources, où nous appelons *Yago* l'approche qui utilise uniquement les concepts explicites pour tous les termes de la requête (Sect. 4.3.1, page 51), et *LSI* celle qui n'utilise que les concepts implicites pour étendre tous les mots de la requête (Sect. 4.3.2, page 55). Le tableau 6.4 résume les approches prise en compte dans nos expériences.

6.2.3 Mesures d'évaluation

Toutes nos requêtes sont lancées pour trouver 1000 documents au maximum. Pour évaluer la précision, qui est la notion la plus importante dans le contexte Web, nous utilisons les mesures *MAP*, *P@10* et *MRR*. Pour évaluer le rappel, nous utilisons la mesure *MOR* que nous avons proposée dans le chapitre 6. En plus de ces mesures numériques, le comportement de chaque méthode est représenté avec une courbe rappel/précision.

6.2.4 Réglage de paramètres

La comparaison de nos approches avec les modèles de références doit prendre en compte les paramètres que chaque modèle utilise. Le tableau 6.5 représente tous les

1. Nous utilisons les acronymes anglais (QL et RM3) car ils sont souvent utilisés dans le domaine pour designer le modèle de recherche par vraisemblance de la requête (QL) et le modèle d'expansion de la requête par le modèle de pertinence (RM3).

Paramètre	Description	Approche
μ	le paramètre de lissage de Dirichlet pour le modèle QL et RM3	Toutes
n	le nombre de documents de retour de pertinence	LSI, ASMER et RM3(n_{Rm3})
t	le nombre de termes d'expansion de la requête étendue (sac de mots)	RM3
m	le nombre de termes pour chaque concepts	LSI et ASMER
k	le nombre de dimensions à garder pour LSI	LSI et ASMER
$\alpha 1$	le seuil de certitude	LSI, Yago et ASMER
$\alpha 2$	le seuil de spécificité	LSI, Yago et ASMER
λ	le poids de la requête originale par rapport à la requête étendue	LSI, Yago, ASMER et RM3(λ_{Rm3})

TABLE 6.5 – Les paramètres libres des approches que nous comparons

paramètres de chaque approche avec leur description. Plusieurs approches existent pour comparer deux modèles qui dépendent de paramètres différents. Nous avons constaté que deux de ces approches sont souvent utilisées dans l'état de l'art : l'optimisation des paramètres sur l'ensemble de toutes les requêtes (ex. Metzler and Croft [2005]) et la considération des requêtes d'apprentissage pour optimiser les paramètres, puis comparer les modèles sur un autre ensemble de test, éventuellement avec une procédure de validation croisée (ex. Bendersky and Croft [2008]). Nous choisissons pour notre étude la première approche où la mesure que nous souhaitons optimiser est le *MAP*. Par contre, nous considérons les points suivants :

— μ est un paramètre spécifique au modèle de recherche d'information QL que toutes les approches évaluées utilisent. Nous utilisons donc sa valeur par défaut dans Indri (2500) dans toutes nos expériences.
— Nous avons réalisé une étude expérimentale dans laquelle nous avons observé un lien entre les seuils de qualité $\alpha 1$ et $\alpha 2$ et l'informativité des mots. Nous avons remarqué que les mots deviennent moins informatifs en dessus du niveau 7 dans WordNet, ce que nous avons fixé comme seuil de spécificité. Nous avons constaté également un lien entre le seuil de certitude et le nombre de dimensions de LSI. Plus le nombre de dimension est grand plus petites deviennent les valeurs de similarité entre les termes. Bien qu'il soit intéressant d'étudier de plus près ce phénomène, cette étude ne fera pas partie de cette thèse, et nous fixons la valeur $0,4$ comme seuil de certitude pour les approches Yago, LSI et ASMER.
— Le paramètre λ a le même rôle pour toutes les approches de reformulation que nous évaluons. Deux points de vue peuvent être considérés : soit de voir λ comme un paramètre libre et chercher à l'optimiser pour chaque approche,

	Valeurs testées	Inex06	Inex09	Wt10g	Gov2
μ	–	**2500**	**2500**	**2500**	**2500**
n	$\{10, 20, 30\}$	20	10	30	10
n_{Rm3}	$\{10, 20, 30\}$	10	10	10	10
m	$\{3, 5, 7\}$	5	7	3	7
t	$\{10, 15, 20\}$	20	20	20	20
λ	{0,2 0,5 0,8}	0,8	0,8	0,5	0,8
λ_{Rm3}	{0,2 0,5 0,8}	0,5	0,8	0,8	0,8
k	$\{5, 10, 15\}$	10	5	10	5
$\alpha 1$	–	**0,4**	**0,4**	**0,4**	**0,4**
$\alpha 2$	–	**7**	**7**	**7**	**7**

TABLE 6.6 – Les paramètres des approches comparées et leur valeur optimale selon la collection.

soit de le fixer pour ne comparer que la partie étendue pour toutes les approches. Nous choisissons de considérer λ comme un paramètre libre pour considérer plutôt les approches dans leur intégralité dans leurs meilleures performances.

— Pour LSI, il faut que le nombre de documents, n, soit supérieur au nombre de dimensions que nous souhaitons retenir, k, pour que la réduction de la matrice ait du sens. Cela diminue les combinaisons possibles entre ces deux paramètres.

— le nombre de termes d'expansion n'a pas la même signification entre nos approches et l'approche RM3. Alors que pour RM3 il s'agit du nombre total de termes t dans la partie étendue de la requête, il s'agit pour nos approches du nombre de termes d'expansion m pour chaque concept dans la requête. Pour cette raison, nous ne faisons pas varier ces deux paramètres dans le même ensemble de valeurs.

En considérant les points précédents, nous présentons dans le tableau 6.6 les valeurs des paramètres que nous avons retenues pour chaque jeu de test. Les valeurs en gras sont les valeurs qui ont été fixées empiriquement, alors que les autres valeurs ont été optimisées (selon leur *MAP*) sur toutes les requêtes d'évaluation pour chaque collection. Nous allons voir l'effet du changement des paramètres optimisés (qui ne sont pas en gras dans le tableau 6.6) dans la section 6.3.2.

6.3 Évaluation générale d'ASMER

Nous étudions dans cette section la performance d'ASMER en moyenne sur l'ensemble des requêtes pour chaque collection. Cela comprend l'évaluation du rappel et de la précision de ce modèle en comparaison avec nos modèles de référence, puis une évaluation de la sensibilité de notre modèle par rapport aux paramètres.

		MAP	*P@10*	MRR	*MOR*
Inex06	QL	33,00	53,00	81,97	83,19
	RM3	**35,96**	**55,00**	80,37	**84,61**
	ASMER	34,78	53,71	**84,81**	83,71
Inex09	QL	34,17	**97,50**	97,79	45,89
	RM3	34,06	96,76	97,43	45,87
	ASMER	**34,41**	97,21	**98,53**	**46,18**
Wt10g	QL	20,16	29,18	58,54	70,74
	RM3	20,49	29,08	56,10	71,06
	ASMER	**21,69**	**29,80**	**59,42**	**71,40**
Gov2	QL	29,41	53,51	72,36	70,57
	RM3	29,97	52,97	68,86	71,15
	ASMER	**30,82**	**56,22**	**75,84**	**71,70**

TABLE 6.7 – Les résultats d'évaluation sur les quatre collections de tests. Les valeurs en gras sont les plus grandes dans leur colonne pour la collection concernée.

6.3.1 Performance en précision et en rappel

Nous présentons dans le tableau 6.7 la performance d'ASMER comparée à QL et RM3 selon plusieurs métriques d'évaluation. Le pourcentage et la significativité statistique d'amélioration entre chaque couple de ces approches est présenté dans le tableau 6.8.

Avant de parler de la performance d'ASMER, il est intéressant de signaler que le *MAP* des requêtes sans expansion en utilisant le modèle QL pour la collection Wt10g (qui vaut *20,16*) est bien inférieur (-37%) à la moyenne des *MAP* des trois autres collections (qui vaut *32,19*). En revenant sur le tableau 6.3 (page 97) de distribution du nombre de mots dans les requêtes 6.3, nous constatons que pour Wt10g 17% des requêtes se compose d'un seul mot utile, contre 3%, 2% et 1% pour les collections Inex2006, Inex2009 et Gov2 respectivement. En revanche, nous remarquons que la majorité des requêtes de la collection Inex2009 se composent d'au moins quatre mots utiles, et que les requêtes de base sans expansion obtiennent le meilleur *MAP* pour cette collection par rapport aux autres collections de tests. La relation entre la taille de la requête et la performance d'un modèle de recherche n'est pas nouvelle, il est d'ailleurs la motivation des études sur l'expansion de la requête. Mais nous insistons sur cette observation pour faire le lien avec la performance de nos approches d'expansion.

6.3.1.1 MAP

Les tableaux 6.7 et 6.8 indiquent qu'ASMER a été capable d'améliorer significativement le *MAP* de trois collections sur quatre par rapport au modèle QL sans expansion. Il est intéressant de noter que pour la quatrième collection (Inex2009), qui a déjà un bon *MAP*, aucune amélioration par rapport à l'utilisation des re-

		MAP	*P@10*	MRR	*MOR*
Inex06	RM3/QL	+ 8,97*	+ 3,77*	− 1,95	+ 1,71
	ASMER/QL	+ 5,39*	+ 1,40	+ 3,46	+ 0,63*
	ASMER/RM3	− 3,28	− 2,35	+ 5,52*	− 1,06
Inex09	RM3/QL	− 0,32	− 0,76	− 0,37	+ 0,04
	ASMER/QL	+ 0,70	− 0,30	+ 0,76	+ 0,63
	ASMER/RM3	+ 1,03	+ 0,47	+ 1,13	+ 0,68
Wt10g	RM3/QL	+ 1,64	+ 0,34	− 4,16	+ 0,45
	ASMER/QL	+ 7,59*	+ 2,12	+ 1,50	+ 0,93
	ASMER/RM3	+ 5,86*	+ 2,48	+ 5,92	+ 0,48
Gov2	RM3/QL	+ 1,90*	− 1,00	− 4,84	+ 0,82*
	ASMER/QL	+ 4,79*	+ 5,06*	+ 4,91*	+ 1,60*
	ASMER/RM3	+ 2,84*	+ 6,13*	+10,14*	+ 0,77

TABLE 6.8 – Le pourcentage d'amélioration en *MAP*, *P@10*, MRR et *MOR* sur les quatre collections entre chaque couple d'approches. L'étoile signifie une significativité statistique (p<0,05) pour les deux tests : t-test et le test de randomization.

	nb requêtes	nb requêtes étendues partiellement	nb requêtes non étendues
Inex2006	70	43(61%)	2
Inex2009	68	38(56%)	1
Wt10g	98	37(38%)	5
Gov2	148	35(24%)	2

TABLE 6.9 – Le rôle des contraintes de spécificité et de certitude sur l'expansion sélective des requêtes.

quêtes sans expansion est significative, que ce soit en utilisant ASMER ou RM3. Cela s'applique également pour les autres mesures de performance. Par contre, pour cette collection, ASMER est statistiquement meilleur que RM3. La raison en est que l'expansion de la requête pour ASMER n'est pas systématique, les contraintes de spécificité et de certitude font que certaines requêtes ne sont pas étendues ou étendues partiellement [2] , ce qu'on peut constater dans le tableau 6.9. Cette expansion sélective n'est pas appliquée par l'approche RM3 qui ajoute systématiquement le même nombre de termes d'expansion à chaque requête. On voit du tableau 6.9 que plus que la moitié des requêtes longues d'Inex2009 n'ont pas été étendues entièrement, ce qui a joué en faveur d'ASMER par rapport à RM3 qui ajoute 20 termes d'expansion à chaque requête de cette collection.

Pour vérifier l'avantage d'ASMER sur RM3 pour les requêtes longues, nous avons

2. On suppose qu'une requête est étendue partiellement si les formules étendues des termes originaux sont absente ou contiennent moins de termes d'expansion que le nombre fixé par le paramètre m du tableau 6.6 (page 100)

	#requêtes	% de requêtes longues
MAP(ASMER) > MAP(RM3)	38	66%
MAP(ASMER) < MAP(RM3)	30	46%

TABLE 6.10 – Le pourcentage de requêtes longues (>3 mots) parmi les requêtes améliorées/dégradées en *MAP* lors de l'utilisation d'ASMER par rapport à RM3 sur la collection INEX2009.

calculé le pourcentage des requêtes longues parmi celles où ASMER est supérieur à RM3 sur la collection Inex2009 (Table. 6.10). Comme ce tableau l'indique, l'expansion sélective d'ASMER a été plus efficace que l'expansion par RM3 sur les requêtes longues d'Inex2009, et c'est la raison pour laquelle ASMER a réussi à améliorer légèrement le *MAP* de cette collection (où les requêtes longues sont majoritaires) alors que RM3 l'a dégradé.
Pour les trois collections Inex2006, Wt10g et Gov2, la majorité de requêtes contiennent 2 à 3 mots utiles, ce qui correspond au constat de plusieurs papiers de l'état de l'art que la taille des requêtes Web est en entre 2 et 3 mots. Le comportement d'ASMER en *MAP* pour ces trois collections est homogène, on obtient une amélioration statistiquement significative entre $4,79$ et $7,59\%$, ce qui n'est pas le cas de RM3 qui a une performance supérieure à ASMER sur Inex2006 mais moins bonne sur les collections de TREC. La différence entre la collection Inex2006 et les collections de TREC est principalement le nombre de documents dans la collection et la taille moyenne des documents, qui est bien supérieure dans les collections de TREC. Ces deux éléments ont un rôle pour ASMER et RM3. D'un côté, la taille de la collection et la fréquence d'un terme dans la collection font partie de la formule RM3 (Sect. 3.4.1.2, page 30), de l'autre côté, la taille des documents de retour de pertinence est décisive pour LSI : le fait d'avoir des documents très courts pour une requête n'amènera pas suffisamment d'information pour LSI, et on va avoir une mauvaise performance pour cette requête.

6.3.1.2 *P@10* et MRR

Le comportement des deux approches RM3 et ASMER concernant la précision à 10 est très proche de leur comportement vis-à-vis la mesure *MAP*. La collection Inex2009 pose un problème pour les deux approches, RM3 se comporte mieux sur Inex2006 qu'ASMER alors que ce dernier a une meilleure performance sur les collections de TREC. Ce qui est remarquable est la supériorité statistiquement significative d'ASMER sur la collection la plus grande collection GOV2, où on voit clairement l'effet bénéfique des documents de grande taille dont dispose cette collection (900 termes en moyenne par document). L'effet de la taille de la collection sur la précision à dix documents est présenté dans la figure 6.1 , où nous présentons le pourcentage d'amélioration en *P@10* apporté par RM3 et ASMER pour les quatre collections, ordonnés sur l'axe horizontal par la taille ascendante des collections.

La figure 6.1 permet de voir une relation entre la taille de la collection et la

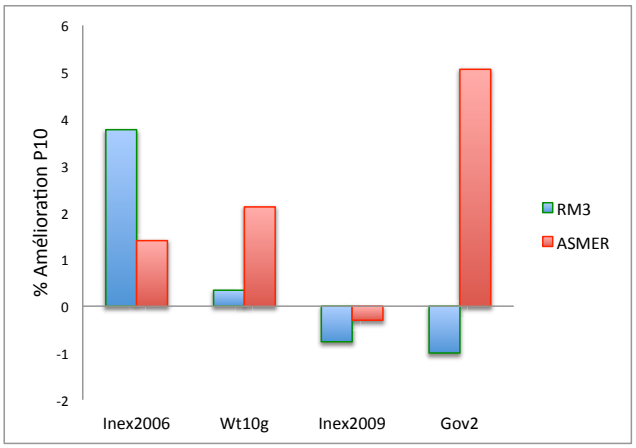


FIGURE 6.1 – Pourcentage d'amélioration en *P@10* pour RM3 et ASMER sur les quatre collections ordonnées d'une manières ascendante par rapport à leur taille.

précision à 10 de chaque méthode. Cette relation est décroissante pour RM3 : plus la collection est grande moins le modèle est précis au dixième rang. Alors que c'est le cas inverse pour ASMER, à l'exception de la collection Inex2009 qui a la particularité d'avoir des requêtes longues à l'origine.
Avec les tableaux 6.7 et 6.8, nous constatons qu'ASMER est capable de trouver un premier document pertinent avant RM3 et QL (MRR) pour les quatre collections de tests, ce qui est un comportement apprécié pour la recherche exploratoire sur le Web, car il peut signifier moins d'effort pour l'utilisateur qui n'a pas besoin de plus d'un document pertinent. En observant la collection Gov2, nous remarquons qu'ASMER est nettement mieux en MRR que RM3, RM3 dégrade même la performance par rapport aux requêtes non étendues sur cette collection.

6.3.1.3 *MOR*

Concernant le rappel selon la mesure *MOR*, les tableaux 6.7 et 6.8 (page 101) montrent que les deux approches, ASMER et RM3, n'apportent pas une amélioration importante par rapport à l'utilisation des requêtes simples sans expansion. Cela veut dire qu'aucune de ces deux approches n'a été capable de trouver plus de documents pertinents par rapport aux requêtes originales. Pour interpréter ce comportement, nous avons trois arguments. Le premier argument se trouve dans le tableau 6.11, où on voit que pour les collections Inex2006, Wt10g et Gov2, les requêtes sans expansion trouvent la majorité des documents pertinents, c'est-à-dire que le modèle de base se comporte suffisamment bien qu'aucune des approches d'expansion n'arrive à faire mieux. En deuxième lieu, nous rejoignons Deveaud [2013] dans son idée sur le rôle du faible pourcentage de documents jugés par rapport au grand nombre de documents dans les collections de test. La figure 6.2 (page 105) montre le pourcentage (en moyenne) de documents non jugés parmi ceux trouvés par chaque requête pour chaque collection de test. En se rappelant que nous trou-

	QL
Inex06	83,85
Inex09	45,95
Wt10g	72,03
Gov2	71,05

TABLE 6.11 – Le rappel à 1000 pour le modèle QL pour les quatre collections de test

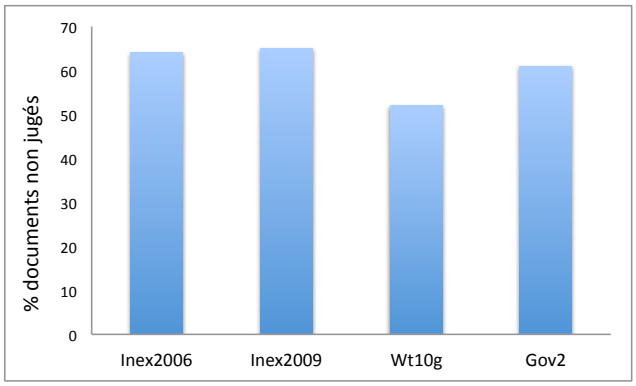


FIGURE 6.2 – La moyenne des pourcentage de documents non jugés par requête pour les quatre collections

vons 1000 documents (au plus) pour chaque requête, moins de 500 de ces document ont été jugés (positivement ou non) par un assesseur. Cela veut dire qu'avec un rappel déjà élevé, il y a une grande chance qu'un document pertinent trouvé par une approche d'expansion n'ai pas été jugé. Finalement, pour expliquer pourquoi ASMER se comporte mieux en précision qu'en rappel, nous revenons sur l'idée de la similarité sémantique qui est la base de notre approche : les documents qui utilisent plusieurs termes, sémantiquement similaires, pour exprimer le même objet seront récompensés par ASMER et obtiendrons un meilleur score grâce à ces termes que notre approche considères comme des « synonymes ». Une précision améliorée et un rappel moins amélioré par rapport aux requêtes sans expansion peut donc signifier que, pour nos collections de test, dans la plupart de cas, un document est soit pertinent et fait référence au même objet en utilisant plusieurs mots, soit il ne contient aucun de ces synonymes et n'est donc pas considéré comme pertinent.

6.3.1.4 Courbe rappel/précision

Les commentaires que nous avons faits jusqu'à présent sur les tableaux 6.7 et 6.8 (page 101) peuvent être visualisés dans leur ensemble grâce aux courbes Rappel/Précision que nous présentons par la figure 6.3. Avec ces courbes, on peut constater facilement la spécificité de la collection Inex2009, où toutes les approches trouvent la majorité des documents pertinents plus tôt dans la liste de résultats par

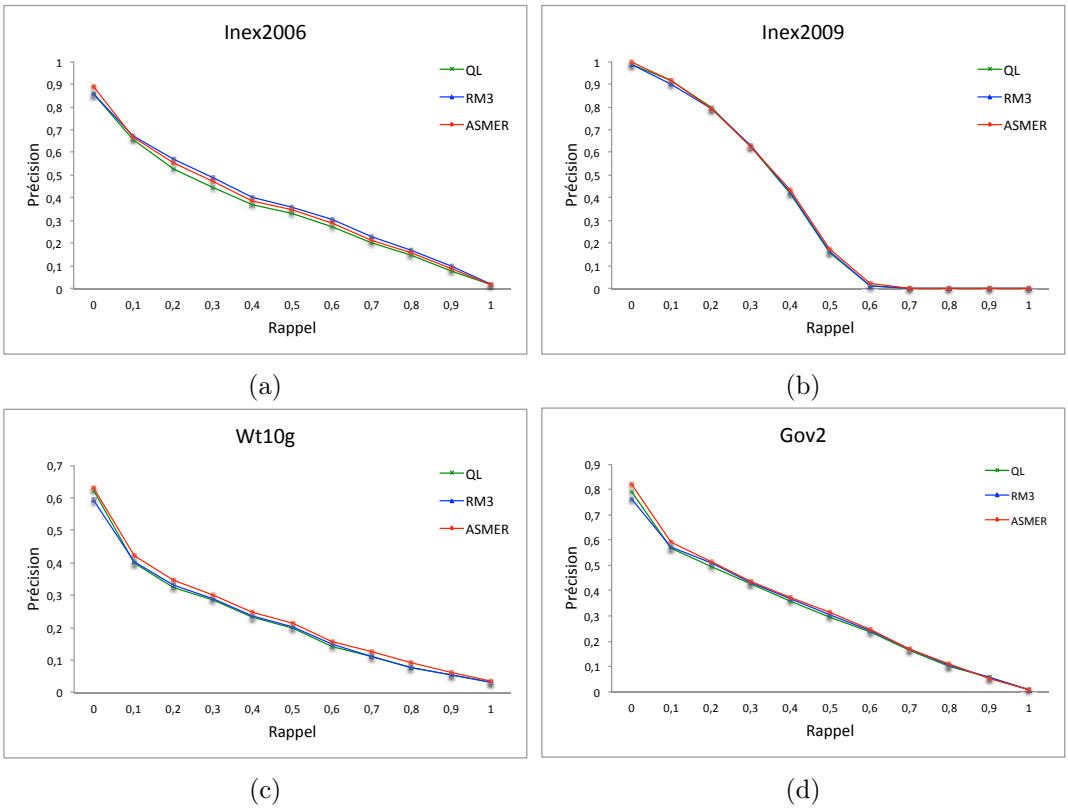


FIGURE 6.3 – Les courbes Précision/Rappel de nos quatre collections de tests et les trois approches comparées (QL, RM3 et ASMER).

rapport aux autres collections, mais elles cessent d'en trouver plus à partir du point 0,7 de rappel. Pour les trois autres collections, les courbes sont plus habituelles, et ASMER se comporte mieux que les autres approches pour les collections WT10G et Gov2 sur tous les points de rappel, alors que pour la collection Inex2006 c'est RM3, même si ASMER trouve toujours le premier document pertinent plus tôt que RM3 sur les quatre collections de tests. Pour les quatre collections, RM3 et ASMER ont une courbe très semblable à celle de QL, ce qui signifie que, même si ces approches améliorent la *MAP*, la proportion entre le rappel et la précision n'est pas largement modifiée par rapport au modèle de base sans expansion.

6.3.2 Robustesse

ASMER dépend de paramètres globaux et locaux : les paramètres globaux contrôlent le poids de la requête originale λ et le nombre de termes à ajouter dans chaque ensemble d'expansion m. Les paramètres locaux sont spécifiques à l'approche LSI, il s'agit du nombre de documents de retour de pertinence n, et du nombre de dimensions à garder pour l'espace conceptuel généré par LSI k. À part des seuils de spécificité et de certitudes qui sont également globaux mais que nous avons fixés d'une manière empirique, l'approche YAGO ne dépend pas de paramètres. Comme

nous l'avons spécifié avant (Sect. 4.3.1, page 51), une fois le concept identifié dans l'ontologie, nous utilisons les liens de similarité sémantique qui nous donnent des termes non pondérés similaires au terme original, que nous ne pouvons pas ordonner. Pour cela, même si le nombre de mots dans chaque concept, m, est théoriquement global, en réalité il ne s'applique qu'aux ensembles d'expansions issus de LSI. Nous étudions dans cette section la sensibilité d'ASMER aux paramètres globaux uniquement. Les paramètres locaux de LSI (n et k) seront étudiés dans la section dédiée à l'approche LSI.

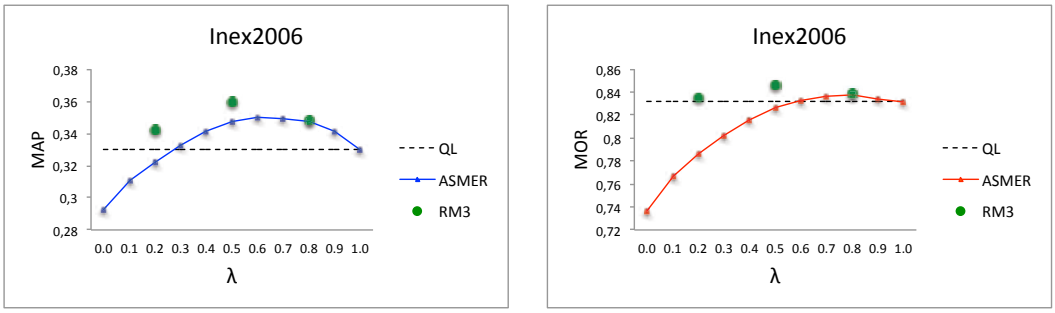
6.3.2.1 Poids de la requête originale (λ)

La figure 6.4 présente les mesures MAP et MOR pour les quatre collections en fonction de λ. Il est clair de la figure 6.4 qu'ASMER est sensible à λ, ce qui n'est pas une exception par rapport aux approches semblables qui combinent d'une manière linéaire la requête originale et la requête étendue. Nous remarquons qu'en général, ASMER se comporte le mieux pour des valeurs de lambda entre 0.6 et 0.8 concernant le rappel et la précision. La bonne performance d'ASMER sur la collection WT10G peut être obtenue même avec un poids très faible de la requête originale ($\lambda = 0,2$), contrairement à RM3 qui pour le même λ obtient une performance nettement moins bonne sur la collection Wt10g. Pour la collection Inex2006, RM3 a une performance supérieure à ASMER même avec une petite valeur de λ. À part ces deux exceptions (le MAP sur Inex2006 et Wt10g), les courbes d'ASMER et RM3 sont proches dans les autres cas. Ces deux exceptions sont en lien avec les tableaux 6.7 et 6.8 où on a constaté que la collection Inex2006 est celle qui a permis à RM3 d'obtenir la meilleur amélioration en MAP par rapport à QL, alors que pour ASMER c'est la collection Wt10g. Logiquement, quand une approche d'expansion marche bien sur une collection, il y a moins besoin de renforcer la requête originale par une valeur élevée de λ. Bien que ce constat soit visible pour le MAP, il est moins applicable pour le rappel, vu que RM3 et ASMER n'ont pas réussi à trouver suffisamment plus de documents pertinents par rapport aux requêtes sans expansion.

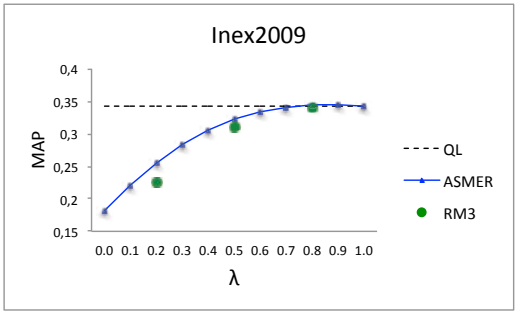
6.3.2.2 Nombre de termes d'expansion pour chaque concept (m)

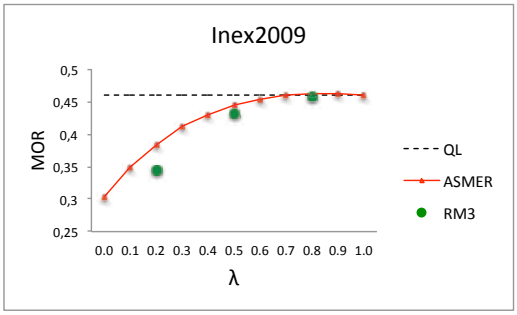
La figure 6.4 présente l'effet de changer le paramètre m sur les mesures MAP et MOR pour les collections de test. De la figure 6.5 nous constatons qu'ASMER est moins sensible au nombre de termes dans les ensembles d'expansions qu'au λ. Dans cette figure, nous ne présentons pas le nombre de termes d'expansion de RM3 car il n'a pas la même signification (termes par concept) qu'avec ASMER.

Il est intéressant de noter que dès le premier mot ajouté, ASMER arrive à dépasser la performance des requêtes sans expansion. L'ajout de nouveaux mots ne détruit pas la performance même pour la collection problématique Inex2009 qui a des requêtes longues à l'origine. Il faut savoir qu'une explication à cette faible sensibilité vient en grande partie de la contrainte de certitude. Car même en fixant le paramètre m à 7, l'ajout de nouveau termes va s'arrêter dès que la similarité avec

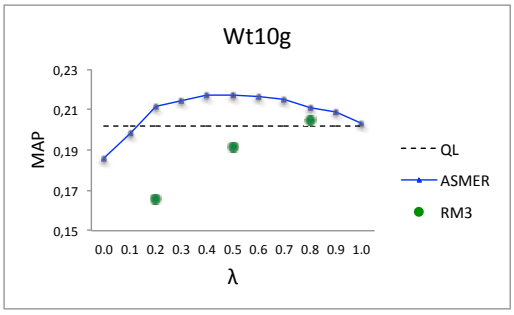


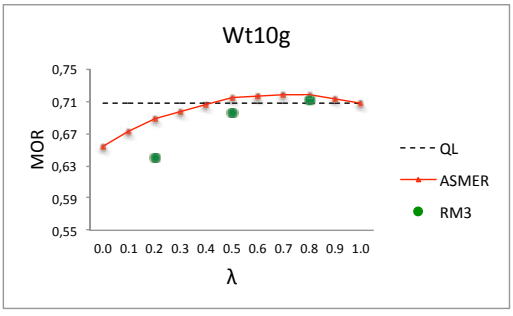
(a)



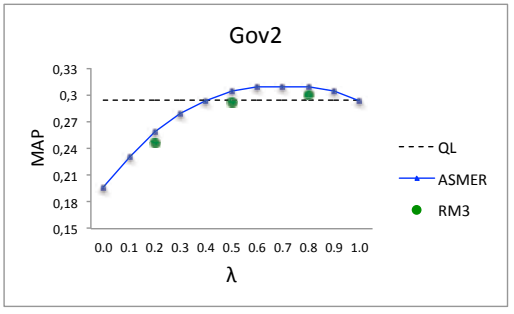


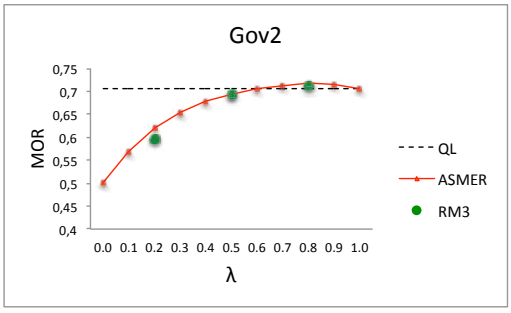
(b)





(c)





(d)

FIGURE 6.4 – L'effet sur *MAP* et *MOR* du poids de la requête originale (λ) dans la requête reformulée

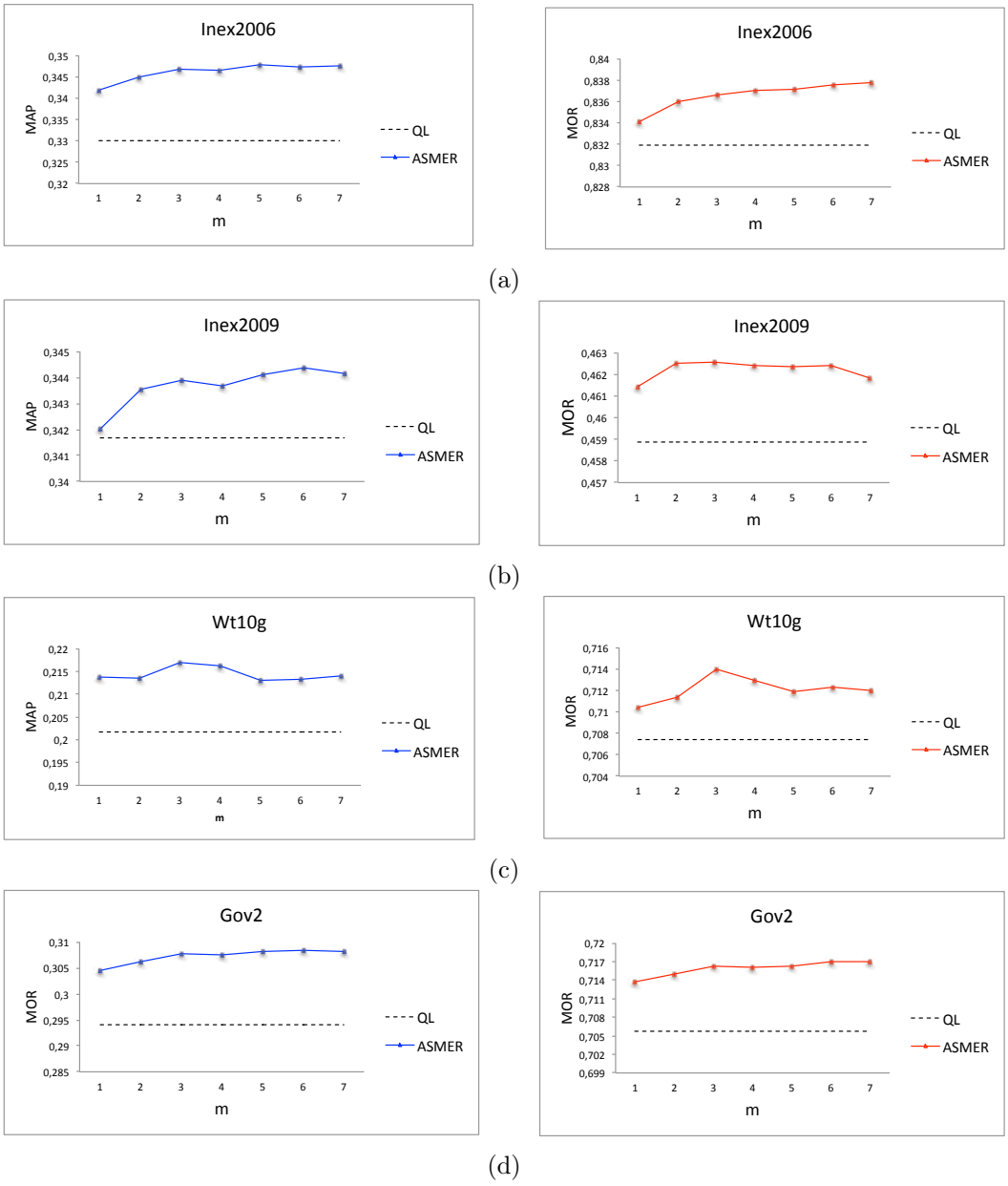


FIGURE 6.5 – L'effet (*MAP* et *MOR*) du nombre de terme dans un ensemble d'expansion (m)

	#+	#0	#−	max	Σ des +	min	Σ des −
Inex2006	50	4	16	+ 91,60 %	+10,85 %	−15,72 %	−4,22 %
Inex2009	37	1	30	+ 15,28 %	+04,00 %	−29,39 %	−4,66 %
Wt10g	49	9	40	+258,47 %	+37,77 %	−78,25 %	−28,05 %
Gov2	101	0	47	+265,51 %	+19,57 %	−64,20 %	−11,33 %

TABLE 6.12 – Le nombre de requêtes améliorées (+), non changées (0) et dégradée (−) en *MAP*, en plus du min, le max et la moyenne (Σ) d'amélioration/dégradation pour les requêtes améliorées/dégradées en *MAP* en utilisant ASMER.

	#+	#0	#−	max	Σ des +	min	Σ des -
Inex2006	42	4	24	+23,94 %	+1,97 %	−09,45 %	−0,97 %
Inex2009	39	1	28	+11,34 %	+2,62 %	−23,89 %	−2,63 %
Wt10g	49	9	40	+36,94 %	+6,45 %	−21,80 %	−4,50 %
Gov2	90	0	58	+92,48 %	+5,75 %	−27,79 %	−3,24 %

TABLE 6.13 – Le nombre de requêtes améliorées (+), non changées (0) et dégradée (−) en *MOR*, en plus du min, le max et la moyenne (Σ) d'amélioration/dégradation pour les requêtes améliorées/dégradées en *MOR* en utilisant ASMER.

le terme original baisse au dessous du seuil de certitude, il est possible que certains termes ne soient même pas étendus. De plus, les termes d'expansion obtenues par YAGO font partie d'ASMER pour toutes les valeurs de m présentées dans la figure 6.5. L'avantage apporté en utilisant des termes provenant des deux ressources par rapport à l'utilisation d'une seule ressource sera abordé dans la section 6.5.

6.4 Évaluation au niveau des requêtes

Nous avons ordonné la performance avec les mesures *MAP* et *MOR* de chaque requête de chaque collection par valeur décroissante dans la figure 6.6. Nous présentons également, dans les tableaux 6.12 et 6.13, le nombre de requêtes améliorés (+), non changées (0) et dégradées (−) pour chaque collection d'un point de vue *MAP* et *MOR*. Nous constatons que le nombre de requêtes améliorées par ASMER est supérieur à celui de requêtes dégradées. Mais ce qui fait la différence pour ASMER est sa capacité de décider de ne pas étendre une requête quand les contraintes ne sont pas satisfaites, ce qui est arrivé 9 fois avec le jeu de requêtes Wt10g. Un autre constat intéressant est que les requêtes qui ont profité le plus de l'approche ASMER ont obtenu une amélioration très importante en *MAP* surtout avec les collections de TREC. Cela s'applique également au *MOR* même si l'amélioration dans ce cas est un peu moins importante.
En observant de plus près, nous constatons que pour les collections Inex2006 et Wt10g, les requêtes les plus améliorées ont bénéficie du traitement de fusion de concepts que nous avons expliqué dans la section 4.3.2 (Fig. 4.3, page 57). Nous

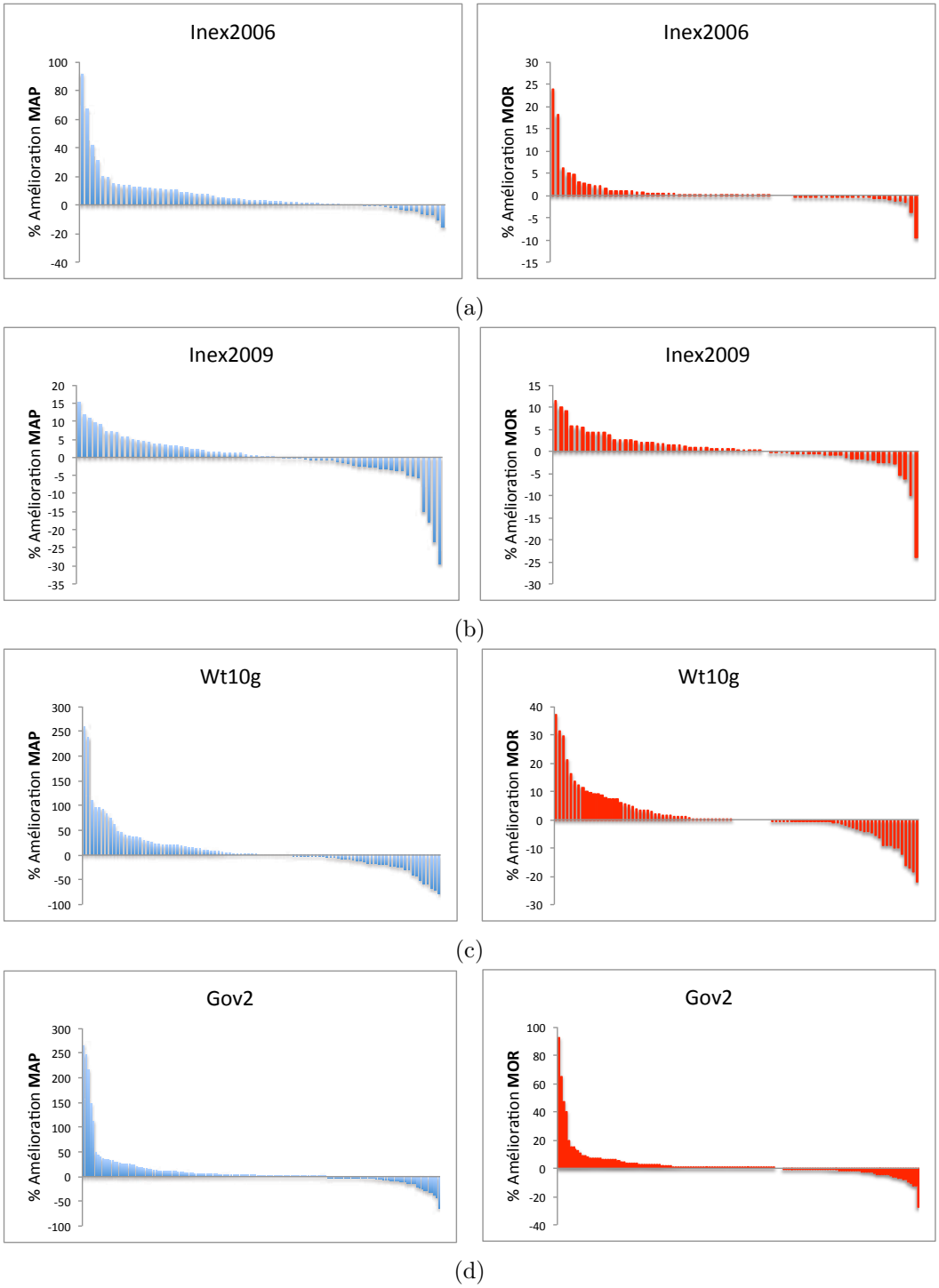


FIGURE 6.6 – Le pourcentage de amélioration/dégradation sur l'ensemble de requêtes par ordre décroissant

présentons l'une de ces requêtes (numéro 646 de la collection INEX2006).

INEX2006, requête 646 ($+91,60\%$)

```
#weight(
  0,8 #combine(Records Management metadata)
  0,2 #combine(
       #weight(1.0 Records 0.51 manage 0.48 microfilm
               0.45 migration 0.45 captured 0.44 processes )
       #weight(1.0 metadata) ) )
```

Selon LSI (sur 20 documents de retour de pertinence et 10 dimensions), le terme « *manage* » est le terme le plus proche de « *Records* » dans l'espace sémantique généré par LSI. Pour cela, ces deux mots se retrouvent dans le même terme étendue, et nous n'obtiendrons au final que deux formules étendues pour les trois mots de la requête. Cette amélioration n'est pas visible en rappel, car nous n'obtiendrons que $0,25\%$ d'amélioration en *MOR* pour cette requête.

Pour la collection Gov2, la requête qui a obtenue $265,51\%$ d'amélioration est la requête 842, qui est une requête courte composé de deux mots.

Gov2, requête 842 ($+265,51\%$)

```
#weight(
  0,8 #combine(David McCullough)
  0,2 #combine(
       #syn( #1(David McCullough) ) ))
```

Cette fois, c'est l'utilisation de l'ontologie YAGO qui a permis l'amélioration car ASMER a détecté que ces deux mots forment une seule entité nommé reconnaissable dans YAGO. Même si aucun autre terme d'expansion n'a été ajouté, le fait d'ajouter la notion de proximité entre les mots originaux a permis une amélioration très importante en *MAP*. La valeur de *MOR* de cette requête a été amélioré de $65,32\%$. Du côté des requêtes dégradées, nous observons la requête 527 qui a été le plus dégradée en *MAP* mais aussi en *MOR* pour la collection Wt10g.

Wt10g, requête 527 ($-78,25\%$)

```
#weight(
  0,8 #combine(info Booker Washington)
  0,2 #combine(
       #weight(1.0 info )
       #weight(1.0 Booker )
       #syn( Washington
```

```
                #1(The Usa)
                #1(Untied States of America)
                #1(The united states of america) ...)
        ))
```

Le problème principale de cette requête est un problème de désambiguisation. L'outil AIDA a échoué à identifier le nom de la personne "Booker Washington" dans l'ontologie YAGO, mais il a identifié le mot Washington en tant que la capitale des État-Unis. Même si le mot Booker n'a pas obtenu des synonymes acceptable selon les contraintes d'ASMER, l'erreur de la désambiguisation a été suffisante pour dégrader largement la performance de cette requête. Pour les autres trois collections, les requêtes les plus dégradées n'avaient pas le souci de désambiguisation des entités nommées, mais ils ont reçu des « mauvais » termes par LSI. Ces mauvais termes ont été soit pas assez spécifiques (malgré notre contrainte de spécificité[3]), soit loin du sujet de la requête, où dans ce cas la requête sans expansion avait une faible précision à la base, elle a donc fourni des documents pas pertinents à LSI.

6.5 Approches individuelles d'ASMER

ASMER est une combinaison de deux approches différentes, l'utilisation d'une ontologie, et les documents de pseudo retour de pertinence. Nous analysons dans cette partie chacune de ces approches d'une manière individuelle où chaque approche est utilisée pour étendre tous les termes de la requête. Nous comparons donc la performance entre ASMER et chaque approche seule.

6.5.1 Yago

Nous considérons dans cette section l'utilisation de l'ontologie Yago pour l'expansion de la requête. C'est-à-dire que cette approche n'utilise aucune information sur les documents, elle ajoute des appellations aux entités nommées de la requête, tandis que les autres termes sont également étendus avec la partie WordNet de Yago (Sect. 4.3.1). Une spécificité de cette approche est liée à la désambiguisation. Nous rappelons que pour trouver des termes d'expansion, nous cherchons le concept qui correspond à chaque terme de la requête pour utiliser les termes sémantiquement proches aux termes de la requête en utilisant la relation « rfd :label » dans Yago. Nous testons dans cette section l'utilité de la désambiguisation par rapport à la technique simple qui choisit le concept qui représente le sens le plus commun dans l'ontologie pour chaque terme de la requête. Dans la figure 6.7, nous présentons la performance de ces deux méthodes que nous appelons YagoDis pour l'approche d'expansion de la requête par YAGO en utilisant la désambiguisation, et YagoNoDis

3. Par exemple, l'une des requêtes (Gov2 #831) a le mot « three » comme terme d'expansion. Ce mot se trouve au niveau 7 dans le taxonomie de WordNet, il est donc considéré comme un terme acceptable par ASMER.

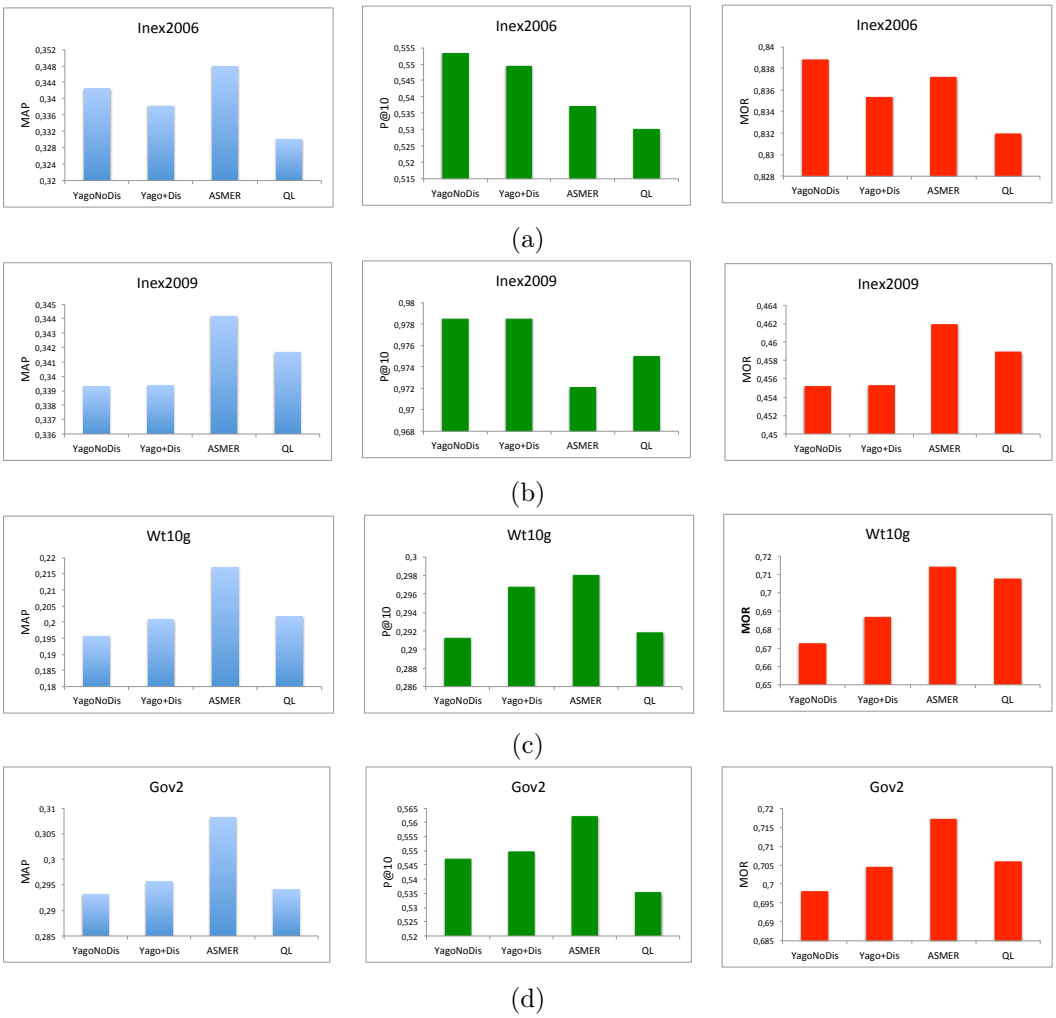


FIGURE 6.7 – La performance en *MAP*, *P@10* et *MOR* en utilisant YAGO, avec et sans désambiguïsation, par rapport à l'utilisation d'ASMER (Yago+LSI)

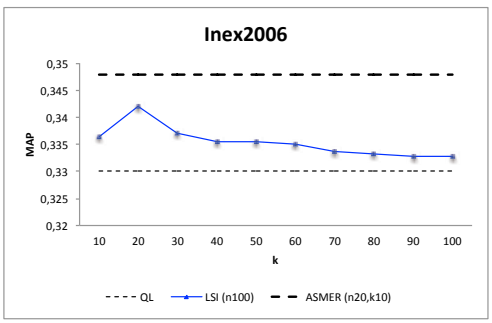
pour l'approche qui choisit le sens le plus commun. Avec ces deux méthodes nous présentons également la performance du modèle de base avec les requêtes sans expansion (QL) et celle d'ASMER (avec les paramètres du tableau 6.6, page 100). La performance est évaluée selon trois mesures : le *MAP*, *P@10* et *MOR* sur les quatre collections de tests. En observant la figure 6.7, nous constatons que l'ajout d'une technique locale (ASMER) permet d'améliorer le *MAP* que ce soit par rapport aux requêtes sans expansion, ou par rapport à la seul utilisation d'une ontologie (avec ou sans désambiguisation). Le même constat concerne le rappel (mesure MOR), où ASMER trouve plus de documents pertinents que l'approche YAGO à l'exception de la collection Inex2006. Nous remarquons que pour cette collection il est plus utile de choisir le sens le plus commun que de faire la désambiguisation. En revenant vers le tableau 6.3 (page 97), nous constatons que les requêtes utilisées avec la collection Inex2006 contiennent le plus petit pourcentage en requêtes longues et le plus

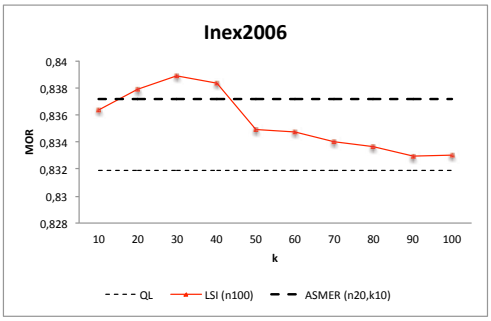
grand pourcentage en requêtes de 2 à 3 mots. Ainsi, pour ces requêtes 2 à 3 mots, la désambiguisation est délicate, car il y a un mot ou deux avec le mot qu'on souhaite desambiguiser dans la requête, ce qui ne garantit pas que le sens le plus commun sera choisi, et en même temps ne suffit pas pour une désambiguisation acceptable. L'évaluation de *P@10* dans la figure 6.7 montre qu'une amélioration importante peut être obtenue avec l'expansion par une ressource sémantique externe uniquement. Cette amélioration est supérieure à celle obtenue par ASMER pour les collections INEX et assez proche de l'amélioration d'ASMER pour les collections TREC.

6.5.2 LSI

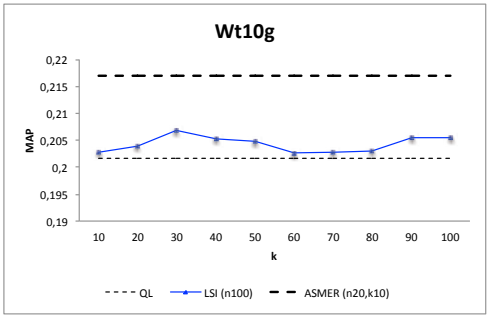
Dans cette partie, nous nous intéressons à un problème souvent lié à la technique LSI : le choix de nombre de dimension. Il est bien connu que l'un des obstacles les plus importants à l'utilisation de LSI est le coût très élevé, en terme de complexité, de la décomposition de la matrice termes/documents. Avec les petits nombres de documents de retour de pertinence que nous avons utilisés avec ASMER, ce problème ne se pose pas. Par contre, la question est de savoir si la technique SVD et la réduction de dimension avec une matrice contenant un nombre très petit de document a du sens. En fait, en observant les statistiques de nos collections de tests (Table. 6.1, page 96), nous pouvons voir que la taille moyenne de documents est entre 400 et 900 termes. En générale, le nombre total de termes dans les documents de retour de pertinence a dépassé les 8000 termes dans toutes nos expériences. Ce nombre vaut également le nombre de lignes et des colonnes de la matrice U généré par SVD (voir l'annexe A). Cette matrice (U) est celle que nous utilisons comme notre thésaurus. Réduire le nombre de colonnes de cette matrice à 10, 20 ou 30 dimensions a donc un effet non négligeable, ce que nous allons montrer par l'expérience suivante.

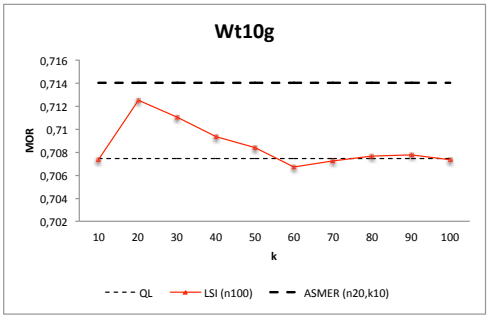
Pour voir l'effet du nombre de documents et le nombre de dimensions de LSI, nous avons fait l'expérience de prendre 100 documents de retour de pertinence et de varier le nombre de dimensions entre 10 et 100 avec un pas de 10. Cette expérience a été testée sur les collections Inex2006 et Wt10g. La performance en *MAP* et en *MOR* est présentée par la figure 6.8. Dans cette figure, nous montrons également la performance du modèle de base QL et celle d'ASMER avec les paramètres cités dans le tableau 6.6 (page 6.6), ce qui correspond à 20 documents de retour de pertinence pour Inex2006, 30 pour Wt10g, et 10 dimensions (k) pour les deux cas. Nous constatons que l'utilisation de 100 documents de retour de pertinence peut améliorer la précision et le rappel, mais il est moins bien que l'utilisation de 20 ou 30 documents par l'approche ASMER. Avec un petit nombre de documents, nous avons plus de chance que ces documents soit focalisés sur les thématiques évoquée par la requête, alors qu'avec 100 documents, le risque de dérive de la requête augmente, ce qui dégrade la précision mais peut être bénéfique en rappel (ce qui est le cas pour la collection Inex2006 avec 30 dimensions). Ce constat est contradictoire avec les expériences de Zhao and Callan [2010], qui ont choisi un nombre bien plus grand (200 documents pour la collection Wt10g) pour calculer la nécessité de termes dans une requête en utilisant LSI. Il serait intéressant de savoir l'effet de l'utilisation de





(a)





(b)

FIGURE 6.8 – La performance en *MAP*, *P@10* et *MOR* en utilisant YAGO, avec et sans désambiguïsation, par rapport à l'utilisation d'ASMER (Yago+LSI)

20 ou 30 documents dans leur contexte sur la nécessité de termes.

6.6 Résumé

Nous avons présenté dans ce chapitre l'évaluation expérimentale d'ASMER, notre approche sémantique pour l'expansion et la reformulation de requêtes, sur quatre collections standard de tests. En utilisant le modèle de vraisemblance de la requête comme modèle de recherche, nous avons comparé la performance d'ASMER avec l'utilisation des requêtes sans expansion (QL) et l'utilisation des requêtes étendues par le modèle de pertinence (RM3), qui est un modèle réputé et souvent utilisé comme référence dans l'état de l'art. Nous avons constaté qu'ASMER a dépassé les autres modèles d'une façon significative surtout en précision, alors que l'amélioration est moins importante selon la mesure *MOR* que nous avons proposé pour l'évaluation du rappel. Après cette évaluation, nous avons étudié la sensibilité d'ASMER aux paramètres, où nous avons trouvé qu'il se comporte le mieux pour des valeurs λ entre 0.6 et 0.8, alors qu'il donne déjà de bons résultats à partir d'un seul mot d'expansion. En observant le comportement des requêtes individuelles par rapport à ASMER, nous avons montré que l'effet d'ASMER est bénéfique sur la majorité des cas. Nous avons prouvé par la suite l'efficacité du mélange des approches YAGO et LSI par rapport à l'utilisation de chaque approche individuellement. En plus, nous avons analysé les facteurs les plus importants pour chaque approche, qui sont l'effet de la désambiguisation pour l'approche YAGO et l'effet du nombre de documents et le nombre de dimensions pour LSI. Nous avons constaté l'efficacité de la désambiguisation sur les requêtes longues, et l'avantage de l'aspect sémantique que LSI peut apporter même avec un petit nombre de documents de pseudo retour de pertinence.

Chapitre 7

Conclusion

Sommaire

7.1 Synthèse des contributions

Notre travail s'intéresse à l'utilisation de la sémantique pour l'expansion et la reformulation des requêtes Web. Dans ce mémoire, nous avons commencé par une présentation de l'état de l'art sur les notions générales de la sémantique, de la recherche d'information, et les différentes approches qui peuvent être utilisées pour introduire le sens dans un système de recherche d'information. Ensuite, nous avons fait une étude plus détaillée sur les approches qui modifient automatiquement les requêtes, ce qui nous a permis de catégoriser ces approches et de prendre une position pour développer notre propre vision sur la reformulation et l'expansion sémantique des requêtes. Ainsi, nous avons proposé deux approches qui prennent la notion de concept d'un point de vue différent. La première approche, basée sur ce qu'on appelle des concepts explicites, lie les termes d'une requête avec des concepts dans une ontologie. La deuxième approche, utilise la technique LSI sur des documents de retour de pertinence pour créer un espace sémantique, dans lequel on définit des concepts dits implicites. Pour les deux approches, une fois les concepts définis, c'est la relation de similarité que nous utilisons pour extraire les termes d'expansion correspondant aux termes de la requête. Dans le but de profiter de ces deux approches, nous avons proposé l'approche sémantique mixte d'expansion et de reformulation (ASMER) que nous accompagnons d'une stratégie sélective qui permet de filtrer les « mauvais » termes d'expansion. Avant d'évaluer les différents aspects de nos approches, nous avons fait une étude sur l'évaluation du rappel, car nous avons trouvé que les mesures actuelles ne couvrent pas les contraintes nécessaires pour évaluer cette notion. Cette étude nous a amené à proposer la mesure MOR qui a fait partie de nos mesures d'évaluation dans la partie expérimentale de la thèse. Or dans cette partie, nous avons évalué notre idée sur la modélisation sémantique des requêtes sous plusieurs aspects : la comparaison avec une approche de l'état de l'art, la sensibilité aux paramètres et l'effet du mélange des deux approches de concepts explicites et implicites. Pour les quatre collections de tests que nous avons

utilisées, nous avons constaté qu'en MAP, ASMER est meilleure que l'utilisation des requêtes sans expansion, et elle est supérieure, dans la majorité des cas, à une approche performante de l'état de l'art (RM3). Bien qu'ASMER se comporte moins bien en rappel, elle a réussi à trouver le premier document pertinent avant les autres approches pour toutes nos collections de tests.
En conclusion, ASMER est une approche d'expansion et de reformulation de la requête adaptée au contexte Web, mais qui peut être aussi intéressante dans un contexte interactif où l'utilisateur peut visualiser et modifier la requête reformulée. Nos arguments sont les suivants :

— ASMER ne nécessite pas de statistiques sur la collection de documents pour générer les termes d'expansion. Une liste de documents liés à la requête (retour de pertinence réel ou implicite) et une ressource pour les entités nommées suffisent.
— La performance d'ASMER en MAP est stable par rapport à celle de RM3, et cela avec des collections de documents et des requêtes de tailles différentes.
— La requête générée par ASMER peut être facilement interprétée, car les liens entre les termes de la requête et les termes d'expansion sont explicitement représentés par la requête reformulée. Ainsi, un utilisateur est capable de comprendre, ajouter, supprimer, ou modifier la requête reformulée plus facilement qu'avec une approche d'expansion basée sur le sac de mots.
— ASMER n'étend pas les requêtes d'une manière systématique. Uniquement les « bons » termes d'expansion sont utilisés pour étendre une requête.

7.2 Réponses aux questions de recherche

Nous proposons des réponses dans cette section aux questions de recherche que nous avons posées dans l'introduction (Chap. 1).

Question 1 : Pourquoi a-t-on besoin d'une nouvelle approche de reformulation de la requête ?

En étudiant l'état de l'art, nous avons constaté que la plupart des approches d'expansion de la requête ajoutent les termes d'expansion sous forme de sac de mots, où, au mieux, ce sac de mot est combiné d'une manière linéaire avec la requête originale (également un sac de mots). De plus, peu d'études s'intéressent à l'expansion des entités nommées malgré la présence de plusieurs travaux sur leur importance dans les requêtes Web. Par ailleurs, nous avons estimé qu'il y a besoin d'étudier différemment l'utilisation de la sémantique pour améliorer les requêtes. Sur ce sujet, nous n'avons pas trouvé une approche dans l'état de l'art qui prend soin de choisir les bons termes d'expansion en respectant les liens entre les concepts qui les contiennent et les termes originaux de l'utilisateur, tout en prenant en compte l'importance des entités nommées.

Question 2 : Comment connaitre les concepts d'une requête courte de l'utilisateur ?

Pour connaitre le besoin d'information, il est important d'utiliser d'autres ressources que la requête courte. Notre hypothèse est qu'avec une ressource sémantique et une technique locale, nous pouvons extraire des termes d'expansion qui appartiennent aux concepts de la requête, sans avoir accès aux information supplémentaires sur le besoin d'informations. Pour cela, nous avons proposé une approche sémantique mixte qui permet de profiter de ces deux types d'approches en nous basant sur les points suivants :

— L'état de l'art montre l'importance des entités nommées. Le traitement de ces éléments avec une technique locale de l'expansion de la requête n'est pas efficace, car avec une telle technique il est compliqué de considérer les termes composés de plusieurs mots, ce qui est le cas de beaucoup d'entités nommées.

— L'utilisation d'une ontologie pour étendre tous les termes de la requête ne permet pas de considérer les liens sémantiques entre les termes de la requête, même si la technique de désambiguïsation prend en compte tous les termes de la requête. Ainsi, en l'absence d'autres informations sur le besoin en information, l'utilisation d'une technique locale est le seul moyen de dévoiler les thématiques auxquelles certains termes appartiennent.

Ainsi, pour découvrir les concepts d'une requête, notre approche mixte distingue les entités nommées des autres termes de la requête. Les entités nommées sont identifiées dans une ressource sémantique adaptée, et les autres termes sont liés aux concepts généré par LSI qui utilise les documents de retour de pertinence récupérés par l'exécution de la requête originale [Audeh et al., 2014b].

Question 3 : Comment exprimer un concept dans une requête textuelle ?

Une fois idéntifiés les concepts explicites et implicites auxquels les termes de la requête appartiennent, la relation de similarité sémantique entre les termes originaux et les autres termes du même concept permet de limiter les effet de dérive de la requête. Ainsi, pour les concepts explicites nous prenons les termes sémantiquement similaires aux entités nommées, alors que pour la technique locale, nous prenons les termes les plus proches des termes de la requête dans l'espace sémantique généré par LSI. Pour ajouter ces termes d'expansions d'origines différentes, nous considérons les points suivants :

— Afin de garder la possibilité d'interpréter et modifier les requêtes étendues, l'utilisateur doit être capable de comprendre pourquoi un terme d'expansion a été ajouté. Pour cela, nous partons du principe de respecter les termes employés par l'utilisateur, en supposant que chaque terme original appartient à un concept. Ces concepts sont conservés dans la requête étendue, où les termes originaux sont remplacés par leur formules étendues. Pour exprimer

une telle formule, les opérateurs utilisés dépendent de l'origine de ces termes (concepts explicites ou implicites).

— Les termes d'expansion des entités nommées s'agit de termes sémantiquement similaires qui décrivent le même objet. Pour cela, l'existence de chacun de ces terme contribu de la même manière aux calculs du score d'un document que les termes similaires. Ces termes ne sont pas pondérés, ils ne peuvent donc pas être ordonnés.

— Les termes d'expansion provenant de la technique locale ont été choisis par leur haute similarité aux termes de la requête dans un espace sémantique construit avec les documents de retour de pertinence. Plus un terme d'expansion est proche du terme original plus importante est la relation sémantique entre ces termes. Malgré leur similarité sémantique, les termes d'expansion de LSI ne peuvent pas être considérés comme de vrais synonymes. Pour cela, c'est la combinaison de ces termes avec les termes originaux qui doit contribuer au score d'un document, tout en donnant plus de poids aux termes qui sont le plus proches des termes originaux.

— Les entités nommées ou les termes composés de plusieurs mots doivent être considérés comme des expressions exactes. Exiger la proximité maximale entre les mots de la même entité peut éliminer des documents pertinents dans certains cas, mais il diminue le risque de dérive de la requête.

— L'origine d'un terme d'expansion ne garantit pas sa qualité. Pour qu'un terme d'expansion soit utile, il doit être assez spécifique et non redondante. De plus, on doit être assez sûr qu'il appartient au même concept que le terme original. Pour cela, des contraintes de spécificité et de certitude sont nécessaires pour n'ajouter que les « bons » termes comme termes d'expansion.

En appliquant ces règles au modèle de vraisemblance de requête, et en utilisant les opérateurs fournis par l'outil Indri, nous exprimons les expressions exactes par l'opérateur #1, les ensembles d'expansion des concepts explicites par #syn, et les ensembles d'expansion des concepts implicites par #weight en utilisant la similarité entre le terme d'expansion et le terme original comme une pondération [Audeh et al., 2014a], [Audeh et al., 2014c].

Question 4 : Comment évaluer le rappel ?

Notre étude de l'état de l'art de l'évaluation a montré qu'aucune mesure n'a été capable de reproduire un jugement humain dans un contexte rappel. C'est-à-dire que les mesures existantes sont soit biaisées par les rangs de documents trouvés et n'arrivent donc pas à favoriser un système qui trouve plus de documents pertinents, soient ne considèrent pas les rangs, et un système qui rend tous les documents de la collection peut donc obtenir le meilleur score. Une étude détaillée de ces mesures nous a amené à proposer la mesure MOR qui spécifie bien les besoins d'évaluation dans un contexte de rappel. Cette mesure a été capable de reproduire la préférence humaine sur des exemples simples, et d'avoir une corrélation forte avec la notion

pure du rappel sans être complètement identique à celle-ci [Audeh et al., 2013b].

7.3 Perspectives

Notre travail attire l'attention sur plusieurs points importants concernant l'expansion et la reformulation sémantique de requêtes, notamment le rôle non négligeable des entités nommées, l'intérêt d'ajouter les termes d'expansion en utilisant d'autres méthodes que le sac de mot pour permettre une représentation plus adaptée à la notion de concepts, et l'importance de contrôler la qualité de termes d'expansion pour que l'ajout de termes ne se soit pas systématique. Il fallait faire beaucoup de choix pour explorer ces trois aspects. Certains de ces choix ont été évidents alors qu'il reste intéressant d'explorer les autres possibilités dans d'autres cas.

Premièrement, bien que nous pensons que notre choix de l'ontologie YAGO est bien adapté au contexte Web, nous pensons qu'un choix plus sophistiqué de la relation sémantique à utiliser pour trouver les termes d'expansion peut être envisageable. Comme par exemple de choisir la relation sémantique en fonction du type de l'entité nommée (personne, lieu,...) ou en fonction des autres termes présents avec elle dans la requête.

En deuxième lieu, nous avons pris l'hypothèse forte que tous les concepts ont la même importance, en nous appuyant sur le fait que les concepts redondants seront fusionnés, et que les termes qui n'ont pas de termes d'expansion acceptables dans leur concept ne serons pas étendus. Nous avons vu que même sans pondérer les formules étendues, nous avons réussi à améliorer la performance par rapport aux requêtes non étendues, et à être statistiquement meilleur qu'une approche forte d'expansion de la requête. Pour pouvoir pondérer des formules étendues de nature différente, il faut trouver une logique qui permet de comparer un concept explicite avec un concept implicite pour pouvoir leur donner un poids qui exprime l'importance de l'un par rapport à l'autre dans une requête reformuler. Nous avons l'intention de considérer ce problème de plus près dans nos prochaines études.

Concernant le troisième aspect, nous sommes convaincus de l'importance du choix des termes d'expansion, et de son avantage par rapport à l'expansion systématique pour toutes les requêtes par tous les termes d'expansion disponibles. Notre initiative était de proposer les contraintes de spécificité et de certitude, ces contraintes sont en harmonie avec notre stratégie de considérer le point de vue humain et d'être indépendant des statistiques de la collection de documents et de la spécificité du modèle de recherche. Nous pensons, néanmoins, qu'une étude sur le lien éventuel de ces contraintes avec d'autres mesures de qualité des termes, comme celle de Cao et al. [2008] et Zhao and Callan [2010] est intéressante.

Annexe A

Indri et les modèles de langue en RI

A.1 Principe des modèles de langues

Les modèles de langues en recherche d'information sont des modèles probabilistes. Ils sont fondés sur l'hypothèse que le choix des mots qu'un humain fait pour générer un texte est une procédure qu'on peut modéliser : on suppose qu'une séquence de mots est produite par un modèle statistique, qu'on appelle le modèle de langue. A l'origine, ce principe a été utilisé dans des applications liées au traitement de la langue, comme la reconnaissance automatique de la parole et la traduction automatisée [Rosenfeld, 2000]. En recherche d'information, les modèles de langues supposent que pour générer sa requête, l'utilisateur cherche à deviner les termes qui ont été utilisés pour rédiger un document pertinent. Pour appliquer cette idée à la recherche de documents, plusieurs approches ont été proposées comme l'approche par vraisemblance de la requête et l'approche de divergence de Kullback-Leibler. Dans notre travail, seul le modèle de vraisemblance de la requête a été pris en compte.

A.2 La mise en correspondance par vraisemblance de la requête

Dans les modèles probabilistes, pour calculer le score d'un document d par rapport à une requête Q nous cherchons à estimer la probabilité $p(d|Q)$. En appliquant la règle de Bayes, cette probabilité correspond à l'équation A.1.

$$p(d|Q) = \frac{p(Q|d)p(d)}{p(q)} \propto p(Q|d) \tag{A.1}$$

où $p(q)$ est une constante qu'on peut ignorer car elle est indépendante des documents. $p(d)$ est la probabilité a priori pour choisir un document. Elle peut être utilisée pour influencer le choix des documents selon un certain critère, comme par exemple de privilégier les documents qui ont plus de citations. Souvent dans la recherche d'information ad hoc, cette probabilité est considérée comme uniforme, ce qui signifie que tous les documents ont a priori la même chance d'être dans une liste de résultats. Pour cela, $p(d)$ de l'équation A.1 est également ignorée. Ainsi, l'évaluation d'un document par rapport à une requête revient à estimer la probabilité $p(Q|d)$.

Le modèle de vraisemblance de la requête [Ponte and Croft, 1998; Ponte, 1998], souvent appelé QL (de *Query Likelyhood*), considère que le degré de pertinence d'un

document pour une requête est la vraisemblance (probabilité) que le modèle qui a généré ce document M_d soit utilisé pour générer la requête, ce qu'on exprime dans le modèle de langue par $p(Q|M_d)$. De plus, le document est considéré comme un sac de mots, et M_d est une distribution des mots à partir de laquelle le document a été généré. La probabilité $p(Q|M_d)$ peut être calculée par l'équation A.2.

$$p(Q|M_d) = \prod_{q \in Q} p(q|M_d) \tag{A.2}$$

La méthode la plus fréquente pour calculer $p(q|M_d)$ est de calculer le nombre d'occurrences de q dans d ($tf_{q,d}$) tout en normalisant par le nombre de mots dans le document $|d|$ (eq. A.3).

$$p(q|M_d) = \frac{tf_{q,d}}{|d|} \tag{A.3}$$

Un dernier détail sur ce modèle concerne le produit dans l'équation A.2. Le problème est que l'absence d'un terme de la requête va donner un score 0 à un document même s'il contient d'autres termes de la requête. Pour éviter ce problème, plusieurs méthodes de lissage peuvent être utilisées. L'une des méthodes les plus efficaces est celle de Dirichlet [Zhai and Lafferty, 2004] (et c'est la méthode de lissage utilisée par défaut dans Indri) comme présenté par l'équation A.4.

$$p_{dir}(q|M_d) = \frac{|d|}{|d| + \mu} p(q|M_d) + \frac{\mu}{|d| + \mu} p(q) \tag{A.4}$$

où μ est une constante, et la probabilité $p(q)$ d'observer q dans la collection de documents D est calculée par l'équation A.5.

$$p(q) = \frac{tf_{q,D}}{|D|} \tag{A.5}$$

où $|D|$ est le nombre d'occurrences de mot dans la collection de documents D.

A.3 Le modèle structuré de langue

Le modèle structuré de langue [Metzler and Croft, 2004] a été proposé pour combiner le pouvoir représentatif des réseaux d'inférence [Turtle and Croft, 1990] avec la bonne performance du modèle de vraisemblance de la requête [Ponte and Croft, 1998]. Cette combinaison a été appelée Indri [Strohman et al., 2004], et elle est restée le modèle de recherche par défaut dans l'outil Indri.

Le réseau d'inférence selon Turtle and Croft [1990] est un réseau bayésien [1] dans lequel les nœuds représentent les probabilités liées aux événements d'une procédure de recherche d'information. On distingue deux parties dans le réseau, la partie « documents » et la partie « requêtes » (Fig. A.1). Dans la partie « documents » du réseau, Turtle and Croft [1990] séparent l'aspect abstrait d'un document d_i de sa

1. Un réseau bayesien est un modèle graphique probabiliste représentant des variables aléatoires sous la forme d'un graphe orienté acyclique (*définition Wikipédia*).

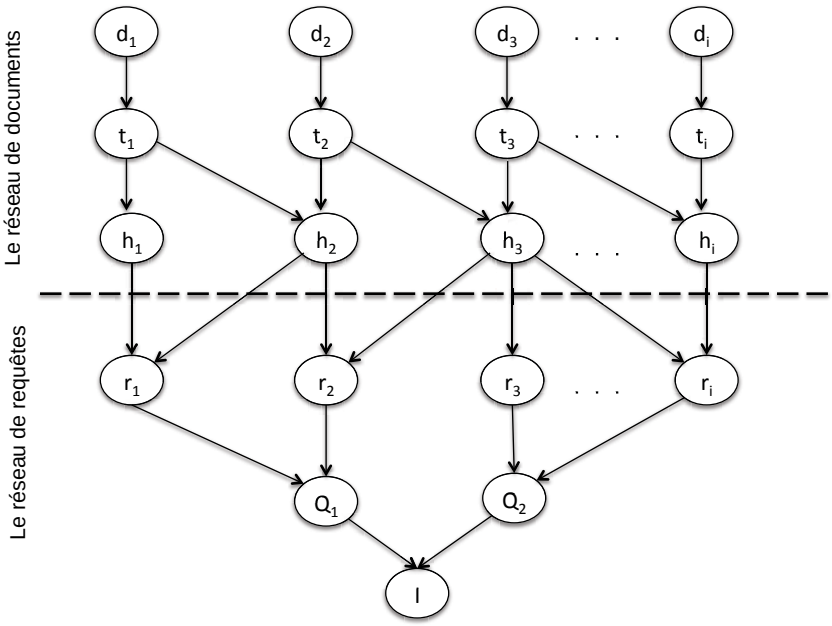


FIGURE A.1 – L'exemple de Turtle and Croft [1990] d'un réseau d'inférence

représentation textuelle t_j et des éléments d'index h_k qui peuvent être extraits par des techniques différentes (ex. mots simples, extraction d'expressions, indexation manuelle, ...). La partie « requêtes » a comme but de représenter le besoin d'information de l'utilisateur I. Ce besoin peut être exprimé par plusieurs requêtes (Q_1 et Q_2 dans notre exemple). Chaque requête est la combinaison de plusieurs concepts. Les concepts des requêtes peuvent correspondre aux mots simples ou combinés (par le lien de proximité par exemple) de la requête. Dans le cas le plus simple, chaque nœud r est lié à un seul nœud h.

Metzler and Croft [2004] ont proposé une version simplifiée de ce réseau d'inférence. Ils considèrent un nœud d_i comme l'événement qu'un document (dans son intégralité) soit observé ou non. Concrètement, si un nœud r représente une expression, l'événement correspondant à ce nœud est que cette expression soit extraite d'un ensemble de la collection de documents. Dans ce modèle, chaque représentation d'une requête est liée à tous les nœuds de type document. Ainsi, pour exprimer la mise en correspondance entre une représentation r de la requête et un document d_j, on cherche à estimer la probabilité $p(r = true|d_j = true, d_{i\neq j} = false)$ d'où le nombre exhaustif de flèches entre les nœuds r et les nœuds d. Cette probabilité, qu'on va appeler $b(r, d)$ pour simplifier, est estimée dans le travail orignal de Turtle and Croft [1990] sur la base tf/idf comme présenté par l'équation suivante :

$$b(r, d) = tf_{r,d} idf_r \tag{A.6}$$

Pour éviter les probabilités nulles qui éliminent les documents qui n'ont pas tous les mots de la représentation, une étape de lissage est souvent utilisée. Dans le système

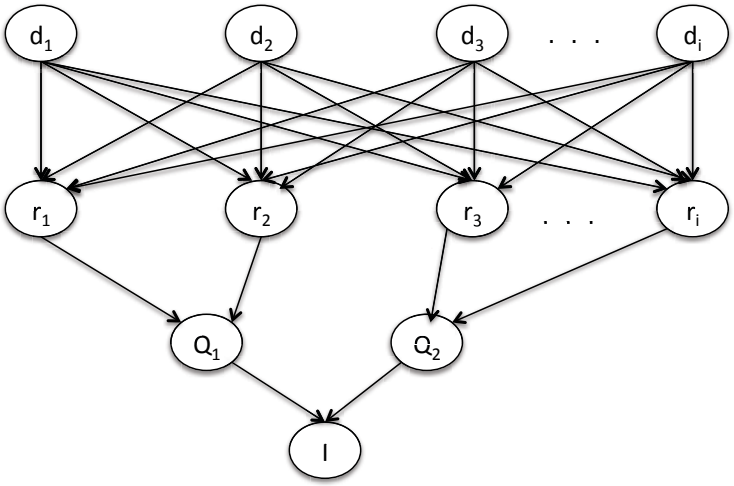


FIGURE A.2 – L'exemple simplifié du réseau d'inférence [Metzler and Croft, 2004]

Inquery, le lissage utilisé est présenté par l'équation A.7.

$$b(r,d) = \lambda + (1-\lambda) tf_{r,d} idf_r \tag{A.7}$$

où λ est une constante arbitraire (par défaut $\lambda = 0,6$). Pour combiner les croyances sous ce modèle, Inquery proposait plusieurs possibilités, comme la multiplication (#and), la somme (#sum) où la somme pondérée (#wsum) avec laquelle on peut associer une importance (pondération) à un nœud.

La combinaison faite par Indri de ce modèle d'inférence avec QL correspond à remplacer le calcul tf/idf de l'équation A.7, par le principe de vraisemblance de l'équation A.4. Ainsi, avec la modélisation par un réseau d'inférence, Indri a pu hériter de tous les opérateurs d'Inquery. Par contre, les auteurs ont justifié que la combinaison de croyances par la somme (#wsum) utilisée par défaut dans Inquery n'est pas performante avec la nouvelle considération de la vraisemblance. Pour cela, les auteurs proposent un nouvel opérateur (#weight [2]). Pour combiner les croyances par cet opérateur on considère l'équation A.8.

$$b_{\#weight}(q,d) = \prod_i b(r_i,d)^{w_i} \tag{A.8}$$

Le fait de considérer le poids ($w_i = 1$) pour tous les termes de la requête, nous ramène donc à la fonction du modèle de vraisemblance de la requête (eq. A.2) et correspond également à l'opérateur #combine dans la langage d'Indri. En plus des représentations correspondant aux opérateurs #combine #weight et #wsum, Indri propose des opérateurs de proximité qui permettent d'exprimer des expressions en exigeant (ou non) l'ordre d'apparition des mots dans ces expressions (`http://ciir.cs.umass.edu/~metzler/indriquerylang.html`).

2. Dans le papier de Metzler and Croft [2004] ce nouveau opérateur a été appelé #wand.

Annexe B

L'analyse sémantique latente (LSI)

L'analyse ou l'indexation sémantique latente LSI « Latent Semantic Indexing » dépend de la décomposition de la matrice de termes et des documents par la technique de décomposition par valuers singulières SVD « *Singular Value Decomposition* ». Nous commençons donc par introduire le principe de SVD, puis son application selon LSI.

B.1 SVD

L'idée de SVD est que chaque matrice rectangulaire $A_{\{m,n\}}$ peut être écrite de la façon suivante :

$$A_{\{m,n\}} = U_{\{m,m\}} S_{\{m,n\}} V^T_{\{n,n\}} \tag{B.1}$$

où S est la matrice diagonale qui contient les valeurs singulières de la matrice A, U contient les vecteurs propres orthonormés de la matrices AA^T, et V contient ceux de la matrice A^TA. Les colonnes de ces trois matrices sont ordonnée d'une manière décroissante par rapport aux valeurs propres respectives. Pour mieux comprendre les détails nous prenons l'exemple de la matrice A [1] (eq. B.2).

$$A = \begin{pmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{pmatrix} \tag{B.2}$$

Pour calculer la matrice U, il faut d'abord trouver les valeurs et les vecteurs propres de la matrice AA^T. Nous commençons donc par calculer la matrice carrée $L = AA^T$ (eq. B.3) :

$$L = AA^T = \begin{pmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{pmatrix} \begin{pmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 11 & 1 \\ 1 & 11 \end{pmatrix} \tag{B.3}$$

Selon la définition, une valeur λ et un vecteur $\overrightarrow{v}$ sont valeur propre et vecteur propre de la matrice carré L s'ils satisfont l'équation $L\overrightarrow{v} = \lambda\overrightarrow{v}$. En appliquant cette équation à notre exemple, nous obtenons l'équation B.4.

$$\begin{pmatrix} 11 & 1 \\ 1 & 11 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \lambda \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{B.4}$$

1. Exemple légèrement modifié de http://www.ling.ohio-state.edu/~kbaker/pubs/Singular_Value_Decomposition_Tutorial.pdf.

Cela correspond aux équations suivantes (B.5)

$$\begin{aligned}(11-\lambda)x_1 + x_2 &= 0\\ x_1 + (11-\lambda)x_2 &= 0\end{aligned} \tag{B.5}$$

Pour trouver les valeurs de λ, nous mettons à 0 le déterminant de la matrice des coefficients comme dans l'équation B.6.

$$\begin{vmatrix} (11-\lambda) & 1 \\ 1 & (11-\lambda) \end{vmatrix} = 0 \tag{B.6}$$

La solution de l'équation B.6 nous donne deux valeurs propres possibles : $\lambda_1 = 10$ et $\lambda_2 = 12$. En appliquant la valeur λ_1 aux équations B.4, nous obtenons $x_1 = -x_2$, donc plusieurs valeurs possibles de ces deux variables, nous considérons par exemple $x_1 = 1$ et $x_2 = -1$ ce qui donne le premier vecteur propre $\overrightarrow{v_1} = (1, -1)$. De la même façon, nous obtenons le vecteur propre $\overrightarrow{v_2} = (1, 1)$ pour λ_2 (plusieurs solutions possibles aussi en considérant λ_2). Ces deux vecteurs sont les colonnes de de la matrice U' (eq. B.7). Nous les ordonnons d'une manière décroissante selon leurs valeurs propres correspondantes. Pour cela, le vecteur $\overrightarrow{v_2}$ est la première colonne de U' car $\lambda_2 > \lambda_1$.

$$U' = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{B.7}$$

Une transformation est nécessaire pour que cette matrice devienne une matrice orthogonale. Une approche possible est cellle de Gram-Schmidt [2] qui donne comme résultat la matrice U qu'on cherche (eq. B.8).

$$U = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \tag{B.8}$$

B.2 Réduire le nombre de dimensions

Réduire le nombre de dimensions d'un espace qui représente un ensemble de données est une technique bien connue. Dans des applications où on représente un ensemble d'objets par des vecteurs contenant beaucoup de dimensions, les techniques de réductions de dimensionalité cherchent les dimensions qui permettent de représenter les objets originaux avec moins de dimensions tout en conservant une approximation acceptable des données originales. Dans certains cas, cette représentation est même supérieure à la représentation qui utilise toutes les dimensions, si les dimensions enlevées sont des dimensions qui introduisent du bruit. Nous avons vu que SVD permet de décomposer une matrice A en trois matrices. La génération de ces trois matrices se base sur les vecteurs et les valeurs proposes des matrices AA^T et A^TA. La réduction de dimensionalité dans ce cas repose sur l'idée que ces trois matrices contiennent des valeurs qui sont dans un ordre décroissant par rapport aux valeurs propres des matrices qui les génèrent (par exemple AA^T génère

2. http ://jeff560.tripod.com/g.html

U). Nous supposons donc que les informations que ces matrices représentent sont plutôt concentrées dans leurs premières colonnes (ce qui correspond aux dernières lignes si on considère la transposé V^T de la matrice V). Ainsi, le fait de supprimer les dimensions les moins informatives donnera une approximation acceptable de la matrice originale A. C'est-à-dire qu'on peut considérer l'équation B.9.

$$A_{\{m,n\}} \approx A_{k\{m,n\}} = U_{\{m,k\}} S_{\{k,k\}} V^T_{\{k,n\}} \tag{B.9}$$

où k est un entier inférieur à m et n. Le choix de k est un défi pour les applications qui utilisent cette technique. Malgré les nombreuses études qui cherchent à trouver une heuristique pour le choix de k [Kontostathis, 2007; Bradford, 2008], ce choix reste souvent empirique et dépend de chaque expérience.

B.3 LSI

En recherche d'information, LSI est une application directe de SVD et de la réduction de dimensionalité sur la matrice de termes et des documents. Nous présentons dans cette section l'exemple démonstratif de Grossman [2004], qui est souvent utilisé pour illustrer l'utilisation de LSI en tant que modèle de mise en correspondance entre une requête et des documents.

Supposons que nous avons ces trois documents :

d_1 : `Shipment of gold damaged in fire`
d_2 : `Delivery of silver arrived in a silver truck`
d_3 : `Shipment of gold arrived in a truck`

et la requête Q suivante :

`q : gold silver truck`

La matrice A qui contient les fréquences des termes dans les documents est représentée dans la figure B.1. La décomposition de la matrice A selon la section B.1 donne les trois matrices suivantes (Fig. B.2).

Avec cet exemple simple, on peut considérer un nombre de dimensions réduit à 2 ($k = 2$). Dans ce cas, on ignore la dernière colonne de la matrice U, la dernière ligne de la matrice V^T et les dernière lignes et colonnes de la matrice S. Avec quelques règles d'algèbre nous pouvons déduire que les coordonnées de la requête q dans ce nouvel espace sont calculées par l'équation B.10.

$$\overrightarrow{q_k} = q^T U_k S_k^{-1} \tag{B.10}$$

Ainsi, pour la requête q, le vecteur $\overrightarrow{q} = [0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 1]$ de fréquence de termes dans cette requête devient le vecteur réduit $q_k = [-0,2140, -0,1821]$ dans

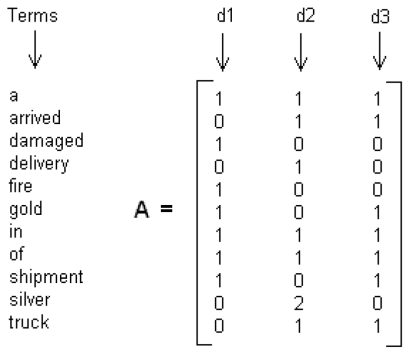


FIGURE B.1 – L'exemple de la matrice qui contient les fréquences de termes dans les trois documents d_1, d_2 et d_3.

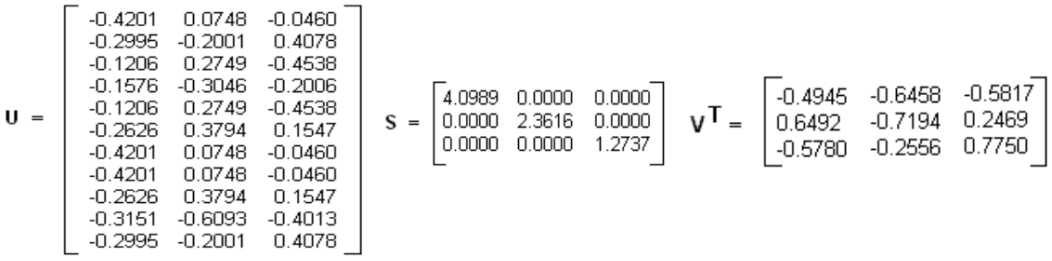


FIGURE B.2 – Les trois matrices de la décomposition de la matrice A.

le nouvel espace. La mise en correspondance dans ce nouvel espace correspond à calculer la cosinus-similarité (par exemple) entre le vecteur de la requête et les vecteurs des documents de la collection (Fig. B.3).

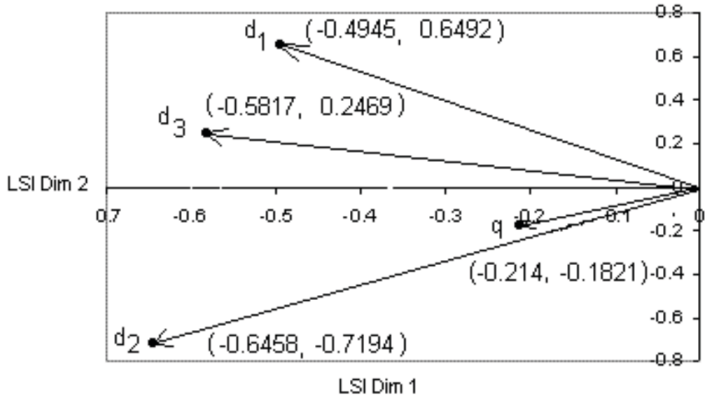


FIGURE B.3 – La mise en correspondance entre une requête et des documents dans l'espace réduit de LSI.

Dans cet exemple, d_2 est le document le plus proche de la requête q dans l'espace produit par LSI, et d_1 est le document le moins pertinent. Ainsi, la liste de résultats

de la requête q sera donnée dans l'ordre suivant : d_2, d_3, d_1.

Une caractéristique intéressante de la technique LSI est qu'elle capture la co-occurrence à plusieurs niveaux, c'est-à-dire que le fait que le document d_3 qui ne contient pas le mot « *silver* », mais contient les mots « *arrived* » et « *truck* » qui eux co-occurrent avec le mot « *silver* » ailleurs (dans le document d_2) récompense ce document par rapport à la requête qui contient le mot « *silver* ». Ce comportement peut être également en lien avec la figure B.4 qui présente la matrice A_k (eq. B.9) qui est l'approximation de la matrice A qu'on voit dans la même figure. Ce qu'on peut

			d1	d2	d3
a	=	k1	0.9662	0.9850	1.0453
arrived	=	k2	0.3003	1.1328	0.5974
damaged	=	k3	0.6659	-0.1478	0.4478
delivery	=	k4	-0.1476	0.9347	0.1982
fire	=	k5	0.6659	-0.1478	0.4478
gold	=	k6	1.1140	0.0506	0.8473
in	=	k7	0.9662	0.9850	1.0453
of	=	k8	0.9662	0.9850	1.0453
shipment	=	k9	1.1140	0.0506	0.8473
silver	=	k10	-0.2955	1.8692	0.3960
truck	=	k11	0.3003	1.1328	0.5974

$= A_k \approx A =$

d1	d2	d3
1	1	1
0	1	1
1	0	0
0	1	0
1	0	0
1	0	1
1	1	1
1	1	1
1	0	1
0	2	0
0	1	1

FIGURE B.4 – La comparaison entre les valeurs de la matrice A et son approximation A_k.

déduire de la figure B.4, est que le mot « *silver* », qui a une fréquence nulle dans les documents d_1 et d_3 de la matrice A, obtient une valeur plus élevée dans le document d_3 par rapport au document d_1 dans la matrice A_k. Par contre, la relation exacte entre la fréquence des termes dans les documents, et les valeurs correspondant à ces termes dans la matrice approximative n'est pas facile à expliquer [Kenmogne, 2005].

Annexe C

Mots vides de nos expériences

Alors qu'il semble un détail, la liste de mots vides joue un rôle essentiel sur la performance d'une expérience de recherche d'information. Voici la liste de mots vides utilisés dans nos expériences. Cette liste correspond à la liste standard d'Inquery à l'exception du mot « *usually* » que nous avons ajouté.

a	about	above	according	across	after
afterwards	again	against	albeit	all	almost
alone	along	already	also	although	always
am	among	amongst	an	and	another
any	anybody	anyhow	anyone	anything	anyway
anywhere	apart	are	around	as	at
av	be	became	because	become	becomes
becoming	been	before	beforehand	behind	being
below	beside	besides	between	beyond	both
but	by	can	cannot	canst	certain
cf	choose	contrariwise	cos	could	cu
day	did	do	does	doesn't	doing
dost	doth	double	down	dual	during
each	either	else	elsewhere	enough	et
etc	even	ever	every	everybody	everyone
everything	everywhere	except	excepted	excepting	exception
exclude	excluding	exclusive	far	farther	farthest
few	ff	first	for	formerly	forth
forward	from	front	further	furthermore	furthest
get	go	had	halves	hardly	has
hast	hath	have	he	hence	henceforth
her	here	hereabouts	hereafter	hereby	herein
hereto	hereupon	hers	herself	him	himself
hindmost	his	hither	hitherto	how	however
howsoever	i	ie	if	in	inasmuch
inc	include	included	including	indeed	indoors
inside	insomuch	instead	into	inward	inwards
is	it	its	itself	just	kind
kg	km	last	latter	latterly	less
lest	let	like	little	ltd	made
many	may	maybe	me	meantime	meanwhile
might	moreover	most	mostly	more	mr
mrs	ms	much	must	my	myself
namely	need	neither	never	nevertheless	next
no	nobody	none	nonetheless	noone	nope
nor	not	no	notwithstanding	now	nowadays
nowhere	of	off	often	ok	on
once	one	only	onto	or	other

others	otherwise	ought	our	ours	ourselves
out	outside	over	own	per	perhaps
plenty	provide	quite	rather	really	round
said	sake	same	sang	save	saw
see	seeing	seem	seemed	seeming	seems
seen	seldom	selves	sent	several	shalt
she	should	shown	sideways	since	slept
slew	slung	slunk	smote	so	some
somebody	somehow	someone	something	sometime	sometimes
somewhat	somewhere	spake	spat	spoke	spoken
sprang	sprung	stave	staves	still	such
supposing	than	that	the	thee	their
them	themselves	then	thence	thenceforth	there
thereabout	thereabouts	thereafter	thereby	therefore	therein
thereof	thereon	thereto	thereupon	these	they
this	those	thou	though	thrice	through
throughout	thru	thus	thy	thyself	till
to	together	too	toward	towards	ugh
unable	under	underneath	unless	unlike	until
up	upon	upward	upwards	us	use
used	using	very	via	vs	want
was	we	week	well	were	what
whatever	whatsoever	when	whence	whenever	whensoever
where	whereabouts	whereafter	whereas	whereat	whereby
wherefore	wherefrom	wherein	whereinto	whereof	whereon
wheresoever	whereto	whereunto	whereupon	wherever	wherewith
whether	whew	which	whichever	whichsoever	while
whilst	whither	who	whoa	whoever	whole
whom	whomever	whomsoever	whose	whosoever	why
will	wilt	with	within	without	worse
worst	would	wow	ye	yet	year
yippee	you	your	yours	yourself	yourselves
text	thing	things	usually		

Table des figures

Liste des tableaux

Bibliographie

R. L. Ackoff. From data to wisdom. *Journal of Applied System Analysis*, 16 :3–9, 1989. (Cité en page 13.)

ANSI/NISO. Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies. Technical report, NISO Standard (ANSI), An American National Standard developed by the National Information Standards Organization, 2005. (Cité en pages 8, 9 et 137.)

Timothy G Armstrong, Alistair Moffat, William Webber, and Justin Zobel. Has Adhoc Retrieval Improved Since 1994 ? Categories and Subject Descriptors. In *SIGIR*, 2008. (Cité en page 18.)

Bissan Audeh. Experiments on two Query Expansion Approaches for a Proximity-based Information Retrieval Model. In *Rencontre des Jeunes Chercheurs en Recherche d'Information (RJCRI)*, Bordeaux, France, 2012a. ARIA. (Non cité.)

Bissan Audeh. Semantic query expansion for fuzzy proximity information retrieval model. In *Early Career Symposium at 7th International Conference on Formal Ontology in Informaion Systems (FOIS)*, Graz, Autriche, 2012b. (Non cité.)

Bissan Audeh, Philippe Beaune, and Michel Beigbeder. Expansion sémantique des requêtes pour un modèle de recherche d' information par proximité. In *INFORSID*, Paris, France, 2013a. (Non cité.)

Bissan Audeh, Philippe Beaune, and Michel Beigbeder. Recall-Oriented Evaluation for Information Retrieval Systems. In *Information Retrieval Facility Conference (IRFC)*, Limassol, Chypre, 2013b. (Cité en page 123.)

Bissan Audeh, Philippe Beaune, and Michel Beigbeder. Exploring Query Reformulation for Named Entity Expansion in Information Retrieval . In *29th Symposium On Applied Computing (SAC)*, Gyeongju, Korea, 2014a. ACM. (Cité en page 122.)

Bissan Audeh, Philippe Beaune, and Michel Beigbeder. L'utilisation des entités nommées pour l'expansion sémantique des requêtes Web. In *14èmes Journées Francophones "Extraction et Gestion des Connaissances" (EGC)*, Rennes, France, 2014b. RNTI. (Cité en page 121.)

Bissan Audeh, Philippe Beaune, and Michel Beigbeder. La reformulation hybride des requêtes exploratoires à l'aide de concepts explicites et implicites. In *COnférence en Recherche d'Information et Applications (CORIA)*, Nancy, France, 2014c. RNTI. (Cité en page 122.)

Holger Bast, Alexandru Chitea, Fabian Suchanek, and Ingmar Weber. Ester : efficient search on text, entities, and relations. In *Proceedings of the 30th annual*

international ACM SIGIR conference on Research and development in information retrieval, pages 671–678. ACM, 2007. (Cité en page 35.)

Michael Bendersky and W Bruce Croft. Discovering key concepts in verbose queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 491–498. ACM, 2008. (Cité en pages 23, 41, 42, 43 et 99.)

Michael Bendersky and W Bruce Croft. Learning Concept Importance Using a Weighted Dependence Model. In *WSDM*, 2010. (Cité en page 41.)

Michael Bendersky, Marina Rey, and W Bruce Croft. Parameterized Concept Weighting in Verbose Queries. In *SIGIR*, 2011. (Cité en pages 31, 41, 42, 43 et 51.)

Michael Bendersky, Donald Metzler, and W Bruce Croft. Effective query formulation with multiple information sources. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 443–452. ACM, 2012. (Cité en page 41.)

J. Bhogal, a. Macfarlane, and P. Smith. A review of ontology based query expansion. *Information Processing and Management*, 43 :866–886, July 2007. (Cité en page 32.)

DM Blei, AY Ng, and MI Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3 :993–1022, 2003. (Cité en pages 21 et 43.)

Mohand Boughanem and Jacques Savoy. *Recherche d'information état des lieux et perspectives*. Lavoisier, 2008. (Cité en pages 15 et 16.)

Roger B. Bradford. An empirical study of required dimensionality for large-scale latent semantic indexing applications. In *CIKM*, California, USA, 2008. (Cité en page 131.)

Wladmir Brandao, Altigran Silva, Edleno Moura, and Nivio Ziviani. Exploiting entity semantics for query expansion. In *IADIS International conference WWW/Internet*, Rio de Janeiro, 2011. (Cité en page 45.)

Marie-France Bruandet and Jean-Pierre Chevallet. Utilisation et construction de bases de connaissances pour la recherche d'information. In *Assistance intelligente à la recherche d'information*. Lavoisier, 2003. (Cité en pages 9 et 10.)

Pino Buizza. Indexing concepts and/or named entities. *JLIS. it*, 2, 2011. (Cité en page 43.)

Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, page 243, 2008. (Cité en page 123.)

Claudio Carpineto and Giovanni Romano. A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)*, 44 :1, 2012. (Cité en pages 32 et 36.)

C. W. Cleverdon, J. Mills, and E. M. Keen. *Factors Determining the Performance of Indexing Systems.* College of Aeronautics, Cranfield, aslib cran edition, 1966. (Cité en pages 77 et 79.)

C.W. Cleverdon. Report on the testing and analysis of an investigation into the com- parative efficiency of indexing systems. Technical report, Aslib Cranfield Research Project, Cranfield, 1962. (Cité en page 79.)

Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Predicting query performance. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, page 299, 2002. (Cité en pages 33, 38 et 60.)

Hang Cui, Ji-rong Wen, Jian-yun Nie, and Wei-ying Ma. Query expansion by mining user logs. *IEEE Transactions on Knowledge and Data Engineering*, 15 :829–839, July 2003. (Cité en pages 35 et 37.)

Van Dang and Bruce W. Croft. Query reformulation using anchor text. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, pages 41–50, NY, USA, 2010. ACM. (Cité en pages 35 et 37.)

Scott Deerwester, Susan T Dumais, George W Furnas, and Thomas K Landauer. Indexing by Latent Semantic Analysis. *Society*, 41 :391–407, 1990. (Cité en page 20.)

Romain Deveaud. *Vers une représentation du contexte thématique en Recherche d'Information.* PhD thesis, Université d'Avignon, 2013. (Cité en page 104.)

Romain Deveaud, Ludovic Bonnefoy, and Patrice Bellot. Quantification et identification des concepts implicites d'une requête. In *CORIA 2013, La dixième édition de la COnférence en Recherche d'Information et Applications*, Neuchâtel, 2013. (Cité en pages 31, 32, 39, 41, 43, 49 et 51.)

Ofer Egozi, Shaul Markovitch, and Evgeniy Gabrilovich. Concept-Based Information Retrieval Using Explicit Semantic Analysis. *ACM Transactions on Information Systems*, 29 :1–34, 2011. (Cité en pages 18, 19, 21, 22, 51 et 137.)

Hui Fang. An Exploration of Axiomatic Approaches to Information Retrieval. In *Proceedings of the international ACM SIGIR conference on Research and development in information retrieval*, 2005. (Cité en page 34.)

Hui Fang. A Re-examination of Query Expansion Using Lexical Resources. *Computational Linguistics*, pages 139–147, 2008. (Cité en pages 34 et 37.)

Shaul Gabrilovich, Evgeniy and Markovitch. Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis. In *IJCAI*, 2007. (Cité en page 21.)

Susan Gauch. KeyConcept : A Conceptual Search Engine. Technical report, TR-8646-37, University of Kansas, 2003. (Cité en page 23.)

Éric Gaussier and François Yvon. *Modèles statistiques pour l'accès à l'information textuelle*. Lavoisier, 2011. (Cité en page 21.)

J Gonzalo, F Verdejo, I Chugur, and J Cigarran. Indexing with WordNet synsets can improve Text Retrieval. In *Proceedings of the COLING/ACL 98 Workshop on Usage of WordNet for NLP, Montreal*, 1998. (Cité en pages 23 et 33.)

F. A. Grootjen and Th. P. Van Der Weide. Conceptual query expansion. *Data and Knowledge Engineering*, 56 :174–193, 2004. (Cité en pages 27, 49 et 51.)

David A Grossman. *Information retrieval : Algorithms and heuristics*, volume 15. Springer, 2004. (Cité en page 131.)

Jiafeng Guo, Gu Xu, Xueqi Cheng, and Hang Li. Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR*, 2009. (Cité en pages 43 et 44.)

H.A. Hall and N.H. Weiderman. The Evaluation Problem in Relevance Feedback Systems, Report ISR-12. Technical report, The national Science Foundation, Section XII, Department of Computer Science, Cornell University, NY, 1967. (Cité en pages 79 et 80.)

Ruhan He, Yong Zhu, and Wei Zhan. Using Local Latent Semantic Indexing with Pseudo Relevance Feedback in Web Image Retrieval. *2009 Fifth International Joint Conference on INC, IMS and IDC*, pages 1354–1357, 2009. (Cité en pages 32 et 37.)

Pascal Hitzler, Markus Krotzsch, and Sebastien Rudolph. *Foundations of semantic web technologies*. CRC Press, 2009. (Cité en pages 8 et 11.)

Hanh Huu Hoang and A Min Tjoa. The State of the Art of Ontology-based Query Systems : A Comparison of Existing Approaches. In *International Conference on Computing and Informatics ICOCI*, 2006. (Cité en page 34.)

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Furstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust Disambiguation of Named Entities in Text. In *EMNLP '11 Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 782–792, 2011. (Cité en pages 43 et 52.)

Thomas Hofmann. Probabilistic latent semantic indexing. *Proceedings of the 22nd annual international ACM SIGIR*, pages 50–57, 1999. (Cité en page 21.)

Samuel Huston and W Bruce Croft. Evaluating verbose query processing techniques. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 291–298. ACM, 2010. (Cité en page 45.)

Souhei Ito, Shigeki Hagihara, and Naoki Yonezaki. A formal ontology for business process model tap : Tasks-agents-products. In *Proceedings of the 2008 Conference on Information Modelling and Knowledge Bases XIX*, pages 290–297, Amsterdam,The Netherlands, 2008. IOS Press. (Cité en page 35.)

Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs : a study and analysis of user queries on the web. *Information Processing and Management*, 36 :207–227, March 2000. (Cité en page 26.)

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4) :422–446, 2002. (Cité en page 82.)

Jay J Jiang and David Conrath. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *International Conference on Research in Computational Linguistics*, 1997. (Cité en page 10.)

Yufeng Jing and W Bruce Croft. An Association Thesaurus for Information Retrieval. *RIAO 94 Conference Proceedings*, pages 1–15, 1994. (Cité en pages 29, 32 et 37.)

Sparck Jones. *The Cranfield tests*. Butterworths, London, 1981. (Cité en page 77.)

M.G. Kendall. A new measure of rank correlation. *biometrika*, 30 :81–93, 1938. (Cité en page 93.)

R Newo Kenmogne. *Understanding LSI via the Truncated Term-term Matrix*. PhD thesis, PhD thesis, Universität des Saarlandes, 2005. (Cité en page 133.)

Atanas Kiryakov, Borislav Popov, Ivan Terziev, Dimitar Manov, and Damyan Ognyanoff. Semantic annotation, indexing, and retrieval. *Web Semantics : Science, Services and Agents on the World Wide Web*, 2 :49–79, 2004. (Cité en page 43.)

Daniel Kless, Ludger Jansen, Jutta Lindenthal, and Jens Wiebensohn. A method for re-engineering a thesaurus into an ontology. In *FOIS*, pages 133–146, 2012. (Cité en page 10.)

Matthew B Koll. Weird : An approach to concept-based information retrieval. In *ACM SIGIR Forum*, volume 13, pages 32–50. ACM, 1979. (Cité en page 20.)

April Kontostathis. Essential dimensions of latent semantic indexing (lsi). In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, pages 73–73. IEEE, 2007. (Cité en pages 21 et 131.)

Giridhar Kumaran and Vitor R. Carvalho. Reducing long queries using query quality predictors. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, page 564, NY, USA, 2009. ACM Press. (Cité en page 45.)

Victor Lavrenko and W. Bruce Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, NY, USA, 2001. ACM Press. (Cité en pages 30, 31, 37 et 98.)

M.E Lesk. Word-word associations in document retrieval systems. *American Documentation*, 1969. (Cité en pages 28 et 37.)

Shuang Liu, Fang Liu, Clement Yu, and Weiyi Meng. 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*, pages 266–272, NY, USA, 2004. ACM. (Cité en pages 33 et 37.)

Jing Luo, Bo Meng, Maofu Liu, Xinhui Tu, and Kui Zhang. Query Expansion using Explicit Semantic Analysis. In *ICIMCS*, pages 123–126, 2012. (Cité en pages 32, 37, 49 et 51.)

Yuanhua Lv and Chengxiang Zhai. A Comparative Study of Methods for Estimating Query Language Models with Pseudo Feedback. *Strategies*, 2 :3–6, 2009. (Cité en pages 30, 31 et 98.)

Yuanhua Lv and ChengXiang Zhai. Positional relevance model for pseudo-relevance feedback. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 579–586. ACM, 2010. (Cité en pages 30, 31 et 37.)

Walid Magdy. *Toward Higher Effectiveness for Recall- Oriented Information Retrieval : A Patent Retrieval Case Study Walid Magdy*. PhD thesis, Dublin City University, 2012. (Cité en page 83.)

Walid Magdy and Gareth J F Jones. PRES : A Score Metric for Evaluating Recall-Oriented Information Retrieval Applications. In *SIGIR*, 2010. (Cité en pages 81, 83 et 86.)

Rila Mandala, Tokunaga Takenobu, and Tanaka Hozumi. The use of WordNet in Information Retrieval. In *ACL Workshop on the Usage of WordNet in Information Retrieval.*, pages 31–37. Association for Computational Linguistics, 1998. (Cité en pages 34 et 45.)

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. (Cité en page 26.)

K Tamsin Maxwell and W Bruce Croft. Compact query term selection using topically related text. In *Proceedings of the 36th international ACM SIGIR*, pages 583–592, 2013. (Cité en pages 41 et 45.)

Donald Metzler and W. Bruce Croft. Combining the language model and inference network approaches to retrieval. *Inf. Process. Manage.*, 40 :735–750, September 2004. (Cité en pages 62, 65, 126, 127, 128 et 138.)

Donald Metzler and W. Bruce Croft. 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*, page 472, NY, USA, 2005. ACM Press. (Cité en pages 41, 42, 43, 99 et 137.)

Donald Metzler and W. Bruce Croft. Latent concept expansion using markov random fields. *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, page 311, 2007. (Cité en pages 44, 49 et 51.)

Christian Middleton and Ricardo Baeza-yates. A Comparison of Open Source Search Engines. Technical report, Universitat Pompeu Fabra, Barcelona, Spain, 2007. (Cité en page 18.)

George a. Miller. WordNet : a lexical database for English. *Communications of the ACM*, 38 :39–41, November 1995. (Cité en page 33.)

George A Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, and Katherine J Miller. Introduction to wordnet : An on-line lexical database. *International journal of lexicography*, 3(4) :235–244, 1990. (Cité en pages 10 et 23.)

J Minker, G.A Wilson, and B.H. Zimmerman. An evaluation of query expansion by the addition of clustered terms for a document retrieval system. Information Storage and Retrieval. *Information storage and retrieval*, 8 :329–348, 1972. (Cité en pages 28 et 37.)

Marvin Lee Minsky, Marvin Minsky, and Marvin Minsky. *Semantic information processing*, volume 142. MIT press Cambridge, MA, 1968. (Cité en page 8.)

Mandar Mitra, Amit Singhal, and Chris Buckley. Improving automatic query expansion. *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214, 1998. (Cité en page 38.)

Jesse Montgomery, Luo Si, Jamie Callan, and David A Evans. Effect of varying number of documents in blind feedback : analysis of the 2003 nrrc ria workshop bf_numdocs experiment suite. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 476–477. ACM, 2004. (Cité en page 39.)

David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, pages 1–20, 2007. (Cité en page 43.)

Roberto Navigli. Word sense disambiguation : A survey. *ACM Computing Surveys*, 41 :1–69, February 2009. (Cité en pages 14 et 54.)

Roberto Navigli and Paola Velardi. An analysis of ontology-based query expansion strategies. In *14th European Conference on Machine Learning, Workshop on Adaptive Text Extraction and Mining, Cavtat-Dubrovnik, Croatia (pp. 42-49)*, 2003. (Cité en pages 33, 37 et 44.)

Vuong M Ngo and Tru H Cao. Discovering latent concepts and exploiting ontological features for semantic text search. *IJCNLP*, pages 571–579, 2011. (Cité en page 35.)

Martha Palmer and Zhibiao Wu. Verb semantics and lexical selection. In *Association for Computational Linguistics*, pages 133–138, 1994. (Cité en page 54.)

Helen J. Peat and Peter Willett. The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42 :378–383, June 1991. (Cité en page 28.)

Ted Pedersen, Serguei V S Pakhomov, Siddharth Patwardhan, and Christopher G Chute. Measures of semantic similarity and relatedness in the biomedical domain. *Journal of biomedical informatics*, 40 :288–99, June 2007. (Cité en pages 14 et 137.)

Desislava Petkova and W. Bruce Croft. Proximity-based document representation for named entity retrieval. *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management - CIKM '07*, page 731, 2007. (Cité en page 43.)

Jay M Ponte and W Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998. (Cité en pages 17, 125 et 126.)

Jay Michael Ponte. *A language modeling approach to information retrieval*. PhD thesis, University of Massachusetts, 1998. (Cité en pages 30, 31, 37 et 125.)

Jeffrey Pound, Ihab F Ilyas, and Grant Weddell. Expressive and flexible access to web-extracted data : a keyword-based structured query language. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 423–434. ACM, 2010. (Cité en page 35.)

Yonggang Qiu and H.P. Frei. Concept Based Query Expansion. In *Proceedings of the international ACM SIGIR conference on Research and development in informaion retrieval*, volume 11, page 212, NY, January 1993. ACM. (Cité en pages 29, 37, 49 et 51.)

Bertram Raphael. *SIR : A computer program for semantic information retrieval*. PhD thesis, Massachusetts Institute of Technology, 1964. (Cité en page 22.)

Van Rijsbergen. *Information Retrieval*. Butterworth-Heinemann 1979, Butterworths, 1979. (Cité en pages 83 et 86.)

S.E Robertson and S Walker. Some for Simple Effective Approximations to the 2Poisson Model Probabilistic Weighted Retrieval The. *Search*, 1994. (Cité en page 17.)

Stephen Robertson. On gmap : and other transformations. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 78–83. ACM, 2006. (Cité en page 82.)

J.J Rocchio. Performance indices for document retrieval systems. *Information storage and retrieval*, 1964. (Cité en pages 83 et 86.)

J.J Rocchio, Jr Salton, and G. Information Search Optimization and Iterative Retrieval Techniques. In *Fall Joint Computer Conference*, pages 293–305, 1965. (Cité en pages 30 et 37.)

Ronald Rosenfeld. Two decades of statistical language modeling : Where do we go from here. In *Proceedings of the IEEE*, 2000. (Cité en page 125.)

Tetsuya Sakai. On the reliability of information retrieval metrics based on graded relevance. *Inf. Process. Manage.*, 43(2) :531–548, 2007. (Cité en page 82.)

Mark Sanderson. Retrieving with good sense. *Information retrieval*, 2000. (Cité en page 23.)

Hinrich Schütze. Automatic word sense discrimination. *Computational linguistics*, 24(1) :97–123, 1998. (Cité en page 32.)

Chirag Shah and W Bruce Croft. Evaluating High Accuracy Retrieval Techniques Chirag Shah. In *SIGIR*, 2004. (Cité en pages 33, 37 et 38.)

D. Stenmark. Query Expansion on a Corporate Intranet : Using LSI to Increase Precision in Explorative Search. *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*, 00 :101c–101c, 2005. (Cité en page 38.)

Christopher Stokoe, Michael P. Oakes, and John Tait. Word sense disambiguation in information retrieval revisited. *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, page 159, 2003. (Cité en page 23.)

Trevor Strohman, Donald Metzler, Howard Turtle, and WB Croft. Indri : A language-model based search engine for complex queries. *Proceedings of the International Conference on Intelligence Analysis,*, 2004. (Cité en pages 17, 63 et 126.)

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago : a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706. ACM, 2007. (Cité en pages 12, 22 et 35.)

Don R Swanson. Historical Note : Information Retrieval and the Future of an Illusion. *System*, 39 :92–98, 1988. (Cité en page 18.)

Howard Turtle and W Bruce Croft. Inference Networks for Document Retrieval. *Science*, 1990. (Cité en pages 126, 127 et 138.)

Howard Turtle, Yatish Hegde, and S Rowe. Yet another comparison of lucene and indri performance. In *SIGIR 2012 Workshop on Open Source Information Retrieval*, pages 64–67, 2012. (Cité en page 18.)

Reinout Van Rees. Clarity in the usage of the terms ontology, taxonomy and classification. *CIB REPORT*, 284 :432, 2003. (Cité en page 11.)

E. M. Voorhees and D. M. Tice. The TREC-8 Question Answering Track Evaluation. In *TREC 1999*, 1999. (Cité en page 82.)

Ellen M Voorhees. 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*, pages 171–180. ACM, 1993. (Cité en page 23.)

Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR '94*, 1994. (Cité en pages 33 et 37.)

Kishor Wagh and Satish Kolhe. Information Retrieval Based on Semantic Similarity Using Information Content. *Journal of Computer Science*, 8 :364–370, 2011. (Cité en page 10.)

Xuanhui Wang and ChengXiang Zhai. Mining term association patterns from search logs for effective query reformulation. *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, page 479, 2008. (Cité en pages 35 et 37.)

William Edward Webber. *Measurement in Information Retrieval Evaluation*. PhD thesis, University ofMelbourne, 2010. (Cité en page 83.)

Xing Wei and W. Bruce Croft. LDA-based document models for ad-hoc retrieval. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, page 178, 2006. (Cité en page 21.)

Ryen W White, Ian Ruthven, and Joemon M Jose. *The use of implicit evidence for relevance feedback in web retrieval*. Springer, 2002. (Cité en page 35.)

William A Woods. Conceptual indexing : A better way to organize knowledge. Technical report, Sun Microsystems, Inc., Mountain View, CA, USA, 1997. (Cité en page 23.)

Jiewen Wu, Ihab Ilyas, and Grant Weddell. A study of ontology-based query expansion. Technical report, Technical report CS-2011-04, University of Waterloo, 2011. (Cité en page 32.)

Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11, 1996. (Cité en pages 27 et 32.)

Jinxi Xu and W. Bruce Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Transactions on Information Systems*, 18 : 79–112, January 2000. (Cité en page 44.)

Xuheng Xu, Xiaodan Zhang, and Xiaohua Hu. Using two-stage concept-based singular value decomposition technique as a query expansion strategy. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*, volume 1, pages 295–300. IEEE, 2007. (Cité en pages 32 et 37.)

Yang Xu, Fan Ding, and Bin Wang. Entity-based query reformulation using wikipedia. In *Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08*, page 1441, NY, USA, 2008. ACM Press. (Cité en page 45.)

Xiaobing Xue, Samuel Huston, and W. Bruce Croft. Improving verbose queries using subset distribution. *Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10*, page 1059, 2010. (Cité en page 41.)

Chengxiang Zhai and John Lafferty. Model-based feedback in the language modeling approach to information retrieval. *Proceedings of the tenth international conference on Information and knowledge management - CIKM'01*, page 403, 2001. (Cité en pages 30, 31 et 37.)

Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22 :179–214, 2004. (Cité en page 126.)

Jiuling Zhang, Beixing Deng, and Xing Li. Concept Based Query Expansion Using WordNet. *2009 International e-Conference on Advanced Science and Technology*, pages 52–55, March 2009. (Cité en pages 34, 37, 49 et 51.)

Le Zhao and Jamie Callan. Term necessity prediction. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 259–268. ACM, 2010. (Cité en pages 31, 32, 38, 60, 61, 115 et 123.)

Dong Zhou, Séamus Lawless, and Vincent Wade. Improving search via personalized query expansion using social media. *Information retrieval*, 15 :218–242, 2012. (Cité en pages 35 et 37.)

Justin Zobel. Questioning Query Expansion : An Examination of Behaviour and Parameters. In *SIGIR*, 2004. (Cité en pages 26, 38 et 39.)

Justin Zobel and Laurence A F Park. Against Recall : Is it Persistence, Cardinality, Density, Coverage, or Totality ? In *SIGIR*. ACM, 2009. (Cité en pages 83 et 85.)

École Nationale Supérieure des Mines de Saint-Étienne

NNT : 2014 EMSE 0750
Bissan AUDEH
Dissertation title : Semantic query reformulation for ad hoc information retrieval on the Web
Speciality : Computer science
Keywords : Information retrieval, semantic query reformulation, relevance feedback, semantic resources, recall evaluation.

Abstract : Matching users information needs and relevant documents has always been a basic problem in information retrieval. A query and a relevant document don't necessarily use the same terms to express the same concepts, this fact is even more visible on the Web. As a query expansion and reformulation solution, we are interested in the different ways the semantic could be used to translate users information need into a query. We define two types of concepts : those which we can identify in a semantic resource like an ontology, and the ones we extract from the collection of documents via pseudo relevance feedback procedure. We propose a semantic and mixed approach to query expansion and reformulation (ASMER) that allows to integrate these two types of concepts in an automatically modified query. Our approach considers many challenges, especially selective terms expansion, named entity treatment and query reformulation.

Even though the precision is the evaluation criteria the most adapted to a web context, we also considered evaluating the recall to study the behavior of our model from different aspects. This choice led us to handle a different problem related to evaluating the recall in information retrieval. After realizing that actual measures don't satisfy our constraints, we proposed a new recall oriented measure (MOR) which considers the recall as a priority without ignoring the precision. Among other measures, MOR was considered to evaluate our approach ASMER on four web collection from the standard evaluation campaigns Inex and Trec. Our experiments showed that ASMER improves the precision of the non modified original queries. In most cases, our approach achieved statistically significant enhancements when compared to a state of the art query expansion method. In addition, ASMER retrieves the first relevant document in better ranks than the compared approaches, it also has slightly better recall according to the measure MOR.

École Nationale Supérieure des Mines de Saint-Étienne

NNT : 2014 EMSE 0750
Bissan AUDEH
Titre de la thèse : Reformulation sémantique des requêtes pour la recherche d'information ad hoc sur le Web
Spécialité : Informatique
Mots clés : Recherche d'information, reformulation sémantique de la requête, retour de pertinence, ressources sémantiques, évaluation du rappel.

Résumé : Dans un système de recherche documentaire, mettre en correspondance le besoin d'information de l'utilisateur et les documents pertinents est la problématique de base. Un document sémantiquement pertinent pour une requête n'utilise pas forcément les mêmes termes que cette requête pour exprimer les mêmes concepts, cela est d'autant plus vrai sur le Web. Dans le cadre d'une solution d'expansion et de reformulation de la requête, nous nous intéressons aux différentes façons d'utiliser la sémantique pour mieux exprimer le besoin d'information de l'utilisateur dans un contexte Web. Nous distinguons deux types de concepts : ceux identifiables dans une ressource sémantique comme une ontologie, et ceux que l'on extrait à partir d'un ensemble de documents de pseudo retour de pertinence. Nous proposons une Approche Sémantique Mixte d'Expansion et de Reformulation (ASMER) qui permet de modéliser l'utilisation de ces deux types de concepts dans une requête automatiquement modifiée. Cette approche considère plusieurs défis liés à la modification automatique des requêtes, notamment le choix sélectif des termes d'expansion, le traitement des entités nommées et la reformulation de la requête finale.

Bien que dans un contexte Web la précision soit le critère d'évaluation le plus adapté, nous avons aussi pris en compte le rappel pour étudier le comportement de notre approche sous plusieurs aspects. Ce choix a suscité une autre problématique liée à l'évaluation du rappel en recherche d'information. En constatant que les mesures précédentes ne répondent pas à nos contraintes, nous avons proposé la mesure MOR (Mesure Orientée Rappel), qui permet d'évaluer le rappel en tenant compte de la précision comme importante mais pas prioritaire dans un contexte dirigé rappel. En incluant MOR dans notre stratégie de test, nous avons évalué ASMER sur quatre collections Web issues des campagnes d'évaluation INEX et TREC. Nos expériences montrent qu'ASMER améliore la performance en précision par rapport aux requêtes originales. Dans la plupart des cas, notre approche apporte une amélioration statistiquement significative par rapport à l'utilisation des requêtes étendues par une méthode de l'état de l'art de l'expansion de la requête.