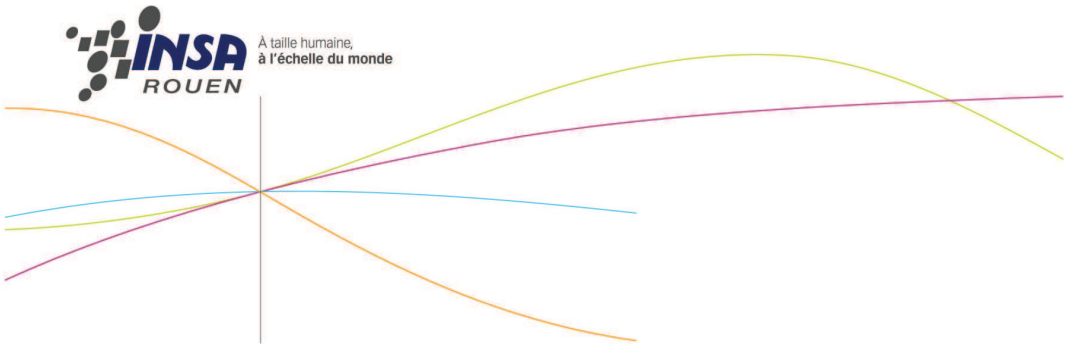
Contribution à la veille stratégique : DOWSER, un système de découverte de sources Web d'intérêt opérationnel

Romain Noël



Institut National des Sciences Appliquées de Rouen
Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes

Thèse de doctorat

Discipline : Informatique

présentée par

Romain NOËL

pour obtenir le grade de

Docteur de l'INSA de Rouen

Contribution à la veille stratégique : DOWSER, un système de découverte de sources Web d'intérêt opérationnel

---

soutenue le *17 octobre 2014* devant le jury composé de :

Sylvie Calabretto	Rapporteur	Professeur des Universités *LIRIS / INSA de Lyon*
Mohand Boughanem	Rapporteur	Professeur des Universités *IRIT / Université Paul Sabatier de Toulouse*
Pierre Maret	Examinateur	Professeur des Universités *LHC / Université Jean Monnet de Saint-Etienne*
Abdel-Illah Mouaddib	Examinateur	Professeur des Universités *GREYC / Université de Caen*
Laurent Vercouter	Directeur	Professeur des Universités *LITIS / INSA de Rouen*
Alexandre Pauchet	Encadrant	Maître de conférences *LITIS / INSA de Rouen*
Nicolas Malandain	Encadrant	Maître de conférences *LITIS / INSA de Rouen*
Bruno Grilheres	Encadrant	Docteur *Airbus, Defence and Space*
Stéphan Brunessaux	Invité	Senior expert *Airbus, Defence and Space*

AIRBUS DEFENCE & SPACE

litis

REMERCIEMENTS

Cette thèse s'est déroulée dans le cadre d'une convention CIFRE entre l'entreprise AIRBUS Defence and Space et le laboratoire du LITIS de L'INSA de Rouen. Je tiens à remercier le jury de ma thèse pour leur implication, ainsi que les personnes qui m'ont permis de mener à bien ce projet professionnel, bien qu'il me soit impossible de toutes les citer.

Mes remerciements vont tout d'abord à Laurent, mon directeur de thèse, Alexandre et Nicolas, mes encadrants, qui m'ont fait découvrir le monde de la recherche en me guidant et en me soutenant durant ces trois années. Leur implication et leur investissement dans mon travail, leurs précieux conseils, ainsi que leur disponibilité m'ont permis d'appréhender cette thèse jusqu'à son achèvement. J'ai eu la chance et la satisfaction d'avoir été encadré par un trio dynamique qui a été la pierre angulaire de ma motivation tout au long de ce parcours. Mes remerciements pour les personnes du laboratoire du LITIS vont également à Brigitte et Sandra qui ont toujours répondu rapidement et consciencieusement à mes demandes et démarches administratives.

Ensuite, je tiens à remercier Patrick qui, via son professionnalisme et la qualité de ses enseignements durant mon Master, m'a encouragé à faire mon stage de fin d'étude au sein du département IPCC d'Airbus DS. Je remercie Stéphan et Jean qui m'ont permis d'entreprendre cette thèse CIFRE à la suite du stage. Stéphan a, de plus, su me transmettre les valeurs et la culture d'entreprise qui m'ont permis de gérer cette thèse comme un projet et de respecter ainsi les contraintes temporelles. Mes remerciements vont à Bruno qui, en m'apportant tantôt son savoir scientifique, tantôt sa connaissance technique, a su me guider de la définition à l'accomplissement des objectifs de cette thèse. Il a su me transmettre sa passion du travail et me souffler un vent d'idées inspirantes qui ont nourri mon imagination. Je remercie ensuite tous mes collègues pour leur aide précieuse, leurs réponses à mes questions, et leur bonne humeur du quotidien. Chacun se reconnaîtra en se rappelant les bonnes parties de baby-foot, les rigolades à la machine à café ou dans les bureaux. Ce sont des moments précieux qui apportent un climat positif et bénéfique à la réussite de nos travaux. Cette bonne ambiance n'en serait rien sans tata Véro qui, en plus de son enthousiasme et de sa générosité, est toujours prête à rendre service avec une très grande efficacité.

Je remercie mes amis avec lesquels j'ai passé de bonnes soirées me permettant de décompresser et avec lesquels j'ai pu me changer les idées en vacances au soleil ou au ski. Autant de péripéties qui viennent s'ajouter à cette aventure qu'est la thèse.

Je terminerai naturellement par remercier ma famille.

Je remercie mes frangines, toujours là pour moi malgré la distance. En tant que petit dernier, vous m'avez toutes les deux montré le chemin et je suis fier de vous avoir pris comme modèles. Merci pour tout ce que vous m'avez apporté, chacune à votre façon, avant et pendant cette thèse. Enfin, même si ces derniers mots ne peuvent atteindre la hauteur de ma reconnaissance, je remercie mes parents pour m'avoir transmis leurs valeurs, pour la confiance qu'ils m'ont insufflé, et leur soutien constant même dans les moments les plus durs.

TABLE DES MATIÈRES

TABLE DES FIGURES

LISTE DES TABLEAUX

CHAPITRE 1

INTRODUCTION

La résolution d'un problème ou l'atteinte d'un objectif visé nécessite un ensemble de connaissances sur un sujet donné. Ce manque de connaissances génère un besoin en information qui se traduit par une activité documentaire afin de satisfaire ce besoin. La Recherche d'Information (RI) fait référence à la manière de répondre à un besoin en information en retournant un ensemble de documents pertinents parmi ceux présents au sein d'un corpus. Un corpus se compose d'un ensemble de documents décrits selon leur contenu et/ou selon des métadonnées associées. Dans le contexte du World Wide Web, le corpus est composé de documents mis en réseau et reliés entre eux par des liens hypertextes. Le contenu des documents peut être du texte, des vidéos, des sons, des images ou des données.

La RI sur le Web est devenue une tâche de la vie quotidienne pour des besoins professionnels ou personnels. Dans cette thèse, nous nous intéressons principalement aux besoins des experts du renseignement qui utilisent Internet comme support pour prendre leurs décisions. Dans le cadre du Renseignement d'Origine Sources Ouvertes, leurs recherches sont souvent liées à des besoins très spécifiques sur des sujets qui peuvent être sensibles, rendant la RI difficile. De plus, la réponse à leurs besoins opérationnels doit également prendre en compte la non-popularité des informations qu'ils recherchent dans un environnement comme le Web qui évolue constamment. Ainsi, nous nous intéressons plus particulièrement au processus de découverte de sources d'intérêt ou encore à la Découverte d'Information (DI). La DI a pour but de constituer le corpus de documents sur lequel travaille la RI. Dans le domaine du renseignement, la tâche de DI peut permettre de découvrir des sources d'intérêt opérationnel tout en faisant face à la perpétuelle évolution du Web et à l'augmentation constante de sa taille.

1.1 Notion d'Information dans le cadre du ROSO

Dans un contexte professionnel, la notion de sources d'information sur le Web s'apparente à un support d'aide à la prise de décision. Nous abordons plus particulièrement la veille stratégique qui regroupe les techniques de recherche et de traitement de l'information

dans le domaine du renseignement. Ainsi, nous associons à une information une notion de pertinence relative à un besoin opérationnel.

1.1.1 Cycle du renseignement

Le cycle du renseignement est un processus itératif présenté sur la figure 1.1. Selon le besoin en information, ce cycle est également appelé cycle de veille ou cycle de l'information.

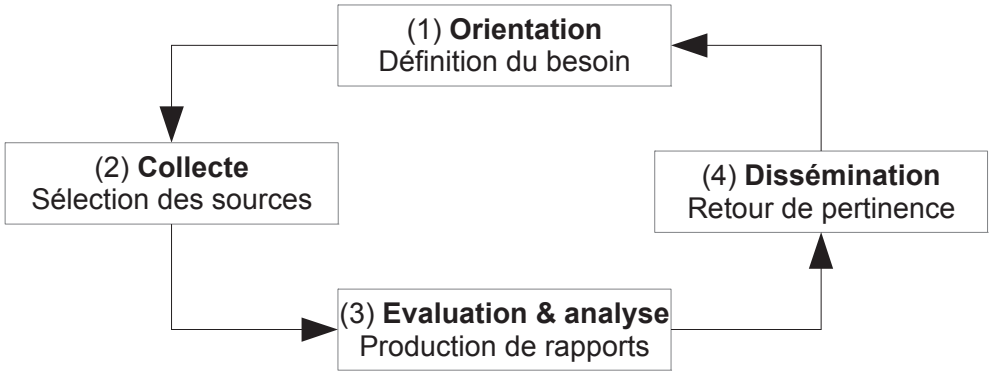


FIGURE 1.1 – Le cycle du renseignement [OTAN, 2002, OTAN, 2005, CDEF, 2008]

Le cycle du renseignement est généralement constitué de quatre étapes principales.

1. L'objectif de la phase d'orientation est de décrire d'une manière précise le besoin en information du professionnel. Afin de savoir si les renseignements obtenus pendant le cycle répondent au besoin, cette phase décrit également les conditions et les critères de succès. Le professionnel utilisant le cycle du renseignement est appelé utilisateur. Le chapitre 3, page 43, décrit notre approche de représentation du besoin utilisateur et de modélisation du profil opérationnel.
2. La phase de collecte a pour but de sélectionner des sources d'information permettant de répondre au besoin en information. Nous nous intéressons de près à cette étape qui permet la découverte de sources d'information. Cette phase est réalisée par des experts, spécialistes d'une zone géographique donnée, d'une organisation en particulier, d'une thématique et/ou d'un sujet spécifique.
3. La phase d'évaluation et d'analyse est une phase complexe qui vise à traiter les différentes informations collectées. Cette phase consiste à comprendre ces informations et à s'assurer qu'elles sont fidèles à la réalité. Elle permet de transformer une information en renseignement puisque les informations gardées ont été évaluées en terme de confiance et jugées comme répondant au besoin en information.
4. Des rapports sont ensuite établis durant la phase de dissémination et présentés aux clients. Un retour de pertinence est mis en œuvre sur ces rapports afin que l'utilisateur juge de l'intérêt de ces informations par rapport à son besoin ou afin de soulever de nouvelles interrogations, déclenchant un nouveau cycle de renseignement.

1.1.2 Capitalisation du renseignement

Lors de la dernière étape, les experts peuvent exploiter le pentagramme du renseignement pour produire leurs rapports. Le concept de pentagramme du renseignement a été proposé par l'OTAN [1] dans leur modèle Aintp-3(A) [OTAN, 2001]. Comme l'illustre la figure 1.2, il définit les cinq éléments principaux pour le renseignement :

- Person : les personnes ;
- Unit : les organisations militaires, commerciales ou associatives ;
- Place : les lieux incluant des notions purement géographiques (par exemple l'isthme de Pérécope), des notions administratives (comme la péninsule de Crimée) ou des infrastructures et bâtiments (tel que l'aéroport de Sébastopol) ;
- Event : les événements comme, par exemple, la prise d'un aéroport (Place) par des troupes (Unit) ;
- Equipment : le matériel correspondant à des moyens de transports, à de l'armement, etc.

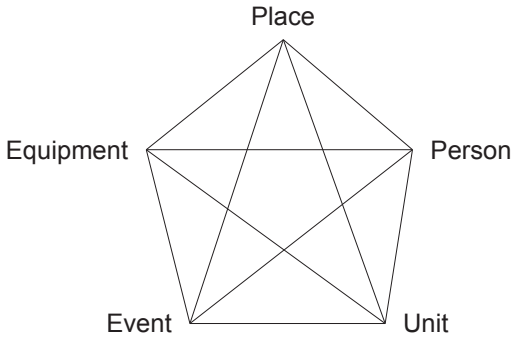


FIGURE 1.2 – Le pentagramme du renseignement d'après [OTAN, 2001]

Les travaux menés dans cette thèse concernent principalement les deux premières phases du cycle du renseignement : la phase d'orientation et la phase de collecte qui sont fortement liées. Il est en effet nécessaire de posséder une description précise et fidèle du besoin pour que les documents collectés y répondent au mieux et ainsi découvrir des sources d'intérêt. Cela nécessite des techniques de questionnement ou de modélisation permettant de cerner les objectifs dans leur globalité et le besoin en information.

1.1.3 Sources d'intérêt

Une source d'information utile au renseignement peut avoir différents aspects. Aussi, les différentes classes de renseignement se distinguent par les différents aspects des sources employées. Dans un contexte militaire, des cellules sont très souvent spécialisées dans un type de sources afin de produire leurs rapports. Une typologie des différentes classes de renseignement permet de classer les différents types de sources. Ces dernières sont illustrées via la figure 1.3 dans laquelle on distingue 4 principaux types de sources :

1. Organisation du Traité de l'Atlantique Nord : http://www.nato.int

1. Le témoignage correspondant au renseignement d'origine humaine (ROHUM, ou HUMINT, de l'anglais Human INTelligence). Un témoignage peut avoir l'aspect d'un rapport, d'une discussion, etc. et est produit par des personnes allant du collaborateur au témoin, ou tout autre personne susceptible d'avoir des informations répondant au besoin courant.
2. Les signaux électromagnétiques (ROEM, ou SIGINT pour signals intelligence) sont des sources d'information issues de radars, de sonars ou d'interceptions téléphoniques, par exemple.
3. L'imagerie satellite et aérienne fournissent des informations au renseignement d'Origine IMage (ROIM, ou IMINT pour IMages INTelligence)
4. Enfin, le Renseignement d'Origine Source Ouverte (ROSO, ou OSINT pour Open Source INTelligence) vise à exploiter les informations issues de sources ouvertes comme les sources du Web. C'est ce type de sources qui sera collecté dans la phase du cycle du renseignement qui nous intéresse. Nous détaillons, dans la section suivante, uniquement ce type de sources qui fait l'objet du cadre applicatif de cette thèse.

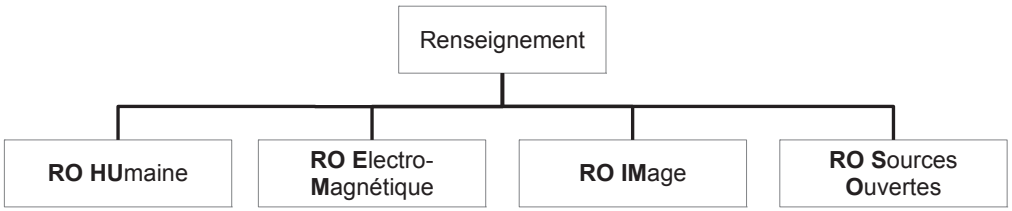


FIGURE 1.3 – Les différents types de renseignement [Mombrun, 2012]

1.1.4 Renseignement d'Origine Sources Ouvertes (ROSO)

Une source ouverte est accessible publiquement et légalement. Ces sources sont généralement accessibles *via* :

- la radio, la télévision,
- les journaux, revues et supports papier en général,
- les sites internet.

Le cadre applicatif de la thèse s'articule autour de sources utiles au ROSO et se focalise sur les sources de type site internet. En effet, la multiplication des canaux de diffusion mondiaux, notamment la démocratisation de l'accès à Internet, a provoqué l'explosion du volume d'information de type sources ouvertes à tel point qu'il est devenu difficile pour les experts de gérer ces sources sans outil adapté. Or, ce volume important d'information disponible sur le Web peut contenir des informations pour le renseignement. Yann Mombrun dans ses travaux de thèse sur *l'évaluation de l'Information disponible sur Internet appliquée au renseignement d'origine sources ouvertes* met en avant quelques exemples [Mombrun, 2012] :

- des informations tirées de blogs de sympathisants d'une activité terroriste,
- des analyses ethnologiques d'une région issues d'une thèse,
- des témoignages extraits de reportages vidéo.

De plus, les sites internet fournissent des informations utiles à l'intelligence économique ou à la veille stratégique sur tous les sujets et thématiques à un rythme soutenu. L'arrivée du Web 2.0 a permis de mettre en place des flux de presse, les réseaux sociaux, qui émettent une énorme quantité d'information en continu.

Les sites internet présentent un certain nombre d'avantages mais également des inconvénients non négligeables pour le ROSO. Les informations contenues sur un site sont facilement échangeables entre services ou experts puisqu'il suffit de fournir le lien hypertexte (URL) vers le document contenant l'information. Ainsi, l'avantage de cette liberté d'accès est que cette information est rarement classifiée : elle ne fait pas l'objet d'une protection nécessitant une demande officielle ou justifiée contrairement aux sources protégées (radar, satellite, personnel) qui demandent une dissémination via canaux sécurisés. Enfin, l'accès aux sites internet ne nécessite que très peu de moyens. Le prix pour l'accès au Web est dérisoire contrairement à l'utilisation d'un radar ou d'un satellite. Les sites internet peuvent donc contenir des informations intéressantes pour le renseignement, de qualité, d'actualité, peu coûteuses en terme d'acquisition, faciles à manipuler et nombreuses sur le Web.

Ce volume d'information disponible peut avoir également des inconvénients. Il y a beaucoup de bruit lié à la surcharge et à la diversité de l'information sur le Web. Autrement dit, les informations intéressantes sont noyées dans la multitude d'informations présente sur la toile. La tâche de détection des informations est donc rendue difficile face à ce flot continu d'information. De plus, la facilité d'émission de l'information sur un site internet, rendu possible notamment grâce au Web 2.0, rend la qualité des informations hétérogènes. L'information peut être pertinente ou non, objective ou non, ajoutée par un expert renommé dans son domaine ou par un inconnu, etc. Certaines informations erronées sont même volontairement publiées sur le Web dans le simple but de désinformer. Ces sites, quelque soit la fiabilité et la pertinence de leur information, sont accessibles de la même manière sur Internet.

1.2 L'information sur le Web

Les informations présentes sur le Web peuvent être utiles aux experts du renseignement mais aussi à toute personne lambda ayant un besoin en information. Les informations y sont mélangées, en abondante quantité et disponibles dans des formats hétérogènes. Que ce soit pour l'expert ou pour l'utilisateur lambda, une partie des problématiques de recherche d'information sur le Web sont communes. De plus, la recherche d'information de l'expert du renseignement est complexifiée par des besoins informationnels spécifiques qui peuvent être sensibles.

1.2.1 État du Web

Depuis 1989, date de sa création, le Web n'a cessé d'évoluer et les techniques de Recherche d'Information (RI) avec lui. Au début du Web, l'URL d'une page devait être connue pour accéder à son information. Avec l'explosion du Web, au début des années quatre-vingt-dix, des outils de RI ont été créés pour faciliter l'accès aux informations. A partir de 1995, des moteurs de recherche et des annuaires, comme respectivement Altavista et Yahoo !, ont vu le jour et sont devenus rapidement indispensables. Depuis, le Web est

devenu le service le plus populaire d'Internet et contient la plus grande base de données existante. Différents facteurs ont influencé cette évolution :

- le développement du Web dynamique [Brewington & Cybenko, 2000], appelé Web 2.0, a fait évoluer le Web vers plus de simplicité et d'interactivité. Son utilisation ne nécessite pas de grandes connaissances techniques ni informatiques et il permet à chacun de contribuer sous différentes formes (forum, blog, etc.). Le nombre de pages a explosé et l'information qu'elles véhiculent est perpétuellement modifiée ou supprimée, ce qui rend l'information très volatile. Le site WorldWideWebSize [2] tente d'estimer la taille du Web via un calcul sur les index des principaux moteurs de recherche d'aujourd'hui : Google, Yahoo et Bing. La taille du Web correspond à la somme des différents index à laquelle est soustrait le recouvrement des index des différents moteurs. En novembre 2010, la taille du Web était estimée à 2,73 milliards de pages et 11, 48 milliards au mois d'octobre 2011, preuve de l'augmentation importante de la taille du Web.
- avec ses 2, 4 milliards d'utilisateurs en 2012 [3], le Web contient un panel d'information aussi multiple et varié que les centres d'intérêt de ses utilisateurs. Avec le Web 2.0, l'utilisateur est un acteur actif du Web et a la capacité de produire de l'information sur le Web. Il a désormais accès à de l'information très hétérogène : nature, origine et support (texte, image, audio et vidéo). Cette hétérogénéité de l'information et sa fragmentation rendent la collecte d'information plus difficile, d'autant qu'elle doit également faire face au manque de structuration de l'information, et aux problèmes de langue.

A mesure que le volume des données augmente et que les médias d'information se diversifient, applications et utilisateurs éprouvent des difficultés pour accéder facilement aux données pertinentes.

1.2.2 La recherche d'information sur le Web

La RI est devenue une problématique majeure, d'autant qu'elle représente une grande part de l'activité des internautes. La RI est étroitement liée aux moteurs de recherche qui font état de 17, 1 milliards de recherches effectuées durant le mois d'août 2011 aux Etat-Unis [4], contre 16, 6 milliards en octobre 2010 et 14, 3 milliards en 2009. En 2012, Google comptabilise 1,2 trillion de requêtes envoyées [5] à son moteur de recherche.

Les moteurs de recherche fournissent, pour chaque requête d'utilisateur, un nombre important de résultats. Une surcharge informationnelle est présente parmi ces résultats, celle-ci peut très bien correspondre à de l'information pertinente, comme à de l'information secondaire voire à du bruit.

La RI est étroitement liée au besoin d'information de l'utilisateur. Schneidermann, Byrd, et Croft définissent ce besoin d'information comme "*le ressenti d'un besoin pour de l'information qui entraîne une personne à utiliser en premier lieu un système de recherche d'information* [Shneiderman et al., 1997]". Il s'agit donc d'enrichir sa connaissance, accroître son savoir sans lien immédiat à des prises de décision. Cependant, les besoins des

2. http://www.worldwidewebsize.com/
3. http://royal.pingdom.com/2013/01/16/internet-2012-in-numbers/
4. http://www.comscore.com/index.php/Press_Events/Press_Releases
5. http://royal.pingdom.com/2013/01/16/internet-2012-in-numbers/

utilisateurs sont diverses. Ce qui a donné à la conception de différents modèles d'interaction entre l'utilisateur et l'outil de recherche d'information. Trois types de modèles sont décrits par Stephenson [Stephenson, 1989] :

- *retrieving* : l'utilisateur possède une connaissance du sujet recherché, ainsi qu'une maîtrise des techniques de recherche d'information.
- *searching* : l'utilisateur possède une connaissance du sujet recherché mais maîtrise moins bien les outils de recherche d'information.
- *browsing* : l'utilisateur n'a pas de notion précise de ses besoins. Il ne peut donc pas les traduire en termes précis. L'outil de recherche doit l'aider à formuler sa requête.

Outre le volume important et l'hétérogénéité des informations sur le Web, la RI doit également prendre en compte la diversité des besoins de l'utilisateur.

1.2.3 RI, DI et ROSO

Représenter précisément le besoin utilisateur est une phase clé de la RI qui influence directement la pertinence des résultats retournés. Cette représentation peut être difficile si l'expression du besoin est insuffisante ou ambiguë. Dans le cadre du ROSO, la représentation du besoin utilisateur est d'autant plus complexe que le besoin peut être très spécifique. Les experts du renseignement travaillent sur des sujets sensibles et les informations qu'ils recherchent sont plus difficiles à trouver qu'une information basique. En effet, l'un des critères utilisé par les moteurs de recherche pour classer et ordonner leurs sources est la popularité. Or, les sujets sensibles sont souvent impopulaires et parfois même non-indexés par les moteurs de recherche. L'important nombre de pages à prendre en compte et la multiplicité des besoins sont deux limitations de la RI auxquelles les experts du renseignement doivent faire face. Malgré un ensemble d'outils présentés dans le chapitre 2.1, la RI comme la RI personnalisée reviennent à chercher une aiguille dans une botte de foin lorsqu'il s'agit d'un besoin sensible. De plus, le processus de recherche est souvent réduit à un traitement linéaire "besoin-réponse". Or, le besoin de l'expert peut évoluer en même temps qu'accroît sa connaissance sur le sujet de recherche. Ainsi, la DI, en explorant le Web et en ciblant sa collecte, offre l'avantage de constituer un corpus de sources d'intérêt en fonction du besoin de l'utilisateur. Il permet de passer outre la popularité d'une source et d'éviter les problèmes liés à l'indexation de pages contenant de l'information sensibles.

1.3 Axes de recherche

Les travaux menés dans cette thèse visent à aider les experts du renseignement dans leur tâche de veille stratégique sur le Web. Ainsi, nos objectifs sont de **pallier le manque de représentativité du besoin opérationnel dans le cadre du ROSO** au travers d'un système de découverte de sources d'intérêt faisant **abstraction de la popularité** des sources et capable de considérer l'**évolution du besoin** de l'expert du renseignement. Nos travaux s'articulent autour de deux axes de recherche : la découverte de nouvelles sources d'information et la modélisation du besoin opérationnel. Ces deux axes sont fortement connexes dans la mesure où la représentation du besoin en information permettra de découvrir et de fournir des sources pertinentes à l'expert.

1.3.1 Notion de découverte d'information sur le Web

L'exploration du Web est une activité importante qui a lieu en amont de la RI. Elle vise à collecter l'ensemble des pages présentes sur le Web afin de les indexer pour les rendre exploitables par les moteurs de recherche. Des robots d'indexation sont chargés de cette tâche. Ils explorent aléatoirement le Web en passant de page en page via des liens hypertextes et indexent chaque page visitée. La notion de DI est utilisée pour décrire un robot d'indexation qui utilise le besoin en information de l'utilisateur pour guider son exploration du Web et visiter uniquement des pages qu'il juge pertinentes. De manière imagée, il s'agit de réduire la taille d'une botte de foin afin d'y trouver plus facilement l'aiguille. Dans nos travaux, la **pertinence** des pages est déterminée **sans considération de leur popularité** en se basant **uniquement sur leur contenu informationnel**. Cet axe de travail a pour but de proposer un **nouveau modèle de découverte de sources évolutif en fonction du besoin utilisateur**. La capacité de découverte de sources de notre modèle, fortement liée au calcul de pertinence d'une page, est évaluée dans une première expérimentation utilisateur.

1.3.2 Modélisation évolutive du besoin

Notre modèle de découverte de sources exploite le besoin utilisateur. Ce besoin pouvant évoluer au cours du temps, il est modélisé au sein d'un **profil utilisateur évolutif**. Nos travaux visent à couvrir le besoin dans son ensemble en le modélisant **globalement** tout en prenant en compte les **spécificités de langage** du sujet de recherche. Cette modélisation du besoin et la construction du profil utilisateur font parties d'un second axe de recherche. L'**intégration flexible** du besoin de l'utilisateur au sein de la découverte de sources, jumelée à un **retour de pertinence**, permet de proposer une meilleure réponse à l'utilisateur. Les capacités d'**adaptabilité** du système et d'**évolution** de la modélisation du besoin sont évaluées dans une seconde expérimentation.

1.4 Organisation du manuscrit

Ce manuscrit est composé de trois parties distinctes correspondant aux travaux de recherche accomplis. La première partie, bibliographique, est scindée en 2 chapitres. Le chapitre 2 correspondant à l'étude des travaux existants autour de la RI et de la DI. Il introduit les travaux sur la RI classique et personnalisée en décrivant les modèles et systèmes existants. Une description des techniques d'exploration du Web pour la DI y est également présente. Le chapitre 3 décrit les approches de modélisation du besoin utilisateur et de retour de pertinence. Les modèles de représentation et d'acquisition du profil utilisateur pour la RI et la DI y sont présentés. Ce chapitre contient également une description des techniques utilisées pour affiner la représentation du profil utilisateur au travers de retours de pertinence.

La seconde partie est consacrée à l'approche proposée dans cette thèse. Le chapitre 4 positionne notre approche par rapport à l'état de l'art décrit dans la partie précédente. Puis, nos contributions dans le domaine de la modélisation du besoin utilisateur et de la découverte de nouvelles sources sont décrites dans deux chapitres distincts. Le chapitre 5 met en avant la difficulté liée à la compréhension du besoin opérationnel. Notre choix d'un modèle de profil utilisateur et la construction de celui-ci y sont présentés. Ce chapitre se

termine par la description de notre modèle d'évolution du profil exploitant les retours de pertinence. Enfin, le chapitre 6 est consacré à la recherche de nouvelles sources d'intérêt et au système mis en place pour explorer le Web et mesurer la pertinence d'une page.

Une troisième partie permet de décrire l'architecture de notre système et de l'évaluer au travers de prototypes et de projets opérationnels. Le chapitre 7 présente le protocole expérimental et les résultats de notre validation expérimentale de la pertinence de notre approche ainsi que du calibrage des paramètres de notre système. Le chapitre 8 décrit l'évolution subie par le système en présentant le développement du prototype jusqu'à son intégration projet.

Dans la conclusion, une synthèse des résultats obtenus et une analyse critique des apports et des limites de notre modèle sont présentés. Les possibles perspectives pour la suite de ces travaux et les axes de recherche encore ouverts sont également introduits dans ce dernier chapitre.

Première partie

État de l'art

CHAPITRE 2

DE LA RECHERCHE D'INFORMATION À LA DÉCOUVERTE DE SOURCES D'INFORMATION

L'étude des méthodes de Recherche d'Information (RI) est un domaine de recherche qui existe depuis de nombreuses années. Il touche de multiples secteurs et sous-domaines de recherche de part la diversité des supports de l'information. À l'origine, la RI est issue des sciences de l'information et à la bibliothéconomie dont l'objectif est de représenter des documents afin de retrouver plus facilement les informations qu'ils contiennent. L'idée d'utiliser des machines pour automatiser et faciliter la RI dans les bibliothèques est née avec l'apparition des ordinateurs et a été popularisée par Bush [Bush, 1945]. Avec l'émergence du Web, la RI a pris une tout autre dimension jusqu'à se démocratiser. Aujourd'hui, l'utilisation de moteurs de recherche a été massivement adoptée par les utilisateurs d'Internet. La première partie de ce chapitre est consacrée aux méthodes et aux techniques employées pour améliorer la RI. Une seconde partie présente le travail en amont de la RI. Il s'agit de la collecte de documents, apparentée à de la découverte d'information (DI), lorsqu'il s'agit de cibler la collecte à des nouveaux documents d'intérêt. Les approches d'exploration ciblée du Web pour la collecte seront mises en avant et les limites et problématiques résultantes seront soulevées dans ce chapitre.

2.1 La recherche d'information

À mesure que le volume des données augmente et que les médias d'information se diversifient, le Web devient trop volumineux pour permettre aux utilisateurs d'accéder facilement aux données pertinentes. Considérant ce problème, des systèmes de RI ont été développés. Les plus connus sont les annuaires et les moteurs de recherche. D'autres systèmes vont plus loin pour personnaliser les résultats en exploitant un profil utilisateur. Certains systèmes sont basés sur des techniques de recommandation et de filtrage collaboratif afin de fournir des résultats personnalisés. Dans cette partie, nous présentons un résumé de l'état de l'art sur la RI et la RI personnalisée.

2.1.1 Les systèmes de RI

Lors d'une recherche d'information sur le Web par un utilisateur, on distingue deux principaux types de méthode : la recherche par navigation, la recherche par requête.

La recherche d'information par navigation correspond à une recherche de page en page, via les hyper-liens, par l'utilisateur qui analyse chacune d'entre elles. Ce type de recherche est pratique pour explorer le Web et lire son contenu, toutefois elle ne convient pas à la recherche d'une information spécifique. En effet, le nombre important de pages sur le Web rend la possibilité, pour l'utilisateur, de trouver ce qu'il recherche très peu probable. Les annuaires peuvent aider à la recherche si l'utilisateur sait catégoriser son besoin en information.

La recherche d'information par requête, au travers de moteurs de recherche, est beaucoup plus employée. Elle permet de retrouver des documents Web via des index représentant le Web, et ce en quelques secondes. Une requête est une formulation du besoin en information de l'utilisateur. L'utilisateur la construit habituellement en spécifiant un ensemble de mots-clés en fonction de son besoin. Les moteurs de recherche font état de 17 milliards de recherches effectuées durant le mois de novembre 2012, aux Etat-Unis [1], contre 16, 6 milliards en octobre 2010 et 14, 3 milliards en 2009. Parmi ces moteurs de recherche, Google est largement utilisé avec 67% des requêtes qui lui sont adressées, aux Etat-Unis en novembre 2012. Il est suivi par les moteurs de recherche Bing et Yahoo (cf. tableau 2.1).

Moteur de recherche	Partage des requêtes de recherche
Google	67%
Bing	16,2%
Yahoo !	12,1%
Autres	4.7%

TABLE 2.1 – Partage des requêtes entre moteurs de recherche en mars 2012 aux Etats-Unis

Ces moteurs de recherche sont des systèmes de RI permettant de sélectionner l'ensemble des documents indexés répondant à un besoin de l'utilisateur représenté par une requête en langage naturel ou structurée. De tels systèmes fonctionnent en respectant deux phases : l'indexation et l'interrogation.

a. L'indexation

La phase d'indexation a pour but de transformer les documents collectés sur le Web en un ensemble de descripteurs afin de représenter le contenu de ces documents [Salton & McGill, 1983]. Ces descripteurs sont souvent basés sur une structure d'ensembles de termes ou de groupes de termes. L'indexation consiste à détecter les mots les plus représentatifs du contenu du document selon trois étapes :

- La simplification : les mots à fortes fréquences sont supprimés du document. Ils font références à une liste de mots anti-lexiques, fréquemment appelée *stoplist*.
- Le regroupement : les formes morphologiquement liées par leur racine sont regroupées. Pour cela, les terminaisons (genre, conjugaison, déclinaison) des mots sont

1. http://www.comscore.com

supprimées. Les processus de racinisation et de lemmatisation permettent de déterminer la fréquence d'apparition de mots sans ternir compte de leurs variations.
– La pondération : chaque terme est pondéré selon son degré d'importance dans le document. La formule de pondération la plus connue est basée sur la fréquence de termes (TF) et la fréquence inverse du document ayant ces termes (voir TF*IDF section 3.1.2, page 46).

Cette méthode d'indexation repose donc sur des algorithmes qui associent des descripteurs aux documents. Chaque mot du document est ainsi potentiellement un index de ce document.

Une autre approche consiste à indexer les documents manuellement. Un expert dans un domaine donné peut choisir lui même les termes qu'il juge pertinent dans le document. L'indexation est ainsi contrôlée et la qualité est accrue. Cependant, une telle méthode n'est pas applicable sur une collection trop importante de documents, et sa fiabilité repose sur les connaissances de l'expert. Aussi, la méthode d'indexation semi-automatique combine les avantages des deux méthodes précédentes : les termes sont extraits automatiquement mais la décision finale sur les descripteurs est laissée à l'expert du domaine [Desmontils & Jacquin, 2002]. Dans ces approches, la taille de la collection de documents à indexer reste une contrainte.

b. L'interrogation

Une fois les documents indexés, l'utilisateur peut interroger le système, via un processus d'interaction, afin que celui-ci réponde aux besoins de l'utilisateur. Trois phases sont à distinguer dans le processus d'interrogation [Maisonnasse, 2008] :

La formulation : l'utilisateur formule son besoin sous la forme d'une requête.

La représentation : la requête est traduite par le système afin d'être comparé aux descripteurs des documents indexés.

La correspondance : les documents présents dans l'index répondant à la représentation de la requête de l'utilisateur sont sélectionnés.

Le système calcule un score de similarité entre les descripteurs des documents indexés et la représentation de la requête afin d'ordonner les documents indexés et de retourner les plus pertinents à l'utilisateur.

2.1.2 Les modèles de systèmes de RI

La mise en place du processus d'indexation et celle du processus d'interrogation permettent de réaliser un système de RI sous la forme d'un *processus en U*. Comme le montre la figure 2.1, un tel système permet de répondre aux besoins en information de l'utilisateur en lui fournissant un ensemble de documents pertinents, parmi ceux collectés, en fonction de sa requête.

Si les modèles de système de RI s'accordent sur ce *processus en U*, ils diffèrent sur le modèle d'appariement requête-document pour le calcul de similarité. Un modèle de RI peut être défini par le quadruplet suivant [Baeza-Yates et al., 1999] :

D : L'ensemble des documents indexés.

R : L'ensemble des requêtes.

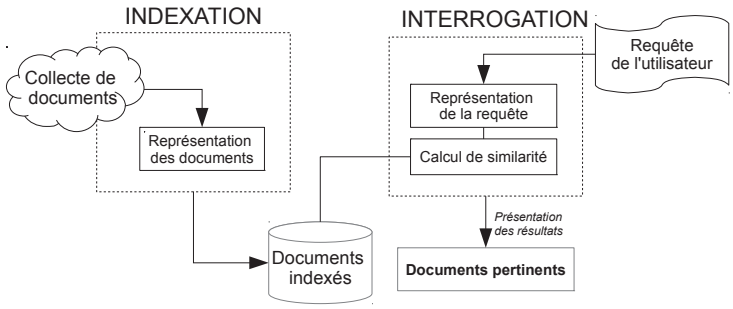


FIGURE 2.1 – Système de RI

F : Modèle de représentation des documents et des requêtes

S(r,d) : Fonction de similarité entre la requête r et le document d.

Où F et S(r,d) varient selon le modèle utilisé. Ces différents modèles se regroupent en 3 principales catégories présentées ci-dessous.

a. Le modèle booléen

Dans ce modèle, les descripteurs des documents sont des ensembles de mots-clés [Salton, 1969]. Le document d est ainsi représenté par une conjonction logique de termes t_i avec i $\in$ [1,n] où n est un nombre entier :

$$d = t_1 \wedge ... \wedge t_n$$

Les termes de la requête forment ainsi une expression booléenne reliés par les opérateurs logiques d'union, d'intersection et de différence. Une requête q peut se présenter, par exemple, sous la forme suivante :

$$q = t_1 \wedge (t_2 \vee t_3)$$

La fonction de similarité consiste alors à considérer les documents dont les termes sont présents dans la requête. Considérant le document d et la requête r, le résultat est binaire :

$$S(r, d) = 0, 1$$

La simplicité de la fonction en fait sa limitation puisque le résultat n'est pas nuancé. Le document répond pleinement à la requête ou alors il n'y répond pas du tout.

b. Le modèle vectoriel

Ce modèle représente les documents et les requêtes par des vecteurs de termes pondérés [Salton, 1971]. Un document d est représenté par son vecteur $\vec{d}$ tel que :

$$\vec{d} = (w_1, ..., w_n)$$

Où $w_i \in$ [0,1] est le poids du terme i appartenant au vocabulaire d'indexation. Le poids du terme w_i est généralement calculé grâce à la mesure TF*IDF (voir TF*IDF section 3.1.2,

page 46). Une mesure de similarité calculant la distance entre le vecteur du document $\vec{d}$ et le vecteur de la requête $\vec{r}$ permet de définir la pertinence du document d par rapport à la requête r. La similarité cosinus (ou mesure cosinus) est fréquemment utilisée. Elle permet de calculer l'angle entre deux vecteurs. Plus cet angle est petit et plus les deux vecteurs sont similaires. La fonction de similarité entre le document d et la requête r est définie ainsi :

$$S(r, d) = \cos(\vec{d}, \vec{r}).$$

La pertinence est ainsi nuancée contrairement au modèle booléen et permet d'obtenir un plus grand nombre de résultats approximatifs.

c. Le modèle probabiliste

Ce modèle est basé sur un calcul visant à estimer la probabilité qu'un document d soit pertinent pour une requête r, notée $\mathrm{P}_r(\mathrm{relevant}/d)$. La probabilité que le document ne soit pas pertinent est également prise en compte, elle est notée $\mathrm{P}_r(\overline{relevant}/d)$ [Nottelmann & Fuhr, 2003]. Ainsi, la similarité est calculée ainsi :

$$S(r, d) = \frac{P_r(relevant/d)}{P_r(\overline{relevant}/d)}$$

Afin de calculer la probabilité qu'un document soit pertinent ou non pertinent, les différentes formules existantes se basent sur les probabilités conditionnelles. Celles-ci prennent en considération la présence d'un terme de la requête dans un document pertinent ou son absence dans un document non pertinent. Le modèle probabiliste a l'avantage de pouvoir prendre en compte une multitude d'éléments de contexte. Le modèle de pondération Okapi BM25 [Robertson & Walker, 1994] se base notamment sur la rareté des termes ou encore la longueur des documents. D'autre approches étendent ce modèle en prenant également en compte le poids et la fréquence des termes [Maron & Kuhns, 1960, Jones et al., 2000] ainsi que la longueur des documents [Robertson & Jones, 1976, Chowdhury, 2004].

2.1.3 Les limites de la RI "classique"

La recherche d'information se fait essentiellement au travers des moteurs de recherche. Ces moteurs se basent sur leurs calculs de pertinence des documents en fonction des requêtes envoyées par les utilisateurs. Cette approche considère une requête invariablement d'un utilisateur à l'autre, sans prendre en considération le besoin divergent des utilisateurs et le contexte des recherches. Prenons l'exemple de la requête avec le mot-clé *chat*. Cette requête est ambiguë puisque le terme *chat* est polysémique. Il peut faire référence à l'animal comme au mot anglais correspondant aux discussions en ligne. La RI devrait prendre directement en considération le besoin utilisateur sous-jacent.

Dans ce même exemple, le contexte pourrait également être utile puisqu'une recherche de *chat* dans un pays anglo-saxon a beaucoup moins de chance de faire référence au mammifère que dans un pays français.

La RI travaille sur un corpus de pages dont la taille tend à s'approcher de celle du Web. Il devient donc difficile d'y trouver de l'information précise, adaptée au besoin spécifique de l'utilisateur, avec des systèmes de RI générique. De plus, l'hétérogénéité des pages présentes dans ce corpus complexifie la recherche d'information.

Ces limitations ont ouvert la voie à la RI personnalisée.

2.1.4 La RI personnalisée

La diversité et l'ambiguïté des besoins en information des utilisateurs, ainsi que la rapide croissance des ressources d'information hétérogènes (données structurées, documents textuels, images, vidéos) sur Internet, ont conduit à l'émergence de la RI personnalisée. Elle a pour but d'adapter le processus de recherche en fonction du contexte et/ou du profil de l'utilisateur afin de répondre au mieux à ses besoins en information. La RI personnalisée s'appuie autant sur l'exploitation des connaissances de l'utilisateur que sur ses centres d'intérêt ou que sur ses préférences de recherche qui sont modélisés dans un profil utilisateur (voir chapitre 3, page 43). Dans cette partie, nous nous intéressons aux approches de personnalisation des systèmes de RI. La RI personnalisée peut opérer à différentes étapes de la recherche que nous résumons en quatre catégories de personnalisation (voir figure 2.2) :

- Personnalisation de présentation (a) : Les systèmes adaptent la présentation des résultats dans un processus de recherche mettant à contribution l'utilisateur.
- Personnalisation des résultats de recherche (b) : L'ordre des résultats retournés est personnalisé en fonction du profil de l'utilisateur.
- Personnalisation de la représentation du besoin utilisateur (c) : La requête de l'utilisateur est modifiée afin de la rendre plus pertinente ou de la désambiguïser en considérant le profil de l'utilisateur.
- Personnalisation de la collection de documents (d) : alors que la RI s'opère sur un ensemble de documents collectés et indexés, cette catégorie vise à personnaliser cette collection de document pendant une phase de découverte et d'exploration du Web.

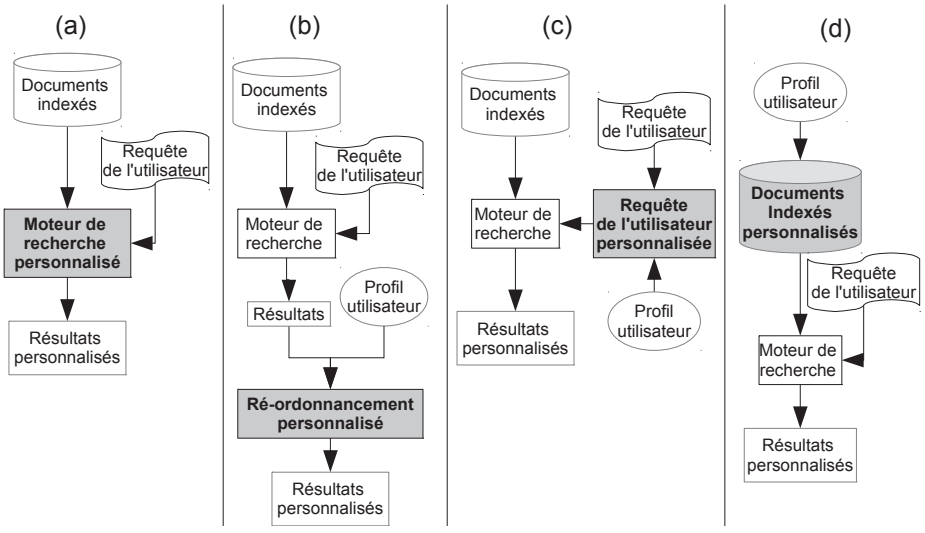


FIGURE 2.2 – Les catégories de systèmes de RI personnalisée

Nous ne détaillons, ci-dessous, que les trois premières catégories et les différentes techniques de personnalisation employées au travers des systèmes existants, qui permettent de fournir à l'utilisateur un résultat de recherche en adéquation avec son besoin. Le chapitre suivant est, en partie, consacré à la personnalisation de la collection de documents qui intervient lors de la découverte de sources, en amont du travail des moteurs de recherche.

a. Personnalisation de présentation

Les moteurs de recherche traditionnels fournissent des résultats sous la forme d'une longue liste de documents en fonction de leur similarité avec la requête de l'utilisateur. Cet utilisateur parcourt la liste et consulte les documents qui lui semblent être pertinents en fonction de leur titre et de leur snippet [2]. Afin de simplifier la recherche, plusieurs services de recherche en ligne, comme Yippy Search [3], Kartoo [4] et [Zamir & Etzioni, 1999], proposent de regrouper les documents de la liste qui concernent un même sujet. Yippy Search est basé sur un service de regroupement, dont le terme plus généralement employé est *clustering*, fourni par Vivisimo [5]. Le regroupement se fait sous la forme de dossiers et sous-dossiers de sujets. Le meta-chercheur [6] Kartoo permet à l'utilisateur de visualiser une bref description du site et des informations supplémentaires telles que la pertinence du site par rapport à la requête entrée par l'utilisateur. Ces moteurs de recherche peuvent être considérés comme des systèmes de RI personnalisée adaptatifs car l'utilisateur peut personnaliser l'affichage des résultats et sélectionner le *cluster* répondant le mieux à ses besoins. Le regroupement par sujet des documents doit se faire aussi rapidement que lors d'une recherche sur un moteur de recherche classique. Les algorithmes de regroupement exploitent le snippet du document plutôt que tout son contenu afin de gagner en temps de traitement. Les choix de regroupement doivent se faire de manière à faciliter le travail de recherche de l'utilisateur : le sujet de chaque regroupement doit être accompagné d'une courte description permettant à l'utilisateur de choisir rapidement le plus pertinent. L'utilisateur peut ainsi personnaliser la présentation des résultats en sélectionnant le regroupement adéquat.

b. Personnalisation des résultats de recherche

La RI classique au travers des moteurs de recherche fournit un ensemble de documents au sein d'une liste ordonnée par pertinence. La place d'un document se fait via le calcul du score d'appariement entre la représentation du document avec la requête courante (voir l'interrogation b. page 15). Une catégorie d'approches de personnalisation de la RI consiste à modifier cette liste pour qu'elle soit ordonnée selon le besoin de l'utilisateur (voir catégorie b de la figure 2.2 page 18).

Adaptation du score d'appariement Les systèmes de RI personnalisée, fournissant à l'utilisateur une liste de documents réordonnancée selon leur besoin, se basent généralement sur le score original du document qui est modifié en fonction du profil de l'utilisateur. C'est notamment le cas de l'approche employée dans [Mc Gowan, 2003] où le vecteur de termes pondérés, permettant de calculer le score de similarité entre le document et les centres d'intérêt de l'utilisateur, est combiné au score d'appariement original du document.

L'approche employée par [Sieg et al., 2007] repose sur une représentation du profil utilisateur basée sur les concepts. Pour chaque document de la liste de résultats, le score de similarité $S(d,r)$ du document d avec la requête r est dans un premier temps calculé, en utilisant la mesure cosinus. Ensuite, la similarité entre le document et chaque concept

2. courte description du contenu
3. http://search.yippy.com/
4. http://fr.kartoo.com/
5. http://vivisimo.com/
6. Outil qui puise ses informations à travers plusieurs moteurs de recherche

du profil utilisateur est calculée pour identifier le concept c le plus pertinent. Un dernier calcul de similarité $S(r,c)$ permet de mesurer la similarité entre le concept retenu et la requête. Le score du document est obtenu ainsi :

$$Score(d) = S(d,r) * S(r,c) * IS_c$$

Où IS_c est le poids représentant l'intérêt du concept c dans le profil utilisateur.

Comme évoqué dans le chapitre précèdent, le moteur de recherche Google propose un système de personnalisation des résultats de recherche en fonction de l'historique de recherche. Un profil utilisateur est créé via l'outil Google Web History [7] et il est utilisé afin de fournir un résultat de recherche en fonction de l'utilisateur : les documents retournés sont valorisés s'ils sont présents dans l'historique de recherche.

Exploitation de la structure hyperliens L'ordonnancement personnalisé de la liste de résultats peut également utiliser des techniques liées à la structure des liens entre les pages. Les approches se basent généralement sur l'utilisation de l'algorithme Hyperlinked Induced Topic Search (HITS) [Kleinberg, 1999] (voir page 31), ou sur l'algorithme PageRank (PR [Page et al., 1999] voir page 31) du moteur de recherche Google. Le PR permet de mesurer l'importance des pages du Web sans considérer la requête ou le besoin de l'utilisateur. Il assigne à chaque page une valeur proportionnelle au nombre de fois que serait susceptible de passer, sur cette page, un utilisateur parcourant le Web aléatoirement. Le parcours de l'utilisateur est considéré comme une marche aléatoire sur le graphe du Web où les sommets sont des pages et les arcs des hyperliens. Plus le nombre de liens pointant vers une page est important et plus le PR de cette page sera élevé. Se basant sur cet algorithme, Chirita et al. [Chirita et al., 2004] propose une plate-forme d'ordonnancement personnalisé en fonction des pages Web en favoris ou des pages fréquemment visitées par l'utilisateur. Le profil utilisateur est exploité au travers d'un module qui étend l'algorithme HITS : son but est de considérer le besoin de l'utilisateur et de collecter des pages Web, appelées *hub*, qui sont des pages dont les liens pointent vers des pages d'information pertinente considérant le besoin de l'utilisateur. L'approche propose une version personnalisée du PR afin de classer les pages collectées répondant au besoin utilisateur.

Une autre approche de ré-ordonnancement des résultats est proposée dans [Haveliwala, 2003]. L'algorithme du PR est également modifié, et appelé TSPR pour *Topic Sensitive* PageRank. Son avantage par rapport à l'approche précédente est que son algorithme se veut sensible aux intérêts de l'utilisateur. Pour ce faire, l'approche utilise les hauts-niveaux de l'ODP (Open Directory Project) comme étant des domaines d'intérêt. L'ODP est un annuaire très complet de pages Web annotées. Le score des pages n'est plus uniquement celui de PR, mais une combinaison linéaire des scores du PR modifiée selon la similarité de la page avec les domaines d'intérêt. Le PR utilisé pour une page est choisi selon la similarité entre la requête utilisateur et les différents domaines d'intérêt.

Exploitation du profil utilisateur IfWeb [Asnicar & Tasso, 1997] est un système reposant sur la modélisation du profil utilisateur construit à partir de ressources fournies

7. http://www.google.com/history

par l'utilisateur : mots-clés, textes, documents. L'utilisateur peut définir si la ressource représente un exemple positif ou négatif par rapport à son besoin. Le profil utilisateur est représenté par un réseau de noeuds correspondant à des concepts extraits des ressources. Le système comprend un mode de fonctionnement appelé *document search* : il navigue de façon autonome sur le Web à partir d'un document fourni par l'utilisateur et propose à l'utilisateur les pages visitées les plus pertinentes. L'objectif est de proposer à l'utilisateur un assistant personnel à la navigation capable de rechercher et de filtrer des documents qui sont adaptés à ses besoins.

Le système Wifs, décrit dans [Micarelli & Sciarrone, 2004], utilise une représentation proche du système précédent. La différence est que les concepts, permettant de construire le profil utilisateur, sont explicités par des experts du domaine et non pas implicitement extraits des ressources fournies par l'utilisateur. La modélisation des documents change de l'approche de modélisation vectorielle habituellement employée en RI. Le profil utilisateur est modélisé par un ensemble de termes T_i, avec i ∈ [1,n], appelés *planètes*. Ces planètes sont des concepts extraits des documents de l'utilisateur et connus des experts du domaine. Ils sont reliés à des termes, appelés *satellites*, non connus par les experts, mais qui co-occurent avec les planètes dans les documents. La fréquence d'apparition d'un satellite est calculé comme suit :

$$Occ(t) = c_1 * freq_{corps}(t) + c_2 * freq_{titre}(t)$$

Où c_1 et c_2 sont des constantes, $freq_{corps}(t)$ est la fréquence du terme t dans le corps du document, et $freq_{titre}(t)$ est la fréquence du terme t dans le titre du document. Le système exploite ce profil pour personnaliser la recherche effectuée sur le moteur de recherche ALTAVISTA. La modélisation des documents retournés par le moteur de recherche est la même que la modélisation du profil utilisateur, à la différence que le lien entre une planète et un satellite n'est pas pondéré. Wifs ré-ordonne les documents retournés en filtrant et en analysant les pages Web HTML. L'évaluation d'un document est basée sur le besoin de l'utilisateur via un vecteur $\vec{Pert}$: ce vecteur est construit au travers du calcul de la pertinence $Pert_i$ du terme i aux vues du profil utilisateur, de sa requête et des termes fournis par les experts. Les expériences montrent que la satisfaction apportée au besoin utilisateur est améliorée de 34% avec Wifs par rapport à un moteur de recherche non personnalisé.

Un profil utilisateur, construit à partir de requêtes et/ou de l'analyse des résultats de recherche, est exploité par Speretta and Gauch [Speretta & Gauch, 2005]. Une représentation ensembliste du profil est utilisée en modélisant les centres d'intérêt par des concepts de l'ODP. Ils utilisent un système externe qui reclasse les résultats de recherche de Google en donnant plus d'importance aux documents en relation avec les intérêts de l'utilisateur. Pour chaque document, les concepts ODP sont extraits des snippets afin de calculer la pertinence du document par rapport aux intérêts de l'utilisateur.

Ces approches de reclassement de résultats ont l'avantage de travailler sur un ensemble de document jugés pertinents pour l'ensemble des utilisateurs. La pertinence des résultats est améliorée en considérant les besoins individuels des utilisateurs.

c. Personnalisation de la représentation du besoin utilisateur

Outre l'ordre des résultats, certains systèmes de RI personnalisent la représentation du besoin de l'utilisateur. La représentation du besoin peut se faire au travers de l'analyse de

l'historique de recherche de l'utilisateur, de son contexte et/ou de ses activités courantes.

Exploitation de l'historique de recherche Les requêtes entrées sur un moteur de recherche sont une source d'information permettant d'identifier le besoin de l'utilisateur. L'avantage de ces informations est qu'elles sont récupérées de manière non invasive (aucune installation d'agent ou de système n'est nécessaire) et non intrusive (aucun accès non consenti aux informations). Les moteurs de recherche peuvent améliorer les résultats de recherche en se basant sur les requêtes précédentes [Lawrence, 2000]. De simples fichiers de *logs* ou des *cookies* peuvent être utilisés pour stocker les requêtes et les pages visitées par les utilisateurs [Pirolli & Pitkow, 1999].

Une base de données de requêtes associées à leurs résultats est utilisée par Raghavan and Sever [Raghavan & Sever, 1995]. Leurs recherches se concentrent sur des mesures de similarité utilisant plusieurs requêtes permettant de pallier le problème de représentation d'une requête prise individuellement. L'idée est de fournir les résultats des requêtes passées lorsqu'elles sont similaires à la requête courante.

Liu and Yu [Liu et al., 2004] utilisent une approche exploitant un profil construit à partir de requêtes de l'historique de recherche. Ces requêtes sont représentées par l'un des 3 premiers niveaux de hiérarchie de l'ODP. L'intérêt de cette approche est d'avoir un contexte de recherche représenté par le niveau de hiérarchie retenu et de désambiguïser les requêtes par la sélection des concepts de ce niveau.

Exploitation de l'activité de l'utilisateur Outre l'ordre des résultats, certains systèmes de RI personnalisent la représentation du besoin de l'utilisateur en prenant en compte le contexte et l'activité de l'utilisateur. Les informations sur le besoin de l'utilisateur sont déduites de l'observation des interactions de l'utilisateur avec le système (durée passée sur une page, texte tapé par l'utilisateur, pages visitées dans la liste de résultats de recherche, etc.). Rhodes [Rhodes & Maes, 2000] propose une approche appelée JITIR pour Just-in-TIME IR qui utilise ces techniques afin de récupérer des documents utiles à l'utilisateur. Il ne nécessite aucune action de recherche de la part de ce dernier. Le besoin est identifié via un système de veille sur les interactions entre l'utilisateur et le système. Ce besoin, représentant le profil de l'utilisateur, est exploité au travers du système d'alerte Google qui fournit des informations en relation avec le besoin implicitement identifié. Dans l'approche JITIR, trois agents sont employés afin d'identifier le besoin utilisateur :

- Le premier agent analyse ce que tape l'utilisateur ou ce qui est affiché à l'écran.
- Un agent ré-écrit les pages visitées en ajoutant des liens vers des documents connus de même thématique.
- Un dernier agent s'occupe du contexte de recherche comme la localisation, la date, etc.

Les informations recueillies par ces agents permettent d'expliciter les centres d'intérêt de l'utilisateur qui seront les sujets des alertes Google.

Une autre approche similaire à ce système, utilise également les ressources sur lesquelles l'utilisateur travaille (pages visitées, documents Word) afin de générer automatiquement des requêtes [Micarelli et al., 2007]. L'approche est également utilisée dans *Watson* [Budzik & Hammond, 2000] qui surveille les actions et les fichiers ouverts de l'utilisateur pour fournir des ressources connexes. L'agent est une application externe qui peut traquer l'utilisateur aussi bien sur le navigateur Internet Explorer que sur le navigateur Mozilla. Chaque

fenêtre ouverte donne lieu à un contexte et à une représentation du besoin. Des requêtes sont ensuite générées en reliant les différents contextes et besoins afin de fournir de l'information pertinente. Cette dernière provient de l'agrégation des résultats de la requête envoyée à plusieurs sources d'information dont ALTAVISTA et YAHOO !.

Une autre exploitation du profil utilisateur au sein d'un système de RI personnalisé en ligne est proposée par Koutrika et Ioannidis [Koutrika & Ioannidis, 2005]. Le profil est représenté par des termes connectés entre eux par des opérateurs logiques (conjonction, disjonction, négation, etc.) pour représenter le besoin de l'utilisateur. Cette représentation permet de construire des requêtes envoyées à des moteurs de recherche. L'évaluation de cette approche montre que la satisfaction du besoin de l'utilisateur est atteinte plus rapidement en utilisant ce système qu'en utilisant un moteur de recherche classique [Koutrika & Ioannidis, 2005].

Ces approches sont valorisées par le fait qu'elles sont non intrusives ; elle ne nécessitent pas l'utilisation de logiciels, ou de proxy.

2.1.5 Synthèse sur la RI

La RI classique a été introduite dans ce chapitre ainsi que ses limitations laissant la place à l'émergence de la RI personnalisée. Le tableau 2.2 propose une synthèse de l'ensemble non exhaustif des systèmes et des approches qui ont été décrits.

Il s'avère que la RI personnalisée peut s'employer dans diverses situations mais toujours en exploitant une forme de représentation des intérêts des utilisateurs afin de personnaliser la recherche et de répondre plus spécifiquement au besoin en information. Les modèles de recherche d'information personnalisée mis en oeuvre aujourd'hui sont viables. En terme de recherche sur le Web, les problématique d'accès et d'exploitation de très grands corpus documentaires sont résolus par les approches présentées dans ce chapitre. Malgré tout, du fait de l'immensité d'Internet, la recherche liée à un besoin spécifique revient à chercher une aiguille dans une botte de foin, et ce même avec des approches innovantes de RI personnalisée. Ce problème est particulièrement vrai pour les moteurs de recherche. De part leur capacité de collecte, ils ont une très bonne vision et représentation du Web. Cependant, cette représentation est une représentation altérée du Web du fait de la vitesse de création, modification, suppression des pages sur la toile. De plus, afin d'améliorer l'ordonnancement de leurs résultats, leur mesure d'appariement document/requête utilisateur est fortement basée sur la popularité des documents (cf. PageRank page 31). Or, ce dernier point est un problème pour les experts du renseignement qui, dans un cadre ROSO, peuvent être amenés à chercher des informations sur des sujets sensibles, non populaires, rendant l'utilisation des moteurs de recherche et des techniques de personnalisation de RI classiques inutiles.

Dans le chapitre suivant, nous présentons les robots d'indexation et les techniques d'exploration ciblée. Le but est de réduire le nombre de pages sur lequel travaillent les systèmes de RI pour focaliser la recherche sur des pages pertinentes en fonction du besoin utilisateur et sans considération de la popularité.

2.2 La Découverte d'Information

Dans cette partie, nous introduisons la notion de robot d'indexation en expliquant les principales étapes liées à la Découverte d'Information (DI) sur le Web. Nous proposons un

TABLE 2.2 – Synthèse des approches de RI personnalisée

Type de personnalisation	Données et méthodes utilisées	Références
Personnalisation de la présentation des résultats	Clusters de documents proposés à l'utilisateur pour personnaliser l'affichage des résultats	Yippy Search [8], Kartoo [9], [Zamir & Etzioni, 1999]
Personnalisation des résultats de recherche	Adaptation du score d'appariement des documents en fonction des concepts du profil utilisateur ou des termes de sa requête	[Lieberman et al., 1995]
	Exploitation de la structure hyper-liens. Amélioration du Page Rank et de HITS	[Mc Gowan, 2003, Chirita et al., 2004, Haveliwala, 2003]
	Nouvelle mesure de pertinence attribuée aux documents en fonction du profil utilisateur (utilisation de concepts, mots-clés, représentation ensembliste, etc.)	[Mc Gowan, 2003, Sieg et al., 2007]
	Exploitation de l'activité de l'utilisateur, des pages visitées et des documents consultés pour générer des requêtes	[Micarelli et al., 2007, Budzik & Hammond, 2000, Pirolli & Pitkow, 1999]
Personnalisation de la représentation du besoin utilisateur	Exploitation de l'historique de recherche pour faire du regroupement de requêtes et de leurs résultats. Désambiguïser les requêtes à l'aide des catégories de hiérarchies d'ontologies identifiées via l'historique de recherche.	[Micarelli & Sciarrone, 2004, Chirita et al., 2004, Lawrence, 2000]
	Exploitation de l'activité de l'utilisateur, des pages visitées et des documents consultés pour générer des requêtes	[Koutrika & Ioannidis, 2005, Rhodes & Maes, 2000]

état de l'art sur les différentes stratégies d'exploration du Web existantes tout en mettant en avant leurs limites. Nous nous intéressons aux robots d'indexation existants et plus particulièrement aux différentes approches d'exploration ciblée.

2.2.1 Exploration du Web

Dans cette partie, nous exposons les techniques et les stratégies d'exploration du Web. Un robot d'indexation, appelé aussi robot d'exploration, *crawler* ou *Web spider*, est un logiciel prenant en entrée un ensemble d'URLs. Une *Uniform Resource Locator* (URL) est un type particulier de *Uniform Resource Identifier* (URI). Une URI permet d'identifier une ressource sur un réseau. Une URL fournit, en plus de l'identifiant, les moyens d'obtenir une représentation de la ressource en décrivant son emplacement réseau. Le robot d'indexation collecte les documents (page HTML, image, fichier son ou video, etc) qui sont accessibles via les URLs fournies en entrée [Pinkerton, 1994, Yuwono et al., 1995]. Le système extrait de ces documents les liens pointant vers d'autres ressources. Cela permet ainsi de collecter de nouveaux documents et d'explorer le contenu du Web.

Le processus classique d'exploration d'un robot d'indexation est constitué de plusieurs étapes. Ces étapes correspondent à des tâches spécifiques exécutées séquentiellement. Dans un premier temps, le robot d'indexation récupère l'ensemble des URLs fournies en entrée du système. Ces URLs sont utilisées afin de construire la frontière d'exploration, c'est-à-dire la liste des URLs à visiter. Les premières URLs qui composent cette frontière sont appelées *les graines*. La frontière est dynamique durant le processus d'exploration puisqu'elle reçoit les nouvelles URLs extraites des documents collectés. La limite de la frontière peut correspondre au nombre maximal d'URLs à explorer ou à une distance, appelée profondeur d'exploration, entre les graines et les URLs découvertes. Cette couverture d'exploration (*scope*) peut également correspondre à un nombre maximal de documents à collecter ou à une durée de collecte prédéfinie. Lorsque celle-ci est définie, le robot d'indexation exécute, pour chacune des URLs de la frontière d'exploration, un ensemble de tâches illustré dans la figure 2.3 et explicité ci-dessous [Pant et al., 2004]. Si cette limite n'est pas fixée, le robot d'indexation peut explorer indéfiniment le Web et le collecter en entier.

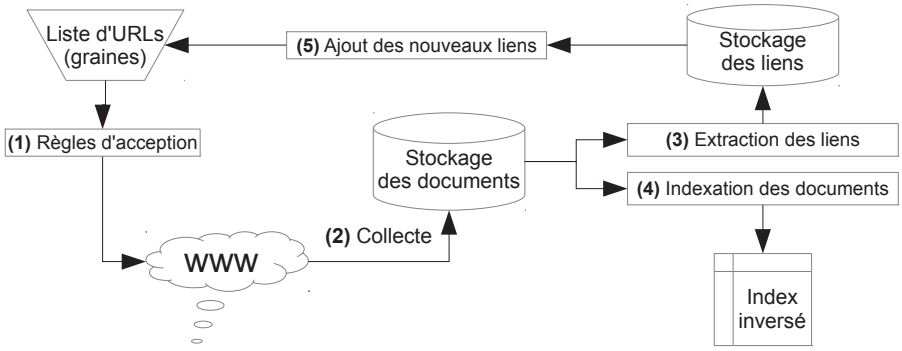


FIGURE 2.3 – Les étapes de l'exploration du Web par un robot d'indexation

1. Le système vérifie qu'aucune règle ou politique d'exclusion n'existe pour cette URL (*robot.txt* interdisant la collecte, nom de domaine défini dans une liste noire, etc.)

2. Le document pointé par l'URL est collecté et stocké
3. Les liens (URLs) contenus dans le document sont extraits
4. Selon des règles prédéfinies, le système décide d'indexer ou non le document collecté.
5. Les liens extraits sont ajoutés à la frontière d'exploration si les conditions d'arrêt ne sont pas atteintes.

Afin d'augmenter la vitesse d'exploration et d'éviter le gaspillage de ressources inutilisées, le robot d'indexation peut appliquer ces tâches à plusieurs URLs en même temps. La frontière du robot d'indexation peut contenir plusieurs files d'attente d'URLs à explorer : par exemple, une file d'URLs par hôte. Ces files d'attente peuvent être triées au sein de la frontière afin de définir une priorité d'exploration. De la même façon, les URLs d'une file d'attente peuvent être ordonnées par priorité. Pendant la phase de configuration, les règles de priorité doivent être définies afin que cet ordonnancement ait lieu.

2.2.2 Stratégies et types d'exploration

Les différentes stratégies et types d'exploration dépendent du but recherché. Il faut prendre en considération le besoin en terme de volume de documents désiré, de la couverture d'exploration recherchée ainsi que des intervalles de collecte souhaités. Autrement dit, le besoin est-il de collecter de préférence une large partie du Web, ou bien cherche-t-on à collecter des informations fraîches sur un sujet donné, ou bien encore de faire de la veille sur des sites spécifiques ? Dans tous les cas, il faut considérer la contrainte liée au dynamisme du Web dont le contenu change perpétuellement. Des milliards de documents sont disponibles sur Internet. Aussi, collecter l'ensemble du Web et maintenir la fraîcheur des documents collectés est très difficile.

Deux différentes stratégies d'exploration, une incrémentale et une immédiate, sont définies par Sigurosson [Sigurosson, 2005] : la stratégie dite immédiate ne considère les pages qu'une seule fois. Lorsqu'une URL connue est redécouverte, celle-ci est ignorée. Cette stratégie permet d'explorer une vaste partie du Web rapidement en se propageant uniquement via de nouvelles URLs. Elle peut également être utile pour explorer une partie spécifique du Web à un moment donné ou dans des intervalles de temps régulier. L'exploration incrémentale est employée pour une exploration continue et récurrente avec une couverture d'exploration limitée. Lorsqu'une URL déjà visitée est redécouverte, elle est ré-injectée dans une file d'attente de la frontière d'exploration. Les files d'attente ne sont ainsi jamais vides et l'exploration se poursuit indéfiniment. Cette pratique permet de surveiller tout changement d'une partie spécifique du Web. Ces stratégies d'exploration se basent sur le comportement de la frontière du robot.

Les robots d'indexation diffèrent non seulement sur la stratégie d'exploration qu'ils utilisent mais également sur le type d'exploration. Les types d'exploration les plus employés sont l'exploration en largeur [Najork & Wiener, 2001] et l'exploration ciblée [Sigurosson, 2005, Liu, 2007].

a. Exploration en largeur

Lors d'une exploration large ou *Breadth First Search*, le robot dispose d'une bande passante élevée afin de récupérer un très grand nombre de pages. L'exploration n'est pas limitée en profondeur. Le but est d'explorer une vaste partie du Web, si ce n'est le Web

en entier. L'algorithme consiste en une généralisation de l'exploration en largeur d'un arbre, c'est-à-dire niveau par niveau, comme illustré en figure 2.4. Cet algorithme est encore largement utilisé et ses résultats sont un repère permettant aux nouvelles approches d'évaluer leurs apports. L'algorithme utilise une file d'attente de manière non récursive comme explicité dans l'algorithme 1.

Algorithm 1 Pseudo-Code algorithme Breadth First Search

```
1: Créer F une file
2: g ← L'URL graine de départ
3: Enfiler g dans F
4: while F non vide do
5:    p ← Défiler F
6:    Visiter p
7:    for chaque pf page fils de p do
8:       if  pf non visité then
9:          Enfiler pf dans F
10:      end if
11:   end for
12: end while
```

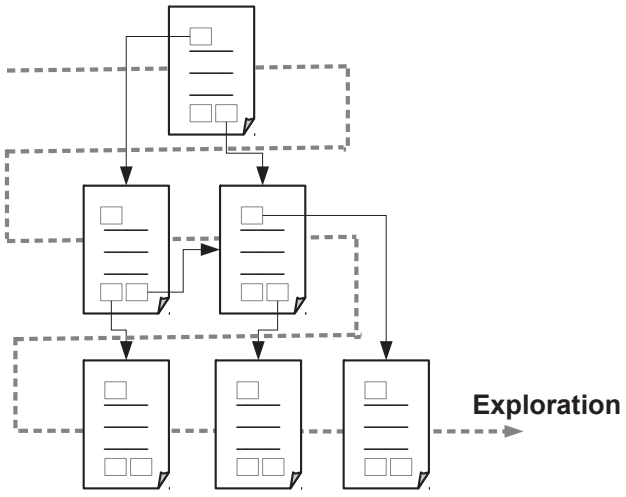


FIGURE 2.4 – Exemple d'exploration en largeur

b. Exploration ciblée

Le robot d'indexation ciblée explore le Web en considérant un certain nombre de critères afin de délimiter la zone de couverture. Les URLs à visiter sont filtrées. Ce type d'exploration est employé dans le but de collecter les pages d'un domaine, des pages sur un sujet donné ou qui répondent à un besoin spécifique. Nous présentons un certain nombre d'approches exploitant ce type d'exploration dans la section 2.2.4, page 29. Un algorithme ciblé, connu sous le nom de *Best First search* consiste à prendre en compte une connaissance supplémentaire permettant de repérer les pages susceptibles d'être pertinentes. Ainsi

l'ordre des pages explorées n'est plus aléatoire ou linéaire, il repose sur la sélection du lien menant vers la page jugée comme étant la plus pertinente.

2.2.3 Caractéristiques des robots d'indexation

Un robot d'indexation est un module utilisé en amont des moteurs de recherche. Son efficacité à collecter rapidement des pages est très importante afin que le moteur de recherche puisse fournir des résultats de recherche avec une vision la plus complète possible du Web et des informations fraîches. Cette efficacité est étroitement liée à la conception et à la réalisation du robot d'indexation, ainsi qu'au réseau et aux types de connexion Web [Najork & Heydon, 2002, Shkapenyuk & Suel, 2002, Diligenti et al., 2004]. Ainsi, quelque soit la stratégie et le type d'exploration que les robots emploient, ils sont soumis à certains points techniques et contraintes d'exploration :

La performance - Un robot doit pouvoir explorer des milliers de pages dans un temps le plus court possible. Cet aspect est fortement lié au matériel utilisé. La bande passante doit être élevée afin d'assurer une collecte rapide des documents, la puissance du CPU doit permettre de traiter ces documents efficacement et le disque dur doit pouvoir stocker un important nombre de données. La configuration du robot peut permettre d'améliorer ces performances. Les tâches de collecte peuvent être distribuées sur plusieurs machines. L'exploration ciblée permet également d'éviter des problèmes de collecte d'un trop grand volume de pages. Pour l'exploration large et la stratégie d'exploration incrémentale, une configuration permettant d'éviter la collecte de pages inchangées peut être employée.

Zone de couverture - Une limite lors de l'exploration doit être fixée afin d'éviter une exploration infinie. Si la zone de couverture n'est pas délimitée, les files d'attente d'URLs à explorer vont croître indéfiniment et empêcheront ainsi le robot d'explorer une page déjà collectée dans un délai raisonnable. La fraîcheur des données risque alors d'en pâtir.

Politique de politesse - Chaque site Web peut définir un ensemble de règles constituant sa politique de politesse. Cela permet de limiter la visite de certains robots d'indexation jugés comme étant trop agressifs ou encore d'empêcher toute visite de robots afin de ne pas être indexé. Les robots d'exploration respectent majoritairement ces règles.

Un robot d'exploration doit également faire face à des cas particuliers du Web comme certains pièges. Un piège pour un robot d'indexation est décrit par Bing Liu [Liu, 2007]. Ce sont des sites où des URLs qui sont créées dynamiquement et modifiées en fonction des actions effectuées par le robot d'indexation. Celui-ci est alors guidé sur des pages sans contenu, indéfiniment. Pour sortir de ces pièges, les robots d'exploration sont souvent configurés pour changer de domaine lorsqu'un important volume de documents a été collecté depuis un même domaine.

a. Robots d'indexation existants

Les approches dans le domaine de la découverte de sources et de l'exploration du Web utilisent généralement des systèmes employant des robots d'indexation existants. Il existe un certain nombre de robots d'indexation tel que Heritrix[10] [Mohr et al., 2004],

10. https ://webarchive.jira.com/wiki/display/Heritrix

Nutch [11] [Khare et al., 2004], JSpider [12], YaCy [13], et Crawler4j [14]. Ces robots d'indexation se distinguent par leur licence, leur langage de programmation ou leur architecture. Dans l'annexe C.1.3, page 201, nous proposons une vue d'ensemble des robots existants et en particulier de ceux qui sont *open source* et dont les fonctionnalités peuvent être étendues.

b. Limites des robots d'indexation "classiques"

La limite des robots d'indexation a déjà été soulevée dans la partie précédente. Les systèmes de RI classique traitent les requêtes utilisateurs en faisant des recherches par similarité dans leur index de documents. Ces documents ont été collectés par un robot d'indexation sans prise en compte du besoin de l'utilisateur. Nous avons déjà évoqué dans le chapitre 2, page 13, une solution apportée par les systèmes de RI personnalisés. La représentation du besoin utilisateur est alors utilisée afin de reformuler la requête de l'utilisateur ou afin de reclasser les résultats de la recherche. Cependant, ces approches travaillent sur un index constitué de millions de documents. Une autre approche est de cibler l'exploration et la collecte des documents afin de réduire la taille de l'index et ainsi de travailler sur un ensemble réduit de documents.

2.2.4 Exploration ciblée du Web

Un robot d'indexation classique convertit un document Web en plein texte afin d'extraire les liens sortants. Le but des robots d'indexation ciblée est de prévoir l'intérêt d'une page avant de la collecter. Ils reposent sur l'analyse des informations hypertextes mais également sur le *topical locality phenomenon* (TLP [Davison, 2000]) qui est un principe selon lequel deux pages Web, connectées l'une à l'autre par un hyperlien, partagent généralement un contenu textuel commun.

a. Informations hypertextes

Les pages Web sont définies comme étant des ressources hypertextes lorsqu'elles sont connectées à d'autres pages par l'intermédiaire de liens. Les systèmes de RI s'intéressent au contenu des documents sans considérer la structure du Web et les liens qui ont permis de les obtenir. Ces informations, appelées hyper-informations [Marchiori, 1997] ne sont pas ignorées par un robot d'indexation ciblée. Les informations extraites de la structure des liens peuvent être utilisées afin de trouver des pages d'intérêt [Brin & Page, 1998]. L'exploration ciblée est basée sur l'exploitation de la structure du Web, comme le TLP et les deux algorithmes hypertextes, HITS [Kleinberg, 1999] et PageRank [Page et al., 1999], mais aussi le contenu des documents avec l'analyse du texte d'un lien et du texte entourant celui-ci.

b. TLP et texte de liens

La grande majorité des robots d'indexation s'appuie sur le TLP introduit par Davison [Davison, 2000]. Une page partage plus d'intérêts communs avec une page à laquelle elle

11. http://nutch.apache.org/
12. http://j-spider.sourceforge.net/
13. http://yacy.net/fr/
14. http://code.google.com/p/crawler4j/

est reliée, qu'avec une page prise au hasard. L'évaluation empirique de Davison montre que ce phénomène est très largement répandu sur le Web [Davison, 2000]. En plus de mettre en avant la relation thématique entre deux pages liées, Davison montre que plus deux liens dans une page sont proches, plus les pages pointées par ces liens partagent un contenu textuel commun (voir figure 2.5). Menczer a étendu l'étude de ce principe en mettant en avant qu'un ensemble de pages connectées les unes aux autres permettent d'identifier une thématique. On parle alors de *cluster* ou regroupement thématique [Menczer, 2004]. Les robots d'indexation ciblée exploitent ce principe afin d'explorer un ensemble de pages connectées à une page considérée comme d'intérêt. La profondeur de l'exploration est limitée par les pages non pertinentes qui sont découvertes.

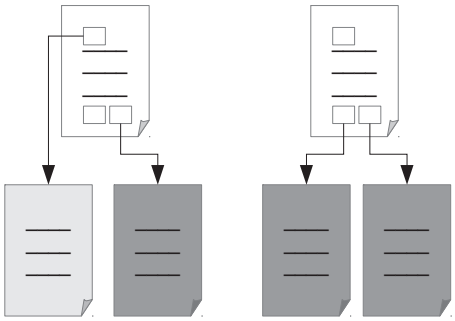


FIGURE 2.5 – Deux liens proches pointent généralement vers des pages de même intérêt

Davidson montre que le texte d'un lien est souvent une source d'information permettant d'avoir un aperçu du contenu du document pointé. Le titre et la description d'un document sont également une source d'information intéressante selon Davidson, contrairement au texte entourant un lien qui n'apporte pas d'information supplémentaire sur le contenu du document. Ces informations ont été utilisées dans de nombreuses tâches, comme la modélisation du besoin de l'utilisateur [Gasparetti & Micarelli, 2005, Gauch et al., 2007], le résumé de documents [Delort et al., 2003], la traduction de requêtes [Lu et al., 2002] mais aussi pour ordonner les URLs à collecter dans un robot d'exploration ciblée [Patel & Schmidt, 2011].

Certains moteurs de recherche associent le texte du lien avec le document pointé dans leur index [Brin & Page, 1998]. L'intérêt est d'avoir une description complète de la page au travers du texte de chaque lien pointant vers le document. L'exploitation du texte des liens peut également être combinée avec des approches exploitants la structure du Web, comme le PageRank présenté ci-dessous, afin de définir les priorités de collecte du robot d'exploration [Uemura et al., 2012].

c. HITS et PageRank

Ces deux algorithmes ont pour but de fournir une mesure de pertinence en prenant en compte uniquement la structure des liens du Web, ignorant le contenu textuel des pages. Ils ont notamment été utilisés dans la RI personnalisée (voir section 2.1.4, page 18). Certains robots d'indexation ciblée s'en servent dans leur algorithme afin d'assigner un rang aux ressources à explorer et pour trouver de nouvelles pages pertinentes plus rapidement [Cho et al., 1998, Chakrabarti et al., 1999b, Rungsawang & Angkawattanawit, 2005].

En règle générale, lorsqu'une volumineuse collection de documents doit être analysée afin de trouver les documents répondant à une requête donnée, le processus de classement est une étape cruciale. Une étude montre que parmi les utilisateurs d'un moteur de recherche, 28,6% ne regardent que la première page de résultats, 19% en regardent 2, et ainsi la moitié des utilisateurs ne regardent pas plus de 20 documents [Spink et al., 2001]. La pertinence d'un document et son classement sont aussi importants dans la RI que dans la DI. Dans ce dernier, le but est d'optimiser le processus de découverte en explorant les documents les plus pertinents en premier. Ces deux algorithmes sont relativement coûteux au point qu'ils ne peuvent pas être exécutés pour chaque requête. Aussi, il convient de planifier le traitement de ces mesures hors ligne. Cependant, les robots d'indexation ciblée adaptent leur exploration afin d'indexer moins de documents. L'utilisation en ligne de ces algorithmes, pendant la phase d'exploration, est alors très appréciée.

L'algorithme HITS La première étape de l'algorithme est de construire, à partir de l'ensemble des pages, un graphe orienté d'hyperliens G = (S,A) où S est un ensemble de noeuds et A un ensemble d'arcs. Un noeud représente une page et un arc correspond à un lien entre deux pages. L'algorithme prend en compte le nombre de *backlinks*[15] afin d'assigner un score de pertinence aux pages. Kleinberg identifie deux catégories de pages [Kleinberg, 1999]. Les pages *authorities* qui ont un contenu pertinent à propos d'un sujet donné, et les pages *hubs*, qui contiennent des pages vers la première catégorie de pages.

Un algorithme récursif permet de calculer le score d'une page pour ces deux catégories. L'algorithme considère l'ensemble de pages de départ fourni par un moteur de recherche en réponse à une requête donnée. Il récupère tous les liens entrants et sortants de ces pages au moyen d'un moteur de recherche qui permet de prendre en compte la structure des liens. Les pages obtenues sont le point de départ de l'algorithme HITS qui va calculer les deux scores de chaque page suivant ces deux mesures :

$$a_p = \sum_{q=i}^{n} h_q$$

$$h_p = \sum_{q=i}^{m} a_q$$

Où a et h sont respectivement des pages *authorities* et des pages *hubs*. Dans le premier calcul, n est le nombre de pages pointant vers p et i est une de ces pages. Dans le second, m correspond au nombre de pages pointées par p et i est une de ces pages. La figure 2.6 illustre le principe de ces deux mesures, qui après un certain nombre d'itérations permettent d'obtenir les deux scores d'une page. L'algorithme, dont le détail est décrit par Kleinberg, fournit en sortie une liste pour chaque catégorie avec les pages qui ont eu le meilleur score [Kleinberg, 1999]. La liste de pages *hub* est intéressante pour les robots d'indexation qui disposent ainsi d'une liste ordonnée de pages pointant vers des pages d'intérêt.

L'algorithme PageRank Alors que l'algorithme HITS se base sur deux mesures pour assigner un poids de pertinence aux pages, l'algorithme de Page et Brin ne repose que sur une mesure appelée PageRank (PR) [Page et al., 1999]. Le PR associe la pertinence d'une

15. backlinks de la page P : liens pointant vers P

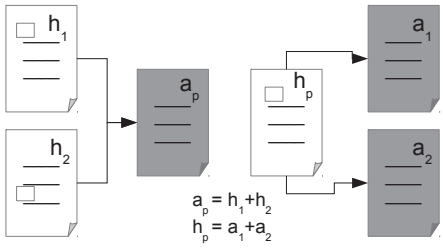


FIGURE 2.6 – Exemple de mesure *authority* et *hub* pour une page p

page au nombre de pages pointant vers elle. Autrement dit, le PR est une amélioration du calcul de *backlinks* d'une page. L'avantage du PR, par rapport à HITS, est qu'il ne nécessite pas un ensemble de pages de départ pour construire un sous-graphe du Web pertinent pour une requête donnée. Pour appliquer cet algorithme, le Web doit être considéré comme un graphe fortement connexe et le PR comme un algorithme basé sur le modèle du *Random surfer*. Le principe du *Random surfer* est de cliquer aléatoirement sur un des liens présent dans une page p avec une equiprobabilité de $1/N_p$, où N_p est le nombre de liens dans la page p. La formule suivante représente le calcul du PR :

$$PR(p) = c\sum_{q=i}^{n}\frac{PR(q)}{N_q}$$

Où c est une constante inférieur à 1, n est le nombre de pages pointant vers p et i est une de ces pages. Le PR procède à ce calcul d'une page à une autre. Tout comme l'algorithme HITS, cette mesure est récursive : le calcul est fait jusqu'à obtenir une mesure de convergence, symbolisant la probabilité qu'un internaute visualise une page. Cependant, le Web n'étant pas fortement connexe, certaines situations peuvent fausser la mesure. C'est le cas lorsque deux pages ne sont connectées qu'entre elles. Dans cet exemple, les deux pages vont augmenter leur PR indéfiniment. Aussi, la formule a été améliorée comme suit :

$$PR(p) = c\sum_{q=i}^{n}\frac{PR(q)}{N_q} + \frac{(1-c)}{N}$$

Le nouveau terme représente la probabilité qu'un utilisateur arrête de cliquer sur le lien d'une page et décide de commencer son exploration à partir d'une nouvelle page. Tout comme HITS, le PR d'une page peut permettre d'améliorer les robots d'indexation en explorant les liens sortants des pages dont le PR est élevé.

d. Approches d'exploration ciblée

L'exploration ciblée du Web peut se faire de différentes manières. Dans cette section, les principales approches existantes sont introduites.

L'approche Fish Search L'algorithme *Fish search* introduit le principe clé qui a permis d'améliorer les robots d'indexation [De Bra & Post, 1994, De Bra et al., 1994]. Il prend en entrée une ou plusieurs graines (URLs), ainsi que la requête de l'utilisateur. Les graines peuvent être soit les premiers résultats d'un moteur de recherche répondant à la requête de

l'utilisateur, soit les pages Web en favoris. La liste des graines est donc une liste ordonnée. A chaque itération, la première graine de la liste est collectée. Son contenu est analysé et comparé aux mots de la requête de l'utilisateur. Des heuristiques décident ou non de poursuivre l'exploration à partir de cette page. Elles sont basées sur un certain nombre d'expériences liées à la recherche d'une stratégie de navigation optimale. Cette stratégie est basée sur temps ou sur le nombre de noeuds qu'il est possible de visiter, et sur les facilités de navigation au sein de la structure hypertexte. Lorsqu'une page pertinente est trouvée, les pages filles sont annotées comme *pages provenant d'une page pertinente* et ajoutées en début de liste des pages à collecter. Sinon, les pages filles sont ajoutées après les pages filles provenant d'une page pertinente. Notons que, malgré l'utilisation d'une requête utilisateur, l'approche est bien liée à la DI et non à la RI car si chaque utilisateur utilisait cette approche pour leurs recherches, les serveurs Web imploseraient dû au travail nécessaire pour cette tâche. Une approche similaire est proposée avec l'algorithme *Shark search* [Hersovici et al., 1998] qui propose une mesure de similarité entre une page et la requête utilisateur plus précise que [De Bra & Post, 1994, De Bra et al., 1994].

L'approche utilisant le Web sémantique Cette méthode d'exploration ciblée repose sur l'exploitation d'ontologies afin d'identifier les graines pertinentes à explorer [Ehrig & Maedche, 2003]. Contrairement à l'approche de l'algorithme *Fish search*, le contenu des pages n'est pas comparé aux mots constituant la requête de l'utilisateur mais aux concepts d'une ontologie fournie par l'utilisateur. Une ontologie permet de décrire un domaine ou un sujet de manière plus exhaustif qu'une requête utilisateur. Ainsi, la comparaison des résultats entre l'approche standard d'exploration ciblée et celle utilisant le Web sémantique montre que ce dernier améliore de façon globale la pertinence des pages découvertes [Ehrig & Maedche, 2003].

L'approche par distillation L'approche proposée par Chakrabati *et al.* fait partie des approches les plus populaires dans le domaine des robots d'indexation ciblée [Chakrabarti et al., 1999a, Chakrabarti et al., 1999b]. Elle est composée de deux processus :

- Un classifieur qui évalue la pertinence des pages en fonction du sujet d'intérêt courant.
- Un distillateur qui utilise l'algorithme HITS afin de repérer les pages considérées comme des points d'accès pertinents (*hubs*).

Avant la phase d'exploration, une phase d'interaction entre l'utilisateur et le système est nécessaire :

- Grâce au classifieur basé sur une taxonomie donnée, comme *Yahoo ! Directory* ou l'ODP, un arbre de catégories est proposé à l'utilisateur en fonction des pages modèles qu'il a fourni au système. L'utilisateur peut alors affiner les catégories en sélectionnant celles correspondant le mieux aux pages modèles. Le système propose également les URLs de pages pointées par les pages modèles. L'utilisateur peut les parcourir et les ajouter aux pages modèles s'il les juge pertinentes.
- Le classifieur intègre les modifications et les choix de l'utilisateur au sein d'un modèle de classe statistique.
- La phase de collecte commence en se basant sur le modèle de classe statistique, et sur le distillateur, qui est lancé par intermittence ou simultanément, afin d'ordonner les pages à explorer par pertinence.

– L'utilisateur peut marquer les pages découvertes comme utiles ou non. Ces informations sont remontées et prises en compte par le distillateur et le classifieur.

L'approche par tunneling Comme évoqué dans la partie 2.2.4, page 29, le TLP part du principe qu'une page sur un sujet pointe généralement vers une autre page de même sujet. Bien évidemment, certaines pages de même sujet ne sont pas obligatoirement liées l'une à l'autre directement. Il est parfois nécessaire de traverser un ensemble de pages hors sujet avant de retrouver une page d'intérêt. Se basant sur ce constat, [Bergmark et al., 2002] suggèrent de laisser le robot d'indexation parcourir un certain nombre de liens qu'un robot d'indexation ciblée aurait exclus. Cette technique est appelée *tunneling*.

Les auteurs analysent, dans leur approche, 500 000 pages qu'ils classent en page d'intérêt ou non. Le but est de trouver un modèle basé sur les chemins hyperliens permettant de définir le nombre raisonnable de pages hors sujet qu'un robot d'indexation peut explorer afin de trouver des pages d'intérêt. Ce nombre permet alors de définir une profondeur d'exploration adaptative.

Les résultats de cette approche montrent qu'un robot d'indexation ciblée est plus efficace pour explorer rapidement le Web avec une profondeur d'exploration adaptative qu'avec une profondeur d'exploration fixe. De plus, cela diminue le nombre de documents collectés. La technique du *tunneling* permet donc d'améliorer l'exploration ciblée grâce à la prise en compte de chemins d'intérêt contenant des pages hors sujet.

L'approche par exploration contextuelle Donner un score de pertinence à une page Web en se concentrant sur la structure des liens et des chemins d'accès aux documents est une problématique à laquelle Diligenti *et al.* se sont également intéressés [Diligenti et al., 2000]. Leur approche se base sur la construction d'un *context* graphe. Il est construit à partir de pages fournies par l'utilisateur. Ces pages constituent la première couche du graphe. Des moteurs de recherche sont utilisés afin de retrouver les pages liées aux pages de l'utilisateur. Ces nouvelles pages composent la seconde couche du graphe. Ce processus de recherche de *backlinks* est répété sur les pages des nouvelles couches autant de fois que l'a spécifié l'utilisateur. Ce graphe par couche permet de garder la distance entre les pages trouvées et les pages de départ et ainsi le nombre minimum de liens à suivre pour naviguer entre ces pages. Ces distances servent d'entrée à un classifieur de type bayésien naïf permettant d'apprendre les différentes catégories. Le but étant de retrouver, pour une page donnée, la catégorie à laquelle elle appartient et ainsi d'identifier sa distance avec la page de départ. La découverte de sources s'effectue alors sur les pages dont la distance probabiliste est la plus courte. L'inconvénient de cette approche est l'utilisation de moteurs de recherche qui doivent être équipés de la fonction de recherche des *backllinks*.

e. Modèles multi-agents

Le modèle de vie artificielle Le modèle *endogenous fitness* propose l'utilisation d'une population d'agents calquée sur des modèles de vie artificielle. Les agents sont autonomes et en concurrence. Chaque agent a un niveau d'énergie qui varie : il baisse à chaque action et augmente lors de la collecte de ressources. Le but d'un agent est donc de survivre en découvrant des ressources. La vie d'un agent est caractérisée par deux seuils d'énergie : un seuil de mort, un seuil de reproduction. L'intérêt est de découvrir les zones riches

en ressources identifiables grâce aux agents qui vont instinctivement s'y concentrer pour survivre et s'y reproduire.

Ce principe a été adapté à la DI sur le Web. Menczer *et al.* transposent le modèle au fonctionnement d'un robot d'indexation de la manière suivante [Menczer et al., 1995] :

- Le robot d'indexation est constitué d'agents.
- Les agents consomment leur énergie en parcourant le Web de lien en lien.
- Les agents gagnent de l'énergie selon la pertinence des pages qu'ils parcourent.

Les agents cherchant à survivre, et donc à limiter leurs actions, le réseau n'est pas saturé. La population d'agents s'auto-équilibre entre les morts et les reproductions. Les zones du Web les plus pertinentes, considérées comme des zones à énergie, sont identifiées par l'importante présence d'agents.

Ce modèle est employé dans le projet InfoSpiders, aussi connu sous le nom de ARACHNID pour *Adaptative Retrieval Agents Choosing Heuristic Neighborhoods for Information Discovery* [Menczer & Belew, 2000]. Les agents sont capables d'évaluer la pertinence d'une page en fonction d'une requête donnée et de raisonner de manière autonome quant au choix de la prochaine page à visiter. Deux fonctions *benefit()* et *cost()* mettent à jour l'énergie de l'agent. La première détermine le gain d'énergie aux vues des termes connus présents dans la nouvelle page visitée. La seconde consomme de l'énergie selon la taille de la page et de son temps de réponse.

My Spider implémente cette approche dans une application Java en ligne [16] [Pant & Menczer, 2002]. Cette implémentation propose de suivre en temps réel le fonctionnement du système grâce à une visualisation de la répartition des agents de collecte. [Chen et al., 1998] propose une approche similaire jumelée avec une approche d'exploration de type *Best-first*. Cependant, ce système, appelé *Itsy Bitsy spider*, n'a pas une fonctionnalité importante, présente dans InfoSpiders, qui est l'autonomie des agents et leur capacité à déterminer et à s'adapter à la DI. Balavonic combine ce modèle avec une approche collaborative détaillée dans [Balabanovic & Shoham, 1997, Balabanovic, 1998].

Le modèle éthologique À l'instar du modèle précèdent, d'autres approches se sont basées sur un raisonnement calqué sur un modèle de vie existant. Gasparetti *et al.* [Gasparetti & Micarelli, 2003, Gasparetti & Micarelli, 2004] proposent un système de DI évolutif et adaptatif dont l'architecture multi-agent est construite à partir de l'observation du modèle de comportement et d'intelligence collective des insectes sociaux comme les fourmis [Bonabeau et al., 2000]. La recherche menée par les biologistes et les éthologues a été de comprendre comment des insectes aveugles, comme les fourmis, arrivent à apporter la nourriture jusqu'à leur nid en prenant le chemin le plus court possible. Ceci s'explique par la phéromone qu'elles libèrent afin de marquer leur chemin. Les autres fourmis peuvent alors suivre une trace, renforçant la quantité de phéromones présente avec la leur. Les premières fourmis qui retournent à leur nid avec de la nourriture auront marqué leur chemin sur le trajet de retour. Les prochaines fourmis à sortir du nid retrouveront ce chemin le plus court étant attiré par la forte présence de phéromones.

Transposé à la DI, un tel modèle fournit au système un ensemble d'agents capable de naviguer sur le Web. Ces agents, grâce à un lot de comportements simples, sont capables de choisir les chemins les plus prometteurs qui peuvent mener à des ressources intéressantes pour l'utilisateur. La phéromone laissée par les agents, correspondant au résultat

16. http://myspiders.informatics.indiana.edu/

de la prospection d'un agent, peut être exploitée par les autres agents afin d'améliorer les prochaines explorations. Cet échange permet d'assurer l'adaptabilité des agents nécessaire à leur travail dans un environnement complexe et dynamique comme le Web.

Le système, en plus de considérer le TLP, ne néglige pas les pages qui ont permis d'accéder à une page intéressante. Le chemin d'accès à une page d'intérêt, comportant des pages non pertinentes, est une unité d'information permettant de satisfaire l'utilisateur [Joachims et al., 1997]. Cette démarche rejoint le concept de *tunneling* vu précédemment [Bergmark et al., 2002]. Lorsqu'un agent atteint une page, il compare son contenu avec le besoin de l'utilisateur. Ainsi, plus un chemin comporte des pages potentiellement intéressantes non explorées, plus il y aura de phéromones sur ce chemin. Si deux chemins mènent à une même page, l'agent ayant pris le chemin le plus court sera également celui qui libérera en premier la phéromone et qui sera de retour en premier avec les informations sur les ressources explorées. Le chemin le plus court est ainsi obligatoirement utilisé par les autres agents qui souhaiteraient atteindre cette ressource.

Le processus d'exploration du système se fait par cycle. Pour chacun de ces cycles, un nombre maximal de mouvements de lien en lien pour un agent est autorisé. Ce nombre est proportionnel au nombre de cycles tel que :

$$maxMouvement = k * nbCycle$$

Où k est une constante et $nbCycle$ est le nombre de cycles effectués. Durant le premier cycle, les chemins ne sont pas marqués de phéromones car les premières explorations sont faites au hasard. Après un certain nombre de cycles, les chemins intéressants vont rapidement être révélés. Un agent adapte son comportement selon les cas suivants :

- avant chaque cycle, l'agent retourne à la ressource de départ en mettant à jour le taux de phéromones sur le chemin de retour,
- si un chemin est déjà *marqué*, l'agent choisit de le suivre selon une probabilité calculée à partir du taux de phéromones sur ce chemin,
- si aucune information n'est encore disponible, le choix du chemin est fait au hasard.

Les calculs, qui définissent la probabilité qu'un agent a de suivre un chemin, prennent en compte le dynamisme de la population ainsi que les cycles d'exploration passés, ce qui permet d'entretenir les chemins vers les ressources. Les chemins d'intérêt se renforcent, persistent ou meurent de façon progressive. Les pages pertinentes sont ainsi rapidement atteintes malgré une marche aléatoire lors du premier cycle. Le parallèle avec le fonctionnement de vie des fourmis est intéressant car il offre une robustesse et une adaptabilité nécessaire à l'exploration du Web du fait de sa constante évolution.

f. Modèles par apprentissage

Les modèles d'approches adaptatives comme les modèles multi-agents vus précédemment sont conçus afin de reconnaître les pages d'intérêt et de conduire l'exploration du robot d'indexation. Dans cette partie, d'autres modèles permettant cette adaptabilité sont présentés. Ils sont basés sur des techniques d'apprentissage automatique.

Un modèle statistique intelligent Le système *Intelligent Crawling*, proposé par [Aggarwal et al., 2001], a pour but d'apprendre un modèle statistique à partir des caractéristiques du Web, et en particulier de la structure de liens entre les pages. L'approche

se base sur un ensemble de prédicats qui sont des mots-clés contenus dans les pages ou dans le texte des liens. Le but est d'apprendre la relation entre les prédicats présents dans une page parent et la probabilité qu'une page fille satisfasse le besoin en information. Au début du processus de collecte, les prédicats ont la même importance, entraînant une exploration aléatoire. Lorsque les premières pages sont collectées, les prédicats de ces pages sont construits et analysés afin d'établir une corrélation avec le besoin en information. Par exemple :

- Si 5% de ces pages analysées contiennent le prédicat d'intérêt P
- Et si 10% des pages qui ont permis de les collecter contiennent ce même prédicat
- Alors le système calcule un ratio pour ce prédicat de la manière suivante : 0.1/0.05 = 2

Le but du système est d'apprendre de l'analyse des pages en calculant des ratios sur l'ensemble des prédicats. Ces ratios permettent d'identifier les meilleurs caractéristiques à prendre en compte pour trouver des pages d'intérêt. Dans notre exemple, considérant l'ensemble des ratios calculés, le système doit-il se focaliser sur les liens des pages qui contiennent le prédicat P ? Le robot d'indexation utilise ces ratios pour décider quelle page candidate sera à explorer afin de satisfaire le besoin de l'utilisateur.

L'intérêt d'un tel système est son autonomie et sa capacité à évoluer. Il ne nécessite pas un apprentissage sur un corpus de documents avant la phase de collecte. Les prédicats sont des mots-clés, définis par les utilisateurs, qui doivent être présents dans le contenu ou le titre des pages par exemple. Le robot d'indexation adapte son comportement en apprenant des corrélations entre les prédicats et les pages d'intérêt. La difficulté réside dans le choix, par l'utilisateur, de prédicats judicieux.

Un modèle d'apprentissage par renforcement Chakrabarti *et al.* exploitent également le contenu des pages parents collectées pour évaluer les pages filles à explorer [Chakrabarti et al., 2002]. Le système proposé repose sur le fonctionnement classique d'un robot d'indexation ciblée [Chakrabarti et al., 1999b]. Un classifieur évalue la pertinence des pages collectées en fonction d'un ensemble de sujets : ils sont définis en amont de l'exploration, via l'explicitation par l'utilisateur des noeuds reflétant son besoin dans l'arborescence de catégories fournie par le classifieur. L'amélioration proposée est l'ajout d'un classifieur permettant d'ordonner par priorité les pages candidates. Pour ce faire, le système extrait des caractéristiques de la représentation des pages parents qui est faite sous forme de DOM [17]. Ce second classifieur utilise le premier afin d'apprendre les caractéristiques permettant de déduire l'importance d'une page à explorer. L'apprentissage par renforcement a pour but de décider des actions que le système doit entreprendre afin de maximiser une notion de récompense sur le long terme. Cette notion de récompense est, comme le système proposé par Rennie *et al.*, associée à la découverte d'une page d'intérêt et les actions à l'exploration d'un lien [Rennie & McCallum, 1999]. Le modèle d'apprentissage est basé sur une classification naïve bayésienne qui permet de déterminer la pertinence d'un lien en se basant sur les mots de ce lien ou le texte environnant.

17. DOM : le Modèle Objet de Document permet la représentation de documents HTML valides et XML bien-formés

g. Évaluation

L'évaluation d'un robot d'indexation classique repose sur des aspects techniques déjà évoqués dans la première partie de ce chapitre. La capacité de stockage, la bande passante, l'architecture distribuée, le système d'exploitation sont des points techniques importants pour une exploration efficace [Srinivasan et al., 2005]. Cependant, dans le cas d'un robot d'indexation ciblée, l'évaluation va porter sur la pertinence des pages collectées répondant au besoin de l'utilisateur. Le système est jugé sur sa capacité à filtrer les pages d'intérêt et non pas sur son potentiel à collecter rapidement le Web entier. Afin d'évaluer la capacité du robot à collecter uniquement des pages pertinentes, le calcul de précision P est fait tel que :

$$P = \frac{NbPagesPertinentesC}{NbPagesC}$$

Où la pertinence repose sur le nombre de pages collectées répondant au besoin en information de l'utilisateur et sur le nombre total de pages collectées. Une autre mesure, appelée rappel (*recall*) R, permet de connaître la proportion de pages pertinentes collectées par rapport au nombre de pages pertinentes disponibles dans l'environnement d'exploration :

$$R = \frac{NbPagesPertinentesC}{NbPagesPertinentesDisponibles}$$

La figure 2.7 met en évidence ces deux mesures qui peuvent être représentées sous la forme d'intersections.

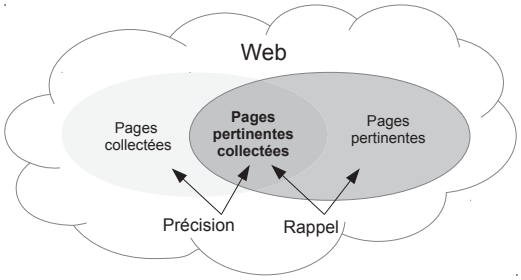


FIGURE 2.7 – Représentaton de la précision et du rappel

En RI, des corpus de documents peuvent être employés pour évaluer un système. Les documents pertinents sont connus, et les techniques utilisées doivent permettre de les retourner à l'utilisateur. Dans le cadre de la DI, le corpus n'est autre que le Web. Aussi, l'ensemble des pages pertinentes n'est pas connu, le but étant d'ailleurs de les découvrir. Il n'est donc pas possible de mesurer le rappel [Pant et al., 2004]. Travailler sur une sous partie du Web déjà identifiée pour constituer un corpus ne permet pas non plus d'obtenir une évaluation satisfaisante. Dans ce dernier cas, on considère une vue biaisée du Web. Il est donc difficile de comparer les résultats des approches présentées ci-avant. Cependant, afin de mesurer la pertinence des systèmes de DI, la mesure de précision a été modifiée et adaptées au contexte du Web.

Adaptation de la mesure de précision L'adaptation de la mesure de précision permet d'évaluer la pertinence d'un système de DI sans la mesure de rappel. Deux principales adaptations sont utilisées :

- Mesure du taux d'acquisition. Dans un système où les pages pertinentes et les non pertinentes sont différentiables de façon booléenne, le taux d'acquisition peut être mesuré. Si 200 pages sont pertinentes sur les 2000 premières collectées, alors le taux de récolte est de 10% [Aggarwal et al., 2001].
- Mesure de la pertinence moyenne. Si le score de pertinence n'est pas booléen mais continue, la moyenne des scores de pertinence peut être utilisée pour évaluer le système [Menczer et al., 2001, Chakrabarti et al., 1999b]. La moyenne peut être calculée tout au long de la collecte [Menczer et al., 2001]. Par exemple sur les 100 premières pages collectées, puis les 100 suivantes, et ainsi de suite. Elle peut aussi être calculée sur un intervalle donné, comme par exemple, sur les 50 dernières pages à l'instant t [Chakrabarti et al., 1999b].

Comparaison de système de DI D'une façon générale, les approches se différencient par rapport aux algorithmes utilisés et à leurs complexités, aux graines de départ employées (aléatoire ou URLs de pages d'intérêt), à leur adaptabilité (apprentissage ou non). De plus, les résultats sont fortement dépendants de l'état du Web au moment de la collecte. Autant de critères qui rendent la comparaison des approches et de leurs résultats difficile. L'évaluation et la comparaison des approches peuvent néanmoins être établies selon la prise en compte du besoin utilisateur, et la prise en compte de la structure du Web. Une expérimentation utilisateur, avec des experts maîtrisant la RI, peut permettre d'établir la pertinence des résultats retournés.

2.2.5 Synthèse sur la DI

Les robots d'indexation ciblée permettent de collecter une partie du Web spécifique à un besoin en information et ainsi d'éviter la recherche d'information sur un ensemble de pages trop volumineux. Dans ce chapitre, un ensemble d'approches a été décrit, ainsi que des algorithmes tels que le PageRank et HITS, qui exploitent la structure du Web pour juger de la pertinence d'une page. Les différentes approches sont difficiles à comparer, notamment à cause des difficultés liées à l'évaluation des résultats d'un robot d'indexation. Cependant, elles prennent en considération différents paramètres et répondent à différents besoins. L'approche *Fish search* [De Bra & Post, 1994] s'appuie sur l'explicitation par requête du besoin alors que [Ehrig & Maedche, 2003] se sont intéressés à une définition plus exhaustive au travers d'ontologies. D'autres approches renforcent leurs méthodes en utilisant la structure du Web et des métriques afin de ne pas se focaliser uniquement sur l'exploitation du besoin de l'utilisateur [Chakrabarti et al., 1999b, Chakrabarti et al., 1999a, Diligenti et al., 2000]. Les approches par apprentissage sont particulièrement adaptées au Web et à ses changements. Leurs avantages résident notamment dans leur adaptabilité et leur personnalisation considérant le besoin de l'utilisateur, en particulier si ce besoin évolue.

Comme le montre le tableau 2.3, les robots d'indexation des approches décrites commencent l'exploration sur un ensemble d'URLs fournies. Or, considérant le TLP, le Web est composé de sous parties thématiques. Une limitation de ces approches est de se restreindre aux URLs fournies par les moteurs de recherche ou par l'utilisateur comme unique point de départ. En effet, l'utilisateur risque de restreindre le robot d'exploration dans une sous partie du Web et de ce fait, le ralentir dans la découverte d'autres parties d'intérêt. Le tableau 2.3 met également en évidence une autre limitation quant à la sélection des URLs

TABLE 2.3 – Comparaison des approches de collecte ciblée

Entrée du système	Sélection d'une URL	Références
URLs en favoris ou résultats d'un moteur de recherche liés à la requête de l'utilisateur	Comparaison de la page collectée avec les mots de la requête utilisateur	*Fish search* [De Bra & Post, 1994, De Bra et al., 1994, Gasparetti & Micarelli, 2003, Gasparetti & Micarelli, 2004]
URLs fournies par l'utilisateur	Comparaison de la page avec les concepts d'une ontologie	[Ehrig & Maedche, 2003]
	Comparaison de la page avec les concepts d'une ontologie et utilisation de HITS	[Chakrabarti et al., 1999b, Chakrabarti et al., 1999a]
Une URL d'intérêt	Modèle basé sur le *tunneling*	[Bergmark et al., 2002]
Résultats d'un moteur de recherche liés à la requête de l'utilisateur	Texte des liens avec les mots de la requête utilisateur connus par l'agent	*My Spider* [Pant & Menczer, 2002]
	Texte des liens avec les mots de la requête utilisateur connus par l'agent et approche collaborative	[Balabanovic & Shoham, 1997, Balabanovic, 1998]
Aléatoire, même si une initialisation avec des URLs d'intérêt est préférable	Apprentissage de prédicats contenu dans les pages pertinentes	*Intelligent Crawling* [Aggarwal et al., 2001]

pertinentes. Les approches qui s'appuient uniquement sur le contenu textuel des URLs sont limitées par une vision trop partielle du contenu d'un document. De même, le besoin utilisateur est souvent représenté au travers d'une requête, ce qui est également une représentation insuffisante du besoin comme nous allons l'expliquer dans le chapitre suivant. La collecte ciblée reste cependant une très bonne alternative à la RI personnalisée. C'est notamment vrai dans un cadre du ROSO puisqu'il permet aux experts du renseignement d'obtenir des corpus restreints de documents d'intérêt en fonction de leur besoin et ainsi de découvrir de l'information pertinente.

CHAPITRE 3

MODÉLISATION ET ÉVOLUTION DU BESOIN UTILISATEUR

Un profil utilisateur est un ensemble de données permettant de modéliser le besoin de l'utilisateur. Le chapitre précédent a mis en avant les limites des requêtes pour représenter un besoin en information. Les problèmes de polysémie mais aussi des aspects contextuels, que ce soit temporel ou géographique, peuvent fausser la représentation du besoin. Un profil utilisateur permet de représenter toutes les informations nécessaires à la définition du besoin à un moment donné. L'aspect temporel est important puisque le besoin de l'utilisateur peut changer, se préciser, évoluer au fur et à mesure que l'utilisateur consomme de l'information. Les aspects et techniques d'acquisition, de représentation, d'adaptation et d'évolution pour la construction du profil utilisateur sont abordés dans ce chapitre.

3.1 Profil utilisateur pour la RI et la DI

Les modèles de représentation du profil utilisateur pour la RI ont émergés durant la dernière décennie pour aider les utilisateurs à faire face à la quantité croissante d'information disponible sur Internet. Ces profils sont une source de connaissances contenant de multiples informations sur l'utilisateur qui peuvent être utiles au sein d'un système de RI ou de DI (cf. chapitre 2, page 13). Dans cette partie, nous présentons un état de l'art sur la modélisation du profil utilisateur.

3.1.1 Définitions et modèles de profil utilisateur

La personnalisation de la RI, ainsi que l'exploration ciblée en DI, dépendent essentiellement de l'exhaustivité et de l'exactitude du profil modélisé qui est utilisé. Aucun consensus n'existe sur la définition même d'un profil utilisateur. Les profils diffèrent sur leur contenu, sur la méthode d'acquisition employée, mais également sur la représentation et leur adaptation au fil du temps. Cependant, deux types de profil se distinguent dans la littérature.

Le profil cognitif. Il est présent dans différents systèmes personnalisés [Lieberman et al., 1995, Leung et al., 2006, Pazzani et al., 1996] et notamment des systèmes de RI personnalisés [Bouidghaghen, 2009, Daoud et al., 2009]. Il est basé sur les connaissances de l'utilisateur ou sur ses centres d'intérêt. Ces derniers peuvent être à long terme lorsqu'ils désignent les domaines d'intérêt généraux de l'utilisateur. Les centres d'intérêt à court terme sont spécifiques à une session de recherche.

Le profil qualitatif. Il est basé sur des préférences liées à la recherche et aux résultats de la recherche. Dans [Harrathi & Calabretto, 2006], les préférences sont liées à la qualité des informations des résultats. Cette notion de qualité est relative à la satisfaction des besoins de l'utilisateur en termes de choix et d'appréciation des facteurs de la qualité comme la fraîcheur, la crédibilité des sources d'information, la cohérence. Ces aspects couvrent le contexte des documents retournés et ils sont inclus dans la taxonomie du contexte multidimensionnel.

Le profil multidimensionnel. Ces deux profils ont permis à Kostadinov d'introduire la notion de profil multidimensionnel qui couvre les centres d'intérêt et les préférences de l'utilisateur, les caractéristiques de l'environnement et du système [Kostadinov, 2003]. Ce profil reprend l'ensemble des informations présentes dans le profil cognitif et dans le profil qualitatif, d'où la notion de profil multidimensionnel [Kostadinov, 2008, Tamine et al., 2007]. Les différentes dimensions d'un profil multidimensionnel sont les suivantes :

- Le domaine d'intérêt est la dimension centrale du profil utilisateur qui définit le domaine d'expertise et le niveau de qualification pour un domaine d'intérêt donné. En RI, le domaine d'intérêt est généralement représenté par un vecteur de concepts pondérés. Les concepts correspondent à des mots-clés, auxquels s'intéresse l'utilisateur, qui peuvent être organisés en graphe conceptuel ou en ontologie explicitant les relations sémantiques entre concepts.
- Les données personnelles sont la dimension du profil qui caractérise l'utilisateur selon des informations qui lui sont propres telles que son identité (nom, prénom, etc), des données démographiques (date de naissance, genre, état civil, etc) et ses contacts (carnet d'adresses).
- Les données de qualité sont la dimension liée à la qualité attendue des résultats et du processus de RI. La fraîcheur et l'exactitude de l'information représente la qualité du contenu. La qualité de la source d'information est caractérisée par son niveau de confiance et sa fiabilité. Enfin, la qualité du processus de recherche est donnée par le temps de réponse et la précision des résultats.
- Les données de livraison correspondent à la dimension contenant les aspects et les modalités de présentation des résultats de recherche. Des informations liées à la plate-forme de l'utilisateur, à la nature et au volume des informations délivrées, ainsi qu'aux préférences esthétiques ou visuelles y sont renseignées. Deux catégories de données de livraison sont à noter. Celles correspondant aux informations sur les formats de document, leur nombre ou leur taille, et celles liées aux interactions de l'utilisateur, avec des caractéristiques de temps ou de moyen d'accès sur le système.
- Les données de sécurité traitent de la sécurité du profil utilisateur, de la sécurité des résultats et de la sécurité du processus de production des résultats.

Alors que plusieurs systèmes s'accordent sur la définition des différents types de profil, les approches en RI personnalisée diffèrent selon les techniques de modélisation et d'évo-

lution du profil utilisateur utilisées, ainsi que sur l'exploitation du profil. Afin de décrire le contexte dans lequel le profil utilisateur peut être employé, Kostadinov y ajoute un modèle de contexte en trois dimensions :

- La dimension spatiale correspond au lieu où se trouve l'utilisateur. Cela peut correspondre à des coordonnées géographiques comme à une étiquette (travail, maison, etc.).
- La dimension temporelle précise le moment de l'interaction, ce qui se peut se traduire par une date précise, une période ou un moment particulier (matin, soir, etc.).
- L'équipement permet de compléter les informations spatio-temporelles. Cette donnée fournit des informations sur l'équipement utilisé lors de l'interaction (matériel, logiciel, etc.).

Kostadinov a introduit un modèle générique qui permet ainsi de représenter et de classifier l'ensemble des informations sur une personne [Kostadinov, 2003, Kostadinov, 2008]. Contrairement aux autres approches citées précédemment, toutes les dimensions liant un utilisateur, ses informations et son interaction avec le système sont couvertes. Kostadinov propose de constituer de tels profils via une acquisition explicite des données sur l'utilisateur. L'approche ne traite pas de la construction automatique du profil à partir d'une acquisition implicite des données utilisateur.

Nous venons de voir les types de profil utilisateur et les informations que chacun couvre. La section suivante présente les différentes manières de représenter ces informations.

3.1.2 Représentation de profil utilisateur

Une information au sein d'un profil utilisateur peut être représenté par l'historique de recherche, par une représentation ensembliste de mots-clés, connexionniste et conceptuelle.

a. Représentation basée sur l'historique de recherche

Un type de représentation du profil utilisateur est basé sur l'historique de recherche comme le montre la figure 3.1. En fonction des requêtes effectuées et des pages visitées, les centres d'intérêt de l'utilisateur sont représentés par les 10 requêtes les plus tapées, les 10 pages les plus visitées et les 10 pages sur lesquelles l'utilisateur a le plus cliqué. Ces informations sont utilisées par Google lors de futures recherches afin d'auto-compléter le champs de recherche ou de proposer des pages déjà visitées selon la requête tapée. Raghavan et Sever construisent le profil utilisateur aux travers d'une sauvegarde des requêtes de recherche et des clics de l'utilisateur sur les résultats dans une base de données [Raghavan & Sever, 1995]. Cette base de données, basée ainsi sur l'historique de recherche, est la représentation du profil utilisateur.

b. Représentation ensembliste de mots-clés

La représentation ensembliste repose généralement sur le modèle vectoriel de Salton qui utilise un ensemble de termes (mots-clés) ou de vecteurs de termes pondérés pour représenter le profil utilisateur [Salton & Yang, 1973]. Ces ensembles traduisent les centres d'intérêt de l'utilisateur. L'importance d'un centre d'intérêt, qui peut être vu comme le degré d'intérêt de l'utilisateur pour un ensemble de termes, est représenté par le poids des termes correspondant. En RI, ce poids est communément calculé en fonction de la

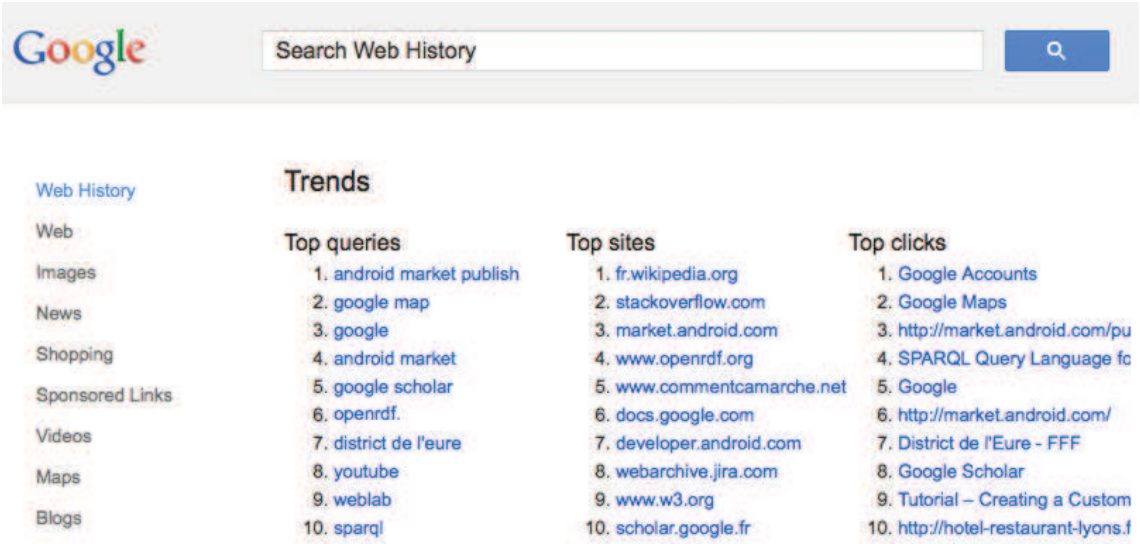


FIGURE 3.1 – Représentation de profil par Google History

fréquence d'apparition des termes dans des documents jugés pertinents implicitement ou explicitement par l'utilisateur [Salton & Yang, 1973]. La formule de pondération TF*IDF est la plus connue. Elle est basée sur la fréquence de termes (TF) et la fréquence inverse du document (IDF) :

- *Term Frequency TF* : La fréquence d'un terme est le nombre d'occurrences de ce terme dans un document. Plus un terme est fréquent, plus il est considéré comme important. Soit le terme t_i et le document d_j, alors la fréquence TF_{ij} du terme dans le document est exprimé ainsi :

$$TF_{ij} = \frac{n_{ij}}{\Sigma_k n_{kj}}$$

  où n_{ij} est le nombre d'occurrences du terme ti dans dj. Le dénominateur est le nombre d'occurrences de tous les termes dans le document d_j
- *Inverse Document Frequency IDF* : La fréquence inverse de document est une mesure de l'importance du terme dans l'ensemble du corpus. Le but est de donner un poids plus important aux termes les moins fréquents, considérés comme plus discriminants. Le logarithme de l'inverse de la proportion de documents du corpus qui contiennent le terme est calculé de la manière suivante :

$$IDF_i = \log \frac{|D - d|}{d}$$

  où d est la proportion des documents contenant le terme et D le nombre total de documents dans le corpus.
- *TF-IDF* : la fonction de pondération est obtenue comme suit :

$$TF\text{-}IDF = TF * IDF$$

La représentation ensembliste existe sous trois formes :

- un ensemble de termes pondérés où chaque terme représente un centre d'intérêt possible de l'utilisateur. Letiza utilise la construction de profil implicite via un historique de recherche, pour ensuite représenter le profil utilisateur comme un ensemble de couples termes-poids [Lieberman et al., 1995].

– un ensemble de vecteurs de termes pondérés où chaque vecteur représente un centre d'intérêt possible de l'utilisateur [Lieberman, 1997, White et al., 2005]. Le système WebMate utilise cette approche pour représenter le profil par N vecteurs où N est le nombre de centres d'intérêt de l'utilisateur [Chen & Sycara, 1998].
– un ensemble de vecteurs de termes pondérés où un sous-ensemble de vecteurs représente un centre d'intérêt possible de l'utilisateur. Cette approche permet de représenter un profil avec des centres d'intérêt multiples pour une recherche [Sieg et al., 2004, Mc Gowan, 2003, Tamine-Lechani et al., 2006].

La représentation ensembliste est relativement simple à mettre en oeuvre et permet de représenter une multiplicité des centres d'intérêt de l'utilisateur. Malgré cela, cette représentation manque de structuration : les aspects de cohérence et de relation de corrélation entre les centres d'intérêt ne sont pas pris en compte. De plus, aucune granularité n'est présente impliquant une absence de représentation de niveau de spécificité ou de généralité au sein des centres d'intérêt.

c. Représentation connexionniste

Avec une représentation connexionniste, l'ensemble des centres d'intérêt est représenté sous la forme d'un réseau de noeuds pondérés où chaque noeud est un centre d'intérêt. Plusieurs systèmes de RI utilisent cette représentation après avoir extrait des termes à partir de documents jugés pertinents par l'utilisateur. Cependant, les approches diffèrent quant à l'utilisation des relations entre noeuds. Pour Micarelli *et al.*, un arc reliant deux noeuds traduit une relation de co-occurence de termes dans un même document [Micarelli & Sciarrone, 2004]. La pondération associée à la relation représente alors la fréquence de co-occurrence entre ces deux termes. Une relation sémantique entre les noeuds permet ainsi de résoudre les problèmes de corrélation et d'incohérence entre les centres d'intérêt. En effet, les termes extraits d'une collection de pages, comme l'historique de navigation, peuvent contenir plusieurs sessions de recherche ou appartenir à un besoin informationnel différent. Les relations permettent ainsi d'identifier les termes représentant un même besoin et d'éviter l'utilisation de termes non-connexes lors de l'exploitation du profil. Les relations entre les termes peuvent également traduire des réécritures possibles entre ces termes lorsqu'ils sont des opérateurs logiques (conjonction, disjonction, substitution et négation). Koutrika utilise cette approche pour transformer une requête utilisateur en une version personnalisée pour mieux décrire le besoin utilisateur [Koutrika & Ioannidis, 2005].

Enfin, les relations entre les termes peuvent être une représentation hiérarchique des centres d'intérêt prenant en compte les niveaux de généralité et de spécificité liés aux centres d'intérêt. Pour ce faire, le réseau est construit en se basant sur un algorithme de clusterisation des termes [Kim & Chan, 2003]. Le tableau 3.1 contient un échantillon de données. La première colonne contient le numéro des pages, la seconde les termes extraits de ces pages. Ces termes peuvent être présentés sous la forme d'une UIH (*User Interest Hierarchy*). C'est à dire une hiérarchie de termes allant des centres d'intérêt généraux de l'utilisateur à ses centres d'intérêt spécifiques comme le montre la figure 3.2, page 48. Dans cet exemple, *perceptron* et *ann* peuvent faire parties d'une catégorie *algorithme de réseaux neuronaux*, alors que *id3* et *c4.5* ne le peuvent pas. Cependant, ces deux ensembles de termes peuvent être réunis dans une catégorie supérieure qu'on appellerait apprentissage statistique.

Le modèle utilisateur est enrichi de sémantique via cette représentation connexion-

TABLE 3.1 – Échantillon de données tiré de [Kim & Chan, 2003]

Page	Content
1	ai machine learning ann perceptron
2	ai machine learning ann perceptron
3	ai machine learning decision tree id3 c4.5
4	ai machine learning decision tree id3 c4.5
5	ai machine learning decision tree hypothesis space
6	ai machine learning decision tree hypothesis space
7	ai searching algorithm bfs
8	ai searching algorithm dfs
9	ai searching algorithm constraint reasoning forward checking
10	ai searching algorithm constraint reasoning forward checking

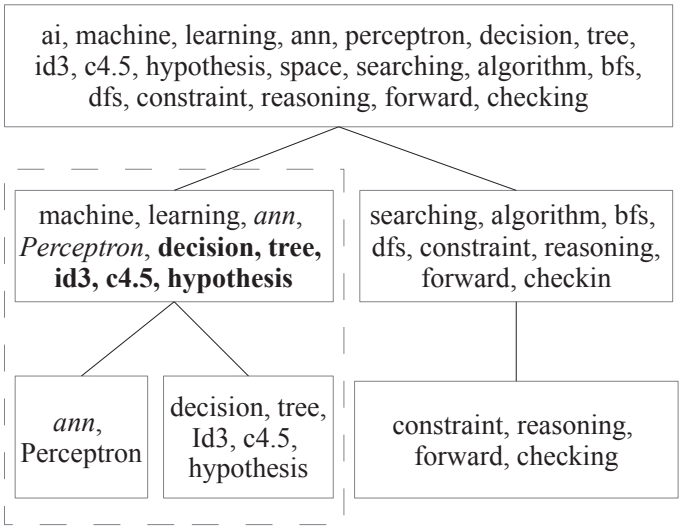


FIGURE 3.2 – Echantillon d'une hiérarchie d'intérêt utilisateur tiré de [Kim & Chan, 2003]

niste. Ainsi, ce modèle permet de pallier les problèmes de la représentation ensembliste. Cependant, certaines limitations sont à noter. Les termes du profil utilisateur sont extraits de l'historique de recherche, ce qui représente une limite en soit. Que ce soit avec cette représentation ou avec les deux précédentes, l'utilisateur est cantonné aux termes qu'il connaît et donc à son profil, ce qui rend difficile la découverte de nouveaux besoins et de nouveaux termes représentant ses centres d'intérêt.

d. Représentation conceptuelle

Afin de pallier le problème de cloisonnement du profil aux termes connus par l'utilisateur, des approches proposent une représentation conceptuelle du profil utilisateur basée sur l'exploitation d'ontologies du domaine. Il s'agit d'une représentation ensembliste où

les mots-clés sont remplacés par des concepts. Ainsi, le profil utilisateur est représenté par un réseau de noeuds de concepts de l'ontologie. Ces concepts sont les centres d'intérêt de l'utilisateur et sont reliés entre eux selon la topologie de l'ontologie. Ces relations peuvent être des relations sémantiques ou bien des relations de subsomption (hiérarchie de concepts) [Liu et al., 2004,Gauch et al., 2003,Sieg et al., 2007]. Ces noeuds conceptuels sont des vecteurs de termes pondérés. Comme vu précédemment, le poids permet de traduire le degré d'intérêt de l'utilisateur pour ce concept. Ainsi, la représentation conceptuelle ne repose plus seulement sur les données utilisateurs mais également sur des ressources sémantiques prédéfinies.

Pretschner et Gauch, pionniers dans l'utilisation d'ontologies de domaine pour la modélisation de profil utilisateur en RI, utilisent une ontologie de subsomption afin de classifier les documents par concept [Gauch et al., 2003]. Ils déduisent ainsi de l'analyse de l'historique de recherche, les centres d'intérêt de l'utilisateur.

Dans cet esprit, plusieurs approches utilisent l'ODP. Il se présente sous la forme d'une ontologie de subsomption largement utilisée par les systèmes de RI personnalisée en tant que ressources sémantiques [Liu et al., 2004, Sieg et al., 2007, Chirita et al., 2005, Challam et al., 2007]. L'ontologie représentant le profil utilisateur est une instance annotée de l'ontologie de référence, en vue des requêtes de l'utilisateur (voir figure 3.3 page 50) ou de son historique de recherche [Liu et al., 2004]. Chaque concept est associé à un poids en fonction de sa pertinence par rapport aux intérêts de l'utilisateur. Une mesure de similarité est calculée entre les documents et les concepts de l'ontologie de référence. Le poids d'un concept c_i est calculé par accumulation de son score de similarité avec les documents d . La mesure de similarité de c_i avec un document d_j est calculée ainsi :

$$similarity(c_i, d_j) = \sum_{k=1}^{n} w_{ki} * d_{kj}$$

où n est le nombre total de termes présents dans les documents, w_{ki} est le poids du terme t_k dans le concept c_i, et d_{kj} le poids de ce même terme t_k dans le document d_j.

L'instance de l'ontologie est utilisée pour personnaliser les recherches suivantes de l'utilisateur [Challam et al., 2007], ou les désambiguïser [Liu et al., 2004]. Elle peut également servir à réorganiser les résultats de recherche afin de présenter les pages les plus pertinentes à l'utilisateur [Sieg et al., 2007].

3.1.3 Construction de profil utilisateur

La modélisation d'un profil utilisateur comprend une étape d'acquisition des informations et des données de l'utilisateur. Cette acquisition peut être explicite et/ou implicite.

a. Acquisition explicite

Lors d'une acquisition explicite, l'utilisateur peut intervenir sur son profil directement, par exemple avec une saisie manuelle de ses centres d'intérêt ou en fournissant au système un ensemble de documents représentant ses centres d'intérêt. Dans le cadre de la RI personnalisée, l'acquisition explicite des informations et des données se fait au travers de formulaires ou grâce à des techniques de rétroaction (*feedback*) de la part de l'utilisateur. Le *feedback* permet à l'utilisateur de noter les documents pertinents qui lui sont retournés lors d'une RI. Ces techniques servent à construire et à mettre à jour le profil utilisateur.

Plusieurs systèmes de RI personnalisée en ligne utilisent ce type d'acquisition. C'est notamment le cas de Google Alertes [1] qui propose à l'utilisateur de le tenir au courant sur un sujet donné. L'utilisateur renseigne des mots clés, décrivant ainsi ses centres d'intérêt qui seront le sujet de l'alerte. Les options avancées de ce service permettent à l'utilisateur d'expliciter d'autres aspects de son profil en renseignant, par exemple, le volume de résultats contenu dans chaque alerte, ou encore la source pour spécifier un type particulier de résultat. Ces dernières informations permettent de renseigner les *données de livraisons*.

La recherche avancée du moteur de recherche Bing [2] permet à l'utilisateur de saisir explicitement un ensemble de termes qui représente ses centres d'intérêt. Le choix des termes peut être limité par une hiérarchie de concepts [Sieg et al., 2004]. Le système proposé permet l'utilisation d'une ontologie de subsomption comme l'Open Directory Projet (ODP) [3]. L'utilisateur entre une requête à laquelle le système répond par la présentation de concepts liés aux mots-clés de la requête. L'utilisateur peut ensuite sélectionner les portions de concepts en rapport avec son besoin et retirer les portions non pertinentes (cf Figure 3.3).

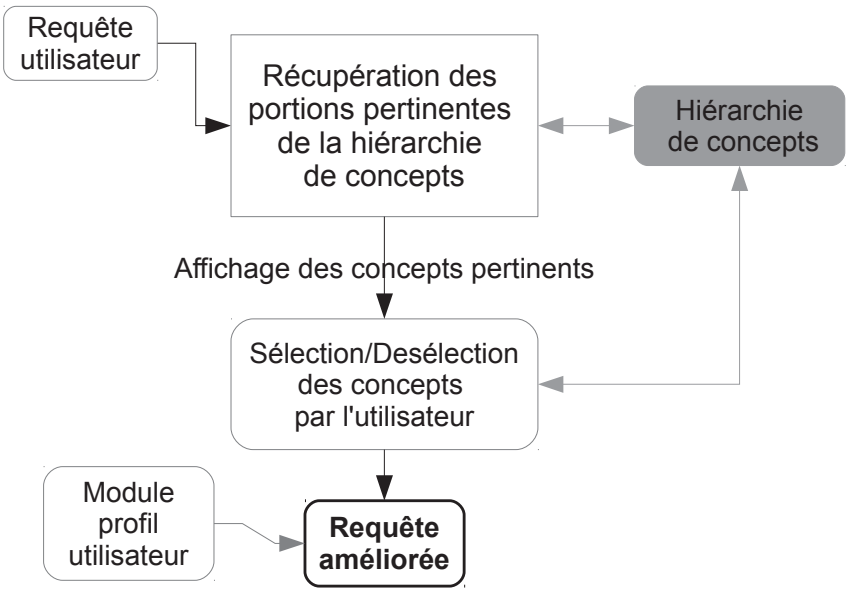


FIGURE 3.3 – Construction explicite du profil par [Sieg et al., 2004]

L'ODP est aussi utilisé par Liu et Yu afin que l'utilisateur puisse clarifier son besoin après une requête, en sélectionnant les concepts adéquats [Liu et al., 2004]. Cela permet d'avoir un contexte de recherche lié à une requête et de désambiguïser les mots utilisés dans les requêtes. Le feedback explicite permet également d'obtenir des informations sur les préférences et centres d'intérêt de l'utilisateur. Ce système sert plus souvent à mettre à jour un profil existant qu'à en construire un nouveau. Koutrika et Ioannidis l'utilisent afin que l'utilisateur puisse identifier les documents pertinents et dans un second temps pour construire un réseau de termes représentant le profil utilisateur [Koutrika & Ioannidis,

1. http://www.google.fr/alerts
2. http://www.bing.com/
3. http://www.dmoz.org/

2005].

b. Acquisition implicite

En RI, l'acquisition implicite de données pour remplir un profil utilisateur est généralement basée sur l'historique de recherche de cet utilisateur. L'historique est exploité de deux manières différentes. La première consiste à utiliser l'ensemble des requêtes formulées par l'utilisateur et les résultats de recherches retournés [Rich, 1979, Pazzani et al., 1996, Tanudjaja & Mui, 2002]. Cette construction implicite peut être améliorée en analysant l'historique des données de clics de l'utilisateur sur les résultats des recherches [Tan et al., 2006]. La seconde utilise uniquement les sites présents dans l'historique qui sont les pages Web visitées par l'utilisateur [Lieberman et al., 1995, Chen & Sycara, 1998, Mladenic, 1996, Gauch et al., 2003, Liu et al., 2004]. L'historique de recherche peut être utilisé afin de constituer une base de données de requêtes et de résultats associés. Cette base permet ensuite de trouver les requêtes similaires à celle en cours d'évaluation par le système [Raghavan & Sever, 1995].

Le moteur de recherche Google propose un système équivalent à ces approches afin de personnaliser les résultats de recherche en fonction du profil utilisateur. Ce profil utilisateur est construit par Google en fonction de l'historique des pages Web visitées et des recherches réalisées dans Google Web History [4]. Il suffit à l'utilisateur de s'inscrire sur ce site afin que la collecte de données pour la construction de son profil puisse se faire de manière implicite.

Outre l'historique de navigation, l'analyse des favoris de l'utilisateur permet également de construire son profil. Contrairement aux approches précédentes, l'avantage des favoris est d'analyser des pages pertinentes pour l'utilisateur contrairement au bruit qui peut être présent dans l'historique des pages visitées. Michlmayr *et al.* extrait des favoris les mots-clés (*tags*) [Michlmayr & Cayzer, 2007]. Les mots-clés les plus fréquents présents dans les différentes pages servent à identifier les centres d'intérêt de l'utilisateur.

L'identification des pages pertinentes peut également se faire au travers de l'analyse des interactions de l'utilisateur avec le système. Une page Web présente dans un résultat de recherche peut s'avérer pertinente si l'utilisateur a cliqué dessus, s'il est resté un certain temps sur la page, l'a sauvegardé ou imprimé [Pretschner & Gauch, 1999, Daoud et al., 2009]. Ces techniques sont des techniques de rétroactions implicites : elles s'appuient sur les actions de l'utilisateur durant sa navigation pour déduire ses centres d'intérêt.

3.1.4 Aspect temporel du profil utilisateur

Le profil utilisateur représente, entre autres, le besoin et les centres d'intérêt de l'utilisateur. Ces informations peuvent changer au cours du temps. Aussi, le profil utilisateur doit pouvoir évoluer afin de prendre en considération ces variations. L'ajout ou la suppression incrémental d'informations dans le profil utilisateur permet d'avoir une perspective temporelle du profil. Cette dimension temporelle permet d'introduire la notion de profil à court terme et à long terme mais aussi la notion de session.

4. http://www.google.com/history

a. Profil à court terme

La notion de court terme fait référence à une tâche de recherche courante, à un besoin spécifique. Shen *et al.* exploitent le profil utilisateur à court terme afin de mieux cibler la recherche aux vues des données spécifiques et pertinentes qu'il contient [Shen et al., 2005]. Autrement dit, ce profil permet d'améliorer la précision des recherches en relation directe avec les centres d'intérêt courant de l'utilisateur. Dans cette optique, des travaux [Shen et al., 2005, Sieg et al., 2007] assimilent le profil utilisateur à court terme à un besoin en information unique. Un tel profil n'est utile que durant une seule session de recherche. Par exemple, le profil à court terme dans [Sieg et al., 2004] varie à chaque soumission de requête en analysant le sujet de celle-ci. Le profil conceptuel évolue également à court terme dans [Sieg et al., 2007, Gauch et al., 2003] où les poids des concepts du profil évoluent en fonction de l'information récemment visualisée ou ajoutée par l'utilisateur.

b. Profil à long terme

Le profil à long terme correspond à un profil modélisant des centres d'intérêt généraux, persistants ou récurrents. Contrairement au profil à court terme qui, dans certains cas, ne dure que le temps d'une session, le profil à long terme peut améliorer la recherche pour n'importe quelle requête soumise par l'utilisateur. Google Alertes [5] construit explicitement le profil utilisateur avec un ensemble de domaines de recherche considérés comme stables. Dans les approches dites proactives [Sun et al., 2005, Mc Gowan, 2003], une étape préliminaire est nécessaire afin de construire le profil utilisateur via une analyse des requêtes et pages Web visitées. Pour le profil à long terme, il est important de prendre en considération l'aspect évolutif des besoins et des centres d'intérêt des utilisateurs. Google personalized search, qui propose un moteur de recherche personnalisé, gère cette évolution de profil en analysant les nouvelles pages Web visitées et les dernières requêtes entrées. Le profil est ensuite modifié en fonction d'éventuels centres d'intérêt appris lors de cette analyse. Dans [Sieg et al., 2004], si le profil à long terme contient un contexte de recherche correspondant à la requête courante, le contexte contenu dans le profil est mis à jour. Un nouveau contexte de recherche est ajouté au profil dans le cas où la requête en cours d'évaluation ne s'apparente à aucun contexte connu. Afin de différencier ce profil long terme du profil court terme, la notion de session a été introduite.

c. Les sessions

En RI, la notion de profil à court terme fait référence à la délimitation de sessions de recherche. Une session peut être délimitée selon un intervalle de temps ou selon une séquence de requêtes sur un même besoin. Durant un intervalle de temps, une session peut regrouper des requêtes définissant des besoins différents. Trois types d'approches ont été définies afin de différencier ses sessions de recherche :

– Dimension temporelle. Une des approches de différenciation des sessions de recherche repose sur la notion de *Timeout* [He & Göker, 2000], ou de temps mort, entre deux sessions. La session est définie par un ensemble de requêtes de recherche. Le *Timeout* correspond à une durée moyenne entre deux sessions de recherche. Lorsque cette durée est atteinte entre deux requêtes de l'utilisateur, l'approche considère que la

5. http://www.google.fr/alerts

dernière requête correspond à une nouvelle session. Le calcul de ce seuil de temps moyen est cependant problématique dans la mesure où la durée d'une recherche peut varier significativement d'un utilisateur à l'autre et même pour un même utilisateur, d'une session à une autre. Une méthode basée sur le temps d'affichage des pages est proposée dans [Patterns, 1999]. La différenciation des sessions est faite selon la durée de lecture des pages par l'utilisateur. Si la durée, appelée *reference length*, est atteinte lors d'une lecture d'une page, alors cette page répond au besoin de l'utilisateur et la session de recherche est terminée. La limite de cette approche est dûe au postulat selon lequel une seule page pertinente suffit à répondre au besoin de l'utilisateur.

– Calcul de similarité. La différenciation des sessions de recherche peut être faite via l'analyse du contenu des requêtes. Une méthode consiste à calculer le nombre de termes en commun entre deux requêtes successives p et q [Wen et al., 2001] :

$$similarity(p,q) = \frac{ntc(p,q)}{Max(nt(p),nt(q))}$$

où ntc est le nombre de termes communs présents dans les requêtes p et q, et le dénominateur est le nombre maximum de termes présents dans ces deux requêtes. Une autre méthode consiste à calculer la distance entre deux requêtes successives en considérant chaque caractère [Gusfield, 1997].

– Feedback. Un dernier type de différenciation des sessions repose sur la notion de partage de documents entre deux résultats de requêtes. Une même page visitée via deux requêtes différentes permet de regrouper ces requêtes et ainsi de définir des sessions de recherche [Wen et al., 2001]. Cette méthode a été étendue afin de prendre également en considération une distance conceptuelle entre les pages visitées. Le calcul est basée sur l'utilisation d'une hiérarchie de concepts (voir la représentation conceptuelle dans la section 3.1.2, page 45).

3.1.5 Synthèse sur la modélisation du besoin

Un ensemble d'approches de modélisation, de construction et de représentation du profil utilisateur, a été présenté dans ce chapitre. Les informations contenues dans un profil utilisateur peuvent être plus ou moins complètes, et leurs représentations diverses comme l'illustre le tableau 3.2.

L'avantage de l'acquisition explicite est d'avoir un contrôle sur la construction du profil utilisateur. Cependant, celle-ci est limitée par l'interface du système qui doit être performante et bien pensée afin que l'utilisateur puisse expliciter son besoin correctement, sans ambiguïté [Liu et al., 2004], et dans sa globalité [Wærn, 2004]. Prenons comme exemple le cas où le besoin est de trouver de nouvelles pages dont le sujet est le même que des pages connues par l'utilisateur. Celui-ci peut simplement fournir au système ses pages d'intérêt, ce qui ne demande pas une interface complexe, ni un effort particulier à l'utilisateur. A contrario, si les pages fournies traitent de différents sujets, et que l'utilisateur veut expliciter les différents centres d'intérêt qui en découlent et différencier plusieurs sessions de recherche, l'interface devient plus lourde et l'effort demandé à l'utilisateur est plus important. Cet effort supplémentaire est souvent mal perçu. Des travaux se sont orientés vers une acquisition implicite des informations utilisateur afin de contrer ces limitations [Mladenic, 1996, Gauch et al., 2003, Pazzani et al., 1996, Tanudjaja & Mui, 2002]. Contrairement à

6. https ://www.google.com/history/

TABLE 3.2 – Synthèse des approches de modélisation du profil utilisateur

Type de représentation	Acquisition des informations	Références
Représentation basée sur l'historique de recherche : top requêtes, top pages visitées	Historique de recherche. Mise à jour du profil au fil de la navigation	Google history [6]
Représentation ensembliste	Extraction de termes pondérés des pages de l'historique de recherche. Mise à jour du profil au fil de la navigation	[Lieberman et al., 1995]
	Extraction d'un ensemble de vecteurs termes pondérés. Centres d'intérêt multiples pour une recherche.	[Sieg et al., 2004, Mc Gowan, 2003, Tamine-Lechani et al., 2006, Lieberman, 1997, White et al., 2005]
Représentation connexioniste	Extraction de termes reliés par des arcs de co-occurrence. Ajout de noeuds au fur et à mesure que l'utilisateur fournit de nouveaux documents	[Micarelli & Sciarrone, 2004]
	Extraction de termes reliés par des connecteurs logiques. Ajout de noeuds au fur et à mesure que l'utilisateur fournit de nouveaux documents	[Koutrika & Ioannidis, 2005]
	Extraction de termes reliés par des relations hiérarchiques. Utilisation d'un algorithme de clusterisation	[Kim & Chan, 2003]
Représentation conceptuelle	Graphe de termes reliés par des relations sémantiques selon une ontologie de domaine	[Liu et al., 2004, Sieg et al., 2007, Chirita et al., 2005]
	Analyse de l'historique de recherche et classification de documents en fonction des concepts d'une ontologie de subsomption.	[Gauch et al., 2003]

l'acquisition explicite, une perte de pertinence et de précision apparaît avec l'acquisition implicite. Les données sont utilisées pour représenter le profil utilisateur. Les différents types de représentation sont directement liés à l'emploi qui est fait des profils utilisateur par le système : exploitation des informations personnelles, des préférences de livraison, d'un ou plusieurs centres d'intérêt, etc.
Afin d'éviter les tâches d'explicitation du besoin et de parer au manque d'information lors de la construction du profil, des travaux ont été menés afin d'améliorer le profil avec des retours de pertinence. Ces travaux, présentés dans la partie suivante, permettent de faire évoluer le profil utilisateur tout en prenant en compte des changements dans son besoin en information.

3.2 Le retour de pertinence

Les chapitres précédents montrent que les utilisateurs rencontrent les difficultés pour exprimer leur besoin en information lors de leurs recherches en ligne. Comme nous l'avons vu, des techniques existent afin de modéliser ce besoin sous forme de profil utilisateur et de l'exploiter afin de trouver l'information via des approches de RI ou d'exploration du Web. Aussi, comme l'utilisateur est en mesure de juger l'information qui lui est retournée, des techniques de retour de pertinence peuvent être employées. Le retour de pertinence (RP), ou rétroaction de pertinence (relevance feedback), couvre un ensemble de techniques destinées à améliorer la requête ou la représentation du besoin d'un utilisateur. Dans ce chapitre, différentes approches de RP seront présentées dont l'approche implicite qui constitue l'un des axes d'étude de notre approche.

3.2.1 Itérations de RP

Les techniques de RP permettent une interaction entre l'homme et la machine et offre la possibilité au système d'améliorer la description du besoin de l'utilisateur. A chaque itération, l'utilisateur, en jugeant les documents retournés comme étant pertinents ou non pertinents, permet au système d'affiner la modélisation du besoin informationnel. Ainsi, en capitalisant ces informations de manière explicite ou implicite, le système sera en mesure de retourner des documents plus pertinents lors des itérations suivantes.

a. Le RP explicite

L'utilisation de questionnaires ou d'annotations permet un retour de l'utilisateur sur la pertinence des résultats obtenus. Cependant, les utilisateurs sont souvent réticents à l'idée de consacrer du temps dans une démarche additionnelle consistant à juger les documents retournés. L'intérêt d'une telle démarche n'est pas perçue comme utile par l'utilisateur, surtout pour les novices en matière de RI. Pour les experts du renseignement dont le travail consiste à analyser les sources d'intérêt découvertes, la tâche du jugement des résultats est plus naturelle. Aussi, les systèmes utilisant des techniques de RP tendent à simplifier au maximum l'interface homme/machine pour limiter la pénibilité de la tâche d'évaluation. Le moteur de recherche Google, avant de créer son réseau social Google+, utilisait une simple étoile qui permettait à l'utilisateur de marquer son intérêt pour un résultat particulier comme l'illustre la figure 3.4[7]. Un autre système a été proposé par Ruthven [Ruthven

7. http://www.waebo.com/google-les-etoiles-disparaissent-des-pages-de-resultats.html

et al., 2002] : un curseur gradué et continu permet à l'utilisateur de juger simplement de la pertinence d'un document (figure 3.5). Afin d'éviter à l'utilisateur la tâche de jugement, une solution consiste à opter pour des techniques de RP implicite.

FIGURE 3.4 – L'étoile pour le RP de Google

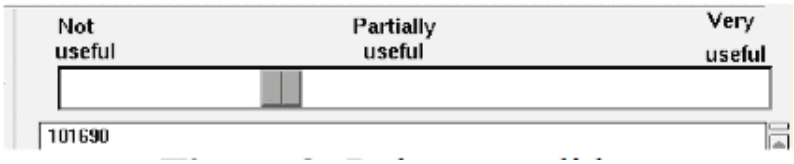


FIGURE 3.5 – Exemple de système avec retour de pertinence [Ruthven et al., 2002]

b. Le RP implicite

Le RP implicite consiste à observer le comportement et les interactions de l'utilisateur autour des documents présentés afin d'en déduire ceux qui sont pertinents. Les approches de RP implicite se basent sur le temps de lecture, la sélection et les mouvements du pointeur de souris, les demandes d'impression, de sauvegarde, ou tout autre interaction pouvant être exploitée. Si un utilisateur passe du temps sur un document, le système peut en déduire que l'utilisateur y porte un intérêt. Il en est de même lorsque l'utilisateur imprime ou sauvegarde un document. Ces informations présentent donc une alternative au RP explicite mais il est nécessaire de procéder à une analyse fine des différentes interactions et de les interpréter judicieusement afin d'éviter des erreurs sur la pertinence d'un document. C'est l'objet d'une partie des travaux de Dupont qui s'intéresse à l'apprentissage implicite pour la RI au travers de ces différents critères [Dupont et al., 2011]. Le RP implicite ou explicite peut être exploité dans différents modèles en fonction de la représentation du besoin informationnel et du système de RI.

3.2.2 Les modèles de RP

Qu'il soit explicite ou déduit implicitement, le RP a toujours pour vocation d'améliorer la représentation du besoin de l'utilisateur par le système. Il existe différentes approches permettant de modéliser le RP et de l'exploiter. Une partie de ces approches est présentée ci-dessous. La définition suivante du RP permet de couvrir l'ensemble de ces modèles :

"L'apprentissage par retour de pertinence consiste à apprendre un certain modèle de l'utilisateur par des déductions basées sur l'étude d'indicateurs de comportement issus de l'interaction de ce même utilisateur avec un système de recherche d'information."

a. Le modèle booléen

La logique booléenne permet de construire des requêtes avec de simples opérateurs "ET" et "OU" entre chaque mot composant la requête. Harman propose d'étendre ce modèle avec du RP [Harman, 1992]. Son approche suggère à l'utilisateur une liste de nouveaux mots-clés à ajouter dans la requête. Ces mots-clés sont extraits des documents jugés comme pertinents par l'utilisateur. Khoo et Poo proposent d'adapter automatiquement la requête en laissant leur système ajouter les mots-clés et les opérateurs en exploitant également les documents jugés pertinents par l'utilisateur [Khoo & Poo, 1994]. Ces approches sont limitées à une correspondance exacte entre les termes de la requête et les documents recherchés. Par exemple, la requête "système ET recherche ET information" retournera les documents qui contiennent les trois mots "système", "recherche" et "information". A l'inverse, la requête "système OU (recherche ET information)" retournera les documents qui contiennent le mot "système" et ceux qui contiennent à la fois "recherche" et "information". Le choix de l'opérateur par le système de retour de pertinence est donc un choix délicat qui influence énormément les résultats.

Ce modèle n'utilise pas de pondération sur les termes de la requête. Le résultat n'est donc pas ordonné et l'utilisateur doit ajouter, retirer des termes ou complexifier la requête si il veut rendre la taille du résultat utilisable. De plus, l'ordre des termes n'est pas exploité de la même façon d'un système à l'autre donnant des résultats différents pour une même requête [Borgman, 1996]. Une alternative à la correspondance exacte est l'utilisation de la fréquence d'apparition des termes de la requête dans les documents retournés. L'utilisation de mesures, comme le TF*IDF [Salton & Yang, 1973], permet d'ordonner la liste des résultats en fonction de la pertinence des documents. Willie and Bruza soutiennent que le problème de ce modèle n'est pas seulement la correspondance exacte de la requête avec les documents [Willie & Bruza, 1995]. Ils pensent que cette technique n'est pas adaptée à la façon de penser des utilisateurs lors d'une recherche. Cependant, elle reste populaire car elle a l'avantage d'offrir à l'utilisateur un contrôle explicite de son besoin.

b. Le modèle vectoriel

Dans le modèle vectoriel, un document est représenté sous la forme d'un vecteur composé de termes (mots-clés ou concepts) présents dans ce document. Le besoin informationnel de l'utilisateur est également représenté par un vecteur qui peut correspondre aux termes d'une requête, à une représentation conceptuelle du besoin, ou à une autre forme de représentation comme présentée dans le chapitre 3, 43, sur la modélisation du profil utilisateur. L'avantage du modèle vectoriel sur le modèle booléen est de pouvoir pondérer

les termes. Cette pondération peut refléter dans le vecteur du document le nombre d'occurrences d'un terme. Dans le vecteur représentant le besoin informationnel, la pondération permet de donner plus ou moins d'importance à chaque terme.

Les travaux de Rocchio sont à l'origine du RP explicite. Ses travaux sont basés sur un modèle vectoriel dont le RP a pour but d'affiner la requête de l'utilisateur [Rocchio, 1971]. Le système reformule la requête initiale posée par l'utilisateur et avance par itérations dans la recherche en fonction des jugements de l'utilisateur sur les documents retournés. La figure 3.6 représente le fonctionnement de ce système de RI.

Dans le modèle proposé par Rocchio, le jugement de l'utilisateur sur un document permet d'y associer une valeur de pertinence, souvent comprise dans le couple de valeurs {+1 ;-1} correspondant à pertinent ; non-pertinent. Avec cette information, à chaque itération, le système optimise la requête indépendamment de tout modèle, suivant cette équation :

$$q^* = arg_{q_i} max(sim(S^{q_i}, S_r) - sim(S^{q_i}, S_{\bar{r}}))$$

où q^* est la requête optimale, q_i la requête courante, S^{q_i} l'ensemble des documents retournés par la requête courante, S_r et $S_{\bar{r}}$ représentent respectivement l'ensemble des documents jugés pertinents et non pertinents par l'utilisateur et la fonction $sim(x, y)$ est une fonction de similarité entre deux ensembles de documents.

Dans le domaine du RP, l'équation de Rocchio est plus connue sous sa forme vectoriel adaptée par Salton et Buckley [Salton & Buckley, 1997] :

$$q_{i+1} = \alpha q_i + \frac{\beta}{|S_r|} \sum_{d_k \in S_r} d_k - \frac{\gamma}{|S_{\bar{r}}|} \sum_{d_k \in S_{\bar{r}}} d_k$$

où α, β et γ sont des termes pondérateurs compris entre 0 et 1, et d_k est la représentation vectorielle de la $k^{ème}$ ressource.

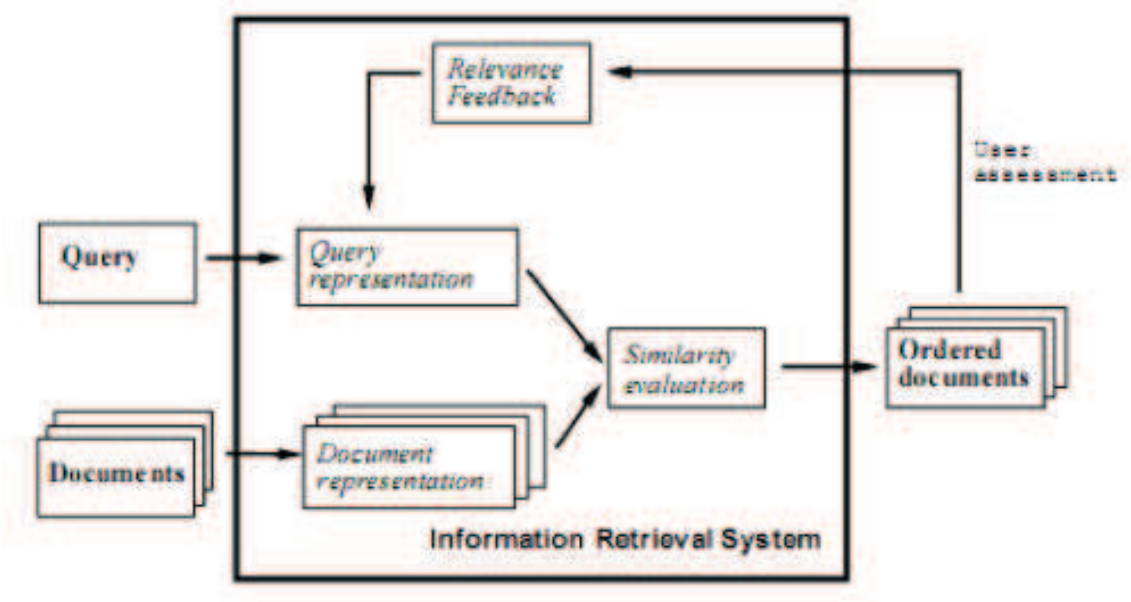


FIGURE 3.6 – Exemple de système avec retour de pertinence [Crestani, 1995]

Afin d'éviter que la requête contienne un nombre excessif de termes, les termes dont le poids est inférieur à un seuil fixé empiriquement sont supprimés. Une autre solution est de limiter la taille de la requête aux N termes de poids les plus forts. Le graphique 3.7 place la requête dans un espace de documents. La requête initiale retourne des documents qui selon les jugements de pertinence permettra de construire une nouvelle requête et ainsi de suite jusqu'à la production de la requête optimale. L'amélioration des requêtes est un

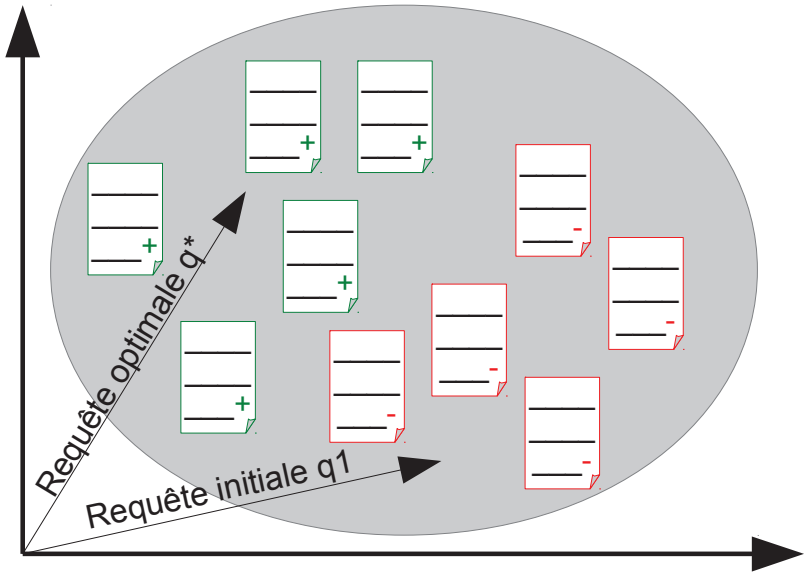


FIGURE 3.7 – Principe d'optimisation d'une requête avec RP

comportement adopté par les utilisateurs lors d'une session de RI classique. Un système qui utilise le RP pour améliorer les requêtes permet d'enlever une tâche à l'utilisateur et se rapproche donc d'un comportement humain. La modélisation du besoin n'est pas nécessairement une requête et l'utilisation de ce modèle de RP peut également servir à d'autres formes de représentation du besoin (cf. partie 3.1.1, page 43).

c. Le modèle probabiliste

Le modèle probabiliste repose également sur une représentation du besoin sous forme de vecteurs mais la similarité entre le besoin informationnel et le document est remplacé par une fonction d'appariement probabiliste. Maron et Kuhns [Maron & Kuhns, 1960], ainsi que d'autres approches [Robertson & Jones, 1976, Van Rijsbergen, 1986], utilisent le RP afin d'ajuster leurs différentes probabilités sur les documents à retourner à l'utilisateur. Ce modèle considère que les termes sont indépendants les uns des autres dans la mesure où la probabilité de voir un terme t n'est pas influencée par la présence du terme s dans le même document. Deux hypothèses d'indépendance sont proposées par Robertson et Spark Jones [Robertson & Jones, 1976] :

– La distribution des termes dans les documents pertinents est indépendante ainsi que leur distribution dans tous les documents.

– La distribution des termes dans les documents pertinents est indépendante ainsi que leur distribution dans les documents non-pertinents.

Ces deux hypothèses se distinguent par le fait de mesurer la probabilité d'apparition des termes des documents pertinents par rapport à leur présence dans les documents non-pertinents ou bien par rapport à leur présence dans l'ensemble du corpus. Ainsi, comme dans l'approche de Rocchio, le modèle probabiliste peut exploiter, ou non, les retours de pertinence sur les documents jugés non-pertinents.

3.2.3 Le retour de pertinence négatif

La plupart des techniques de retour de pertinence compare le contenu des documents jugés pertinents par l'utilisateur avec le contenu des documents non-pertinents. Ces derniers peuvent cependant correspondre à deux différents groupes de documents :

– ceux qui ont été explicitement marqués comme étant non-pertinents par l'utilisateur.
– ceux qui n'ont pas été marqués pertinents par l'utilisateur.

Dans un corpus avec peu de documents retournés à l'utilisateur, les documents non marqués par l'utilisateur peuvent être assimilés à des documents jugés non pertinents. Lorsqu'il s'agit d'un corpus de taille importante, comme celui fourni par l'initiative TREC [8], un ensemble de documents est noté non pertinent. Ceci signifie que les documents ont été jugés comme ne comportant pas d'information pertinente. Les documents non marqués par l'utilisateur peuvent être des documents que celui-ci n'a pas évalué ou alors qu'il a implicitement jugé non pertinent. Pour trouver les documents jugés implicitement non pertinents par l'utilisateur, les approches s'accordent à dire qu'un document consulté par l'utilisateur et non marqué comme pertinent est un document non pertinent.

Dans la mesure où les documents non-pertinents sont reconnus par le système, leur exploitation est une difficulté supplémentaire. Un document est-il non pertinent car l'information qu'il contient ne couvre pas le besoin en information, ou alors l'information qu'il contient est déjà connu de l'utilisateur ? Il existe diverses raisons pour lequel un document est non pertinent à la vue de l'utilisateur. Quelque soit le modèle utilisé pour exploiter le retour de pertinence, la pondération affectée aux retours de pertinence négatif est inférieure aux retours positifs. Ces derniers étant sans équivoques contrairement aux jugements négatifs.

3.2.4 Exploitation du retour de pertinence

Outre le modèle qui diffère d'une approche à l'autre, l'exploitation du retour de pertinence varie également. Il sert dans tous les cas à mieux définir le besoin de l'utilisateur mais l'impact de son exploitation peut intervenir en différents points. L'exploitation du retour de pertinence est fortement liée à la personnalisation d'une recherche ou à la découverte d'information ciblée puisque le jugement de l'utilisateur est pris en compte. Aussi, il est logique de retrouver des approches présentées dans le chapitre sur la RI et la DI (chapitre 2, page 13) qui utilisent le retour de pertinence pour améliorer leur système. La plupart du temps, le RP est exploité dans le but d'enrichir et d'améliorer la requête de l'utilisateur afin de trouver la représentation optimale du besoin de l'utilisateur. Le schéma 3.8 inclut le RP au sein d'un système de RI en reprenant le schéma 2.1 de la page 16. Les flèches en pointillées montrent les différents impacts possibles du RP dont les principaux sont

8. http://trec.nist.gov/

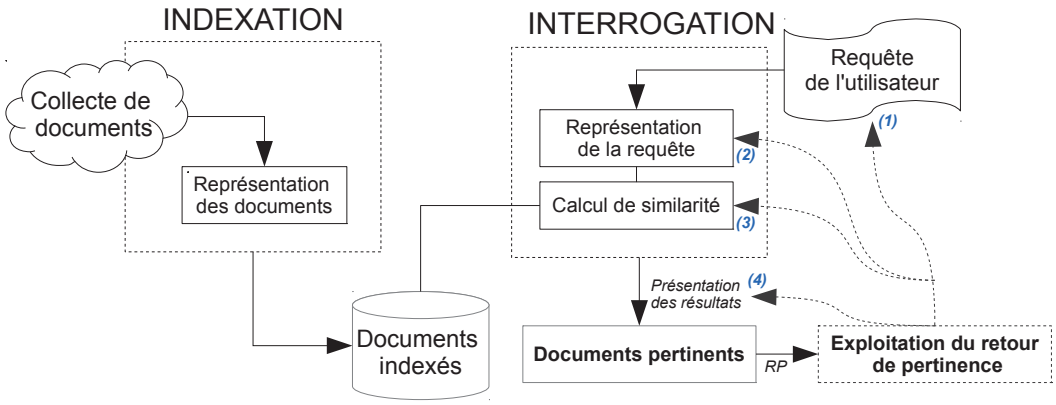


FIGURE 3.8 – Système de RI avec RP

décrits ci-dessous. Dans la littérature, l'exploitation du RP intervient pour ré-écrire (1) ou suggérer des requêtes (2), ré-ordonner les résultats de recherche ou suggérer des documents (3), adapter les présentation des résultats (4). Les deux approches les plus courantes sont la reformulation des requêtes et le ré-ordonnancement des résultats.

a. Reformulation de la requête

La reformulation de la requête peut être implicite et donc automatique ou prendre place au sein d'un échange interactif avec l'utilisateur. La reformulation implicite est courante dans les moteurs de recherche qui ajoutent automatiquement des informations dans les requêtes comme la langue de l'utilisateur ou des termes synonymes. Ces actions sont faites sans que l'utilisateur s'en aperçoive. Le RP permet ici d'ajouter des termes ou d'en supprimer, ou bien encore de changer le poids des termes de la requête. Son avantage est également un inconvénient puisqu'il permet de limiter le recours de l'utilisateur mais celui-ci perd également la main sur sa requête. La seconde approche propose à l'utilisateur des reformulations possibles de sa requête aux vues des RP. Le système met à la disposition de l'utilisateur des termes à ajouter ou à supprimer ou même des requêtes entièrement reformulées. C'est l'utilisateur qui prend la décision finale de modifier ou non la représentation de son besoin. Cette intervention de l'utilisateur a l'avantage d'assurer une meilleure représentation du besoin. L'inconvénient est que l'utilisateur est un acteur important dans ce processus et est alors obligé de réaliser une tâche supplémentaire.

Les approches existantes se distinguent à la fois sur cet aspect automatique ou interactif de la reformulation mais également sur la source d'exploitation du RP. L'évaluation des résultats de recherche par l'utilisateur est le plus exploité ainsi que l'historique de recherche [Spink et al., 2000, Lin et al., 2006]. La co-occurence des termes dans la collection de documents [Gauch & Wang, 1997] ou les propriétés des documents [Chang et al., 2006] peuvent également être analysées pour obtenir un RP. La reformulation de la requête est, malgré les différentes techniques employées, une approche fortement utilisée par les travaux sur le retour de pertinence.

b. Ré-ordonnancement des résultats et suggestion de documents

Le retour de pertinence peut également être exploité pour ré-ordonner les résultats d'une requête. L'ordre des documents retournés est fourni par la mesure de similarité entre ces documents et la requête. Le retour de pertinence peut être utilisé au sein de cette mesure. Radlinski utilise une mesure à base de SVM et d'analyse de clics [Radlinski & Joachims, 2006]. À chaque présentation de résultats, les clics de l'utilisateur sont observés afin de déduire les documents pertinents et non-pertinents qui lui sont présentés. Cette observation permet au système de modifier la frontière du SVM et ainsi d'améliorer la mesure de similarité.
Lorsque la mesure de calcul n'est pas accessible, le ré-ordonnancement peut s'opérer directement sur les documents retournés ou alors, en amont, en modifiant la requête. Cette dernière approche, utilisée par Zhang, se différencie de la reformulation de requête dans la mesure où les documents retournés par la nouvelle requête sont les mêmes [Zhang & Seo, 2001]. Seul l'ordre des documents retournés est modifié. Zhang propose également un algorithme d'apprentissage par renforcement pour exploiter un retour de pertinence implicite et ainsi produire une nouvelle requête.

3.2.5 Synthèse sur le RP

Dans cette partie, une liste non exhaustive d'approches utilisant le RP a été présentée. Toutes ces approches se distinguent majoritairement sur la méthode d'acquisition du RP utilisée, implicite ou explicite, et sur le choix d'exploitation de ce RP. Le tableau 3.3 fournit une vue synthétique des différentes approches.

Table 3.3 – Comparaison des approches de RP

	RP implicite	RP explicite
Reformulation de le requête	[Spink et al., 2000, Lin et al., 2006, Gauch & Wang, 1997, Chang et al., 2006]	[Crestani, 1995, Budzik & Hammond, 1999]
Ré-ordonnancement	[Radlinski & Joachims, 2006, Zhang & Seo, 2001]	[Ruthven & Lalmas, 2003, Wen et al., 2001]
Suggestion de requête	[Cao et al., 2008]	[Croft et al., 2001]
Suggestion de documents	[Zhang & Seo, 2001]	[Budzik & Hammond, 1999]

Le RP est une solution utile pour pallier l'incertitude de l'utilisateur lors de la description de son besoin informationnel. Les évaluations des différentes approches existantes, résumées par Ruthven et Lalmas, montrent une certaine stabilité au niveau de la performance et de l'utilité du RP [Ruthven & Lalmas, 2003]. Les différents algorithmes donnent approximativement les mêmes bons résultats. Le RP a été présenté ici dans le contexte de la RI mais il peut aussi s'appliquer sur des ressources non-textuelles. Le domaine de la recherche d'image [Rui et al., 1998, Zhou & Huang, 2003], ou de vidéo [Yan et al., 2003, Li & Zhu, 2013] utilisent également des approches de RP. Cependant, le RP ne peut couvrir

tous les domaines et ne peut pas être employé dans tous les systèmes de RI [Bates, 1990]. Il ne peut être qu'une partie du système et il doit pouvoir s'intégrer et s'adapter à d'autres fonctionnalités. Le RP, qu'il soit implicite ou explicite, repose fortement sur le jugement des documents pertinents et/ou non pertinents. Il est difficile, même pour l'utilisateur, de mesurer la pertinence d'un document d'intérêt par rapport à un autre qui est également d'intérêt. Le système doit également savoir quel niveau de pertinence il doit considérer pour utiliser ou non un document dans ses méthodes de RP. C'est pourquoi les techniques de RP ont tout intérêt à être exploitées dans des systèmes mettant en jeu des utilisateurs maîtrisant les techniques de recherche d'information et capable de mesurer l'importance du RP fait sur les documents qu'ils jugent.

Deuxième partie

Contributions théoriques

CHAPITRE 4

POSITIONNEMENT

Les chapitres précédents ont présenté un ensemble d'approches et de travaux qui ont permis d'améliorer les processus de recherche (voir section 2.1, page 13) et de découverte d'information (voir section 2.2, page 23). La représentation du besoin de l'utilisateur joue un rôle important dans ces processus, tout comme le corpus sur lequel ils travaillent. Les approches existantes permettant de modéliser et de représenter le besoin informationnel ont également été décrites (voir section 3.1, page 43). Le profil utilisateur qui en découle peut être amélioré par des processus de retour de pertinence (voir section 3.2, page 55).

Cependant, ces approches présentent des limites, qui sont introduites en seconde partie de ce chapitre, lorsqu'elles sont associées aux problématiques liées au Renseignement d'Origine Sources Ouvertes, décrites en introduction de ce manuscrit. L'expression du besoin opérationnel sur des sujets spécifiques et/ou sensibles est complexe et les experts du renseignement ont du mal à découvrir de nouvelles sources d'intérêt sur ces sujets. La résolution de ce verrou permettrait d'améliorer les résultats du cycle de veille stratégique. Ainsi, la dernière partie de ce chapitre introduit nos objectifs de recherche et nos axes d'étude sur l'expression du besoin opérationnel et sur la découverte de sources d'information dans le cadre du ROSO.

4.1 Problématiques de la DI pour le ROSO

Dans cette section, les caractéristiques et les spécificités des sources recherchées par les experts du renseignement sont présentées. Nous montrons en quoi ces caractéristiques les rendent difficiles à trouver sur le Web. Cette problématique de DI est renforcée par la spécificité du besoin des experts et de leur tâche de veille stratégique.

4.1.1 Spécificités des sources d'intérêt opérationnel

Les sources recherchées par les experts du renseignement ont un contenu qui peut être **sensible** et/ou **spécifique**. Or, les principaux outils de RI tel que [Mc Gowan, 2003,Chirita et al., 2004,Haveliwala, 2003] et de DI [Chakrabarti et al., 1999b,Chakrabarti et al., 1999a]

considèrent la popularité des pages pour ordonner leurs résultats. Cependant, la popularité n'est pas un gage de pertinence pour les opérationnels du renseignement. Au contraire, les sujets sensibles et les informations spécifiques peuvent être **impopulaires** et donc fournies par des pages très peu fréquentées sur le Web, les rendant difficiles à trouver avec les outils existants. Ainsi, *rechercher une aiguille dans une botte de foin* est une métaphore appropriée à la recherche de ces sources. Elle illustre également la difficulté de trouver des sources spécifiques dans un corpus de grande taille comme le Web. Travailler sur un corpus d'une telle envergure est une problématique partagée par les outils de RI et de DI. En ce sens, ces outils sont complémentaires et non concurrents lorsqu'il s'agit de réussir à fournir de nouvelles sources d'intérêt aux experts du renseignement.

4.1.2 Spécificités de la tâche de veille stratégique

L'expression du besoin pour la veille stratégique se distingue de celle utilisée pour une recherche d'information classique. En effet, une requête ponctuelle spécifique ne permet pas de représenter un domaine de veille. Le besoin est beaucoup plus **complexe** à représenter puisqu'il doit couvrir tout un domaine. De plus, lors de la veille, le besoin de l'expert peut évoluer au fur et à mesure qu'il reçoit et analyse de l'information sur son sujet de recherche. Les opérationnels du renseignement sont également confrontés à **l'évolution** et à la **complexité du vocabulaire** de certains domaines de recherche, ce qui peut limiter leur **faculté d'expression du besoin**. La spécificité de la tâche de veille stratégique réside donc dans la complexité à représenter et modéliser le besoin opérationnel qui sert à la DI.

4.2 Limites de l'état de l'art pour le ROSO

Cette section met en exergue les limites des approches existantes dans le contexte du ROSO et de la recherche de sources spécifiques par les experts du renseignement. Les raisons pour lesquelles les approches de DI existantes sont inadéquates aux besoins des experts du renseignement y sont présentées. De plus, nous décrivons les limites des approches de modélisation du profil utilisateur lorsqu'il s'agit d'un besoin spécifique et/ou sensible.

4.2.1 Limites des approches de RI et de DI

Les approches de RI travaillent sur une représentation du Web qui tend à être la plus complète possible. Il est donc difficile d'y trouver de l'information précise, adaptée au besoin spécifique de l'utilisateur (voir section 2.1.2, page 15). En effet, la masse d'information est difficile à trier et rechercher une information précise est une tâche complexe. Dans le cadre du ROSO, cette problématique est renforcée par la recherche de sources spécifiques qui sont d'autant plus difficile à trouver. De plus, les approches de RI travaillent sur une représentation tronquée du Web puisque la vitesse de création, modification, suppression des pages et des informations ne permet pas d'avoir un corpus à jour en temps réel (voir section 2.1.1, page 14). Aussi, la DI via l'exploration et la collecte ciblée d'information permet de pallier les limites de la RI (voir section 2.2.4, page 29) en constituant un corpus de documents d'intérêt, fraîchement collecté sans prise en compte de la popularité.

Afin de déterminer la pertinence des pages, les approches de RI et de DI exploitent principalement la popularité des pages [Kleinberg, 1999, Page et al., 1999], même si certaines exploitent d'avantage le besoin utilisateur [Asnicar & Tasso, 1997, Micarelli & Sciarrone, 2004, Speretta & Gauch, 2005, Liu et al., 2004]. Cependant, les approches existantes [Chakrabarti et al., 1999b, Bergmark et al., 2002, Pant & Menczer, 2002] ne prennent pas en compte le contexte du ROSO qui nécessite une représentation du besoin adaptée aux spécificités de la tâche de veille stratégique comme l'explique la section suivante.

4.2.2 Limites de la représentation du besoin

Les moteurs de recherche ont démocratisé l'utilisation de requêtes (voir section 2.1.1, page 14) via le format "boite de recherche" qui permet de taper une liste de mots-clés [Salton, 1969]. Ce modèle, qui a fait ses preuves, a également montré une limitation : le résultat de l'appariement requête/document n'est pas nuancé. Avec un tel modèle booléen, le document répond pleinement à la requête ou alors il n'y répond pas du tout (voir section 2.1.2, page 15). Cette représentation est pourtant très utilisée dans les approches de RI personnalisée ou de DI [Gasparetti & Micarelli, 2003, Gasparetti & Micarelli, 2004]. D'autres approches s'appuient sur une représentation plus complète basée sur des listes de mots-clés [Salton, 1971, Pant & Menczer, 2002] ou de concepts pondérés [Ehrig & Maedche, 2003, Chakrabarti et al., 1999a]. Cependant, ils ne combinent pas l'utilisation conjointe des mots-clés et des concepts et il n'existe pas d'approche unique et formalisée permettant la modélisation, la représentation et l'exploitation du besoin en information de l'utilisateur dans le contexte du ROSO. L'exploration ciblée du Web est d'autant plus efficace quand l'expression du besoin est complète et précise. Ainsi, ces approches sont limitées dans le cadre du ROSO par :

- une représentation unique du besoin via une requête qui est insuffisante. Par conséquent, l'amélioration de l'expression du besoin doit passer par une représentation de celui-ci prenant en compte les différents avantages des techniques existantes. Dans le cadre du ROSO, un besoin spécifique peut être difficile à exprimer avec des mots-clés. L'utilisation complémentaire d'ontologies [Ehrig & Maedche, 2003, Chakrabarti et al., 1999a] est envisageable pour une représentation plus conceptuelle et thématique du besoin. L'utilisation conjointe de mots-clés et de concepts peut permettre de faire face à l'évolution et à la complexité du vocabulaire de certains domaines de recherche.
- la linéarité du processus : le processus de recherche, qu'il se fasse au travers de moteurs de recherche (voir section 2.1, page 13), d'outils de RI personnalisée (voir section 2.1.4, page 18) ou de système de DI (voir section 2.2.1, page 25), est souvent réduit à un traitement linéaire "besoin - réponse". Cependant, l'expression du besoin est itératif d'autant qu'il peut évoluer en fonction des réponses retournées par le système. Les experts du renseignement travaillent généralement sur un besoin qui se précise au fur et à mesure de l'actualité et de l'évolution de leur connaissance autour de ce besoin. L'exploitation du retour de pertinence est une méthode particulièrement efficace pour mettre à jour le besoin utilisateur [Spink et al., 2000, Lin et al., 2006, Gauch & Wang, 1997, Chang et al., 2006, Crestani, 1995, Budzik & Hammond, 1999]. Cependant, elle est fortement sous exploitée dans les systèmes de DI et de tels systèmes sont très rarement mis en oeuvre.

4.3 DOWSER, un système de découverte de sources Web d'intérêt opérationnel

L'analyse des approches existantes en matière de RI, DI et modélisation du besoin utilisateur nous a permis de mettre en évidence un certain nombre de limites. Appliquées au contexte du ROSO, ces limites bloquantes laissent la place à des perspectives de recherche afin d'améliorer les systèmes de DI pour le cycle du renseignement. Ces possibles améliorations sont au coeur des objectifs de cette thèse et s'articulent autour de deux principaux axes de recherche : la modélisation du besoin opérationnel et son exploitation dans un système de découverte de nouvelles sources d'information.

4.3.1 Le besoin opérationnel

Cet axe de recherche a pour but de modéliser et représenter le besoin opérationnel des experts du renseignement. Le besoin utilisateur dans le cadre du ROSO est associé à un besoin spécifique en sources d'information non populaires. Ce cas d'utilisation contextuel est présenté dans le chapitre suivant. Une approche de construction implicite du besoin utilisateur, basée sur une couverture terminologique et thématique du besoin, est introduite. Les aspects de construction et de prise en compte du retour de pertinence adaptés à ce profil y seront également introduits.

4.3.2 La découverte de nouvelles sources

L'enjeu de la collecte ciblée dans le contexte du ROSO est le second axe d'étude de cette thèse. Le but est d'exploiter le profil utilisateur dans un système d'exploration ciblée du Web afin de recueillir de nouvelles sources d'informations pertinentes. Une mesure de similarité basée sur une combinaison de critères permettant la prise en compte terminologique et thématique du besoin est introduite dans le chapitre 6. Enfin, le système résultant, capable de s'adapter à l'évolution du besoin utilisateur, fournira des résultats en temps réel. Ce dernier point se fera sans l'aide de moteur de recherche et assurera la présentation, à l'utilisateur, de sources d'intérêt à jour.

4.4 Conclusion

Ce chapitre a mis en avant les limites liées au contexte du ROSO présentes dans les systèmes de RI et de DI décrits dans la première partie de ce manuscrit. La sensation de rechercher une aiguille dans une botte de foin sur le Web couplée à une expression unique du besoin dans un processus linéaire de recherche ou de découverte, limite la production de résultats pertinents. Dans un contexte de ROSO, cette limite est une vrai problématique pour les opérationnels du renseignement qui doivent redoubler d'effort pour couvrir leur besoin en information. Dans les sections suivantes de cette partie, nous présentons les travaux de cette thèse qui visent à répondre aux besoins des experts du renseignement. Les contributions liées à la modélisation du besoin opérationnel sont introduites dans le chapitre 5. L'exploitation de ce besoin, au travers d'un système de découverte de nouvelles sources d'intérêt, fait l'objet du chapitre 6. Enfin, le chapitre 7 présente des expérimentations et des résultats permettant d'évaluer notre approche et le chapitre 8 fournit une description de l'architecture et des processus du système.

CHAPITRE 5

DOWSER, MODÉLISER LE BESOIN UTILISATEUR

La difficulté de modéliser et de représenter le besoin opérationnel des experts du renseignement est une des problématiques qui ont été soulevées dans le chapitre précédent. L'évolution et la complexité du vocabulaire de certains domaines de recherche rendent difficile l'expression du besoin par l'expert. De plus, l'information recherchée peut être diffuse et/ou sensible sur des sujets impopulaires et/ou illégaux. Parfois non-indexées ou désindexées, les pages d'intérêt sur ces sujets sont difficilement trouvables via les moteurs de recherche. Les liens vers ces sites se retrouvent plus facilement sur des blogs ou des forums que les experts doivent d'abord découvrir puis éplucher pour trouver le lien d'intérêt. Une tâche laborieuse qui demande énormément de temps.

Les travaux de cette thèse prennent en compte ces spécificités et visent à proposer une approche de découverte de nouvelles sources d'intérêt pour répondre aux besoins opérationnels dans le cadre du ROSO. Considérant les limites des systèmes de RI et de DI présentés dans le chapitre 4, page 67, ainsi que les problématiques des experts du renseignement, l'approche s'articule autour de deux axes :

- une modélisation du besoin opérationnel qui permet de pallier l'inadéquation de représentativité des requêtes et de couvrir la terminologie et la thématique du domaine de recherche,
- un système de découverte de sources d'intérêt opérationnel capable de considérer le besoin opérationnel afin de collecter des informations fraîches, sans considération de la popularité des sources.

Cette approche est appelée DOWSER pour *Discovery Of Web Sources Evaluating Relevance*. Cet acronyme forme le mot anglais *dowser* qui signifie "sourcier" en français et fait donc référence à la découverte de sources d'eau. C'est un clin d'oeil indirect à la recherche de sources d'information sur le Web dont la problématique, traitée dans cette thèse, est très bien résumée par la citation de Mitchell Kapor :

Getting information of the Internet is like taking a drink from a fire hydrant.

Mitchell Kapor

Nous présentons, dans ce chapitre, nos contributions liées à la modélisation et la construction du besoin opérationnel. Dans un premier temps, des scénarios opérationnels sont présentés afin d'illustrer la difficulté de modélisation du besoin opérationnel. Puis, notre modèle de profil utilisateur adapté au ROSO est présenté ainsi que sa capacité à couvrir la terminologie et la thématique du besoin. L'étape de construction d'un profil est également décrite. Enfin, nous présentons le processus de retour de pertinence utilisé dans notre approche pour affiner et faire évoluer notre profil opérationnel.

5.1 Scénarios opérationnels

Afin de comprendre en quoi les limites énumérées précédemment sont vraiment impactantes pour les experts du renseignement, nous présentons deux scénarios opérationnels qui serviront dans la suite de ce chapitre à illustrer les choix techniques. Ce sont des scénarios inspirés d'opérations réelles menées par les experts du renseignement.

5.1.1 Surveillance de bateau

Ce scénario opérationnel concerne la surveillance d'un bateau que l'on appellera le "Nasty-Boat". Ce navire est connu pour effectuer régulièrement une ligne entre plusieurs continents. Les escales de ce navire laissent penser que son commerce peut dériver de ses produits originaux vers des produits illicites revendus ensuite sur le marché noir.

L'objectif des experts du renseignement est de trouver des informations d'intérêt sur le navire en question, sous la forme de différents types de médias, textes, images du navire et de ses équipements, vidéos, etc. Les informations qu'ils contiennent sont également de plusieurs types. Ils peuvent renseigner sur l'historique du navire (sa provenance, son montant, la structure mise en oeuvre pour son acquisition), son état courant (localisations, marchandises transportées, etc.), les acquisitions d'équipements installés sur le navire, etc.

Le scénario consiste à utiliser ces informations afin de :

- déterminer avec précision la ligne empruntée par le navire ainsi que ses dates d'entrées et de sorties dans les différents ports où il fait escale,
- déterminer les caractéristiques techniques du navire et le type de marchandises transportées,
- récupérer des informations sur l'environnement du bateau (équipage, armateur, etc.).

Ce navire n'est ni célèbre, ni même connu. Aussi, trouver de l'information en sources ouvertes le concernant est délicat, surtout des informations de cet ordre. En effet, il n'a pas de site Web dédié ou de page Wikipedia [1] qui lui sont formellement consacrés. L'information est donc diffuse et partielle, ce qui la rend d'autant plus difficile à trouver. La fraîcheur de ces informations est également primordiale. Par exemple, les experts du renseignement sont bien plus intéressés par la marchandise transportée actuellement que par celle transportée l'an dernier.

Les experts du renseignement ont un ensemble de sources Web déjà identifiées comme étant d'intérêt pour obtenir un minimum d'informations pertinentes. Le site *Port Arrivals* [2] permet par exemple d'avoir la liste des ports par pays et pour chacun d'entre eux, la liste des navires qui ont prévu d'y faire escale. C'est une source d'intérêt pour permettre

1. Encyclopédie collective sur Internet : http://fr.wikipedia.org/
2. http://www.portarrivals.com

d'estimer la ligne empruntée par le "Nasty-Boat". Par contre, les informations concernant les caractéristiques du navire, sa marchandise et son équipage sont des données plus diffuses qu'il faut retrouver au sein d'articles, de forums, etc. Cependant, les experts du renseignement ont exprimé leur difficulté à compléter et à enrichir leurs informations en utilisant les moteurs de recherche. Les limites des systèmes de RI évoquées dans le chapitre précédent sont des obstacles directs à la découverte de nouvelles sources d'intérêt.

5.1.2 Ventes de médicaments illicites

Le but de ce second scénario est de détecter de nouvelles tendances de consommation et des fraudes liées à la vente de médicaments illicites. L'objectif des experts du renseignement est de trouver de nouveaux sites de ventes illicites de produits médicamenteux afin de :

- détecter des produits contrefaits et des médicaments interdits avec ou sans autorisation de mise sur le marché,
- détecter des produits dangereux, comme par exemple les produits à base de corne d'antilope (interdit par la convention de Washington) qui auraient des vertus aphrodisiaques.

Les experts du renseignement connaissent déjà des exemples de sites vendant de tels produits. En découvrir de nouveaux leur permettrait d'établir des tendances sur les produits en vogue et de croiser les informations sur les différents vendeurs.

Ce genre de site étant très surveillé et contrôlé, ils sont difficiles à trouver. Plusieurs raisons peuvent justifier leur absence dans les résultats des moteurs de recherche :

- La requête de l'expert du renseignement peut être imprécise. Le médicament peut être référencé sous un autre nom méconnu de l'expert ou sous un nom de code ayant émergé récemment. Les activités de renseignement touchent des domaines divers qui peuvent avoir des particularités de langage, des termes techniques spécifiques ou d'argot qui peuvent échapper aux experts du renseignement. De nouveaux termes, associés par exemple à de nouveaux produits, peuvent apparaître. Les experts du renseignement doivent faire face à ces contraintes de vocabulaire qui peuvent les empêcher d'exprimer clairement leur besoin.
- La vente de produits illicites est réprimandée sur le Web [3]. Un tribunal peut demander la fermeture d'un site mais cela se fait au vue des lois en vigueur dans le pays hébergeant ce dernier. Un site à contenu illicite a tendance à bien choisir son hébergeur. La fermeture d'un site est donc difficile, mais les tribunaux peuvent aussi demander aux moteurs de recherche de le mettre sur liste noire (black-lister). Le black-listage annonce la fin d'une partie du trafic d'un site. Les moteurs de recherche étant unanimement utilisés par les utilisateurs d'Internet, être mis sur liste noire est bien plus contraignant pour ces sites que d'avoir une amende prononcée par un tribunal ou par une administration.
- Le Web évolue plus vite que la justice, si bien que lorsqu'un site ferme, il est aussitôt remplacé par un autre avec un titre, un contenu et une adresse Web quasiment identiques. Ces *doubles* ne peuvent pas être automatiquement fermés. Ils nécessitent une nouvelle décision d'un tribunal. S'ils ne sont pas présents dans les résultats des moteurs de recherche c'est qu'ils n'ont pas encore été découverts par les robots d'exploration de ces moteurs de recherche.

3. http://www.pcinpact.com/news/80893-fermeture-plus-9-000-sites-lies-a-commerce-illegal-medicaments.htm

– Les sites Web recherchés ne souhaitent pas être indexés par les moteurs de recherche. Ils peuvent le spécifier dans un fichier *robots.txt* propre à chaque site Web. L'adresse du site n'est alors connu que par *le bouche à oreille* via des forums de discussions, des blogs, etc.

5.2 L'expression du besoin dans DOWSER

Les scénarios pris en exemple illustrent les spécificités des recherches menées par les experts du renseignement ainsi que la complexité de la modélisation du besoin informationnel. Les systèmes de RI et DI présentés dans notre état de l'art mettent en avant l'utilisation d'un profil utilisateur dans des approches de recherche personnalisée ou de collecte ciblée de l'information. Considérant les spécificités liées au sujets de recherche des experts du renseignement, nous définissons dans cette section un profil utilisateur adapté au problème du ROSO. Ce profil se veut être un profil cognitif, basé sur une représentation ensembliste thématique et terminologique, et construit de manière implicite. Ces choix sont justifiés ci-après.

5.2.1 Un type de profil adapté au ROSO

Un profil peut se définir comme étant un ensemble de caractéristiques d'une personne. Sous cette forme, le terme *profil* peut donc avoir une connotation négative pour les experts du renseignement qui souhaitent être le plus discrets possible et conserver leur anonymat. En effet, capitaliser et stocker des informations personnelles peut être dangereux surtout si elles peuvent être corrélées à des informations professionnelles sur les travaux qu'ils mènent. Les experts doivent donc faire preuve de discrétion sur Internet mais également au sein même de leur bureau. Un expert du renseignement peut être tenu de garder son activité secrète même auprès de ses collègues. Le profil employé dans l'approche DOWSER se doit d'être assez restrictif pour respecter ces contraintes. Dans la section 3.1.1 page 43, un ensemble de profils a été présenté parmi lesquels le profil cognitif, le profil qualitatif et le profil multidimensionnel.

a. Inadéquation au ROSO des profils multidimensionnel et qualitatif

Le profil multidimensionnel regroupe plusieurs dimensions dont les données personnelles et les données de sécurité. Les données personnelles qui ont pour but de caractériser l'utilisateur constituent une dimension non retenue dans notre profil pour les raisons d'anonymat évoquées précédemment. De plus, la dimension sécurité n'est pas nécessaire dans notre profil dans la mesure où elle fait partie intégrante du travail des experts. Nous avons donc écarté le profil multidimensionnel pour nos travaux aux vues des dimensions incompatibles avec l'environnement de travail des experts du renseignement. Le profil qualitatif contient des données sur la crédibilité des sources d'informations, sur la cohérence et la fraîcheur des informations désirées. Y. Mombrun [Mombrun et al., 2010] montre que la tâche d'évaluation de l'information est complexe et risquée, en particulier dans le contexte du ROSO. En effet, la crédibilité associée à une information peut déboucher sur la prise de décisions importantes. La difficulté et l'importance de cette tâche sont telles qu'elle correspond à une phase à part entière dans le cycle du renseignement, présenté page 18. Nos travaux concernent la phase de collecte, qui se situent en amont de cette tâche. Ainsi,

le profil qualitatif est également écarté de notre approche pour ne pas interférer avec la phase d'évaluation et d'analyse opérée par les experts du renseignement.

b. Le choix du profil cognitif

Le profil cognitif, présenté dans la section 3.1.1 page 43, est un profil basé sur les connaissances de l'utilisateur et/ou sur ses centres d'intérêt. Il permet notamment de couvrir les besoins long terme propre à la veille thématique mais aussi de couvrir le besoin court terme. Ce type de profil est tout à fait adapté à notre approche puisqu'il permet de s'intéresser uniquement à l'expression du besoin sans données personnelles sur l'utilisateur ou sur la qualité des sources recherchées. Les scénarios présentés dans la section 5.1 montrent que le besoin des opérationnels du renseignement est de trouver de nouvelles sources d'information sachant qu'ils ont déjà des sources d'intérêt sur leur thématique de recherche P_{si}. Ces sources d'intérêt représentent la connaissance de l'utilisateur et elles sont utilisées dans notre approche pour construire les centres d'intérêt P_{ci} du profil opérationnel P (voir définition 1).

Définition 1 (Le profil opérationnel)**.**

Soit P, un profil opérationnel contenant un ensemble de sources d'intérêt P_{si} et de centres d'intérêt P_{ci} représentatifs du besoin :

$$P = \{P_{si}, P_{ci}\}$$

5.2.2 Construction du profil opérationnel

La construction du profil peut se faire de manière implicite ou explicite (voir section 3.1.3, page 49). Nous avons évoqué, dans ce chapitre, l'évolution rapide du vocabulaire lié aux sujets de recherche des experts du renseignement. Ces experts ne peuvent pas connaître l'ensemble des termes pertinents correspondant à leurs besoins. Aussi, leur demander d'écrire une requête représentant leur besoin informationnel ne fournirait qu'une expression trop restrictive de ce besoin. Cependant, ces termes d'intérêt se trouvent dans l'ensemble des sources d'intérêt qu'ils ont déjà et qui représente leur connaissance sur un sujet donné. Dans notre approche, l'expert fournit les URLs de ces sources et le système se charge de modéliser le besoin en les exploitant. Ce choix de construction de profil permet :

- de couvrir l'ensemble du vocabulaire sur un sujet donné et de faire face à son évolution,
- de ne pas être intrusif en demandant explicitement à l'expert ses sources d'intérêt lié à un besoin en information,
- de modéliser le besoin automatiquement, à partir des sources d'intérêt fournies, sans demander d'autres efforts de la part de l'expert.

Notre construction du profil est donc semi-implicite, comme l'illustre la figure 5.1, page 76. La première étape de la construction permet de renseigner la connaissance de l'utilisateur dans le profil cognitif via l'explicitation de ses sources d'intérêt. La seconde construit les centres d'intérêt de l'utilisateur automatiquement à partir des sources d'intérêt fournies.

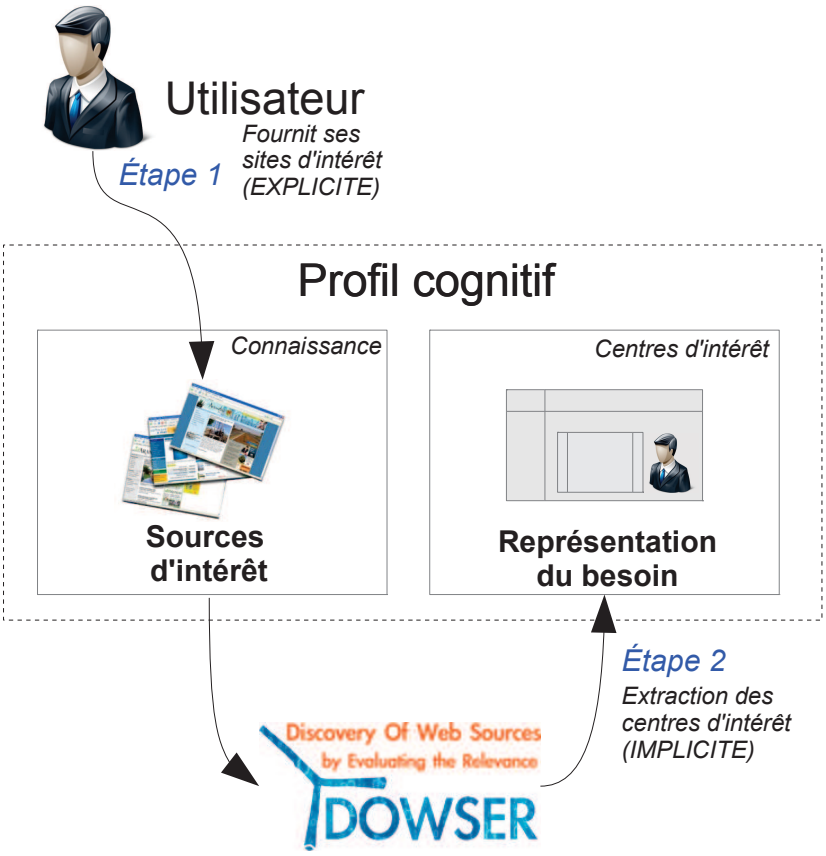


FIGURE 5.1 – Construction du profil dans DOWSER

5.2.3 Représentation ensembliste avec mots-clés et concepts

Le besoin opérationnel des experts du renseignement est de découvrir de nouvelles sources d'intérêt sur un sujet spécifique. Ils travaillent sur des thématiques assez larges dans lesquels ils recherchent des informations précises. Ainsi, le besoin associé à chacun des deux exemples de scénarios opérationnels introduits page 72 peut se définir plus ou moins globalement et avec plus ou moins de précision :

- de manière grossière, le second scénario consiste à rechercher des sites de ventes de produits médicamenteux illégaux,
- de manière plus détaillée, le scénario a pour but de rechercher des sites de ventes d'aphrodisiaques à base de corne d'antilope.

Dans notre approche, la représentation des centres d'intérêt de l'utilisateur doit pouvoir englober ces deux niveaux de détails pour répondre au mieux au besoin en information. Le profil doit modéliser les centres d'intérêt à un niveau thématique mais aussi à un niveau beaucoup plus précis. Pour ce faire, DOWSER utilise une représentation ensembliste composée de concepts pour couvrir la thématique de recherche $\vec{P_C}$ et une représentation ensembliste à base de mots-clés pour couvrir la terminologie du sujet $\vec{P_K}$ (voir définition 2).

Définition 2 (Les centres d'intérêt du profil opérationnel)**.**

Soit P_{ci}, un profil opérationnel contenant des centres d'intérêt représentés par $\vec{P_C}$ un vecteur composé d'instances de concept et $\vec{P_K}$ un vecteur composé de mots-clés.

$$P_{ci} = \{\vec{P_C}, \vec{P_K}\}$$

a. Couverture thématique du besoin opérationnel

La couverture de la thématique du besoin peut être associée à la vision d'un profil long terme puisqu'elle permet de représenter les centres d'intérêt généraux, persistants ou récurrents. Parmi les différentes approches de représentation présentées dans le chapitre 3, page 43, notre choix s'est porté sur la représentation conceptuelle pour couvrir le besoin thématique. Elle a l'avantage de fournir une structuration et des relations de corrélation entre les différents centres d'intérêt permettant de lier et de couvrir en largeur la thématique globale. De plus, cette couverture de la thématique est assurée par l'utilisation d'une ontologie qui couvre un vaste éventail de domaines. Ainsi, nous avons choisi la base de connaissances DBpedia [4] qui contient un ensemble de données structurées dérivé de Wikipedia [5]. Parmi les avantages de cette base de connaissances par rapport aux autres existantes [Bizer et al., 2009], on notera que DBPedia :

- contient 4 millions d'entités dont 3,22 millions sont classées dans une ontologie cohérente,
- évolue automatiquement avec les mises à jour de Wikipedia,
- fait partie du LOD [6] et est ainsi interconnectée avec de très nombreux dépôts du Web de données,
- est multilangue et accessible sur le Web.

De plus, plus de 800 000 catégories permettent de classer les ressources DBPedia. Ces catégories sont des concepts décrits dans le langage formel SKOS [7] et reliés par des relations du schéma de métadonnées générique DCMI terms [8]. Ils forment une hiérarchie où une sous-catégorie peut appartenir à plusieurs catégories parentes.

Afin d'identifier les instances de concept DBPedia représentatives du besoin de l'utilisateur, nous utilisons l'outil DBPedia Spotlight [9]. Cet outil de traitement automatique de la langue permet d'analyser un contenu textuel et d'en extraire des entités sémantiques présentes dans la base de connaissance DBpedia. Les résultats fournis par cet outil en matière de précision, de rappel et de désambiguïsation sont à la hauteur de ses concurrents connus dans la littérature comme Zemanta [10] [Mendes et al., 2011] . Notre choix s'est porté sur DBPedia Spotlight aussi pour des raisons techniques : il est sous licence GPL (Licence Publique Générale) ce qui permet d'avoir sa propre installation de l'outil et sans limite d'utilisation.

4. http://dbpedia.org/About
5. http://www.wikipedia.org/
6. Linked Open Data : http://linkeddata.org/
7. Simple Knowledge Organization System : http://www.w3.org/2004/02/skos/
8. Dublin Core Metadata Initiative : http://www.dublincore.org/documents/dcmi-terms/
9. https ://github.com/dbpedia-spotlight/
10. http://www.zemanta.com

Pour chaque besoin opérationnel, les textes issus des sources d'intérêt fournis par l'expert sont analysés par l'outil DBPedia Spotlight afin de modéliser la représentation conceptuelle (voir définition 3). Les concepts extraits sont ajoutés dans un vecteur pondéré. Cette pondération correspond à la fréquence d'apparition des instances de concept dans l'ensemble des sources d'intérêt. Autrement dit, plus une instance de concept est présent dans les sources de l'utilisateur et plus il représentera le besoin thématique. Cette représentation conceptuelle permet de construire une partie du profil cognitif en fournissant des informations sur les centres d'intérêt de l'expert.

Définition 3 (La couverture thématique du profil opérationnel)**.**

Soit $\vec{P_C}$, un vecteur pondéré de taille n composé d'instances de concept C_i.

$$\vec{P_C} = ((C_1, W_{C_1}), ..., (C_n, W_{C_n}))$$

tel que $C_i \in \mathscr{C}$ l'ensemble des instances de concept,
W_{C_i} est le poids de l'instance C_i avec $W_{C_i} \in \mathbb{Z}$, $0 \leqslant W_{C_i} \leqslant 1$,
et i ∈ [1,n].

L'exemple 1 illustre notre couverture thématique pour un besoin opérationnel sur la vente de médicaments à base de produits illégaux.

Exemple 1 (La couverture thématique du profil opérationnel)**.**

$\vec{P_C}$ = ((Traditional-Chinese-Medecine,0.3),(Herbalism,0.25),(Health,0.2),
(Pain,0.15),(China,0.1))

b. Couverture terminologique du besoin opérationnel

En plus de la thématique, notre profil doit également contenir des informations plus précises afin de couvrir l'ensemble du besoin. La couverture terminologique a pour but de modéliser des détails, des informations plus spécifiques et/ou moins connues.

Par exemple, le "Nasty-Boat", qui est le nom du bateau surveillé dans notre premier exemple de scénario opérationnel (cf. page 72), n'est pas un bateau connu et son nom n'est donc pas célèbre. Il sera donc difficile de le représenter conceptuellement puisqu'il ne sera pas présent dans la base de connaissances DBPedia. Il est possible d'utiliser d'autres bases de connaissances comme des bases créées par les experts contenant des concepts représentatifs de leur sujet de recherche. Cependant, la couverture du besoin a également pour vocation de représenter les centres d'intérêt au travers de termes d'argot, ou des nouveaux termes non connus par les experts comme des noms de nouveaux médicaments (voir le second scénario page 72). Afin de couvrir au mieux la terminologie du sujet de recherche, cette représentation ne peut pas se baser sur une base de connaissances.

Ainsi, dans notre approche, une représentation ensembliste de mots-clés est également utilisée. Cette représentation, présentée dans section 3.1.2, page 45, permet de traduire

les centres d'intérêt de l'utilisateur sous forme de vecteur de termes pondérés. Ces termes sont les mots-clés les plus fréquemment présents dans les sources d'intérêt fournies par l'utilisateur. Ils sont extraits à l'aide de l'outil Apache Lucene[11] et de son plugin NP Chunker basé sur l'approche Noun Phrase Chunker [Ramshaw & Marcus, 1995]. Cet outil est une application open source gratuite permettant la recherche plein texte et l'analyse de contenu textuel. Le poids des termes est déterminé par la méthode de pondération TF-IDF, présentée dans la section 3.1.2, page 45. La représentation terminologique (voir définition 4), modélisé par ce vecteur de termes pondérés $\vec{P_K}$, vient compléter la représentation thématique pour couvrir l'ensemble du besoin opérationnel.

Définition 4 (La couverture terminologique du profil opérationnel)**.**

Soit $\vec{P_K}$, un vecteur pondéré de taille m composé de mots-clés.

$$\vec{P_K} = ((K_1, W_{K_1}), ..., (K_m, W_{K_m}))$$

tel que $K_j \in \mathscr{K}$ l'ensemble des mots-clés,
W_{K_j} est le poids du mot-clé K_j avec $W_{K_j} \in \mathbb{Z}$, $0 \leqslant W_{K_j} \leqslant 1$,
et j ∈ [1,m].

L'exemple 2 illustre notre couverture terminologique pour un besoin opérationnel sur la vente de médicaments à base de produits illégaux.

Exemple 2 (La couverture terminologique du profil opérationnel)**.**

$\vec{P_K}$ = ((TCM,0.32),(antelope horn,0.12),(yang jiao,0.12),(health,0.09), (medicine,0.08),(chinese,0.07),(medecine shop,0.06),(herbs,0.06),(cancer,0.04), (spams-stopping,0.04))

c. Taille et normalisation

Les outils d'extraction utilisés dans DOWSER permettent de constituer les deux vecteurs représentatifs du besoin opérationnel. La taille de ces vecteurs est variable puisqu'elle dépend du nombre de termes et de concepts trouvés par les outils d'extraction dans les sources d'intérêt de l'expert. Dans la littérature, la taille des vecteurs utilisés pour modéliser le besoin de l'utilisateur est définie en appliquant un seuil permettant de limiter le nombre de termes :

- soit en sélectionnant les N termes de poids les plus forts,
- soit en supprimant les termes dont le poids est inférieur à un seuil empirique.

Un nombre trop important de termes peut apporter du bruit dans la description du besoin. À l'inverse, un nombre insuffisant de termes ne permettra pas de couvrir l'ensemble du besoin. Le seuil est à fixer en fonction de l'approche employée. Ainsi, la taille optimale du vecteur de concepts peut être différente de celle du vecteurs mots-clés. De plus, couvrir un

11. http://lucene.apache.org/core/

besoin thématiquement demande à priori moins de termes que pour couvrir la terminologie de ce besoin. Notre profil opérationnel contient donc deux tailles de vecteurs différentes où n est la taille du vecteur thématique et m celle du vecteur terminologique.

Le poids d'une instance de concept ou d'un mot-clé est défini en fonction de sa fréquence d'apparition dans l'ensemble des sources d'intérêt et par rapport à l'ensemble des éléments extraits. Une fois le vecteur limité aux n ou m termes les plus pertinents, il est ensuite normalisé. Le but est d'avoir un poids dont la valeur est représentative de la pertinence du terme. La normalisation du vecteur (voir définition 5) permet d'avoir des poids compris entre 0 et 1.

Définition 5 (Normalisation des vecteurs du profil opérationnel)**.**

$$\sum_{i=1}^{n} W_{C_i} = 1 \; et \; \sum_{j=1}^{m} W_{K_j} = 1$$

où W_{C_i} est le poids du i^{ème} concept dans le vecteur thématique et W_{K_j} le poids du j^{ème} mot-clé dans le vecteur terminologique.

5.2.4 Synthèse du profil DOWSER

Le profil opérationnel dans notre approche DOWSER est donc composé de deux vecteurs pondérés permettant de couvrir la thématique et la terminologie du besoin. Le premier contient des instances de concept extraits des sources d'intérêt de l'expert. Les concepts sont issus de la base de connaissance DBPedia et les instances sont extraites à l'aide de l'outil DBPedia Spotlight. Le second vecteur contient des mots-clés extraits par l'outil Apache Lucene après analyse des sources d'intérêt de l'utilisateur. Le schéma 5.2 résume la construction du profil opérationnel. Un exemple de profil opérationnel, basé sur le scénario de vente de médicaments à base de produits illégaux (voir le second scénario page 72) y est illustré. Les sources d'intérêt fournies par l'expert sont des sites vendant des produits et des herbes de la médecine traditionnelle chinoise. On y trouve également deux sites ventant les bienfaits de l'utilisation de la corne d'antilope en tant que médicament. La thématique générale est bien représentée par les instances de concept : les thèmes de la médecine chinoise, des herbes médicales et de la santé y figurent. Les mots-clés couvrent également le besoin en le précisant davantage avec l'intérêt porté sur la corne d'antilope (*antelope horn*, *yang jiao*) et sur l'achat de médicaments (*health product, medecine shop*). Ce scénario est décrit en détail dans l'annexe A, page 179.

Notre approche permet de prendre en compte les limitations évoquées dans la présentation de scénarios opérationnels, page 72, dans notre modélisation du profil opérationnel. En pondérant les termes, la modélisation vectorielle utilisée permet de pallier le manque de représentativité des requêtes. L'impossibilité pour l'expert de connaître l'ensemble du vocabulaire du domaine de recherche est compensée par une extraction implicite de termes dans ses sources d'intérêt. L'exploitation de ces sources permet de construire le profil de manière non intrusive. Cet avantage peut également être considéré comme un inconvénient dans la mesure où l'expert n'a plus la main sur la représentation de son besoin. La partie suivante, propose d'utiliser des techniques de retour de pertinence afin d'affiner la représentation du besoin opérationnel dans DOWSER.

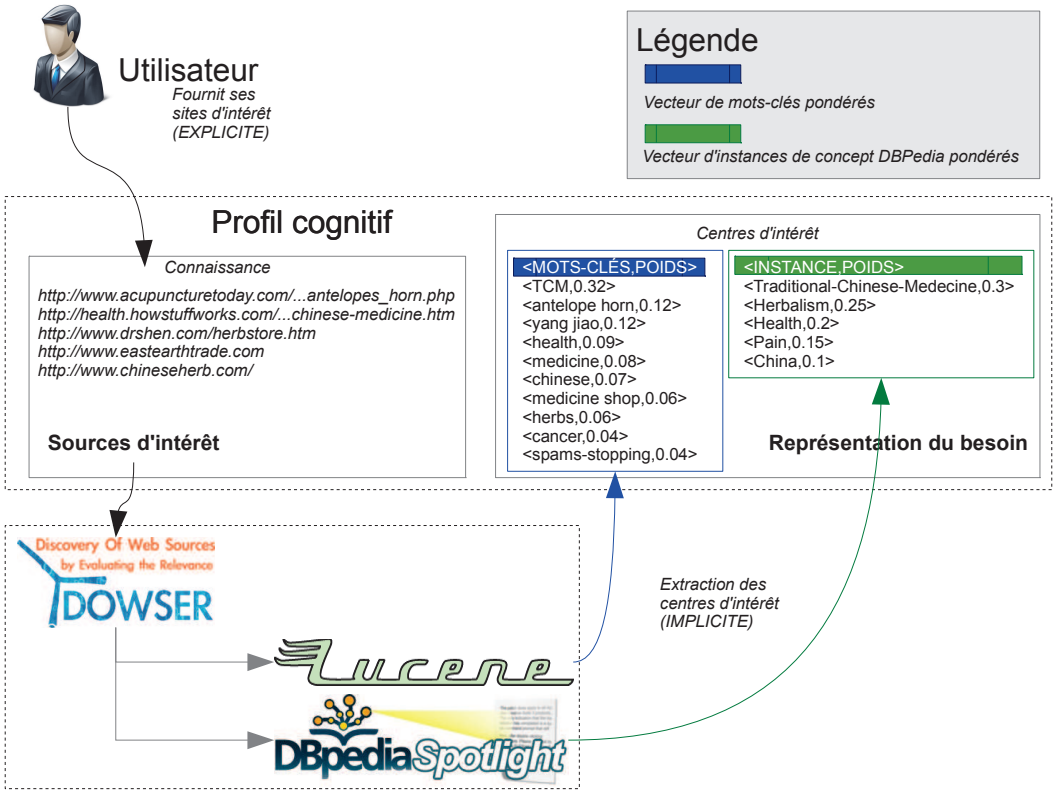


FIGURE 5.2 – Cas d'utilisation : construction du profil dans DOWSER

5.3 Prise en compte du retour de pertinence

La représentation du besoin opérationnel dans DOWSER est basée sur les sources d'intérêt fournies par l'utilisateur. L'extraction des mots-clés et des concepts représentatifs du besoin se fait de manière implicite afin d'alléger le travail de l'expert. Comme évoqué dans la partie précédente, l'inconvénient de cette approche est que la pertinence des termes extraits n'est pas vérifiée par l'expert et repose donc sur la fiabilité des outils utilisés (DBPedia Spotlight et Apache Lucene). La représentation du besoin peut cependant être affinée en améliorant la pondération des termes en remplaçant ceux qui ne sont pas pertinents.

DOWSER repose sur deux axes : la modélisation du besoin opérationnel et un système de découverte de sources. Ce second axe a pour but d'exploiter le profil utilisateur afin de retourner des sites Web jugés comme pertinents. L'utilisateur peut alors confirmer ou infirmer que les pages retournées par le système (voir définition 6, page 82) sont pertinentes. Les pages jugées et leur description ($\vec{D_C}$ et $\vec{D_K}$) sont utilisées pour améliorer le profil et combler les potentielles imprécisions de représentation du besoin dues à la construction implicite. Dans cette section, nous présentons comment le retour de pertinence (voir définition 7, page 82) sur les pages fournies à l'utilisateur, qu'il soit positif R^P ou négatif $R^N P$, impacte le profil opérationnel.

Définition 6 (Description des pages retournées par DOWSER)**.**

Le contenu d'une page est décrite au travers d'un vecteur de mots-clés $\vec{D_K}$ et d'un vecteur d'instances de concept $\vec{D_C}$.

$$\vec{D_C} = ((C_1, W_{C_1}), ..., (C_x, W_{C_x}))$$

tel que $C_i \in \mathscr{K}$,
W_{C_i} est le poids d'instance C_i avec $W_{C_i} \in \mathbb{Z}$, $0 \leqslant W_{C_i} \leqslant 1$,
i ∈ [1,x] ; et

$$\vec{D_K} = ((K_1, W_{K_1}), ..., (K_y, W_{K_y}))$$

tel que $K_j \in \mathscr{K}$ l'ensemble des mots-clés,
W_{K_j} est le poids du mot-clé K_j avec $W_{K_j} \in \mathbb{Z}$, $0 \leqslant W_{K_j} \leqslant 1$,
et j ∈ [1,y].

L'exemple 3 illustre les vecteurs thématique $\vec{D_C}$ et terminologique $\vec{D_K}$ permettant de décrire le contenu d'une page vendant des médicaments.

Exemple 3 (Exemple de description d'une page retournée par DOWSER)**.**

$\vec{D_C}$ = ((Traditional-Chinese-Medecine,0.4),(Health,0.2),(China,0.15),
(Pain,0.15),(Herbalism,0.1))

$\vec{D_K}$ = ((TCM,0.32),(yang jiao,0.16),(antelope horn,0.16),(health,0.08),
(herbal-formula,0.08),(chinese,0.07),(herbs,0.06),(cancer,0.03),
(spams-stopping,0.02),(CANTICER, 0.02)

Définition 7 (Le retour de pertinence)**.**

Soit R, un retour de pertinence composé de retours positifs R^P et de retours négatifs R^{NP}.

$$R = \{R^P, R^{NP}\}$$

tel que

$$R^P = \{D_1^P, ..., D_e^P\}$$

où D_i^P est un document jugé pertinent avec i ∈ [1,e] et

$$R^{NP} = \{D_1^{NP}, ..., D_f^{NP}\}$$

où D_j^{NP} est un document jugé pertinent avec j ∈ [1,f]

5.3.1 Affiner la représentation terminologique

La couverture terminologique du besoin est modélisée par une représentation ensembliste de mots clés. Comme vu dans le chapitre 3.2, page 55, l'approche Rocchio permet d'améliorer la représentation du besoin modélisée sous forme de requêtes [Rocchio, 1971]. Cette approche est adéquate pour affiner notre représentation du besoin d'autant que Salton et Buckley l'ont adaptée à la représentation vectorielle normée [Salton & Buckley, 1997]. Le retour de pertinence permet d'obtenir une liste de sources pertinentes et une liste de sources non pertinentes construites d'après les jugements de l'utilisateur. Nous utilisons Apache Lucene et la méthode de pondération TF*IDF sur ces listes comme nous l'avons fait précédemment pour obtenir le vecteur ensembliste avec les sources d'intérêt de l'expert (voir section 5.2.2, page 75). Deux vecteurs de mots-clés pondérés sont alors produits : le premier contient les termes de forte occurrence partagés par les pages Web jugées pertinentes, et l'autre ceux des pages Web jugées non pertinentes.

Pour rappel, la formule de Rocchio adaptée par Salton et Buckley est la suivante :

$$q_{t+1} = \alpha q_t + \frac{\beta}{|S_r|} \sum_{d_k \in S_r} d_k - \frac{\gamma}{|S_{\bar{r}}|} \sum_{d_k \in S_{\bar{r}}} d_k$$

où α, β et γ sont des termes pondérateurs compris entre 0 et 1, et d_k est la représentation vectorielle de la $k^{ème}$ ressource (voir section 3.2.2, page 57). Dans notre cas, la requête q est remplacée par notre vecteur ensembliste que l'on notera K. Les deux vecteurs normalisés contenant les mots-clés pondérés extraits de l'ensemble des sources jugées pertinentes $R^{P,t}$ et non pertinentes $R^{PN,t}$, à l'itération t, sont respectivement notés $\vec{K}^{P,t}$ et $\vec{K}^{NP,t}$àà (voir définition 8). La construction de ces deux vecteurs est équivalente à celle du vecteur terminologique de notre profil opérationnel (voir section 5.2.3, 76). Ainsi, la taille maximale de ces vecteurs est la même que celle de notre vecteur terminologique et les poids des mots-clés sont également normalisés (voir définition 8).

Définition 8 (Le retour de pertinence pour la couverture terminologique)**.**

$$\vec{P}_K^{t+1} = \alpha \vec{P}_K^{t} + \beta \vec{K}^{P,t} - \gamma \vec{K}^{NP,t}$$

où $\vec{P}_K^{t+1}$ est le vecteur terminologique du profil à l'itération t+1, $\vec{K}^{P,t}$ et $\vec{K}^{NP,t}$ sont respectivement les vecteurs de mots-clés du retour de pertinence R^t, avec

$$\vec{K}^{P,t} = ((K_1^P, W_{K_1^P}), ..., (K_u^P, W_{K_u^P}))$$

où $K_i^P \in \mathscr{K}$,
$W_{K_i^P}$ est le poids du mot-clé K_i^P avec $W_{K_i^P} \in \mathbb{Z}$, $0 \leqslant W_{K_i^P} \leqslant 1$,
et i ∈ [1,u].

$$\vec{K}^{NP,t} = ((K_1^{NP}, W_{K_1^{NP}}), ..., (K_o^{NP}, W_{K_o^{NP}}))$$

où $K_j^{NP} \in \mathscr{K}$,
$W_{K_j^{NP}}$ est le poids du mot-clé K_j^{NP} avec $W_{K_j^{NP}} \in \mathbb{Z}$, $0 \leqslant W_{K_j^{NP}} \leqslant 1$,
et j ∈ [1,o]

L'exemple 4 illustre un retour de pertinence pertinent $\vec{K}^{P,t}$ et non pertinent $\vec{K}^{NP,t}$ pour la couverture terminologique.

Exemple 4 (Exemple de retour de pertinence terminologique)**.**

$\vec{K}^{P,t}$ = ((TCM,0.32),(yang jiao,0.26),(antelope horn,0.26),(health,0.16))

$\vec{K}^{NP,t}$ = ((CANTICER, 0.33),(cancer,0.23),(herbal-formula,0.16),(herbs,0.14), (spams-stopping,0.14))

Les valeurs des termes pondérateurs α, β et γ suivent les recommandations des travaux de Zhai [Zhai & Lafferty, 2001]. Il montre que l'importance accordé au retour de pertinence ne doit pas dépasser celle de la représentation d'origine du besoin mais s'en rapprocher. Cependant, dans DOWSER, nous accordons d'autant plus d'importance au vecteur d'origine qu'il est basé sur la connaissance propre de l'utilisateur : ses sources d'intérêt. L'expression du besoin ne souffre pas des lacunes de représentativité d'une requête et du potentiel manque de vocabulaire de l'utilisateur dans son domaine de recherche. Notre représentation du besoin peut cependant manquer de pertinence dû aux outils employés et à sa construction implicite. Ce dernier point justifie l'utilisation de méthodes de retour de pertinence dans notre approche mais en considérant davantage notre vecteur de départ. On a alors :

$$\alpha > \beta + \gamma$$

Il faut aussi distinguer les valeurs données au β et au γ, c'est-à-dire définir l'importance accordée au retour de pertinence positif par rapport au retour de pertinence négatif. Comme évoqué dans le chapitre 3.2, page 55, le retour de pertinence négatif est difficile à prendre en compte dans la mesure où juger un document comme étant négatif n'explicite pas la raison de ce jugement. Si une source est jugée non pertinente, cela peut être dû à un contenu :

- hors-sujet
- qui correspond au sujet mais qui est jugé faux par l'utilisateur
- qui correspond au sujet mais qui n'est pas à jour
- qui correspond au sujet mais que l'utilisateur connaît déjà
- etc.

Contrairement à un jugement positif qui est sans équivoque, il faut considérer le retour négatif avec beaucoup plus de précaution. Aussi, son importance doit être plus faible, ou au maximum égale au retour positif. On a alors :

$$\beta \geq \gamma \; et \; \alpha \gg \gamma$$

La prise en compte du retour de pertinence va permettre d'affiner notre représentation terminologique du besoin opérationnel. Pour l'illustrer, un exemple de retour de pertinence est schématisé dans la figure 5.3, page 85. Il est basé sur le scénario de vente de médicaments à base de produits illégaux (voir le second scénario page 72). Ce schéma illustre les différentes étapes de la prise en compte du retour de pertinence : de la construction du nouveau vecteur à sa normalisation. Dans cet exemple, les retours de pertinence permettent d'augmenter notamment le poids des termes d'intérêt *antelope horn* et *yang jiao* et de baisser le terme *cancer* qui n'est pas un terme d'intérêt à la vue du scénario de recherche. La représentation terminologique du besoin est ainsi affinée. Ce scénario est décrit en détail dans l'annexe A, page 179. Le retour de pertinence peut également être

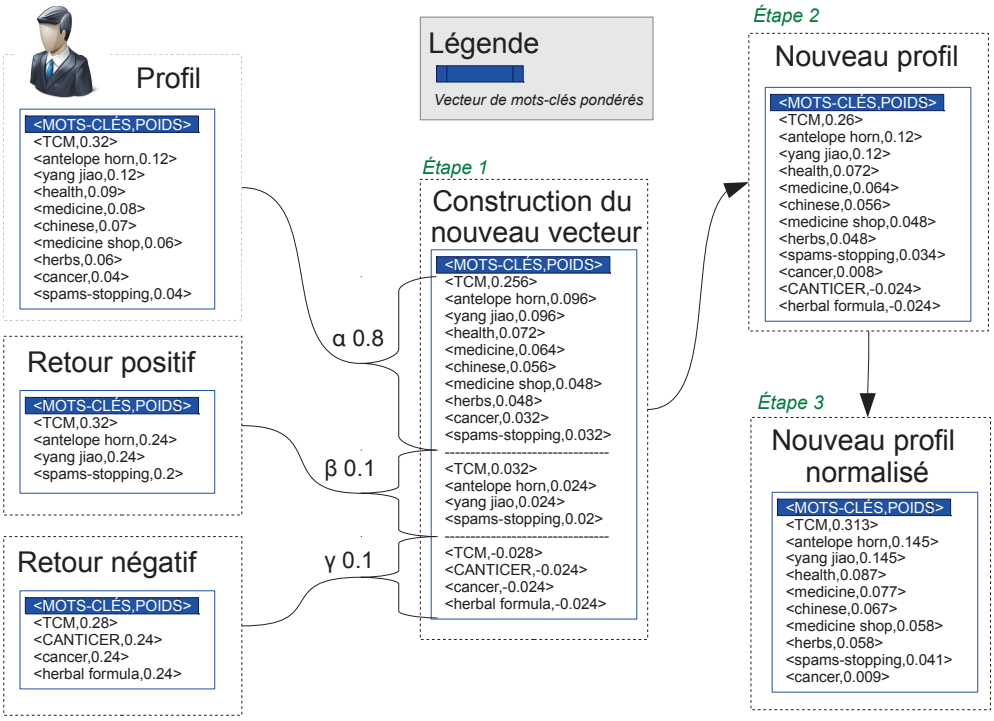


FIGURE 5.3 – Exploitation du retour de pertinence sur le vecteur terminologique

exploité pour améliorer la représentation thématique du besoin, comme nous allons le voir dans la section suivante.

5.3.2 Affiner la représentation thématique

La liste de sources découvertes qui sont jugées pertinentes et celles qui sont jugées non pertinentes permettent d'obtenir deux vecteurs de mots clés mais également deux vecteurs de concepts. En effet, DBPedia Spotlight peut être utilisé sur ces deux listes de sources afin d'extraire les instances de concept relatifs aux pages pertinentes et aux non pertinentes. L'utilisation du retour de pertinence se justifie comme précédemment : il permet d'améliorer la représentation du besoin et de combler les lacunes d'extraction liées à l'outil ou la construction automatique du profil. Cependant, la formule adaptée de Rocchio, utilisée précédemment, ne peut s'appliquer aussi directement avec un vecteur de concepts étant donné les liens hiérarchiques entre ces concepts et la question que cela soulève : Doit-on prendre en considération les liens hiérarchiques des concepts pour améliorer le retour de pertinence et ainsi améliorer la représentation du besoin ? Afin de répondre à cette question, la suite de cette partie examine les mécanismes d'exploitation du retour de pertinence sur des vecteurs de concepts en considérant des liens hiérarchiques.

a. Mécanismes d'exploitation du RP

Chaque instance de concepts DBPedia appartient à une ou plusieurs catégories (voir section 5.2.3, page 76). Deux instances de concept sont donc plus ou moins proches en fonction des catégories qu'elles partagent. Dans la suite de ce manuscrit (voir section 6.3.2, page 106), nous présenterons une mesure de similarité thématique entre le profil et les instances de concept extraites d'une page Web. Il n'est pas nécessaire de décrire cette mesure dans cette partie mais il faut retenir qu'elle utilise le poids W des instances de concept présentes dans le profil mais également les catégories de concepts, pour calculer la distance qui séparent deux instances. Ainsi, lorsque le profil est mis à jour, les poids W des instances de concept du profil augmentent ou diminuent ce qui entraîne respectivement une augmentation ou une diminution du score de similarité thématique $\mathrm{Sim}_c(\vec{P}_C,\vec{D}_C)$ entre le profil P et un document D contenant les instances dont le poids a changé. On notera W_A le poids de l'instance A dans le profil et $\vec{D}_A$ un un vecteur d'instances du document D contenant l'instance A.

$$\text{Si } W_A^{t+1} > W_A^t \text{ alors } Sim_C(\vec{P}_C, \vec{D}_A) \nearrow$$

$$\text{et inversement si } W_A^{t+1} < W_A^t \; alors \; Sim_C(\vec{P}_C, \vec{D}_A) \searrow$$

La mise à jour du profil influence grandement les scores de similarité. Il est donc nécessaire de voir si l'exploitation du retour de pertinence doit également utiliser les informations hiérarchiques et de classification qui peuvent plus ou moins modifier le profil utilisateur. Les solutions suivantes sont possibles :

- Solution 1 : Les instances de concept des documents jugés doivent être ajoutées dans le profil
- Solution 2 : Les instances de concept des documents jugés permettent de faire évoluer le poids des concepts parents qui sont dans le profil ou d'ajouter les concepts parents qu'ils ont en commun avec les instances du profil.
- Solution 3 : Combinaison des solutions 1 et 2

Ces différentes solutions illustrent le fait que le besoin qui se cache derrière un retour de pertinence n'est pas explicite. Pour exemple, si le profil contient l'instance E et que les documents pertinents contiennent l'instance F tel que E est le père de F, alors cela peut vouloir dire que le profil est mal défini. Le besoin de l'utilisateur peut se porter autour de F et non de E. Cependant, cela peut aussi vouloir dire que ces documents sont pertinents car F est thématiquement proche de E et tout autre document contenant une instance de concept fils de E aurait été, lui aussi, pertinent. Parmi les 3 solutions énumérées ci-dessus, il faut choisir celle qui permettra d'améliorer au mieux la représentation du besoin utilisateur. Les deux exemples suivants justifient le choix fait dans nos travaux. Ils sont basés sur l'ensemble des concepts C de DBPedia et sur les instances E, F, G et H. Comme l'illustre la figure 5.4, ces instances sont organisées en hiérarchie tel que E est le père de F, de G et de H. On alors :

$$E, F, G, H \in C$$

$$fils : E \rightarrow F, G, H$$

La distance hiérarchique entre E et F est donc la même qu'entre E et G, et qu'entre E et H.

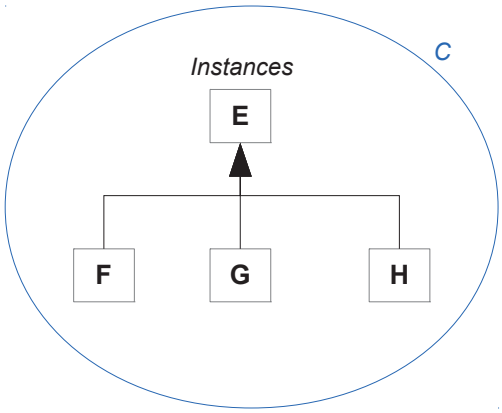


FIGURE 5.4 – Représentation hiérarchique des instances E, F, G et H

b. Exemple 1

Soit P, un profil contenant un vecteur d'instance de concepts $\vec{P_C}$, $\vec{D_C^P}$ et $\vec{D_C^{NP}}$ deux vecteurs contenant respectivement les instances de concept présentes dans les documents jugés pertinents et non pertinents par l'utilisateur à l'itération t. Dans notre exemple, on suppose qu'il existe une instance de concept E présente dans le vecteur thématique $\vec{P_C}$ mais pas dans les vecteurs $\vec{D_C^P}$ et $\vec{D_C^{NP}}$. De même, l'instance de concept F est uniquement présente dans le vecteur de documents pertinents $\vec{D_C^P}$:

$$\exists E \ tq \ E \in \vec{P_C}, \ E \notin \vec{D_C^P}, \ E \notin \vec{D_C^{NP}}$$

$$et \ \exists F \ tq \ F \in \vec{D_C^P}, \ F \notin \vec{P_C}, \ F \notin \vec{D_C^{NP}}$$

Puisque F fait partie du retour de pertinence positif, il va être ajouté au profil à l'itération suivante. L'ajout d'une nouvelle instance dans le profil fait baisser le poids des autres instances car notre profil est normalisé. Ainsi, le poids de E va baisser d'autant plus qu'il n'est pas présent dans les documents jugés pertinents. On a :

$$P^{t+1} = P^t \cup \{F\}$$

et

$$Sim_C(\vec{P_C^{t+1}}, \vec{D}_E) < Sim_C(\vec{P_C^t}, \vec{D}_E)$$
$$Sim_C(\vec{P_C^{t+1}}, \vec{D}_F) > Sim_C(\vec{P_C^t}, \vec{D}_F)$$

Ce retour de pertinence peut cependant représenter deux différents besoins :

- CAS 1.1 : *l'utilisateur s'intéresse plus à F qu'à E.* Après un certain nombre x d'itérations, les poids des instances doivent favoriser la mesure de similarité pour les documents contenant F. On doit avoir :

$$Sim_C(\vec{P_C^{t+x}}, \vec{D}_F) > Sim_C(\vec{P_C^{t+x}}, \vec{D}_E)$$

- CAS 1.2 : *l'utilisateur s'intéresse à E et à ces fils.* Or, les documents retournés par le système, à cette itération, ne parlent que de F. Le profil pourra contenir l'information

concernant l'intérêt de l'utilisateur pour F mais l'importance de celui-ci ne devra pas dépasser celui pour E. De plus, après un certain nombre x d'itérations, on doit toujours avoir :

$$Sim_C(\vec{P}_C^{t+x}, \vec{D}_F) < Sim_C(\vec{P}_C^{t+x}, \vec{D}_E)$$

Le mécanisme d'exploitation du RP doit pouvoir prendre en compte ces deux besoins. Les solutions précédemment présentées sont donc évaluées sur cet exemple.

Évaluation de la solution 1 *Ajouter les instances pertinentes dans le profil.*
Cela revient à ajouter l'instance F dans le vecteur d'instances de concept du profil $\vec{P}_C$.

$$E \in \vec{P}_C^t$$

$$E, F \in \vec{P}_C^{t+1}$$

Le poids W de l'instance E va diminuer. En effet, comme le profil est normalisé, augmenter le poids d'une instance fait baisser le poids des autres. En ajoutant F dans le profil, on a :

$$W_F^{t+1} > W_F^t$$

$$W_E^{t+1} < W_E^t$$

Suite au changement de poids, comme vu en introduction, on a alors

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^t, \vec{D}_F)$$

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) < Sim_C(\vec{P}_C^t, \vec{D}_E)$$

Le score de similarité augmente pour les documents contenant F et baisse pour ceux contenant E.

- En considérant le cas 1.1, le poids de F va continuer d'augmenter après plusieurs itérations. Ainsi, la courbe du poids de F dans le profil P au fil des itérations est croissante et sa dérivée est non nulle. Comme augmenter le poids d'une instance fait baisser le poids des autres et que cette diminution est relative à la valeur de Z, alors on peut conclure qu'il existe un entier x, tel qu'à l'itération $t + x$ on ai :

  $$Sim_C(\vec{P}_C^{t+x}, \vec{D}_F) > Sim_C(\vec{P}_C^{t+x}, \vec{D}_E)$$

  Le besoin du cas 1.1 est pris en compte par la solution 1.
- Dans le cas 1.2, l'ajout de F ne se fera qu'à l'itération t+1. Le score de similarité évoluera en fonction du contenu du retour de pertinence des itérations. Le poids de F dans le profil n'augmentera pas systématiquement et le poids de E ne baissera donc pas à chaque itération comme dans le cas 1.1. De ce fait, le score des documents contenant E resteront meilleurs que ceux contenant F.

  $$Sim_C(\vec{P}_C^t, \vec{D}_F) < Sim_C(\vec{P}_C^t, \vec{D}_E)$$

  Le besoin du cas 1.2 est également pris en compte par la solution 1.

Évaluation de la solution 2 *Faire évoluer le poids des instances parentes qui sont dans le profil (ou ajouter les instances parentes qu'ils ont en commun).*
Dans notre exemple, cela se traduit par une augmentation du poids W de l'instance E à l'itération suivante.

$$E \in \vec{P}_C^t$$
$$E \in \vec{P}_C^{t+1}$$

tel que $\mathrm{W}_E^{t+1} > \mathrm{W}_E^t$.
La fonction de similarité entre le profil et un document contenant E est impactée et son score augmente.

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^t, \vec{D}_E)$$

Cependant, la mesure de similarité est également basée sur la hiérarchie des instances, ce qui implique que :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^t, \vec{D}_F)$$
$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) > Sim_C(\vec{P}_C^t, \vec{D}_G)$$
$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_H) > Sim_C(\vec{P}_C^t, \vec{D}_H)$$

ainsi, après un certain nombre d'itérations x, on aura

$$Sim_C(\vec{P}_C^{t+x}, \vec{D}_F) < Sim_C(\vec{P}_C^{t+x}, \vec{D}_E)$$

Le cas 1.2 est donc pris en compte par ces changements puisque la modélisation du besoin couvre les instances E, F, G et H. Par contre, la spécialisation du profil n'est pas considérée par cette solution car le système retournera des documents sur F comme sur G, même si l'intérêt de l'utilisateur se porte uniquement sur F pendant plusieurs itérations. Le besoin du cas 1.1 n'est pas respecté par la solution 2.

Évaluation de la solution 3 *Ajouter les instances pertinentes dans le profil et faire évoluer le poids des concepts pères.*
Dans notre exemple, cela se traduit par une augmentation du poids de E à l'itération suivante et de l'ajout de F. On a initialement :

$$E \in \vec{P}_C^t$$

avec

$$W_E^t > 0$$
$$W_F^t = 0$$

A la mise à jour du profil, on obtient :

$$\vec{P}_C^{t+1} = \vec{P}_C^t \cup \{F\}$$

avec

$$W_E^{t+1} > W_E^t$$
$$W_F^{t+1} > W_F^t$$

Le système retournera encore des documents sur E même après plusieurs itérations de documents ne parlant que de F puisque W(E) continuera d'augmenter.

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^t, \vec{D}_E)$$

Comme avec la solution 2, le besoin du cas 1.2 n'est pas pris en compte. La modélisation du besoin couvre les documents parlant de E et donc de F, G et H de part le lien hiérarchique qui les unis. On gardera après un certain nombre d'itérations, un meilleur score pour les documents contenant E.

$$Sim_C(\vec{P}_C^{t+x}, \vec{D}_F) < Sim_C(\vec{P}_C^{t+x}, \vec{D}_E)$$

Cependant, le cas 1.1 n'est pas respecté non plus puisque la spécialisation du besoin sur F ne pourra pas se faire avec cette approche.

Récapitulatif des résultats de l'exemple 1 Dans cet exemple, on montre qu'augmenter le poids de l'instance E ne permet pas de spécialiser le profil (hypothèse 2 et 3). L'ajout d'instances provenant des documents pertinents dans le profil P (hypothèse 1) permet de centrer rapidement le profil sur un nouveau besoin au fur à mesure des itérations (cas 1.1). Si une nouvelle instance n'est retournée qu'une fois par les documents pertinents, le profil restera centré sur les instances de poids importants qui le compose (cas 1.2).

c. Exemple 2

Soit P, un profil contenant un vecteur conceptuel, D^P et D^{NP} deux vecteurs contenant respectivement les instances de concept présentes dans les documents jugés pertinents et non pertinents par l'utilisateur à l'itération t. Dans cet exemple, on suppose que les instances F et G sont présentes dans le profil P et dans les documents jugés pertinents. On suppose également que l'instance E n'est ni présente dans le profil, ni dans les documents jugés pertinents ou non pertinents.

$$\exists F \ tq \ F \in \vec{P}_C, \ F \in \vec{D}_C^P, \ F \notin \vec{D}_C^{NP}$$

$$\exists G \ tq \ G \in \vec{P}_C, \ G \in \vec{D}_C^P, \ G \notin \vec{D}_C^{NP}$$

$$\exists E \ tq \ E \notin \vec{D}_C^P, \ E \notin \vec{P}_C, \ E \notin \vec{D}_C^{NP}$$

- CAS 2.1 : *l'utilisateur s'intéresse à l'instance E en général et donc à ses fils F, G.* Puisque le profil ne contient que les instances F et G, il est nécessaire d'ajouter cette information dans le profil pour retourner plus de documents sur F, G mais aussi sur E et H. Le retour de pertinence doit permettre de produire un profil P tel que

  $$F, G, H, E \in \vec{P}_C$$

  et

  $$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^{t}, \vec{D}_E)$$
  $$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^{t}, \vec{D}_F)$$
  $$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) > Sim_C(\vec{P}_C^{t}, \vec{D}_G)$$
  $$Sim_C(\vec{P}_C^{t+1}, \vec{D}_H) > Sim_C(\vec{P}_C^{t}, \vec{D}_H)$$

- CAS 2.2 : l'utilisateur s'intéresse uniquement à F et G mais pas à E ni à H. Le profil est déjà suffisamment spécialisé.

  $$F, G \in \vec{P}_C$$

et

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^{t}, \vec{D}_F)$$
$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) > Sim_C(\vec{P}_C^{t}, \vec{D}_G)$$

Les solutions précédemment présentées sont évaluées sur ce second exemple. En effet, notre mécanisme d'exploitation du RP doit également pouvoir prendre en compte ces deux différents besoins.

Évaluation de la solution 1 *Ajouter les instances pertinentes dans le profil.*
Cela revient à augmenter le poids de F et G déjà présents dans le profil P.

$$F, G \in \vec{P}_C^{t}$$
$$F, G \in \vec{P}_C^{t+1}$$

avec

$$W_F^{i+1} > W_F^{i}$$
$$W_G^{i+1} > W_G^{i}$$

En augmentant les poids de ces concepts, on améliore la mesure de similarité pour les documents comportant ces concepts :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^{t}, \vec{D}_F)$$
$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) > Sim_C(\vec{P}_C^{t}, \vec{D}_G)$$

mais aussi indirectement, et de manière moins importante, le score de similarité des documents conceptuellement liés tels que ceux contenant les instances E ou H :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^{t}, \vec{D}_E)$$
$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_H) > Sim_C(\vec{P}_C^{t}, \vec{D}_H)$$

Le cas 2.2 est strictement respecté puisque le profil n'ajoute pas d'information sur E et H. Le cas 2.1 est également respecté car le score de similarité est amélioré pour les documents sur E et H même s'il l'est moins que pour les documents comportant F et G. De plus, un document jugé pertinent comportant l'instance E et/ou H permettra d'ajouter ces concepts dans le profil et donc d'améliorer la représentation du besoin.

Évaluation de la solution 2 *Faire évoluer le poids des instances parentes qui sont dans le profil (ou ajouter les instances parents qu'ils ont en commun).*
Dans notre exemple, cela se traduit par l'ajout de E dans le profil à l'itération suivante.

$$F, G \in \vec{P}_C^{t}$$
$$\vec{P}_C^{t+1} = \vec{P}_C^{t} \cup \{E\}$$

avec

$$W_F^{t+1} < W_F^{t}$$
$$W_G^{t+1} < W_G^{t}$$
$$W_E^{t+1} > W_E^{t}$$

Comme le profil est normalisé, l'ajout de E entraîne une diminution du poids de F et G. La fonction de similarité entre le profil et un document contenant E est impactée et son score augmente.

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^{t}, \vec{D}_E)$$

Cependant, la mesure de similarité est également basée sur la hiérarchie des instances, ce qui implique que :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_H) > Sim_C(\vec{P}_C^{t}, \vec{D}_H)$$

La diminution du poids de F et G se répercute dans le score de similarité des documents contenant ces deux concepts qui diminue :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) < Sim_C(\vec{P}_C^{t}, \vec{D}_F)$$

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) < Sim_C(\vec{P}_C^{t}, \vec{D}_G)$$

Les documents comportant l'instance E ont un score qui est directement amélioré. L'ajout de E améliore aussi de façon indirecte le score des documents contenant des instances liées (F, G et H). Le besoin du cas 2.1 est donc bien pris en compte.

Cependant, plusieurs itérations avec le même retour de pertinence augmentera le poids de E et diminuera d'avantage celui de F et G. Le score de similarité des documents contenant F et G est diminué par rapport au profil initial. De plus, des documents contenant E et H seront plus facilement retournés, même si le besoin n'est qu'autour de F et G. Le besoin du cas 2.2 n'est donc pas pris en compte avec l'ajout de E.

Évaluation de la solution 3 *Ajouter les instances pertinentes dans le profil et faire évoluer le poids des instances parentes (ou ajouter les instances parentes).*
Dans notre exemple, cela se traduit par une augmentation du poids W des instances F et G à l'itération suivante et de l'ajout de l'instance parente E. On a initialement :

$$F, G \in \vec{P}_C^{t}$$

avec

$$W_F^t > 0$$

$$W_G^t > 0$$

$$W_E^t = 0$$

A la mise à jour du profil, on obtient :

$$\vec{P}_C^{t+1} = \vec{P}_C^{t} \cup \{E\}$$

avec

$$W_E^{t+1} \nearrow$$

$$W_F^{t+1} \nearrow$$

$$W_G^{t+1} \nearrow$$

La fonction de similarité entre le profil et un document contenant E, F ou G est impactée et son score augmente.

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_E) > Sim_C(\vec{P}_C^{t}, \vec{D}_E)$$

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_F) > Sim_C(\vec{P}_C^{t}, \vec{D}_F)$$

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_G) > Sim_C(\vec{P}_C^{t}, \vec{D}_G)$$

Cependant, la mesure de similarité est également basée sur la hiérarchie des intances, ce qui implique que :

$$Sim_C(\vec{P}_C^{t+1}, \vec{D}_H) > Sim_C(\vec{P}_C^{t}, \vec{D}_H)$$

Les documents comportant l'instance E ont un score qui est directement amélioré. L'ajout de E améliore aussi de façon indirecte le score des documents contenant des instances liées (F, G et H), sans compter que le score des documents contenant F et G est aussi directement amélioré. Le besoin du cas 2.1 est donc bien pris en compte.

Cependant, comme précédemment, le besoin du cas 2.2 n'est pas pris en compte puisque des documents contenant E et H seront plus facilement retournés, alors que le besoin n'est qu'autour de F et G.

Récapitulatif des résultats de l'exemple 2 Dans ce second exemple, on montre qu'ajouter une instance parente ne permet pas de garder une spécialisation du profil (solution 2 et 3). L'ajout des instances provenant des documents pertinents dans le profil P (solution 1) permet de rester centré sur le besoin (cas 2.2) ou d'élargir celui-ci en fonction du retour de pertinence (cas 2.1).

d. Synthèse des résultats

Les deux exemples précédents permettent de différencier les cas de figure qui sont sous-jacents à un retour de pertinence. Il parait naturel de vouloir rajouter des informations utiles dans le profil suite à un retour de pertinence. L'ajout d'une instance parente à deux instances d'intérêt peut sembler logique. Parmi les 3 solutions possibles présentées dans cette partie, celles qui utilisent les informations hiérarchiques entre les instances pour redéfinir le profil ne permettent pas de couvrir les différents besoins de l'utilisateur de manière satisfaisante (cf. tableau 5.1).

TABLE 5.1 – Récapitulatif des résultats formels

	Cas 1.1	Cas 1.2	Cas 2.1	Cas 2.2
Solution 1	Ok	Ok	Ok	Ok
Solution 2	Ok	/	Ok	/
Solution 3	Ok	/	Ok	/

Le raisonnement suivi pour exploiter le retour de pertinence et affiner la représentation terminologique peut donc s'adapter à notre représentation thématique, puisqu'aucune information quant à la hiérarchie DBPedia ne doit être prise en compte. Nous pouvons donc introduire la définition 9, page 94, pour la retour de pertinence de la couverture thématique.

Définition 9 (Le retour de pertinence pour la couverture thématique)**.**

$$\vec{P}_C^{t+1} = \alpha\vec{P}_C^{t} + \beta\vec{C}^{P,t} - \gamma\vec{C}^{NP,t}$$

où $\vec{P}_C^{t+1}$ est le vecteur thématique du profil à l'itération t+1, $\vec{C}^{P,t}$ et $\vec{C}^{NP,t}$ sont respectivement les vecteurs d'instances de concept du retour de pertinence R, avec

$$\vec{C}^{P,t} = ((C_1^P, W_{C_1^P}), ..., (C_a^P, W_{C_a^P}))$$

où $C_i^P \in \mathscr{C}$,
$W_{C_i^P}$ est le poids du mot-clé C_i^P avec $W_{C_i^P} \in \mathbb{Z}$, $0 \leqslant W_{C_i^P} \leqslant 1$,
et i $\in$ [1,a].

$$\vec{C}^{NP,t} = ((C_1^{NP}, W_{C_1^{NP}}), ..., (C_b^{NP}, W_{C_b^{NP}}))$$

où $C_j^P \in \mathscr{C}$,
$W_{C_j^P}$ est le poids du mot-clé C_j^P avec $W_{C_j^P} \in \mathbb{Z}$, $0 \leqslant W_{C_j^P} \leqslant 1$,
et j $\in$ [1,b].

La justification relative aux choix des valeurs pour les termes pondérateurs est la même que dans la partie précédente, page 83. Cette prise en compte du retour de pertinence permet d'améliorer la représentation du besoin mais également d'appréhender un changement de besoin.

5.3.3 Evolution du profil opérationnel

Le besoin de l'expert peut évoluer au fur et à mesure qu'il reçoit et analyse l'information. Il peut changer, se préciser, s'élargir et toutes ses variations sont prisent en compte dans le profil opérationnel grâce aux étapes suivantes :

1. Présentation à l'utilisateur de nouvelles sources jugées d'intérêt par DOWSER ;
2. Jugement par l'utilisateur des sources présentées ;
3. Construction des vecteurs de mots-clés et d'instances de concept à partir des sources jugées pertinentes ;
4. Construction des vecteurs de mots-clés et d'instances de concept à partir des sources jugées non pertinentes ;
5. Application des formules de retour de pertinence sur le profil P^t
6. Sauvegarde du nouveau profil créé P^{t+1}
7. Retour à l'étape 1 ;

L'avantage de la prise en compte du retour de pertinence est double. Il permet d'affiner la représentation du besoin mais également d'en appréhender les variations. Si une nouvelle tendance se dégage dans l'extraction des termes extraits des sources jugées pertinentes (ou non pertinentes), celle-ci impactera directement le vecteur terminologique et le vecteur thématique. Ces changements se répercuteront dans la modélisation du besoin et permettront au système DOWSER d'adapter sa découverte de nouvelles sources d'intérêt.

5.4 Synthèse

Dans ce chapitre, le besoin en information dans le cadre du ROSO a été modélisé au sein d'un profil opérationnel. Il est construit implicitement à partir de sources d'intérêt explicitées par l'expert. Ces sources représentent la connaissance de l'utilisateur sur le sujet de recherche. Elles sont exploitées afin de construire un vecteur d'instances de concept permettant d'avoir une couverture thématique du besoin. Elles sont également utilisées pour construire un vecteur de mots clés représentatif de la terminologie du besoin. Ces deux vecteurs modélisent les centres d'intérêt de l'utilisateur et forment avec les sources d'intérêt, fournies par l'expert, un profil cognitif. Ce profil pallie le manque de représentativité des requêtes et permet de couvrir l'ensemble du vocabulaire sur le sujet de recherche et de faire face à son évolution. La représentation du besoin est affinée au fur et à mesure que le système DOWSER découvre de nouvelles sources d'intérêt par le biais de retour de pertinence basée sur l'approche Rocchio [Rocchio, 1971]. L'évaluation, par l'expert, des sources découvertes permet de faire évoluer le profil afin de prendre en compte les variations du besoin ou d'améliorer la représentation du besoin opérationnel. Ce profil est exploité par le système de découverte de nouvelles sources de DOWSER, présenté dans le chapitre suivant.

CHAPITRE 6

DOWSER, À LA DÉCOUVERTE DE NOUVELLES SOURCES D'INTÉRÊT

Nous avons introduit dans le chapitre précédent un profil opérationnel permettant de modéliser le besoin en information des experts du renseignement. Cette modélisation du besoin est capitale pour fournir des sources pertinentes à l'expert. Dans le cadre du ROSO, la découverte de sources d'intérêt opérationnel sur des sujets spécifiques et/ou sensibles est un enjeu important et la pierre angulaire du cycle du renseignement. Pour rappel (voir section 4.2, page 68), cette tâche de découverte de sources doit faire face à :

- l'importante taille du Web sur lequel les systèmes travaillent, qui rend difficile la recherche d'information spécifique ;
- les métriques de pertinence en partie basées sur la popularité de la source alors qu'une information sensible, discrète est souvent impopulaire ;
- la vision tronquée du Web qu'ont les systèmes de part la vitesse de création et de modification de l'information sur Internet.

Le second axe de recherche de cette thèse, présenté dans ce chapitre, a pour but de pallier ces limites. Notre proposition s'articule autour d'un système de découverte de nouvelles sources d'information pertinentes capable :

- d'exploiter le profil opérationnel pour juger de la pertinence des pages visitées ;
- de parcourir le Web de manière à collecter uniquement des pages d'intérêt ;
- de fournir à l'expert des résultats en temps réel avec des pages d'intérêt fraîchement collectées ;
- de faire évoluer le profil opérationnel en même temps que le besoin de l'utilisateur évolue, et d'en impacter directement le processus de découverte de sources.

D'un point de vue opérationnel, l'outil proposé doit permettre de découvrir des sources d'intérêt opérationnel avec de l'information pertinente et ainsi de fournir des résultats complémentaires aux outils de RI classiques.

Dans ce chapitre, le processus complet de découverte de nouvelles sources de DOWSER est présenté. Les phases d'exploration ciblée du Web et d'amorçage du processus de collecte y sont décrits ainsi que le module d'enrichissement des pages collectées, de sauvegarde et

de présentation des sources à l'utilisateur.

6.1 Exploration ciblée du Web

Afin de constituer un corpus de nouvelles sources d'intérêt, nous présentons dans cette section notre processus d'exploration ciblée du Web permettant d'identifier des grappes thématiques et de collecter des pages d'intérêt.

6.1.1 Un corpus de sources d'intérêt

À l'instar de la RI, la DI a pour but de répondre à un besoin en information en fournissant un ensemble de pages d'intérêt sur un sujet de recherche. La RI travaille sur un corpus de documents déjà constitué au moment de la recherche. Considérant un besoin donné, ce corpus peut contenir des pages pertinentes ou non. Les moteurs de recherche travaillent sur un corpus qui tend à être représentatif de l'état du Web (voir section 2.1.1, page 14). La taille de celui-ci est une des limites exprimées précédemment : il est difficile de trouver une information précise dans cet immense ensemble. L'un des objectifs de la DI est de construire un corpus constitué uniquement d'informations pertinentes.

Chaque page Web présente dans le corpus d'un système de DI a été *découverte* et collectée par celui-ci. Une fois collectée, une page Web devient un document au sein du système. Dans notre approche, on appelle *source d'intérêt*, une page Web ou un ensemble de pages Web provenant du même domaine qui a été découverte par notre système et qui contient de l'information pertinente pour le sujet de recherche de l'expert. Pour trouver ces nouvelles sources d'intérêt, notre système de DI doit explorer le Web de manière ciblée en considérant le besoin de l'utilisateur afin de collecter des pages Web pertinentes.

6.1.2 Cibler les grappes thématiques et les sources d'intérêt

Les robots d'indexation sont des outils de DI qui peuvent être configurés afin d'explorer le Web de manière ciblée. Un ensemble non exhaustif d'approches utilisant des robots d'indexation ciblée a été présenté dans le chapitre 2.2, page 23. Ces différentes approches utilisent la structure du Web pour explorer celui-ci et exploitent le besoin de l'utilisateur pour cibler les pages pertinentes. Dans notre cas, il doit faire abstraction de la popularité des pages afin de pouvoir récupérer des pages sensibles et discrètes, tout en considérant le profil opérationnel, et ainsi fournir une alternative aux outils existants utilisés par les experts.

L'exploration du Web dans DOWSER se base sur les travaux de Davison et sur son *Topical Locality Phenomenon* (TLP) [Davison, 2000]. Pour rappel, Davison montre que deux pages Web, connectées l'une à l'autre par un hyperlien, partagent généralement un contenu textuel commun. Autrement dit, considérant un besoin lambda, une page pertinente est souvent liée à une autre page pertinente. Si l'on considère ce phénomène et que l'on représente le Web par un immense graphe de pages connectées, on peut déceler des grappes de pages thématiques comme l'illustre le schéma 6.1, page 99. Les points bleus foncés représentent des pages pertinentes connexes à d'autres pages thématiquement proches représentées par un point bleu clair. La figure 6.2, page 100, est basée sur le second scénario opérationnel présenté page 72. Elle illustre ces grappes thématiques : en plus des liens vers des pages internes, la page `http://www.herbsbuy360.com/`, qui propose à la

vente des produits médicamenteux chinois, est connexe à deux autres sites Web proposant également la vente de ce genre de produits (`http://www.globalchineseherbhealing.com/index.php` et `http://www.canceranti.com/index.html`). Cette grappe thématique, ainsi que le scénario opérationnel, sont décrits en détail dans l'annexe A, page 179.

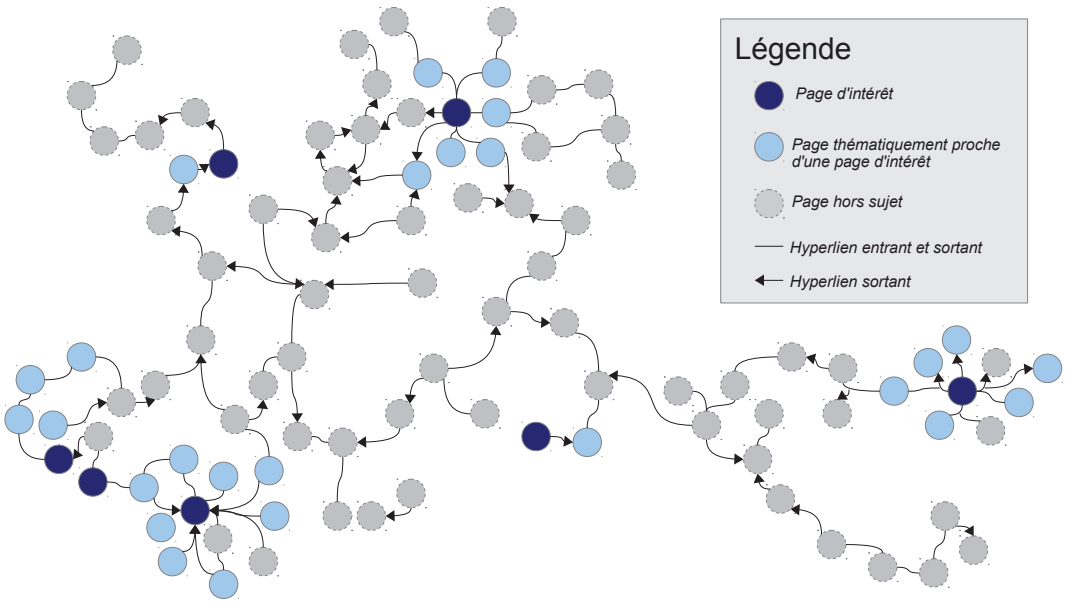


FIGURE 6.1 – Représentation par grappes thématiques du Web considérant un besoin lambda

6.1.3 Le processus d'exploration ciblée de DOWSER

Le processus d'exploration ciblée réside dans la découverte de ces pages d'intérêt et la collecte en priorité des pages connexes en suivant les étapes décrites ci-dessous, illustrées dans le schéma 6.3, page 101 :

1. Amorçage de la collecte avec les URLs d'intérêt de l'utilisateur P_{si} et avec le module d'extension ;
2. Collecte des pages Web pointées par les URLs de la pile ;
3. Stockage des pages Web collectées ;
4. Enrichissement du contenu des documents stockés ;
5. Sauvegarde des documents enrichis ;
6. Ajout des URLs présentes dans les documents enrichis et de leur priorité dans la pile d'URLs à explorer ;
7. Retour à l'étape 2 ;

La pile d'URLs contient l'ensemble des URLs à explorer ordonné par priorité. En accord avec le TLP, la priorité des liens à visiter est déterminée par la pertinence du document dont ils sont extraits. Cette pertinence est calculée en fonction d'une mesure de similarité entre le contenu du document et le profil opérationnel (voir section 6.3.2, page 106). A

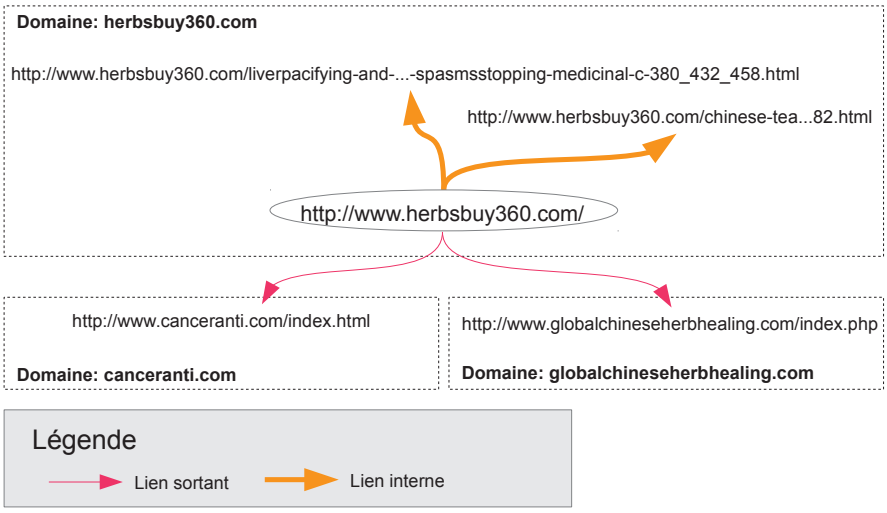


FIGURE 6.2 – Exemple de grappe thématique

priorité égale, un lien pointant vers une page dont le domaine est différent de la page parente sera visité avant un lien pointant vers une page du même domaine. Le but étant ici de favoriser la découverte de nouvelles sources en élargissant l'exploration et d'éviter de collecter un site dans son ensemble. Le schéma 6.4, page 102, illustre l'exploitation qui est faite des scores de similarité pour guider la collecte dans le cas du scénario de vente de médicaments à base de produits illégaux (voir le second scénario page 72). Dans cet exemple, 4 pages provenant du site `http://www.herbsbuy360.com/` ont été analysées et un score leur est affecté. Ce dernier permet de définir l'ordre dans lequel les liens extraits de ces pages seront explorés (illustrés par les chiffres en rouge dans le scéma 6.4). Ainsi, le système visitera en priorité les liens de la page contenant un produit à base de corne d'antilope (rang 1) que les liens de la page proposant des produits médicinaux chinois contre le cancer (rang 4).

Le processus de collecte est donc un processus qui boucle en collectant en priorité les pages qu'il juge d'intérêt pour le sujet de recherche en cours. La section suivante présente comment les pages Web collectées sont modélisées et sauvegardées dans le système DOWSER.

6.2 Amorçage du processus d'exploration

L'amorçage du processus d'exploration constitue la première étape de la collecte ciblée comme l'illustre le schéma 6.3, page 101. Notre système doit démarrer l'exploration du Web par des pages pertinentes afin de trouver rapidement de nouvelles sources répondant au besoin de l'expert. En démarrant la collecte par des pages pertinentes, le système pourra exploiter la grappe thématique où se trouvent ces pages. Dans le cas contraire, le robot d'indexation devrait explorer aléatoirement le Web jusqu'à ce qu'il trouve des grappes thématiques sur le sujet de recherche. Dans DOWSER, l'amorçage du processus d'exploration exploite le profil opérationnel et également un module d'extension de la zone de collecte.

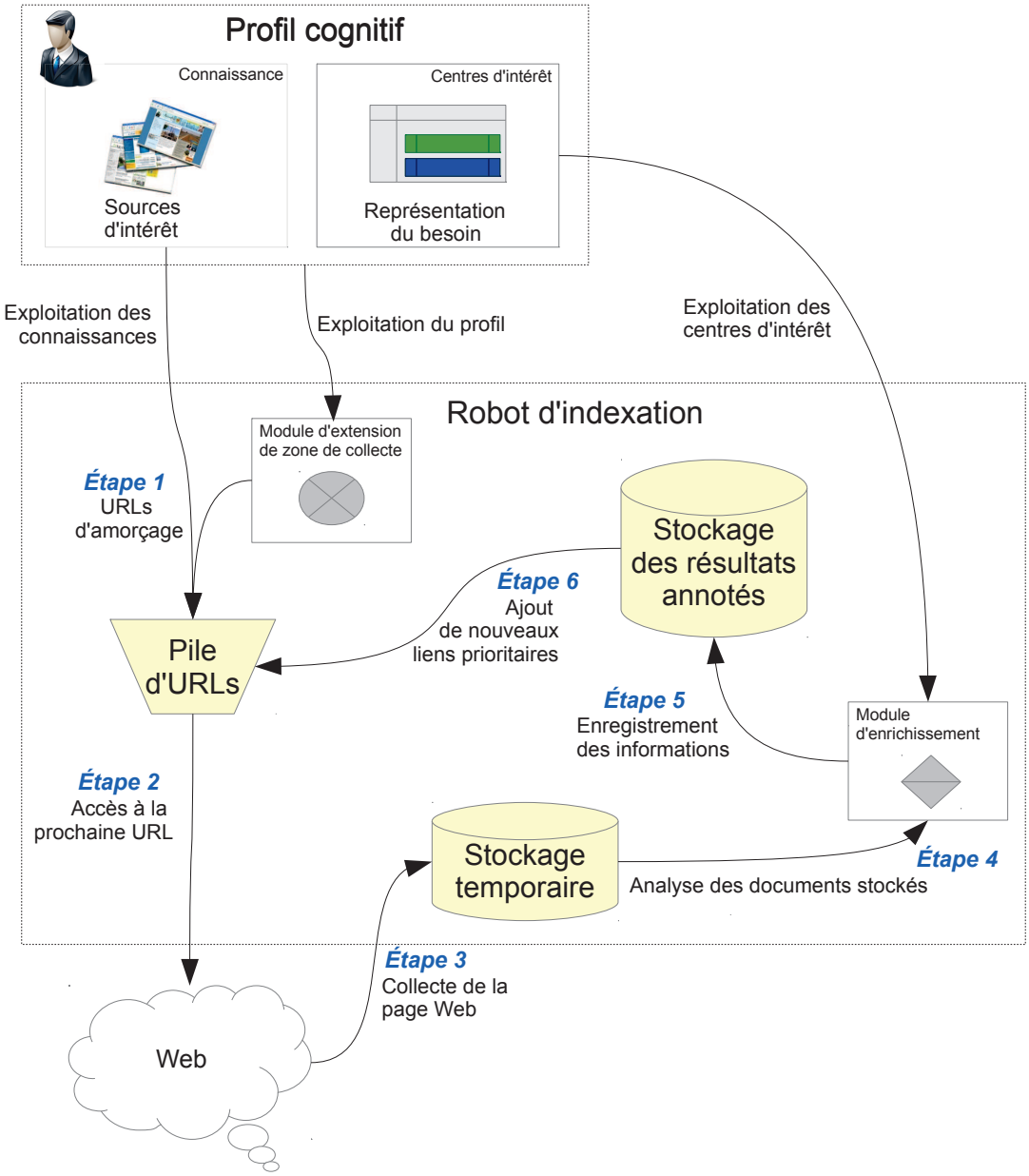


FIGURE 6.3 – Processus de collecte de DOWSER

6.2.1 Exploitation du profil opérationnel

Le profil opérationnel dans DOWSER est construit à partir de sources d'intérêt fournies par l'expert P_{si} . Elles constituent la connaissance de celui-ci pour un sujet de recherche donné (voir définition 1, page 75). Ses pages étant des pages pertinentes, elles sont un excellent point de départ pour le robot d'exploration qui peut démarrer sa collecte directement dans des grappes thématiques. L'inconvénient réside dans le nombre limité de sources d'intérêt d'amorçage. Comme nous l'avons expliqué dans les exemples de scénarios opérationnels, l'expert du renseignement ne dispose que d'un nombre très limité de sources

FIGURE 6.4 – Exemple d'exploitation des scores de similarité pour guider la collecte

d'intérêt, son objectif étant d'en trouver de nouvelles.

6.2.2 Extension de la zone de collecte

L'exploitation des sources d'intérêt connues de l'expert est un point de départ pour notre robot d'exploration. Cependant, pour ne pas se cantonner à une zone d'intérêt trop restreinte, et afin d'améliorer la découverte de nouvelles grappes thématiques, le système DOWSER est muni d'un processus d'extension de zone de collecte permettant de trouver plus rapidement de nouvelles pages d'intérêt.

Le processus d'extension de la zone de collecte a pour but de fournir de nouvelles pages d'intérêt de départ au robot d'exploration. Il repose sur l'utilisation de méthodes de RI, de l'exploitation de la structure du Web et des centres d'intérêt modélisés dans le profil opérationnel.

a. Utiliser les liens rentrants

Un lien sortant d'une page A est un lien présent dans cette page qui pointe vers une page B. Du point de vue de cette page B, ce lien est un lien rentrant. Le système DOWSER exploite déjà les liens sortants des pages d'intérêt de l'utilisateur puisque ces pages sont explorées par le robot d'exploration. Considérant le TLP, les liens rentrants sont également des pages à explorer car elles peuvent être thématiquement proches. Les pages contenant des liens vers les pages d'intérêt de l'expert peuvent être des pages pertinentes menant à de nouvelles grappes thématiques. Pour trouver ces pages, il est possible d'utiliser des moteurs de recherche ou des outils spécialisés. Par exemple, la requête *link :monUrl* permet, sur Google, de trouver les pages contenant des liens rentrants vers *monUrl*. L'exploration peut ensuite se poursuivre à partir de ces nouvelles pages.

b. Utiliser les moteurs de recherche

Koutrika et Ioannidis utilisent les mots-clés contenus dans leur profil utilisateur pour construire des requêtes [Koutrika & Ioannidis, 2005]. Dans DOWSER, les mots-clés présents dans la représentation terminologique de notre profil opérationnel peuvent servir de la même façon à créer des requêtes représentatives du besoin. Ces requêtes sont envoyées à différents moteurs de recherche et les pages retournées peuvent être exploitées par notre système. Cependant, fournir directement à l'expert ces pages ne va pas permettre de proposer une alternative aux systèmes existants. Il peut d'ailleurs potentiellement les retrouver s'il utilise, de lui même, la même combinaison de mots-clés pour formuler sa requête. L'avantage de ces pages est d'obtenir de nouvelles pages d'entrée à explorer afin d'élargir la zone de collecte et trouver plus rapidement de nouvelles grappes thématiques sur le sujet de recherche.

c. Utiliser les liens des concepts DBPedia

Dans DOWSER, la représentation conceptuelle du besoin en information de l'expert est constituée de concepts DBPedia. Chacun de ces concepts DBpedia est décrit par un ensemble d'attributs permettant de définir le nom du concept, un résumé, le lien vers la page Wikipedia, etc. Deux attributs sont particulièrement intéressants pour notre approche dans la mesure où ils contiennent des liens HTML vers des pages Web externes ainsi que

des liens RDF vers des sources de données externes [1]. Ces deux attributs comprenant des liens HTML sont :

- dbpedia:reference dont les liens pointent vers des pages Web en relation avec le concept,
- dbpedia:homepage contenant l'URL vers ce qu'on peut appeler la "page officielle" du concept.

Les pages pointées par ces liens peuvent être des sources d'intérêt car leur contenu est lié aux concepts représentatifs du besoin thématique de l'utilisateur. Ainsi, elles peuvent permettre à notre système de découvrir de nouvelles grappes thématiques en élargissant sa zone de collecte.

d. Capitaliser sur les tâches de découverte précédentes

Les experts du renseignement sont limités dans leurs actions de collaboration avec leurs collègues dans la mesure où certains sujets de recherche sont confidentiels et qu'ils n'ont pas pour habitude d'échanger de l'information discrète sur des sujets sensibles. L'avantage de DOWSER est de pouvoir exploiter de manière transparente des sources déjà découvertes lors d'anciennes tâches de recherche pour améliorer les suivantes. Une mesure de similarité permet d'établir la pertinence des pages collectées en fonction du profil opérationnel (voir section 6.3.2, page 106). Le score des pages stockées par DOWSER durant une collecte ultérieure peut être recalculé en considérant le profil opérationnel courant. Ainsi, les pages pertinentes pour ce nouveau besoin peut servir d'URL d'amorçage sans que l'expert connaisse la provenance de celles-ci. Cela se fait donc sans considération des profils opérationnels des experts utilisant le système. L'anonymat des experts et la discrétion sur leurs sujets de recherche sont donc préservés.

Le processus d'extension se fait automatiquement, sans intervention de l'expert, et celui-ci n'a pas connaissance des pages utilisées. Le nombre de pages que fournit le module d'extension n'est pas limité. En effet, le module d'extension fait partie intégrante du processus de collecte et c'est ce dernier qui gère la liste des URLs à visiter ainsi que leur priorité. Le fonctionnement global du processus de collecte est présenté ci-après.

6.3 Module d'enrichissement

Les pages Web collectées sont transformées en documents afin d'être analysées. Cette analyse permet d'enrichir la connaissance sur les pages en les annotant d'informations sur le contexte de la collecte, sur les mots-clés et les instances de concept extraits de leur contenu et sur leur pertinence au regard du profil opérationnel.

6.3.1 De la page Web au document

Une fois rapatriée du Web vers l'espace de stockage du système, une page collectée devient un document. C'est une version modifiable de la page Web qui contient donc l'URL de celle-ci, son titre et un résumé de son contenu. Ces informations sont extraites des métadonnées contenues dans la page. Le module d'annotation, présent dans le processus

1. http://wiki.dbpedia.org/Datasets

de collecte (voir figure 6.3), de décrire un document selon la définition 10 en enrichissant le document avec les informations supplémentaires suivantes :

- Le contexte Ctx : le nom du profil opérationnel et l'URL de la page parente qui a permis de découvrir cette nouvelle page sont ajoutés.
- Le score de pertinence S : afin de parler de source d'intérêt, la pertinence de la page doit être évaluée. Pour ce faire, une mesure de similarité adaptative, présentée page 106, permet d'obtenir un score de pertinence.
- Les liens L : les URLs présentes dans la page sont extraites et ajoutées comme annotation.
- Les mots-clés $\vec{D_K}$: à l'instar de la construction du profil (voir chapitre 5, page 71), les mots-clés représentatifs du contenu de la page sont extraits via l'outil Apache Lucene [2] et enregistrés dans le document.
- Les concepts $\vec{D_C}$: la thématique représentative du contenu de la page est également une annotation ajoutée au document via l'extraction des instances de concept DBPedia présents dans celle-ci. L'outil DBPedia Spotlight [3] est utilisé de la même manière que pour la construction du profil opérationnel.

Les vecteurs de mots-clés $\vec{D_K}$ et d'instances de concept $\vec{D_C}$ sont décrits dans la définition 6, page 82. L'exemple 5, page 106, décrit le contenu d'une page vendant des médicaments à base de produits illicites, collectée par DOWSER.

Définition 10 (Les pages collectées enrichies par le système DOWSER)**.**

Soit D, la description d'une page collectée contenant Ctx un ensemble d'informations contextuelles, S le score de pertinence, L un ensemble de liens, $\vec{D_K}$ le vecteur de mots-clés et $\vec{D_C}$ le vecteur d'instances de concept.

$$D = \{Ctx, S, L, \vec{D_C}, \vec{D_K}\}$$

tel que

$$S \in \mathbb{Z}, 0 \leqslant S \leqslant 1$$

Les informations contextuelles et le score de pertinence permettent de différencier les documents des différentes tâches de collecte mais aussi de les comparer entre eux. La section 6.4.2, page 114, explique comment ces informations sont utilisées pour fournir à l'expert des sources d'intérêt opérationnel pour son sujet de recherche courant. Les concepts et mots-clés sont sauvegardés afin d'être exploités par le module d'extension de zone de collecte pour les prochaines tâches de découverte de sources. Ces données sont aussi utilisées par le module de retour de pertinence (voir section 5.3, page 81) lorsque l'expert juge les documents qui lui sont suggérés. La section suivante présente comment le score de pertinence est calculé au travers de notre mesure de similarité adaptative.

2. http://lucene.apache.org/core/
3. https://github.com/dbpedia-spotlight/

Exemple 5 (Exemple de page collectée enrichie par DOWSER)**.**

$$D = \{Ctx, S, L, \vec{D_C}, \vec{D_K}\}$$

avec Ctx={12068_TCM, `http://www.herbsbuy360.com/`},

S=0.823,

L={`http://www.herbsbuy360.com/boilfree-herbs-liverpacifying-and-windextinguishing-medicinal-c-380_432.html`, `http://www.herbsbuy360.com/chinese-tea-c-382.html`, ..., `http://www.globalchineseherbhealing.com/index.php`, `http://www.canceranti.com/index.html` },

$\vec{D_C}$ = ((Traditional-Chinese-Medecine,0.4),(Health,0.2),(China,0.15), (Pain,0.15),(Herbalism,0.1)),

$\vec{D_K}$ = ((TCM,0.32),(yang jiao,0.16),(antelope horn,0.16),(health,0.08), (herbal-formula,0.08),(chinese,0.07),(herbs,0.06),(cancer,0.03), (spams-stopping,0.02),(CANTICER, 0.02).

6.3.2 Mesure de similarité adaptative

La mesure de similarité vise à évaluer l'intérêt du contenu d'une page Web par rapport à un besoin en information. Le profil opérationnel modélise ce besoin et il est donc exploité dans cette mesure. Il contient une représentation thématique et une représentation terminologique qui permettent une couverture complète du besoin. La mesure de similarité doit prendre en considération ces deux représentations différentes.

a. Mesure de similarité pour la couverture terminologique

La couverture terminologique du besoin est représentée dans DOWSER par les mots-clés extraits des pages d'intérêt de l'expert. L'outil Apache Lucene, qui permet d'obtenir cette représentation terminologique, est également utilisé sur les documents collectés correspondant aux pages Web visitées. Pour chaque document, un vecteur de mots-clés pondérés $\vec{D_K}$ est construit et il est comparé à celui du profil $\vec{P_K}$ afin d'évaluer la pertinence du document. La mesure de similarité doit calculer le plus finement possible la distance entre les deux vecteurs afin de conserver la précision offerte par la couverture terminologique. C'est ce que permet de faire la mesure cosinus. Comme détaillé section 2.1.2, page 15, elle considère chaque terme un à un et fournit une mesure nuancée entre 0 et 1 de correspondance entre les deux vecteurs $\vec{P_K}$ et $\vec{D_C}$ comme décrit dans la définition 11.

Définition 11 (La mesure de similarité terminologique)**.**

$$Sim_K(\vec{P}_K, \vec{D}_K) = \cos(\vec{P}_K, \vec{D}_K).$$

où $\vec{P}_K$ et $\vec{D}_K$ sont respectivement les vecteurs de mots-clés du profil opérationnel P et du document D et

$$Sim_K(\vec{P}_K, \vec{D}_K) \in \mathbb{Z}, 0 \leqslant Sim_K(\vec{P}_K, \vec{D}_K) \leqslant 1$$

Plus le résultat de la mesure cosinus est proche de 1, c'est à dire plus l'angle entre les deux vecteurs est faible, et plus le document couvre correctement le besoin opérationnel. Cette mesure permet d'évaluer une partie de la pertinence d'une page Web en s'intéressant à la représentation terminologique du besoin. La section suivante présente la mesure de similarité exploitant la représentation thématique du besoin opérationnel.

b. Mesure de similarité pour la couverture thématique

Les instances de concept sont extraites des pages collectées et permettent de représenter le contenu thématique de celui-ci. Un vecteur d'instances de concept $\vec{D_C}$ est construit et comparé avec celui présent dans le profil opérationnel $\vec{P_C}$. La distance qui sépare ces deux vecteurs permet d'évaluer la pertinence du document : plus la distance est faible et plus le document couvre le besoin thématique de l'expert. Le calcul de distance utilisée est celui proposé par Aurélien Saint-Requier [Saint Requier A., 2012] qui permet de déterminer la distance entre des instances DBpedia extraites d'une requête et ceux d'un profil utilisateur. La similarité sémantique entre une instance DBpedia C et les concepts DBpedia du profil utilisateur P se base sur l'approche de Milne et Witten [Witten & Milne, 2008]. Cependant, il utilise les catégories (correspondant au classement thématique des instances) en commun des concepts DBpedia pour déterminer si deux concepts sont sémantiquement proches. Dans notre approche, ce calcul, décrit en définition 12, est normalisé pour garder une valeur comprise entre 0 et 1 comme dans la mesure de similarité terminologique :

Définition 12 (La mesure de similarité sémantique)**.**

$$SimSem(\vec{P_C}, C) = \sum_{i=0}^{N} W_{P_{C_i}} * \frac{1}{Max(|cat(C)|, |cat(P_{C_i})|)} * \sum_{j=0}^{|cat(C)|} \sum_{k=0}^{|cat(P_{C_i})|} SimSem(cat_j(C), cat_k(P_{C_i})) \quad (6.1)$$

où :

$$SimSem(cat_j(C), cat_k(P_{C_i})) = \begin{cases} 1 & \text{si } cat_j(C) = cat_k(P_{C_i}) \\ 0 & \text{sinon} \end{cases}$$

avec $W_{P_{C_i}}$ le poids de la ressource P_{C_i} et $cat(X)$ l'ensemble des catégories de l'instance de concept X.

La définition 13 décrit le score de pertinence thématique pour l'ensemble des concepts d'un document, représenté par un vecteur de concepts $\vec{D}_C$.

Définition 13 (La mesure de similarité thématique)**.**

$$Sim_C(\vec{P}_C, \vec{D}_C) = \sum_{C}^{\vec{D}_c} \frac{SimSem(\vec{P}_c, C)}{|\vec{D}_c|}$$

où $\vec{P}_C$ et $\vec{D}_C$ sont respectivement les vecteurs d'instances de concept du profil opérationnel P et du document D et

$$Sim_C(\vec{P}_C, \vec{D}_C) \in \mathbb{Z}, 0 \leqslant Sim_C(\vec{P}_C, \vec{D}_C) \leqslant 1$$

Contrairement à la mesure de similarité mots-clés qui est basée sur une correspondance exacte des termes, permettant ainsi d'évaluer la couverture terminologique du besoin, celle entre les concepts est plus permissive. La figure 6.5, page 109, illustre cela au travers du scénario de vente de médicaments à base de produits illégaux (voir le second scénario page 72). Cette figure schématise la comparaison qui est faite entre les catégories des instances présentes dans la page collectée et les catégories des instances du profil. On constate que même si deux instances sont différentes, comme l'instance *Herbal_tonicsm* et *Herbalism*, la similarité n'est pas nulle puisqu'ils ont en commun une partie de leurs catégories. En se référant aux catégories, cette mesure de similarité permet de mettre en relation des concepts différents et ainsi d'évaluer la couverture thématique d'une page par rapport au besoin opérationnel.

c. Combinaison de critères

Le profil opérationnel est basé à la fois sur une représentation terminologique et une représentation thématique permettant de couvrir le besoin en information dans son ensemble. Les deux mesures précédentes permettent d'obtenir deux scores de similarité entre un document D et le profil opérationnel P. Ces deux mesures étant complémentaires, elles sont combinées au sein d'une seule mesure afin d'obtenir un score de pertinence global comme décrit dans la définition 14.

Définition 14 (La mesure de similarité adaptative)**.**

$$Sim(P, D) = \delta * Sim_C(\vec{P}_C, \vec{D}_C) + (1 - \delta) * Sim_K(\vec{P}_K, \vec{D}_K)$$

où $\vec{P}_C$ et $\vec{D}_C$ sont respectivement les vecteurs d'instances de concept du profil opérationnel P et du document D, $\vec{P}_K$ et $\vec{D}_K$ sont respectivement les vecteurs de mots-clés du profil opérationnel P et du document D et

$$Sim(P_{ci}, D) \in \mathbb{Z}, 0 \leqslant Sim(P_{ci}, D) \leqslant 1$$

La valeur de δ permet de donner plus d'importance à la couverture thématique ou à la couverture terminologique du besoin en pondérant respectivement ces deux scores.

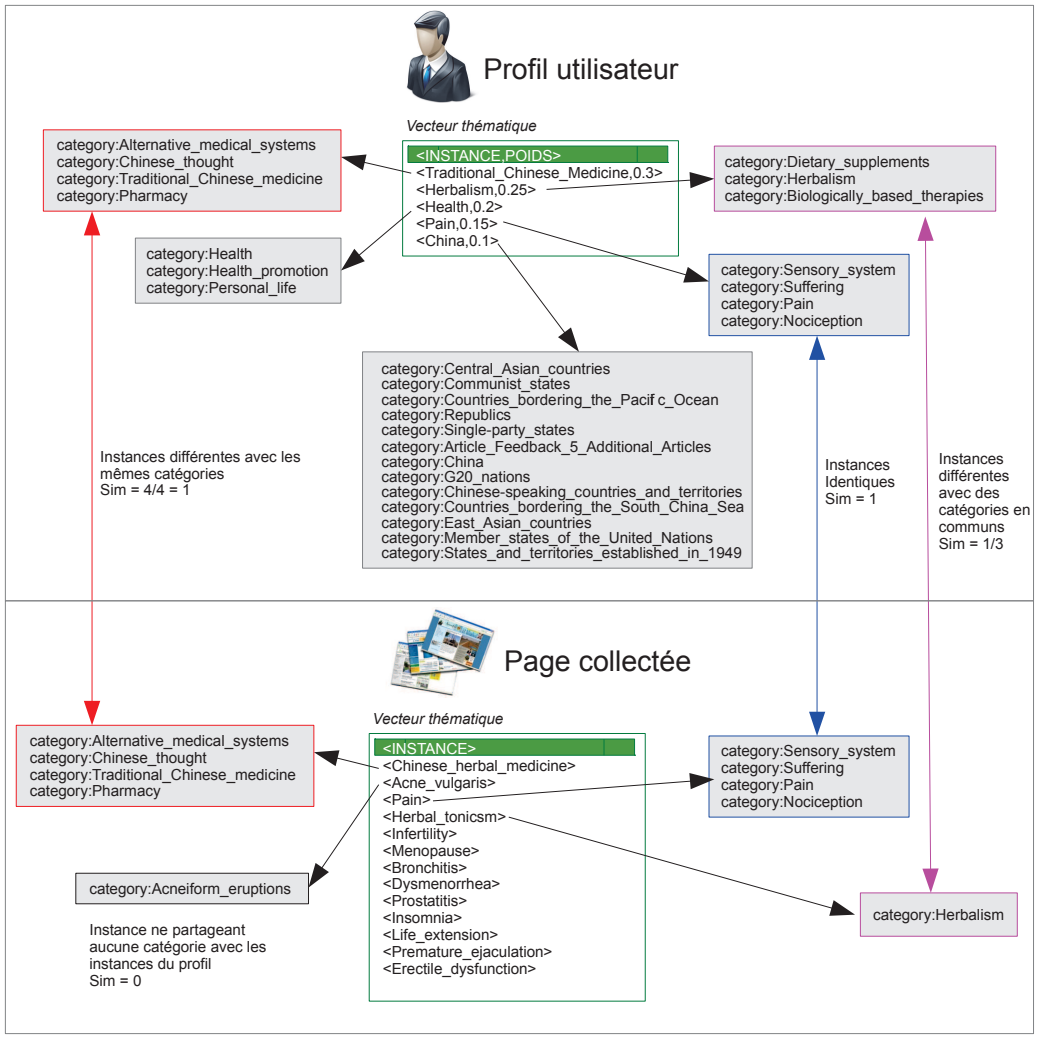


FIGURE 6.5 – Exemple d'exploitation des catégories par la mesure de similarité thématique

Les concepts sont utilisés pour représenter la thématique alors que les mots-clés couvrent plus précisément le besoin en définissant la terminologie du sujet de recherche. Il est donc logique d'accorder plus d'importance à la représentation terminologique du besoin et de considérer plus faiblement la thématique de la page afin de mieux cibler la pertinence d'une page. Ainsi, utiliser une valeur trop forte de δ permettra d'identifier des documents thématiquement proches du besoin mais dont le contenu ne répond pas précisément au besoin de l'utilisateur. À l'inverse, en utilisant une valeur faible pour δ, la mesure va permettre d'identifier uniquement les documents très spécifiques, au détriment de documents d'intérêt partageant moins de mots-clés avec le profil.

d. Adaptabilité de la mesure

La mesure de similarité utilisée dans DOWSER se veut adaptative. Elle doit être capable de prendre en compte l'évolution du besoin opérationnel de l'expert. En effet, celui-ci peut être intéressé dans un premier temps par des informations générales sur son sujet de recherche. Puis, au fur et à mesure qu'il analyse de l'information, son besoin peut se porter sur des détails ou sur des parties plus précises du sujet global. L'exemple de scénario opérationnel sur les sites de ventes de médicaments illicites, présenté page 72, peut illustrer ce changement de besoin. L'évolution du besoin peut, par exemple, suivre le cheminement suivant :

1. l'expert s'intéresse aux sites de ventes de médicaments illicites,
2. l'expert s'intéresse en particulier aux sites vendant des médicaments illicites dangereux,
3. l'expert s'intéresse plus particulièrement aux sites vendant des médicaments illicites dangereux à base de corne d'antilope.

Le retour de pertinence, présenté dans la section 5.3, page 81, permet de mettre à jour le profil opérationnel et, ainsi, de faire évoluer le besoin en information. La mesure de similarité globale s'appuie sur le profil opérationnel pour évaluer la pertinence d'une page collectée, et indirectement, guider la collecte. En faisant évoluer le profil (voir l'algorithme en section 5.3.3, page 94), la mesure de similarité s'adapte et fournit un score qui prend en compte le nouveau profil. L'exploration ciblée est directement impactée ce qui permet de collecter des pages répondant au nouveau besoin de l'expert.

L'adaptabilité de la mesure permet également de mettre à jour les pages déjà collectées. Un document stocké dans la base de connaissances est représenté par les concepts et les mots-clés extraits du contenu de celui-ci. Ils ont été utilisés pour fournir un score de pertinence au document. Lorsque le profil opérationnel évolue, ils peuvent être réutilisés afin de recalculer ce score de pertinence. Un document, considéré comme non-pertinent, peut le devenir si son contenu répond à la nouvelle définition du besoin en information de l'expert. Les sources d'intérêt présentées à l'utilisateur prennent donc en compte le besoin courant de l'expert, sans considération de la popularité d'une page, comme expliqué dans la section suivante.

e. Une mesure adaptée au contexte du ROSO

Notre mesure de similarité permet d'évaluer la pertinence d'une page au regard de son contenu. Les aspects de popularité, fortement utilisés dans les approches existantes,

notamment au travers de l'utilisation de la mesure HITS et PageRank (voir section 2.2.4, page 29) ne sont pas exploités dans notre approche. Comme expliqué dans nos exemples de scénarios opérationnels, les informations recherchées peuvent être sensibles et les sites Web contenant ces informations peuvent se faire discrets. Aussi, faire abstraction de la popularité dans notre mesure est un choix pertinent dans le contexte du ROSO qui permet de se distinguer des outils traditionnels. Comme la mesure est utilisée pour guider la collecte, cette dernière pourra explorer des zones du Web impopulaires mais qui contiennent de l'information pertinente pour le sujet de recherche courant. Les pages collectées sont présentées à l'utilisateur en considérant uniquement notre mesure de similarité. DOWSER peut donc retourner des pages Web qu'un moteur de recherche, par exemple, n'aurait pas indexées, ou désindexées, ou mal ordonnées dû à leur impopularité. Ce dernier processus de présentation des sources pertinentes découvertes à l'utilisateur est décrit dans la section suivante.

6.4 Stockage et présentation des sources découvertes

Une fois les pages Web collectées analysées, elles peuvent être stockées dans le système afin d'être présentées à l'utilisateur en fonction leur pertinence.

6.4.1 Stockage des documents

Seuls l'URL de la page, son titre, son résumé et les informations énumérées ci-dessus sont sauvegardés. Le document, correspondant à une page Web collectée, n'est conservé que le temps d'analyser son contenu et d'enrichir les informations le concernant. De manière à exploiter la sémantique présente dans les documents et dans les profils opérationnels, ces informations sont stockées dans une base de connaissances. Une ontologie a été construite afin de modéliser ces différentes informations d'intérêt pour notre système. Le schéma 6.6 présente les classes et leurs relations.

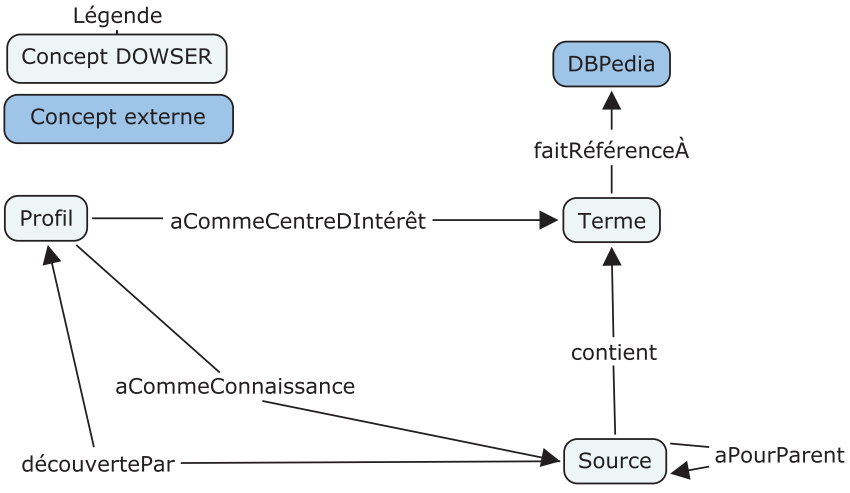


FIGURE 6.6 – Schéma des classes de l'ontologie DOWSER

La classe ***profil*** permet de modéliser le profil opérationnel présenté section 5, page 71. Elle est reliée à la classe ***source*** et à la classe ***terme*** pour respectivement modéliser les connaissances et les centres d'intérêt de l'expert. La classe ***source*** est reliée à la classe ***profil*** ce qui permet d'identifier la collecte d'où provient la source découverte. Une source est reliée à une autre pour modéliser les liens de parenté. Si une source n'est pas reliée à une source parente, c'est qu'elle fait partie des connaissances de l'expert ou qu'elle provient du module d'extension de la zone de collecte. Enfin, la classe ***terme*** permet de représenter les mots-clés et les concepts, ces derniers étant directement reliés au concept correspondant dans l'ontologie DBPedia. Les autres informations sont stockées sous forme d'attributs dans chacune des classes comme l'illustre les figures 6.7, 6.8 et , 6.9.

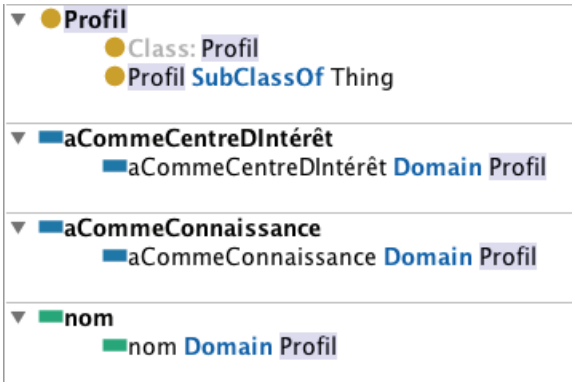


FIGURE 6.7 – La classe ***profil*** de l'ontologie DOWSER

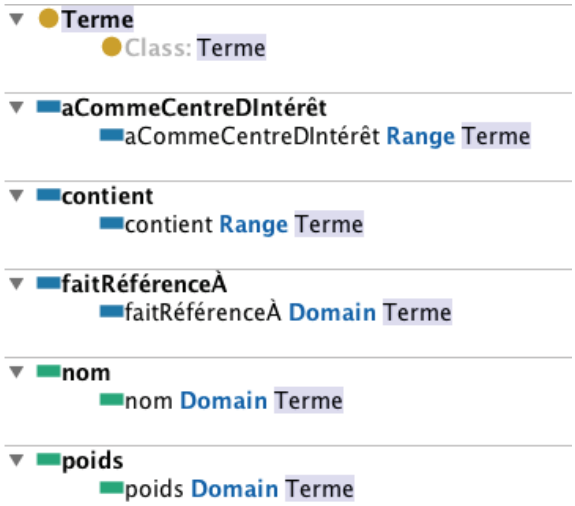


FIGURE 6.8 – La classe ***terme*** de l'ontologie DOWSER

L'utilisation d'une base de connaissances et d'une ontologie n'est pas obligatoire pour stocker et représenter les informations qui nous intéressent. Une base de données pourrait suffire si le but n'était que de créer des fiches descriptives des pages Web collectées. Cependant, l'utilisation d'une base de connaissances permet :

– de prendre en compte les évolutions du modèle contrairement à une base de données et à son schéma qui sont sensibles aux modifications. Ainsi, si notre schéma de classes évolue avec de nouvelles informations à sauvegarder sur les pages collectées,

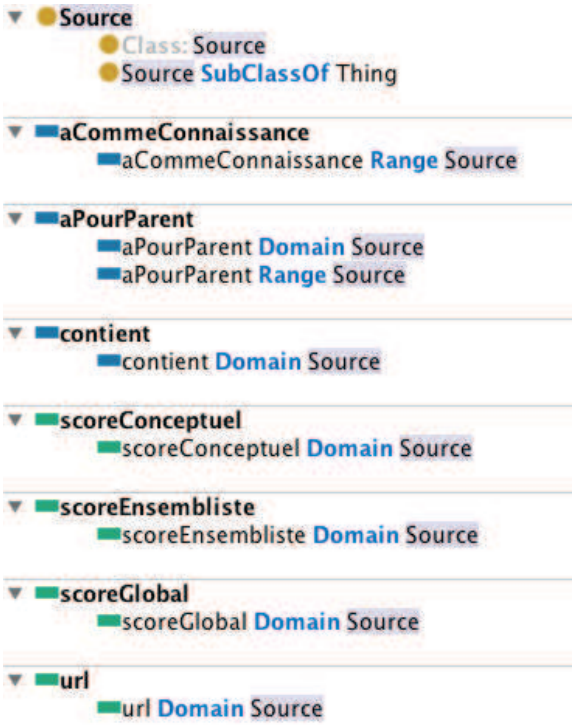


FIGURE 6.9 – La classe ***source*** de l'ontologie DOWSER

ou si notre modélisation du profil change, l'utilisation de la base de connaissances permettra au système de s'adapter plus facilement.

– l'interopérabilité avec d'autres ontologies. Notre représentation du contenu sous forme de mots-clés et de concepts peut être étendue afin, par exemple, de considérer l'ontologie Wookie [Serrano et al., 2012], utilisée par [Mombrun, 2012] qui évalue et modélise la qualité d'une source dans ses travaux. Cela peut permettre de distinguer, parmi les termes extraits, ceux de type place, personne, équipement, unité ou événement comme dans le pentagramme du renseignement (voir section 1.1.1, page 2).
– Les mécanismes de raisonnement propres aux bases de connaissances, comme les raisonnements déductifs, peuvent être utilisés dans le module d'extension de zone de collecte (voir section 6.2.2, page 103). L'objectif est de faire du rapprochement de profil opérationnel afin d'enrichir les nouvelles tâches de recherche avec des sources d'intérêt déjà collectées. Des sources jugées pertinentes par un utilisateur peuvent s'avérer d'intérêt pour un second utilisateur s'ils partageant des centres d'intérêt communs (mots-clés et concepts en commun dans le profil).

Enfin, le score et les liens présents dans le document sont envoyés vers la pile d'URLs à visiter. Le score de pertinence de la page définit l'importance des nouvelles URLs. La section suivante est consacrée à la mesure de similarité adaptative permettant d'évaluer la pertinence d'une page par rapport à un besoin opérationnel.

6.4.2 Présentation des sources découvertes

Une fois collectées et annotées, les sources découvertes sont présentées à l'expert en fonction de leur pertinence. Dans cette section, le mécanisme de présentation de ces sources est détaillé ainsi que l'interface homme-machine mise en place pour capturer le retour de pertinence.

a. Résultats en temps réel

Le processus de collecte permet de visiter des pages, de les analyser et de stocker leur URL ainsi que leur score de similarité entre leur contenu et le profil opérationnel. Ce score permet d'évaluer la pertinence d'une page par rapport au sujet de recherche de l'expert. Les sources découvertes peuvent donc être ordonnées en fonction de ce score de pertinence. DOWSER propose à l'utilisateur les 10 URLs des pages découvertes les plus pertinentes en affichant leur titre et un court résumé de leur contenu. Cela permet de ne pas déstabiliser l'expert qui retrouve une page de résultat qui s'apparente à celles des outils qu'il utilise habituellement. En effet, les trois populaires moteurs de recherche Google [4], Bing [5] et Yahoo [6] ont opté pour des pages de résultats contenant le titre et un court résumé du contenu des 10 liens jugés les plus pertinents par rapport à la requête de l'utilisateur.

DOWSER se distingue des moteurs de recherche car il ne nécessite pas de requête. En effet, il n'est pas nécessaire que le besoin soit une nouvelle fois exprimé. Il est déjà modélisé dans le profil utilisateur et c'est ce même profil qui a été utilisé par la mesure de similarité pour calculer la pertinence des pages collectées. Ainsi, DOWSER ordonne par pertinence l'ensemble des pages collectées et retourne directement à l'expert les 10 meilleures. Ces résultats peuvent être consultés à tout moment par l'expert pendant que le système continue d'explorer le Web. Dès qu'une page est collectée, sa pertinence est évaluée et le top 10 des pages les plus pertinentes est donc actualisé. L'utilisateur est averti lorsqu'une nouvelle page fait son apparition dans le top 10. Cette fonctionnalité représente un gain de temps intéressant pour les experts du renseignement qui peuvent être alertés à chaque nouvelle découverte. DOWSER est donc un système de veille et d'alerte en temps réel de découverte de nouvelles sources d'intérêt opérationnel. L'utilisateur peut consulter ces nouvelles sources et également les évaluer afin de fournir un retour de pertinence comme l'explique la section suivante.

b. Capture du retour de pertinence

Afin d'évaluer une source, l'expert peut consulter la page pointée par le lien fourni par le système DOWSER en cliquant sur le titre de celle-ci (voir illustration 6.10). En plus du titre et du court résumé fournis avec chaque lien du top 10 des sources découvertes, DOWSER met à la disposition de l'expert 3 différentes options d'évaluation afin qu'il juge de la pertinence des sources découvertes. Une source peut être considérée :

- soit d'intérêt ; la source contient des informations que l'expert recherchait,
- soit potentiellement intéressante ; la source contient des informations plus ou moins proches du sujet, ou déjà connues de l'expert, qui pourrait s'avérer d'intérêt plus tard,

4. https://www.google.fr/
5. http://www.bing.com/
6. http://search.yahoo.com/

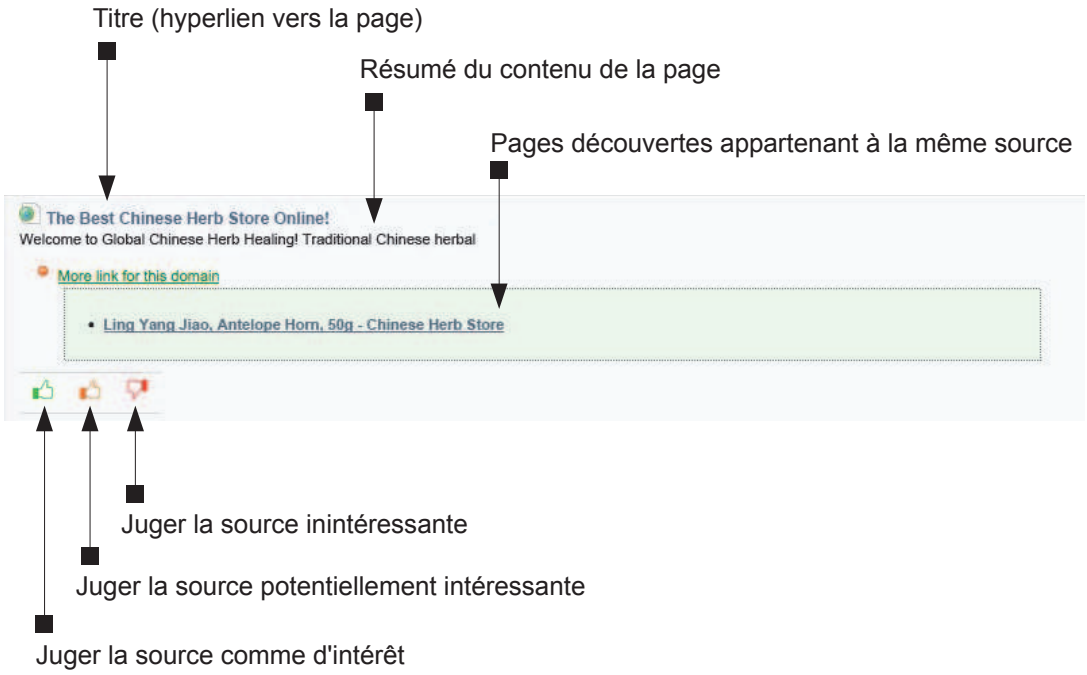


FIGURE 6.10 – Exemple de source retournée par DOWSER

– soit inintéressante ; la source ne contient pas d'information d'intérêt pour l'expert.

Ces trois options sont représentées par trois boutons différents illustrés dans la capture d'écran 6.10. On peut également y voir un bouton intitulé *"more link for this domain"* qui permet d'afficher des liens supplémentaires provenant du même domaine que le lien présenté. Le but est de permettre à l'expert de juger la source et pas seulement une page de celle-ci. En regroupant les pages jugées pertinentes par le système qui proviennent d'un même domaine, l'expert peut les consulter individuellement et donner un avis global sur la pertinence de la source. Les sources jugées comme pertinentes sont ajoutées à ses pages d'intérêt, c'est à dire comme connaissance dans son profil opérationnel. Les sources pouvant présenter de l'intérêt sont consultables dans un autre onglet de l'interface du système. L'expert peut y accéder à tout moment et changer son jugement sur ces sources en les notant pertinentes ou non pertinentes. Enfin, une source jugée non pertinente n'est plus présentée à l'expert. Elle est cependant conservée afin d'être exploitée par le système de retour de pertinence. Les sources jugées pertinentes sont également exploitées afin de mettre à jour le profil opérationnel comme expliqué dans la section 5.3, page 81.

6.5 Synthèse

Dans ce chapitre, un système de découverte de nouvelles sources pertinentes pour le ROSO a été détaillé. DOWSER est basé sur une exploration ciblée du Web et la prise en compte du besoin opérationnel présentée dans le chapitre précédent. Il collecte les pages Web qu'il estime d'intérêt pour le sujet de recherche courant et les transforme en documents qu'il enrichit d'informations permettant de calculer leur pertinence. Cette

dernière est évaluée via une mesure de similarité qui permet d'ordonner les pages collectées et de guider la collecte du Web. Les pages estimées d'intérêt par le système sont présentées à l'expert qui peut juger de leur pertinence.

DOWSER est donc un système de veille capable de considérer un besoin opérationnel et de découvrir de nouvelles sources d'intérêt répondant à ce besoin. Au travers de la mise en place des différents processus, des contributions théoriques ont été apportées :

- l'insuffisance de représentativité des requêtes a été comblée par l'utilisation d'un profil contenant une double représentation du besoin. L'ensemble du besoin est couvert thématiquement et précisément via l'utilisation de concepts et de mots-clés extraits des pages Web d'intérêt de l'expert. Ainsi, l'expression du besoin n'est plus restreinte à une vision unique et limitée fournie par l'expert.
- la linéarité du processus de recherche est remplacée par un processus dynamique de découverte de sources. Le traditionnel traitement linéaire "besoin - réponse" est remplacé par une boucle dynamique sous forme de "besoin - réponse - jugement de pertinence - nouveau besoin - etc.". Cette boucle est mise en oeuvre dans le système DOWSER et permet de guider la collecte tout en prenant en compte l'évolution du besoin opérationnel.
- le problème de la recherche d'une aiguille dans une botte de foin est résolu par la construction d'un corpus constitué uniquement de pages Web estimées pertinentes par le système. Cette pertinence est évaluée par le biais d'une mesure de similarité adaptative qui ne considère par la popularité des pages collectées. Ceci permet à DOWSER de trouver des pages discrètes sur des sujets sensibles qui sont souvent impopulaires. Enfin, le système d'alerte permet à l'expert de consulter les pages estimées pertinentes fraîchement découvertes.
- outre la mesure de similarité adaptative et le profil opérationnel, une ontologie a été proposée afin de modéliser les sources découvertes. Elle offre des perspectives d'enrichissement des informations représentatives des sources et de raisonnement déductif permettant d'améliorer les tâches de collecte via un mécanisme de collaboration transparent.

La partie suivante décrit l'architecture du système DOWSER et évalue les contributions théoriques avancées dans ce chapitre par le biais d'expérimentations.

Troisième partie

Expérimentations et mise en oeuvre

CHAPITRE 7

EXPÉRIMENTATIONS ET ÉVALUATION

Ce chapitre est dédié aux expérimentations menées durant cette thèse. Dans une première partie, nous présentons une validation expérimentale de notre approche. Son protocole et les objectifs d'évaluation y sont décrits. Une seconde expérimentation est ensuite introduite afin de valider nos propositions concernant le retour de pertinence dans notre système. Les résultats des expérimentations sont analysés afin de conclure sur nos contributions dans le domaine de la découverte de nouvelles sources d'intérêt opérationnel.

7.1 Validation expérimentale de notre approche

La première expérimentation a pour but d'évaluer notre modélisation du besoin (voir section 5, page 71) et sa capacité à guider le processus de découverte de sources (voir section 6, page 97).

7.1.1 Présentation de l'étude

Notre modélisation du besoin repose sur deux représentations vectorielles. La première utilise des mots-clés pour représenter un besoin précis alors que la seconde utilise des instances de concept DBPedia pour couvrir thématiquement le besoin. L'objectif est de couvrir l'ensemble du besoin opérationnel de l'expert en renseignement et d'exploiter ce profil au sein d'un processus d'exploration ciblée du Web pour collecter des pages d'intérêt. Afin de valider cette approche, décrite plus précisément dans le chapitre II, page 67, cette expérimentation doit permettre de répondre aux questions suivantes :

- Question 1 : Les mots-clés extraits des sources de l'utilisateur représentent-ils une partie de son besoin ?
- Question 2 : Les instances de concept extraits des sources de l'utilisateur représentent-ils une partie de son besoin ?
- Question 3 : Les mots-clés et les concepts extraits des sources de l'utilisateur représentent-ils l'ensemble de son besoin ?

– Question 4 : L'utilisation des mots-clés pour la mesure de similarité permet-elle de guider la collecte et de trouver des sources pertinentes ?
– Question 5 : L'utilisation d'instances de concept pour la mesure de similarité permet-elle de guider la collecte et de trouver des sources pertinentes ?
– Question 6 : L'exploration ciblée, basée sur notre modélisation du profil opérationnel, permet-elle à l'expert de découvrir des pages d'intérêt ?

7.1.2 Protocole expérimental

Le protocole décrit en détail le contexte de l'expérimentation : le corpus mis en jeu, les utilisateurs participants à l'évaluation et le déroulement étape par étape de l'expérimentation. Les résultats sont analysés dans la section suivante afin de valider notre approche.

a. Utilisateurs, corpus et thématique de recherche

Les utilisateurs participants à cette expérimentation sont 20 personnes aguerries à la DI sur Internet et habituées aux activités de veille informationnelle et technologique. Même s'il ne s'agit pas d'experts du renseignement, notre système est adaptable à leur contexte de travail dans lequel ils sont souvent confrontés à de la DI sur des sujets spécifiques. Leur expertise dans ce domaine est suffisante pour manipuler, tester et évaluer notre système.

Le système DOWSER est conçu pour explorer le Web et y découvrir des pages d'intérêt opérationnel. L'utilisation d'un corpus fermé de documents pour notre expérimentation biaiserait l'évaluation de le processus de découverte de sources. En effet, la notion de découverte doit prendre en considération les spécificités du Web : son importante taille et l'évolution constante de son contenu. Durant l'expérimentation, le périmètre d'action de notre système est préservé. Notre corpus est donc le Web et toutes les pages Internet accessibles par URL. Cela permet de se rapprocher des conditions d'utilisation réelles de notre système.

Dans cette optique, aucune contrainte sur la thématique de recherche n'est imposée. Le système doit pouvoir s'adapter à des sujets sensibles ou spécifiques comme à des sujets classiques. L'utilisateur est donc libre de choisir le sujet de recherche qu'il souhaite. La seule restriction imposée à l'utilisateur est de se limiter à des pages d'intérêt en anglais pour représenter son besoin en information. Cette limite est mise en place afin que l'outil DBPedia Spotlight, qui est configuré pour extraire des concepts DBPedia depuis des textes en anglais, puisse fonctionner correctement.

b. Présentation des fonctionnalités évaluées

Seulement une partie des fonctionnalités de l'approche DOWSER est utilisée durant cette expérimentation afin de mettre en avant les points à évaluer pour répondre aux différentes questions posées dans la présentation de l'étude.

Parmi les modifications, on note que le vecteur de mots-clés et le vecteur de concepts servent à modéliser deux profils différents par utilisateur. Le but est de les confronter et de juger de leur capacité à couvrir le besoin opérationnel indépendamment l'un de l'autre. Dans cette optique, le processus de découverte de sources, qui est initialement basé sur un seul profil opérationnel, est également modifié. Deux collectes distinctes parcourent le Web. Ainsi, la première collecte utilise le profil contenant le vecteur de concepts et la mesure

de similarité pour la couverture thématique. La seconde collecte utilise le profil contenant le vecteur de mots-clés et la mesure de similarité pour la couverture terminologique. Cela offre la possibilité d'évaluer à la fois la couverture du besoin et la pertinence des pages découvertes via l'approche conceptuelle et via l'approche mots-clés. Enfin une troisième collecte est également lancée. C'est une collecte en largeur (voir section 2.2.2, page 26) qui n'est pas ciblée, et qui n'utilise donc aucun profil. Elle sert de référence afin de juger de la pertinence de nos collectes ciblées par rapport à une collecte classique.

L'extension de la zone de collecte, permettant de trouver plus rapidement les grappes thématiques, n'est pas présente dans cette version de DOWSER. Seules les pages d'intérêt fournies par l'utilisateur sont utilisées pour démarrer la collecte. Le but est de juger si ce point de départ est viable et permet de découvrir de nouvelles sources même sans le module d'extension de la zone de collecte.

Une interface homme/machine a été créée sur mesure pour cette expérimentation. Elle permet à l'utilisateur d'initialiser son profil en fournissant un ensemble d'URLs pointant vers des pages représentatives de son besoin en information et de juger les pages retournées par le système. Cette interface est différente de l'interface du système DOWSER dans la mesure où elle permet à l'utilisateur de visualiser et d'évaluer son profil.

La présentation des résultats à l'utilisateur a également subi des modifications afin de s'adapter aux trois collectes différentes de ce prototype. Afin de comparer la pertinence des trois collectes, la page de résultats mélange les sources découvertes par ces collectes. Au total, 15 sources sont présentées à l'utilisateur dans lesquelles on retrouve :

1. les 5 sources les plus pertinentes de la collecte ciblée utilisant les instances DBPedia ;
2. les 5 sources les plus pertinentes de la collecte ciblée utilisant les mots-clés ;
3. et 5 sources de la collecte classique prises au hasard.

Enfin, la capture du retour de pertinence a également subi une modification. Contrairement à un contexte opérationnel, il n'est pas nécessaire de juger une source "potentiellement intéressante" dans la mesure où même si elle peut s'avérer d'intérêt plus tard, l'expérimentation ne prévoit pas de retour de pertinence sur le long terme. Aussi, si les jugements "pertinente" et "non pertinente" sont conservés, le jugement "potentiellement intéressante" est remplacé par "peu pertinente" et "assez pertinente" permettant d'évaluer et de mesurer plus précisément la pertinence d'une source par rapport au besoin informationnel de l'utilisateur.

c. Déroulement de l'expérimentation

L'expérimentation comprend plusieurs étapes, de l'inscription de l'utilisateur jusqu'au jugement des pages collectées en passant par la construction de profil. Le déroulement de l'expérimentation est décrit étape par étape dans cette partie.

Première étape : exploration. La première étape consiste à se connecter au prototype, à créer son profil et à lancer les processus d'exploration.

1. L'utilisateur se rend sur le site, crée un compte utilisateur et se rend sur l'onglet de construction du profil utilisateur intitulé "My Profile".
2. L'utilisateur fournit au système des URLs représentant son besoin informationnel via la vue "User Profile". D'un point de vue opérationnel, l'expert du renseignement

a habituellement un nombre relativement limité de pages d'intérêt sur son sujet de recherche. Il est demandé à l'utilisateur de fournir au minimum 5 URLs afin d'avoir un contenu textuel assez important pour extraire les vecteurs d'instances de concept DBpedia et de mots-clés représentatifs de son besoin informationnel. L'utilisateur doit également renseigner une requête, similaire à celle qu'il taperait sur un moteur de recherche, représentant ce besoin en information.

3. L'utilisateur lance la construction de son profil via cette même page. Il patiente quelques secondes le temps que la construction du profil prenne fin. Une fois la construction du profil terminée, le système propose deux listes à l'utilisateur. L'une contient 20 concepts et l'autre 20 mots-clés utilisés pour représenter le besoin utilisateur. La taille des listes permet d'avoir un nombre suffisamment important de termes extraits mais c'est aussi un nombre raisonnable pour une évaluation manuelle des termes par un utilisateur. Afin d'éviter tout biais concernant la présentation des deux listes, la première liste à être présentée est tirée aléatoirement. De même, l'ordre des termes dans les listes est aléatoire, il n'est pas représentatif du poids du terme dans le vecteur. L'utilisateur est alors invité à évaluer les instances de concept DBPedia et les mots-clés en indiquant pour chaque terme s'il le juge inintéressant, optionnel, connexe ou intéressant par rapport à son besoin en information (voir figure 7.1).

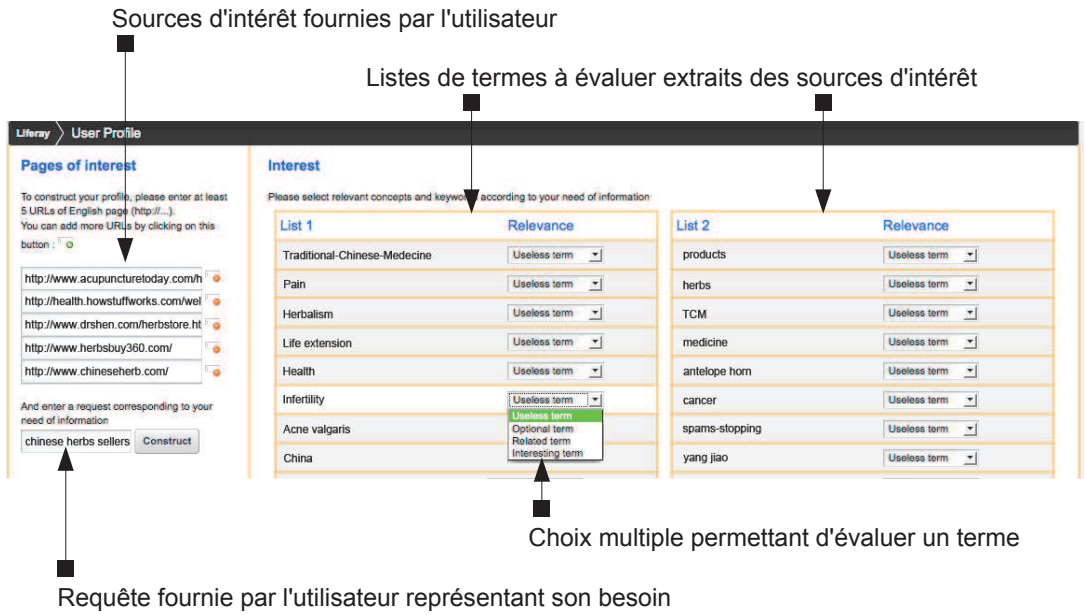


FIGURE 7.1 – Impression d'écran : construction du profil DOWSER

4. Une fois le profil construit et évalué par l'utilisateur, celui-ci peut lancer les processus de collecte décrits précédemment. Cette opération se fait via un simple bouton "Valider" présent en bas de page.

Au lieu d'utiliser le système d'alertes présenté dans la section 6.4.2, page 114, les processus de collecte sont lancés pendant 12h. L'utilisateur est prévenu par courriel lorsque la collecte est terminée. Cette alternative a été mise en place pour faire gagner du temps à l'utilisateur. Il n'a ainsi pas besoin de se reconnecter sur le prototype à chaque fois qu'une page d'intérêt est découverte.

Seconde étape : évaluation des sources découvertes. Au bout de 12h, les collectes sont arrêtées automatiquement et la seconde étape de l'expérimentation, qui consiste à juger la pertinence des pages collectées, peut démarrer. L'utilisateur se rend sur l'onglet de résultat intitulé "Collect Result". Il consulte une par une les 15 sources (les 5 sources les plus pertinentes de la collecte ciblée utilisant les instances DBPedia, les 5 sources les plus pertinentes de la collecte ciblée utilisant les mots-clés et 5 sources de la collecte classique prises au hasard) proposées par le prototype. Il les évalue individuellement en indiquant si elles sont non pertinentes, peu pertinentes, assez pertinentes ou pertinentes en fonction de son besoin informationnel (voir figure 7.2). Après avoir jugé les pages découvertes,

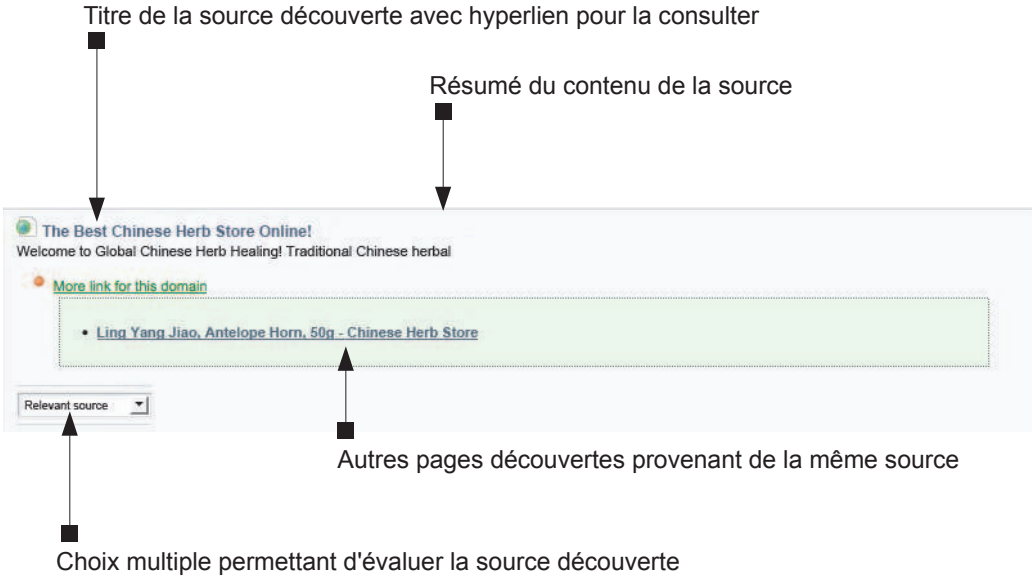


FIGURE 7.2 – Impression d'écran : évalution d'une source découverte

l'utilisateur est invité à répondre à un questionnaire dont le contenu ainsi que les réponses aux questions sont présentés parmi les résultats dans la section suivante.

7.1.3 Résultats et discussion

L'ensemble des utilisateurs a terminé l'expérimentation permettant ainsi d'obtenir 1 jeu de données de 20 utilisateurs exploitable pour l'analyse des résultats. Les sujets de recherche des utilisateurs sont variés, comme le montre les quelques exemples suivants :

- les émeutes ethniques liées au taux de chômage à Conakry en Guinée,
- la guerre au Mali,
- la sexualité du cougar.

Parmi les diverses thématiques de recherche, on trouve des besoins informationnels spécifiques, des sujets d'actualité, et potentiellement des termes ambigus. Dans le questionnaire de fin d'expérimentation, l'utilisateur est invité à renseigner le(s) type(s) de sa recherche parmi les possibilités suivantes : "sujet sensible", "sujet d'actualité", "sujet spécifique" et "sujet générique". Si son sujet de recherche ne rentre dans aucun de ces quatre critères, il peut expliciter manuellement le type de recherche qu'il a effectué. Les réponses à cette question sont présentées dans la figure 7.3. Elles montrent qu'une grande majorité des

sujets de recherche sont spécifiques. Une plus faible part correspond à des sujets d'actualité. On trouve également des sujets génériques et/ou sensibles en plus faible quantité. Cette diversité est intéressante pour notre expérimentation puisqu'elle permet d'évaluer le prototype DOWSER avec des types de besoin différents.

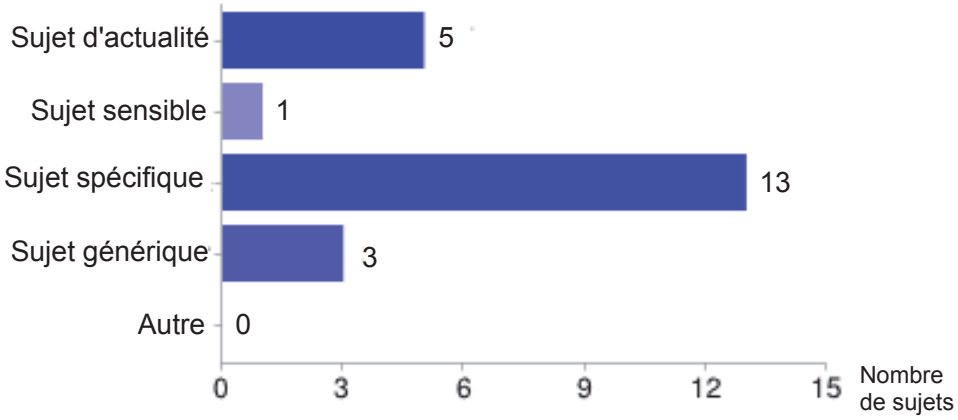


FIGURE 7.3 – Les types de sujet de recherche de l'expérimentation

a. Évaluation du profil opérationnel

L'utilisateur évalue les instances de concept et les mots-clés extraits des pages pointées par les URLs d'intérêt qu'il a fourni au système. Son jugement sur l'extraction de ces termes est représenté par la figure 7.4. Le nombre de concepts et de mots-clés jugés par l'utilisateur comme ayant un intérêt pour représenter le besoin informationnel est quasiment le même : 36% des instances de concept proposées ont été sélectionnées, contre 39% pour les mots-clés. Le nombre d'instances de concept jugées intéressantes, connexes ou optionnelles est très proche. Ce résultat illustre la capacité des concepts à couvrir thématiquement le besoin. Par ailleurs, la majorité des utilisateurs jugent les mots-clés extraits comme étant intéressants ou connexes au sujet de recherche et ils les considèrent plus pertinents que les concepts pour représenter le besoin. Ce résultat tend à soutenir notre approche qui exploite conjointement les mots-clés et le concepts : les mots-clés couvrent précisément le besoin avec une majorité de mots-clés jugés intéressants alors que les concepts couvrent plus thématiquement le besoin avec des concepts à la fois intéressants, connexes et optionnels.

Dans le questionnaire de fin d'expérimentation, il est demandé à l'utilisateur d'évaluer globalement les termes qui lui sont proposés au travers de la question suivante : *Quelle est la pertinence des instances de concept et des mots-clés proposés par rapport à votre besoin en information ?*. La figure 7.5 illustre les réponses à cette question. Seulement un quart des utilisateurs trouve les termes proposés, que ce soit des concepts ou des mots-clés, pertinents pour représenter leur besoin en information (voir figure 7.6, page 125). Ce résultat, associé au taux de sélection des termes (36% des concepts et 39% des mots-clés), montre que la construction automatique du profil dans l'état est insuffisante pour couvrir l'intégralité du besoin informationnel. Cette information est validée par l'utilisateur au travers de la question suivante : "Les termes proposés sont-ils suffisants pour couvrir

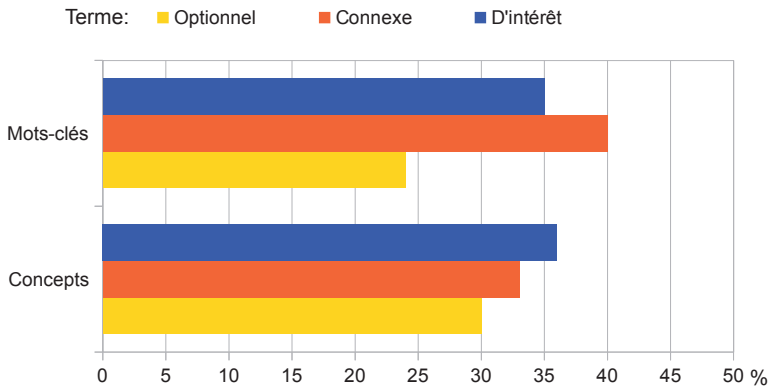


FIGURE 7.4 – Taux de pertinence des termes proposés

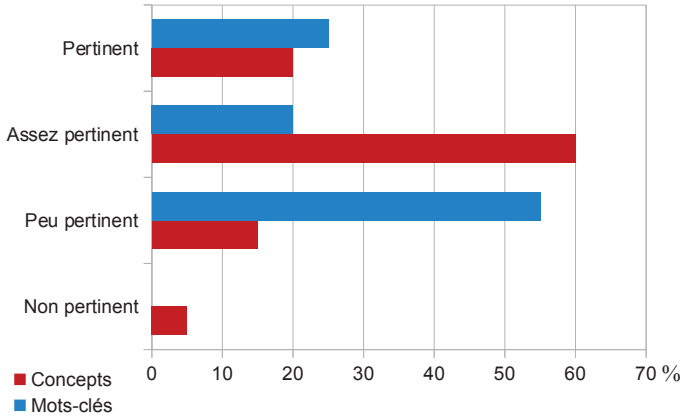


FIGURE 7.5 – Taux de pertinence globale des listes proposées

votre besoin informationnel ?". La majorité répond que filtrer les termes non pertinents est nécessaire mais qu'il faudrait également pouvoir en ajouter des non proposés.

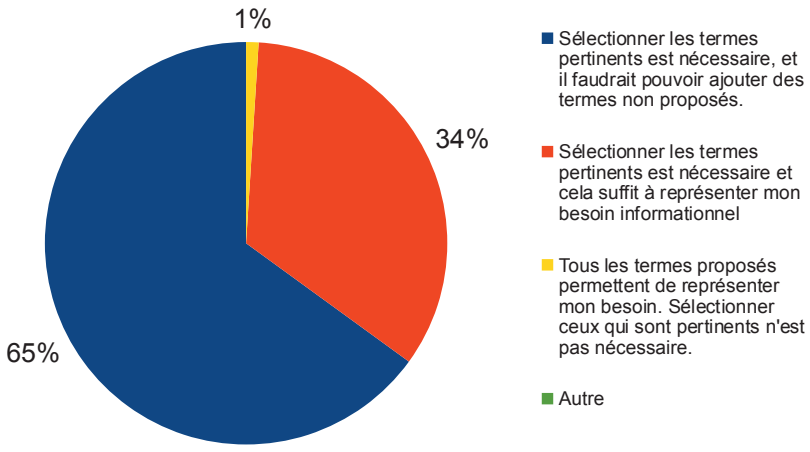


FIGURE 7.6 – Complétude des termes proposés

L'évaluation proposée ci-dessus montre une certaine lacune du prototype à couvrir l'ensemble du besoin informationnel de l'utilisateur. Ces résultats peuvent venir de notre approche : la condition opérationnelle imposée, qui consiste à exploiter des sources d'intérêt de l'utilisateur pour constituer son profil, peut être la cause de ce problème. Il peut être dû également aux outils utilisés pour extraire les concepts et les mots-clés. En effet, l'efficacité de ces outils peut affecter et impacter nos résultats. Pour répondre aux questions posées en début d'expérimentation (voir page 119), les instances de concept et les mots-clés permettent de couvrir une partie du besoin informationnel (question 1 et 2). Cependant, ils ne permettent pas de couvrir l'ensemble de ce besoin (question 3). Ce résultat a conduit à intégrer, au sein du système DOWSER, le processus de retour de pertinence, présenté dans la section 5.3, page 81. Ce processus prend en considération les termes présents dans les pages jugées pertinentes et dans celles jugées non pertinentes par l'utilisateur afin d'affiner la représentation du besoin opérationnel.

Malgré une couverture du besoin informationnel partielle, les instances de concept et les mots-clés jugés pertinents par l'utilisateur permettent d'initialiser la modélisation de son besoin utile au processus de découverte de sources. La section suivante présente les résultats obtenus par le processus d'exploration ciblée de DOWSER basée sur ce profil.

b. Évaluation de la collecte de pages Web

La pertinence des pages Web collectées par le système DOWSER est évaluée par l'utilisateur. Au travers de son jugement, c'est la pertinence de notre mesure de similarité basée sur le profil utilisateur qui est évaluée. Pour rappel, l'utilisateur consulte et juge 15 sources issues des trois collectes lancées pendant l'expérimentation (voir section 7.1.2, page 120). La figure 7.7, page 127, illustre les résultats obtenus : 91% des sources fournies par la collecte exploitant les mots-clés et 83% des sources fournies par la collecte exploitant les concepts sont jugées comme des sources d'intérêt par l'utilisateur. Seulement 25% des sources aléatoires présentées à l'utilisateur obtiennent un jugement positif. Ce résultat permet de conclure que notre mesure de similarité, qu'elle soit terminologique ou thématique, est capable de retourner des sources d'intérêt à l'utilisateur. On note que l'approche utilisant les mots-clés en retourne cependant un plus grand nombre. Ce résultat va dans le sens de notre approche : les mots-clés couvrant plus précisément le besoin informationnel que les concepts, le nombre de pages d'intérêt découvertes via ces mots-clés, est plus important. C'est une des raisons pour laquelle les moteurs de recherche classiques utilisent les mots-clés dans leur approche. L'utilisation des concepts permet de globalement mieux cibler la thématique et ainsi de couvrir une plus large portion du besoin au détriment de la précision. Cela se voit par la répartition très proche de sources intéressantes, assez intéressantes et peu intéressantes fournies par la collecte basée sur les concepts.

Si les mesures de similarité thématique et terminologique permettent de retourner des sources d'intérêt à l'utilisateur en exploitant le profil construit en amont, permettent-elles de guider la collecte (question 4 et 5, voir page 119) ? La section suivante présente l'évaluation de notre système d'exploration du Web afin de répondre à ces deux questions.

c. Évaluation de l'exploration ciblée du Web

Les deux premières évaluations de cette expérimentation mettent en exergue que même si le profil utilisateur ne couvre pas l'ensemble du besoin informationnel et que l'utilisateur

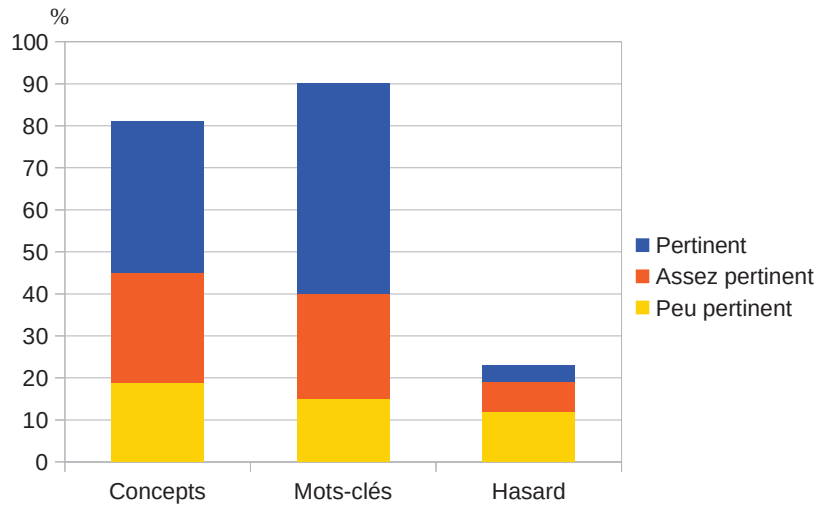


FIGURE 7.7 – Taux de pertinence des sources collectées présentées à l'utilisateur

le juge perfectible, les mesures de similarité qui l'exploitent permettent de retourner des sources d'intérêt. Ces mesures doivent également permettre au processus de collecte du système de cibler des pages d'intérêt et de les collecter en priorité. Cette information nous est donnée en calculant le score de similarité moyen des pages collectées en fonction du temps. Les courbes obtenues par les différentes collectes sont représentées dans la figure 7.8, page 128, et dans la figure 7.9, page 129. Dans les deux figures, le score moyen des pages collectées par la collecte classique (non ciblée) est comparé à celui des pages collectées par la collecte ciblée. Que ce soit en exploitant des mots-clés ou des concepts, la collecte ciblée collecte des pages Web avec un meilleur score de similarité que la collecte classique. Cette donnée valide l'utilisation du Topical Locality Phenomenon appliquée à nos deux mesures de similarité.

On note que les premières pages collectées par la collecte classique ont en moyenne un score de similarité meilleur que celui de la collecte ciblée. Ceci s'explique par les algorithmes de collecte utilisés. La collecte classique, qui est une exploration en largeur, visite toutes les pages proches des sources d'intérêt fournies par l'utilisateur, qu'elles appartiennent au même domaine ou non. Elle se concentre donc sur la même grappe thématique au début de l'exploration avant de diverger. A contrario, la collecte ciblée donne une priorité plus importante aux liens à explorer qui pointent vers un nom de domaine inexploré. Elle s'éloigne donc plus rapidement des pages d'intérêt fournies par l'utilisateur pour découvrir de nouvelles grappes thématiques. Ceci explique pourquoi la collecte classique a un meilleur score moyen de similarité au début de l'exploration. Afin d'évaluer l'efficacité de la collecte, il faut donc considérer le nombre de pages collectées présentant un intérêt pour le besoin utilisateur plutôt que le score moyen de toutes les pages collectées. Autrement dit, il s'agit d'écarter les pages collectées jugées non pertinentes par le système. Les figures 7.10, page 130, et 7.11, page 131 représentent donc le nombre de pages d'intérêt collectées par les deux processus de collecte ciblée par rapport à la collecte classique. L'efficacité d'une collecte est assimilée à sa capacité à collecter le plus grand nombre de pages d'intérêt, et ce, le plus rapidement possible.

Les deux collectes ciblées sont plus performantes que la collecte classique. C'est d'au-

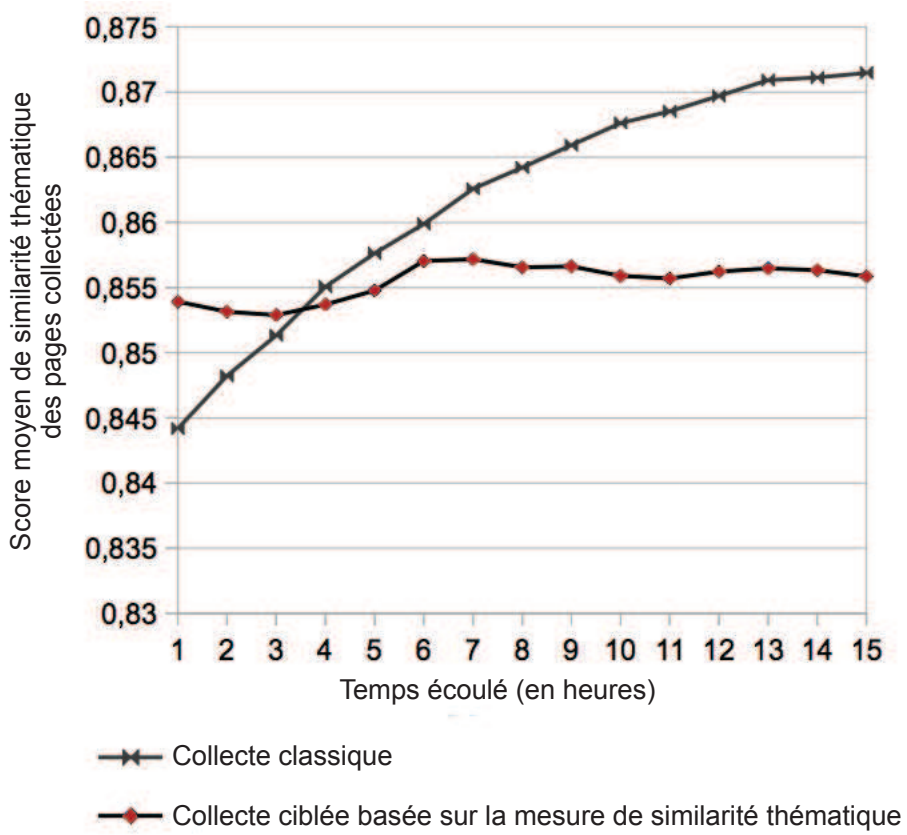


FIGURE 7.8 – Score moyen de similarité thématique des pages collectées en fonction du temps

tant plus vrai pour la collecte ciblée basée sur la mesure de similarité utilisant les concepts qui découvre un nombre bien plus important de sources sur la thématique de recherche que la collecte ciblée mots-clés. Cette performance se justifie par notre mesure de similarité qui utilise la proximité entre deux concepts (voir section 6.3.2, page 106). L'utilisation de concepts dans la mesure de similarité permet de couvrir plus largement le besoin utilisateur. De ce fait, un plus grand nombre de pages collectées sont classées comme d'intérêt par le système.

Cette évaluation montre que le prototype est en mesure de collecter des sources d'intérêt efficacement et de les fournir à l'utilisateur. Cependant il reste à déterminer si l'utilisation du prototype DOWSER est indispensable pour trouver ces sources et si elles apportent de l'information supplémentaire à l'utilisateur par rapport aux sources fournies par les moteurs de recherche. Autrement dit, il s'agit d'évaluer la pertinence et la capacité de notre approche à découvrir de nouvelles sources d'intérêt opérationnel.

d. Évaluation de la pertinence et de la découverte de nouvelles sources

Pour les besoins de l'expérimentation, une requête représentant son besoin a été demandée à l'utilisateur lors de la construction de son profil. Elle est utilisée afin d'évaluer

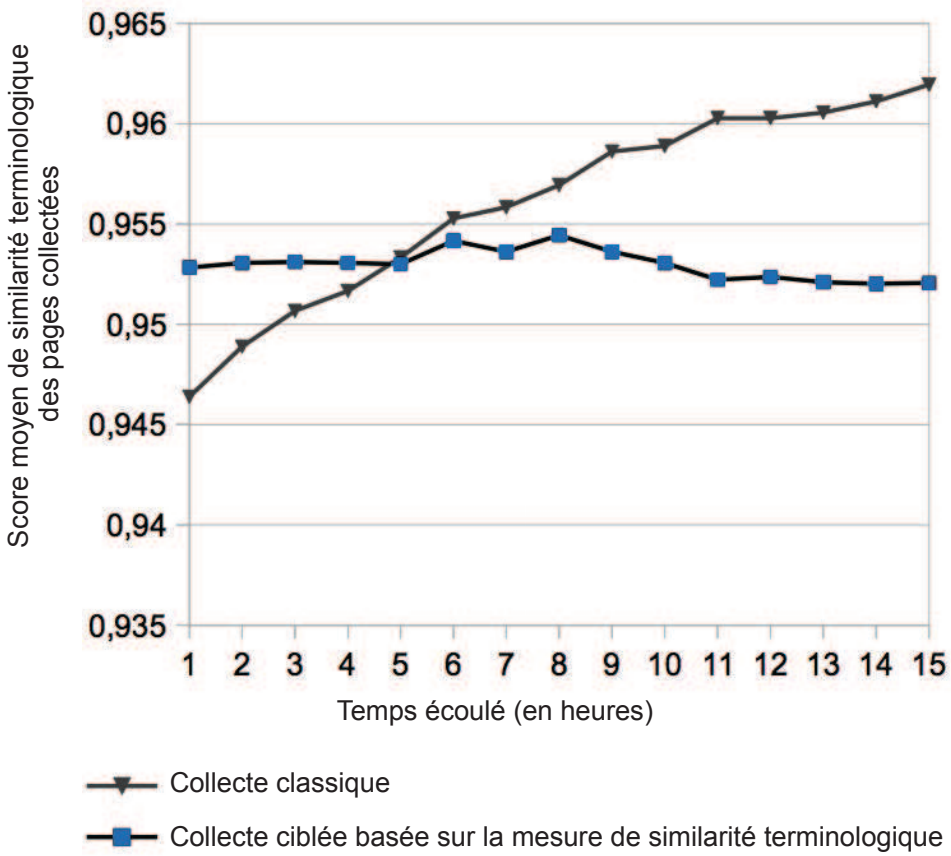


FIGURE 7.9 – Score moyen de similarité terminologique des pages collectées en fonction du temps

la capacité de découverte du système DOWSER. En parallèle du processus de collecte, la requête est envoyée aux principaux moteurs de recherche existants : Google [1], Yahoo ! [2] et Bing [3]. Ces trois moteurs de recherche étaient les plus populaires en avril 2013 d'après l'étude eBizMBA [4]. Le top 100 des URLs retournées par ces trois moteurs de recherche, correspondant aux 10 premières pages de résultats, est sauvegardé temporairement pour chaque requête. Sur l'ensemble des pages retournées par DOWSER et jugées comme ayant de l'intérêt par l'utilisateur (175 sources), 77,4% provenant de la collecte basée concepts et 81,5% provenant de celle basée mots-clés, ont un nom de domaine non partagé avec les pages du top 100 Google. Le tableau 7.1, page 130, montre des résultats similaires avec Yahoo ! et Bing. Ces résultats mettent en avant la capacité de DOWSER à découvrir des sources d'intérêt que l'utilisateur n'aurait pas pu trouver en utilisant sa requête sur l'un des trois moteurs de recherche. C'est le résultat attendu avec notre approche qui collecte et ordonne les pages indépendamment de leur popularité contrairement à ces moteurs de recherche. Cependant, si la majorité des sources proposées par DOWSER n'apparaissent

1. http://www.google.com/
2. http://www.yahoo.com/
3. http://www.bing.com/
4. http://www.ebizmba.com/articles/search-engines

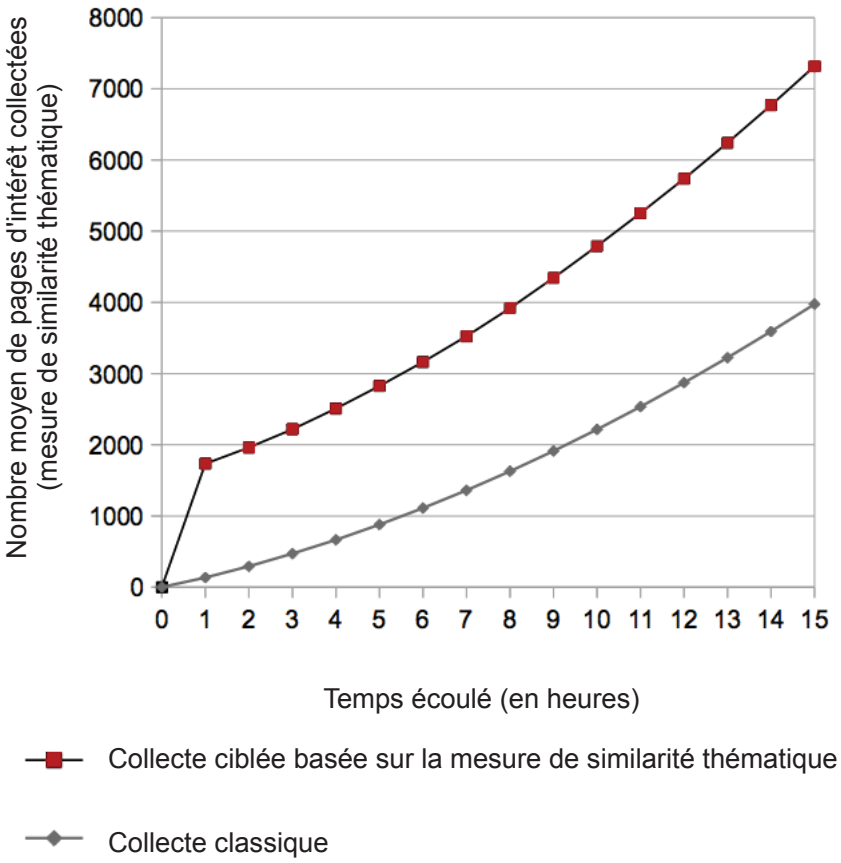


FIGURE 7.10 – Nombre de pages d'intérêt collectées en fonction du temps (mesure de similarité thématique)

TABLE 7.1 – Pourcentage moyen des pages découvertes par DOWSER

Pages d'intérêt de la collecte avec mesure de similarité thématique	Pages d'intérêt de la collecte avec mesure de similarité terminologique	
77,4%	81,5%	des domaines non présents dans le top 100 Google
77,9%	79,8%	des domaines non présents dans le top 100 Bing
85,7%	82,4%	des domaines non présents dans le top 100 Yahoo !

pas dans les résultats des moteurs de recherche, cela peut également s'expliquer :

- soit par l'expression d'une requête, par l'utilisateur, qui n'est pas représentative de son besoin : les résultats des moteurs de recherche ne peuvent donc pas être pertinents et on ne peut pas y retrouver les sources proposées par DOWSER ;
- soit par un jugement biaisé de l'utilisateur concernant les sources proposées par

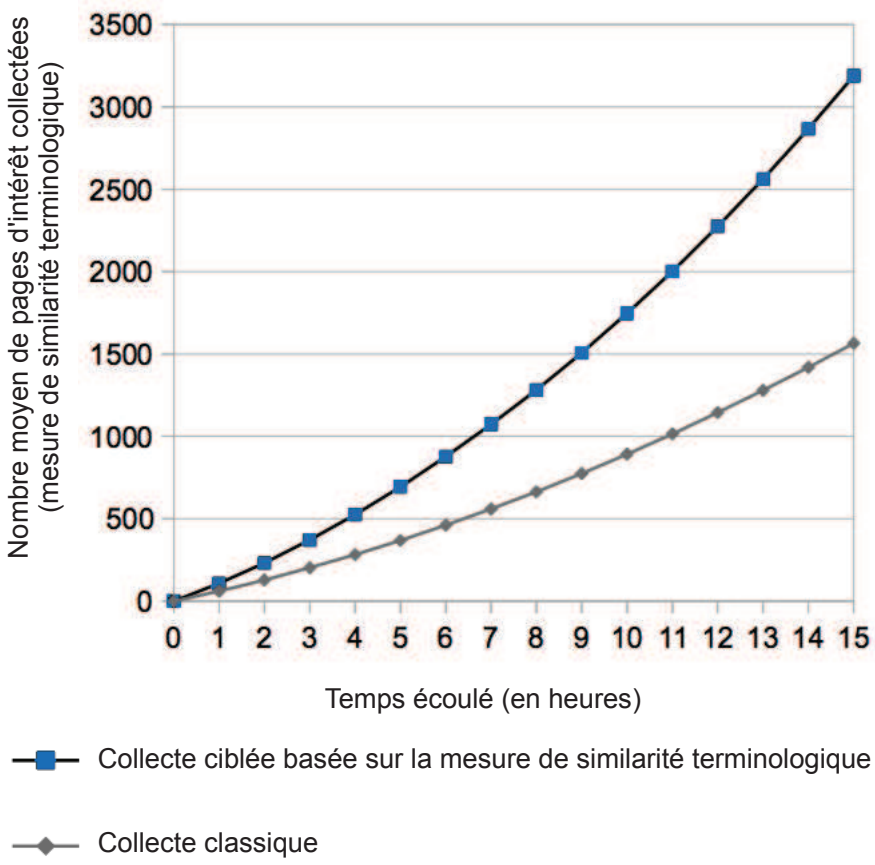


FIGURE 7.11 – Nombre de pages d'intérêt collectées en fonction du temps (mesure de similarité terminologique)

DOWSER puisque les résultats de notre système et des moteurs de recherche ne sont pas confrontés.

Cette dernière possibilité sous-entend que les sources fournies par DOWSER, et jugées d'intérêt par l'utilisateur, peuvent être moins pertinentes que les pages proposées par les moteurs de recherche. Elles ne sont pas donc pas retournées par ces derniers.

Afin d'écarter ces deux cas de figure et de juger de la pertinence de DOWSER à découvrir de nouvelles sources, une dernière évaluation est menée. Elle vise à comparer la pertinence des résultats fournis par les moteurs de recherche avec ceux fournis par DOWSER. Il est demandé à l'utilisateur de noter entre 0 et 10 une liste de pages Web sachant que la note 0 signifie que la page est non pertinente, et inversement pour la note 10. Cette liste se compose des 5 premières URLs retournées par Google et des 5 meilleurs URLs de pages découvertes par DOWSER. L'ensemble des 10 URLs est mélangé et l'utilisateur n'a pas connaissance de la provenance du lien. La figure 7.13 illustre les résultats obtenus. La note moyenne donnée par les utilisateurs aux URLs découvertes par DOWSER est de 6 contre 4,6 pour les URLs fournies par Google. Cette tendance est confortée par les résultats de la figure 7.12. Dans cette dernière, la mesure globale de DOWSER est utilisée sur l'ensemble des pages pointées par les 10 URLs. La pertinence des pages découvertes par

DOWSER y est également meilleure. Pour valider ces résultats, un intervalle de confiance à 95% a été calculé. Les intervalles obtenus ne se chevauchant pas, la différence entre les résultats de DOWSER et Google est significative dans cette expérimentation. Les notes données aux pages Google ne sont pas nulles, ce qui implique que la requête de l'utilisateur n'est pas inadéquate à son besoin. Enfin, puisque l'utilisateur juge à la fois les pages de Google et celles de DOWSER, il n'y a plus de biais concernant son jugement. Ainsi, la capacité de DOWSER à découvrir des sources d'intérêt est donc validée (question 6, voir section 7.1.1, page 119).

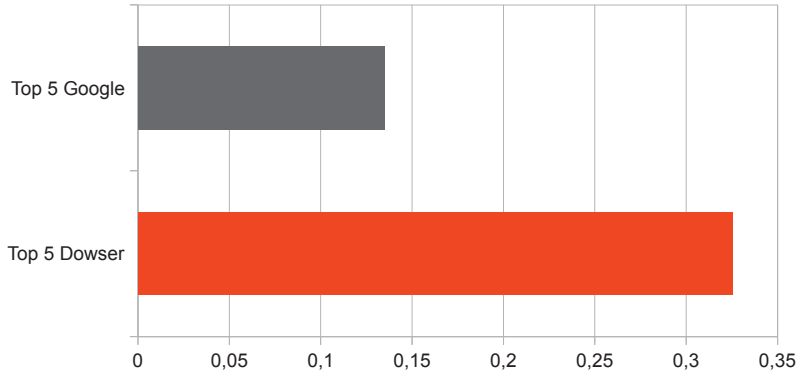


FIGURE 7.12 – Score moyen des pages des top 5 de Google et DOWSER

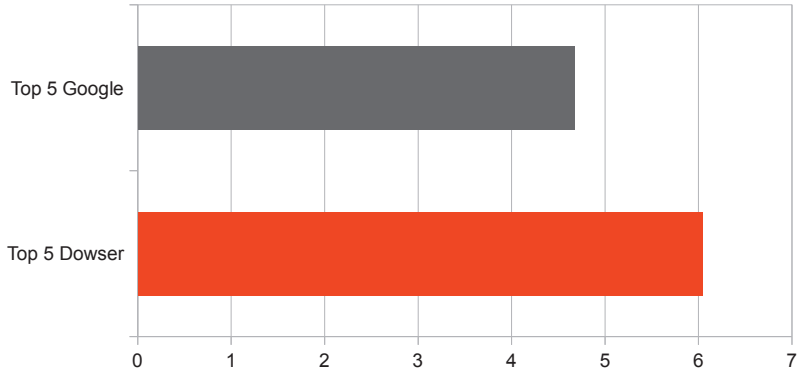


FIGURE 7.13 – Note moyenne donnée par l'utilisateur aux pages des top 5 de Google et DOWSER

7.1.4 Synthèse sur l'évaluation de notre approche

L'approche DOWSER a été globalement évaluée dans cette expérimentation. Les résultats obtenus montrent que le besoin de l'utilisateur est partiellement couvert lors de la construction du profil opérationnel. L'utilisation de mots-clés et de concepts permet de couvrir ce besoin au niveau thématique mais aussi au niveau terminologique. Le manque de couverture de ce besoin a ouvert la voix à l'utilisation du retour de pertinence dans DOWSER, présenté section 5.3 page 81, et évalué dans la section suivante. Cependant, le profil opérationnel permet, en l'état, de guider la collecte dans le processus d'exploration

du Web et de fournir à l'utilisateur de nouvelles sources d'intérêt. Les résultats obtenus permettent de mettre en exergue la capacité de DOWSER à découvrir des sources d'intérêt. L'expérimentation présentée dans cette section ne vise pas à comparer la pertinence et l'efficacité de notre système avec ceux des moteurs de recherche comme Google. Par contre, cette expérimentation valide notre approche et propose une alternative aux outils de DI et RI classique pour la découverte de sources d'intérêt.

7.2 Calibrage expérimental des paramètres

La seconde expérimentation réalisée a pour but d'améliorer la représentation du besoin de notre profil opérationnel en calibrant expérimentalement la taille des vecteurs dans le profil, le poids donné aux concepts et aux mots-clés dans la mesure de similarité adaptative et l'importance accordée au retour de pertinence dans le système DOWSER. Différentes valeurs sont testées pour chacun de ces paramètres afin d'identifier celles qui impactent favorablement la capacité du système à détecter des pages d'intérêt.

7.2.1 Présentation de l'étude

Dans cette expérimentation, on vise à améliorer la représentation du besoin de l'utilisateur, ainsi que le score de pertinence qui en découle, en faisant évoluer notre profil opérationnel. Pour ce faire, le système DOWSER est testé sur un corpus de documents. Une vérité terrain, sous forme de couples questions/réponses où les réponses sont des documents du corpus, est utilisée. Le calibrage expérimental a pour objectif de faire varier les paramètres du système et d'évaluer leur impacte sur la capacité de DOWSER à trouver les documents pertinents pour chaque question. Au travers de cette étude, le but est de répondre aux questions suivantes :

Question 1 - Quelle taille de vecteur d'instances de concept permet d'impacter favorablement le score de similarité en retournant un plus grand nombre de sources d'intérêt ?

Question 2 - Quelle taille de vecteur de mots-clés permet d'impacter favorablement le score de similarité en retournant un plus grand nombre de sources d'intérêt ?

Question 3 - Quel ratio, assigné à la mesure terminologique par rapport à la mesure thématique dans la mesure de similarité adaptative, permet de trouver un plus grand nombre de sources d'intérêt ?

Question 4 - Quel poids, donné au retour de pertinence, permet d'affiner la représentation du besoin ?

Question 5 - Le score de similarité global permet-il de trouver un plus grand nombre de pages d'intérêt que le score de similarité thématique seul et que le score de similarité terminologique seul ?

La dernière question ne correspond pas a un calibrage de paramètre mais plutôt à une validation de notre approche concernant notre mesure de similarité adaptative. En effet, contrairement à l'expérimentation précédente, la mesure adaptative est ici utilisée et son efficacité doit être éprouvée. Cette expérimentation se déroule sans l'aide d'utilisateur, seulement au travers d'un corpus de documents associé à un ensemble de questions/réponses comme le décrit le protocole expérimental.

7.2.2 Protocole expérimental

Le protocole décrit en détail le contexte de l'expérimentation : le corpus mis en jeu et le déroulement étape par étape de l'expérimentation. Les résultats obtenus sont analysés dans la section suivante afin de répondre aux questions posées ci-dessus.

a. Corpus et thématique de recherche

Le corpus utilisé est le *FIRE Collection*[5]. Il offre les avantages suivants :
- c'est un corpus reconnu et utilisé par une partie de la communauté de RI [Paik et al., 2011, Mittal et al., 2010, Bhaskar et al., 2010],
- le contenu des documents est assimilable à celui de pages Web, que ce soit au niveau de la taille, des sujets variés traités et des langues.
- le corpus est gratuit et récupérable facilement via Internet

Pour cette expérimentation, nous utilisons une sous partie du corpus de 2010, dans laquelle sont gardés uniquement les documents en anglais. En effet, pour rappel, l'outil d'extraction de concepts DBPedia utilisé est configuré pour traiter des textes en anglais. Au final, le corpus de notre expérimentation se compose de 15000 documents. Un ensemble de couple questions/réponses est également fourni avec le corpus. Les réponses aux questions correspondent aux noms des documents répondant à la question. Nous avons filtré les questions dont le nombre de réponses dans notre corpus est inférieur à 20. En effet, notre expérimentation nécessite un nombre suffisant de réponses afin que l'impact de nos tests paramétriques soient représentatifs. Ainsi, nous avons utilisés 20 questions dont le nombre de réponses varient entre 20 et 130 documents.

Cette expérimentation s'apparente d'avantage à une tâche de RI que de DI dans la mesure où le calibrage expérimental n'utilise pas le processus de découverte de sources de DOWSER. Ceci permet d'utiliser un corpus fermé contrairement à l'expérimentation précédente qui s'est effectuée sur le Web.

b. Présentation des fonctionnalités utilisée

Comme lors de la première expérimentation, une version modifiée de l'approche DOWSER a été utilisée afin de répondre aux différentes questions posées.

Ce module de DOWSER ne contient pas la tâche d'extension de la zone de collecte, ni la tâche d'exploration ciblée, puisque l'expérimentation s'effectue sur un corpus déjà constitué. L'interface homme/machine est également retirée puisque l'expérimentation se déroule sans utilisateur. Cependant, l'activité de utilisateur est simulée dans ce module afin de remplacer ce dernier dans la construction du profil opérationnel et dans le jugement des pages que fournit DOWSER par rapport à la vérité terrain du corpus.

La mesure de similarité adaptative, combinant l'utilisation de mots-clés et de concepts, est appliquée sur les documents du corpus afin d'évaluer leur pertinence. Cependant, la mesure de similarité terminologique du besoin, et celle thématique, seront toutes deux également utilisées comme dans l'expérimentation précédente. Ainsi, les résultats obtenus avec notre mesure adaptative pourront être comparés à ceux obtenus avec ces deux autres mesures.

5. http://www.isical.ac.in/~fire/

Enfin, le retour de pertinence est présent dans le sous-système de cette expérimentation. L'algorithme utilisé active et désactive sa prise en compte afin de mesurer son impact. Le déroulement de l'expérimentation est décrit dans la section suivante.

c. Déroulement de l'expérimentation

L'expérimentation comprend plusieurs étapes qui sont répétées avec des paramètres différents afin de mesurer leur impact sur le système. Cette partie décrit le déroulement global de l'expérimentation ainsi que les valeurs testées pour chacun des paramètres.

Pré-traitement Afin de simuler la tâche de collecte, le corpus est traité avant l'expérimentation et les documents qu'il contient sont analysés. Chaque document est indexé avec les mots-clés et les concepts extraits de son contenu comme s'il s'agissait d'une page Web collectée par la tâche d'exploration. Ces documents sont indexés sans score de pertinence puisque le profil opérationnel n'est modélisé que dans l'étape suivante.

Construction du profil opérationnel La première étape consiste à créer, pour une question donnée, le profil opérationnel correspondant. Pour ce faire, notre module récupère du corpus 5 pages tirées aléatoirement parmi les pages répondant à la question. Ces pages sont considérées par le système comme les sources d'intérêt du profil opérationnel. Leur contenu est donc exploité afin de modéliser le besoin informationnel au sein d'un profil opérationnel (voir section 5.2.2, page 75). Ce besoin informationnel est une représentation de la question par notre système. Cette construction est illustrée par l'étape 1 du schéma 7.14, page 136.

Classement par pertinence des documents du corpus Le module DOWSER conçu dans cette expérimentation calcule la pertinence de chaque document du corpus. Pour ce faire, la mesure de similarité adaptative est utilisée entre les documents et le profil opérationnel (voir étape 2 du schéma 7.14, page 136). Chaque document est annoté avec son score de similarité adaptatif mais aussi avec son score de similarité thématique et terminologique.

Évaluation du top 10 Comme les moteurs de recherche les plus populaires, le top 10 des documents est retourné par le système. Il s'agit des 10 documents les plus pertinents, au regard de leur score de pertinence adaptatif. Ils sont retirés du corpus et sont jugés automatiquement par notre module. La vérité terrain permet de mettre en exergue les documents retournés qui répondent réellement à la question donnée (vrais positifs) et ceux qui n'y répondent pas (faux positifs). Le schéma 7.14, page 136, illustre cette troisième étape.

Boucle et décision d'arrêt Lorsque le retour de pertinence est utilisé, le jugement opéré par notre module est exploité pour construire les retours positifs et négatifs et ainsi impacter le profil opérationnel (voir section 5.3, page 81). Un nouveau profil est alors construit et l'expérimentation se poursuit en répétant la seconde étape. Si le retour de pertinence n'est pas utilisé, l'expérimentation répète directement la seconde étape avec

le profil opérationnel courant. L'expérimentation boucle ainsi pendant 10 itérations correspondant donc aux 100 premiers résultats traités. L'objectif de l'expérimentation est de répondre aux questions posées page 133 dès les premières itérations, c'est pourquoi le module n'exploite que les 100 premiers résultats.

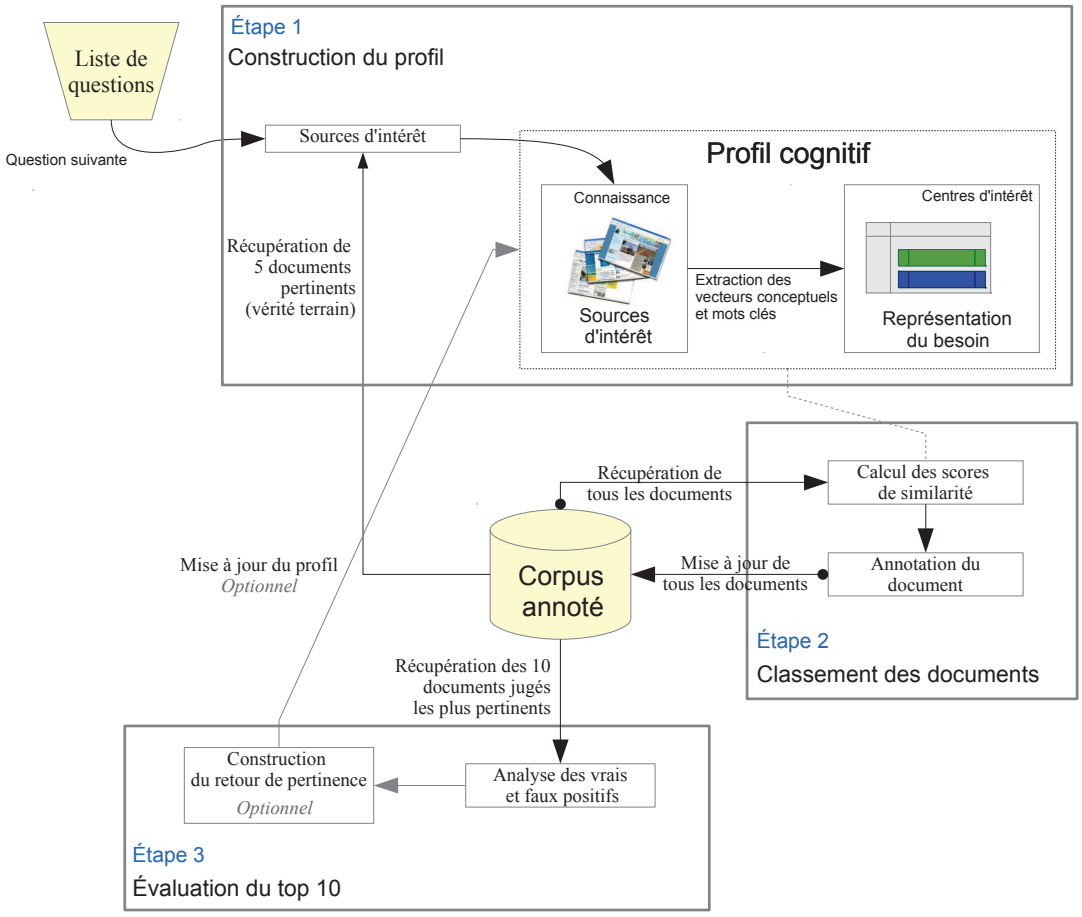


FIGURE 7.14 – Déroulement étape par étape de l'expérimentation

7.2.3 Méthode de calibrage

Afin de calibrer expérimentalement les différents paramètres du système DOWSER, les questions utilisées et le nombre d'itérations sont les mêmes pour chaque test durant toute l'expérimentation. Comme il nous est impossible de tester une infinité de valeurs pour chacun de ces paramètres, le calibrage se fait par paliers progressifs en fonction du paramètre testé. Par exemple, si la valeur d'un paramètre peut osciller entre 0 et 1, l'expérimentation se fera avec des valeurs allant de 0.0 à 1.0 avec un palier de 0.1 entre chaque valeur. Pour chaque test, afin d'éviter tout biais lié à la modélisation du besoin, l'expérimentation est lancée une seconde fois mais la construction du profil se fait avec un ensemble de documents pertinents différent.

Pour identifier la valeur optimale de chacun des paramètres testés, le rappel est calculé à chaque itération et pour chaque test. La courbe représentative de son évolution permet d'évaluer l'impact de la valeur du paramètre testé. En effet, le rappel correspond au nombre

de documents pertinents retrouvés au regard du nombre de documents pertinents dans le corpus. Cette mesure est donc importante puisqu'elle représente la capacité du système à constituer un corpus de documents pertinents sur un sujet d'intérêt donné. La mesure de la précision est également calculée. Elle permet de connaître le nombre de documents pertinents retrouvés rapporté au nombre total de documents proposés par le système. Or, notre système retourne 100 documents alors que certaines questions ne contiennent que 20 bonnes réponses sur les 15000 documents du corpus. Ainsi, on s'attend à voir le score de la mesure de précision diminuer à cause du faible nombre de documents pertinents. Cependant, cette mesure permet tout de même de constater une amélioration entre deux différents paramètres au travers de la F-Mesure : cette dernière fournit des indications sur la performance globale du système en prenant en compte à la fois le rappel et la précision au sein d'une seule mesure :

$$F - Mesure = \frac{2 * Précision * Rappel}{Précision + Rappel}$$

La mesure du rappel est donc utilisée pour évaluer la capacité de notre système à découvrir l'ensemble des sources pertinentes. Sa capacité à identifier les sources pertinentes est déterminée par la mesure de précision. Enfin, la F-Mesure nous renseigne sur la performance globale de notre système.

7.2.4 Résultats et discussions

Cette section présente les résultats obtenus en faisant varier les valeurs des différents paramètres du système DOWSER durant l'expérimentation présentée ci-dessus. Les paramètres testés sont la taille des vecteurs du profil (voir section 5.2.2, page 75), la valeur de δ dans la mesure de similarité adaptative (voir section 6.3.2, page 106), et la valeur de α durant le processus de retour de pertinence (voir section 5.3, page 81).

a. Impact de la taille des vecteurs

Lors de la première expérimentation, la taille des vecteurs de concepts et de mots-clés constituant le profil opérationnel a été fixée à 20. Cependant, que ce soit pour les concepts ou pour les mots-clés, un nombre trop faible de termes peut limiter la représentation du besoin. À l'inverse, un nombre trop important peut apporter du bruit, fausser la compréhension du besoin mais également rallonger les temps de traitement. De plus, l'approche conceptuelle par rapport à l'approche mots-clés est bien différente. Leur mesure de similarité sont différentes et, comme montré lors de la première expérimentation, l'un couvre la terminologie du besoin alors que l'autre couvre la thématique. Ainsi, la taille optimale du vecteur d'instances de concept peut être différente de la taille optimale du vecteur de mots-clés.

Comme évoqué précédemment, il est impossible de tester une infinité de valeurs. Ainsi, un premier test a été réalisé en faisant varier conjointement la taille des deux vecteurs. Un second test a été mené pour affiner les résultats en ne faisant varier que la taille d'un des deux vecteurs.

Le graphique 7.15, page 139, illustre le rappel au fur et à mesure des itérations selon leurs différentes valeurs. Par exemple, la légende C5-K5 signifie que la taille du vecteur de concepts est de 5 (C5) et celle du vecteur de mots-clés est 5 également (K5). On constate

qu'à partir de 40 concepts et 40 mots-clés (C40-K40), le rappel n'évolue presque plus et stagne à 63% de couverture. Ce constat de stagnation est également visible sur la figure illustrant la précision et la F-Mesure (voir figure 7.17, page 139, et figure 7.16, page 139). On en déduit que la combinaison C40-K40 est le meilleur compromis : surcharger le profil avec plus de 40 termes par vecteur n'apporte qu'un faible gain et ajoute des coûts de traitement supplémentaires.

Parmi les valeurs testées, la combinaison C40-K40 est optimale lorsque la même taille est utilisée pour nos deux vecteurs. À partir de ce premier constat, nous avons fait varier la taille d'un des deux vecteurs et gardé l'autre fixe afin d'affiner nos résultats. Dans un premier temps, la taille du vecteur de concepts a été modifiée sans toucher aux 40 mots-clés, puis inversement en faisant varier uniquement le vecteur de mots-clés. Les figures 7.18, page 140, et 7.19, page 140, illustrent les résultats obtenus sur la F-Mesure.

À la vue des résultats, les courbes montrent qu'une modélisation du profil à 5 concepts permet de légèrement améliorer la couverture thématique du besoin alors qu'un seul est insuffisant et que 10 concepts ou plus produisent un moins bon résultat en raison d'un apport de bruit. Cette légère amélioration se constate également sur la courbe du rappel : celui-ci est amélioré de 3% avec la combinaison C5-K40 par rapport à la combinaison C40-K40 (voir figure 7.20, page 140). Concernant les mots-clés, les courbes montrent qu'avec un nombre de termes égal à 60, le besoin est défini un peu plus précisément. Le rappel passe de 61% à 63% avec la combinaison C40-K60 (voir figure 7.21, page 141). Les courbes de précision suivent la tendance arborée par les courbes du rappel, confortant ainsi nos résultats (voir figure B.1 en annexe page 193, et figure B.2 en annexe 194).

En fusionnant ces deux résultats au travers de la combinaison C5-K60, la courbe de la F-Mesure, illustrée en figure 7.22, page 141, montre un gain de 4% par rapport à notre combinaison C40-K40. L'amélioration obtenue est constatée dans les courbes de rappel et de précision comme l'illustre respectivement les figures B.3 en annexe page 194, et B.4 en annexe page 195. Le score de rappel à 65% au bout des 10 itérations est le meilleur de nos scores obtenus jusqu'alors, toutes combinaisons confondues.

b. Évaluation de la mesure de similarité adaptative

L'optimisation de la couverture du besoin passe d'abord par la définition du profil opérationnel. En définissant une taille de vecteur optimale, nous avons amélioré la modélisation du besoin dans notre profil opérationnel. L'amélioration de la couverture du besoin passe, dans un second temps, par l'optimisation de la mesure de similarité utilisée. Pour rappel, notre mesure de similarité adaptative combine le score de similarité thématique (concepts) et le score de similarité terminologique (mots-clés). Elle est définie comme suit :

$$Sim(P, D) = \delta * Sim_C(\vec{P}_C, \vec{D}_C) + (1 - \delta) * Sim_K(\vec{P}_K, \vec{D}_K)$$

Où $\text{Sim}_C(\vec{P}_C,\vec{D}_C)$ et $\text{Sim}_K(\vec{P}_K,\vec{D}_K)$ correspondent respectivement aux mesures de similarité thématique et terminologique entre le document D et le profil opérationnel P (voir définition 14, page 108).

Dans les figures 7.20, page 140, et 7.21, page 141, on constate que la variation de la taille du vecteur de concepts impact moins le rappel que la variation de la taille du

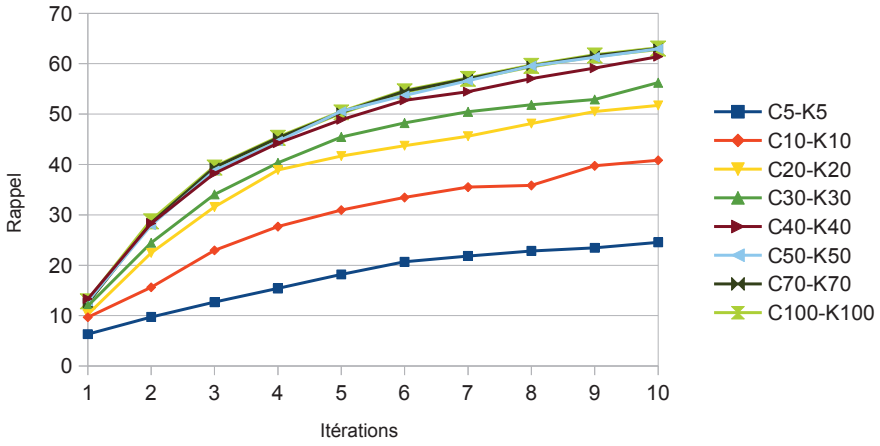


FIGURE 7.15 – Évolution du rappel en fonction de la taille des vecteurs

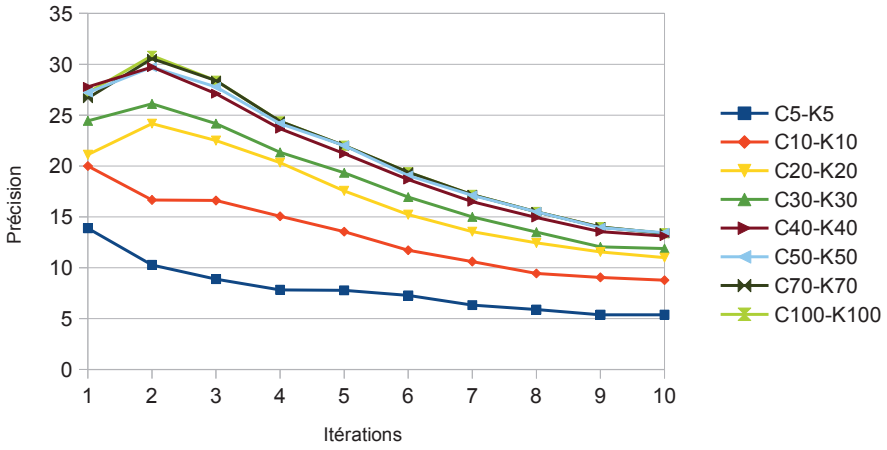


FIGURE 7.16 – Évolution de la précision en fonction de la taille des vecteurs

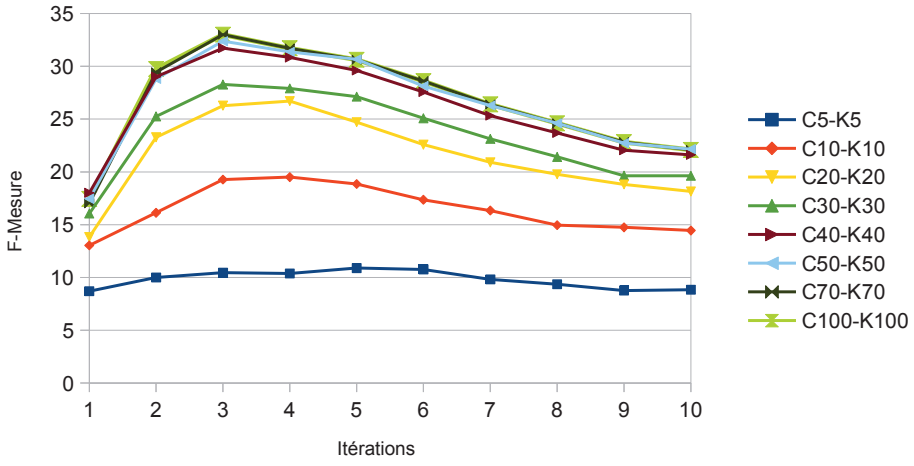


FIGURE 7.17 – Évolution de la F-Mesure en fonction de la taille des vecteurs

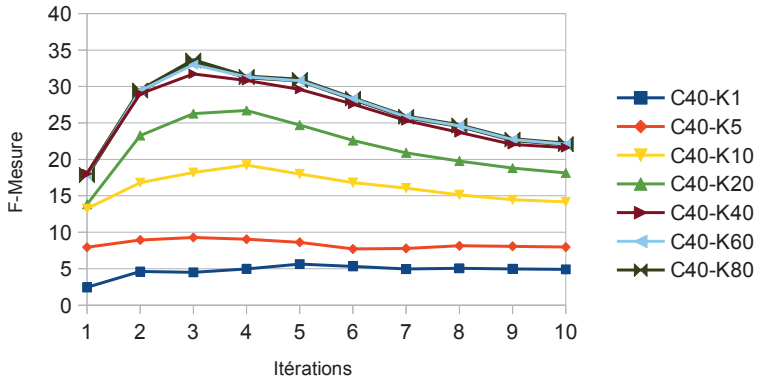


FIGURE 7.18 – Évolution de la F-Mesure en fonction de la taille du vecteur de mots-clés

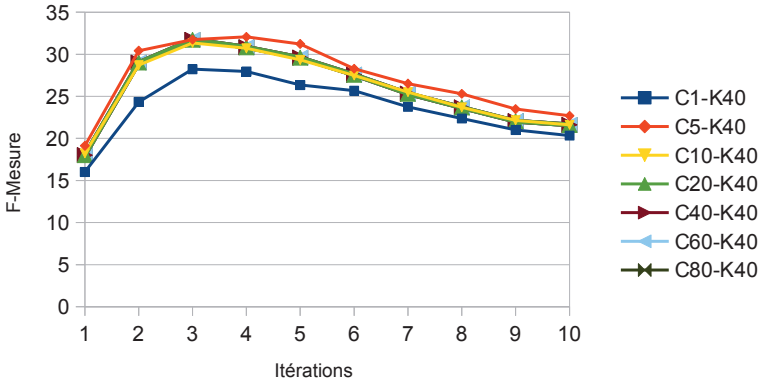


FIGURE 7.19 – Évolution de la F-Mesure en fonction de la taille du vecteur de concepts

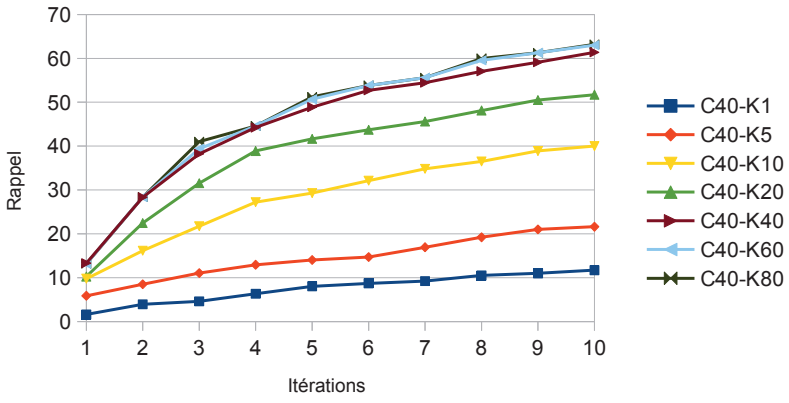


FIGURE 7.20 – Évolution du rappel en fonction de la taille du vecteur de mots clés

vecteur de mots-clés. Cela est notamment dû à la mesure de similarité mots-clés qui se veut plus précise que la mesure concepts qui est plus permissive (voir section 6.3.2, page 106) ; la variation de ce vecteur mots-clés se ressent donc plus fortement sur notre mesure

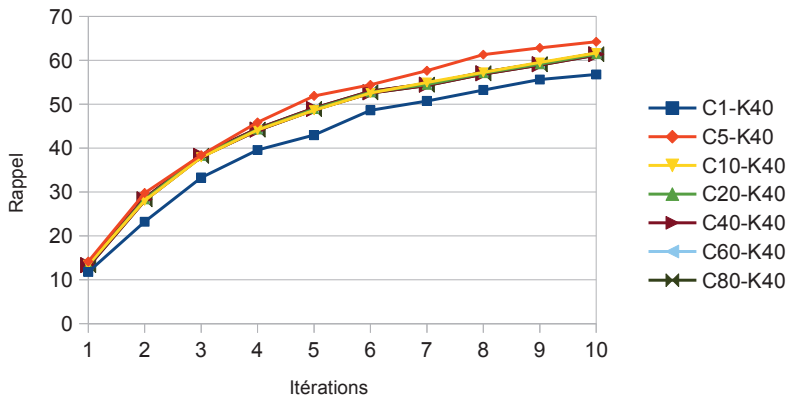


FIGURE 7.21 – Évolution du rappel en fonction de la taille du vecteur de concepts

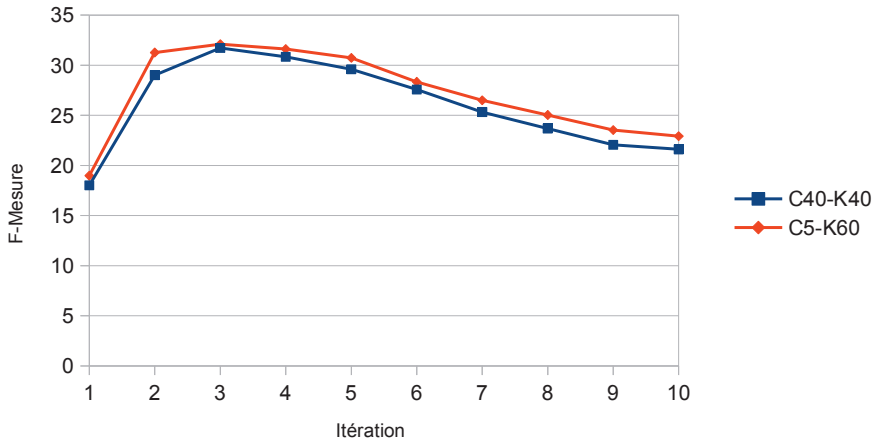


FIGURE 7.22 – Amélioration de la F-Mesure avec la taille du profil optimisée

de similarité adaptative. Or cette dernière, selon la valeur de δ, donne plus d'importance à la couverture thématique ou à la couverture terminologique du besoin (voir c. page 108). Pour optimiser la modélisation du profil et trouver les tailles optimales de vecteurs, la valeur de δ a été fixée afin de garder une proportion équivalente entre les deux scores, tel que :

$$\delta = 0.5$$

et donc :

$$Sim(P, D) = 0.5 * Sim_C(\vec{P_C}, \vec{D_C}) + 0.5 * Sim_K(\vec{P_K}, \vec{D_K})$$

La valeur δ, comprise entre 0 et 1, n'est sans doute pas optimale à 0.5. Lorsque δ est égal à 1, la mesure adaptative utilise uniquement la mesure de similarité terminologique. Avec une valeur de δ à 0, c'est uniquement la mesure de similarité thématique qui est utilisée. Tester δ à ses bornes permet d'évaluer notre mesure de similarité adaptative par rapport à ces deux mesures spécifiques. Elle a donc été évaluée durant notre expérimentation avec

notre profil de 5 concepts et 60 mots-clés. Les résultats obtenus sur la F-Mesure sont illustrés dans la figure 7.23, page 142. On constate tout d'abord que l'utilisation de notre

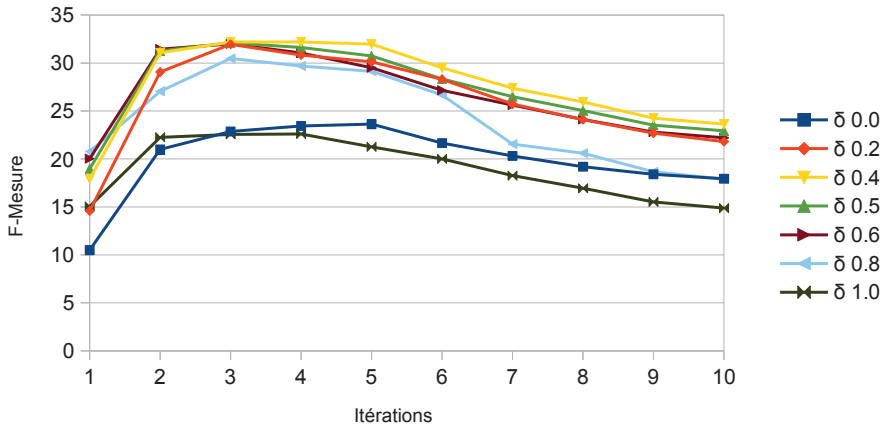


FIGURE 7.23 – Évolution de la F-Mesure en fonction du δ

mesure adaptative améliore la couverture du besoin par rapport à l'utilisation de la mesure mots-clés ou de la mesure conceptuelle (δ respectivement à 0 et 1). Cela valide donc notre approche de mesure combinatoire basée sur ces deux mesures. Il reste à déterminer la valeur optimale de δ dans cette mesure. Cette valeur influence relativement faiblement les résultats obtenus. Le δ à 0.4 apporte cependant les meilleurs résultats à la vue des courbes de la F-Mesure. Ce résultat est constatable également dans les courbes de précision (voir figure B.6 en annexe page 196) et de rappel (voir figure B.5 en annexe page 195) où le δ à 0.4 l'emporte également sur les autres valeurs testées. De plus, la valeur optimale de δ est la même avec d'autres configurations de taille de vecteurs, comme le montre les courbes des figures B.7 et B.8 en annexe page 196. Ce résultat confirme la logique de vouloir donner plus de poids à la mesure terminologique qu'à la mesure thématique qui couvre moins précisément le besoin.

Retour de pertinence Les résultats obtenus nous ont permis de définir la taille des vecteurs de notre profil opérationnel et du poids de chacun de ses vecteurs dans la mesure de similarité adaptative. La modélisation du profil et son exploitation étant optimisées, la dernière étape consiste à exploiter les résultats obtenus à chaque itération pour affiner la représentation du besoin, c'est-à-dire le contenu des vecteurs. Pour rappel, notre formule pour la prise en compte du retour de pertinence est la suivante :

$$\vec{P}^{t+1} = \alpha\vec{P}^{t} + \beta\vec{V}^{P,t} - \gamma\vec{V}^{NP,t}$$

où V^P un vecteur de termes (mots-clés ou concepts) issus des documents pertinents parmi les 10 retournés à l'itération t, et V^{NP} un vecteur de termes (mots-clés ou concepts) issus des documents non pertinents parmi les 10 retournés à l'itération t (voir la définition 9 page 83, et la définition 8 page 94). Il nous faut déterminer l'importance donnée au retour de pertinence afin d'optimiser son impact sur le profil opérationnel. Zhai [Zhai & Lafferty, 2001] montre que l'importance accordée au retour de pertinence ne doit pas dépasser celle

de la représentation d'origine du besoin mais s'en rapprocher. Le poids des retours négatifs doit donc être inférieur ou égal à celui des retours positifs. On a alors :

$$\begin{cases} \alpha > \beta + \gamma \\ 0.5 <= \alpha < 1 \\ \beta >= \gamma \end{cases} \tag{7.1}$$

Nous avons donc testé les valeurs de α comprises entre 0.5 et 1. Les résultats sont illustrés dans les figures 7.27, page 145, et 7.28, page 145. L'application de l'approche Rocchio

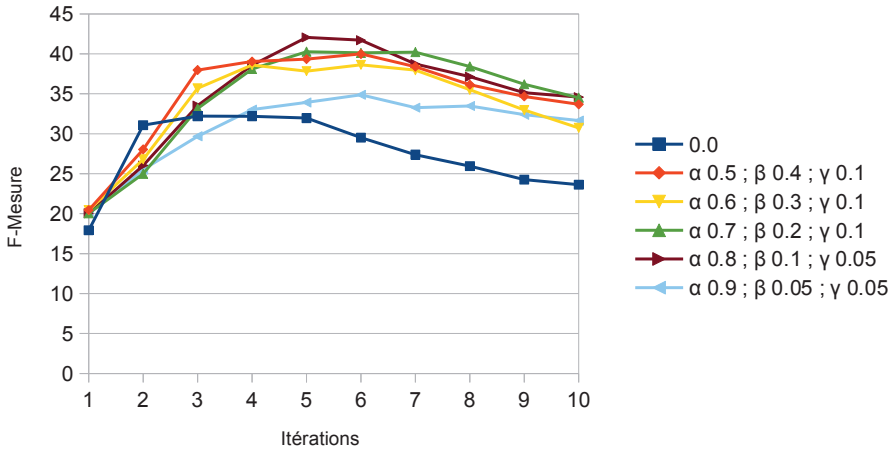


FIGURE 7.24 – Évolution de la F-Mesure en fonction de α, β et γ

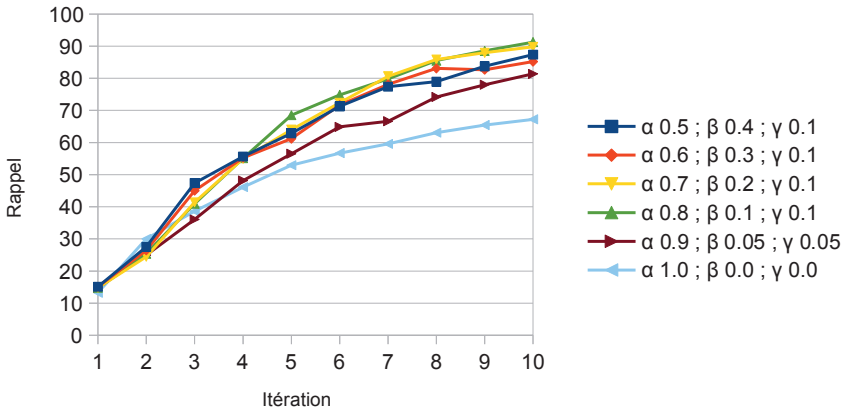


FIGURE 7.25 – Évolution du rappel en fonction de α, β et γ

sur notre profil opérationnel améliore fortement la couverture du besoin, et donc la précision et la F-Mesure. Pour une valeur de α à 0.7 ou 0.8, le rappel passe de 67% sans l'exploitation du retour de pertinence à 91%. L'utilisation du retour de pertinence fait également gagner 10% de précision comme l'illustre la figure 7.26, page 144 : la précision

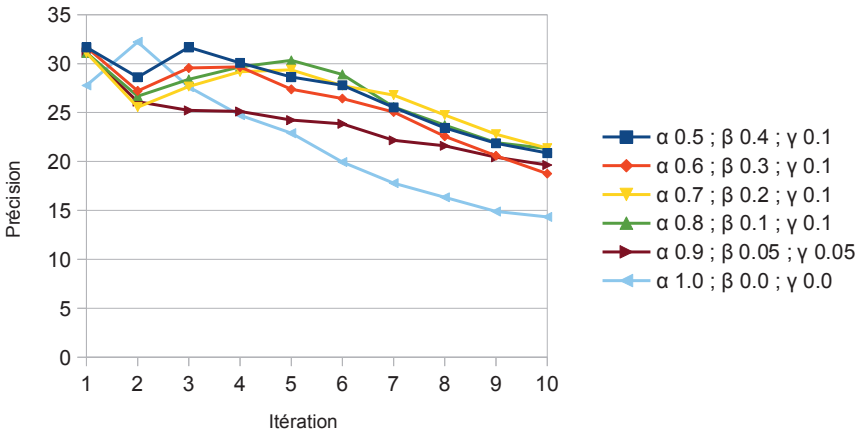


FIGURE 7.26 – Évolution de la précision en fonction de α, β et γ

atteint les 23% avec α à 0.7 ou 0.8 contre 14% sans retour de pertinence. Cette nette progression se justifie au regard de l'écart-type. Les courbes présentées sont le résultat d'une moyenne appliquée sur les 20 questions au fur et à mesure des 10 itérations. Les figures 7.27, page 145, et 7.28, page 145, contiennent respectivement les courbes de rappel sans l'utilisation du retour de pertinence et avec, en affichant l'écart-type. On constate que ce dernier est important lorsque le retour de pertinence n'est pas utilisé. Ce qui signifie que la construction de notre profil opérationnel n'est pas efficace pour toutes les questions. Les documents pertinents pour une question donnée, utilisés pour construire le profil, peuvent ne pas être pas assez représentatifs. Certaines questions peuvent aussi être plus complexes que d'autres et nécessiter un plus grand nombre de documents d'intérêt avant que le système ne capture le besoin associé. L'utilisation du retour de pertinence permet à la fois de pallier ces problèmes mais aussi d'améliorer la représentation du besoin de l'ensemble des questions. Ce résultat se constate sur la figure 7.28, page 145, où l'écart-type est fortement réduit. En plus d'être moins important, l'écart-type diminue au fil des itérations, montrant une convergence de la couverture du besoin sur toutes les questions.

Un second paramètre important du retour de pertinence est la fréquence de prise en compte des jugements pour constituer les retours positifs et négatifs. Dans le premier test du retour de pertinence, le système prenait en compte le top 10 des documents les plus pertinents pour construire les vecteurs de termes positifs et négatifs en fonction de la vérité terrain. Ce top 10 fait référence aux 10 résultats par page fournis par les moteurs de recherche. Or, d'un point de vue opérationnel, il est difficile de demander à l'utilisateur de juger des ensembles de 10 pages. De plus, les analyses oculométriques (*eyetracking* montrent que l'utilisateur de moteurs de recherche ne consulte généralement que la première page de résultats et s'intéresse principalement qu'aux 5 premiers liens proposés [Granka et al., 2004].

L'évaluation de l'impact du retour de pertinence sur DOWSER a été renouvelée en prenant en compte le top 5 au lieu du top 10. Afin de pouvoir comparer les deux approches, 20 itérations ont été faites au lieu de 10 afin que le système évalue le même nombre de documents (100 documents). Le test a également été effectué avec le top 2 et 50 itérations.

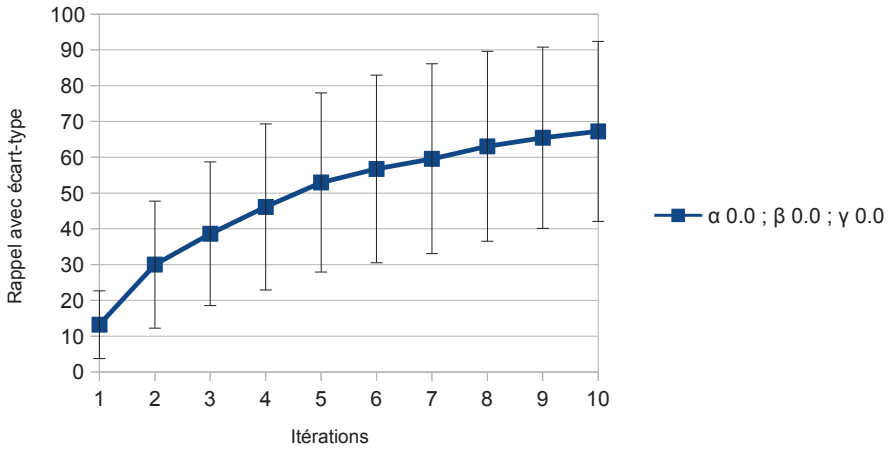


FIGURE 7.27 – Évolution de l'écart-type du rappel sans le retour de pertinence

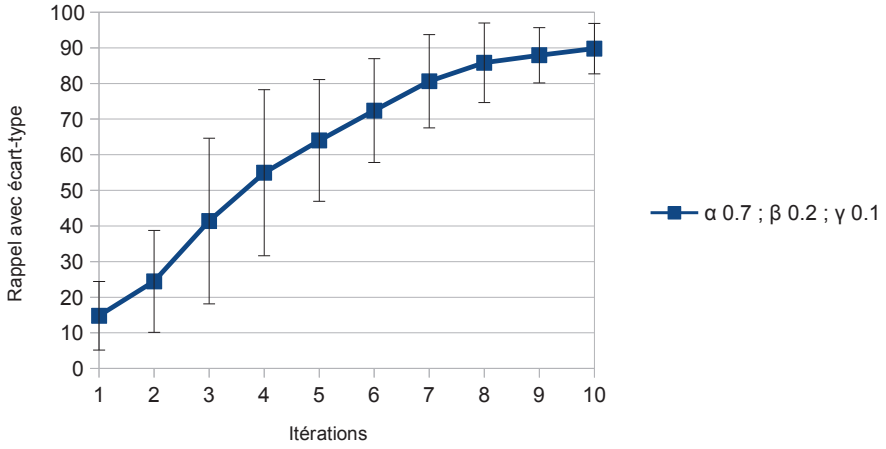


FIGURE 7.28 – Évolution de l'écart-type du rappel avec le retour de pertinence

Les résultats obtenus sur la F-Mesure, pour ces deux tests, sont respectivement illustrés dans la figure 7.29, page 146 et dans la figure 7.30, page 146.

Comme avec l'expérimentation utilisant le top 10, les courbes montrent que la valeur du α à 0.7 ou 0.8 est optimale pour l'exploitation du retour de pertinence. La figure 7.31, page 147, compare l'évolution de la F-Mesure entre l'exploitation du top 2, du top 5 et du top 10, avec un α à 0.8. Les courbes comparatives de rappel et de précision sont consultables en annexe via les figures B.9 et B.10, page 197. On constate que le besoin est globalement mieux couvert en utilisant le top 5 pour le retour de pertinence. Durant les toutes premières itérations, le profil se précise plus lentement, donnant l'avantage à l'approche top 10 en début d'expérimentation. Cependant, l'approche top 2 et top 5 fournissent rapidement de meilleurs résultats au fil des itérations suivantes. L'approche top 2 ne permet cependant pas d'affiner le profil de manière optimale ; le score de la F-Mesure est globalement meilleur avec l'approche top 5 durant l'expérimentation. Les courbes de rappel et de précision suivent cette tendance. Le nombre de documents pertinents dans le corpus étant faible,

les courbes chutent en fin d'expérimentation. Les résultats sont cependant satisfaisants et devraient améliorer les résultats de collecte sur le Web de notre système. Exploiter le top 5 pour le retour de pertinence permet à la fois de réduire la tâche de jugement de l'expert tout en améliorant la capacité du système DOWSER à couvrir le besoin informationnel.

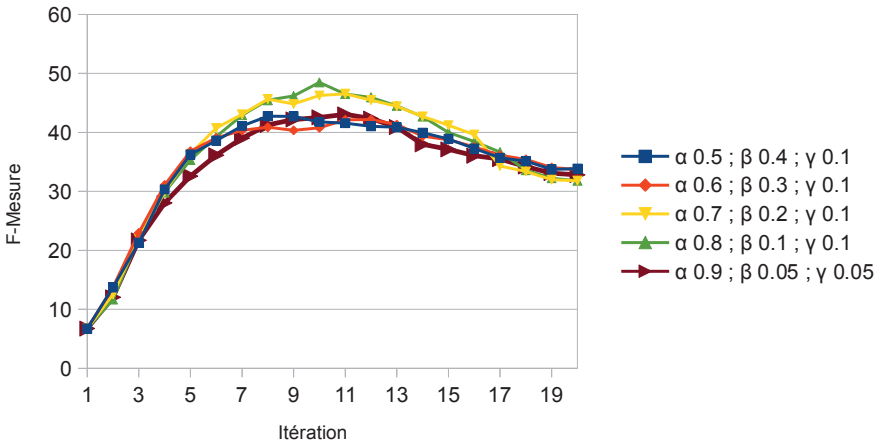


FIGURE 7.29 – Évolution de la F-Mesure en exploitant 5 pages jugées dans le processus de retour de pertinence

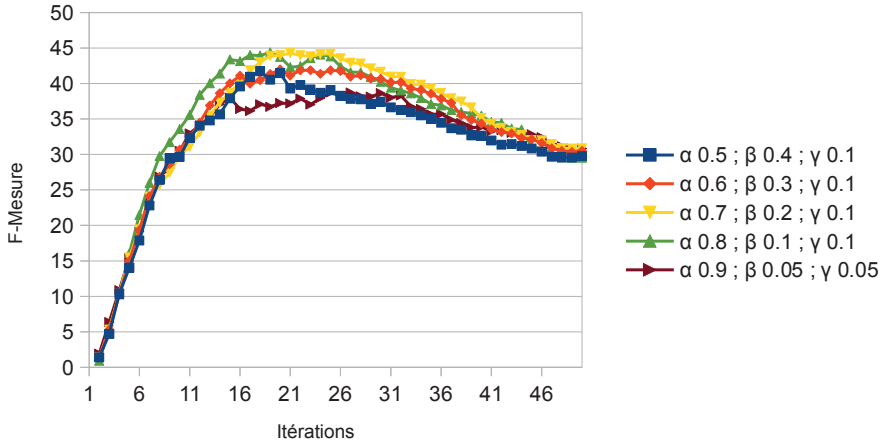


FIGURE 7.30 – Évolution de la F-Mesure en exploitant 2 pages jugées dans le processus de retour de pertinence

7.2.5 Synthèse sur le calibrage expérimental des paramètres

Cette seconde expérimentation s'est déroulée en trois étapes :

1. Détermination de la taille optimale des vecteurs du profil opérationnel,
2. Exploitation de ce profil opérationnel en fixant l'importance accordée à l'approche conceptuelle par rapport à l'approche mots-clés dans la mesure adaptative,

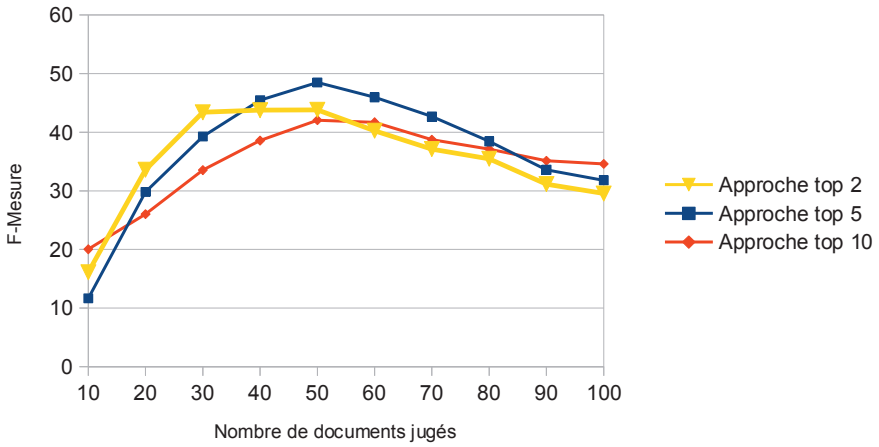


FIGURE 7.31 – Comparaison de la F-Mesure en exploitant 2, 5 et 10 pages jugées dans le processus de retour de pertinence

3. Amélioration du contenu du profil opérationnel, au travers du retour de pertinence, afin d'obtenir des vecteurs définissant au mieux le besoin informationnel et une mesure adaptative capable d'identifier un plus grand nombre de documents pertinents.

Chaque étape a permis d'améliorer la couverture du besoin et la précision du système. Les résultats montrent que 5 concepts suffisent pour couvrir thématiquement le besoin. A contrario, d'un point de vue terminologique, un plus grand nombre de mots-clés sont nécessaires pour améliorer la modélisation du besoin. Au final, 5 concepts et 60 mots-clés semble être la combinaison optimale aux vues des évaluations menées. De plus, en favorisant la mesure terminologique dans la mesure adaptative (60% contre 40% pour la mesure thématique), les résultats tendent à s'améliorer. Enfin, ces deux premières améliorations sont valorisées par le retour de pertinence qui offre une définition du besoin beaucoup plus précise, y compris pour les questions dont le profil était initialement peu représentatif du besoin. Les résultats obtenus par ces optimisations sont illustrés dans la figure 7.32, page 148, où l'on peut constater l'évolution de la F-Mesure au fur et à mesure des optimisations par rapport au profil initial utilisé dans la première expérimentation. Pour rappel, celui-ci était composé de 20 concepts et 20 mots-clés, la mesure adaptative ne donnait l'avantage à aucun des deux vecteurs et le retour de pertinence n'était pas utilisé.

7.3 Discussion

Par le biais de ces deux expérimentations, l'approche DOWSER a été évaluée et optimisée. Alors que la première expérimentation montre que le besoin de l'utilisateur est partiellement couvert lors de la construction du profil opérationnel, la seconde a permis d'améliorer cette couverture du besoin en optimisant le profil, son contenu et la mesure adaptative via un calibrage paramétrique. Ce calibrage a été réalisé en utilisant des paliers de valeurs pour tester l'impact des différents paramètres sur le système. L'utilisation de paliers est une limite à cette seconde expérimentation. Tester de nouvelles valeurs autour des valeurs optimales obtenues lors du calibrage fournirait des résultats plus précis tout en améliorant le système DOWSER.

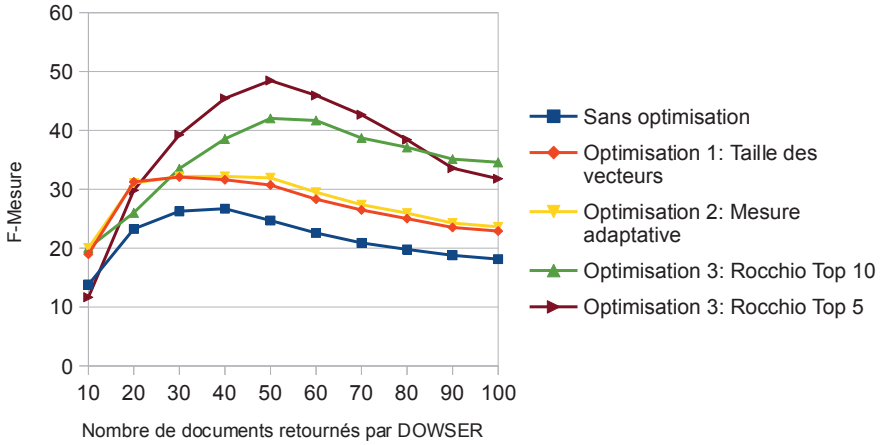


FIGURE 7.32 – Évolution de la F-Mesure au fil des optimisations

La première expérimentation a également montré que l'utilisation de mots-clés et de concepts permet de couvrir respectivement la thématique et la terminologie du besoin. La seconde expérimentation a permis d'identifier dans quelle proportion ces deux types de couvertures, fusionnées dans une seule mesure de similarité, améliorent les résultats de DOWSER. En effet, la mesure adaptative, qui combine ces deux mesures, a été validée au cours du calibrage expérimentale des paramètres. Cependant, elle n'a pas été utilisée pour guider la collecte du Web lors de nos expérimentations. Son efficacité au sein du module de découverte de sources doit encore être validée.

Enfin, la capacité de DOWSER à découvrir des sources d'intérêt a été mise en avant durant la première expérimentation. Or, cette validation expérimentale s'est faite sans le module d'extension de la zone de collecte, dont l'objectif est de fournir de nouvelles pages d'intérêt pour amorcer la collecte et de trouver plus rapidement des sources pertinentes. Cette fonctionnalité doit être testée et son efficacité évaluée.

Ainsi, une troisième expérimentation renforcerait les résultats obtenus lors de ces deux expérimentations (calibrage paramétrique plus précis). Une validation expérimentale, avec des experts du renseignement, permettrait de tester notre système dans un contexte opérationnel et fournirait des résultats complémentaires sur les fonctionnalités restant à évaluer (module d'extension de la zone de collecte et mesure adaptative pour guider la collecte).

CHAPITRE 8

DU PROTOTYPE À L'INTÉGRATION PROJET

Ce chapitre décrit l'architecture du système DOWSER et son implémentation au sein de projets de recherche. DOWSER est constitué de deux parties distinctes. La première correspond à notre robot d'exploration ciblée qui gère la collecte des pages Web. La seconde est basée sur la plate-forme WebLab et est chargée du traitement des pages Web collectées. La figure 8.1 illustre le fonctionnement global de notre système. Pour rappel, le profil opérationnel est stocké dans une base de connaissances exploitée par le robot d'exploration pour guider son exploration et sa collecte du Web. Les pages collectées sont analysées et les instances de concept et les mots-clés qu'elles contiennent sont stockés dans la base de connaissances, ainsi que leur score de similarité. Celui-ci est exploité pour guider la collecte et pour présenter à l'utilisateur des documents d'intérêt.

8.1 Le robot d'exploration

Un robot d'exploration, appelé aussi *crawler* ou *Web spider*, est un logiciel prenant en entrée un ensemble d'URLs (*Uniform Resource Locator*). Une URL permet d'identifier une ressource sur un réseau. Le robot d'exploration collecte les documents (page HTML, image, fichier son ou video, etc.) qui sont accessibles via les URLs fournies en entrée [Pinkerton, 1994, Yuwono et al., 1995]. Le système extrait de ces documents les liens pointant vers d'autres ressources. Cela permet ainsi de collecter de nouveaux documents et d'explorer le contenu du Web.

Le choix du robot d'exploration à utiliser dépend généralement de la tâche de collecte à effectuer. Lorsqu'il s'agit d'une collecte simple d'une petite partie du Web, la plupart des robots d'exploration existants suffisent. On s'intéressera alors à des préférences quant à la plate-forme supportée ou au langage de programmation. Par exemple, le choix de la présence d'un index se justifie par l'utilisation d'un moteur de recherche en aval de la collecte.

Nos critères de sélection veillent à ce qu'une communauté existe, qu'elle soit assez importante et active, et qu'une version du robot stable et récente soit disponible. Deux

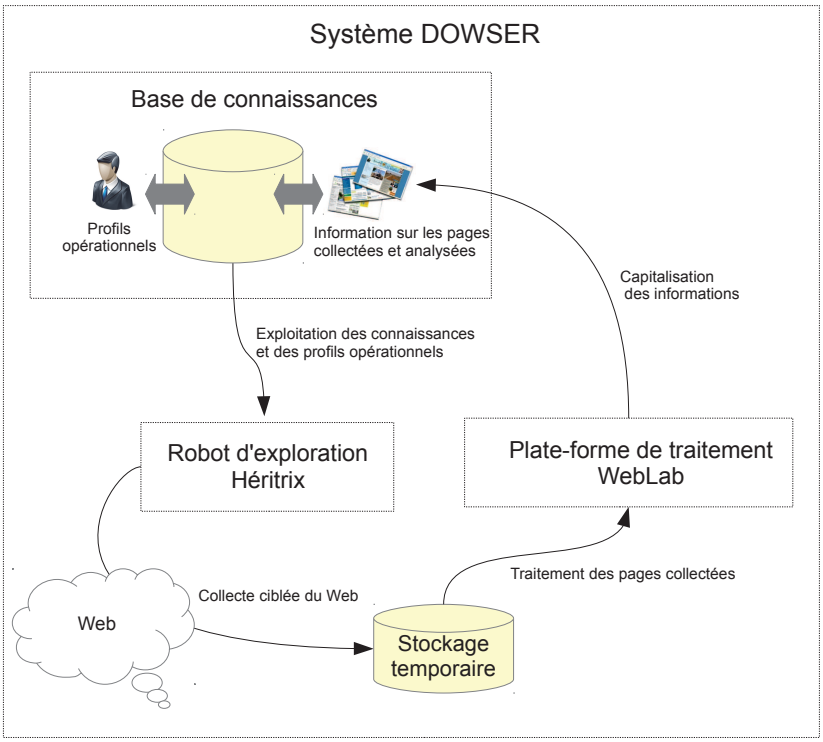


FIGURE 8.1 – Architecture générale de DOWSER

robots répondent à ces critères : Heritrix et Nutch. Notre choix s'est porté sur Heritrix car il offre une prise en main plus simple que Nutch, et son architecture modulaire permet d'étendre les fonctionnalités plus aisément. La documentation complète et l'API d'Heritrix contribuent également à le faire préférer à Nutch. Une présentation comparative de Nutch et Heritrix est incluse en annexe C, page 199.

Dans cette section, les caractéristiques d'Heritrix, son architecture, son fonctionnement et les fonctionnalités que nous avons ajoutées au robot d'exploration sont présentés.

8.1.1 Principe général

Heritrix est un robot d'exploration open source écrit en Java. Il profite d'une communauté active au travers d'une *mailing list* où les utilisateurs obtiennent des réponses rapidement. Heritrix est toujours en cours de développement, sa dernière version stable est la 3.1.0. Ce robot d'exploration est diffusé sous licence GNU Lesser General Public License (LGPL), ce qui permet un accès complet au code source.

La conception d'Heritrix est basée sur un framework générique permettant le chargement de composants. Le développeur peut facilement accéder au code de ses composants afin de les modifier ou d'en créer de nouveaux. Les composants standards d'Heritrix peuvent être remplacés par un composant personnalisé, ce qui rend ce robot très adaptable aux objectifs du développeur. L'architecture d'Heritrix est décrite par Mohr *et al.* [Mohr et al., 2004].
Comme l'illustre la figure 8.2, Heritrix est construit autour de trois composants de base :

Scope - Cette brique, correspondant aux informations relatives à la zone de couverture

(nombre de documents à collecter, etc.), permet de déterminer les URLs à collecter ou non. Cette brique peut être étendue, par exemple afin d'ajouter des vérifications quant au type des documents à collecter.

Frontier - La brique de frontière s'occupe de choisir la prochaine URL à explorer et évite la collecte d'URLs déjà visitées.

Processor Chains - La chaîne de traitement d'URL est une brique modulable qui permet d'exécuter un ensemble d'actions sur chaque URL. Elle permet notamment la collecte des URLs, l'analyse et l'extraction des liens contenus dans un document collecté.

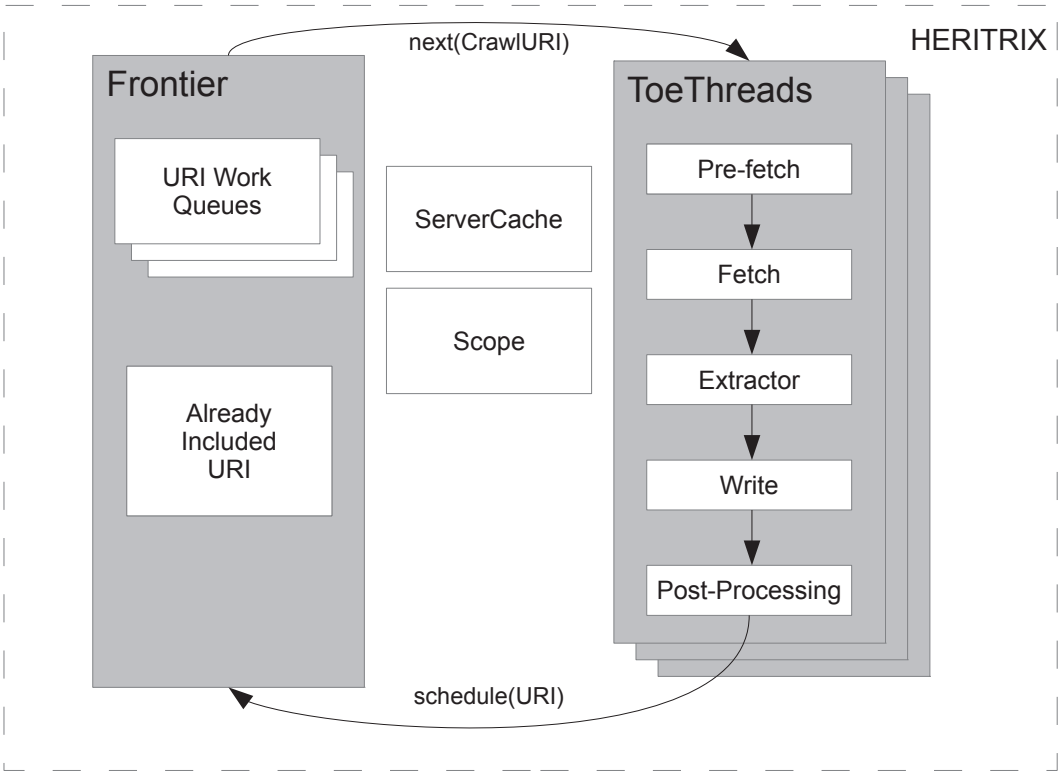


FIGURE 8.2 – Fonctionnement général d'Heritrix [1]

8.1.2 Fonctionnement

Un utilisateur peut créer un profil de collecte via l'interface Web fournie par Heritrix. Il peut configurer l'exploration en renseignant des paramètres tels que les URLs graines ou les composants à utiliser. Ces informations permettent de configurer les 3 briques *Scope*, *Frontier* et *Processor Chains*. La brique de *Frontier* fournit ensuite les premières URLs à visiter à la brique *Processor Chains*. L'exploration des URLs se fait, par défaut, en largeur.

La configuration de l'utilisateur permet notamment de définir le nombre de processus de collecte à lancer en parallèle. Ce nombre doit être assez grand pour que la tâche d'exploration soit la plus rapide possible mais pas trop élevé afin d'éviter une surcharge CPU du système. Chaque processus de collecte est appelé *ToeThread*. Ces processus fonctionnent de la manière suivante :

– Appel à la brique *Frontier* pour récupérer la prochaine URL à explorer.
– La brique *Processor Chains* est appelée pour explorer l'URL.
– Une fois l'URL explorée, un rapport est créé.

Le rapport contient un identifiant lié à l'URL appelé *CrawlURI*, ainsi que des informations relatives à l'exploration comme la durée, l'état de la collecte, le résultat.

Processus de collecte La brique *Processor Chains* contient, de base, 5 processus décrits ci-après. Comme l'illustre la figure 8.2, ces processus s'exécutent successivement.

Prefetch Chain - Vérification que l'URL est bien dans la zone de couverture

Fetch Chain - Collecte du document pointé par l'URL

Extract Chain - Extraction des liens contenus dans le document

Write Chain - Stockage du document

Postprocess Chain - Finalisation de la collecte, écriture du rapport et mise à jour des informations sur la tâche d'exploration.

N'importe quel processus peut être déplacé ou modifié. Un processus personnel peut être rajouté dans cette chaîne. Il faut cependant faire attention à l'ordre et à la place des processus entre eux. Cela n'a pas de sens, par exemple, de procéder au stockage du document avant sa collecte.

8.2 La plate-forme d'intégration WebLab

Les travaux menés durant cette thèse s'inscrivent dans le cadre d'une thèse en convention CIFRE [2] entre le laboratoire du LITIS et l'entreprise AIRBUS DS. En plus de profiter du savoir et des travaux de recherche de ces deux entités, notre thèse a également bénéficié de supports logiciels. Le système DOWSER repose notamment sur le WebLab. Il s'agit d'une plate-forme d'intégration développée par l'équipe IPCC [3] de AIRBUS DS. Elle a pour but de faciliter le développement d'applications dédiées à la fouille et au traitement de documents multimédia via l'intégration de composants logiciels [Giroux et al., 2008]. Au travers son expertise dans le traitement d'informations non structurées, l'équipe IPCC développe cette plate-forme depuis 10 ans et exploite toute sa richesse dans des projets de recherche (Web-Content, VITALAS [4], Virtuoso [5]) et dans des projets industriels. Cette section présente l'architecture et le modèle d'échange du WebLab, ainsi que les solutions apportées par la plate-forme.

8.2.1 Architecture

La plate-forme WebLab se compose de trois différents niveaux comme l'illustre la figure 8.3, page 154. Le socle technique, appelé WebLab Core, permet l'interopérabilité entre les composants. C'est une base open source constituée d'un modèle d'échange, d'interfaces de services, d'un *Enterprise Service Bus* (ESB), et d'un portail pour l'interface

1. https ://webarchive.jira.com/wiki/display/Heritrix/Heritrix+3.0+and+3.1+User+Guide
2. Conventions Industrielles de Formation par la REcherche
3. Information Processing Control and Cognition
4. Video and image Indexing and reTrievAl in the LArge Scale, http://vitalas.ercim.org/
5. Versatile InfoRmation Toolkit for end-Users oriented Open Sources explOitation

homme-machine (IHM). L'ensemble des services logiciels et des composants d'IHM d'une application pouvant être intégrés avec le WebLab Core, sont regroupés dans le WebLab Services. Enfin, le troisième niveau appelé WebLab Applications, correspond au résultat obtenu en intégrant l'ensemble des WebLab Services via le WebLab Core. Le système DOWSER est basé sur cette architecture.

Les WebLab Applications reposent donc sur une architecture orientée service (SOA) intégrant des composants. Comme l'illustre la figure 8.4, page 155, chaque composant appartient à la couche *Components* et est chargé de gérer les données de la couche *Data*. Ces composants sont exposés via des services Web (WS), de la couche *Service*, reposant sur des interfaces spécialisées pour le traitement de documents multimédia. La couche *Integration* permet de faire communiquer les composants et de combiner les différentes fonctionnalités du système via l'*Enterprise Integration Patterns* (EIP). Par exemple, la détection de la langue est une fonctionnalité qui peut être combinée au composant d'extraction d'entités nommées afin d'améliorer globalement la pertinence de l'extraction. Cette couche permet d'orchestrer et de coordonner les différents services via des chaînes de traitements. La figure 8.5, page 155, est un exemple de chaîne de traitement dans laquelle :

1. un service de normalisation (basé sur l'outil Tika [6] transforme des documents (PDF, page Web, fichier vidéo, etc.) ou une archive Web (WARC) au format pivot WebLab ;
2. chaque document normalisé est enrichi sémantiquement via l'ajout d'annotations correspondant à sa langue (service *Language Detector*) et aux entités nommées présentes dans le document (service *Named Entities Extractor*) ;
3. chaque document normalisé est indexé avec ses annotations via un service d'indexation plein texte (SolR [7]) ou via un service d'indexation sémantique pour capitaliser les données dans une base de connaissances (*Repository*).

Enfin, la couche *Access* correspond à un portail Web hébergeant les applications métiers. Il fournit à l'utilisateur l'accès aux fonctionnalités de l'application.

8.2.2 Le modèle d'échange

Le modèle d'échange du WebLab permet de faciliter l'orchestration en rendant possible l'interopérabilité des services. C'est un format pivot qui limite la complexité de l'écriture des chaînes de traitement puisque c'est le seul type d'objet échangé entre les services. Il permet d'éviter des coûts de traitement liés à des transformations d'objets entre chaque service. Ce format pivot est modélisé en XML via la syntaxe XML Schema (XSD) et suit les recommandations W3C. Au sein de ce format pivot se trouve le concept de *Resource* identifiable par une URI [9]. Une *Resource* contient un ensemble d'*Annotations* permettant de la décrire. Ces *Annotations* sont des sous-classes d'un second concept de base du modèle pivot : la *PieceOfKnowledge* (POK). Il s'agit d'une enveloppe renfermant de la connaissance au format RDF/XML, ce qui permet d'annoter les ressources via l'utilisation d'ontologies. Pour finir, le modèle est également défini au travers de *MediaUnit* représentant les différents types de multimédia (audio, image, texte et vidéo) analysables par le WebLab. La *MediaUnit Text* nous intéresse tout particulièrement dans le cadre du

6. http://tika.apache.org/
7. https ://lucene.apache.org/solr/
8. http://weblab-project.org/index.php?title=Architecture
9. Uniform Resource Identifier

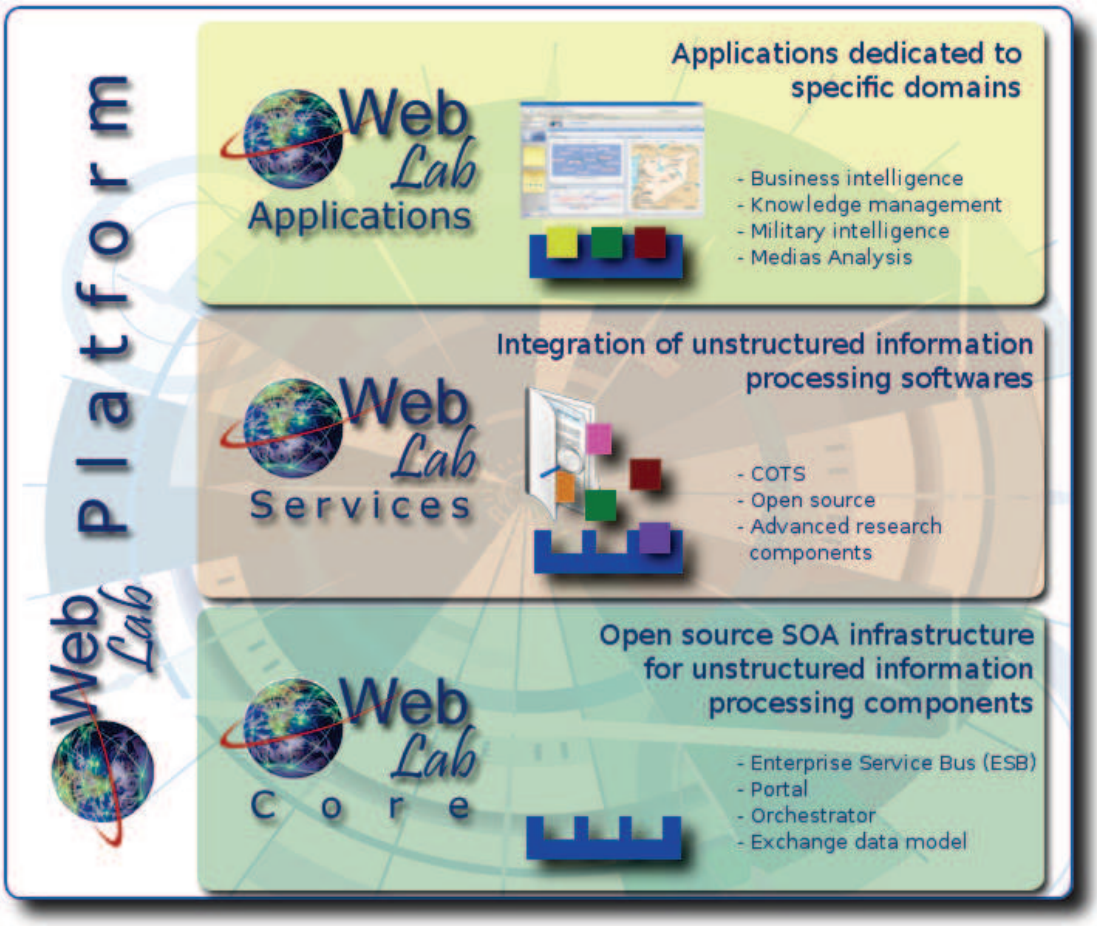


FIGURE 8.3 – Les différentes couches de la plate-forme WebLab [8]

système DOWSER. Enfin, les *Segment* permettent de pointer des éléments d'intérêt au sein d'une *MediaUnit*. Par exemple, l'utilisation de *LinearSegment* est faite pour repérer les entités nommés présentes dans une *MediaUnit Text*. La figure 8.6 représente ce format pivot via une modélisation UML. Afin de renforcer l'interopérabilité entre les services, les entrées/sorties des services utilisent ce modèle pivot en respectant des contrats modélisés sous forme d'interfaces génériques. Les interfaces et leurs méthodes sont illustrées dans la figure 8.7, page 157. Parmi elles, l'interface *Analyser* permet de définir les services qui enrichissent d'annotations les ressources reçues en entrée. C'est une interface fortement employée dans le système DOWSER.

8.2.3 Avantages et limites de la plate-forme Weblab

La plate-forme WebLab est adaptée à la mise en oeuvre d'applications de veille et, par extension, de découverte de sources. L'utilisation de cette plate-forme pour nos travaux de thèse est donc particulièrement adaptée, surtout au regard des avantages fournis par le WebLab :

10. http://weblab-project.org/index.php?title=WebLab_1.2.5#Exchange_Model

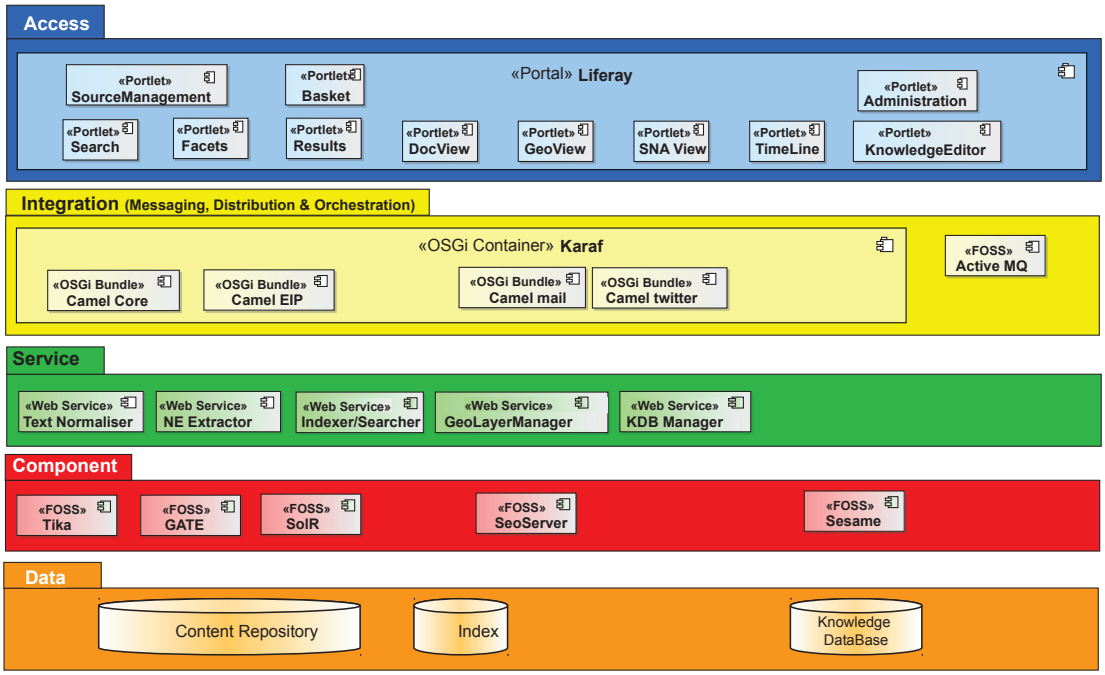


FIGURE 8.4 – Vue globale de l'architecture de la plate-forme WebLab

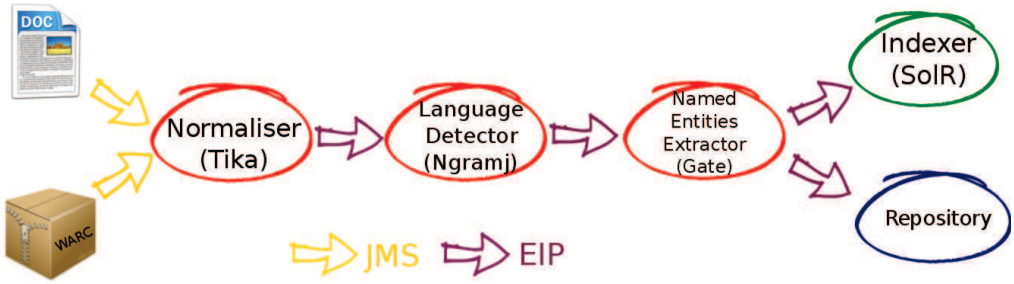


FIGURE 8.5 – Exemple de chaîne de traitement WebLab

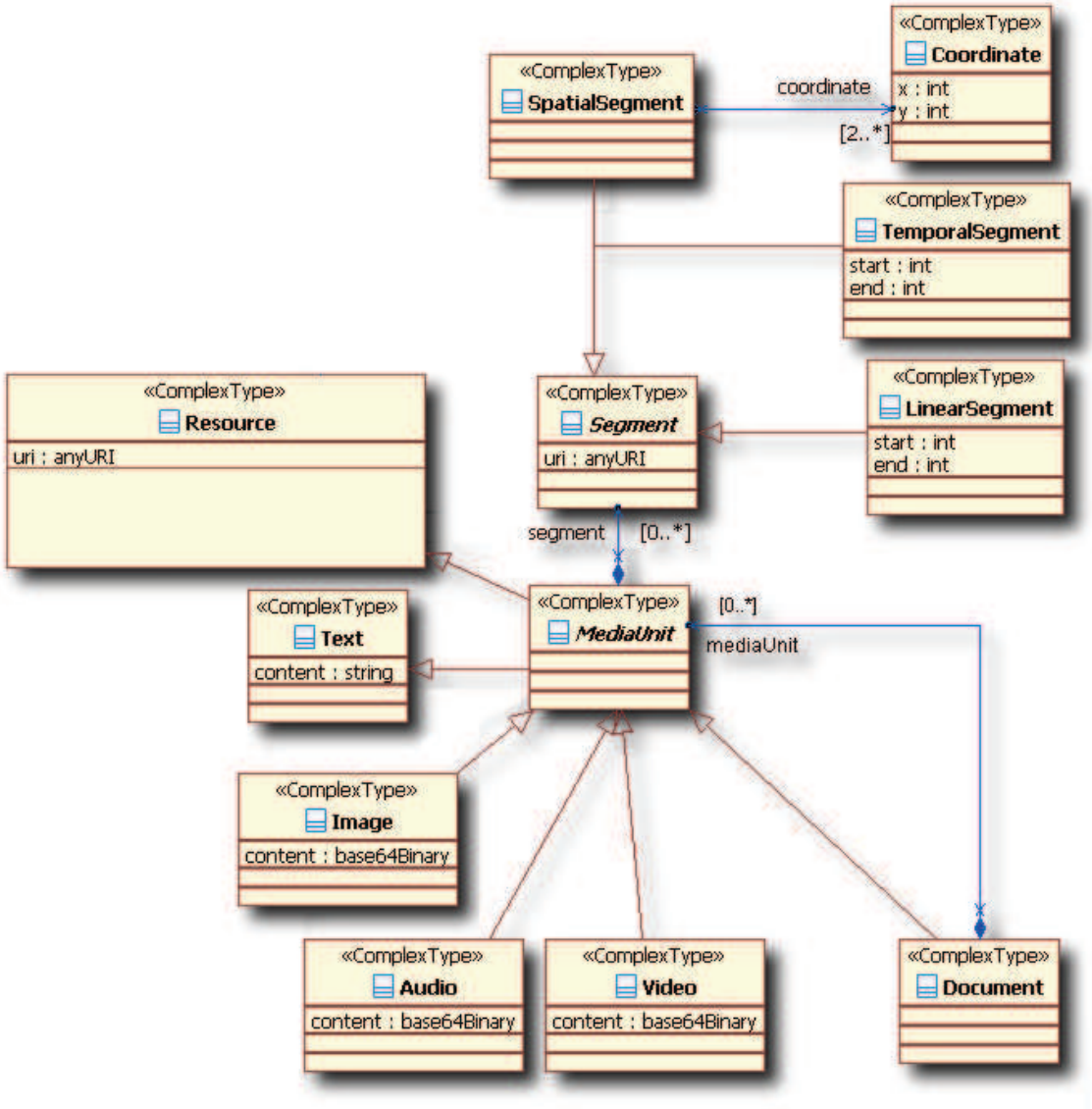


FIGURE 8.6 – Modélisation UML du format d'échange WebLab [10]

Traitement de masse : Que ce soit dans une application de veille ou de découverte de sources, la masse des documents à traiter est importante. Le WebLab a l'avantage de s'adapter aux besoins en redimensionnant l'application et en répartissant les calculs, l'espace de stockage, les besoins réseaux, etc.

Adaptabilité fonctionnelle : Grâce à l'interopérabilité et aux interfaces génériques, l'ajout de services et la création de nouvelles chaînes de traitement sont facilités. Ainsi, l'intégration de nouvelles fonctionnalités, adaptées aux besoins de l'application, est simple et rapide.

IHM personnalisable : Outre les services, l'ajout de vues au sein de l'IHM est grandement simplifié au travers l'utilisation de la technologie Portlet. Des interfaces spécifiques avec des accès restreints peuvent être mises en place. L'ensemble de l'application reste accessible au travers d'un seul et même portail, personnalisable en fonction des besoins et des utilisateurs.

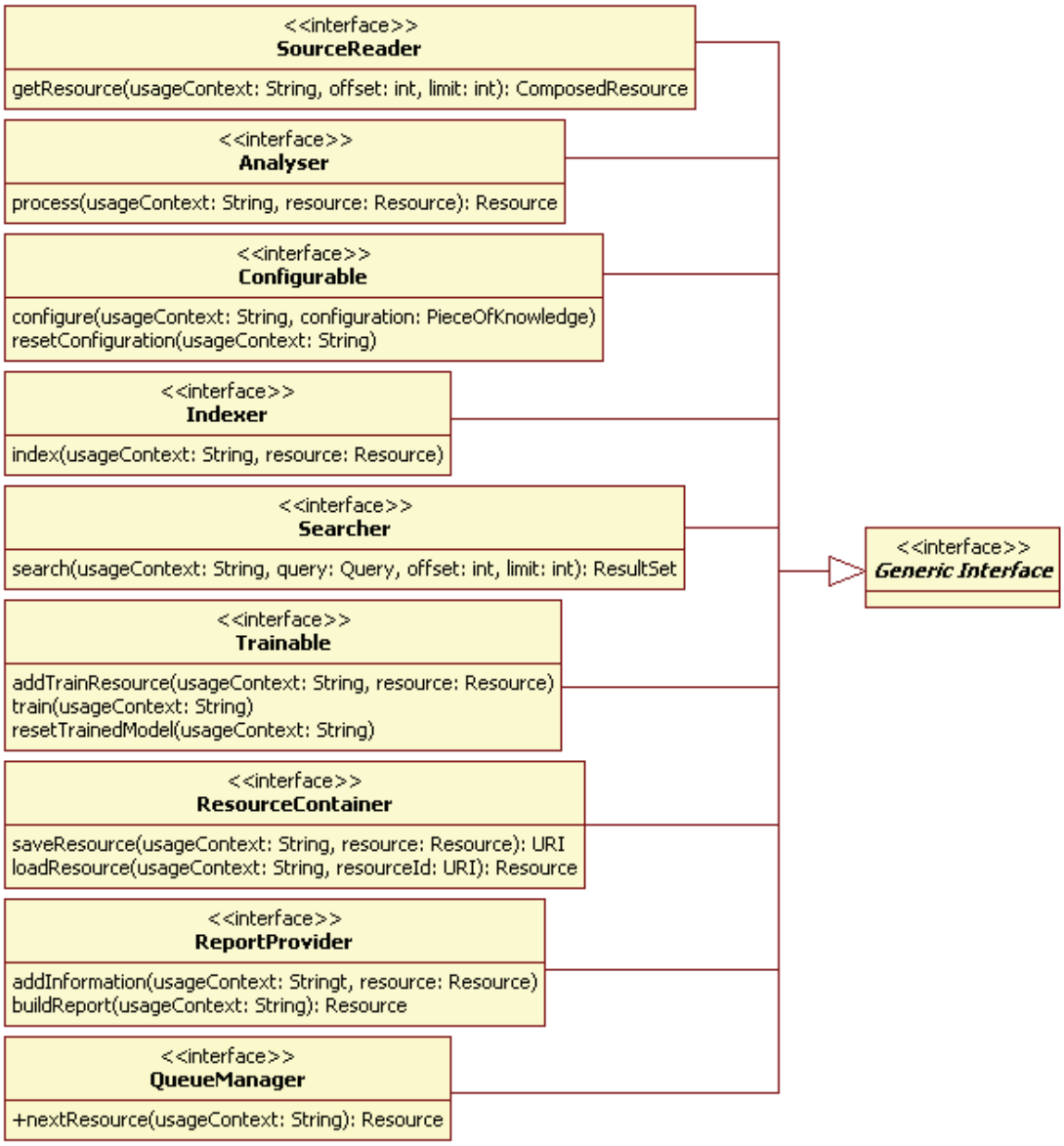


FIGURE 8.7 – Vue des interfaces génériques du model WebLab

La flexibilité de la plate-forme Weblab et ses fonctionnalités de base pour le traitement de données non structurées en font une base solide pour notre système de découverte de nouvelles sources d'intérêt. Cependant, l'intégration de nouveaux composants propres à DOWSER est une tâche complexe qui nécessite de maîtriser les technologies utilisées et d'avoir des compétences d'analyste programmeur. Ce ne sont pas des compétences propres aux experts du renseignement. Une partie des travaux de cette thèse a été consacrée à la réalisation de composants intégrables au WebLab afin de constituer un outil utilisable par les experts du renseignement basé sur notre approche théorique décrite dans le chapitre précédent.

8.3 Les composants du système DOWSER

Cette section présente les composants développés spécifiquement pour notre système au sein du robot d'exploration Heritrix et de la plate-forme WebLab.

8.3.1 Modifications d'Heritrix

Le robot d'exploration Heritrix n'est initialement pas prévu pour explorer le Web de manière ciblée. Cependant, il comporte des composants permettant de donner une priorité aux URLs à ajouter dans la file de liens à explorer. Le composant *BaseUriPrecedencePolicy* affecte une valeur unique à toutes les URLs. Cette valeur peut être changée durant la collecte pour les nouvelles URLs à collecter. Ceci permet, par exemple, de baisser la priorité d'un trop grand nombre d'URLs provenant d'un même domaine. Le composant *HopsUriPrecedencePolicy* permet d'affecter une priorité à une URL en fonction de sa distance avec les premières pages collectées. Cette distance correspond au nombre de liens parcourus pour accéder à cette nouvelle URL.

Dans notre cas, la priorité d'une URL est liée à la pertinence de sa page parent par rapport au profil utilisateur. Or, le score de pertinence n'est pas calculé à la volée dans Heritrix. Cela nécessite un traitement spécifique sur le contenu des pages opéré par le biais de la plate-forme Weblab. La gestion des URLs à visiter n'est donc plus laissée à Heritrix : la fonction *schedule* est retirée comme l'illustre la figure 8.8. À la place, la fonctionnalité *ActionFolder* est activée. Elle permet de donner à la volée de nouvelles URLs à visiter à Heritrix en écrivant directement les URLs dans un fichier lu à la volée par le robot d'exploration. Ainsi, les liens à visiter en priorité sont écrits avant les autres dans ce fichier.

La figure 8.8 illustre également le remplacement du composant *Write* par *UW-Write* pour *Unique Warc Writer*. Le composant *Write* contient les instructions de stockage des documents. Heritrix crée des fichiers Warcs qui sont des fichiers compressés contenant un ensemble de pages collectées. Il est initialement possible dans Heritrix de définir la taille maximale d'un fichier Warc. Cependant, dans DOWSER, nous voulons avoir accès au contenu d'une page dès qu'elle est collectée afin de calculer sa pertinence au plus tôt et non pas lorsque la taille du Warc est atteinte. Ainsi, le composant *UW-Write* permet de créer un fichier Warc pour chaque page Web collectée et de fournir rapidement un accès à celle-ci.

Grâce à ces modifications, notre version d'Heritrix est en mesure de cibler son exploration sur les pages jugées prioritaires qui lui sont données en entrée tout en assurant un accès dès que possible aux pages collectées. Ces pages sont alors traitées spécifiquement au travers de la plate-forme WebLab.

8.3.2 Intégration à la plate-forme WebLab

Que ce soit au niveau de l'interface homme/machine ou au niveau de la chaîne de traitement, il a fallu enrichir la plate-forme WebLab avec des *portlets* et des composants spécifiques aux fonctionnalités de DOWSER.

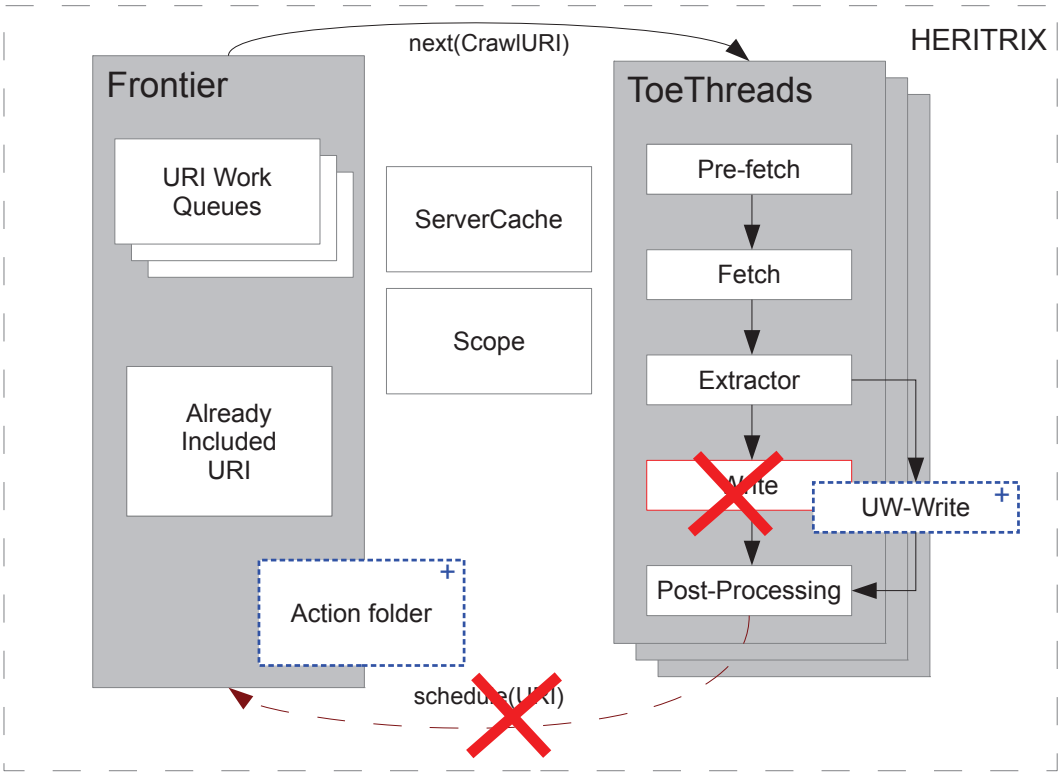


FIGURE 8.8 – Fonctionnement modifié d'Heritrix

a. Web services et chaîne de traitement

La chaîne de traitement du système DOWSER doit prendre en entrée une archive Warc produite par Heritrix et indexer son contenu en l'enrichissant d'informations utiles pour la collecte et la présentation des sources à l'utilisateur. Cette chaîne est illustrée dans la figure 8.9. Les composants représentés par des rectangles sont des composants pré-existants dans la plate-forme WebLab, ceux représentés par des cercles on été conçus pour les besoins du système DOWSER.

Tika Normalizer Ce service permet de normaliser le contenu natif d'une page Web présente dans une archive Warc. Tika [11] est utilisé pour extraire le contenu de la page. Les publicités et les menus sont retirés grâce à l'outil Boilerpipe [12] intégré à Tika. Le document se présente uniquement sous forme de MediaUnit Text car les images et les vidéos, non traitées dans notre approche, sont retirées. Il est enrichi par les métadonnées éditoriales présentes dans la page Web tel que son titre, son domaine ou sa date de dernière modification ainsi que des métadonnées déterminées par Tika comme la langue, le format, etc. Ce service, présent dans la plate-forme Weblab, a été modifié afin d'enrichir le document avec des annotations liées à la collecte. Le nom du projet, permettant de lier un document à un profil opérationnel, est notamment ajouté. De plus, toutes les URLs présentes dans le

11. http://tika.apache.org
12. http://boilerpipe-web.appspot.com

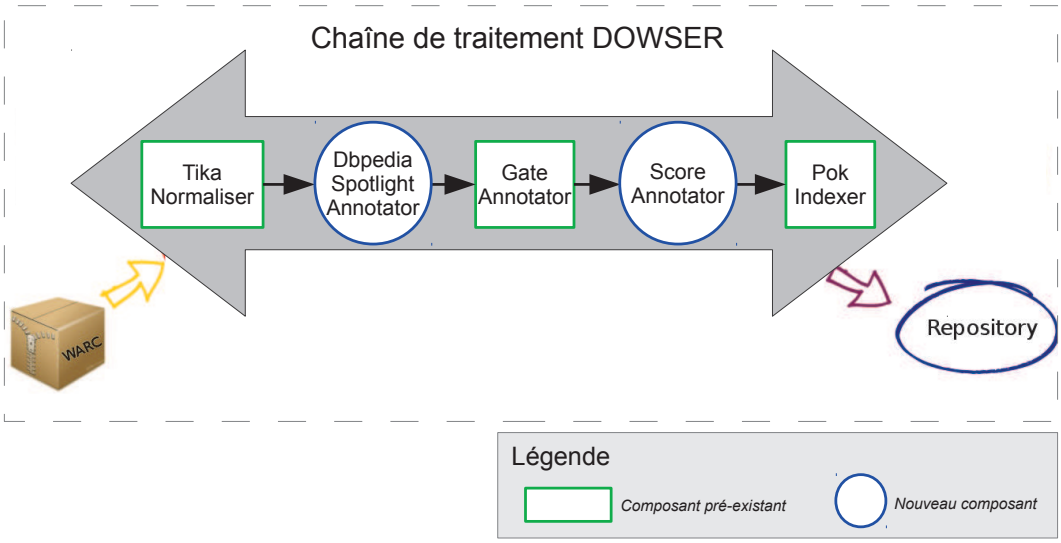


FIGURE 8.9 – La chaîne de traitement DOWSER

document sont également ajoutées sous forme d'annotations. Ces annotations sont utilisées par la suite pour fournir à Heritrix les URLs à visiter.

DBPedia Spotlight Annotator Ce service permet d'enrichir le document avec les concepts extraits à l'aide de l'outil DBPedia Spotlight [13]. Ce dernier est un service Web qui expose ses fonctionnalités en REST (REpresentational State Transfer). Il prend en entrée un texte et il fournit en sortie le même texte annoté de concepts DBPedia. Cette sortie est utilisée par notre service pour enrichir notre document.

Gate Annotator Ce service permet d'enrichir le document avec les mots-clés extraits du texte à l'aide du framework open source d'analyse de texte Gate [14]. Il est configuré de sorte à utiliser son plugin *NP Chunker* basé sur l'approche Noun Phrase Chunker [Ramshaw & Marcus, 1995].

Score Annotator Comme son nom l'indique, ce service a pour objectif d'annoter le document avec son score de similarité. Il utilise les annotations fournies par Tika contenant les informations de collecte afin de rapatrier le profil opérationnel associé au document. Ce service se connecte donc directement à la base de connaissances où sont stockés les profils opérationnels. Il exploite les annotations contenant les concepts et les mots-clés extraits pour calculer le score de similarité pour la couverture terminologique et thématique du besoin, présenté dans la section 6.3.2, page 106. Pour la mesure de similarité thématique, la distance séparant deux instances de concept a besoin d'être connue. Aussi, ce service interroge un service indépendant appelé DBPedia lookup. Celui-ci permet de se connecter à une base de connaissances contenant l'ensemble des instances de concept DBPedia et leurs catégories. Il permet ainsi de récupérer les informations sur une instance fournie en

13. http://spotlight.dbpedia.org
14. https ://gate.ac.uk

entrée. Les concepts DBPedia indexés sont ceux fournis dans le *dataset* de juillet 2011 [15]. Les deux scores de similarité obtenus, ainsi que le score de similarité adaptatif, sont ajoutés sous forme d'annotations au document et correspondent à la pertinence du document par rapport au profil opérationnel.

Pok Indexer Ce dernier service transforme un document au format pivot annoté en une description sémantique complète au format RDF/XML encapsulée dans un POK. La description contient les informations de structure du document et les annotations produites lors de l'analyse du document. Les descriptions sémantiques sont ensuite indexées dans une base de connaissances via une requête SPARQL [16]. Initialement présente dans la plate-forme WebLab, la requête SPARQL de ce service a été modifiée afin de filtrer les informations annotées à indexer. Les annotations, fournies par les 4 services précédents, sont stockées mais le contenu textuel du document ne l'est pas.

Routes camel L'orchestration des services Web sous Karaf, fourni par la plate-forme WebLab, permet également d'écrire des routes Camel [17] spécifiques aux besoins du système. Dans DOWSER, deux routes ont été créées afin de faire la jonction entre le robot d'exploration Heritrix, notre chaîne de traitement et la base de connaissances :

WarcListenerRoute : cette route permet d'écouter le dossier dans lequel Heritrix stocke les archives Warcs des pages qu'il collecte. Dès qu'une archive apparaît, elle est envoyée à la chaîne de traitement.

SeedsTimerRoute : cette seconde route permet de fournir à Heritrix les URLs qu'il doit explorer et collecter. Elle utilise une requête SPARQL qui permet de récupérer le top des URLs jugées d'intérêt en utilisant le score de pertinence des pages parentes présentes dans la base de connaissances. La requête donne un avantage aux URLs dont le nom de domaine n'a pas encore été visité.

b. La base de connaissances

La base de connaissances est un entrepôt de triplets RDF appelé *Repository*. Elle est accessible via un point d'accès (SPARQL endpoints) qui est fourni aux différents services et composants qui communiquent avec elle. Dans DOWSER, cette base de connaissances est découpée en trois sous-ensembles comme l'illustre la figure 8.10. La gestion de sous-ensemble de triplets est permise avec les graphes nommés (Named Graph) et SPARQL dans sa version 1.1.

Le premier sous-ensemble contient toutes les informations concernant les profils opérationnels. Les triplets décrivent les classes et les attributs présents dans l'ontologie DOWSER définie dans la section 6.3, page 104. Le second sous-ensemble est constitué de documents WebLab annotés et indexés par le service *Pok Indexer* (voir page 161). Ces documents correspondent aux pages Web collectées par le robot d'exploration. Enfin, le dernier sous-ensemble contient des triplets décrivant l'ontologie DBPedia en juillet 2011. Les concepts retenus sont ceux en anglais et les propriétés indexées sont l'étiquette (*label*),

15. http://wiki.dbpedia.org/Downloads37?v=u9u
16. http://www.w3.org/TR/rdf-sparql-query/
17. http://camel.apache.org/routes.html

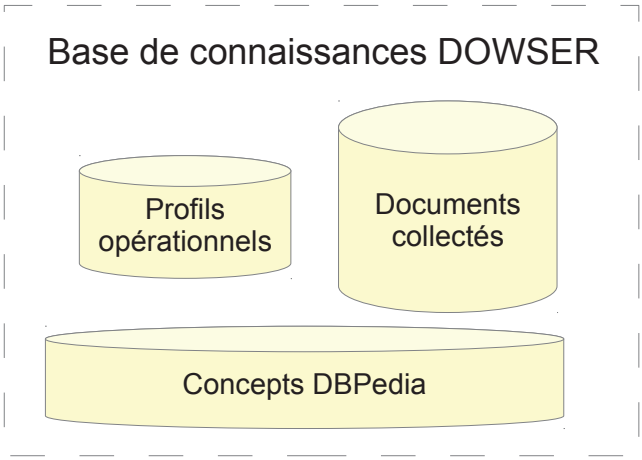


FIGURE 8.10 – La base de connaissances DOWSER

le type, les catégories ainsi que les liens vers les pages internes et externes (isPrimaryTopicOf, wikiPageExternalLink). Les catégories sont utilisées pour la mesure de similarité entre concepts (voir section 6.3.2 page 106) et les liens internes et externes pour le module d'extension du robot d'exploration (voir section 6.2.2 page 103).

c. Interface Homme/Machine

Afin de concevoir des vues spécifiques à DOWSER, nous avons profité de la personnalisation d'IHM offerte par la plate-forme WebLab. Nous avons donc utilisé la norme Portlet 2.0, JSR-28618, qui permet de définir des composants d'interface indépendants les uns des autres, et intégrable dans le portail fourni par la plate-forme WebLab. Les vues du système DOWSER diffèrent en partie de celles proposées lors de la première expérimentation (voir section 7.1.2 page 120). Contrairement au prototype d'expérimentation, le profil opérationnel n'est pas évalué dans le système DOWSER. Il est construit automatiquement à partir des sources d'intérêt fournies par l'utilisateur. Ce dernier n'a alors pas connaissance du profil associé à son besoin. De plus, les sources découvertes sont consultables par l'utilisateur dès qu'elles ont été indexées et non pas 12h après le lancement de la collecte comme dans l'expérimentation. Ainsi, notre système se dote d'une seule page contenant une *portlet* de gestion des sources et une de découverte de sources. L'interface homme/machine est limitée en fonctionnalités et en nombre de pages afin d'apporter aux experts du renseignement un outil facile à prendre en main.

La Portlet de gestion des sources Cette vue permet à l'utilisateur de gérer ses sources d'intérêt et de les classer dans des dossiers et sous-dossiers. L'utilisateur peut ajouter, modifier, supprimer des sources et des dossiers et les déplacer à sa guise. L'impression d'écran 8.11 illustre son organisation en arbre. Chaque dossier correspond ainsi à un besoin défini au travers des sources qu'il contient.

La Portlet de découverte de sources Cette vue permet de créer le profil, lancer la découverte de sources sur un besoin donné et consulter les résultats. Comme l'illustre

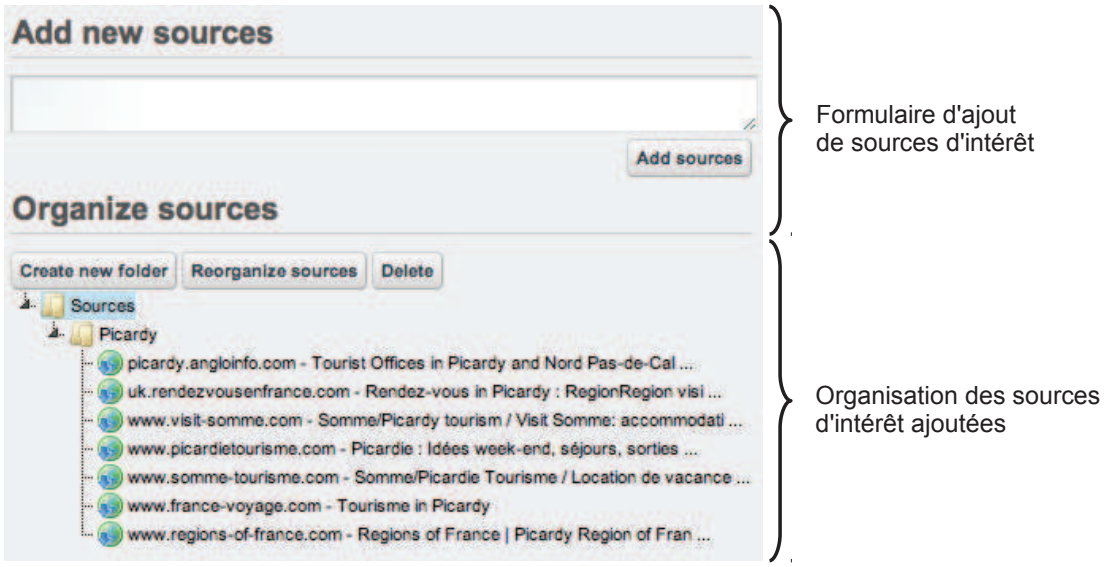


FIGURE 8.11 – Impression d'écran de la Portlet de gestion des sources

l'impression d'écran 8.12, deux menus déroulants sont accessibles à l'utilisateur :

- Le premier menu contient les dossiers créés par l'utilisateur dans la Portlet de gestion des sources. En sélectionnant un dossier, l'utilisateur indique au système DOWSER qu'il souhaite une découverte de sources sur la thématique associée à ce dossier. Le profil opérationnel est alors automatiquement créé à partir des sources contenues dans le dossier et une tâche d'exploration est lancée.
- Le second menu correspond aux tâches d'exploration en cours. Un dossier sélectionné dans la première liste se retrouve ensuite dans celle-ci. En sélectionnant un dossier de cette liste, le système fournit à l'utilisateur les 10 meilleures sources découvertes.

Lorsque les sources découvertes sont affichées, l'utilisateur peut les juger d'intérêt ou non. Une source jugée d'intérêt est ajoutée au dossier correspondent à la tâche de découverte de sources et elle apparaît donc dans la vue de gestion des sources de celui-ci. Les sources jugées non pertinentes ne sont plus affichées à l'utilisateur. Ce dernier peut également mettre de côté une source pour la juger plus tard. Un bouton étiqueté *Show sources temporarily kept* permet d'afficher les sources gardées temporairement et de les juger comme étant pertinentes ou non pertinentes. Chaque jugement est enregistré par le système et permet de mettre à jour le profil opérationnel (voir section 5.3, page 81) lorsque l'utilisateur termine sa session de recherche.

Au travers de ces deux composants, l'utilisateur a accès à l'intégralité des fonctionnalités offertes par le système DOWSER. L'interface se veut ergonomique afin d'être utilisable quotidiennement par les experts du renseignement.

8.3.3 Vue d'ensemble du système DOWSER

La figure 8.13, page 165, illustre le fonctionnement global du système DOWSER. Notre version du robot d'exploration Heritrix cohabite avec la plate-forme WebLab en commu-

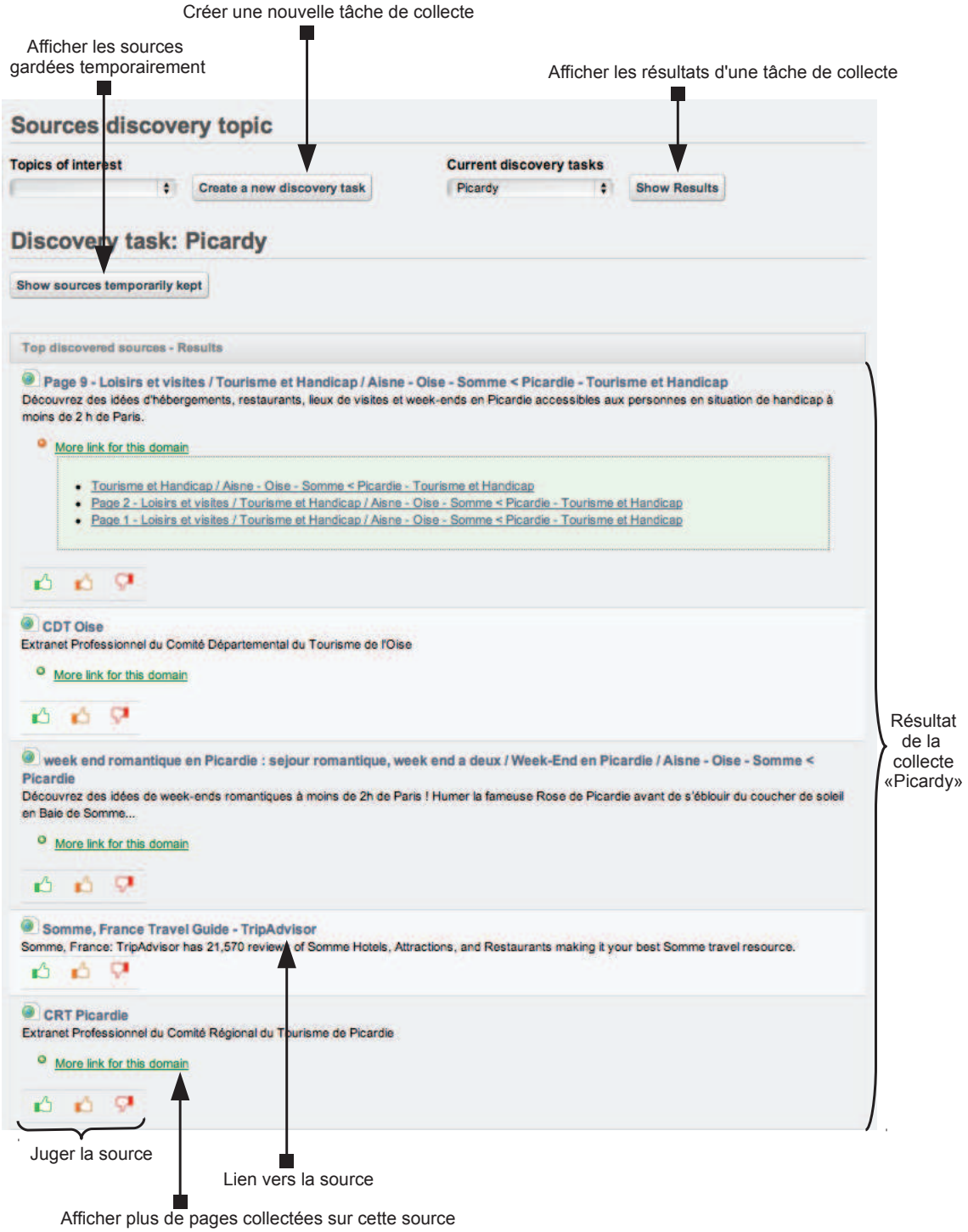


FIGURE 8.12 – Impression d'écran de la Portlet de découverte de sources

niquant au travers de routes Camel (trait en pointillé bleu et vert) et de l'IHM (trait orange). La base de connaissances est enrichie de profils opérationnels grâce à l'IHM et de pages Web collectées traitées par la chaîne de traitement. Le tout offre un système de dé-

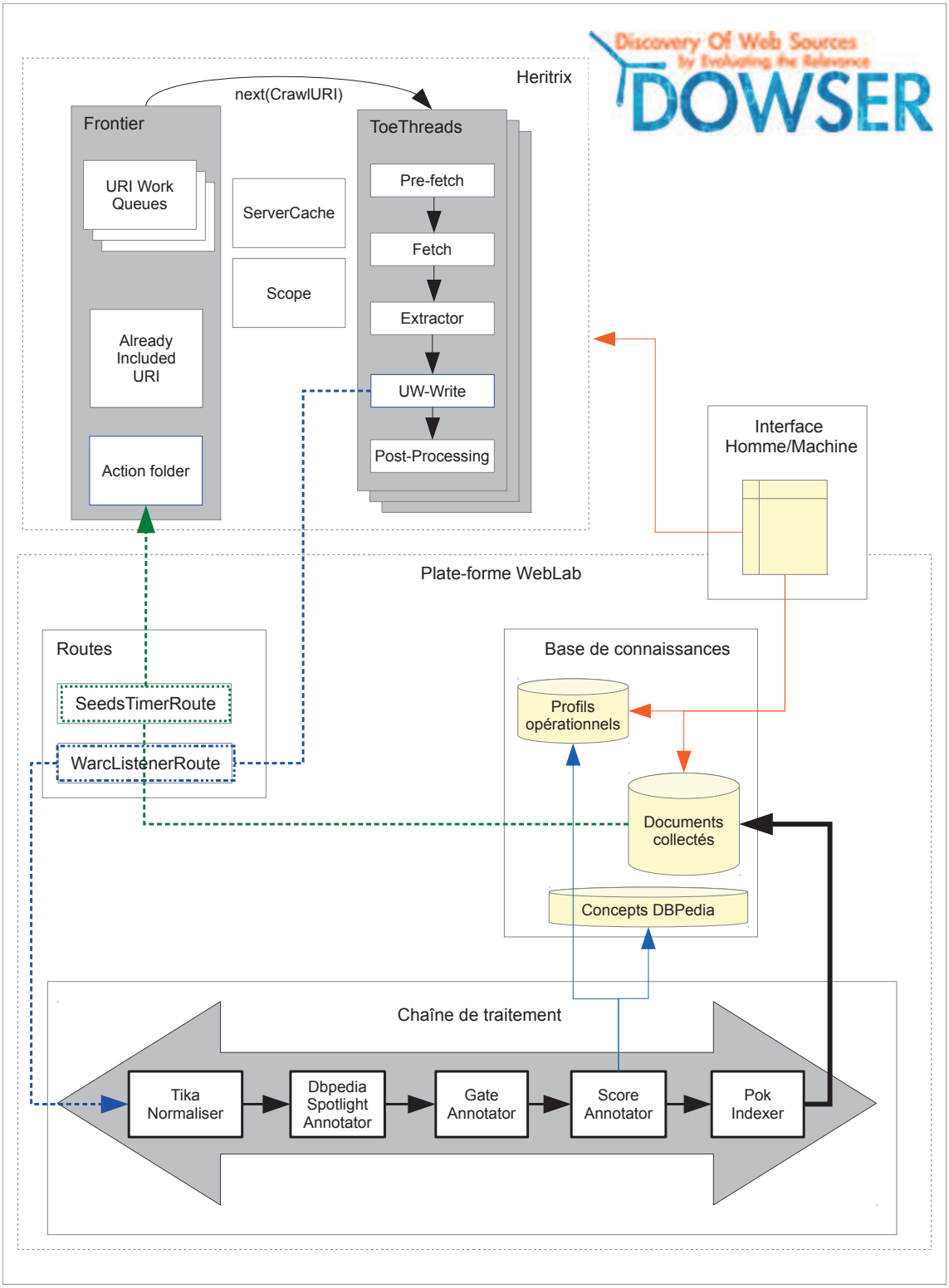


FIGURE 8.13 – Vue d'ensemble du système DOWSER

couverte de sources fonctionnel et simple d'utilisation pour les experts du renseignement. L'architecture de DOWSER lui permet de s'intégrer facilement dans des projets tierces

reposant sur la plate-forme WebLab.

8.4 Intégration projet

Comme expliqué précédemment, l'architecture de DOWSER lui offre la possibilité de s'intégrer dans des projets de plus grande envergure dont la tâche de découverte de sources ne serait qu'une sous partie fonctionnelle. Les expérimentations conduites sur notre système ont permis de valider notre approche. DOWSER a alors pu être intégré dans des projets menés par l'équipe IPCC d'Airbus Defense and Space. Dans la première partie de cette section, le comportement de notre système en conditions réelles lors de son utilisation par des experts du renseignement est introduit. Le projet TWIRL, intégrant DOWSER, est présenté dans une seconde partie.

8.4.1 Comportement en conditions réelles

Le système DOWSER a été utilisé dans un cadre opérationnel en adéquation avec notre problématique. Il n'est pas possible de détailler les tenants et les aboutissements de ce projet de part son aspect sensible. L'utilisation qui a été faite de DOWSER servait à un service du Ministère de la Défense pour retrouver des blogs en rapport avec la guerre au Mali. Les retours exprimés par les experts du renseignements étaient encourageants dans la mesure où notre système leur a été utile. Dans de telles conditions, l'utilisation de DOWSER ne permet pas d'en ressortir des données exploitables pour améliorer l'approche comme cela l'a été avec nos expérimentations. Enfin, les possibilités fonctionnelles offertes par DOWSER ont entraîné Airbus DS à déposer un brevet industriel sur nos travaux mais aussi à l'intégrer dans le projet TWIRL. Ce dernier offre un cadre applicatif intéressant au système DOWSER. Cependant, la problématique de ce projet est un peu éloignée de celle à laquelle notre approche doit répondre. Elle n'implique pas directement des sources impopulaires sur des sujets sensibles et elle ne fait pas intervenir des experts du renseignement.

8.4.2 Intégration au projet TWIRL

TWIRL est un projet ITEA2 [18] de 25 mois débuté en mars 2012. TWIRL signifie *Twinning virtual World (on-line) Information with Real world (off-Line) data sources.* L'équipe IPCC participe à ce projet en tant que leader technique en *text-mining*, découverte de sources et collecte de données. Elle a également le rôle d'intégrateur puisque le projet utilise la plate-forme d'intégration WebLab.

a. Objectifs du projet TWIRL

Comme présenté dans le premier chapitre de ce manuscrit, l'utilisation d'Internet a augmenté ces dernières années à un rythme important et est devenue omniprésente dans nos vies personnelles et professionnelles. Le projet TWIRL s'inscrit dans ce contexte. La diversité des informations disponibles sur Internet est devenue telle que tout produit ou service, à destination des individus, des entreprises ou des administrations, offre une valeur

18. https://itea3.org/project/twirl.html

ajoutée significative aux activités quotidiennes, qui peut être encore améliorée, notamment grâce à la réalité augmentée. Ces informations, issues d'une multitude de sources (sites Web, bases de données publiques, capteurs, Systèmes d'Information (SI) d'entreprises, réseaux sociaux, etc.), soulèvent néanmoins les problématiques suivantes :

- énormes quantités de données accessibles,
- hétérogénéité des formats et types de données (textes, commentaires, vidéos, images, etc.),
- difficultés d'agréger des données provenant de sources multiples (sites Web, forums, réseaux sociaux, etc.),
- qualité des données disponibles (y compris les problèmes d'ambiguïté).

L'objectif de TWIRL est de résoudre ces problématiques en proposant un environnement permettant d'agréger ces données hétérogènes et de les utiliser afin d'enrichir les applications les plus utilisées quotidiennement. Cet environnement permet de fournir :

- une architecture de référence permettant la construction d'applications TWIRL,
- des connecteurs vers différentes sources de données,
- des mécanismes de gestion et de capitalisation des connaissances,
- des modules de traitement des informations.

Les résultats du projet sont intégrés dans deux démonstrateurs *Augmented Life* et *Augmented business* qui permettent de valider l'approche TWIRL sur des cas concrets. Le consortium français s'occupe du premier démonstrateur *Augmented Life*. Airbus D&S a pour rôle d'intégrer les composants des différents acteurs du projet grâce à sa plate-forme WebLab. De plus, les compétences de l'équipe IPCC en text-mining, découverte de sources et collecte de données sont nécessaires pour ce démonstrateur.

b. La tâche de découverte de sources

Nos travaux de thèse sont exploités dans une tâche de découverte de sources implémentée dans le démonstrateur *Augmented Life*. Ce démonstrateur permet d'agréger des données provenant de différentes sources (Web, réseaux sociaux, bases de données en ligne, etc.) afin d'aider l'utilisateur à préparer et à mener à bien une activité ou une action de la vie réelle.

Parmi les applications visées par ce démonstrateur, on peut citer :

- le support pour la préparation d'un voyage ou d'une sortie (avant, pendant et après) ;
- le support pour l'organisation d'un événement ;
- le support pour l'acquisition d'un bien immobilier (choix, informations administratives et financiers, informations environnementales, etc.) ;

La réalisation de ces tâches a été considérablement facilitée par l'apparition des nouvelles technologies et par l'abondance des informations disponibles en ligne. Cependant, les utilisateurs sont rapidement submergés par des informations qui ne sont pas nécessairement utiles à leur activité.

Les données permettant de réaliser un tel démonstrateur proviennent essentiellement des sources suivantes :

- Sites Web et moteurs de recherche généralistes comme Wikipedia, Yahoo, Google, etc.
- Sites Web spécialisés (administratifs, monuments, musées, allociné, etc.)
- Bases de données libres de géolocalisation

– Sites Web de cartographie comme Google maps, Maporama, etc.
– Réseaux sociaux (Twitter, Facebook, Ipernity, etc.)

La combinaison et l'enrichissement de ces données suivent le processus suivant :

1. Identifier les sources de données pertinentes
2. Collecter les données à partir des différentes sources
3. Extraire les connaissances contenues dans ces données
4. Filtrer et enrichir ces connaissances
5. Fournir une vue synthétique et des recommandations répondant au besoin de l'utilisateur

Le système DOWSER intervient dans le second type de sources : la découverte de sources Web spécialisées. Notre approche doit être en mesure d'identifier ces sources et de les collecter (étape 1 et 2 du processus TWIRL). Le besoin ne s'inscrit pas dans une démarche de veille de type ROSO comme le ferait un expert du renseignement. Cependant, il s'agit d'une tâche assez spécifique pour que DOWSER puisse être utile afin de renforcer et compléter les données trouvées au travers d'un moteur de recherche. Le projet TWIRL étant basé sur la plate-forme WebLab, aucune modification concernant l'architecture de DOWSER n'a été nécessaire. Notre système est intégré tel quel et il est accessible au travers d'une page de l'IHM dédiée à la découverte de sources.

Le principe d'utilisation reste le même : l'utilisateur fournit en entrée un ensemble d'URLs représentatif de son besoin en information. Dans le cas de TWIRL, l'exemple appliqué est la recherche d'activité touristique en Picardie. Bien loin d'être une tâche sensible, elle reste néanmoins spécifique. L'avantage de DOWSER est d'apporter des sources d'intérêt automatiquement. Dans notre approche, la découverte de sources ne tient pas compte de la popularité des pages. Ceci permet ainsi de trouver des activités intéressantes qui ne profitent pas d'une publicité ou d'une popularité lui permettant une bonne position dans les résultats des moteurs de recherche. Les premiers tests opérés sur ce démonstrateur illustrent l'intérêt et l'efficacité de DOWSER pour enrichir les sources de données pertinentes autour d'une activité spécifique. La suite des travaux et des expérimentations sur ce projet permettront d'avoir des résultats plus précis sur les avantages d'utilisation du système DOWSER dans un tel cadre de découverte de sources d'information.

8.4.3 Synthèse

L'architecture du système DOWSER repose sur l'utilisation de la plate-forme WebLab qui permet d'intégrer facilement différents modules liés au traitement des sources découvertes. Ces dernières sont collectées via un robot d'exploration open-source adapté aux besoins de notre approche. Au final, au travers de la présentation de l'architecture du système DOWSER, ce chapitre a montré la capacité de notre approche à s'intégrer dans des projets grâce à cette modularité et à son utilisation d'outils open-sources. Le projet TWIRL en est l'illustration puisqu'il exploite notre approche dans une tâche de découverte de sources. Enfin, DOWSER peut également être utilisé comme un système autonome et contribuer à la veille stratégique pour aider les experts du renseignement.

Quatrième partie

Conclusion et Perspectives

CHAPITRE 9

CONCLUSION

Dans ce chapitre, nous présentons les apports et les limites de nos propositions concernant la modélisation du profil opérationnel, la représentation du besoin utilisateur, la découverte de sources d'intérêt, le système DOWSER qui en découle et les expérimentations menées. En second partie de ce chapitre, nous identifions les limites de notre approche et de nos travaux. Les perspectives possibles sont introduites en dernière partie.

9.1 Synthèse des contributions

Le second chapitre de cette thèse a permis de présenter les approches et les travaux existants en matière de RI et de DI. En considérant les besoins des experts du renseignement, nous avons mis en exergue les limites de ces approches. Dans le cadre du Renseignement d'Origine Sources Ouvertes, les experts recherchent des informations très spécifiques sur des sujets qui peuvent être sensibles. Les approches existantes de modélisation du profil utilisateur ne sont pas adaptées aux besoins spécifiques de la veille stratégique sur le Web. De plus, la DI doit prendre en considération que les pages à collecter sont difficiles à trouver à cause de leur contenu sensible et de leur non-popularités. Dans cette section, nous présentons les contributions de nos travaux d'un point vue scientifique et opérationnel.

9.1.1 Contributions scientifiques

Afin de pallier le manque de représentativité du besoin dans les systèmes de RI et de DI, nous avons proposé une double représentation vectorielle. Ainsi, la représentation unique au travers d'une requête ou d'un ensemble de termes (mots-clés ou concepts) est remplacée par une double représentation complémentaire : la première est une représentation conceptuelle pour une couverture thématique du besoin et la seconde est basée sur des mots-clés pour une couverture terminologique. La représentation thématique permet de couvrir le besoin dans son ensemble avec un point vue global sur le sujet. À contrario, la représentation terminologique tend à modéliser les éléments de langage spécifiques à un

sujet de recherche afin de couvrir avec plus de précision le besoin. Un calibrage paramétrique a permis de déterminer l'importance à donner à ces deux représentations afin de couvrir de manière optimale le besoin opérationnel. Les expérimentations ont montré que cette double représentation complémentaire affecte favorablement les résultats de notre système par rapport à l'exploitation d'une représentation simple du besoin.

Cette double représentation est intégrée au sein d'un profil opérationnel avec respectivement un vecteur de mots-clés pondérés et un vecteur d'instances de concept pondérés. Ce profil opérationnel est affiné au fur et à mesure que l'utilisateur juge les sources découvertes par notre système. Ainsi, DOWSER intègre un processus de retour de pertinence qui est un mécanisme rarement mis en oeuvre dans les systèmes de DI. Nous avons montré que l'approche de Rocchio [Rocchio, 1971] pouvait s'appliquer sur notre modélisation vectorielle d'instances de concept sans prise en compte de la hiérarchie de concepts associée. Un calibrage paramétrique nous a permis de définir le poids à donner aux retours de pertinence positifs et négatifs afin d'impacter de façon optimale le profil opérationnel. Ce retour de pertinence permet de pallier le problème de linéarité du processus de découverte. Le traitement linéaire "besoin - réponse" est remplacé par une boucle dans notre processus de DI permettant d'affiner la modélisation du besoin et ainsi améliorer le processus de découverte de sources d'intérêt.

Notre second axe d'étude dans cette thèse, la découverte de sources d'intérêt, apporte également un certain nombre de solutions aux limites de DI existantes dans le cadre du ROSO. Nous utilisons un robot d'exploration dont l'exploration est guidée afin de collecter en priorité des pages d'intérêt sur le Web. Cibler la collecte permet de faire face à l'important volume d'information disponible sur le Web et permet également de constituer un corpus composé uniquement de sources d'intérêt opérationnel. Nous avons introduit une mesure de similarité adaptative afin de guider notre robot d'exploration. Elle utilise la double représentation du besoin du profil opérationnel. Un calibrage paramétrique nous a permis de définir l'importance à donner à la couverture terminologique par rapport à la mesure thématique au sein de cette mesure. Le score obtenu entre une page collectée et le profil opérationnel correspond à la pertinence de la page par rapport au besoin de l'utilisateur. La contribution liée à cette mesure adaptative est de passer outre la popularité d'une page en se basant uniquement sur son contenu textuel. Ainsi, les pages pertinentes qui sont sensibles et/ou impopulaires, non indexées, ou *black listées* par les moteurs de recherche peuvent être découvertes par notre système et présentées à l'utilisateur. De plus, l'utilisateur peut consulter une page d'intérêt dès qu'elle est collectée, assurant ainsi de la fraîcheur de son contenu. Enfin, puisque l'exploration et la présentation des résultats exploitent le profil opérationnel, le système de découverte de sources d'intérêt est capable d'évoluer en même temps que le besoin opérationnel. Le système résultant permet de pallier les limites de la DI rencontré par les experts du renseignement.

Une autre contribution provient de l'architecture choisie pour le système DOWSER. En exploitant des outils open-source et la plate-forme WebLab, notre système peut fonctionner de manière autonome ou être facilement intégré dans des projets de plus grande envergure nécessitant une tâche de découverte de sources d'intérêt. De plus, l'utilisation de chaînes de traitements et de Web services offre la possibilité de moduler les processus appliqués aux pages collectées selon les besoins des projets.

9.1.2 Contributions opérationnelles

D'un point de vue opérationnel, les travaux menés durant cette thèse offrent un outil de veille stratégique aux experts du renseignement. La construction automatique du profil opérationnel, à partir de sources d'intérêt fournies par l'utilisateur, permet de modéliser le besoin informationnel implicitement. Les spécificités de langage et les mots d'argot relatifs au sujet de recherche, qui peuvent être méconnus de l'expert, sont pris en compte dans notre approche grâce à une extraction des mots-clés et des instances de concept d'intérêt présents dans les sources fournies. Notre processus de retour de pertinence permet de prendre en considération l'évolution de ce vocabulaire et les variations du besoin utilisateur. En effet, le besoin des experts du renseignement peut évoluer et se préciser au fur et à mesure qu'ils accroissent leur connaissance du sujet.

L'utilisation de concepts dans notre modélisation du besoin utilisateur fournit une contribution opérationnelle supplémentaire adaptée au ROSO. Il permet d'exploiter les connaissances du Web sémantique et d'améliorer la représentation du besoin au travers de la capitalisation des connaissances. L'utilisation de concepts DBPedia permet de couvrir un large éventail de thématiques grâce à l'importante taille des ressources disponibles dans cette ontologie. De plus, l'ontologie employée dans notre système peut être aisément remplacée afin de s'adapter aux spécificités des besoins. Par exemple, l'ontologie Wookie, qui représente le pentagramme du renseignement, pourrait améliorer la modélisation des besoins spécifiques des experts du renseignement.

L'utilisation conjointe de cette couverture thématique et de la couverture terminologique offre une couverture globale du besoin quelle que soit la sensibilité ou la spécificité de la demande en information. Ainsi, même si DOWSER est conçu pour répondre au besoin des experts du renseignement, cette contribution opérationnelle peut dépasser le cadre du ROSO et servir à des tâches de découverte de sources avec des besoins divers.

9.2 Limites

Les travaux menés durant cette thèse fournissent des contributions scientifiques et opérationnelles pour la découverte de sources d'intérêt. Cependant, ils comportent également des limites relatives à nos choix d'approche.

9.2.1 Limites de notre approche

Nos choix concernant notre approche se sont portés sur une double représentation du besoin opérationnel exploitée par un système d'exploration ciblée du Web.

Ce profil utilisateur à double représentation permet de modéliser le besoin thématique et terminologique de l'expert mais il ne permet pas de mettre en exergue toutes les facettes du besoin opérationnel. Ainsi, la notion de fraîcheur de l'information, qui consiste à découvrir des sources dont l'information est récente, n'est pas prise en compte dans notre profil. C'est une limite qui peut desservir notre approche si le besoin des experts du renseignement concerne uniquement des informations d'actualité. La langue et le multilinguisme sont également une facette qui peut caractériser les sources à découvrir. Or, elle n'est pas modélisée dans notre profil tout comme le type de la source (blog, forum, ...).

Notre approche de modélisation du besoin est donc limitée à une représentation du besoin informationnel sans considération de la langue souhaitée, du type de sources attendues, etc. Notre construction implicite du profil opérationnel à partir des sources de l'utilisateur peut également être une limite dans la mesure où l'expert n'a pas connaissance de la représentation de son besoin. Il est possible que des termes d'intérêt ne soient pas présents dans les sources fournies ou qu'ils ne soient pas extraits par les outils utilisés. Ainsi, notre représentation du besoin peut être erronée, rendant nos résultats d'exploration du Web non-pertinents. Cependant, le processus de retour de pertinence mis en place dans notre approche, tend à pallier cette limite en affinant la représentation du besoin.

Outre la modélisation du besoin, notre système de découverte de sources est également limité par les choix de notre approche. Nous considérons le Topical Locality Phenomenon pour guider notre exploration du Web. Il consiste à explorer en priorité les pages connexes à des pages d'intérêt. Or, il n'est pas démontré que ce phénomène s'applique à des sources sensibles et spécifiques comme celles recherchées par les experts du renseignement. De plus, les sources retournées par le système DOWSER sont limitées aux sources en accès direct. Autrement dit, les sources nécessitant la complétion d'un formulaire, les sources dont le contenu est généré dynamiquement ou les sources qui requièrent une authentification ne sont pas accessibles par notre système. Cependant, ces sources peuvent contenir de l'information pertinente et s'avérer être des sources d'intérêt opérationnel.

9.2.2 Limites de notre prototype

Une limite de notre système est d'utiliser des outils externes comme DBPedia Spotlight. En effet, lorsque nous testons la capacité de DOWSER à couvrir le besoin opérationnel, les résultats obtenus sont dépendants de la capacité de DBPedia Spotlight à extraire des concepts pertinents des pages Web. Il en est de même avec l'extraction de mots-clés et la méthode *NP Chunker* employée. Il faudrait alors répéter les expérimentations menées avec d'autres méthodes d'extraction afin de voir leurs impacts sur la représentation du besoin.

L'utilisation du retour de pertinence dans notre approche permet d'affiner la représentation du besoin. Cependant, ce processus est limité car il ne distingue pas l'importance donnée aux vecteurs conceptuels et aux vecteurs de mots-clés lors de l'exploitation des retours. En effet, ces deux vecteurs n'ont pas le même objectif : le premier représente la thématique et le second la terminologie des pages jugées positivement ou négativement par l'utilisateur. Aussi, il faudrait évaluer ce processus en exploitant de façon distincte les vecteurs de mots-clés et les vecteurs conceptuels issus des retours de pertinence.

Le calibrage paramétrique a été effectué pour déterminer la taille optimale des vecteurs du profil, pour identifier dans quelle proportion combiner la couverture terminologique et thématique au sein de la mesure adaptative et également pour définir l'importance donnée au retour de pertinence. Il est impossible de tester une infinité de valeurs pour ces paramètres, aussi nous avons effectué les tests par paliers de valeurs. C'est une limite de notre expérimentation à laquelle nous pourrions remédier en testant nos paramètres avec de nouvelles valeurs proches des valeurs optimales fournies par ce premier calibrage. Cela permettrait de définir avec plus de précision les valeurs optimales et d'améliorer les résultats du système DOWSER.
De plus, les paramètres testés sont fixés au début du calibrage paramètre. Nous n'avons pas

évalué les résultats de notre système en faisant varier ces paramètres au fil des itérations. Le retour de pertinence peut, par exemple, être nécessaire durant les premières itérations afin d'affiner le profil, puis s'avérer moins utile. De la même façon, nous n'avons pas fait varier, durant l'expérimentation, la proportion accordée à la couverture terminologique par rapport à la couverture thématique dans notre mesure adaptative. Comme le besoin opérationnel évolue au fur et à mesure que l'expert accroît ses connaissances sur le sujet de recherche, il est peut être nécessaire de faire évoluer et d'ajuster, en parallèle, la valeur de nos paramètres.

Malgré cela, la première expérimentation menée durant nos travaux a permis de valider notre approche de découverte de sources alors que la seconde a permis d'améliorer la modélisation du profil opérationnel et son contenu. Une troisième expérimentation permettrait de constater de l'impact de ce profil opérationnel optimisé sur la tâche de découverte de sources. En effet, outre la définition de la taille du profil et de la mesure de similarité adaptative, la seconde expérimentation a montré l'intérêt de l'utilisation du retour de pertinence qui n'était pas utilisé dans la première expérimentation. Ces différents paramètres optimisés devraient améliorer considérablement l'exploration ciblée du Web et la découverte de sources d'intérêt opérationnel.

Toujours en terme d'expérimentation, notre système a été testé hors du cadre ROSO et également dans le cadre du ROSO avec des experts du renseignement mais de manière très succincte. Nous n'avons pas pu mesurer l'intérêt porté au système ni mesurer ses résultats. Même si l'avis qualitatif qui en ressort est positif, il sera nécessaire de tester scrupuleusement DOWSER en conditions réelles d'utilisation. L'approche a été pensée de sorte à pouvoir découvrir des sources sensibles mais cela reste encore à prouver dans un contexte opérationnel de veille stratégique.

9.3 Perspectives et travaux futurs

Concernant les perspectives, il est nécessaire de considérer les limites énumérées ci-dessus pour améliorer globalement notre approche. De plus, il est important d'implémenter notre module d'extension de la zone de collecte pour le robot d'indexation. Ce module, qui doit permettre de découvrir des grappes d'intérêt sur le Web plus rapidement, devra être testé et son efficacité validée au sein du processus de collecte.

De plus, des travaux sur la perception et la fiabilité des sources à présenter à l'utilisateur peuvent être intégrés. Les travaux de [Mombrun, 2012], orientés autour de l'évaluation des informations obtenues sur Internet, peuvent permettre d'enrichir notre approche pour présenter des sources d'intérêt opérationnel dont la qualité intrinsèque, relationnelle et réputationnelle seraient prises en compte. Ces notions permettant d'améliorer notre représentation du besoin opérationnel peuvent être complétées par les informations concernant la langue et le type des sources recherchées, comme expliqué précédemment en page 173.

L'utilisation du Web sémantique, de concepts et d'une ontologie pour définir et modéliser le profil opérationnel est sous exploitée. Ils pourraient servir à une approche colla-

borative de découverte de sources où une distance sémantique entre profils opérationnels pourrait être calculée afin de trouver les intérêts en commun. Le score de similarité sémantique pourrait être également optimisé en s'intéressant à davantage de relations entre les concepts en plus des catégories qu'ils partagent. Les relations de similarité (*sameAs*) ou de redirection (*wikiPageRedirects*) de l'ontologie DBPedia pourraient notamment être exploitées. De plus, l'approche sémantique peut aussi être améliorée en exploitant des ontologies de domaine plus spécifique au besoin opérationnel. Durant les expérimentations, l'ontologie DBPedia a été utilisée. Cependant, exploiter des ontologies plus adéquates aux besoins des experts du renseignement pourrait améliorer les résultats de notre système : l'ontologie Wookie [Serrano et al., 2012], par exemple, qui est basée sur le pentagramme du renseignement, pourrait apporter une modélisation du besoin plus représentative du domaine de recherche. Ce sont des modifications qui, une fois mises en place, nécessiteront d'être testées au travers d'expérimentations.

Cinquième partie

Annexes

ANNEXE A

CAS D'UTILISATION

Cette annexe illustre un cas d'utilisation fictif de DOWSER basé sur le second scénario présenté section 5.1, page 72. Il s'agit de découvrir des sources Web autour de la vente et de la fabrication de produits médicamenteux à base de cornes d'antilope. Le profil construit dans cet exemple est constitué uniquement de 5 instances de concept et de 10 mots-clés afin de rendre facilement compréhensible ce cas d'utilisation. Aucun jugement n'est porté sur la légalité des produits vendus par les sites utilisés pour illustrer notre cas d'utilisation fictif.

A.1 Construction du profil utilisateur

Les sources d'intérêt fournies par l'expert sont des sites vendant des produits et des herbes de la médecine traditionnelle chinoise. On y trouve également deux sites ventant les bienfaits de l'utilisation de la corne d'antilope en tant que médicament :

- `http://www.acupuncturetoday.com/herbcentral/antelopes_horn.php`
- `http://health.howstuffworks.com/wellness/natural-medicine/chinese/traditional-chinese-medicine.htm`
- `http://www.drshen.com/herbstore.htm`
- `http://www.herbsbuy360.com/`
- `http://www.chineseherb.com/`

Le profil opérationnel, construit par le système DOWSER (voir section 5.2.2, page 75), est illustré dans la figure A.1. La thématique générale est bien représentée par les instances de concept : les thèmes de la médecine chinoise, des herbes médicales et de la santé y figurent. Les mots-clés couvrent également le besoin en le précisant davantage avec l'intérêt porté sur la corne d'antilope (*antelope horn*, *yang jiao*) et sur l'achat de médicaments (*health product, medecine shop*).

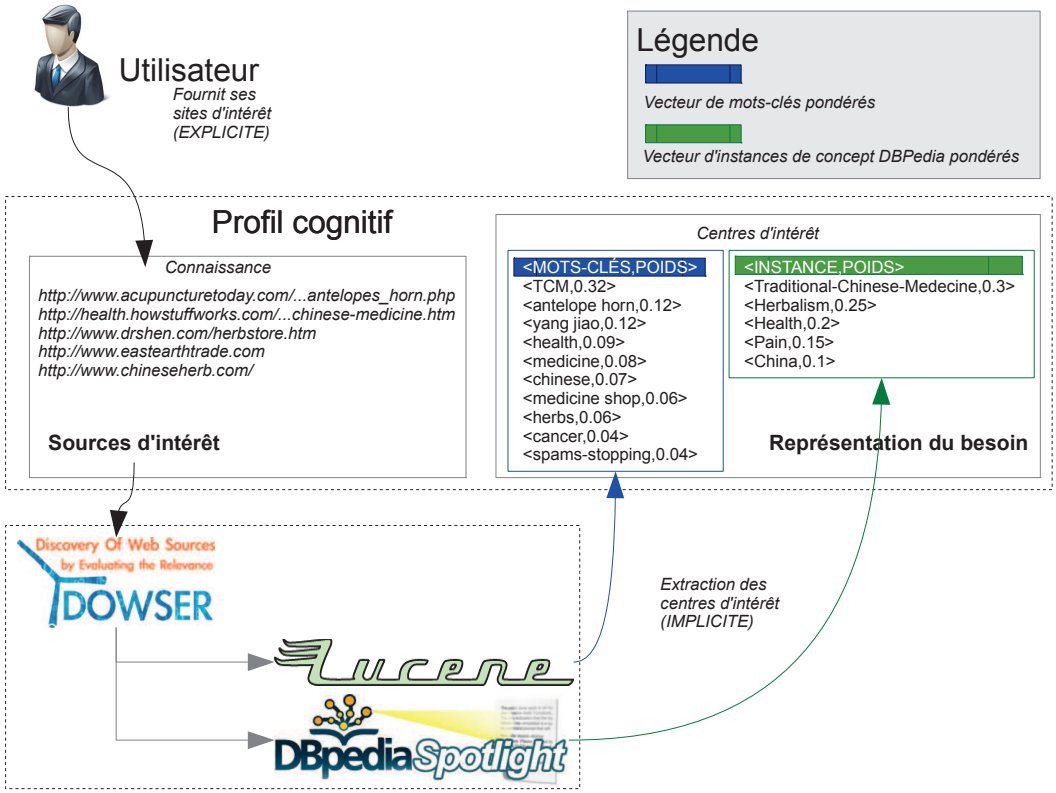


FIGURE A.1 – Cas d'utilisation : construction du profil dans DOWSER

A.2 Découverte de nouvelles sources

En accord avec le processus de découverte de sources de DOWSER (voir section 6, page 97), les 5 URLs fournies par l'utilisateur vont servir d'URLs d'amorçage pour l'exploration ciblée du Web. Afin d'illustrer ce processus au travers d'un exemple, on considère que la page `http://www.herbsbuy360.com/` a été collectée, analysée et que les liens présents dans cette page ont été empilés dans la liste d'URLs à explorer. Comme l'illustre l'impression d'écran A.2, page 181, cette page contient des liens vers les différentes catégories du site ainsi que vers des produits mais aussi des liens vers des sites partenaires. Parmi les liens extraits de cette page, on s'intéresse à deux liens pointant vers des pages internes et deux liens pointant vers des sources différentes :

- Lien interne : `http://www.herbsbuy360.com/boilfree-herbs-liverpacifying-and-windextinguishing-medicinal-c-380_432.html`
- Lien interne : `http://www.herbsbuy360.com/chinese-tea-c-382.html`
- Lien externe : `http://www.globalchineseherbhealing.com/index.php`
- Lien externe : `http://www.canceranti.com/index.html`

Ces quatre liens sont illustrés dans la figure A.3, page 182. Puisqu'ils proviennent de la même page, ils ont tous les quatre une priorité équivalente. Cependant, ce sont les pages externes qui sont explorées en premier puisque le système DOWSER a pour but de découvrir de nouvelles sources. Dans cet exemple, le traitement de la page `http://www.globalchineseherbhealing.com/index.php` est détaillé. Puis, plus succinctement, le

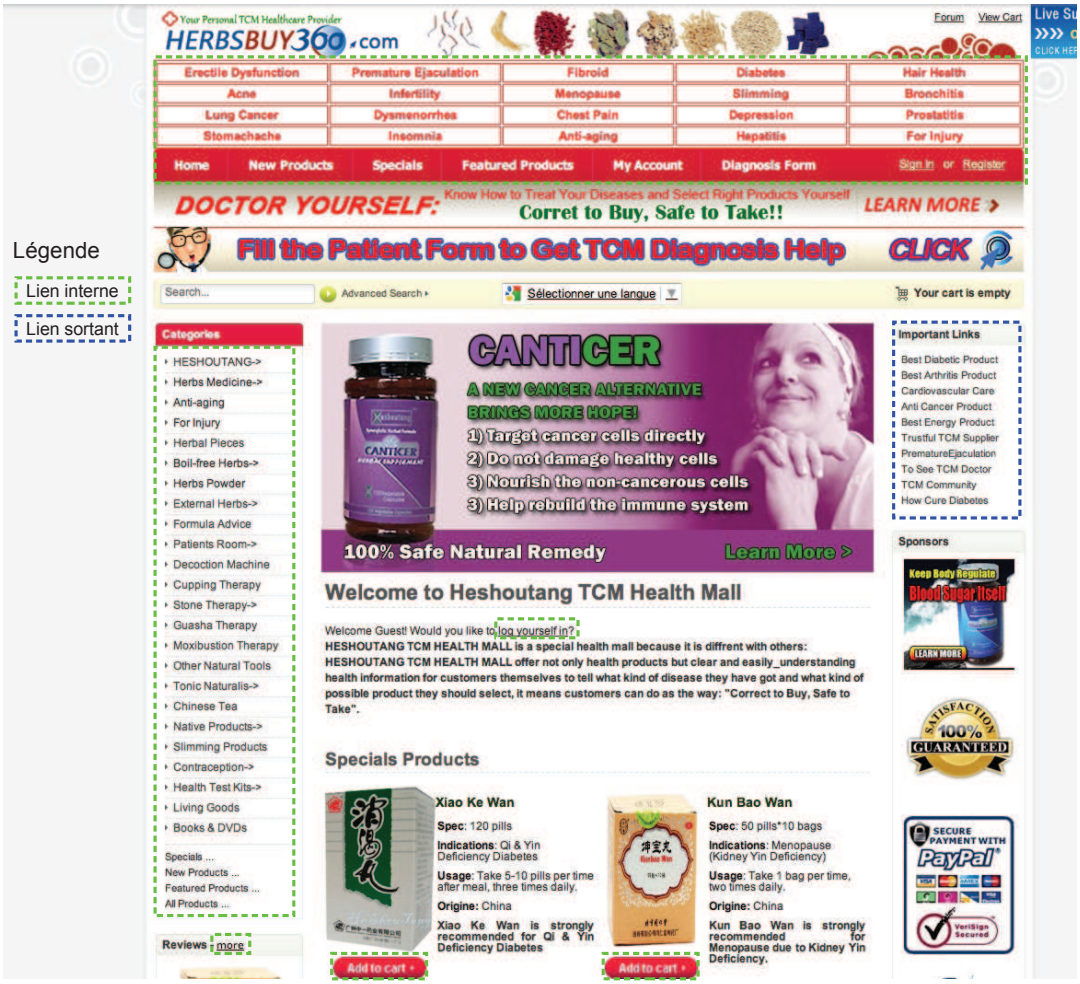


FIGURE A.2 – Cas d'utilisation : extraction des liens d'une page dans DOWSER

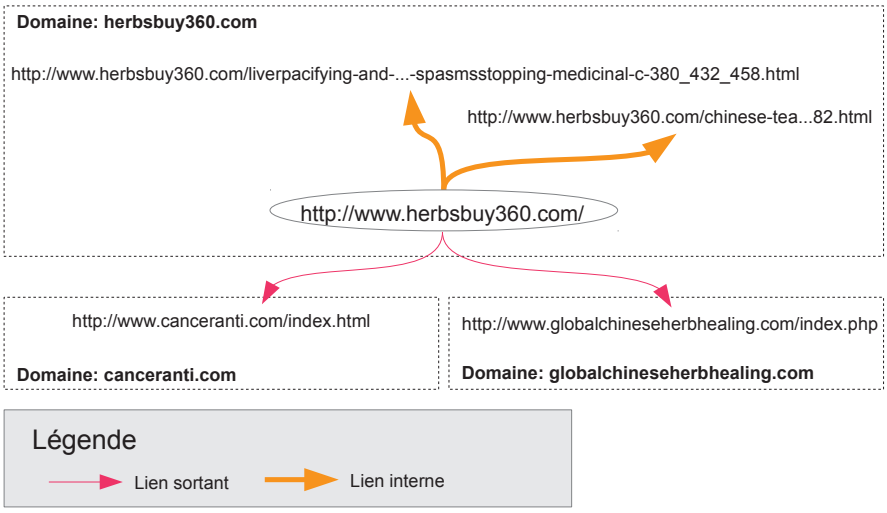


FIGURE A.3 – Cas d'utilisation : liens internes et externes extraits d'une page

résultat du traitement des pages pointées par les 3 autres liens est présenté afin d'illustrer les choix d'exploration ciblée opérés par le système DOWSER.

A.2.1 Analyse d'une page collectée

Les pages collectées sont stockées temporairement afin d'être analysées par la chaîne de traitement de DOWSER (voir section 8.3.2, page 158). Le premier service appelé dans cette chaîne de traitement est le service Tika.

a. Service Tika Normaliser

Le service Tika a pour rôle de créer, à partir du document traité, une ressource WebLab et d'y ajouter les premières annotations.

```
<resource xsi:type="wl:Document" uri="weblab://warcCreator/7006062253767"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xmlns:wl="http://weblab.ow2.org/core/1.2/model#">
        <!--Information sur la collecte:-->
            <annotation uri="weblab://warcCreator/7006062253767#a0">
               <data xmlns:wl="http://weblab.ow2.org/core/1.2/model#">
                  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
                  xmlns:wlp="http://weblab.ow2.org/core/1.2/ontology/processing#">
                     <rdf:Description rdf:about="weblab://warcCreator/7006062253767"
                     xmlns:dc="http://purl.org/dc/elements/1.1/"
                     xmlns:dcterms="http://purl.org/dc/terms/"
                     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
                        <wlp:hasNativeContent
                              rdf:resource="http://localhost:8080/api/secure/webdav/guest/document_library/
WebLab/963c91b0-ca27-4560-9b9d-0894277b5611"/>
                        <dcterms:creator>12068_TCM</dcterms:creator>
                        <wlp:hasGatheringDate>2013-08-28T10:01:17+0200</wlp:hasGatheringDate>
                        <dc:source>http://www.globalchineseherbhealing.com/index.php</dc:source>
                        <dcterms:source>http://www.herbsbuy360.com/</dcterms:source>
                        <dc:format>text/html;</dc:format>
```



```
                    <wlp:hasNormalisedContent
                        rdf:resource="http://localhost:8080/api/secure/webdav/guest/document_library/
WebLab/b9ee67c6-65a4-4725-89fc-b3942a5359c5"/>
                </rdf:Description>
            </rdf:RDF>
        </data>
    </annotation>
    <!--Métadonnées sur la page collectée:-->
    <annotation uri="weblab://warcCreator/7006062253767#a2">
        <data xmlns:wl="http://weblab.ow2.org/core/1.2/model#">
            <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
                xmlns:dc="http://purl.org/dc/elements/1.1/"
                xmlns:dct="http://purl.org/dc/terms/"
                xmlns:wlp="http://weblab.ow2.org/core/1.2/ontology/processing#"
                xmlns:wlr="http://weblab.ow2.org/core/1.2/ontology/retrieval#">
                <rdf:Description rdf:about="weblab://warcCreator/7006062253767#a2">
                    <dct:created>2013-08-28T10:04:31.924+02:00</dct:created>
                    <wlp:isProducedBy
                        rdf:resource="http://weblab.ow2.org/service/normaliser/tika"/>
                </rdf:Description>
                <rdf:Description rdf:about="weblab://warcCreator/7006062253767">
                    <dc:format>text/html; charset=UTF-8</dc:format>
                    <dc:description>Welcome to Global Chinese Herb Healing! Traditional
                    Chinese herbal medicine has been used for over 5,000 years in China
                    and East-Asian countries to treat ...</dc:description>
                    <dc:title>The Best Chinese Herb Store Online!</dc:title>
                    <dc:creator>globalchineseherbhealing.com</dc:creator>
                </rdf:Description>
            </rdf:RDF>
        </data>
    </annotation>
    <!--Texte extrait:-->
    <mediaUnit xsi:type="wl:Text" uri="weblab://warcCreator/7006062253767#1">
        <content>
Forum
View Cart
Erectile Dysfunction
Premature Ejaculation
Fibroid
Diabetes
Hair Health
Acne
Infertility
Menopause
Slimming
Bronchitis
Lung Cancer
Dysmenorrhea
Chest Pain
Depression
Prostatitis
Stomachache
Insomnia
Anti-aging
Hepatitis
For Injury
Home
New Products
Specials
Featured Products
My Account
Diagnosis Form
Sign In or Register
Advanced Search
Your cart is empty
Categories
HESHOUTANG
```

```
Herbs Medicine
Anti-aging
For Injury
Herbal Pieces
Boil-free Herbs
   Superficies-releasing Medicinal
   Heat-Clearing Medicinal
   Precipitating Medicinal
   Removing Wind-damp Medicinal
   Damp-resolving Medicinal
   Damp-draining Diuretic
   Interior-warming Medicinal
   Qi-regulating Medicinal
   Digestant Medicinal
   Anthelmintics
   Hemostatic Medicinal
   Blood-activating and Stasis-resolving Medicinal
   Phlegm-resolving and Cough and Dyspnea-relieving Medicinal
   Tranquilizing Medicinal
   Liver-pacifying and Wind-extinguishing Medicinal
     Liver-pacifying and Yang-subduing Medicinal
     Wind-extinguishing and Spasms-stopping Medicinal
   Resuscitative Medicinal
   Tonics
   Astringent Medicinal
   Dispelling Wind and Moistening Dryness
Herbs Powder
External Herbs
Formula Advice
Patients Room
Decoction Machine
Cupping Therapy
Stone Therapy
Guasha Therapy
Moxibustion Therapy
Other Natural Tools
Tonic Naturalis
Chinese Tea
Native Products
Slimming Products
Contraception
Health Test Kits
Living Goods
Books DVDs
Specials ...
New Products ...
Featured Products ...
All Products ...
Reviews - more
Kun Bao Wan is a really good product for Menopause
Hua Tuo Zai Zao Wan--excellent product for Meridian Stroke
My wife has stomach problem for about 3 months. I looked at...
Very good product for heart problemmmmmm
I baught 6 boxes, it works well thanks!
Featured - more
GlucoNature--Chinese Traditional Supplement for Diabetes
$119.99
UPOWER--Pure Herbal Energetic Product for Anti Fatigue
$179.99
Specials - more
Bian Stone Knee Pad for Joints Pain
$136.99 $109.99
Save: 20% off
Yang Xue Sheng Fa Capsule-For Hair Loss(Blood Deficiency)
$9.99 $8.99
Save: 10% off
Information
```

About Us
Shipping Returns
Privacy Notice
Conditions of Use
Contact Us
Site Map
Gift Certificate FAQ
Discount Coupons
Newsletter Unsubscribe
Home :: Boil-free Herbs :: Liver-pacifying and Wind-extinguishing Medicinal
Liver-pacifying and Wind-extinguishing Medicinal

Liver-pacifying and Yang-subduing Medicinal
Wind-extinguishing and Spasms-stopping Medicinal
New Products For June - Liver-pacifying and Wind-extinguishing Medicinal
Pearl Shell/ Concha Margaritifera/ Zhen zhu mu
$0.63
Oyster Shell/ Concha Ostrea/ Mu li
$0.57
Hematite/ Haematitum/ Dai zhe shi/ Zhe shi
$0.82
Prepared Oyster Shell/ Prepared Concha Ostrea/ Duan mu li
$0.58
Prepared Sea-ear Shell/ Prepared Concha Haliotidis/ Duan shi jue
$0.73
Antelope Horn/ Cornu Saigae Tataricae/ ling yang jiao
$22.63
Gallstone of Cows/ Calculus Bovis/ Niu huang
$1.68
Pearl/ Margarita/ Zhen zhu
$0.78
Gambir Plant/ Ramulus Uncariae cum Uncis/ Gou teng
$0.63
Tall Gastrodia Tuber/ Rhizoma Gastrodiae Elatae/ Tian ma
$1.42
Earthworm/ Lumbricus/ Di long
$1.87
Scorpion/ Scorpio/ Quan xie
$4.32
Centipede/ Wu gong
$4.18
Dead Body of Sick Silkworm/ Herba Patriniae/ Bai jiang can/ Jian
$1.72
Dogbane Leaf/ Folium Apocyni veneti/ Luo bu ma
$0.98
Important Links
Best Diabetic Product
Best Arthritis Product
Cardiovascular Care
Anti Cancer Product
Best Energy Product
Trustful TCM Supplier
Premature Ejaculation
To See TCM Doctor
TCM Community
How Cure Diabetes
Sponsors
Bestsellers
Centipede Wu gong
Whos Online
There currently are 13 guests online.
Quick Links
Home
Featured Products
Specials
New Products
All Products ...


```
Information
About Us
Site Map
Gift Certificate FAQ
Discount Coupons
Newsletter Unsubscribe
Customer Service
Contact Us
Shipping Returns
Privacy Notice
Conditions of Use
My Account
Important Links
Best Diabetic Product
Best Arthritis Product
Cardiovascular Care
Anti Cancer Product
Best Energy Product
Copyright © 2014 HESHOUTANG TCM HEALTHCARE CO .
            </content>
          </mediaUnit>
</resource>
```

La ressource WebLab créée contient des informations sur la collecte comme l'identifiant de l'utilisateur (*12068*) et le sujet du projet (*TCM* pour Traditional Chinese Medicine), l'URL de la page analysée ainsi que la date de l'analyse, et l'URL qui a permis de trouver cette page (`http://www.herbsbuy360.com/`). Le service Tika ajoute une seconde annotation contenant les métadonnées de la page. Elle contient le titre, le résumé, le nom de domaine ainsi que la date à laquelle la page a été collectée. Enfin, la ressource WebLab est créée avec une *Media Unit* de type *Text* contenant le texte extrait de la page. Dans cet exemple, celui-ci ressemble à une liste de mots puisque le site se compose uniquement de catégories et de produits, sans paragraphe ni longue description textuelle. Ce texte est ensuite analysé par le service DBPedia Spotlight.

b. Service DBPedia Spotlight Annotator

Ce service ajoute à la ressource des annotations contenant les instances DBPedia identifiées dans le contenu de la page :

```
<annotation uri="weblab://warcCreator/7006062253767#a4">
  <data>
    <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dct="http://purl.org/dc/terms/"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:wlp="http://weblab.ow2.org/core/1.2/ontology/processing#"
        xmlns:wlr="http://weblab.ow2.org/core/1.2/ontology/retrieval#">
      <rdf:Description rdf:about="weblab://dowser/spotlightSegment">
        <wlp:refersTo rdf:resource="weblab:spotlightdbpediaInstance/#7558800756973807"/>
      </rdf:Description>
      <!--Informations sur l'instance identifiée:-->
      <rdf:Description rdf:about="weblab:spotlightdbpediaInstance/#7558800756973807">
        <rdfs:isDefinedBy>http://dbpedia.org/resource/Traditional_Chinese_medicine</rdfs:isDefinedBy>
        <rdfs:label>Traditional Chinese medicine</rdfs:label>
        <rdf:type>http://dbpedia.org/resource</rdf:type>
      </rdf:Description>
    </rdf:RDF>
  </data>
</annotation>
```


Cette annotation correspond à l'instance *Traditional Chinese medicine* (*rdf :label*) qui est une ressource DBPedia (*rdf :type*) définie sur la page `http://dbpedia.org/resource/Traditional_Chinese_medicine`. Un *Linear Segment* est ajouté à la *Media Unit Text* afin de lier cette annotation au(x) terme(s) du texte qui a(ont) permis d'identifier l'instance de concept :

```
<segment xsi:type="ns15:LinearSegment" start="125" end="125"
    uri="weblab://warcCreator/7006062253767#1-0"/>
<segment xsi:type="ns15:LinearSegment" start="125" end="125"
    uri="weblab://dowser/spotlightSegment"/>
```

Le service ajoute autant de *Linear Segment* qu'il y a de termes ayant permis d'identifier cette ressource. La liste suivante contient une partie des instances identifiées dans cette page :

- Chinese herbal medicine
- Acne vulgaris
- Pain
- Herbal_tonicsm
- Infertility
- Menopause
- Bronchitis
- Dysmenorrhea
- Prostatitis
- Insomnia
- Life_extension
- Premature_ejaculation
- Erectile_dysfunction
- Herbalism

c. Service Gate Annotator

Le service Gate est ensuite utilisé pour extraire les mots-clés du texte. De la même façon que le service précédent, il ajoute à la ressource WebLab des annotations et des *Linear Segment* pour définir ces mots-clés :

```
<rdf:Description rdf:about="weblab:gateInstance/NounChunk#7635121935305842">
  <rdfs:label>Spasms-stopping</rdfs:label>
  <rdf:type rdf:resource="http://gate.ac.uk/gatemodel#NounChunk"/>
</rdf:Description>
```

Parmi les mots-clés extraits de cette page, on peut relever les suivants :

- TCM
- products
- spasms-stopping
- wind-extinguishing
- Liver-pacifying
- health
- herbs
- hematite
- tall gastrodia tuber

- antelope horn
- earthworm
- scorpion

d. Service Score Annotator

Avant d'être indexées dans notre base de connaissances, ces informations sont utilisées afin de calculer la pertinence du document grâce au service Score Annotator. Ce dernier enrichit la ressource avec une nouvelle annotation contenant trois informations. Les deux premières correspondent respectivement au score de similarité thématique et au score de similarité terminologique. La dernière information ajoutée concerne le score de similarité adaptatif :

```
<annotation uri="weblab://warcCreator/7006062253767#a13">
  <data>
      <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:discovery="http://org.ow2.weblab/ontology/discovery#">
          <rdf:Description rdf:about="weblab:scoreService/#20564273359">
              <discovery:hasTerminologicalScore
                  rdf:datatype="http://www.w3.org/2001/XMLSchema#double">
                   0.793
              </discovery:hasTerminologicalScore>
              <discovery:hasThematicScore
                  rdf:datatype="http://www.w3.org/2001/XMLSchema#double">
                   0.843
               </discovery:hasThematicScore>
              <discovery:hasScore rdf:datatype="http://www.w3.org/2001/XMLSchema#double">
                   0.823
               </discovery:hasScore>
          </rdf:Description>
       </rdf:RDF>
  </data>
</annotation>
```

L'utilisation des catégories dans notre mesure de similarité thématique est explicitée en exploitant cet exemple. La figure A.4, page 189, illustre la comparaison du vecteur d'instances DBPedia du profil avec celui extrait de la page collectée. Dans DBPedia, l'instance *Traditional_Chinese_Medicine* a une relation d'équivalence (relation *owl :sameAs*[1]) avec l'instance *Chinese_herbal_medicine*. Ces relations DBPedia ne sont pas utilisées dans notre mesure. Cependant, cette information est tout de même prise en compte puisqu'en partageant les mêmes catégories, ces deux instances sont jugées comme étant similaire par notre mesure. Lorsque deux instances, comme les instances *Herbal_tonicsm* et *Herbalism*, ne partagent qu'une partie de leurs catégories, alors la similarité est partielle. Enfin, quand une instance ne partage aucune catégorie avec les autres instances, sa similarité est nulle.

A.2.2 Choix d'orientation de la collecte

Nous considérons que les 3 autres pages de notre exemple ont été collectées et analysées par la chaîne de traitement. Les vecteurs extraits de ces pages sont présentés dans la figure A.5, page 191, ainsi que les scores de similarité obtenus.

Les pages `http://www.herbsbuy360.com/boilfree-herbs-liverpacifying-and-windextinguishing-medicinal-c-380_432.html` et `http://www.herbsbuy360.com/chin`

1. http://www.w3.org/2002/07/owl#sameAs

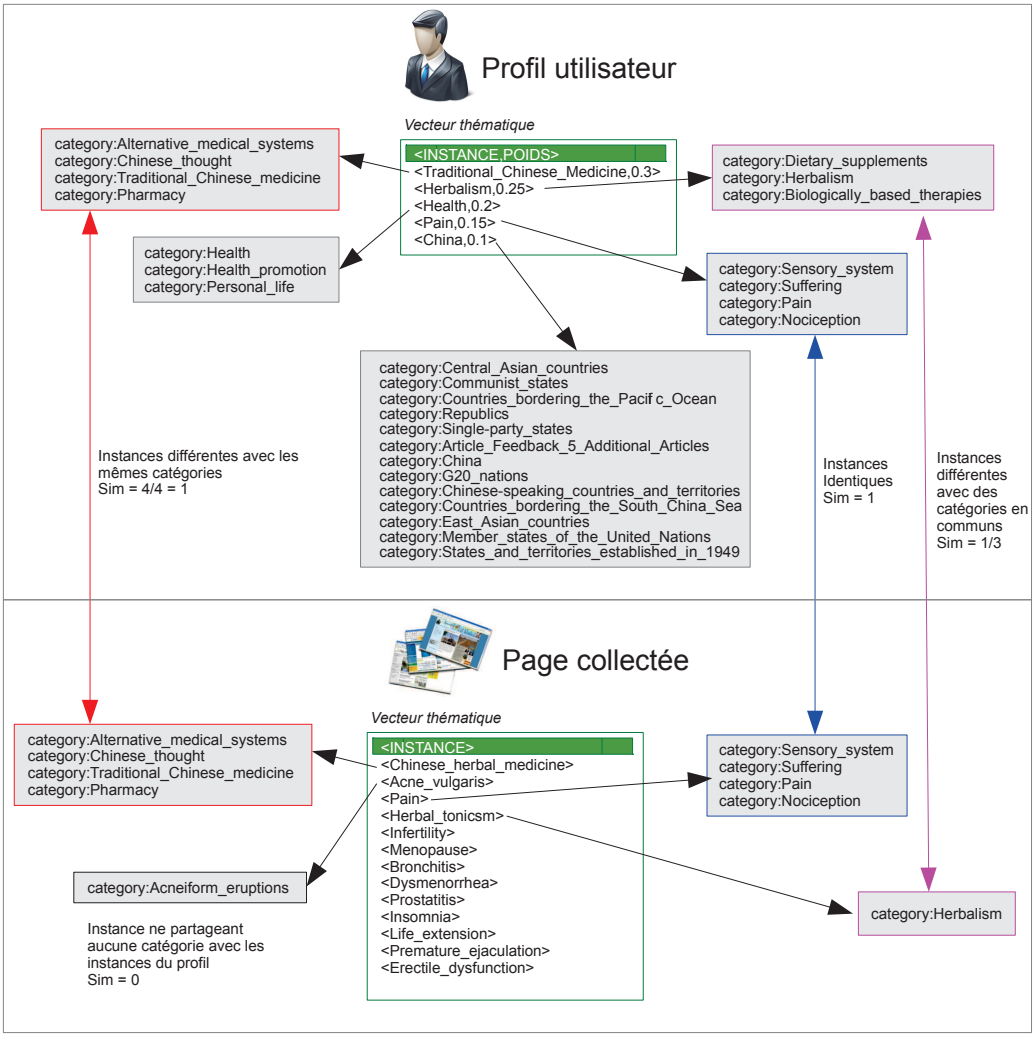


FIGURE A.4 – Cas d'utilisation : exploitation des catégories par la mesure de similarité thématique dans DOWSER

`ese-tea-c-382.html` sont des pages internes du site parent. La première d'entre elles obtient un score de similarité supérieur à la seconde puisqu'elle contient plus de mots-clés et d'instances de concept en commun avec le profil utilisateur. En analysant manuellement ces deux pages, on peut confirmer ce score donnant l'avantage à la première page : cette dernière présente une liste de produits et d'herbes médicales dont l'un à base de corne d'antilope. C'est exactement le genre de produit recherché dans ce scénario. Les liens extraits de cette page seront donc explorés avant ceux de la seconde page qui propose la vente de thé. Ceci permettra ainsi de trouver la page contenant la fiche du produit recherché (`http://www.herbsbuy360.com/http://www.herbsbuy360.com/antelope-horn-cornu-saigae-tataricae-ling-yang-jiao-p-807.html`).

La page `http://www.globalchineseherbhealing.com/index.php` obtient également un bon score de similarité. Cette page correspond à un site de vente de produits mé-

dicamenteux chinois partenaire de notre page parent. L'exploration de ces pages se fera dans un second temps et permettra également de découvrir une page d'intérêt. En effet, la page `http://www.globalchineseherbhealing.com/index.php/ling-yang-jiao-antelope-horn-50g.html` permet d'acheter un produit à base de corne d'antilope.

Enfin, la dernière page analysée (`http://www.canceranti.com/index.html`) obtient le moins bon score de similarité. Cette page reste thématiquement proche du besoin en proposant des produits médicamenteux chinois mais elle est cependant spécialisée dans la lutte contre le cancer, comme l'illustre les vecteurs représentatifs de son contenu. Les liens qu'elle contient seront donc visités en dernier.

Les liens extraits de ces 4 pages sont donc ordonnés en fonction de la pertinence de leur page parent. Le chiffre en rouge dans la figure A.5, page 191, illustre à la fois le rang de pertinence de la page parent collectée mais aussi le rang des URLs qu'elle contient dans la pile d'URLs à explorer. Cette pile évolue dès qu'une nouvelle page est collectée et analysée : les liens nouvellement extraits peuvent s'intercaler avec les liens déjà présents dans la pile en fonction de la pertinence de leur page parent. Par exemple, le système DOWSER collectera la page `http://www.herbsbuy360.com/antelope-horn-cornu-saigae-tataricae-ling-yang-jiao-p-807.html`, vendant de la corne d'antilope, avant de visiter les liens présents sur le site `http://www.canceranti.com/index.html` car le score de pertinence de cette dernière est moins élevé que celui de la page parent `http://www.herbsbuy360.com/boilfree-herbs-liverpacifying-and-windextinguishing-medicinal-c-380_432.html`. C'est de cette façon que DOWSER collecte en priorité les pages jugées d'intérêt.

A.3 Retour de pertinence

Afin d'illustrer le retour de pertinence dans ce cas d'utilisation du système DOWSER, nous définissons deux vecteurs pondérés constitués de mots-clés. Ils sont construits à partir de retours de pertinence positifs et négatifs et permettent de mettre à jour le profil opérationnel en suivant le processus décrit section 5.3, page 81. La première étape, illustrée dans la figure A.6, page 192, consiste à construire un seul vecteur K_{i+1} à partir de ces deux vecteurs de retour de pertinence K^{P} et K^{NP} et du vecteur mots-clés du profil opérationnel K_i. Le poids de chaque terme est redéfini en fonction du ratio accordé au vecteur d'origine (celui du profil), et aux ratios de retours de pertinence positifs et négatifs. Dans cet exemple, la formule de retour de pertinence utilisée est la suivante :

$$\vec{P}_K^{t+1} = 0,8 \times \vec{P}_K^{t} + 0,1 \times \vec{K}^{P,t} - 0,1 \times \vec{K}^{NP,t}$$

Une fois ce vecteur créé, la seconde étape a pour but de regrouper et d'ordonner les termes de ce vecteur. Par exemple, le terme *TCM*, présent dans le profil utilisateur mais aussi dans le vecteur de retour positif et dans le vecteur de retour négatif, voit son poids W_{TCM} évoluer :

$$W_{TCM} = 0,8 \times 0,32 + 0,1 \times 0,32 - 0,1 \times 0,28$$

$$W_{TCM} = 0,256 + 0,032 - 0,028 = 0,26$$

Enfin, la dernière étape consiste à normaliser le profil en fonction de sa taille. Dans notre exemple, la taille du profil est fixée à 10. Ainsi, les 10 termes de poids les plus forts sont gardés et leur poids est normalisé. Ces retours de pertinence permettent d'augmenter

FIGURE A.5 – Cas d'utilisation : exploitation des scores de pertinence pour guider la collecte

notamment le poids des termes d'intérêt *antelope horn* et *yang jiao* et de baisser le terme *cancer* qui n'est pas un terme d'intérêt au vue du scénario de recherche. La représentation du besoin est ainsi affinée.

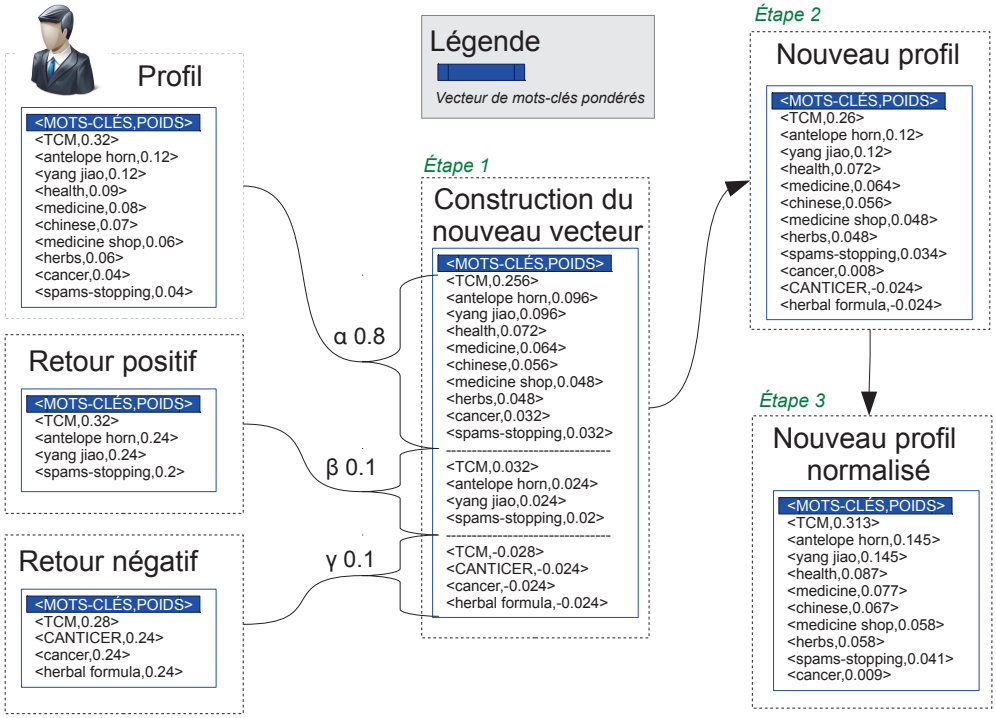


FIGURE A.6 – Cas d'utilisation : exploitation du retour de pertinence sur le vecteur terminologique

ANNEXE B

RÉSULTATS COMPLÉMENTAIRES

Cette annexe présente des résultats complémentaires du calibrage expérimental des paramètres (voir 7.2, page 133). Les figures B.1 et B.2 illustrent l'évolution de la précision du système DOWSER lorsque la taille du vecteur thématique ou du vecteur terminologique évolue.

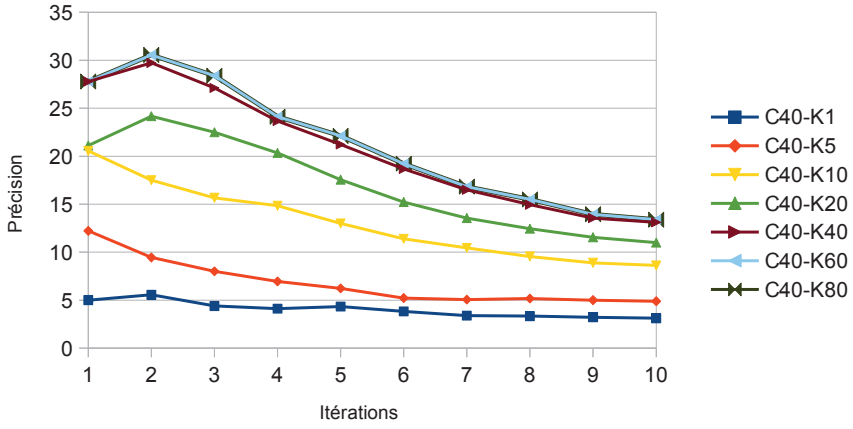


FIGURE B.1 – Évolution de la précision en fonction de la taille du vecteur mots clés

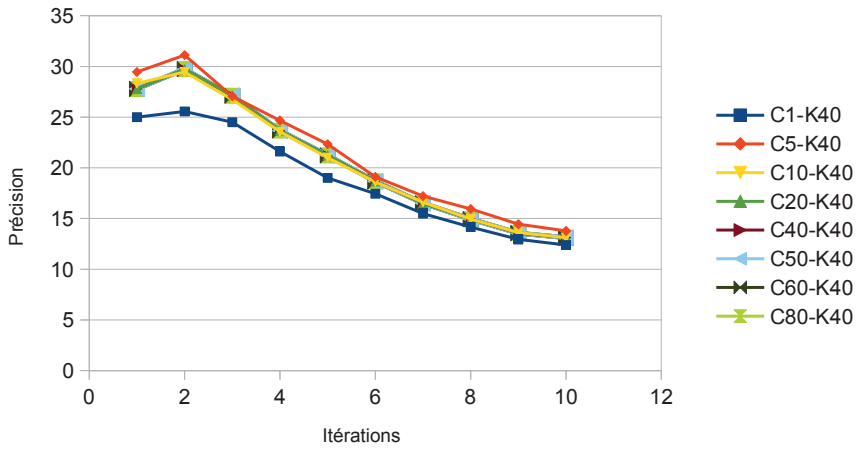


FIGURE B.2 – Évolution de la précision en fonction de la taille du vecteur concepts

Ces graphiques permettent de mettre en exergue le couple optimal C5-K60. Le rappel et la précision obtenus dans cette configuration sont illustrés dans les figures B.3 et B.4, page 194 et 195.

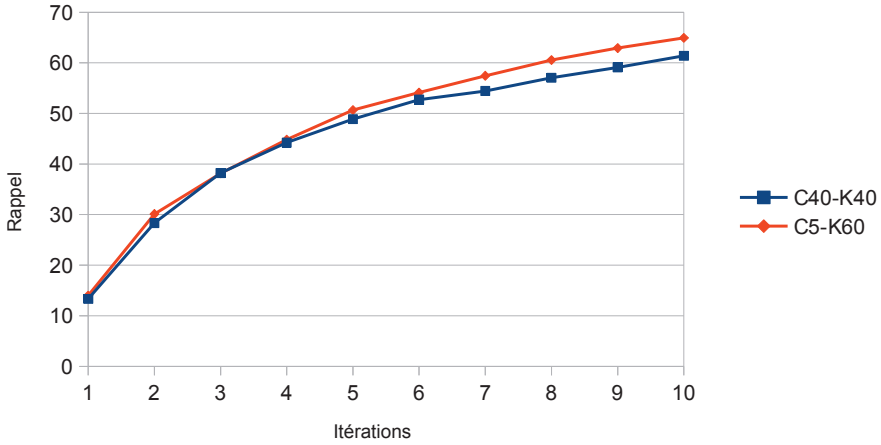


FIGURE B.3 – Amélioration du rappel avec la taille du profil optimisée

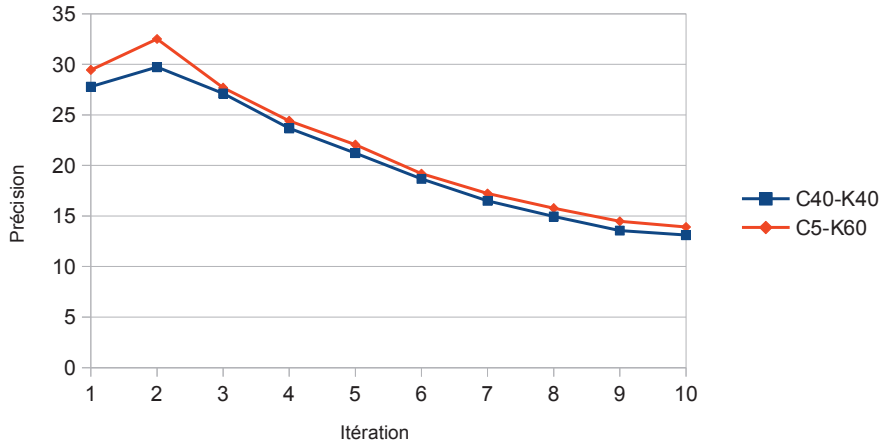


FIGURE B.4 – Amélioration de la précision avec la taille du profil optimisée

La figure B.5, page 195, et la figure B.6, page 196, illustrent respectivement l'évolution du rappel et de la précision en fonction de la valeur du δ dans notre mesure de similarité adaptative (voir section 6.3.2, page 106).

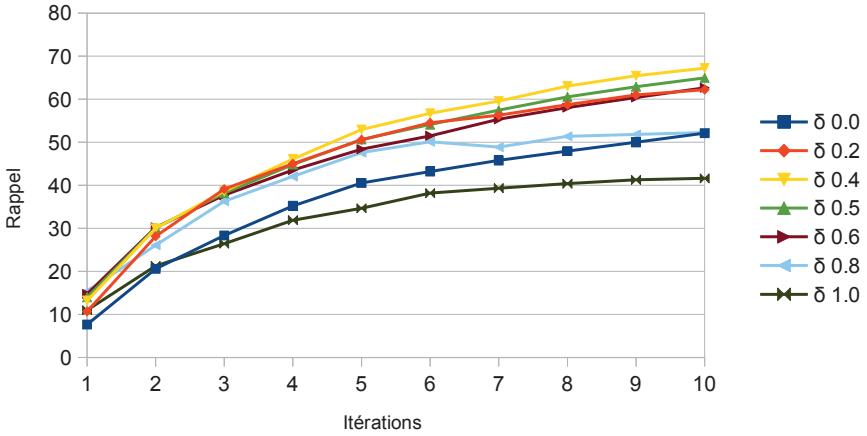


FIGURE B.5 – Évolution du rappel en fonction de δ

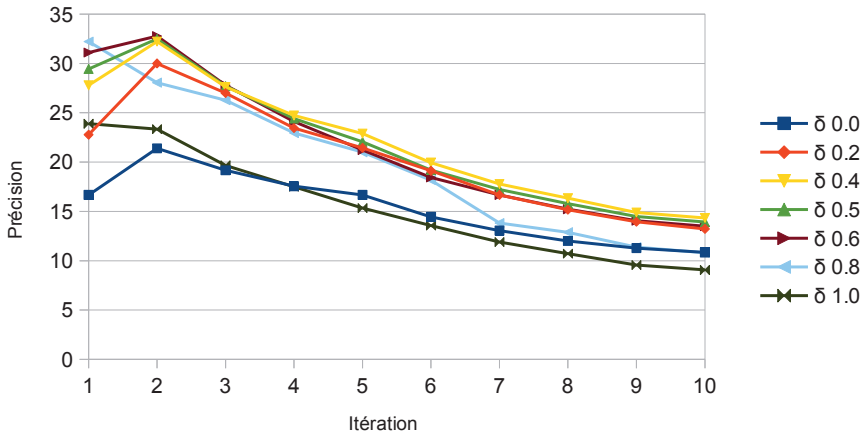


FIGURE B.6 – Évolution de la précision en fonction de δ

Les figures B.7 et B.8 correspondent à des résultats complémentaires de la F-Mesure lorsque le δ varie avec une taille différente pour les vecteurs du profil opérationnel.

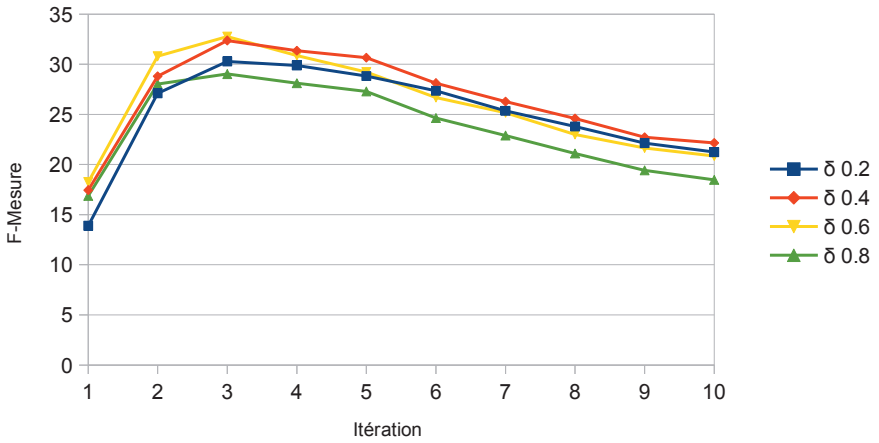


FIGURE B.7 – Évolution de la F-Mesure en fonction de δ sur un profil C50-K50

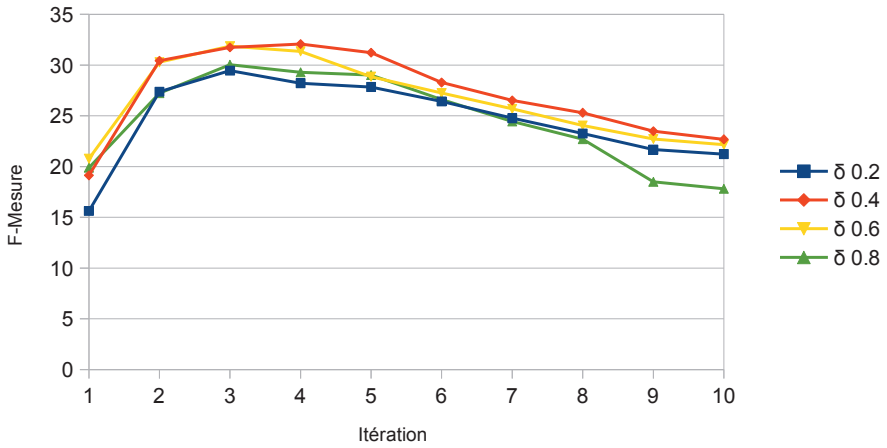


FIGURE B.8 – Évolution de la F-Mesure en fonction de δ sur un profil C5-K40

Les comparatifs du rappel et de la précision en fonction du nombre de pages utilisées pour le retour de pertinence sont illustrés dans les figures B.9 et B.10.

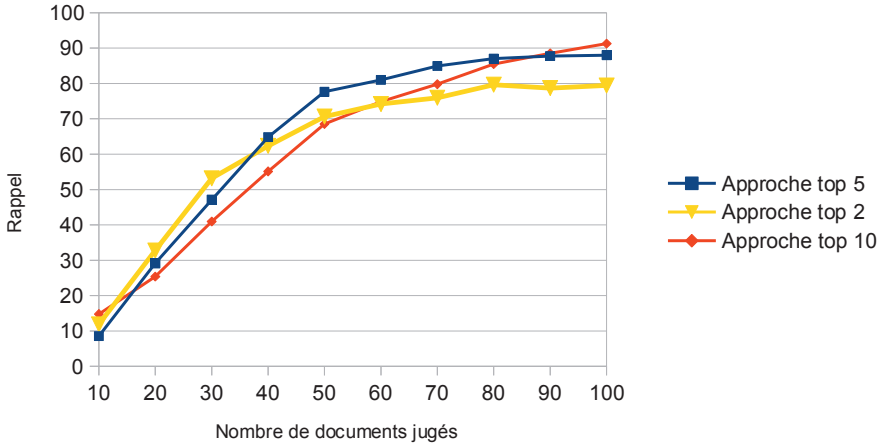


FIGURE B.9 – Comparaison du rappel en exploitant 2, 5 et 10 pages jugées dans le processus de retour de pertinence

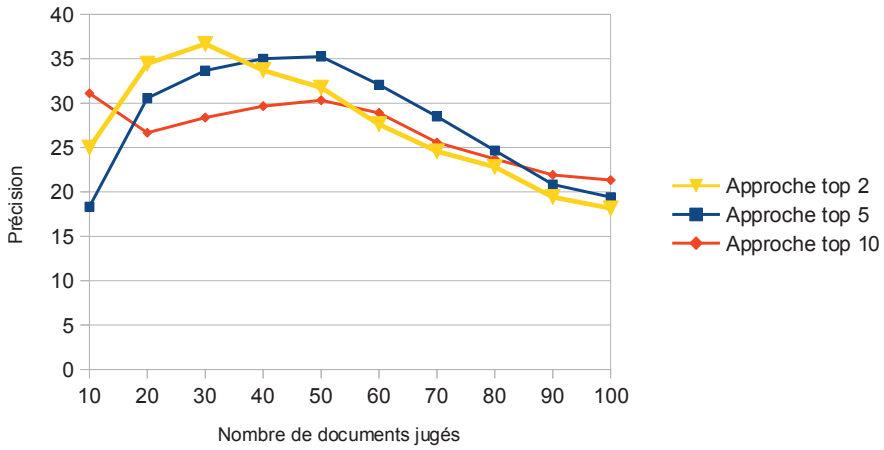


FIGURE B.10 – Comparaison de la précision en exploitant 2, 5 et 10 pages jugées dans le processus de retour de pertinence

ANNEXE C

PRÉSENTATION COMPARATIVE DE NUTCH PAR RAPPORT À HERITRIX

Notre système de découverte de sources comporte un robot d'indexation adapté à notre approche. Le choix du robot d'indexation s'est porté sur Heritrix qui a été présenté section 8.1, page 149. Ce choix a été fait après avoir comparé les caractéristiques de Nutch avec celles d'Heritrix. Cette annexe présente Nutch et la comparaison entre les deux robots d'indexation.

C.1 Nutch

Nutch est également un robot d'indexation open source écrit en Java. Comme Heritrix, la communauté dispose d'une mailing list active. Ce robot d'indexation est diffusé sous licence Apache, ce qui permet aussi un accès complet au code source. Il est basé sur Lucene8 [1] ce qui lui permet, en plus de l'exploration, de proposer un système d'indexation et d'interrogation.

C.1.1 Principe général

Nutch est composé de 4 briques distinctes :
- *Crawler* - Cette brique d'exploration assure la découverte et la collecte de pages Web.
- *WebDB* - Les URLs explorées et le contenu des documents collectés sont stockés dans cette brique.
- *Indexer* - Construit, à partir de la base de données *WebDB*, un index de mot clés.
- *Search Web Application* - Une interface utilisateur de recherche de documents indexés.

1. http://lucene.apache.org/core/

Nous nous intéressons ici aux caractéristiques d'exploration de Nutch en nous basant sur la documentation fournie sur le site [2] et par Khare *et al.* [Khare et al., 2004].

C.1.2 Fonctionnement

La figure C.1 illustre le fonctionnement par itérations de Nutch. Des URLs graines sont également le point d'entrée de ce robot d'indexation. Ces graines sont ajoutées dans la base de données *WebDB* fraîchement créée. Le processus suivant génère un ensemble de listes comportant les URLs à explorer de la base de données *WebDB*. Ces listes sont distribuées à des processus de collecte. Ces processus travaillent ainsi en parallèle et s'occupent d'explorer les URLs de leur liste. Dès qu'un nouveau lien est extrait, il est ajouté dans la base *WebDB*. Ceci permet de générer de nouvelles listes à explorer et ainsi de recommencer le cycle. Le fichier de configuration de Nutch permet de gérer les paramètres de répétition de ce cycle.

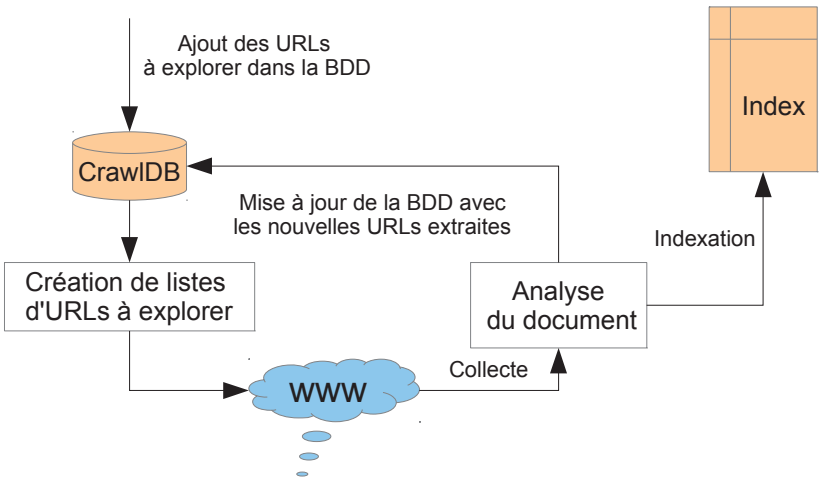


FIGURE C.1 – Fonctionnement de Nutch

Processus de collecte

Le processus de collecte est composé d'un ensemble de scripts, décrits ci-dessous, qui peuvent être lancés individuellement :

- *1 admin db-create* - Création de la brique de stockage *WebDB*.
- *2 inject* - Ajout des graines dans la base de données.
- *3 generate* - Génération des listes d'URLs.
- *4 fetch* - Collecte du contenu pointé par les ULRs des listes.
- *5 updatedb* - Ajout des liens extraits dans la base de données.

Les étapes 3 à 5 sont répétées jusqu'à ce que la profondeur d'exploration soit atteinte, puis :

- *6 updatesegs* - Mise à jour des listes d'URLs avec le score et les liens provenant de la base de données.
- *7 index* - Création d'index des documents collectés pour chaque liste.

2. http://nutch.apache.org/

– *8 dedup* - Elimination du contenu et des URLs en double dans les index.
– *9 merge* - Fusion des index.

Les étapes 6 à 9 sont permises par l'outil Lucene. Ce processus comprend ainsi à la fois un processus d'exploration et un d'indexation.

a. Comparaison d'Heritrix et Nutch

Le tableau C.1 montre une comparaison des robots d'indexation Nutch et Heritrix en se basant sur les caractéristiques décrites précédemment. Ces deux robots peuvent intégrer

Table C.1 – Comparaison entre Heritrix et Nutch

	Heritrix	Nutch
Documentation	+ + +	+
Facilité d'installation	+ + +	+ + +
Langage	Java	Java
Activité Mailing list	+ + +	+ +
Interface par ligne de commande	non	oui
Interface graphique	non	non
Interface Web	oui	non
Prise en main	+ +	+
Collecte ciblée	+	+
Capacité d'extension	+ + +	+ +

des fonctionnalités de collecte ciblée. Cependant, l'avantage d'Heritrix sur Nutch, dans ce domaine, est dû à sa prise en main plus simple, et à son architecture modulaire qui permet d'étendre les fonctionnalités plus aisément. La documentation complète et l'API d'Heritrix contribuent également à le préférer à Nutch. De plus amples comparaisons ont été faites par Della *et al.* [Della Valle et al., 2008] sur ces deux robots d'indexation.

C.1.3 Vue d'ensemble des robots d'indexation

Dans cette section, nous présentons brièvement d'autres robots d'indexation open source. Le tableau C.2 fournit pour chacun d'entre eux une description rapide, le langage de programmation, la présence d'index et les plates-formes supportées.

Le choix du robot d'indexation à utiliser dépend généralement de la tâche de collecte à effectuer. Lorsqu'il s'agit d'une collecte simple d'une petite partie du Web, la plupart des robots d'indexation suffisent. On s'intéressera alors à des préférences quant à la plate-forme supportée ou au langage de programmation. Le choix de la présence d'un index se justifie par l'utilisation d'un moteur de recherche en aval de la collecte. Pour des collectes de grandes collections de documents, YaCy avec l'utilisation du peer2peer, est une solution intéressante : la vitesse n'a ici pas d'importance, l'intérêt repose sur son utilisation simple et son architecture distribuée. Enfin pour des collectes de moyenne envergure, rapide et fiable, on préférera Nutch et surtout Heritrix, pour les raisons évoquées précédemment.

TABLE C.2 – Robots d'indexation open source

Nom	Description	Langage	Index	Plate-forme
ASPseek	Robot d'indexation, recherche en arrière plan	C++	oui	Linux
Bixo	Robot d'indexation vertical (exploration ciblée)	Java	non	Multi
Crawler4j	Robot d'indexation et interface d'exploration programmable	Java	non	Multi
DataparkSearch	Robot d'indexation avec moteur de recherche	C	oui	Multi
Ebot	Robot d'indexation basée sur Erlang	Erlang	oui	Linux
Grub	Robot d'indexation distribuée (p2p)	C#	oui	Linux
Heritrix	Robot d'indexation extensible avec des qualités d'archivage	Java	non	Unix
Hounder	Robot d'indexation, exploration ciblée et moteur de recherche	Java	oui	Multi
HTTrack	Robot d'indexation simple, site miroir	C/C++	non	Multi
Hyper Estraier	Robot d'indexation en ligne de commande	C/C++	oui	Multi
Nutch	Robot d'indexation, utilisant Lucene. Système par plugin	Java	oui	Multi
OpenWebSpider	Robot d'indexation pour la plate-forme .NET	C#/PHP	oui	Multi
Pavuk	Robot d'indexation en ligne de commande, site miroir	C	non	Unix
YaCy	Robot d'indexation p2p	Java	oui	Multi

ANNEXE D

PUBLICATIONS LIÉES À LA THÈSE

[Noël, 2012] Noël R., Automatic relevant source discovery over the internet based on user profile, in *Journée Jeunes Chercheurs, COnférence en Recherche d'Information et Applications - Colloque International Francophone sur l'Écrit et le Document (CORIA-CIFED)*, pages 401-406, Bordeaux, France, 2012.

[Noël et al., 2013a] Noël R., Pauchet A., Grilheres B., Malandain N., Vercouter L. & Brunessaux S., DOWSER : Discovery of Web Sources by Evaluating Relevance, in *atelier "des sources ouvertes au web des données", Sources Ouvertes et Services - Données Liées pour un Web de Données (SOS-DLWD), associé à la plate-forme I.A.*, Lille, France, 2013.

[Noël et al., 2013b] Noël R., Pauchet, A. & Grilheres, B., Procédé de découverte d'un ensemble de sources définissant des pages Web. Brevet 13 02389. Dépôt 15 octobre 2013

[Noël et al., 2014a] Noël R., Pauchet A., Grilheres B., Malandain N., Vercouter L. & Brunessaux S., Dowser : Discovery ofweb sources by evaluating relevance, in *Revue des Nouvelles Technologies de l'Information*, des Sources Ouvertes au Web de Données, RNTI-W-2, pages 41-58, 2014

[Noël et al., 2014b] Noël R., Pauchet A., Grilheres B., Malandain N., Vercouter L. & Brunessaux S., Relevant sources of information are not necessarily popular ones, in *International Conference on Web Intelligence*, Warsaw, Pologne, 8p., 2014.

BIBLIOGRAPHIE

[Aggarwal et al., 2001] Aggarwal, C., Al-Garawi, F., & Yu, P. (2001). Intelligent crawling on the world wide web with arbitrary predicates. In *Proceedings of the 10th international conference on World Wide Web* (pp. 96–105). : ACM.

[Asnicar & Tasso, 1997] Asnicar, F. & Tasso, C. (1997). ifweb : a prototype of user model-based intelligent agent for document filtering and navigation in the world wide web. In *Proceedings of WorkshopAdaptive Systems and User Modeling on the World Wide Web'at 6th International Conference on User Modeling, UM97, Chia Laguna, Sardinia, Italy* (pp. 3–11).

[Baeza-Yates et al., 1999] Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval*, volume 82. Addison-Wesley New York.

[Balabanovic, 1998] Balabanovic, M. (1998). Exploring versus exploiting when learning user models for text recommendation. *User Modeling and User-Adapted Interaction*, 8(1), 71–102.

[Balabanovic & Shoham, 1997] Balabanovic, M. & Shoham, Y. (1997). Fab : content-based, collaborative recommendation. *Communications of the ACM*, 40(3), 66–72.

[Bates, 1990] Bates, M. J. (1990). Where should the person stop and the information search interface start ? *Information Processing & Management*, 26(5), 575–591.

[Bergmark et al., 2002] Bergmark, D., Lagoze, C., & Sbityakov, A. (2002). Focused crawls, tunneling, and digital libraries. *Research and Advanced Technology for Digital Libraries*, (pp. 49–70).

[Bhaskar et al., 2010] Bhaskar, P., Das, A., Pakray, P., & Bandyopadhyay, S. (2010). Theme based english and bengali ad-hoc monolingual information retrieval in fire 2010. *Corpus*, 1, 25–586.

[Bizer et al., 2009] Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., & Hellmann, S. (2009). Dbpedia-a crystallization point for the web of data. *Web Semantics : Science, Services and Agents on the World Wide Web*, 7(3), 154–165.

[Bonabeau et al., 2000] Bonabeau, E., Dorigo, M., & Theraulaz, G. (2000). Inspiration for optimization from social insect behaviour. *Nature*, 406(6791), 39–42.

[Borgman, 1996] Borgman, C. L. (1996). Why are online catalogs still hard to use ? *JASIS*, 47(7), 493–503.

[Bouidghaghen, 2009] Bouidghaghen, O. (2009). Recherche contextuelle d'information dans un environnement mobile. In *CORIA* (pp. 479–486).

[Brewington & Cybenko, 2000] Brewington, B. & Cybenko, G. (2000). How dynamic is the web ? 1. *Computer Networks*, 33(1-6), 257–276.

[Brin & Page, 1998] Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7), 107–117.

[Budzik & Hammond, 1999] Budzik, J. & Hammond, K. (1999). Watson : Anticipating and contextualizing information needs. In *Proceedings of the Annual Meeting-American Society for Information Science*, volume 36 (pp. 727–740). : Citeseer.

[Budzik & Hammond, 2000] Budzik, J. & Hammond, K. (2000). User interactions with everyday applications as context for just-in-time information access. In *Proceedings of the 5th international conference on Intelligent user interfaces* (pp. 44–51). : ACM.

[Bush, 1945] Bush, V. (1945). As we may think.

[Cao et al., 2008] Cao, H., Jiang, D., Pei, J., He, Q., Liao, Z., Chen, E., & Li, H. (2008). Context-aware query suggestion by mining click-through and session data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 875–883). : ACM.

[CDEF, 2008] CDEF (2008). Doctrine du renseignement de l'armée de terre. *Rens 100, Doctrine du renseignement de l'armée de terre, Tome II, Le cycle du renseignement.*

[Chakrabarti et al., 2002] Chakrabarti, S., Punera, K., & Subramanyam, M. (2002). Accelerated focused crawling through online relevance feedback. In *Proceedings of the 11th international conference on World Wide Web* (pp. 148–159). : ACM.

[Chakrabarti et al., 1999a] Chakrabarti, S., Van den Berg, M., & Dom, B. (1999a). Distributed hypertext resource discovery through examples. In *PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES* (pp. 375–386). : Citeseer.

[Chakrabarti et al., 1999b] Chakrabarti, S., Van den Berg, M., & Dom, B. (1999b). Focused crawling : a new approach to topic-specific web resource discovery. *Computer Networks*, 31(11-16), 1623–1640.

[Challam et al., 2007] Challam, V., Gauch, S., & Chandramouli, A. (2007). Contextual search using ontology-based user profiles. In *RIAO*.

[Chang et al., 2006] Chang, Y., Ounis, I., & Kim, M. (2006). Query reformulation using automatically generated query concepts from a document space. *Information processing & management*, 42(2), 453–468.

[Chen et al., 1998] Chen, H., Chung, Y., Ramsey, M., & Yang, C. (1998). A smart itsy bitsy spider for the web.

[Chen & Sycara, 1998] Chen, L. & Sycara, K. (1998). Webmate : a personal agent for browsing and searching. In *Proceedings of the second international conference on Autonomous agents* (pp. 132–139). : ACM.

[Chirita et al., 2005] Chirita, P., Nejdl, W., Paiu, R., & Kohlschütter, C. (2005). Using odp metadata to personalize search. In *Proceedings of the 28th annual international ACM*

SIGIR conference on Research and development in information retrieval (pp. 178–185). : ACM.

[Chirita et al., 2004] Chirita, P., Olmedilla, D., & Nejdl, W. (2004). Pros : A personalized ranking platform for web search. In *Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 431–461). : Springer.

[Cho et al., 1998] Cho, J., Garcia-Molina, H., & Page, L. (1998). Efficient crawling through url ordering. *Computer Networks and ISDN Systems*, 30(1-7), 161–172.

[Chowdhury, 2004] Chowdhury, G. (2004). Introduction to modern information retrieval. *Introduction to Information Retrieval*, (c).

[Crestani, 1995] Crestani, F. (1995). Implementation and evaluation of a relevance feedback device based on neural networks. In *From Natural to Artificial Neural Computation* (pp. 597–604). Springer.

[Croft et al., 2001] Croft, W. B., Cronen-Townsend, S., & Lavrenko, V. (2001). Relevance feedback and personalization : A language modeling perspective. In *DELOS Workshop : Personalisation and Recommender Systems in Digital Libraries*.

[Daoud et al., 2009] Daoud, M., Tamine-Lechani, L., Boughanem, M., & Chebaro, B. (2009). A session based personalized search using an ontological user profile. In *Proceedings of the 2009 ACM symposium on Applied Computing* (pp. 1732–1736). : ACM.

[Davison, 2000] Davison, B. (2000). Topical locality in the web. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 272–279). : ACM.

[De Bra et al., 1994] De Bra, P., Houben, G., Kornatzky, Y., & Post, R. (1994). Information retrieval in distributed hypertexts. In *Proceedings of the 4th RIAO Conference* (pp. 481–491). : Citeseer.

[De Bra & Post, 1994] De Bra, P. & Post, R. (1994). Searching for arbitrary information in the www : The fish-search for mosaic. In *WWW Conference*.

[Della Valle et al., 2008] Della Valle, E., Cerizza, D., Celino, I., Turati, A., Lausen, H., Steinmetz, N., Erdmann, M., & Funk, A. (2008). Realizing service-finder : Web service discovery at web scale. In *European Semantic Technology Conference (ESTC), Vienna*.

[Delort et al., 2003] Delort, J., Bouchon-Meunier, B., & Rifqi, M. (2003). Enhanced web document summarization using hyperlinks. In *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia* (pp. 208–215). : ACM.

[Desmontils & Jacquin, 2002] Desmontils, E. & Jacquin, C. (2002). *Indexing a web site with a terminology oriented ontology*. IOS Press.

[Diligenti et al., 2000] Diligenti, M., Coetzee, F., Lawrence, S., Giles, C., & Gori, M. (2000). Focused crawling using context graphs. In *Proceedings of the 26th international conference on very large data bases* (pp. 527–534).

[Diligenti et al., 2004] Diligenti, M., Maggini, M., Pucci, F., & Scarselli, F. (2004). Design of a crawler with bounded bandwidth. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters* (pp. 292–293). : ACM.

[Dupont et al., 2011] Dupont, G., Adam, S., Lecourtier, Y., et al. (2011). Apprentissage par renforcement pour la recherche d'information interactive. *Actes des 6emes Journées Francophones de Planification, Décision et Apprentissage pour la conduite de systèmes*.

[Ehrig & Maedche, 2003] Ehrig, M. & Maedche, A. (2003). Ontology-focused crawling of web documents. In *Proceedings of the 2003 ACM symposium on Applied computing* (pp. 1174–1178). : ACM.

[Gasparetti & Micarelli, 2003] Gasparetti, F. & Micarelli, A. (2003). Adaptive web search based on a colony of cooperative distributed agents. *Cooperative Information Agents VII*, (pp. 168–183).

[Gasparetti & Micarelli, 2004] Gasparetti, F. & Micarelli, A. (2004). Swarm intelligence : Agents for adaptive web search. In *ECAI*, volume 16 (pp. 1019). : Citeseer.

[Gasparetti & Micarelli, 2005] Gasparetti, F. & Micarelli, A. (2005). User profile generation based on a memory retrieval theory. In *Proc. 1st International Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces (WPRSIUI'05)* (pp. 59–68). : Citeseer.

[Gauch et al., 2003] Gauch, S., Chaffee, J., & Pretschner, A. (2003). Ontology-based personalized search and browsing. *Web Intelligence and Agent Systems*, 1(3/4), 219–234.

[Gauch et al., 2007] Gauch, S., Speretta, M., Chandramouli, A., & Micarelli, A. (2007). User profiles for personalized information access. *The adaptive web*, (pp. 54–89).

[Gauch & Wang, 1997] Gauch, S. & Wang, J. (1997). A corpus analysis approach for automatic query expansion. In *Proceedings of the sixth international conference on Information and knowledge management* (pp. 278–284). : ACM.

[Giroux et al., 2008] Giroux, P., Brunessaux, S., Brunessaux, S., Doucy, J., Dupont, G., Grilheres, B., Mombrun, Y., Saval, A., & des Portes, P. d. (2008). Weblab : An integration infrastructure to ease the development of multimedia processing applications. In *International Conference on Software and System Engineering and their Applications (ICSSEA)* (pp. 129).

[Granka et al., 2004] Granka, L., Joachims, T., & Gay, G. (2004). Eye-tracking analysis of user behavior in www search. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 478–479). : ACM.

[Gusfield, 1997] Gusfield, D. (1997). *Algorithms on strings, trees, and sequences : computer science and computational biology*. Cambridge Univ Pr.

[Harman, 1992] Harman, D. (1992). Relevance feedback and other query modification techniques. *Information Retrieval : data structures and algorithms*, (pp. 241–263).

[Harrathi & Calabretto, 2006] Harrathi, R. & Calabretto, S. (2006). Un modele de qualité de l'information. *EGC, volume RNTI-E-6 of Revue des Nouvelles Technologies de l'Information*, (pp. 299–304).

[Haveliwala, 2003] Haveliwala, T. (2003). Topic-sensitive pagerank : A context-sensitive ranking algorithm for web search. *Knowledge and Data Engineering, IEEE Transactions on*, 15(4), 784–796.

[He & Göker, 2000] He, D. & Göker, A. (2000). Detecting session boundaries from web user logs. In *Proceedings of the BCS-IRSG 22nd annual colloquium on information retrieval research* (pp. 57–66).

[Hersovici et al., 1998] Hersovici, M., Jacovi, M., Maarek, Y., Pelleg, D., Shtalhaim, M., & Ur, S. (1998). The shark-search algorithm. an application : tailored web site mapping. *Computer Networks and ISDN Systems*, 30(1-7), 317–326.

[Joachims et al., 1997] Joachims, T., Freitag, D., Mitchell, T., et al. (1997). Webwatcher : A tour guide for the world wide web. In *International Joint Conference on Artificial Intelligence*, volume 15 (pp. 770–777). : LAWRENCE ERLBAUM ASSOCIATES LTD.

[Jones et al., 2000] Jones, K., Walker, S., & Robertson, S. (2000). A probabilistic model of information retrieval : development and comparative experiments.

[Khare et al., 2004] Khare, R., Cutting, D., Sitaker, K., & Rifkin, A. (2004). Nutch : A flexible and scalable open-source web search engine. *Oregon State University*.

[Khoo & Poo, 1994] Khoo, C. S. & Poo, D. C. (1994). An expert system approach to online catalog subject searching. *Information processing & management*, 30(2), 223–238.

[Kim & Chan, 2003] Kim, H. & Chan, P. (2003). Learning implicit user interest hierarchy for context in personalization. In *Proceedings of the 8th international conference on Intelligent user interfaces* (pp. 101–108). : ACM.

[Kleinberg, 1999] Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5), 604–632.

[Kostadinov, 2003] Kostadinov, D. (2003). *La personnalisation de l'information, définition de modèle de profil utilisateur. rapport de dea*. PhD thesis, Master's thesis, Université de Versailles, France.

[Kostadinov, 2008] Kostadinov, D. (2008). *Personnalisation de l'information : une approche de gestion de profils et de reformulation de requï¿½tes*. PhD thesis, Universitï¿½ de Versailles.

[Koutrika & Ioannidis, 2005] Koutrika, G. & Ioannidis, Y. (2005). A unified user profile framework for query disambiguation and personalization. In *PIA 2005 Workshop on New Technologies for Personalized Information Access* (pp.44). : Citeseer.

[Lawrence, 2000] Lawrence, S. (2000). Context in web search. *IEEE Data Engineering Bulletin*, 23(3), 25–32.

[Leung et al., 2006] Leung, C., Chan, S., & Chung, F. (2006). A collaborative filtering framework based on fuzzy association rules and multiple-level similarity. *Knowledge and Information Systems*, 10(3), 357–381.

[Li & Zhu, 2013] Li, Z. & Zhu, M. (2013). A light-weight relevance feedback solution for large scale content-based video retrieval.

[Lieberman, 1997] Lieberman, H. (1997). Autonomous interface agents. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 67–74). : ACM.

[Lieberman et al., 1995] Lieberman, H. et al. (1995). Letizia : An agent that assists web browsing. In *International Joint Conference on Artificial Intelligence*, volume 14 (pp. 924–929). : LAWRENCE ERLBAUM ASSOCIATES LTD.

[Lin et al., 2006] Lin, H.-C., Wang, L.-H., & Chen, S.-M. (2006). Query expansion for document retrieval based on fuzzy rules and user relevance feedback techniques. *Expert Systems with Applications*, 31(2), 397–405.

[Liu, 2007] Liu, B. (2007). *Web data mining : exploring hyperlinks, contents, and usage data*. Springer Verlag.

[Liu et al., 2004] Liu, F., Yu, C., & Meng, W. (2004). Personalized web search for improving retrieval effectiveness. *IEEE Transactions on knowledge and data engineering*, (pp. 28–40).

[Lu et al., 2002] Lu, W., Chien, L., & Lee, H. (2002). Translation of web queries using anchor text mining. *ACM Transactions on Asian Language Information Processing (TALIP)*, 1(2), 159–172.

[Maisonnasse, 2008] Maisonnasse, L. (2008). *Les supports de vocabulaires pour les systèmes de recherche d'information orientés précision : application aux graphes pour la recherche d'information médicale.* PhD thesis, Université Joseph-Fourier-Grenoble I.

[Marchiori, 1997] Marchiori, M. (1997). The quest for correct information on the web : Hyper search engines. *Computer Networks and ISDN Systems*, 29(8-13), 1225–1235.

[Maron & Kuhns, 1960] Maron, M. & Kuhns, J. (1960). On relevance, probabilistic indexing and information retrieval. *Journal of the ACM (JACM)*, 7(3), 216–244.

[Mc Gowan, 2003] Mc Gowan, J. (2003). *A multiple model approach to personalised information access.* PhD thesis, University College Dublin.

[Menczer, 2004] Menczer, F. (2004). Lexical and semantic clustering by web links. *Journal of the American Society for Information Science and Technology*, 55(14), 1261–1269.

[Menczer & Belew, 2000] Menczer, F. & Belew, R. (2000). Adaptive retrieval agents : Internalizing local context and scaling up to the web. *Machine Learning*, 39(2), 203–242.

[Menczer et al., 1995] Menczer, F., Belew, R., & Willuhn, W. (1995). Artificial life applied to adaptive information agents. In *In Proceedings of the 1995 AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments* : Citeseer.

[Menczer et al., 2001] Menczer, F., Pant, G., Srinivasan, P., & Ruiz, M. E. (2001). Evaluating topic-driven web crawlers. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 241–249). : ACM.

[Mendes et al., 2011] Mendes, P. N., Jakob, M., García-Silva, A., & Bizer, C. (2011). Dbpedia spotlight : Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems (I-Semantics).*

[Micarelli et al., 2007] Micarelli, A., Gasparetti, F., Sciarrone, F., & Gauch, S. (2007). Personalized search on the world wide web. *The Adaptive Web*, (pp. 195–230).

[Micarelli & Sciarrone, 2004] Micarelli, A. & Sciarrone, F. (2004). Anatomy and empirical evaluation of an adaptive web-based information filtering system. *User Modeling and User-Adapted Interaction*, 14(2), 159–200.

[Michlmayr & Cayzer, 2007] Michlmayr, E. & Cayzer, S. (2007). Learning user profiles from tagging data and leveraging them for personal (ized) information access. In *Proceedings of the Workshop on Tagging and Metadata for Social Information Organization, 16th International World Wide Web Conference (WWW2007)* : Citeseer.

[Mittal et al., 2010] Mittal, N., Nayak, R., Govil, M. C., & Jain, K. C. (2010). Evaluation of a hybrid approach of personalized web information retrieval using the fire data set. In *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India* (pp.5̃2). : ACM.

[Mladenic, 1996] Mladenic, D. (1996). Personal webwatcher : design and implementation.

[Mohr et al., 2004] Mohr, G., Stack, M., Rnitovic, I., Avery, D., & Kimpton, M. (2004). Introduction to heritrix. In *4th International Web Archiving Workshop.*

[Mombrun, 2012] Mombrun, Y. (2012). Évaluation de l'information disponible sur internet application au renseignement d'origine sources ouvertes.

[Mombrun et al., 2010] Mombrun, Y., Pauchet, A., Grilhères, B., Canu, S., et al. (2010). Collecte, analyse et évaluation d'informations en sources ouvertes. In *Atelier COTA des 21es Journées francophones d'Ingénierie des Connaissances*.

[Najork & Heydon, 2002] Najork, M. & Heydon, A. (2002). High-performance web crawling. *Handbook of massive data sets*.

[Najork & Wiener, 2001] Najork, M. & Wiener, J. (2001). Breadth-first crawling yields high-quality pages. In *Proceedings of the 10th international conference on World Wide Web* (pp. 114–118). : ACM.

[Nottelmann & Fuhr, 2003] Nottelmann, H. & Fuhr, N. (2003). From retrieval status values to probabilities of relevance for advanced ir applications. *Information retrieval*, 6(3-4), 363–388.

[OTAN, 2001] OTAN (2001). Aintp-3(a). *The NATO Military Intelligence Data Exchange Standard*.

[OTAN, 2002] OTAN (2002). Supreme allied commander atlantic of the north atlantic treaty organization. *Intelligence Exploitation of the Internet, novembre*.

[OTAN, 2005] OTAN (2005). North atlantic treaty organization standardization agency. *AJP-2.1 (A)*.

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

[Paik et al., 2011] Paik, J. H., Mitra, M., Parui, S. K., & Järvelin, K. (2011). Gras : An effective and efficient stemming algorithm for information retrieval. *ACM Transactions on Information Systems (TOIS)*, 29(4), 19.

[Pant & Menczer, 2002] Pant, G. & Menczer, F. (2002). Myspiders : Evolve your own intelligent web crawlers. *Autonomous agents and multi-agent systems*, 5(2), 221–229.

[Pant et al., 2004] Pant, G., Srinivasan, P., & Menczer, F. (2004). Crawling the web. In *Web Dynamics* (pp. 153–177). Springer.

[Patel & Schmidt, 2011] Patel, A. & Schmidt, N. (2011). Application of structured document parsing to focused web crawling. *Computer Standards & Interfaces*, 33(3), 325–331.

[Patterns, 1999] Patterns, B. (1999). Robert cooley ?, bamshad mobasher, and jaideep srivastava department of computer science and engineering. *Knowledge and information systems*, 1, 00–00.

[Pazzani et al., 1996] Pazzani, M., Muramatsu, J., Billsus, D., et al. (1996). Syskill & webert : Identifying interesting web sites. In *Proceedings of the national conference on artificial intelligence* (pp. 54–61).

[Pinkerton, 1994] Pinkerton, B. (1994). Finding what people want : Experiences with the webcrawler. In *Proceedings of the Second International World Wide Web Conference*, volume 94 (pp. 17–20). : Chicago.

[Pirolli & Pitkow, 1999] Pirolli, P. & Pitkow, J. (1999). Distributions of surfers' paths through the world wide web : Empirical characterizations. *World Wide Web*, 2(1), 29–45.

[Pretschner & Gauch, 1999] Pretschner, A. & Gauch, S. (1999). Ontology based personalized search. In *Tools with Artificial Intelligence, 1999. Proceedings. 11th IEEE International Conference on* (pp. 391–398). : IEEE.

[Radlinski & Joachims, 2006] Radlinski, F. & Joachims, T. (2006). Evaluating the robustness of learning from implicit feedback. *arXiv preprint cs/0605036*.

[Raghavan & Sever, 1995] Raghavan, V. & Sever, H. (1995). On the reuse of past optimal queries. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 344–350). : ACM.

[Ramshaw & Marcus, 1995] Ramshaw, L. & Marcus, M. (1995). Text chunking using transformation-based learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora* (pp. 82–94). : Cambridge MA, USA.

[Rennie & McCallum, 1999] Rennie, J. & McCallum, A. (1999). Using reinforcement learning to spider the web efficiently. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-* (pp. 335–343). : MORGAN KAUFMANN PUBLISHERS, INC.

[Rhodes & Maes, 2000] Rhodes, B. & Maes, P. (2000). Just-in-time information retrieval agents. *IBM Systems journal*, 39(3.4), 685–704.

[Rich, 1979] Rich, E. (1979). User modeling via stereotypes*. *Cognitive science*, 3(4), 329–354.

[Robertson & Jones, 1976] Robertson, S. & Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3), 129–146.

[Robertson & Walker, 1994] Robertson, S. E. & Walker, S. (1994). Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 232–241). : Springer-Verlag New York, Inc.

[Rocchio, 1971] Rocchio, J. J. (1971). Relevance feedback in information retrieval.

[Rui et al., 1998] Rui, Y., Huang, T. S., Ortega, M., & Mehrotra, S. (1998). Relevance feedback : a power tool for interactive content-based image retrieval. *Circuits and Systems for Video Technology, IEEE Transactions on*, 8(5), 644–655.

[Rungsawang & Angkawattanawit, 2005] Rungsawang, A. & Angkawattanawit, N. (2005). Learnable topic-specific web crawler. *Journal of Network and Computer Applications*, 28(2), 97–114.

[Ruthven & Lalmas, 2003] Ruthven, I. & Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18(02), 95–145.

[Ruthven et al., 2002] Ruthven, I., Lalmas, M., & van Rijsbergen, K. (2002). Ranking expansion terms with partial and ostensive evidence. In *fourth international conference on conceptions of library and information science : emerging frameworks and methods* (pp. 199).

[Saint Requier A., 2012] Saint Requier A., Dupont G., A. S. (2012). Selection adaptative de services de recherche d'information web en fonction du besoin de l'utilisateur. In *Proceedings of the sos-dlwd2012*.

[Salton, 1969] Salton, G. (1969). A comparison between manual and automatic indexing methods. *American Documentation*, 20(1), 61–71.

[Salton, 1971] Salton, G. (1971). The smart retrieval system—experiments in automatic document processing.

[Salton & Buckley, 1997] Salton, G. & Buckley, C. (1997). Improving retrieval performance by relevance feedback. *Readings in information retrieval*, (pp. 355–364).

[Salton & McGill, 1983] Salton, G. & McGill, M. J. (1983). Introduction to modern information retrieval.

[Salton & Yang, 1973] Salton, G. & Yang, C. (1973). On the specification of term values in automatic indexing. *Journal of documentation*, 29(4), 351–372.

[Serrano et al., 2012] Serrano, L., Bouzid, M., Charnois, T., & GRILHERES, B. (2012). Vers un système de capitalisation des connaissances : extraction d'événements par combinaison de plusieurs approches. *SOS-DLWD'2012 at EGC*.

[Shen et al., 2005] Shen, X., Tan, B., & Zhai, C. (2005). Implicit user modeling for personalized search. In *Proceedings of the 14th ACM international conference on Information and knowledge management* (pp. 824–831). : ACM.

[Shkapenyuk & Suel, 2002] Shkapenyuk, V. & Suel, T. (2002). Design and implementation of a high-performance distributed web crawler. In *Data Engineering, 2002. Proceedings. 18th International Conference on* (pp. 357–368). : IEEE.

[Shneiderman et al., 1997] Shneiderman, B., Byrd, D., & Croft, W. (1997). Clarifying search : A user-interface framework for text searches. *D-lib magazine*, 3(1), 18–20.

[Sieg et al., 2007] Sieg, A., Mobasher, B., & Burke, R. (2007). Web search personalization with ontological user profiles. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management* (pp. 525–534). : ACM.

[Sieg et al., 2004] Sieg, A., Mobasher, B., Lytinen, S., & Burke, R. (2004). Using concept hierarchies to enhance user queries in web-based information retrieval. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications* (pp. 226–234). : Citeseer.

[Sigurosson, 2005] Sigurosson, K. (2005). Adaptive revisiting with heritrix.

[Speretta & Gauch, 2005] Speretta, M. & Gauch, S. (2005). Personalized search based on user search histories. In *Web Intelligence, 2005. Proceedings. The 2005 IEEE/-WIC/ACM International Conference on* (pp. 622–628). : IEEE.

[Spink et al., 2000] Spink, A., Jansen, B. J., & Ozmultu, H. C. (2000). Use of query reformulation and relevance feedback by excite users. *Internet research*, 10(4), 317–328.

[Spink et al., 2001] Spink, A., Wolfram, D., Jansen, M., & Saracevic, T. (2001). Searching the web : The public and their queries. *Journal of the American society for information science and technology*, 52(3), 226–234.

[Srinivasan et al., 2005] Srinivasan, P., Menczer, F., & Pant, G. (2005). A general evaluation framework for topical crawlers. *Information Retrieval*, 8(3), 417–447.

[Stephenson, 1989] Stephenson, G. A. (1989). Knowledge browsing - front ends to statistical databases. In *Proceedings of the 4th International Working Conference SSDBM on Statistical and Scientific Database Management* (pp. 327–337). London, UK : Springer-Verlag.

[Sun et al., 2005] Sun, J., Zeng, H., Liu, H., Lu, Y., & Chen, Z. (2005). Cubesvd : a novel approach to personalized web search. In *Proceedings of the 14th international conference on World Wide Web* (pp. 382–390). : ACM.

[Tamine et al., 2007] Tamine, L., Zemirli, N., & Bahsoun, W. (2007). Approche statistique pour la définition du profil d'un utilisateur de système de recherche d'information. *Information-Interaction-Intelligence (I3)*, 7(1), 5–25.

[Tamine-Lechani et al., 2006] Tamine-Lechani, L., Boughanem, M., & Chrisment, C. (2006). Accès personnalisé à l'information : Vers un modèle basé sur les diagrammes d'influence. *Information-Interaction-Intelligence (I3)*, 69, 90.

[Tan et al., 2006] Tan, B., Shen, X., & Zhai, C. (2006). Mining long-term search history to improve search accuracy. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 718–723). : ACM.

[Tanudjaja & Mui, 2002] Tanudjaja, F. & Mui, L. (2002). Persona : A contextualized and personalized web search. In *System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on* (pp. 1232–1240). : IEEE.

[Uemura et al., 2012] Uemura, Y., Itokawa, T., Kitasuka, T., & Aritsugi, M. (2012). An effectively focused crawling system. In *Innovations in Intelligent Machines–2* (pp. 61–76). Springer.

[Van Rijsbergen, 1986] Van Rijsbergen, C. J. (1986). A non-classical logic for information retrieval. *The computer journal*, 29(6), 481–485.

[Wærn, 2004] Wærn, A. (2004). User involvement in automatic filtering : An experimental study. *User Modeling and User-Adapted Interaction*, 14(2), 201–237.

[Wen et al., 2001] Wen, J., Nie, J., & Zhang, H. (2001). Clustering user queries of a search engine. In *Proceedings of the 10th international conference on World Wide Web* (pp. 162–168). : ACM.

[White et al., 2005] White, R., Ruthven, I., & Jose, J. (2005). A study of factors affecting the utility of implicit relevance feedback. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 35–42). : ACM.

[Willie & Bruza, 1995] Willie, S. & Bruza, P. (1995). Users' models of the information space : the case for two search models. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 205–210). : ACM.

[Witten & Milne, 2008] Witten, I. & Milne, D. (2008). An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence : an Evolving Synergy, AAAI Press, Chicago, USA* (pp. 25–30).

[Yan et al., 2003] Yan, R., Hauptmann, A. G., & Jin, R. (2003). Negative pseudo-relevance feedback in content-based video retrieval. In *Proceedings of the eleventh ACM international conference on Multimedia* (pp. 343–346). : ACM.

[Yuwono et al., 1995] Yuwono, B., Lam, S., Ying, J., & Lee, D. (1995). A world wide web resource discovery system.

[Zamir & Etzioni, 1999] Zamir, O. & Etzioni, O. (1999). Grouper : a dynamic clustering interface to web search results. *Computer Networks*, 31(11), 1361–1374.

[Zhai & Lafferty, 2001] Zhai, C. & Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management* (pp. 403–410). : ACM.

[Zhang & Seo, 2001] Zhang, B.-T. & Seo, Y.-W. (2001). Personalized web-document filtering using reinforcement learning. *Applied Artificial Intelligence*, 15(7), 665–685.

[Zhou & Huang, 2003] Zhou, X. S. & Huang, T. S. (2003). Relevance feedback in image retrieval : A comprehensive review. *Multimedia systems*, 8(6), 536–544.

