Utilisation de profils utilisateurs pour l'accès à une bibliothèque numérique

Thanh Trung Van

▶ **To cite this version:**

Thanh Trung Van. Utilisation de profils utilisateurs pour l'accès à une bibliothèque numérique. Bibliothèque électronique [cs.DL]. Ecole Nationale Supérieure des Mines de Saint-Etienne, 2008. Français. NNT : 2008EMSE0036 . tel-00785130

HAL Id: tel-00785130
https://theses.hal.science/tel-00785130

Submitted on 5 Feb 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Ecole Nationale Supérieure des Mines
SAINT-ETIENNE

N° d'ordre: 501 I

THÈSE

présentée par

Thanh-Trung VAN

pour obtenir le grade de

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

spécialité INFORMATIQUE

Utilisation de profils utilisateurs pour l'accès à une bibliothèque numérique

Soutenue à Saint-Étienne, le 01 décembre 2008

Membres du jury

Présidente :	Brigitte GRAU	Professeur, ENS d'Informatique pour l'industrie et l'entreprise, Evry
Rapporteurs :	Philippe MULHEM	Chargé de recherche CNRS HDR, Grenoble
	Jean-Marie PINON	Professeur, INSA de Lyon
Examinateur :	Mohand BOUGHANEM	Professeur, Université Paul Sabastier, Toulouse
Directeur de thèse :	Alexandre DOLGUI	Professeur, ENS des Mines de Saint-Étienne
Directeur de recherche :	Michel BEIGBEDER	Maître-Assistant, ENS des Mines de Saint-Étienne

Spécialités doctorales :	**Responsables :**
SCIENCES ET GENIE DES MATERIAUX	J. DRIVER Directeur de recherche - Centre SMS
MECANIQUE ET INGENIERIE	A. VAUTRIN Professeur - Centre SMS
GENIE DES PROCEDES	G. THOMAS Professeur - Centre SPIN
SCIENCES DE LA TERRE	B. GUY Maître de recherche - Centre SPIN
SCIENCES ET GENIE DE L'ENVIRONNEMENT	J. BOURGOIS Professeur - Centre SITE
MATHEMATIQUES APPLIQUEES	E. TOUBOUL Ingénieur - Centre G2I
INFORMATIQUE	O. BOISSIER Professeur - Centre G2I
IMAGE, VISION, SIGNAL	JC. PINOLI Professeur - Centre CIS
GENIE INDUSTRIEL	P. BURLAT Professeur - Centre G2I
MICROELECTRONIQUE	Ph. COLLOT Professeur - Centre CMP

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

AVRIL	Stéphane	MA	Mécanique & Ingénierie	CIS
BATTON-HUBERT	Mireille	MA	Sciences & Génie de l'Environnement	SITE
BENABEN	Patrick	PR 2	Sciences & Génie des Matériaux	CMP
BERNACHE-ASSOLANT	Didier	PR 1	Génie des Procédés	CIS
BIGOT	Jean-Pierre	MR	Génie des Procédés	SPIN
BILAL	Essaïd	DR	Sciences de la Terre	SPIN
BOISSIER	Olivier	PR 2	Informatique	G2I
BOUCHER	Xavier	MA	Génie Industriel	G2I
BOUDAREL	Marie-Reine	MA	Sciences de l'inform. & com.	DF
BOURGOIS	Jacques	PR 1	Sciences & Génie de l'Environnement	SITE
BRODHAG	Christian	MR	Sciences & Génie de l'Environnement	SITE
BURLAT	Patrick	PR 2	Génie industriel	G2I
CARRARO	Laurent	PR 1	Mathématiques Appliquées	G2I
COLLOT	Philippe	PR 1	Microélectronique	CMP
COURNIL	Michel	PR 1	Génie des Procédés	SPIN
DAUZERE-PERES	Stéphane	PR 1	Génie industriel	CMP
DARRIEULAT	Michel	ICM	Sciences & Génie des Matériaux	SMS
DECHOMETS	Roland	PR 1	Sciences & Génie de l'Environnement	SITE
DESRAYAUD	Christophe	MA	Mécanique & Ingénierie	SMS
DELAFOSSE	David	PR 1	Sciences & Génie des Matériaux	SMS
DOLGUI	Alexandre	PR 1	Génie Industriel	G2I
DRAPIER	Sylvain	PR 2	Mécanique & Ingénierie	SMS
DRIVER	Julian	DR	Sciences & Génie des Matériaux	SMS
FOREST	Bernard	PR 1	Sciences & Génie des Matériaux	CIS
FORMISYN	Pascal	PR 1	Sciences & Génie de l'Environnement	SITE
FORTUNIER	Roland	PR 1	Sciences & Génie des Matériaux	CMP
FRACZKIEWICZ	Anna	MR	Sciences & Génie des Matériaux	SMS
GARCIA	Daniel	CR	Génie des Procédés	SPIN
GIRARDOT	Jean-Jacques	MR	Informatique	G2I
GOEURIOT	Dominique	MR	Sciences & Génie des Matériaux	SMS
GOEURIOT	Patrice	MR	Sciences & Génie des Matériaux	SMS
GRAILLOT	Didier	DR	Sciences & Génie de l'Environnement	SITE
GROSSEAU	Philippe	MR	Génie des Procédés	SPIN
GRUY	Frédéric	MR	Génie des Procédés	SPIN
GUILHOT	Bernard	DR	Génie des Procédés	CIS
GUY	Bernard	MR	Sciences de la Terre	SPIN
GUYONNET	René	DR	Génie des Procédés	SPIN
HERRI	Jean-Michel	PR 2	Génie des Procédés	SPIN
KLÖCKER	Helmut	MR	Sciences & Génie des Matériaux	SMS
LAFOREST	Valérie	CR	Sciences & Génie de l'Environnement	SITE
LI	Jean-Michel	EC (CCI MP)	Microélectronique	CMP
LONDICHE	Henry	MR	Sciences & Génie de l'Environnement	SITE
MOLIMARD	Jérôme	MA	Sciences & Génie des Matériaux	SMS
MONTHEILLET	Frank	DR 1 CNRS	Sciences & Génie des Matériaux	SMS
PERIER-CAMBY	Laurent	PR 1	Génie des Procédés	SPIN
PIJOLAT	Christophe	PR 1	Génie des Procédés	SPIN
PIJOLAT	Michèle	PR 1	Génie des Procédés	SPIN
PINOLI	Jean-Charles	PR 1	Image, Vision, Signal	CIS
STOLARZ	Jacques	CR	Sciences & Génie des Matériaux	SMS
SZAFNICKI	Konrad	CR	Sciences de la Terre	SITE
THOMAS	Gérard	PR 1	Génie des Procédés	SPIN
VALDIVIESO	François	MA	Sciences & Génie des Matériaux	SMS
VAUTRIN	Alain	PR 1	Mécanique & Ingénierie	SMS
VIRICELLE	Jean-Paul	MR	Génie des procédés	SPIN
WOLSKI	Krzysztof	CR	Sciences & Génie des Matériaux	SMS
XIE	Xiaolan	PR 1	Génie industriel	CIS

Glossaire :	
PR 1	Professeur $1^{ère}$ catégorie
PR 2	Professeur $2^{ème}$ catégorie
MA(MDC)	Maître assistant
DR (DR1)	Directeur de recherche
Ing.	Ingénieur
MR(DR2)	Maître de recherche
CR	Chargé de recherche
EC	Enseignant-chercheur
ICM	Ingénieur en chef des mines

Centres :	
SMS	Sciences des Matériaux et des Structures
SPIN	Sciences des Processus Industriels et Naturels
SITE	Sciences Information et Technologies pour l'Environnement
G2I	Génie Industriel et Informatique
CMP	Centre de Microélectronique de Provence
CIS	Centre Ingénierie et Santé

Remerciements

Je tiens à remercier mon directeur de recherche, M. Michel Beigbeder, pour avoir encadré mes recherches et pour m'avoir donné des conseils précieux pendant ces trois années de thèse.

Je remercie mes rapporteurs, M. Philippe Mulhem et M. Jean-Marie Pinon, pour avoir consacré du temps à lire le manuscrit et pour leurs remarques très pertinentes.

J'adresse mes remerciements à Mme. Brigitte Grau et M. Mohand Boughanem pour avoir accepté d'être examinateurs de mon jury.

Je remercie mon directeur de thèse, M. Alexandre Dolgui, pour m'avoir permis d'effectuer mes recherches au sein du centre G2I.

Je remercie tout le personnel de l'École des Mines de Saint-Etienne pour leur accueil, pour leur amitié, et pour leur aide. En particulier, je pense à Annie, Jean-Jacques, Marie-Line, Amélie, Hoan, Annabelle, Xavier, Olga, Hung, Kafil, Hien, Jean-François, Hai, Gregory, Steven.

Enfin, je remercie ma famille et mes amis pour m'avoir encouragé pendant ces années et pour la confiance qu'ils m'ont accordée.

Table des matières

Table des matières **i**

1 Introduction générale **1**
1.1 Introduction à la recherche d'information 1
1.1.1 Une courte histoire 1
1.1.2 Définition de la recherche d'information 2
1.2 Problématique de la thèse 3
1.3 Contribution de la thèse 4
1.4 Organisation du mémoire 5

2 Recherche d'information **7**
2.1 La recherche d'information : côté de l'utilisateur et côté du système 7
2.1.1 Côté de l'utilisateur 7
2.1.2 Côté du système 8
2.1.2.1 Première phase : création de la collection de documents et indexation 8
2.1.2.2 Deuxième phase : Interaction avec l'utilisateur 10
2.2 Quelques modèles connus de la recherche d'information 11
2.2.1 Modèle booléen 11
2.2.2 Modèle vectoriel 12
2.2.3 Modèles basés sur les approches probabilistes 13
2.2.3.1 Modèle probabiliste 13
2.2.3.2 Modèle de langue 14
2.3 Évaluation 15
2.3.1 Collections de test 15
2.3.2 Notion de pertinence 16
2.3.2.1 Jugements de pertinence 16
2.3.3 Méthode *pooling* 17
2.3.4 Mesures de performance 18
2.3.4.1 Précision et rappel 18
2.3.4.2 La courbe précision/rappel 18
2.3.4.3 Précision à n documents 19
2.3.4.4 Précision moyenne (MAP) 19
2.3.5 Campagnes d'évaluation 19
2.3.5.1 TREC 19
2.3.5.2 INEX 20

2.4 Méthodes de combinaison pour la recherche d'information 21
2.4.1 Combinaison produit 21
2.4.2 Combinaison linéaire 22
2.4.3 Combinaison basée sur la théorie Dempster-Shafer 22
2.5 Outils 24
2.5.1 SMART 24
2.5.2 Lucy/Zettair 24
2.5.3 trec_eval 24
2.6 Bilan 25

3 Systèmes Personnalisés Actuels 27
3.1 Profils utilisateurs 27
3.1.1 Définition de profils utilisateurs 27
3.1.2 Modèles de représentation de profils utilisateurs 28
3.1.2.1 Modèle vectoriel 28
3.1.2.2 Modèle sémantique à base d'ontologie 28
3.1.2.3 Modèle multidimensionnel 29
3.1.2.4 Autres modèles de représentations 30
3.1.3 Acquisition d'information 31
3.1.3.1 Acquisition explicite 31
3.1.3.2 Acquisition implicite 32
3.1.3.3 Approche hybride 33
3.1.4 Techniques de construction et de mise à jour de profils utilisateurs . 33
3.1.4.1 TF-IDF 33
3.1.4.2 Les méthodes de classification 34
3.1.4.3 Les méthodes de *clustering* 34
3.1.4.4 Combinaison de plusieurs méthodes 35
3.2 Utilisation de profils utilisateurs dans les systèmes personnalisés 35
3.2.1 Reformulation de requêtes utilisateurs 36
3.2.2 Visualisation de résultats 37
3.2.3 Re-classement de résultats 37
3.2.4 Systèmes de recommandation 38
3.2.4.1 Systèmes de recommandation basés sur le contenu 38
3.2.4.2 Systèmes de recommandation collaborative 41
3.2.4.3 Systèmes hybrides 42
3.3 Bilan 42

4 La personnalisation dans les bibliothèques numériques 45
4.1 Bibliothèques numériques 45
4.1.1 Qu'est-ce qu'une bibliothèque numérique ? 45
4.1.2 Histoire et évolution de bibliothèques numériques 46
4.1.3 Architecture d'une bibliothèque numérique moderne 47
4.1.4 Recherche d'information dans les bibliothèques numériques 50
4.1.5 Avantages et inconvénients des bibliothèques numériques 50
4.2 Quelques exemples de bibliothèques numériques 52
4.2.1 Projet Gutenberg 52

4.2.2 Gallica . . . 52
4.2.3 Google recherche de livres . . . 52
4.2.4 CiteSeer . . . 53
4.2.5 Bibliothèque CODESNET . . . 53
4.3 Bibliothèques numériques personnalisées . . . 53
4.3.1 Les approches de personnalisation . . . 54
4.3.2 Quelques systèmes actuels . . . 54
4.4 Bilan . . . 56

5 Analyse de liens et de citations et applications dans la recherche d'information 59
5.1 Bibliométrie . . . 59
5.1.1 Références et citations . . . 60
5.1.1.1 Motivation des citations . . . 60
5.1.1.2 Graphe de citations . . . 61
5.1.2 Lois de la bibliométrie . . . 61
5.1.3 Les applications de la bibliométrie . . . 62
5.1.3.1 Bibliométrie évaluative . . . 62
5.1.3.2 Bibliométrie relationnelle . . . 64
5.2 Webométrie . . . 67
5.2.1 Similarités et différences entre liens Web et citations scientifiques . 69
5.2.2 Bibliométrie, scientométrie, infométrie, cybermétrie, et webométrie 69
5.3 Applications des méthodes d'analyse de citations et de liens . . . 70
5.3.1 Classification et regroupement . . . 70
5.3.2 Algorithme PageRank . . . 71
5.3.3 Algorithme HITS . . . 72
5.3.4 Utilisation de contexte de citations . . . 72
5.3.5 Stratégies de recherche basée sur les citations . . . 73
5.3.6 Quelques autres applications . . . 74
5.4 Les méthodes basées sur les citations/liens sont-elles toujours bonnes ? . . . 75
5.4.1 Dérive de sujet . . . 75
5.4.2 Caractéristiques des collections de documents . . . 75
5.5 Bilan . . . 77

6 Notre approche pour la recherche d'information personnalisée dans les bibliothèques numériques 79
6.1 Introduction à notre approche . . . 79
6.2 Bases de données bibliographiques . . . 82
6.2.1 *Web of Science* de Thomson ISI . . . 83
6.2.1.1 Accès au *Web of Science* . . . 84
6.2.1.2 Avantages et inconvénients du *Web of Science* . . . 86
6.2.2 Quelques autres bases de données bibliographique . . . 86
6.3 La méthode des co-citations sur le Web . . . 87
6.3.1 Citations sur le Web . . . 88
6.3.2 Co-citations sur le Web . . . 90
6.3.3 Avantages et inconvénients de la méthode des co-citations sur le Web 91

6.4 Mesures de similarité 91
6.5 Combinaison des scores 93
6.6 Bilan 94

7 Expérimentations et résultats **97**
7.1 Méthodes d'évaluation 97
7.1.1 Conduite des expérimentations avec les vrais utilisateurs 97
7.1.2 Construction des jeux de données standards 98
7.1.3 Simulation 98
7.1.4 Approche retenue 101
7.2 Collection de test INEX 102
7.2.1 Corpus de documents 103
7.2.2 Les topics dans la collection INEX 105
7.2.2.1 Format des topics 105
7.2.2.2 Préparation des requêtes 106
7.2.3 Jugement de pertinence 106
7.3 Première expérimentation 108
7.3.1 Procédure d'évaluation 108
7.3.2 Résultats 109
7.3.3 Discussion sur le résultat 109
7.4 Deuxième expérimentation 112
7.4.1 Les méthodes de validation 113
7.4.2 Procédure d'évaluation 114
7.4.3 Résultats 115
7.4.4 Discussion sur le résultat 117
7.5 Bilan 118

8 Conclusions et perspectives **121**
8.1 Conclusions générales 121
8.2 Perspectives 122

Résumé

Aujourd'hui, les bibliothèques numériques deviennent de plus en plus populaires. Ces bibliothèques fournissent plusieurs services pour leurs utilisateurs. Le service de recherche d'information est un service indispensable pour ces bibliothèques. La personnalisation de ce service pour mieux répondre aux exigences des utilisateurs est une approche qui attire beaucoup d'attention de la communauté scientifique. Plusieurs systèmes de recherche d'information personnalisée actuels ont choisi de re-trier les résultats d'un moteur de recherche en prenant en compte les similarités entre ces résultats et le profil utilisateur afin de rendre des résultats plus pertinents pour les utilisateurs. Cependant, la plupart de ces systèmes n'utilise que les approches basées sur le contenu textuel pour ce but. Dans nos travaux, nous proposons d'utiliser également des méthodes basées sur les citations telles que la méthode des co-citations et la méthode du couplage bibliographique pour calculer les similarités document-profil. Nous étudions la performance de la méthode des co-citations avec différentes bases de données bibliographiques. Nous utilisons également différentes fonctions de combinaison pour combiner les scores individuels. Les approches proposées ont été validées par des expérimentations sur une collection de test utilisée dans INEX 2005.

Chapitre 1

Introduction générale

1.1 Introduction à la recherche d'information

Depuis longtemps, nous avons besoin de méthodes efficaces pour faciliter l'accès aux informations qui nous sont nécessaires. Par exemple, dans les bibliothèques, les livres sont souvent organisés selon leurs thèmes pour que les utilisateurs puissent les localiser. Dans chaque livre il y a souvent une partie « index » qui contient des mots clés des chapitres/sections/pages. Ces méthodes d'organisation et d'indexation manuelles ont été utilisées pendant des milliers d'années. Néanmoins, avec le temps, le nombre de documents que nous publions augmente très rapidement. Dans la période de Renaissance, un savant pouvait maîtriser plusieurs domaines. Léonard de Vinci est un polymathe typique de cette époque. Il fut sculpteur, inventeur, scientifique, ingénieur, peintre, musicien ... Cependant, avec le temps les connaissances de l'homme deviennent si grandes qu'une personne doit passer plusieurs années pour être expert dans seulement un sous-domaine d'un métier ; on dit que Henri Poincaré, mathématicien français (1854-1912), fut la dernière personne qui a pu comprendre l'ensemble des mathématiques de son époque. Nous pouvons imaginer que les informations que l'homme produit augmentent avec une vitesse correspondante à cette croissance de connaissances. Surtout au vingtième siècle, avec le développement rapide des sciences, le nombre de documents scientifiques publiés chaque année augmente très rapidement. Les méthodes d'organisation et de recherche d'information traditionnelles ne sont plus appropriées. Nous avons donc besoin de systèmes qui peuvent nous aider à rechercher rapidement les documents nécessaires. C'est une prémisse importante pour la naissance des systèmes de recherche d'information modernes.

1.1.1 Une courte histoire

La recherche d'information est née assez tôt par rapport aux autres domaines de l'Informatique. En 1945, Vannevar Bush a écrit son article classique *Tel que nous pourrions penser*[1] [20] et publié dans *Atlantic Monthly*. Cet article décrit le système « Memex » qui contient plusieurs idées révolutionnaires comme l'accès aux grandes quantités de données en utilisant des nouveaux matériels pour stocker les documents, créer des liens entre les documents etc. Ce système est souvent considéré comme le premier modèle d'une bi-

[1] *As We May Think* en anglais.

bliothèque numérique et du Web. Cependant, dans ce système Bush n'a pas prévu de mécanisme pour indexer et rechercher automatiquement des documents. Dans les années 1950 et 1960, avec le développement de la technologie de l'ordinateur, les premiers systèmes de recherche d'information qui utilisent des ordinateurs ont été construits. Parmi eux, le système le plus notable est le système SMART développé par Gerard Salton et son équipe. Ce système a introduit plusieurs concepts importants comme le modèle vectoriel, la méthode de retour de pertinence, le regroupement de documents (*clustering*) etc.

En parallèle avec le système SMART, au début des années 1960, les travaux de Cleverdon [24] sont parmi des premiers travaux pour construire une collection de test qui permet d'évaluer des systèmes de recherche d'information. Plusieurs années après, l'idée d'utiliser des collections de test pour comparer des systèmes de recherche d'information est encore utilisé dans des campagnes d'évaluation comme TREC, INEX, ...

Les années 1970 est 1980 ont vu plusieurs changement dans le domaine. De nouveaux modèles de RI ont été proposés, la recherche d'information n'est plus limité à la recherche de documents textuels, les images et d'autres médias sont pris en compte. On commence également les recherches sur les systèmes de recherche d'information distribuée qui fournissent l'accès aux données réparties sur plusieurs machines qui peuvent être localisées à différentes positions géographiques.

L'histoire de la recherche d'information a été changée pour toujours avec l'apparition de la Toile (*World Wide Web* - WWW). La Toile est devenu le plus grand répertoire de données réparties jamais vu, et qui est accessible pour tout individu ayant une connexion Internet. Avec l'apparition du WWW, la recherche d'information a attiré de plus en plus d'attention des chercheurs ainsi que des entreprises. Pour répondre aux besoins de rechercher des informations sur le WWW, les moteurs de recherche sur le Web ont été créés. Ce sont les systèmes de recherche d'information à grande échelle qui peuvent indexer des milliards de pages Web et servir des millions d'utilisateurs chaque jour. Pour être efficients, ces systèmes utilisent des mécanismes de parallélisation et de distribution pour l'accès aux données. En 2005, le moteur de recherche Google a pu indexer huit milliards de pages Web.

1.1.2 Définition de la recherche d'information

Dans cette partie nous abordons quelques définitions sur la recherche d'information et les systèmes de recherche d'information :

- Définition de Baeza-Yates ([4], page 1) : la recherche d'information concerne la représentation, le stockage, l'organisation et l'accès aux éléments d'information[2].
- Définition de Kowalski ([72], page 2) : Kowalski définit un système de recherche d'information comme « un système qui est capable du stockage, de la recherche, et de la maintenance de l'information »[3]. Selon lui, les informations peuvent être des textes, des images, des séquences audio, des vidéos, et les autres objets multi-médias.

De plus, il y a des différences entre les systèmes de recherche d'information, les systèmes de recherche de données et les systèmes de filtrage d'information. On distingue ces systèmes comme suit.

[2] *Information retrieval deals with the representation, storage, organisation of, and access to information items.*

[3] *A system that is capable of storage, retrieval, and maintenance of information.*

Recherche d'information et recherche de données : Dans [4], Baeza-Yates fait une distinction claire entre les systèmes de recherche d'information (*information retrieval*) et les systèmes de recherche de données (*data retrieval*). Selon l'auteur, un système de recherche de données comme une base de données relationnelle concerne la recherche des objets qui satisfont exactement les conditions données (par exemple avec les langages de requêtes comme SQL). Par contre, un système de recherche d'information se base sur un modèle plus flou. De ce fait, il peut retourner les résultats qui ne satisfont pas totalement les besoins d'utilisateur où la notion de *pertinence* joue un rôle très important.

Recherche d'information et filtrage d'information : Le filtrage d'information (FI) est un processus pour sélectionner et délivrer l'information aux personnes concernées. Le filtrage d'information est basé sur l'utilisation de profils utilisateurs pour enlever/filtrer des données non pertinentes à partir d'un flux de données afin de ne présenter que des données utiles aux utilisateurs. Il y a beaucoup de points similaires entre un système de RI et un système de FI. Ils ont le même but de fournir des informations pertinentes aux utilisateurs. C'est pourquoi on a appelé ces deux types de systèmes d'information « deux faces d'un même pièce » [8]. Cependant, dans les systèmes de RI, les utilisateurs jouent un rôle actif de rechercher les informations tandis que dans les systèmes de FI ils jouent un rôle passif de recevoir les informations.

1.2 Problématique de la thèse

Actuellement, nous sommes confrontés à l'augmentation de la masse d'informations sur le Web aussi dans la vie professionnelle. Cette augmentation cause un problème que l'on peut imager par « recherche d'une aiguille dans une meule de foin » pour les personnes qui veulent chercher des informations sur le Web. Pour trouver des informations qui nous sont nécessaires, nous utilisons très souvent les moteurs de recherche. Cependant pour chaque requête, les moteurs de recherche populaires tels que Google[4] ou AltaVista[5] nous renvoient un grand nombre (des milliers ou des millions) de réponses. Beaucoup d'entre elles ne sont pas pertinentes. Une des raisons principales est que les requêtes des utilisateurs sont souvent courtes [134] et donc ambiguës. Par exemple, la même requête « java » peut être formulée par une personne qui s'intéresse au langage de programmation « java », et par une autre qui veut chercher des informations concernant une île en Indonésie. Cependant les moteurs de recherche renvoient le même résultat pour ces deux personnes. Même avec une plus longue requête comme « langage programmation java » ; nous ne savons pas quels types de document cet utilisateur veut chercher. Si c'est un(e) programmeur(e), peut-être il/elle s'intéresse aux documents techniques sur le langage Java, si c'est un(e) enseignant(e), peut-être il/elle s'intéresse aux tutoriels de Java pour ses cours.

A partir de cet exemple, nous pouvons voir que les intérêts de différents utilisateurs d'un même système de recherche d'information sont différents. De plus, une même personne peut avoir différents intérêts à différents moments. Un bon système de recherche d'information devrait tenir compte de ces différences pour satisfaire ses utilisateurs. C'est un besoin important non seulement avec les systèmes de recherche d'information mais

[4] http://www.google.com
[5] http://www.altavista.com

aussi avec les autres types de systèmes d'information. Ce besoin a entraîné l'apparition des *systèmes personnalisés* : ce sont des systèmes qui s'adaptent aux besoins d'information spécifiques de différents utilisateurs. Cette capacité est normalement basée sur leurs *profils*. D'une manière générale, nous pouvons définir un profil d'utilisateur comme un ensemble structuré d'informations qui décrit les intérêts et/ou les préférences de cet utilisateur.

Les systèmes de filtrage d'information qu'on a présenté au-dessus est un exemple de système personnalisé. Par contre, les systèmes de recherche d'information personnalisée sont les systèmes qui permettent de renvoyer les résultats de recherche adaptés aux profils utilisateurs en modifiant les requêtes des utilisateurs, en re-triant, ou en visualisant de manière personnalisée ces résultats.

Dans les années récentes, les bibliothèques numériques sont devenues de plus en plus importantes. Surtout en informatique, les bibliothèques numériques sont maintenant une part indispensable pour les chercheurs, les élèves, les professionnels etc. Nous pouvons lister des systèmes très connus dans ce domaine comme CiteSeer[6], la bibliothèque numérique d'ACM[7], la bibliothèque numérique d'IEEE *Xplore*[8] etc. Ces bibliothèques numériques fournissent aux utilisateurs un environnement de travail où ils peuvent chercher, consulter rapidement des documents en ligne. Cependant, les bibliothèques numériques, comme les autres systèmes d'information, ont besoin de services personnalisés afin de pouvoir mieux servir leurs utilisateurs. Comme le service de recherche d'information est parmi les plus importants services d'une bibliothèque numérique, la personnalisation de ce service est maintenant un besoin urgent. C'est dans ce contexte que notre travail se place.

1.3 Contribution de la thèse

Nos travaux se concentrent sur la RI personnalisée dans les bibliothèques numériques contenant des articles scientifiques. Chaque fois que l'utilisateur soumet une requête à un moteur de recherche d'une bibliothèque numérique, le système va filtrer une liste de n premiers documents. Puis il va calculer les similarités de ces documents avec le profil utilisateur. Ensuite, le système va calculer les nouveaux scores pour les documents dans la liste en combinant leurs scores originaux calculés par le moteur de recherche et les similarités document-profil. Ces nouveaux scores seront utilisés pour re-trier la liste de résultats.

Actuellement, la plupart des systèmes de RI personnalisés utilisent des approches basées sur le contenu textuel pour représenter les profils et représenter les documents et pour calculer la similarité entre eux. Par exemple, ils représentent les documents et les profils utilisateurs par le modèle vectoriel et calculent les similarités entre eux par la méthode du cosinus. L'originalité de notre travail est que, à côté de ces méthodes basées sur le contenu textuel, nous considérons aussi des méthodes basées sur les citations des articles et des approches hybrides (contenu textuel et citations) pour ces buts. Les méthodes basées sur les citations que nous utilisons dans nos travaux sont la méthode des co-citations et la méthode du couplage bibliographique. L'application de la méthode

[6] http://citeseer.ist.psu.edu/
[7] http://portal.acm.org/dl.cfm
[8] http://ieeexplore.ieee.org/

des co-citations demande d'utiliser des bases de données bibliographiques (ou bases de données de citations) pour connaître les relations bibliographiques entre les documents. Nous avons examiné plusieurs bases de données bibliographiques et nous avons décidé d'utiliser la base de données Thomson ISI qui est la base de données dominante pour les études bibliométriques. De plus, nous proposons d'utiliser le Web comme une base de données bibliographiques pour trouver les relations entre les articles scientifiques. La méthode proposée qui utilise le Web pour ce but s'appelle la méthode des co-citations sur le Web.

Un autre sujet que nous abordons dans nos travaux est le problème de la combinaison des scores (ce sont des scores originaux calculés par moteurs de recherche et les similarités entre documents-profils) afin d'obtenir des scores finaux pour re-trier les documents. Nous étudions différentes fonctions de combinaison dans nos travaux. Ce sont la combinaison produit, la combinaison linéaire, et la combinaison basée sur la théorie Dempster-Shafer.

Nous avons conduit des expérimentations pour valider nos approches. Comme nous n'avons pas une vraie bibliothèque numérique pour l'évaluation, nous utilisons donc une approche de simulation en utilisant une collection de test de l'INEX. Les résultats des expérimentations montrent que les approches proposées sont prometteuses et applicables pour les bibliothèques numériques.

1.4 Organisation du mémoire

Le mémoire est organisé comme suit. Après cette introduction générale, le chapitre 2 présente les bases de la recherche d'information : la vision du côté du système et du côté de l'utilisateur, les modèles connus, les méthodes d'évaluation des systèmes de RI, les outils etc. Le chapitre 3 présente les systèmes personnalisés. On aborde dans ce chapitre les sujets importants comme les modèles de représentation de profils utilisateurs, les méthodes d'acquisition des informations des utilisateurs, les techniques de constructions et de mise à jour de profils, les systèmes personnalisés de recherche et de filtrage d'information. Puis, dans le chapitre 4, on présente les bibliothèques numériques ainsi que le problème de personnalisation de ces bibliothèques. Le chapitre 5 concerne l'application des méthodes basées sur les citations dans la bibliométrie ; les méthodes que nous choisissons d'utiliser dans nos travaux comme la méthode des co-citations et la méthode du couplage bibliographique sont introduites dans ce chapitre. Ensuite, le chapitre 6 présente notre approche pour personnaliser les bibliothèques numériques qui contiennent des articles scientifiques. Ces propositions sont validées par des expérimentations dans le chapitre 7. Enfin, le dernier chapitre donne des conclusions et des perspectives.

Chapitre 2

Recherche d'information

Dans ce chapitre nous allons présenter de manière synthétique les notions et les connaissances de base de la RI. Les sujets abordés sont le processus de la RI du côté de l'utilisateur et du côté du système, les modèles connus, les aspects concernant l'évaluation des SRI, les méthodes des combinaisons de scores, et quelques outils utilisés dans notre équipe.

2.1 La recherche d'information : côté de l'utilisateur et côté du système

Dans cette section, nous décrivons les aspects fondamentaux du processus de la RI, du côté de l'utilisateur et du côté du système. Pour le côté de l'utilisateur, ce sont les problèmes concernant la définition du besoin d'information, la formulation de requête, l'évaluation des résultats de recherche. Pour le côté du système, ce sont les problèmes liés à la construction de la collection de documents et de l'index, le traitement des requêtes des utilisateurs.

2.1.1 Côté de l'utilisateur

Du côté de l'utilisateur, la RI est un processus en 3 étapes : 1) définir un besoin d'information 2) formuler la requête pour le SRI 3) recevoir et évaluer les résultats renvoyés par le SRI.

Tout d'abord, l'utilisateur doit définir son besoin d'information de la façon la plus claire possible. Cependant, ce n'est pas toujours facile car dans plusieurs cas l'utilisateur ne sait pas exactement ce qu'il veut. Par exemple, quand une personne veut lire « quelque chose » sur l'histoire de la France, il va taper quelques mots clés comme « Histoire France » sur Google. Puis après avoir lu quelques pages Web, suivi quelques liens, peut-être il aura l'intention de découvrir la biographie de Victor Hugo ou peut-être il voudra continuer de lire sur la révolution française à la fin de la dix-huitième siècle.

La formulation de la requête est aussi importante. L'exactitude des résultats renvoyés par le SRI est largement dépendante de cette étape. Actuellement, le modèle « sac de mots » est le modèle le plus utilisé. L'utilisateur doit sélectionner les termes qu'il trouve importants et les mettre dans sa requête. L'ordre de ces termes n'est pas important dans

ce modèle. Cependant, il y a des SRI qui demandent aux utilisateurs une connaissance plus avancée. Par exemple, dans les SRI utilisant le modèle booléen (cf. section 2.2.1), une connaissance sur les expressions booléennes est nécessaire. De plus, même dans les systèmes qui utilisent le modèle « sac de mots » comme Google[1], il y a des fonctionnalités de recherche avancées qui demande un apprentissage de la part des utilisateurs s'ils veulent exploiter de manière efficace le système : recherche dans les titres des documents[2], recherche d'une expression exacte en utilisant des guillemets, recherche selon le format des fichiers[3] etc. Une description plus détaillé sur le problème du besoin d'information à la formulation de requête est décrit dans [93].

Enfin, la dernière étape pour l'utilisateur est de regarder et d'évaluer les résultats renvoyés par le SRI. Si l'utilisateur n'est pas satisfait avec ces résultats, il peut reformuler sa requête ou faire un *retour de pertinence*[4]. Le retour de pertinence est un processus dans lequel l'utilisateur va spécifier au SRI les documents pertinents et non pertinents parmi les résultats renvoyés. Le SRI peut utiliser ces informations pour augmenter l'exactitude de ses résultats.

2.1.2 Côté du système

Du côté du système, la RI se compose de deux phases. La première phase est la phase de création de la collection de documents et la construction d'un index. La deuxième phase est la phase d'interaction avec l'utilisateur.

2.1.2.1 Première phase : création de la collection de documents et indexation

Création de la collection La création de la collection est le premier pas pour tous les SRI. La collection de documents peut-être construite manuellement ou automatiquement. Par exemple, dans les moteurs de recherche sur le Web, la collection de documents est collectée par les *crawler* en traversant récursivement les hyperliens trouvés à partir d'un ensemble de pages Web originales. Dans les campagnes d'évaluation, afin de comparer leurs SRI les participants utilisent les mêmes collections distribuées par les organisateurs.

Indexation Après avoir collecté une collection de documents, l'étape suivante est de construire un index. Quand l'utilisateur donne une requête à un système de RI, le système doit retrouver les documents qui contiennent ces termes et puis calculer la similarité document-requête pour le classement des documents renvoyés. La dernière étape requiert souvent de savoir les fréquences des termes dans les documents. Pour une petite collection de documents, le système peut traverser les documents séquentiellement chaque fois qu'il reçoit une requête de l'utilisateur. Cependant, ce n'est pas une bonne idée avec les grandes collections. C'est pourquoi on a besoin d'une solution pour accélérer cette procédure.

La procédure d'*indexation* a pour but d'associer les termes aux documents qui les contiennent dans une structure de données qui s'appelle un *index*. Par exemple, dans les

[1] Avec le moteur Google, le modèle est plutôt une liste de mots. En effet l'ordre des mots dans la requête a une influence sur la liste de résultats. Par exemple, la requête *histoire France* et la requête *France histoire* ne donnent pas la même liste de résultat.

[2] Commande *intitle* de Google.

[3] Commande *filetype* de Google.

[4] Cependant, plusieurs systèmes de RI ne supportent pas cette fonctionnalité.

livres, il y a souvent une partie « index » qui associe les mots aux chapitres/sections/pages où ils apparaissent. Il y a plusieurs types d'index. Cependant, le type le plus utilisé est l'*index inversé*. Les index inversés (ou fichiers inverses) ont été utilisés dans la RI et dans les systèmes de gestion de base de données depuis les années 1960 ([49], page 134). Un index inversé se compose d'une liste des termes et chaque terme dans cette liste est associé à une liste de *postings*[5]. Une liste de *posting* typique contient des paires $< doc_id, fq >$ où doc_id est l'identifiant du document et fq est le nombre d'occurrences du terme dans ce document. On peut modifier cette liste pour qu'il puisse contenir aussi les positions des occurrences du terme dans le document en utilisant la forme $< doc_id, fq, pos_1, pos_2, ..., pos_{fq} >$ où pos_i indique la position de la i-ème occurrence du terme dans le document doc_id. Avec cette nouvelle représentation, on peut rechercher d'une expression exacte dans les documents[6]. Les entrées dans cette liste peuvent contenir également les poids assignés aux termes dans les documents. Pour construire l'index, le système traverse toute la collection et traite tous les termes. L'indexation peut réduire nettement le nombre des entrées/sorties pour répondre aux requêtes des utilisateurs [49]. Chaque fois que le système de RI reçoit une requête, il consulte l'index pour retrouver les listes de *posting* correspondantes et il utilise les informations de ces listes pour calculer les similarités requête-documents afin de trier les documents. Plusieurs techniques d'implémentation des fichiers inverses sont présentés dans [153]. L'index inversé est illustré dans la figure 2.1.

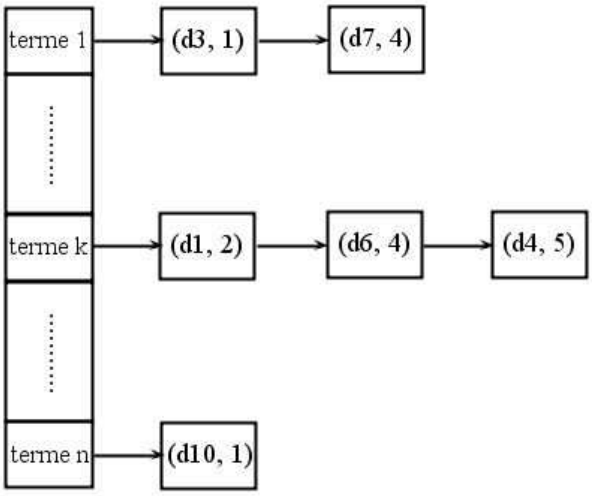

FIG. 2.1 – Index inverse avec les listes de posting

Plusieurs opérations peuvent être appliquées à la collection pour réduire la taille des index et augmenter la performance du système. Nous allons présenter quelques techniques connues ci-dessous.

Élimination des mots vides Les mots qui apparaissent trop souvent dans les documents ont peu de valeur pour la recherche d'information. Ces mots sont appelés des *mots vides* (e.g. *le*, *la*, *il*... en français). Ces mots peuvent être supprimés dans le processus

[5] *posting list* en anglais.
[6] *phrase search* en anglais.

d'indexation en utilisant une liste des mots vides. Cependant, l'élimination des mots vides peut diminuer le taux de rappel. Par exemple, on ne peut pas chercher une expression exacte qui contient des mots vides comme « Monsieur le Président de la République » avec cette approche.

Lemmatisation La lemmatisation d'un mot est un processus qui transforme un mot en sa forme de base. Par exemple, les mots « commune », « communs », « communes » peuvent être transformés en forme de base « commun ». Si on veut favoriser le rappel, on peut appliquer la lemmatisation sur la collection des documents. Par exemple, c'est fréquent que l'utilisateur donne un mot dans sa requête mais il n'y a que des variantes de ce mot dans la collection. Cependant, les bénéfices de la lemmatisation sont encore sujets à débat ([4], page 168).

Parmi les algorithmes de lemmatisation connus, nous pouvons citer celui de Porter pour la langue anglaise [106] ou celui de Savoy pour la langue française [115].

Indexation manuelle et automatique Dans l'indexation automatique, le système choisit automatiquement les termes à inclure dans l'index. Dans le cas le plus simple, tous les termes dans le document sont utilisés pour l'indexation. Par contre, l'indexation manuelle est un processus dans lequel la sélection des termes d'index des documents est faite par une (des) personne(s). L'indexation manuelle/automatique peut également s'appuyer sur l'utilisation des vocabulaires libres ou contrôlés[7].

2.1.2.2 Deuxième phase : Interaction avec l'utilisateur

La deuxième phase est la phase d'interaction avec l'utilisateur qui se compose des étapes ci-dessous :

Réception de la requête de l'utilisateur Tout d'abord, le système doit recevoir la requête de l'utilisateur et la transformer en utilisant des opérations correspondant au modèle de recherche (par exemple, calculer les poids des termes de la requête dans le modèle vectoriel).

Calcul de la similarité entre la requête et les documents Après avoir analysé la requête de l'utilisateur, le système va interroger son index pour calculer la similarité entre la requête et les documents en utilisant une *fonction de correspondance*. Cette similarité est une mesure de la pertinence des documents par rapport à la requête utilisateur.

Retour des résultats Quand le système a fini le calcul de la similarité entre la requête et les documents, il va renvoyer les résultats aux utilisateurs, normalement en présentant une liste des documents. Cette liste est souvent ordonnée selon le niveau de pertinence des documents. Ce peut être une liste des hyperliens pour les moteurs de recherche sur le Web ou les numéros d'identification/les noms des documents pour les SRI locaux.

[7] Le vocabulaire contrôlé correspond aux listes d'autorité ou aux thésaurus de descripteurs, pour lesquels les descripteurs utilisés sont des termes préférentiels appartenant à une liste fixée, et d'usage parfaitement codifié ([81], page 77).

Quelques systèmes peuvent fournir des informations supplémentaires à côté de cette liste. Par exemple Google peut suggérer des mots apparentés à la requête ou proposer une correction d'orthographe avec les mots qu'il pense incorrects. Le processus général de la RI est illustré dans la figure 2.2.

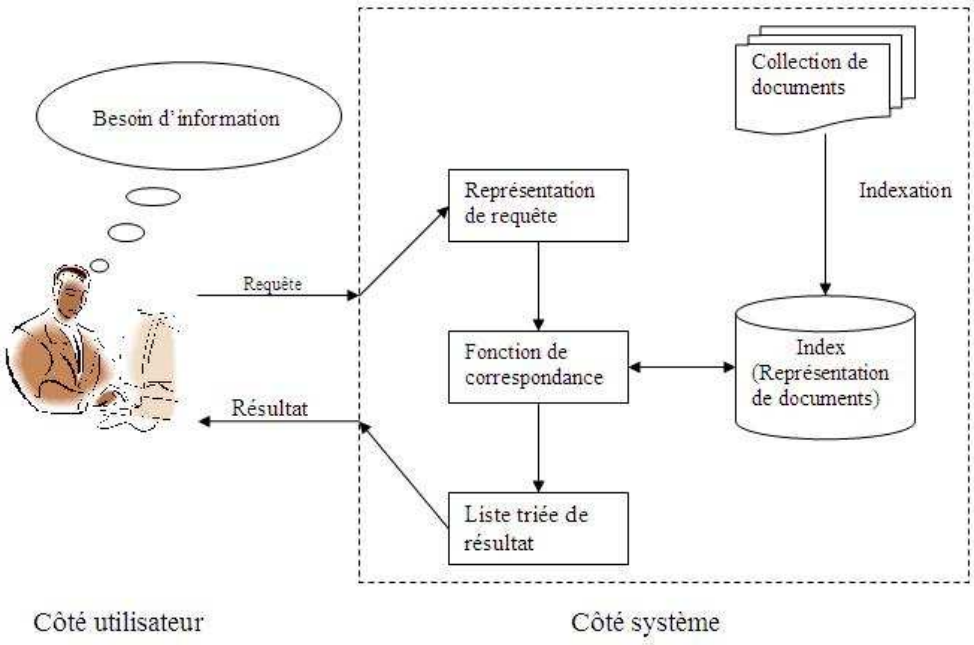

FIG. 2.2 – Processus de la RI : côté utilisateur et côté système.

Dans cette section, nous venons de présenter les aspects fondamentaux du processus de la RI du côté de l'utilisateur et du côté du système. Dans la section suivante nous allons présenter quelques modèles connus pour représenter les documents et les requêtes et pour calculer la similarité entre eux. Les modèles présentés sont le modèle booléen, le modèle vectoriel et les modèles basés sur les approches probabilistes.

2.2 Quelques modèles connus de la recherche d'information

2.2.1 Modèle booléen

Le modèle booléen [4] est basé sur la théorie des ensembles et sur l'algèbre Boole. Dans ce modèle, les documents peuvent être représentés comme un vecteur de termes. Les poids des termes ont une valeur binaire zéro ou un qui signifie si le terme existe dans le document ou non.

Les requêtes sont formulées avec les opérateurs booléens classiques AND (ET), OR (OU) et NOT (NON). Par exemple, si l'on veut chercher les documents qui traitent des entreprises Google et Yahoo et ne traite pas de Microsoft, la requête d'utilisateur peut être formulé comme suite : (Google AND Yahoo) AND (NOT Microsoft). Dans le modèle booléen, la similarité entre une requête et un document peut avoir seulement une des deux valeurs 0 ou 1, qui correspondent à un jugement *non pertinent* ou *pertinent*.

L'avantage du modèle booléen est qu'il est simple et qu'il peut fournir une expression « riche » du besoin d'information de l'utilisateur. Cependant, comme la similarité requête-document ne peut avoir qu'une valeur binaire 0 ou 1, alors il n'y a pas de classement relatif entre les documents retournés. De plus, la construction des requête booléennes complexes est difficile pour la plupart des utilisateurs.

2.2.2 Modèle vectoriel

Dans le modèle vectoriel, les documents sont représentés sous la forme de vecteurs des poids des termes. Un document d est représenté par un vecteur $\vec{d} = (w_{t1,d}, w_{t2,d}, ..., w_{tn,d})$ dans lequel $t1...tn$ sont tous les termes dans le vocabulaire du système et $w_{t1,d}, ..., w_{tn,d}$ sont des poids de ces termes dans le document d. Similairement, une requête est représentée par un vecteur $\vec{q} = (w_{t1,q}, w_{t2,q}, ..., w_{tn,q})$. Cependant, pour l'utilisateur une requête est simplement un *sac de mots*.

La similarité requête-document peut être calculée par plusieurs formules. La formule la plus utilisée est la formule du cosinus :

$$sim(q,d) = \frac{\sum_{i=1}^{n} w_{ti,q} w_{ti,d}}{\sqrt{\sum_{i=1}^{n} w_{ti,q}^2 \sum_{i=1}^{n} w_{ti,d}^2}}$$

A côté de la formule du cosinus, d'autres mesures sont utilisées comme les mesures de Dice ou de Jaccard ([72], page 154). Dans la formule du cosinus, la similarité requête-document est représentée par un nombre réel dans l'intervalle de 0 à 1. La valeur de 0 signifie que la requête et le document sont entièrement différents tandis qu'une valeur de 1 signifie qu'ils ont la même expression. La stratégie de pondération la plus utilisée pour les documents dans le modèle vectoriel est le tf-idf (de l'anglais *term frequency* (fréquence du terme) et *inverse document frequency* (fréquence inverse de document)). Dans cette stratégie, deux mesures sont considérées : le tf qui décrit l'importance du terme dans le document et le idf qui décrit l'importance du terme dans le corpus. Le tf est calculé par le nombre d'occurrences du terme dans le document et le idf est calculé par le logarithme de la fraction de nombre total de document dans le corpus par rapport au nombre de documents qui contiennent le terme. Le poids final d'un terme peut être calculé en multipliant ces deux mesures. Il existe plusieurs variantes de cette stratégie. Pour la requête, Salton et Buckley [4] suggèrent la formule suivante pour pondérer les termes dans la requête :

$$w_{t,q} = \left(0.5 + \frac{0.5 f_{t,q}}{max_l f_{l,q}}\right) \log \frac{N}{n_i}$$

Dans cette formule, $w_{t,q}$ est le poids du terme t dans la requête, $f_{t,q}$ est la fréquence du terme t dans la requête.

Les avantages de ce modèle est qu'il peut donner une classement des documents, ce modèle est assez simple par rapport à sa performance. Cependant, l'utilisation d'un sac de mots clés pour formuler la requête ne permet pas beaucoup d'expressivité dans le besoin d'information contrairement au modèle booléen.

2.2.3 Modèles basés sur les approches probabilistes

Dans cette section nous allons présenter quelques modèles basés sur les approches probabilistes : modèle probabiliste et modèle de langue.

2.2.3.1 Modèle probabiliste

Le modèle probabiliste a été proposé par S. E. Robertson et K. Spark Jones en 1976 [4]. Avec une requête q et un document d_j, le modèle probabiliste va estimer la probabilité que ce document soit pertinent. Supposons que R est l'ensemble des documents pertinents, et $\overline{R}$ est l'ensemble des documents non pertinents. $P(R|\overrightarrow{d_j})$ est la probabilité que le document d_j soit pertinent à la requête q et $P(\overline{R}|\overrightarrow{d_j})$ est la probabilité que d_j ne le soit pas. Alors la similarité entre le document d_j et la requête q est calculé par :

$$sim(q, d_j) = \frac{P(R|\overrightarrow{d_j})}{P(\overline{R}|\overrightarrow{d_j})}$$

En appliquant la loi de Bayes, on a :

$$sim(q, d_j) = \frac{P(\overrightarrow{d_j}|R) \times P(R)}{P(\overrightarrow{d_j}|\overline{R}) \times P(\overline{R})}$$

$P(\overrightarrow{d_j}|R)$ est la probabilité qu'un document d_j soit sélectionné aléatoirement à partir de l'ensemble R et $P(R)$ est la probabilité qu'un document pris à partir de la collection entière soit pertinent. Similairement, $P(\overrightarrow{d_j}|\overline{R})$ est la probabilité qu'un document d_j soit sélectionné aléatoirement à partir de l'ensemble $\overline{R}$ et $P(\overline{R})$ est la probabilité qu'un document pris à partir de la collection entière ne soit pas pertinent. Comme $P(R)$ et $P(\overline{R})$ ont les mêmes valeurs pour tous les documents, alors on peut les enlever :

$$sim(q, d_j) \propto \frac{P(\overrightarrow{d_j}|R)}{P(\overrightarrow{d_j}|\overline{R})}$$

Les étapes suivantes de ce calcul sont basées sur l'hypothèque de l'indépendance des termes dans l'index. Finalement, en prenant le logarithme, la similarité $sim(q, d_j)$ peut être calculée par :

$$sim(q, d_j) \propto \sum_t w_{t,q} \times w_{t,j} \times \left(\log \frac{P(t|R)}{1 - P(t|R)} + \log \frac{1 - P(t|\overline{R})}{P(t|\overline{R})} \right)$$

$P(t|R)$ est la probabilité que le terme t soit présent dans un document sélectionné aléatoirement à partir de R. $P(t|\overline{R})$ est la probabilité que le terme t soit présent dans un document sélectionné aléatoirement à partir de $\overline{R}$.

Dans le moteur de recherche Zettair [104], une implémentation de ce modèle utilisant la pondération Okapi BM25 a été faite. La similarité $sim(q, d_j)$ est calculée comme suit :

$$sim(q, d_j) = \sum_{t \in q \bigcap d_j} w_t \times \frac{(k_1 + 1) f_{d,t}}{K + f_{d,t}} \times \frac{(k_3 + 1) f_{q,t}}{k_3 + f_{q,t}}$$

Dans cette formule, $w_t = \log_e \left(\frac{N_D - N_{D_t} + 0.5}{N_{D_t} + 0.5} \right)$; N_D est le nombre total des documents dans la collection et N_{D_t} est le nombre des documents qui contiennent le terme t ; $K = k_1 \times \left((1-b) + \frac{b.W_d}{W_{AL}} \right)$ - W_d est la longueur du document et W_{AL} est la longueur moyenne des documents ; k_1, b, k_3 sont des constants. $f_{d,t}$ et $f_{q,t}$ sont respectivement les fréquences de t dans le document et dans la requête.

2.2.3.2 Modèle de langue

L'idée de l'approche par modèle de langue est d'estimer un modèle de langue pour chaque document, et de classer les documents selon la probabilité que la requête soit générée à partir de ce modèle [131, 152]. Cette approche a été utilisée dans les domaines de traitement des langages naturels, reconnaissance de la parole et récemment dans la RI. Supposons que nous avons une requête $q = q_1 q_2 ... q_n$ et un document $d = d_1 d_2 ... d_m$ (q_i et d_i sont les termes de la requête et du document), nous voulons estimer la probabilité que d génère q : $p(d|q)$. En appliquant la loi de Bayes et en supprimant une constante qui est indépendante des documents, nous avons :

$$p(d|q) \propto p(q|d)p(d)$$

$p(q|d)$ est la vraisemblance (*likelihood*) de la requête par rapport au document, $p(d)$ est la probabilité a priori que d soit pertinent par rapport à toutes les requêtes q. Dans le cas le plus simple, on suppose que $p(d)$ est uniforme et n'a pas d'influence sur le classement des document. Dans ce cas, il faut seulement calculer la probabilité $p(q|d)$. Si l'on suppose que les occurrences des mots sont indépendantes, alors :

$$p(q|d) = \prod_i p(q_i|d)$$

Maintenant on a besoin de deux modèles, un modèle $p_s(q_i|d)$ pour les termes qui existent dans le document d et un autre modèle $p_u(q_i|d)$ pour les termes qui n'existent pas dans le document d. Supposons que $p_u(q_i|d) = \alpha_d p(q_i|C)$ dans lequel α_d est une constante qui est dépendante du document d et $p(q_i|C)$ est le modèle de langue de la collection ; $c(q_i; d)$ est le nombre d'occurrences du terme q_i dans le document d ; n est la longueur de la requête. En appliquant la fonction logarithme dans la formule ci-dessus, on a :

$$\begin{aligned}
\log p(q|d) &= \sum_i \log p(q_i|d) \\
&= \sum_{i:c(q_i;d)>0} \log p_s(q_i|d) + \sum_{i:c(q_i;d)=0} \log p_u(q_i|d) \\
&= \sum_{i:c(q_i;d)>0} \log \frac{p_s(q_i|d)}{p_u(q_i|d)} + \sum_i \log p_u(q_i|d) \\
&= \sum_{i:c(q_i;d)>0} \log \frac{p_s(q_i|d)}{\alpha_d p(q_i|C)} + n \log \alpha_d + \sum_i \log p(q_i|C)
\end{aligned}$$

Dans cette formule, $\sum_i \log p(q_i|C)$ est indépendant du document d et alors peut être supprimé dans le classement des résultats. Dans la méthode *Dirichlet-smoothing*, pour estimer la probabilité $\log p(w|d)$ d'un mot w, on utilise la formule suivante :

$$p(w|d) = \frac{c(w;d) + \mu p(w|C)}{\sum_w c(w;d) + \mu}$$

Dans la formule ci-dessus, μ est une constante. Dans le travail de Zhai et Lafferty [152], la valeur optimale de μ est autour de 2000. Le paramètre α_d est estimé de la façon suivante :

$$\alpha_d = \frac{\mu}{\sum_w c(w;d) + \mu}$$

Le reste de ce modèle est d'estimer $p(w|C)$. Dans l'implémentation de ce modèle dans le moteur de recherche Zettair [104], $p(w|C)$ est estimé par : $p(w|C) = \frac{N_{D_w}}{N_D}$; N_D est le nombre de documents dans la collection et N_{D_w} est le nombre des documents qui contiennent w[8]. En appliquant $p(w|C)$ et $p(w|d)$ dans la formule de calcul du $\log p(q|d)$ au-dessus, la similarité entre un document et une requête est calculée de la façon suivante :

$$sim(q,d) = n \log \alpha_d + \sum_{q_i : c(q_i;d)>0} \log \left(\frac{N_D c(q_i;d)}{\mu N_{D_{q_i}}} + 1 \right)$$

Dans la formule ci-dessus, N_D est le nombre de documents dans la collection et $N_{D_{q_i}}$ est le nombre des documents qui contiennent q_i.

2.3 Évaluation

Dès les premiers jours de la RI, l'évaluation des systèmes et des méthodes de RI a attiré l'attention des chercheurs dans ce domaine. Les expérimentations Cranfield de Clerverdon et al. [24] au début des années 1960 sont souvent cités comme les premières évaluations dans le domaine de RI [56]. Les composants des expérimentations Cranfield sont : une collection de documents, un ensemble des requêtes, et un ensemble de jugements de pertinence qui sont des documents jugés comme pertinents par rapport avec chaque requête. Aujourd'hui, ce sont encore les composants principaux dans les collections de test des campagnes d'évaluation.

Dans les sections suivantes, nous allons aborder les aspects concernant l'évaluation des systèmes de recherche d'information : les collections de test, la notion de pertinence, la méthode de *pooling*, les mesures de performance. Nous allons présenter également deux campagnes d'évaluations connues : TREC et INEX.

2.3.1 Collections de test

Pour avoir une comparaison juste, les expérimentations devraient être exécutées sur la même collection de test en utilisant les même mesures de performance. Une collection de test se compose d'un ensemble de documents, un ensemble de besoins d'information

[8] Dans cet article, les auteurs n'écrivent pas explicitement cette estimation.

<top>
<num> Number : 451
<title> What is a Bengals cat ?
<desc> Description :
Provide information on the Bengal cat breed.
<narr> Narrative :
Item should include any information on the Bengal cat breed, including description, origin, characteristics, breeding program, names of breeders and catteries carrying bengals. References which discuss bengal clubs only are not relevant. Discussions of bengal tigers are not relevant.
</top>

FIG. 2.3 – Exemple d'un besoin d'information.

(*topic*) avec des jugements de pertinence pour ces besoins d'informations. Ces jugements de pertinence déterminent quels sont les documents pertinents dans la collection par rapport à chaque besoin d'information. Les documents peuvent être jugés avec une mesure binaire (pertinent/non pertinent) ou avec des mesures multivalués.

La figure 2.3 est un exemple d'un *topic* dans TREC (cf. la section 2.3.5.1).

2.3.2 Notion de pertinence

La notion de « pertinence » est une des notions les plus importantes dans la RI. Néanmoins, il faut dire que c'est aussi une notion très difficile à définir. Généralement, la pertinence d'une réponse d'un SRI est une mesure de la satisfaction de cette réponse avec la requête de l'utilisateur. Du côté du système, c'est une valeur précise assignée pour un document dans la liste de réponses. Cette valeur est calculée par la fonction de correspondance (ou fonction d'appariement) du système. Ce peut être une valeur binaire (pertinent/non pertinent) dans le cas du modèle booléen, ou une valeur réelle dans l'intervalle $[0, 1]$ dans le modèle cosinus. Du côté de l'utilisateur, c'est une notion subjective et floue. Pour une même requête, un document pertinent pour une personne peut être non pertinent pour une autre personne. Même pour une personne, un document pertinent à un moment donné peut être non pertinent à un autre moment parce que le critère de pertinence de cette personne a été changée.

2.3.2.1 Jugements de pertinence

Les collections de test utilisées dans les campagnes d'évaluation sont souvent accompagnées par des topics représentant des besoins d'information et des jugements de pertinence de ces topics, c'est-à-dire les évaluations/estimations des documents concernant ces topics. Normalement, ces évaluations sont faites par des participants de la campagne. Cependant, chaque campagne a ses propres critères d'évaluation. Par exemple, dans TREC[9], un document est jugé comme pertinent si une part de ce document est pertinent, malgré sa

[9] http://trec.nist.gov/data/reljudge_eng.html

qid	iter	docno	rel
451	0	WTX003-B26-240	0
451	0	WTX003-B26-249	1
451	0	WTX003-B26-252	0
451	0	WTX003-B26-263	0
451	0	WTX003-B31-203	0
451	0	WTX004-B07-355	0

FIG. 2.4 – Extrait d'un fichier de jugement de pertinence de TREC.

taille par rapport au reste du document[10]. La figure 2.4 est un extrait d'un fichier de jugement de pertinence de TREC. Le *qid* est le numéro de la requête, *docno* est le numéro de document, *rel* est la pertinence du document, *iter* est un champ ignoré.

Dans la campagne d'évaluation INEX (cf. la section 2.3.5.2), les éléments retournés par les SRI ne sont pas seulement des documents entiers mais aussi des composants XML. De ce fait les jugements de pertinence ont été fait au niveau des éléments et aussi au niveau de documents. INEX utilise deux mesures pour calculer la pertinence d'un élément : i) l'*exhaustivité (e)* qui décrit à quel niveau cet élément traite le topic de la requête et ii) la *spécificité (s)* qui décrit à quel niveau cet élément se concentre sur le topic de la requête.

INEX propose plusieurs fonctions pour combiner ces deux mesures en une valeur de pertinence unique [63]. Nous listons à titre d'exemples quelques fonctions ci-dessous :

$$quant_{strict}(e,s) = \begin{cases} 1 & si\ e = 2\ et\ s = 1 \\ 0 & sinon \end{cases}$$

$$quant_{gent}(e,s) = e \cdot s$$

$$quant_{gentLifted}(e,s) = (e+1) \cdot s$$

2.3.3 Méthode *pooling*

Idéalement, tous les documents dans la collection devraient être examinés pour vérifier leur pertinence. Cependant, en réalité c'est irréalisable avec des grandes collections. Alors on utilise la méthode *pooling*. Dans la campagne d'évaluation TREC, on a utilisé cette méthode pour construire des jugements de pertinence [53] :

- Pour chaque topic, les participant renvoient une liste de n premiers résultats fournis par leurs systèmes de RI. Typiquement, $n = 100$.
- Les listes de résultats des différents participants sont fusionnées pour former un *pool*.
- Les documents dupliqués sont enlevés du *pool*.
- Enfin, chaque document dans le *pool* est examiné et jugé par les participants. Les documents qui ne sont pas dans le *pool* sont considérés comme non pertinents.

Bien qu'il y ait des critiques sur la méthode *pooling*, cette méthode a montré son utilité et elle est largement utilisée pour construire des collections de test.

[10] *A document is judged relevant if any piece of it is relevant (regardless of how small the piece is in relation to the rest of the document).*

2.3.4 Mesures de performance

2.3.4.1 Précision et rappel

Précision : pour une requête, la précision est définie comme le nombre des documents pertinents retournés par un moteur de recherche sur le nombre total des documents qu'il a renvoyés :

$$Précision = \frac{Nombre\ des\ documents\ pertinents\ renvoyés}{Nombre\ total\ des\ documents\ renvoyés}$$

Rappel : pour une requête, le rappel est défini comme le nombre des documents pertinents renvoyés par rapport au nombre total des documents pertinents dans la collection :

$$Rappel = \frac{Nombre\ des\ documents\ pertinents\ renvoyés}{Nombre\ total\ des\ documents\ pertinents}$$

2.3.4.2 La courbe précision/rappel

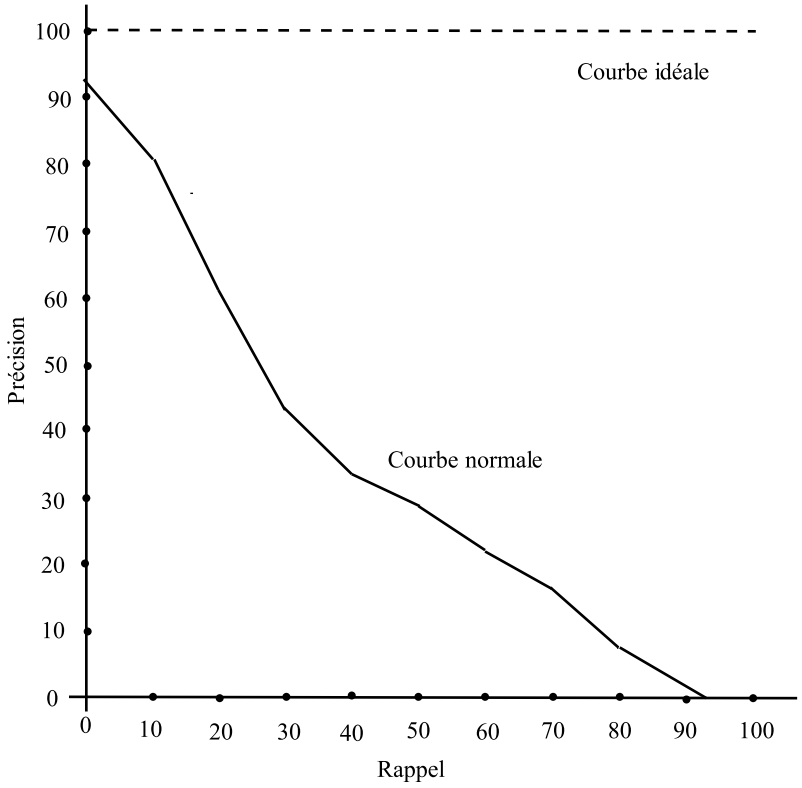

FIG. 2.5 – Précision à 11 points de rappel standards.

La courbe précision/rappel donne des précisions du système aux différents niveaux de rappel pour chaque requête et la moyenne pour l'ensemble des requête. Dans la campagne d'évaluation de TREC on utilise la courbe précision/rappel à 11 points qui correspondent aux niveaux de rappel de 0%, 10%, 20%,..., 100%. Dans cette courbe, l'interpolation est appliquée. Le principe de l'interpolation est le suivant : soient i et j deux niveaux de

rappel différents, alors la précision interpolée $PI(i) = max\ P(j)\ (j \geq i)$. La figure 2.5 illustre les courbes précision/rappel.

2.3.4.3 Précision à n documents

Cette mesure est similaire à la mesure de précision ci-dessus. La précision à n documents est le nombre de documents pertinents parmi les n premiers documents retournés par le systèmes de RI.

$$Précision\ à\ n = \frac{Nombre\ des\ documents\ pertinents\ parmi\ n\ premiers\ documents}{n}$$

Habituellement on s'intéresse aux valeurs 5, 10, 15, 20 ... 1000 pour n.

2.3.4.4 Précision moyenne (MAP)

La précision moyenne (*Mean Average Precision* en anglais) est largement utilisée dans la campagne d'évaluation de TREC. Pour une seule requête, c'est la moyenne des précisions à chaque document pertinent dans la liste ordonnée des résultats. Pour l'ensemble des requêtes, c'est la valeur moyenne des précisions moyennes de chaque requête individuelle. Supposons que $Q = q_1, q_2 ... q_n$ est l'ensemble des requêtes ; m_j est le nombre des documents pertinents du besoin d'information q_j ; $Précision(R_{jk})$ est la précision de la requête q_j lorsque k documents pertinents ont été retrouvés (si ce document n'est pas retrouvé, cette valeur sera 0), alors :

$$Précision\ moyenne = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Précision(R_{jk})$$

2.3.5 Campagnes d'évaluation

Les campagnes d'évaluation sont organisés afin d'encourager les recherches dans le domaine et de comparer les différents systèmes de RI. Dans cette section nous présentons deux campagnes d'évaluation importantes : TREC et INEX.

2.3.5.1 TREC

La campagne d'évaluation TREC[11] est une série d'ateliers (*workshops*) soutenue conjointement par le NIST[12] et par le Département de la Défense des États-Unis depuis 1992. Selon une déclaration sur son site Web[13], le but de TREC est de :

- encourager les recherches de la RI basée sur les grandes collections de test.
- augmenter la communication entre l'industrie, le monde académique, et le gouvernement en créant un forum ouvert pour les échanges des idées de recherche.

[11] Text REtrieval Conference
[12] National Institute of Standards and Technology
[13] http://trec.nist.gov/overview.html

- accélérer le transfert de technologie à partir des laboratoires de recherche aux produits commerciaux en démontrant les améliorations substantielles des méthodologies de la RI dans les problèmes de la vie réelle.
- augmenter la disponibilité des techniques d'évaluation appropriées pour l'utilisation par l'industrie et le monde académique, incluant le développement des nouvelles techniques d'évaluation qui sont plus applicables aux systèmes actuels.

Un workshop de TREC se compose de plusieurs tâches différentes. Chaque tâche correspond à une mission spécifique. En 2008, les tâches suivantes sont organisées[14] :

- *Blog* : cette tâche a pour but d'explorer les comportements de recherche d'information dans la blogosphère.
- *Enterprise* : cette tâche a pour but d'étudier la recherche d'information dans les entreprises.
- *Legal* : le but de cette tâche est de développer des technologies de RI pour satisfaire les besoins d'information des avocats et des juristes.
- *Million Query* : cette tâche a pour but de tester l'hypothèse qu'une collection de test construite à partir des topics qui sont jugés de façon très incomplète est un meilleur outil qu'une collection construite avec la méthode traditionnelle du pooling.
- *Relevance Feedback* : cette tâche a pour but de fournir un *framework* pour explorer l'effet de différents facteurs sur le succès de la stratégie de retour de pertinence.

NIST fournit des ensembles de documents de test et des besoins d'information pour les participants. Les participants exécutent les requêtes sur leurs systèmes de RI et retournent au NIST leur liste des premiers documents retrouvés (*top-ranked*). NIST fait ensuite le *pool* des résultats, juge les documents, et évalue les résultats des participants (cf. la section 2.3.3). Finalement, il organise un *workshop* pour les participants pour qu'ils puissent partager leurs expériences.

2.3.5.2 INEX

La campagne d'évaluation INEX[15] [85] se concentre sur l'évaluation des SRI dans les collections de documents XML. Comme TREC, INEX fournit une série des collections de test, chacun contient un ensemble de documents, un ensemble de besoins d'information et des jugements de pertinence. En 2008, INEX organise les tâches suivantes :

- *Ad Hoc* : c'est la tâche principale de l'INEX qui a le plus grand nombre de participants. Les participants vont comparer leurs SRI en retrouvant les éléments XML pertinents et aussi les passages pertinents par rapport aux topics communs. En 2008 cette tâche utilise la collection Wikipédia qui a été utilisé dans l'INEX 2007.
- *Book* : cette tâche a pour but d'étudier la RI sur une collection de livres construite avec la technologie ROC (reconnaissance optique de caractères). Cette collection utilise une représentation XML pour la structure des livres. Il y a plusieurs sous-tâches concernant différents aspects : les stratégies de classement des livres, l'interface utilisateur et le comportement des utilisateurs, reconnaissance des structures sophistiquées des livres que la ROC ne peut pas faire etc.
- *Efficiency* : cette tâche a pour but d'évaluer l'efficacité et l'efficience des approches de recherche des éléments XML.

[14] http://trec.nist.gov/tracks.html
[15] http://inex.is.informatik.uni-duisburg.de/

- *Entity Ranking* : cette tâche a pour but de comparer et évaluer les techniques de recherche des entités dans les documents XML au lieu des documents entiers ou des éléments. Un entité peut être, par exemple, une date, une personne, un lieu etc.
- *Interactive* : la tâche étudie le comportement des utilisateurs quand ils interagissent avec les éléments des documents XML et d'étudier et développer des approches efficaces pour la recherche des éléments dans les environnements orientés vers les utilisateurs (*user-based environments*).
- *Question Answering* : cette tâche se concentre sur la capacité des SRI travaillant sur des documents XML pour répondre aux besoins d'information précis du monde réel qui sont formulés par des questions en langage naturel.
- *Link-The-Wiki* : cette tâche a pour but d'évaluer des méthodes de découvert des liens entre les documents.
- *XML Mining* : L'objectif de la tâche est de développer des méthodes d'apprentissage automatique pour la fouille de données structurées et pour évaluer ces méthodes. La tâche se concentre sur la classification et le regroupement des documents XML.

2.4 Méthodes de combinaison pour la recherche d'information

Dans la recherche d'information, dans plusieurs cas nous devons combiner plusieurs scores calculés par différentes méthodes afin de pouvoir donner un score final pour chaque document. Le score final d'un document d est calculé à partir des scores individuel en utilisant une fonction de combinaison f comme suivante :

$$score_final(d) = f(score_1(d), score_2(d), ..., score_n(d))$$

Les méthodes de combinaison jouent un rôle très important sur les résultats finaux [80]. Nous allons présenter quelques les fonctions de combinaisons connues pour combiner les scores : combinaison produit, combinaison linéaire, et combinaison basée sur la théorie de Dempster-Shafer. Ce sont également les fonctions de combinaisons que nous allons utiliser dans nos travaux.

2.4.1 Combinaison produit

Dans cette méthode, le score final est calculé en multipliant les scores individuels.

$$score_final(d) = \prod_i score_i(d) \tag{2.1}$$

L'avantage de cette méthode est qu'il est simple, et il n'y a pas de paramètres à configurer comme dans les méthodes de combinaison linéaire ou Dempster-Shafer. Dans [125], les auteurs utilisent une fonction produit pour combiner différents scores individuels afin de re-trier les résultats renvoyés par un moteur de recherche.

2.4.2 Combinaison linéaire

Dans cette méthode, le score final est la somme des scores individuels normalisés et pondérés par des coefficients.

$$score_final(d) = \sum_i \beta_i \times score_i(d) \tag{2.2}$$

Les scores individuels sont normalisés en divisant ces scores par la valeur maximale correspondante. Les méthodes de combinaison linéaire ont été utilisées depuis longtemps pour combiner différentes sources d'information. Par exemple, dans [40], Fox et al. proposent deux algorithmes nommés $CombSUM$ et $ComMNZ$ pour combiner les résultats de différents moteurs (comme les méta-moteurs de recherche). Pour un document d, $CombSUM$ est la somme des scores individuels correspondant aux différentes méthodes (sans coefficients de pondération). $CombMNZ = CombSUM \times k$ où k est le nombre de méthodes qui ont retourné d. La méthode $CombMNZ$ favorise les documents retournés par plusieurs méthodes.

Dans la recherche d'information, normalement on combine les scores calculés par différentes approches/sources pour obtenir un score final qui va servir à trier les documents. Cependant, dans quelques cas, on utilise les rangs de classement d'un document dans les listes de réponses de plusieurs méthodes et puis on combine ces rangs afin de re-trier les documents au lieu de combiner les scores. Par exemple, dans [133], les auteurs présentent une approche pour re-trier les résultats du moteur de recherche Google en utilisant le profil utilisateur. Le rang final d'un document est calculé en combinant deux rangs : le rang original calculé par Google et le rang conceptuel calculé par la similarité de ce document avec le profil utilisateur. Le rang final est ensuite calculé en utilisant la formule suivante :

$$Rang_Final = \alpha \times Rang_Conceptuel + (1 - \alpha) \times Rang_Google$$

2.4.3 Combinaison basée sur la théorie Dempster-Shafer

La troisième méthode de combinaison que nous étudions est la méthode de combinaison basée sur la théorie Dempster-Shafer. Nous choisissons la théorie Dempster-Shafer dans nos travaux parce que c'est une théorie bien connue pour combiner des sources d'évidences. Cette théorie a d'abord été développée par Dempster [31] et puis élargie par Shafer [122]. Dans cette théorie, un domaine de problème est représenté par un cadre de discernement (*frame of discernment*) qui est un ensemble exclusif et exhaustif Θ des états (ou hypothèses/étiquettes/propositions). L'ensemble des disjonctions des éléments de Θ qui contient toutes les sous-parties de Θ (y compris l'ensemble vide) est noté 2^Θ. On définit une fonction de masse m (*basic probability assignment – bpa*) qui assigne une croyance à ces états : $2^\Theta \rightarrow [0, 1]$, vérifiant :

$$m(\emptyset) = 0 \tag{2.3}$$

et

$$\sum_{A \subset \Theta} m(A) = 1 \tag{2.4}$$

Un des principaux avantages de la théorie de Dempster-Shafer est qu'elle permet une représentation explicite de l'ignorance en utilisant $m(\Theta)$ qui est la croyance assigné à tout le cadre de discernement ; $m(\Theta) = 1$ représente une ignorance totale d'une source d'évidences. Cette ignorance est calculée de manière suivante :

$$m(\Theta) = 1 - \sum_{A \subset \Theta, A \neq \Theta} m(A) \tag{2.5}$$

Il faut noter que la quantité $m(A)$ est la croyance assignée exactement pour l'état/l'hypothèse A, ce n'est pas la *croyance totale* assignée pour A. Pour obtenir la croyance totale pour A, nous utilisons une fonction de croyance (*belief function*) notée $Bel : 2^{\Theta} \rightarrow [0, 1]$, et définie par :

$$Bel(A) = \sum_{B \subset A} m(B) \tag{2.6}$$

Une fonction de croyance satisfait les conditions suivantes :

$$Bel(\emptyset) = 0$$
$$Bel(\Theta) = 1$$

Une autre fonction importante dans la théorie de Dempster-Shafer est la fonction de plausibilité Pl (*plausibility function*), définie par :

$$Pl(A) = \sum_{B | B \cap A \neq \emptyset} m(B) \tag{2.7}$$

Les fonctions de croyance et de plausibilité peuvent être considérées comme les bornes inférieures et supérieures de probabilité de A, et la probabilité exacte est dans l'intervalle de ces deux valeurs [120].

Un autre concept important dans la théorie de Dempster-Shafer est *discount*. C'est une opération d'affaiblissement. Si une source d'évidence n'est pas totalement fiable et on a seulement un *degré de confiance* (*degree of trust*) $1 - \alpha$ à cette source ($0 \leq \alpha \leq 1$), nous pouvons réduire la croyance de cette source de manière suivante :

$$m^{\alpha}(A) = \begin{cases} (1 - \alpha) m(A) & si A \neq \Theta \\ (1 - \alpha) m(\Theta) + \alpha & si A = \Theta \end{cases} \tag{2.8}$$

Règle de combinaison de Dempster Supposons que m_1 et m_2 sont deux fonctions de masse dans le même cadre de discernement Θ. Alors quand on combine ces deux fonctions, la fonction combiné est définie de manière suivante :

$$m_{12}(A) = \frac{\sum_{B \cap C = A} m_1(B) m_2(C)}{1 - \sum_{B \cap C = \emptyset} m_1(B) m_2(C)} \tag{2.9}$$

Dans la formule 2.9, le dénominateur $1 - \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$ est un facteur de normalisation qui garantit que le nouveau bpa m_{12} satisfait la condition 2.4. Cependant,

si nous ne devons combiner que deux sources et nous nous intéressons seulement au classement relatif basé sur le score combiné, alors nous pouvons ignorer ce facteur.

La théorie de Dempster-Shafer a été largement utilisée pour plusieurs applications depuis sa publication ; par exemple, pour la recherche multimédia [149], pour la fusion de données sur le Web [140], pour la prédiction de mobilité dans les réseaux adhoc [30] etc. Dans [120] les auteurs fournissent une liste (non exhaustive mais assez large) des applications de cette théorie dans plusieurs domaines.

2.5 Outils

Dans cette section, nous présentons quelques outils pour la RI utilisés au sein de nos équipes.

2.5.1 SMART

Le système de RI SMART[16] a été développé depuis 1964 à l'Université Cornell sous la direction de Gerard Salton. Le développement de ce système dans des versions publiques a été arrêté au début des années 1990. C'est un système pionnier dans le domaine de la RI qui a introduit plusieurs concepts importants comme le modèle vectoriel, la méthode de retour de pertinence, le regroupement de documents (*clustering*) etc. Cependant, ce système n'est pas très facile à utiliser. La dernière version disponible remonte à l'année 1992.

2.5.2 Lucy/Zettair

Le système Zettair[17] (connu auparavant sous le nom Lucy) est un moteur de recherche développé à l'Université RMIT (Australie). La dernière version de Zettair est la version 0.9.3 datée du 29 septembre 2006 et le système est écrit en langage C. Ce système support plusieurs modèles : Okapi BM25, cosinus, *pivoted cosine*, Hawkapi et le modèle de langue Dirichlet. Le modèle par défaut utilisé dans Zettair est le modèle de langue Dirichlet. Parmi les avantages de ce système, il est rapide, assez facile à comprendre et très facile à utiliser.

2.5.3 trec_eval

Trec_eval[18] est l'outil standard pour l'évaluation des résultats des systèmes de RI dans les campagnes d'évaluation de TREC. Trec_eval calcule plusieurs mesures populaires dans la communauté de RI comme la précision/rappel, la précision à n documents,la précision moyenne (MAP), etc.

[16] ftp ://ftp.cs.cornell.edu/pub/smart/
[17] http://www.seg.rmit.edu.au/zettair/
[18] http://trec.nist.gov/trec_eval/

2.6 Bilan

Dans ce chapitre, nous avons présenté les aspects de base de la RI. Cependant, les systèmes de RI que nous présentons dans ce chapitre sont les systèmes de RI traditionnels. Ces systèmes renvoient le même résultats pour différents utilisateurs. Comme nous avons présenté brièvement dans le premier chapitre, une approche intéressante pour améliorer l'exactitude des systèmes de RI est de prendre en compte les besoins d'information spécifiques de différents utilisateurs. Un système de RI ayant cette capacité s'appelle un système de RI personnalisée. Dans le chapitre suivant, nous abordons les systèmes personnalisés actuels. Nous présenterons différents aspects concernant la construction de profils utilisateurs pour représenter les intérêts et les préférences des utilisateurs et l'utilisation de ces profils utilisateurs dans les systèmes personnalisés.

Chapitre 3

Systèmes Personnalisés Actuels

Dans ce chapitre, nous présentons les travaux de personnalisation actuels, en particulier les systèmes de recherche et de filtrage d'information personnalisés. Parce que les profils utilisateurs jouent un rôle déterminant dans les systèmes personnalisés, nous allons présenter d'abord leurs différents aspects : les modèles de représentations, les méthodes d'acquisition d'information, et les techniques de construction et de mise à jour de ces profils. Nous abordons par la suite l'utilisation de ces profils dans différentes applications de recherche d'information et de filtrage d'information.

3.1 Profils utilisateurs

Dans cette section, nous abordons les sujets les plus importants concernant la représentation et la construction de profils utilisateurs.

3.1.1 Définition de profils utilisateurs

Dans [132], l'auteur divise les profils utilisateurs dans deux groupes : les profils qui représentent les préférences de l'utilisateur et ceux qui représentent ses intérêts. Par exemple, dans [146] les auteurs présentent une architecture pour la recherche d'information sur le Web qui utilise des profils représentant les préférences de l'utilisateur. Ils définissent : « un profil [utilisateur] est un ensemble de préférences concernant le comportement d'un moteur de recherche ainsi que les contraintes sur les résultats qu'il présente à l'utilisateur »[1]. Les préférences et contraintes dans ce cas concernent le choix de présentation de résultats (affichage au maximum de 25 documents par page), le format de documents (PDF, HTML ...), le format structurel de documents (abstracts ou document entier) etc.

Les profils représentant les intérêts de l'utilisateur sont plus répandus que ceux qui représentent les préférences. De plus, ils concernent plus d'aspects, aux niveaux de représentation, de construction, et d'utilisation. Dans les parties suivantes, nous allons aborder notamment les travaux utilisant ce type de profils.

[1] *A (user) profile consists of a set of preferences with regard to behavior of a search engine as well constraints on the results it presents to the user.*

Profils à court-terme et profils à long-terme Nous avons différents intérêts à différents moments. Par exemple, un chercheur dans le domaine de la recherche d'information s'intéresse généralement aux documents parlant de la recherche d'information. Cependant, à un moment spécifique, il veut chercher des informations concernant les prix de billets d'avion pour aller à une conférence. Ainsi la « recherche d'information » est son intérêt à long-terme et le « prix des billets d'avion » est son intérêt à court-terme.

A partir de cet exemple, nous pouvons dire que les notions de profils à court-terme et profils à long-terme [135, 48, 10] décrivent respectivement les préférences éphémères et persistantes d'utilisateurs. D'autres auteurs [66] utilisent ces notions pour décrire les préférences spécifiques et générales d'utilisateurs bien que « éphémère » ne correspond pas toujours avec « spécifique » et « persistant » ne correspond pas toujours avec « général ». Normalement, le profil à court-terme est plus important que le profil à long-terme parce qu'il décrit mieux l'intérêt de l'utilisateur au moment présent.

3.1.2 Modèles de représentation de profils utilisateurs

Dans cette partie, nous nous attardons sur les modèles de représentation des profils utilisateurs. Les natures des systèmes personnalisés sont très différents. Comme dans les systèmes de RI, chaque type de système personnalisé demande un modèle de représentation différent du profil pour s'adapter à ses buts et son fonctionnement. Il existe donc plusieurs modèles tels que le modèle vectoriel, le modèle à base d'ontologie, le modèle multidimensionnel, etc. Dans la suite nous décrivons les modèles les plus connus.

3.1.2.1 Modèle vectoriel

Dans le modèle vectoriel, chaque profil utilisateur se compose d'un ou plusieurs vecteurs de termes. Chaque terme est associé avec une valeur. Ce modèle est le modèle le plus utilisé. En utilisant ce modèle, nous pouvons calculer facilement la similarité cosinus d'un document quelconque avec le profil d'un utilisateur si le document est aussi représenté par un vecteur de termes. Parmi les systèmes utilisant ce modèle, nous pouvons citer [5, 126, 84, 121, 22].

Dans un cas particulier de ce modèle, les poids des termes dans le vecteur n'est plus un nombre réel mais une valeur booléenne [10, 102]. Cette valeur représente la présence ou non d'un terme dans le profil.

3.1.2.2 Modèle sémantique à base d'ontologie

Un autre modèle populaire de représentation de profils utilisateurs est le modèle sémantique à base d'ontologie. Dans ce modèle, un profil est une hiérarchie de concepts pondérés. Chaque nœud dans la hiérarchie est un concept. Le poids attaché avec un concept représente l'intérêt de l'utilisateur avec ce concept. Ce poids peut être changé pour mettre à jour l'intérêt de l'utilisateur. De plus, chaque concept est souvent représenté par un vecteur de termes pondérés. Le poids attaché avec un concept représente l'intérêt de l'utilisateur tandis que ce vecteur représente le « contenu » de ce concept. Ce vecteur peut être construit à partir d'un ensemble de documents assignés à ce concept.

Il existe plusieurs répertoires Web tels que celles de *ODP* [2] ou *Yahoo* [3] qui peuvent être utilisés comme hiérarchie de concepts. Dans ce cas, un vecteur de termes pondérés qui représente un concept peut être construits à partir des documents (pages Web) indexés sous ce concept [45] (ou l'ensemble des documents indexés sous ce concept plus les documents indexés sous ses sous-concepts [125]). Un exemple de profil utilisateur représenté par le modèle à base d'ontologie avec le processus de mise à jour est illustrée dans la figure 3.1. Parmi les systèmes qui utilisent ce modèle de représentation, nous pouvons citer [133, 21, 91, 45, 125].

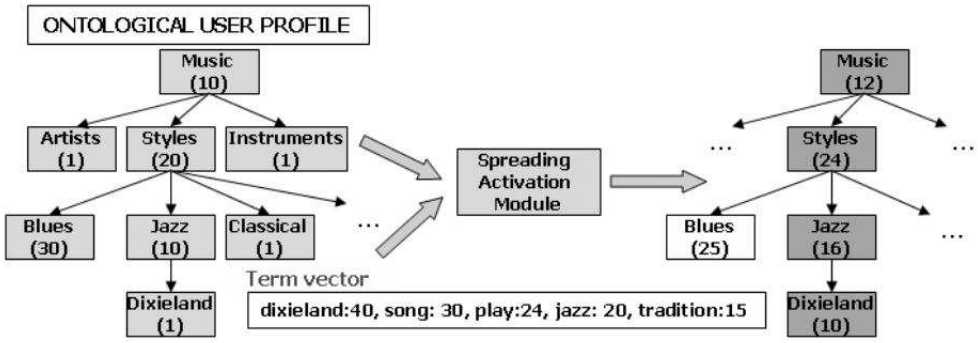

FIG. 3.1 – Exemple du profil utilisateur représenté par le modèle d'ontologie avec le processus de mise à jour des poids des concepts [125].

L'utilisation du modèle ontologie pour représenter les profils utilisateurs peut aider à mieux connaître les intérêts des utilisateurs par rapport au modèle vectoriel. Par exemple, si on représente un profil utilisateur par un vecteur de termes pondérés et dans ce vecteur contient un terme « instrument », on ne sait pas exactement si ce terme concerne des instruments musiques ou les autres types d'instruments. Cependant, si on utilise le modèle ontologie (comme illustré dans la figure 3.1), on peut facilement régler ces ambiguïtés : dans cet exemple, le concept « instruments » est un concept « fils » du concept « music », alors il s'agit des instruments musiques. De plus, un autre avantage du modèle ontologie est qu'on peut propager la valeur d'intérêt d'un concept vers les autres concepts reliés (par exemple, son concept « père ») afin de trouver des nouveaux centres d'intérêt [92].

3.1.2.3 Modèle multidimensionnel

Le travail de Amato et al. [1] est un des premiers travaux vers la construction d'un modèle multidimensionnel pour représenter des profils utilisateurs. Cette représentation donne une description globale des utilisateurs en prenant en compte plusieurs dimensions différentes. Dans leur article, les informations concernant les utilisateurs peuvent être classifiées dans cinq catégories différentes, chaque catégorie est une dimension : i) la *catégorie des données personnelles* contient des données d'identification personnelles de l'utilisateur (nom, date de naissance, contact ...) ii) la *catégorie de recherche* contient des préférences et des restrictions sur les documents que l'utilisateur est en train de rechercher iii) la

[2] http://dmoz.org
[3] http://dir.yahoo.com/

	Item 1	Item 2	Item 3	Item 4
Utilisateur 1	-	6	9	2
Utilisateur 2	2	-	6	8
Utilisateur 3	8	2	-	-

FIG. 3.2 – Matrice Utilisateurs-Items.

catégorie de livraison sont des spécifications concernant le mode de livraison des informations trouvées (courriel, fax, Web, temps de livraison etc.) iv) la *catégorie de données des actions* contient des enregistrements sur l'interaction de l'utilisateur avec le système de recherche et les données de navigation (pages Web visités, documents lus, jugements de pertinence etc.) v) enfin la *catégorie de données de sécurité* est une collection des préférences de l'utilisateur concernant des conditions d'accès aux informations du profil utilisateur.

Dans l'article [151], les auteurs proposent un autre modèle multidimensionnel pour représenter des profils utilisateurs. Dans ce modèle, le contenu d'un profil se compose de trois dimensions (ou catégories) principales : i) la *catégorie des préférences* concernant des préférences de l'utilisateur (domaine d'intérêt, préférences de recherche d'information) ii) la *catégorie des données personnelles* permettant d'identifier l'utilisateur (son identité et sa profession) et iii) la *catégorie des données d'environnement* contenant des informations sur l'environnement de recherche de l'utilisateur (l'emplacement géographique, la configuration logicielle et matérielle). Chaque dimension peut se décomposer en sous-dimensions qui sont plus détaillées. Les auteurs proposent aussi la possibilité d'intégrer ce profil dans la phase de re-formulation de la requête, dans la phase de réduction de l'espace de recherche pour restreindre l'espace de recherche aux documents qui correspondent le mieux aux besoins de l'utilisateur, dans la phase d'appariement document-requête, ou dans la phase de présentation des résultats.

3.1.2.4 Autres modèles de représentations

A côté des modèles ci-dessus, il existe d'autres modèles de représentation de profils utilisateurs.

Modèle matrice Utilisateurs-Items : Un autre modèle souvent utilisé dans les systèmes de recommandation collaborative (cf. la section 3.2.4) est la représentation par une matrice Utilisateurs-Items (voir figure 3.2). Chaque ligne de la matrice représente un utilisateur et chaque colonne représente un item. Une cellule $[i, j]$ de la matrice contient le vote de l'utilisateur i pour l'item j (ou rien si utilisateur n'a pas voté cet item). Dans ce modèle, le profil d'un utilisateur est considéré comme un vecteur des votes de cet utilisateur pour les items.

Hiérarchie des intérêts : Dans l'article [66], les auteurs utilisent un modèle appelé « hiérarchie des intérêts de l'utilisateur » dans lequel un profil est une hiérarchie, chaque nœud dans la hiérarchie est un ensemble de mots représentant l'intérêt d'utilisateur. Les nœuds feuilles représentent des intérêts spécifiques des utilisateurs tandis que les nœuds

plus près de la racine représentent des intérêts généraux. Cependant, ce modèle n'est pas basé sur une hiérarchie de concept spécifique comme dans le modèle ontologie.

Propriétés démographiques : Dans ce modèle, un profil utilisateur représente un ensemble de « propriétés démographiques » [73]. Les utilisateurs doivent répondre à une liste de questions prédéfinies et le système utilise ces informations pour trouver les groupes (*demographic cluster*) (cf. la section 3.1.3.1) qui représentent ces utilisateurs. Les données démographiques sont les informations concernant les activités, les modes de vie, etc. des personnes comme les chaînes de télévision préférées, le sport pratiqué.

3.1.3 Acquisition d'information

Dans cette partie, nous abordons les méthodes d'acquisition des informations nécessaires pour construire les profils utilisateurs. Actuellement, il existe trois méthodes principales pour acquérir ces informations : la méthode d'acquisition implicite, la méthode d'acquisition explicite et la méthode hybride.

3.1.3.1 Acquisition explicite

Dans cette méthode, les utilisateurs doivent fournir explicitement les informations nécessaires au système. Une telle approche a été utilisée depuis longtemps dans plusieurs systèmes de recherche d'information (par exemple la méthode de retour de pertinence de Rocchio [113]). Il existe plusieurs formes d'acquisition explicite :

Entrée directe par les utilisateurs : Les utilisateurs doivent entrer leurs intérêts sous la forme de mots clés. Malgré sa simplicité apparente, c'est une approche qui demande beaucoup d'effort du côté des utilisateurs pour clarifier leurs intérêts.

Classement « binaire » par les utilisateurs : Dans les systèmes [22, 121], les utilisateurs doivent classer les items (pages Web, livres ...) dans deux classes « intéressants » ou « inintéressants ».

Vote : Dans les systèmes [102, 5, 10, 111, 123], les utilisateurs doivent donner des votes explicites pour mesurer leurs intérêts avec les items qu'ils doivent évaluer. Par exemple, dans le système Syskill & Webert [102], les pages Web sont évaluées selon trois niveaux *hot, lukewarm, cold*. Dans le système FAB [5], des notes de -3 à +3 sont données aux pages Web. Dans le système Ringo [123], les artistes et les albums de musique sont évalués par des notes de 1 à 7.

Stéréotype : Quelques systèmes [73] demandent aux utilisateurs de répondre à un ensemble de questions prédéfinies et puis ils utilisent ces informations pour déduire leurs profils.

L'avantage des approches explicites est que les profils ainsi construits sont plus précis que ceux obtenus par acquisition implicite. Cependant, il existe également plusieurs inconvénients à cette approche [95] :

- Les jugements de pertinence de l'utilisateur pour les items sont dépendants des changements de son besoin d'information. Par exemple, un utilisateur qui a déjà lu deux documents pertinents à son besoin d'information attribuera un vote plus faible à un troisième qui traite du même sujet parce que son niveau d'exigence a augmenté.
- Les échelles numériques peuvent être inadéquates pour décrire les réactions des hommes pour les items.
- Les utilisateurs sont souvent hésitants à donner des évaluations si elles ne sont pas directement reliées à leurs besoins immédiats.

A cause de ces inconvénients, plusieurs systèmes ont choisi d'acquérir implicitement des informations nécessaires pour le profilage. Ces techniques d'acquisition implicite sont présentées dans la partie ci-après.

3.1.3.2 Acquisition implicite

Dans cette approche, le système acquiert les informations nécessaires pour construire les profils utilisateurs implicitement en surveillant ou traçant leurs actions. Par exemple, si un utilisateur sauvegarde un document sur son disque, il est probablement intéressé par ce document et il a l'intention de l'utiliser dans l'avenir. De manière similaire, si l'utilisateur utilise beaucoup de temps pour lire un document Web et/ou suit beaucoup de liens dans ce document, c'est aussi une preuve qu'il s'intéresse à ce document. Dans [84], le temps utilisé et les liens suivis dans un document Web ont été utilisés comme des preuves de l'intérêt de l'utilisateur. Une autre approche très populaire est de regarder les historiques de navigation Web de l'utilisateur pour trouver les types de documents qui sont intéressants pour lui [135]. Dans [61], les auteurs présentent une étude concernant l'utilisation des clics de souris (*clickthrough*) comme les informations implicites.

Dans [23], les auteurs ont utilisé un navigateur qui peut enregistrer les actions des utilisateurs (temps utilisé pour lire une page, nombre de clics et temps utilisé pour déplacer le souris et l'ascenseur, les actions sur quatre touches *Page Up*, *Page Down*, *Up Arrow*, et *Down Arrow*) et utilise ces informations pour déduire l'intérêt des utilisateurs pour les pages Web. Ils ont montré que le temps utilisé pour lire une page, le nombre d'utilisation de *l'ascenseur* et la combinaison entre eux ont une forte corrélation avec l'acquisition explicite, tandis que le temps et le nombre de clics sur la souris est inefficace pour inférer l'intérêt de l'utilisateur. Une fois que le système trouve que l'utilisateur s'intéresse à un document Web, ce document sera utilisé pour construire son profil (par exemple, en utilisant les modèles représentés dans la section 3.1.2 et les techniques représentées dans la section 3.1.4).

Oard et Kim [98] ont présenté un cadre général pour modéliser les actions possibles des utilisateurs. Selon les auteurs, les actions des utilisateurs sont classées en quatre catégories générales. La catégorie *examen* contient trois actions *regarder*, *écouter* et *sélectionner*. Normalement, les systèmes d'information fournissent de brèves descriptions sur leurs objets ; lorsque que l'utilisateur choisit *d'examiner* un objet c'est peut-être une preuve d'intérêt pour cet objet. La catégorie *maintien* aborde l'intention d'utiliser un objet dans l'avenir. Elle contient des actions tels que *imprimer*, *sauvegarder dans les signets*, *sauvegarder*, *supprimer*, *acheter* et *souscrire*. La catégorie *référence* a pour but d'établir d'une forme de relation entre deux objets et contient des actions de *copier-coller*, *transférer*, *répondre*, *lier*

et *citer*. Enfin, la catégorie *annotation* sont des actions qui ajoutent intentionnellement de la valeur dans un objet. Les actions dans cette catégorie sont *annoter*, *voter*, *publier* et *organiser*. Kelly et Teevan [64] élargissent la classification ci-dessus en ajoutant quelques actions *faire défiler*, *chercher*, *interroger*, *naviguer* et *email* dans les catégories *examen* et *maintien*. De plus, ils ajoutent la catégorie *création* qui décrit l'action de création des nouveaux objets (par exemple quand on écrit un article).

L'avantage de cette approche est qu'elle ne requiert pas beaucoup d'efforts des utilisateurs dans le processus de profilage. Cependant, dans plusieurs cas sa précision n'est pas aussi bonne que celle de la méthode d'acquisition explicite. Elle est plus difficile à interpréter et potentiellement « bruyante » [61].

3.1.3.3 Approche hybride

Quelques systèmes ont choisi de combiner les deux précédentes méthodes pour obtenir une meilleure performance. Dans [121], un profil utilisateur contient des termes pondérés, chaque fois qu'un document est jugé pertinent, le poids d'un terme dans son profil est mis à jour en utilisant les paramètres suivants : le vote explicite, le temps utilisé pour lire ce document, le nombre de liens suivis et l'action *sauvegarder dans les signets* de ce document.

Dans [95], les auteurs font une liste de 37 systèmes qui utilisent différentes approches d'acquisitions d'information d'utilisateurs. Parmi eux, 20 systèmes utilisent des approches explicites, 8 systèmes utilisent des approches implicites, les approches hybrides sont utilisées par 9 autres systèmes.

3.1.4 Techniques de construction et de mise à jour de profils utilisateurs

Dans cette partie, nous allons aborder les techniques de construction de profils utilisateurs à partir des informations collectées. Bien que dans quelques cas, il ne faille pas appliquer une technique spécifique pour construire des profils à partir des données collectées (par exemple, pour les profils représentés par des matrices utilisateurs-items (cf. la section 3.1.2.4) ou dans la méthode d'acquisition explicite par l'entrée directe de l'utilisateur (cf. la section 3.1.3)). Cependant, dans la plupart de cas une technique de construction de profils utilisateurs est nécessaire.

3.1.4.1 TF-IDF

Dans les systèmes personnalisés, la méthode la plus utilisée pour construire des profils utilisateurs est la technique tf-idf et ses variantes. C'est une technique issue du domaine de la recherche d'information pour la pondération de termes dans le modèle vectoriel. Ce n'est pas une surprise parce que le modèle vectoriel est le modèle le plus utilisé pour représenter des profils utilisateurs. Quelques systèmes utilisant cette approche sont [22, 10]. Dans [22], un profil se compose de N vecteurs de termes pondérés. Chaque vecteur représente un domaine d'intérêt de l'utilisateur. Chaque fois qu'un document est jugé pertinent par l'utilisateur, le système construit le vecteur tf-idf de ce document, après cette étape on obtient un ensemble de N + 1 vecteurs pondérés (N vecteurs profils et 1 nouveau

vecteur de document). Puis le système calcule la similarité cosinus entre chaque paire de vecteurs dans cet ensemble et combine les deux vecteurs les plus similaires. Dans cette approche, le profil est mis à jour incrémentalement chaque jour. Dans le système NewsDude [10], le profil à court-terme d'utilisateur se compose de plusieurs documents pour lesquels il a voté, chaque document est représenté par son vecteur tf-idf. Chaque fois qu'un nouveau document arrive, le système va d'abord extraire le vecteur tf-idf de ce document. Puis il compare la similarité cosinus de ce document avec les autres documents dans le profil : les documents ayant une similarité avec le nouveau document plus élevée qu'un seuil prédéfini seront filtrés (ou sélectionnés). La prévision de vote du nouveau document sera la valeur moyenne de tous les votes que l'utilisateur a effectué pour les documents filtrés. Le système utilise cette prévision de vote pour décider de recommander le nouveau document à l'utilisateur ou non. Parmi les autres systèmes qui utilisent la méthode tf-idf nous pouvons citer [5, 84]

3.1.4.2 Les méthodes de classification

Les méthodes de classification sont les méthodes d'apprentissage supervisé qui sont en charge d'affecter les éléments dans les groupes existants. Ces groupes contiennent déjà des exemples positifs qui sont nécessaires pour les algorithmes d'apprentissage supervisé. Par exemple, dans les travaux de Gauch et al. [45], le profil utilisateur est représenté par le modèle ontologie. Le poids d'un concept représente l'intérêt de l'utilisateur avec ce concept. Pour calculer ces poids, ils utilisent les pages Web que l'utilisateur a lu dans le passé. Ces pages Web sont enregistrées dans le répertoire cache du navigateur. Pour chaque page Web (d_k), ils calculent la similarité de cette page avec tous les concepts dans l'ontologie. La similarité est calculée en utilisant la mesure cosinus et en prenant en compte également le temps que l'utilisateur a utilisé pour lire cette page Web ainsi que la longueur de la page. La page Web est classifiée dans cinq concepts (c_j) qui sont les plus similaires avec cette page Web. Les poids de ces concepts seront augmentés par les valeurs retournées par le classificateur. En bref, cet ajustement est calculé de la manière ci-dessous :

$$similarité(d_k, c_j) = facteur_temps_longeur \times similarité_cosinus(d_k, c_j)$$

Dans cette formule, le $facteur_temps_longeur$ est calculé par une des quatre formules suivantes : $\frac{temps}{longueur}$, $\log \frac{temps}{longueur}$, $\log \frac{temps}{\log longueur}$, $\log \frac{temps}{\log(\log longueur)}$, où $temps$ est le temps (en seconde) que l'utilisateur a utilisé pour lire le document et $longueur$ est le longueur (en octet) de la page Web.

Parmi les autres travaux qui utilisent les méthodes de classification pour construire et mettre à jour les profils utilisateurs nous pouvons citer à titre d'exemple [133].

3.1.4.3 Les méthodes de *clustering*

Les méthodes de *clustering* (ou regroupement) sont les méthodes d'apprentissage non supervisé qui sont en charge d'attribuer les éléments dans des groupes qui n'existent pas à l'avance. Par exemple, dans [66] (cf. la section 3.1.2.4), les auteurs utilisent une hiérarchie d'intérêts de l'utilisateur pour représenter le profil utilisateur. Dans cette hiérarchie, les nœuds feuilles représentent les intérêts spécifiques et les nœuds internes représentent les

intérêts plus généraux. Pour construire cette hiérarchie, ils utilisent un algorithme de regroupement hiérarchique. L'entrée de cet algorithme est un ensemble de pages Web que l'utilisateur a visitées. Ils enlèvent les mots vides et puis font la lemmatisation sur les mots de ces pages Web. Puis ils calculent les similarités entre toutes les paires de mots en utilisant différentes fonctions de similarité (*AEMI, AEMI-SP, Jaccard* etc.). Ensuite, l'algorithme va grouper récursivement ces mots dans les sous-groupes, chaque sous-groupe représente un nœud dans la hiérarchie d'intérêts.

Parmi les autres travaux qui utilisent les méthodes de *clustering* pour construire des profils utilisateurs, nous pouvons citer à titre d'exemple [130, 126, 48].

3.1.4.4 Combinaison de plusieurs méthodes

Dans plusieurs cas, on combine des méthodes différentes pour apprendre les profils des utilisateurs. Par exemple, d'abord un algorithme de classification ou de *clustering* est utilisé pour classifier/regrouper l'ensemble des documents intéressants pour l'utilisateur dans les catégories différentes ; puis on calcule les vecteurs tf-idf de ces catégories, chaque vecteur représente un domaine d'intérêt de l'utilisateur. Dans le système ARCH [126], un utilisateur possède plusieurs vecteurs profils. Pour construire les vecteurs profils d'un utilisateur, d'abord le système collecte un ensemble de documents que l'utilisateur a trouvé intéressants. Puis il va appliquer un algorithme de *clustering* pour regrouper ces documents dans les catégories différentes. Chaque catégorie représente un domaine d'intérêt (ou un vecteur profil) de l'utilisateur. Après le système calcule le vecteur tf-idf des documents dans chaque catégorie. Enfin, les vecteurs centroïdes sont calculés dans ces catégories. Chaque vecteur centroïde représente un profil de l'utilisateur.

Dans [92, 91], le profil d'un utilisateur est représenté par le modèle ontologie (cf. la section 3.1.2.2), l'intérêt de l'utilisateur avec un sujet dans l'ontologie est calculé en utilisant les paramètres suivants :

- le nombre des articles dans ce sujet qui ont été lus par l'utilisateur.
- le nombre des articles recommandés (cf. 3.2.4) dans ce sujet qui sont suivis par l'utilisateur.
- le vote explicite de l'utilisateur pour ce sujet.
- un fonction d'affaiblissement pour les « anciens » articles (lus depuis longtemps).

Pour les profils utilisateurs représentés sous la forme des vecteurs booléen des termes, quelquefois un processus de sélection est nécessaire pour filtrer les mots importants [103] (par exemple, l'élimination des mots vides).

3.2 Utilisation de profils utilisateurs dans les systèmes personnalisés

Dans cette section nous abordons l'utilisation de profils utilisateurs dans les systèmes personnalisés comme les systèmes de recherche d'information ou les systèmes de filtrage d'information. Les systèmes de recherche d'information (RI) et de filtrage d'information (FI) ont un même objet : fournir des informations pertinentes aux utilisateurs. Cependant, il y a plusieurs différences entre ces deux types de systèmes d'information [8]. De plus, le rôle des utilisateurs dans ces deux types de système est très différent. Dans le premier

cas, les utilisateurs jouent un rôle actif : ils donnent des requêtes explicites au système et reçoivent des réponses. Dans le deuxième cas, ils jouent un rôle passif : le système sélectionne des informations pertinentes pour eux, en se basant sur leurs profils, et les délivrent aux utilisateurs. Des exemples des systèmes de RI et de FI sont respectivement les moteurs de recherche sur le Web et les systèmes de recommandation (cf. 3.2.4).

3.2.1 Reformulation de requêtes utilisateurs

La reformulation de requêtes utilisateurs a été utilisé depuis longtemps pour améliorer la performance des systèmes de recherche d'information (par exemple, la méthode de retour de pertinence de Rocchio [113].) Dans le système ARCH décrit dans [126] que nous avons mentionné dans la section 3.1.4.4, le profil d'un utilisateur se compose de plusieurs vecteurs pondérés de termes. Ce système collecte implicitement un ensemble de documents auquel l'utilisateur s'intéresse en prenant en compte plusieurs facteurs : fréquence de visite d'une page, le temps utilisé pour consulter la page, l'action de marque-pages (*bookmarking*) etc. Après qu'un nombre suffisant des documents a été collecté, le système utilise un algorithme de regroupement (*clustering*) pour regrouper ces documents dans les catégories différentes. Puis le système calcule les vecteurs centroïdes de ces catégories. Chaque vecteur centroïde représente un profil individuel (un domaine d'intérêt de l'utilisateur).

Chaque fois qu'un utilisateur soumet une requête Q_1 au système, le contenu de cette requête est comparé avec ces vecteurs pour trouver les vecteurs les plus similaires avec la requête (en utilisant un seuil de similarité). Le système va également comparer la requête avec les concepts dans une hiérarchie de concepts qui représente les domaines de connaissances pour trouver les concepts les plus similaires avec cette requête. Ces concepts sont aussi représentés par des vecteurs pondérés de termes. Enfin, les vecteurs sélectionnés sont comparés avec les concepts sélectionnés. Supposons que les concepts les plus similaires avec les vecteurs profils sont les concepts T_{sel} et les concepts les plus différents avec ces vecteurs sont les concepts T_{desel}, la requête Q_1 est reformulée en utilisant la méthode de Rocchio :

$$Q_2 = \alpha \cdot Q_1 + \beta \cdot \sum T_{sel} - \gamma \cdot \sum T_{desel}$$

Après cette étape, la requête reformulée Q_2 sera utilisée au lieu de la requête originale Q_1.

Un autre exemple est le système UCAIR [124] qui est basé sur le moteur de recherche Google. Ce système utilise les informations collectée implicitement dans la session de recherche actuelle pour élargir la requête utilisateur. Quand l'utilisateur envoie une requête, le système va calculer la similarité entre cette requête et la requête précédente. Si cette similarité dépasse un seuil, alors le système conclut que ces deux requêtes appartiennent à une même session de recherche (et partage le même besoin d'information). Si deux requêtes consécutives appartiennent à une même session de recherche et la fréquence d'un terme dans la première requête ou ses résultats dépasse un seuil prédéfini dans les résultats de la deuxième recherche (par exemple, dans 5 documents sur 50 documents retourné), ce terme sera ajouté à la deuxième requête pour former une requête élargie. Cette requête élargie sera envoyé au moteur de recherche au lieu de la requête originale.

En parallèle avec l'utilisation des informations de la requête précédente, le système UCAIR prend en compte également les informations provenant des documents que l'utilisateur a lu dans le passé pour mettre à jour son besoin d'information. Supposons que la requête originale de l'utilisateur est $\overrightarrow{q}$, au moment t l'utilisateur a lu k documents et les extraits (*snippets*) correspondants de ces k documents sont $s_1, ..., s_k$. Alors le besoin d'information de l'utilisateur sera mis à jour de manière suivante :

$$\overrightarrow{x} = \alpha \overrightarrow{q} + (1 - \alpha)\frac{1}{k}\sum_{i=1}^{k} \overrightarrow{s_i}$$

Le nouveau besoin d'information sera utilisé pour trier à nouveau les documents qui ne sont pas encore lus à ce moment pour la requête actuelle.

3.2.2 Visualisation de résultats

Un profil utilisateur peut être utilisé pour personnaliser la visualisation des résultats de recherche. Le système WEBCLUSTERS [128] permet à l'utilisateur de créer sa propre ontologie en utilisant l'outil WOE (*WEBCLUSTERS Ontology Editor*). Quand l'utilisateur soumet une requête, le système envoie sa requête à un des six moteurs de recherche (Google, Yahoo ! Search, Ask Jeeves, MSN Search, LookSmart, Overture) et reçoit des résultats. Ensuite, le système utilise un classificateur bayésien [87] pour classifier ces résultats dans les concepts correspondants dans l'ontologie personnelle de l'utilisateur. La visualisation personnalisée de résultats peut permettre à l'utilisateur d'identifier rapidement les documents intéressants.

3.2.3 Re-classement de résultats

Le re-classement de résultats de recherche pour donner une meilleure précision n'est pas une nouvelle idée dans la RI. Plusieurs recherches ont été faites pour améliorer la qualité des moteurs de recherche existants en re-triant leurs résultats. Une des approches est d'utiliser des méta-moteurs de recherche [118, 34]. Un méta-moteur envoie une même requête à plusieurs moteurs de recherche (Google, MSN Search, Yahoo ! Search ...), reçoit les résultats, puis re-trie les documents et les présente aux utilisateurs.

Dans un système de RI personnalisé qui utilise cette approche, le système envoie une requête à un moteur de recherche, reçoit des résultats et puis re-trie les résultats selon leurs similarités avec le profil d'utilisateur. Dans les travaux de Speretta et al. [133], les auteur utilisent un modèle à base d'ontologie (cf. 3.1.2.2) pour représenter les profils utilisateurs. Chaque concept dans l'ontologie a un poids représentant l'intérêt de l'utilisateur avec ce concept. Ces poids sont accumulés avec le temps en utilisant l'histoire de recherche de l'utilisateur. Les informations prises en compte pour mettre à jour le profil sont les anciennes requêtes et les titres et les extraits (*snippets*) des résultats sélectionnés par l'utilisateur. Ce système utilise un *wrapper* pour le moteur de recherche Google. Ce *wrapper* est construit en utilisant le *Google API*[4] et surveille les actions de l'utilisateur (requêtes soumise, clics sur les résultats, etc.). Chaque fois qu'une requête est soumise, la

[4]http://www.google.com/apis/

similarité entre le profil de l'utilisateur et un document retourné est calculé par la formule suivante :

$$similarité(u,d) = \sum_{k=1}^{N} wt_{uk} \times wt_{dk}$$

Dans cette formule, wt_{uk} est le poids du concept k dans le profil de l'utilisateur u et wt_{dk} est le poids du concept k dans le document d. Les documents sont triés par leur similarité avec le profil d'utilisateur et ce rang est appelé *rang_ concept* pour le distinguer du rang original de Google. Le classement final de document est calculé en utilisant une combinaison de ces deux rangs :

$$rang_final = \alpha \times rang_concept + (1-\alpha) \times rang_Google$$

où α est un valeur entre 0 et 1. Le $rang_final$ est utilisé pour trier et présenter les documents à l'utilisateur.

Dans les travaux de Gauch et al. [45] (voir également la section 3.1.4.2) qui utilise le modèle ontologie pour représenter des profils utilisateurs. Ils utilisent le profil utilisateur pour trier à nouveau les résultats obtenus par le méta-moteur de recherche ProFusion. Pour chaque résultat r (document retourné) de ce méta-moteur, ils calculent un nouveau score new_wt_r en utilisant le résultat original du moteur, la similarité entre le résultat et les concepts correspondants, et l'intérêt de l'utilisateur avec ces concepts (les poids de ces concepts) :

$$new_wt_r = wt_r(0,5 + \frac{1}{4}\sum_{l=1}^{4} u_{crl})$$

Dans cette formule, wt_r est le score original calculé par le moteur de recherche pour le résultat r, u_{crl} est l'intérêt de l'utilisateur avec le concept c_{rl} dans son profil, et c_{rl} est le l^{eme} concepts parmi les concepts les plus similaires avec le résultat r. Le nouveau score new_wt_r sera utilisé pour trier à nouveau les documents.

Parmi les autres travaux qui utilisent les profils utilisateurs pour le re-classement des résultats de recherche, nous pouvons citer [135, 114, 124, 125].

3.2.4 Systèmes de recommandation

Les systèmes de recommandation sont une forme spéciale des systèmes de filtrage d'information. Ils sont en charge de recommander des items qui sont potentiellement intéressants pour les utilisateurs. Il existe trois principales méthodes dans ces systèmes : recommandation basée sur le contenu des items (*content-based*), recommandation collaborative (*collaborative recommendation*), et approche hybride. Dans les parties suivantes nous allons aborder ces méthodes.

3.2.4.1 Systèmes de recommandation basés sur le contenu

Dans ces systèmes, la recommandation est basée sur l'analyse de contenu des items auxquels les utilisateurs se sont intéressés dans le passé ainsi que le nouvel item. Si la similarité du contenu du nouvel item avec le contenu du profil d'utilisateur dépasse un

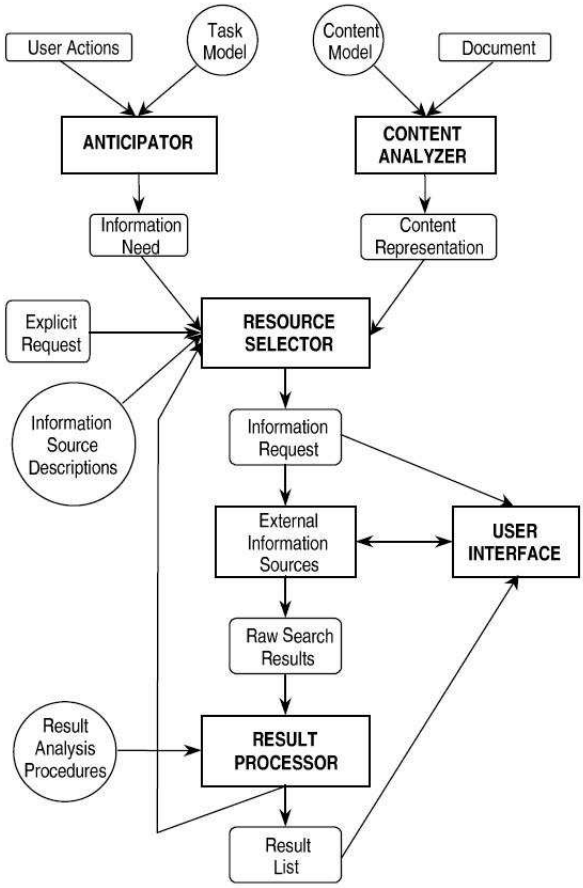

FIG. 3.3 – Architecture d'un assistant de gestion d'information [19].

seuil prédéfini le nouvel item sera recommandé à l'utilisateur. Les contenus des anciens items intéressants sont « stockés » dans le profil utilisateur. Dans les systèmes utilisant des vecteurs de termes pondéré pour représenter les profils utilisateurs, la similarité entre le nouvel item et le profil est souvent calculée par la formule du cosinus [22, 84, 121].

Une autre approche populaire pour la recommandation est d'utiliser des classificateurs [10, 92, 91, 102]. Dans le système Syskill & Webert [102], le profil utilisateur se compose de deux ensembles d'exemplaires : le premier contient des pages Web intéressantes pour l'utilisateur, et le deuxième contient des pages Web qu'il/elle a trouvé inintéressantes. Les pages Web dans le profil sont représentées par des vecteurs booléens de termes. Quand il y a une nouvelle page Web, le système utilise un classificateur bayésien pour estimer la probabilité que cette nouvelle page appartienne à l'ensemble « intéressant » ou « inintéressant ». Cette approche est aussi appelée *apprentissage supervisé*.

Dans les systèmes Foxtrot et Quickstep [92, 91], on utilise un modèle à base d'ontologie des articles scientifiques pour représenter les centres d'intérêts des utilisateurs. Le profil utilisateur est un ensemble de triplets $<$*utilisateur, sujet, valeur*$>$ dans lequel *valeur* représente l'intérêt de l'utilisateur pour le sujet. Les recommandations des nouveaux articles sont calculées en utilisant la corrélation des sujets d'intérêts de l'utilisateur et ces articles. Quand les nouveaux articles arrivent, un classificateur IBk [92, 91] est utilisé pour classifier ces articles dans les centres d'intérêt. La confiance de recommandation (l'intérêt potentiel) d'un utilisateur avec ce nouveau document est estimé par la formule suivante : $Confiance_de_recommandation = Confiance_de_classification \cdot Valeur_de_topic$; les articles non lus avec bonnes confiances de recommandation serons recommandés à l'utilisateur.

Le système Letizia [84] utilise une approche implicite pour construire le profil utilisateur : par exemple, suivre un lien Web peut être considéré comme une preuve que l'utilisateur s'intéresse au document qui contient le lien. Quand l'utilisateur est en train de visiter une page Web, Letizia peut pré-charger un ensemble des pages Web qui sont reliées avec cette page Web. Puis il calcule les similarités de ces pages Web avec le profil utilisateur pour recommander les pages Web qui sont potentiellement intéressantes avec l'utilisateur.

Budzik et al. [19] présentent un système d'assistant de gestion d'information (*Information Management Assistant* - IMA) (figure 3.3). Ce système surveille les interactions de l'utilisateur avec les applications (par exemple, MS Word) pour donner des recommandations. Il y a plusieurs modules dans le système. Le module ANTICIPATOR est en charge d'interpréter les actions de l'utilisateur et prévoir son besoin d'information possible. Le module CONTENT ANALYZER est en charge de produire une représentation du document que l'utilisateur est en train de manipuler. Cette représentation et le besoin d'information sont ensuite utilisés par le module RESOURCE SELECTOR. Ce module utilise ces informations pour créer une *requête d'information*, l'envoyer aux sources externes et recevoir une liste des résultats sous la forme d'une page HTML. Après, cette liste est interprétée et filtrée par le module RESULT PROCESSOR pour éliminer les résultats non pertinents ou pour former les nouvelles requêtes du module RESOURCE SELECTOR. Enfin, une liste de résultats finale est présentée à l'utilisateur dans une autre fenêtre.

3.2.4.2 Systèmes de recommandation collaborative

Dans les systèmes de recommandation collaborative, on utilise souvent le modèle matrice utilisateurs-items pour représenter les profils utilisateurs (cf. 3.1.2.4). Le système doit prévoir l'utilité des items pour un utilisateur quelconque. Selon Breese et al. [16], les algorithmes d'estimation de l'intérêt d'un utilisateur avec un item quelconque peuvent être classés en deux classes principales : les algorithmes basées sur la mémoire (*memory-based algorithms*) qui utilisent toute la base de données utilisateurs-items pour faire des prédictions et les algorithmes basées sur les modèles (*model-based algorithms*) qui utilisent cette base de données pour apprendre un modèle, et puis utilisent ce modèle pour faire des prédictions.

Algorithmes basés sur la mémoire : supposons que I_u est l'ensemble des items pour lesquels un utilisateur u a voté dans le passé, la valeur moyenne des votes de l'utilisateur u est :

$$\overline{v}_u = \frac{1}{|I_u|} \sum_{j \in I_u} v_{u,j}$$

A partir des valeurs moyennes des votes de tous les utilisateurs, la valeur du vote $p_{u',j}$ d'un utilisateur u' pour un item j qu'il ne connaît pas est évaluée par la formule suivante :

$$p_{u',j} = \overline{v}_{u'} + \kappa \sum_{u} w(u,u')(v_{u,j} - \overline{v}_u)$$

Le poids $w(u,u')$ représente la corrélation ou similarité entre l'utilisateur u et l'utilisateur u', κ est un paramètre prédéfini. Dans plusieurs cas, cette corrélation est calculée par l'algorithme *Pearson-r* suivant :

$$w(u,u') = \frac{\sum_j (v_{u,j} - \overline{v}_u)(v_{u',j} - \overline{v}_{u'})}{\sqrt{\sum_j (v_{u,j} - \overline{v}_u)^2 \sum_j (v_{u',j} - \overline{v}_{u'})^2}}$$

Une autre approche pour calculer $w(u,u')$ est d'utiliser une variante de la mesure cosinus. Dans ce cas, les utilisateurs jouent le rôle des documents, les titres des items jouent le rôle des termes, et les votes jouent le rôle des fréquences des termes. Selon cet algorithme, il n'y a pas de votes négatifs et les items qui ne sont pas observés reçoivent un vote *zéro*. Maintenant, la corrélation entre deux utilisateurs u et u' est calculée de manière suivante :

$$w(u,u') = \sum_j \frac{v_{u,j}}{\sqrt{\sum_{k \in I_u} v_{u,k}^2}} \frac{v_{u',j}}{\sqrt{\sum_{k \in I_{u'}} v_{u',k}^2}}$$

Après le calcul, si le vote prévu $p_{u',j}$ du nouvel item j est élevé, cet item sera recommandé à l'utilisateur u'.

Algorithmes basés sur les modèles : dans ces algorithmes, la recommandation peut être considérée comme un processus probabiliste pour prévoir un vote d'utilisateur sur un item quelconque en prenant en compte ses votes sur les autres items. Supposons que les votes sont des valeurs dans l'intervalle $[0,m]$, alors le vote prévu $p_{u',j}$ est calculé par :

$$p_{u',j} = E(v_{u',j}) = \sum_{i=0}^{m} Pr(v_{u',j} = i | v_{u',k}, k \in I_a)$$

Dans leurs travaux, Breese et al. présentent également deux modèles probabilistes pour ce but : modèle *cluster* et modèle bayésien.

Parmi les systèmes de recommandation collaborative connus, nous pouvons citer [69, 123, 92, 111].

3.2.4.3 Systèmes hybrides

Les deux approches de recommandation basée sur le contenu et de recommandation collaborative ont leurs avantages et leurs inconvénients. Dans l'approche basée sur le contenu, les items recommandés aux utilisateurs ne sont que des items similaires aux items qu'ils ont trouvé intéressants dans le passé. Donc un nouvel item différent de ces items ne sera pas recommandé bien que il puisse être intéressant pour eux. De plus, l'analyse de contenu est souvent limité aux documents textuels.

Dans la deuxième approche, si le nombre d'utilisateurs est très petit par rapport au nombre des items, alors il est possible qu'un item n'ait pas reçu un nombre suffisant de votes pour être recommandable. Un autre inconvénient est qu'un utilisateur ayant des intérêts assez différents par rapport aux autres peut recevoir peu de recommandations. De plus, il y a un problème de *« démarrage à froid »* ou *cold-start* : quand le système vient de démarrer, il n'y a pas de votes disponibles donc il ne peut pas faire de recommandations.

A cause des raisons ci-dessus, plusieurs systèmes [6, 92] ont choisi de combiner ces deux méthodes pour bénéficier de leurs avantages et réduire leurs inconvénients. Il existe des panoramas plus complets sur les systèmes de recommandation, nous pouvons citer à titre d'exemple les articles suivants [95, 94, 107].

3.3 Bilan

Dans le chapitre 2, nous avons présenté les concepts de base d'un système de RI. Dans ce chapitre, nous venons de présenter les systèmes personnalisés actuels. Les sujets abordés dans ce chapitre sont les modèles de représentation de profils utilisateurs, les méthodes d'acquisition des informations des utilisateurs, les techniques de construction et de mise à jours de profils utilisateurs, l'utilisation de profils utilisateurs dans des systèmes de RI et de FI actuels. Comme nous l'avons mentionné dans ce chapitre, chaque approche de personnalisation de ces systèmes possède des avantages et des inconvénients. Parmi les modèles de représentation de profils utilisateurs, le modèle vectoriel est facile à implémenter. Cependant, il ne prend pas en compte différents niveaux de généralités caractérisant l'utilisateur et différents niveaux d'importance des centres d'intérêts de l'utilisateur [151] même s'il permet de représenter ces différents centres d'intérêts en utilisant plusieurs vecteurs pondérés. Par contre, le modèle à base d'ontologie permet de mieux connaître les intérêts des utilisateurs par rapport au modèle vectoriel. Cependant, ce modèle est plus difficile à mettre en œuvre, car il demande souvent beaucoup d'information de l'utilisateur (par exemple, les documents qui représentent ses centres d'intérêt) pour construire

l'ontologie qui représente son profil. Le modèle multidimensionnel semble être un bon modèle pour représenter les profils utilisateurs. Il peut contenir également les autres modèles (comme le modèle vectoriel ou le modèle à base d'ontologie). Cependant, chaque système différent a besoin de différents dimensions pour représenter ses utilisateurs.

A nos jours, les bibliothèques numériques sont très populaires, elles ont plusieurs avantages par rapport aux bibliothèques traditionnelles. Elles peuvent être une option remplaçant les sites Web pour les personnes qui souhaitent des articles de haute qualité. Surtout dans le domaine d'information, les bibliothèques spécialisées comme CiteSeer, ACM, Xplore ... sont un outil indispensable pour les informaticiens. Dans le chapitre suivant, nous nous concentrons sur les bibliothèques numériques. En particulier, nous abordons le sujet principal de cette thèse : la personnalisation de ces bibliothèques. Nous sommes intéressés particulièrement aux bibliothèques qui contiennent des articles scientifiques.

Chapitre 4

La personnalisation dans les bibliothèques numériques

Dans ce chapitre, nous traitons des bibliothèques numériques : définition, histoire, évolution, avantages des bibliothèques numériques etc. Nous présentons également quelques exemples de bibliothèques numériques. Enfin, nous abordons le problème de personnalisation dans cet environnement.

4.1 Bibliothèques numériques

4.1.1 Qu'est-ce qu'une bibliothèque numérique ?

D'une manière simple, une bibliothèque numérique est une collection de documents numériques. Cependant, une bibliothèque n'est pas seulement un ensemble de n'importe quels documents numériques. Voici quelques définitions d'une bibliothèque numérique :

- Définition de Michael Lesk [82] : « une bibliothèque numérique n'est pas seulement une collection d'information électronique. C'est un système de données organisé et numérisé qui peut servir comme une ressource riche pour sa communauté d'utilisateurs »[1].

- Définition de William Y. Arms ([2], page 2) : une bibliothèque numérique est « une collection gérée d'informations, avec les services associés, où l'information est stockée sous des formats numériques et accessible au travers d'un réseau »[2].

- Définition de Baker ([4], page 417) : « Les bibliothèques numériques sont construites – collectées et organisées – par une communauté d'utilisateurs. Leurs capacités fonctionnelles supportent les besoins d'information et les utilisations de cette communauté. Une bibliothèque numérique est une extension, amélioration, et intégration d'une variété de diffusion de l'information, comme par exemple les places physiques,

[1] *A digital library is not merely a collection of electronic information. It is an organized and digitized sytem of data that can serve as a rich resource for its user community.*

[2] *A digital library is a managed collection of information, with associated services, where the information is stored in digital formats and accessible over a network.*

où les ressources sont sélectionnées, collectées, organisées, préservées, et mise à disposition d'une communauté d'utilisateur [3] ».

A partir de ces définitions, nous pouvons voir qu'une bibliothèque numérique est beaucoup plus qu'une collection de documents numériques. Ceux-ci sont sélectionnés, organisés, préservés et la bibliothèque fournit un ensemble des services utiles pour les utilisateurs. Une bibliothèque numérique est souvent accessible sur l'Internet bien qu'il y ait des bibliothèques numériques qui ont existé avant l'apparition de l'Internet ; par exemple le projet Gutenberg a commencé à partir de 1971. Les bibliothèques numériques fournissent aux utilisateurs un moyen d'accès aux documents plus confortable et plus rapide par rapport aux bibliothèques traditionnelles. Si nous avons besoin d'articles de haute qualité, les bibliothèques numériques peuvent être une option remplaçant les sites Web. Les documents dans les bibliothèques numériques sont souvent bien organisés et structurés. Ils sont attachés à plusieurs méta-informations qui permettent de les identifier et de les décrire.

4.1.2 Histoire et évolution de bibliothèques numériques

En 1945, Vannevar Bush a publié l'article *Tel que nous pourrions penser* (*As We May Think* en anglais) dans le journal *Atlantic Monthly* [20]. Cet article contient plusieurs idées révolutionnaires dans plusieurs domaines. Dans cet article, Vannevar Bush a présenté un système qui s'appelle Memex (*memory extender* ou mémoire étendue). Ce système utilise des *microfilms* pour enregistrer des documents, cette technologie permet de stocker et d'organiser une grande quantité d'information par rapport aux moyens de stockage traditionnels. Dans Memex, les documents peuvent être inter-connectés par des liens. Ce système est souvent considéré comme le premier prototype d'une bibliothèque numérique et du WWW. Bush a décrit que avec ce système, « l'Encyclopedia Britannica pourrait être réduite au volume d'une boîte d'allumettes. Une bibliothèque d'un million de volumes pourrait être compressé dans le coin d'un bureau »[4]. Cependant, le système Memex n'est qu'un modèle qui n'a jamais été implémenté à cause des limites techniques de cette époque. Ce système est illustré dans le figure 4.1[5].

En 1965, dans un livre intitulé « Bibliothèques du Futur » (*Libraries of the Future*) [83], Licklider a estimé la taille du corpus de littérature des sciences et des connaissances humaines dans l'avenir. Il a également décrit les systèmes nommés « Procognitive » pour stocker et traiter ces informations. Licklider a prévu dès cette époque et avec une haute précision le développement des bibliothèques numériques.

De nos jours, avec le développement des sciences et des technologies, les bibliothèques numériques sont entrées dans la vie réelle et deviennent de plus en plus populaires et leurs capacités dépassent largement l'imagination de Bush. Nous pouvons citer les deux facteurs les plus importants qui ont permis le développement des bibliothèques numériques :

[3] *DLs are constructed - collected and organized - by a community of users. Their functional capabilities support the information needs and uses of that community. DL is an extension, enhancement, and integration of a variety of information institutions as physical places where resources are selected, collected, organized, preserved, and accessed in support of a user community.*

[4] *The Encyclopoedia Britannica could be reduced to the volume of a matchbox. A library of a million volumes could be compressed into one end of a desk.*

[5] Source d'image : http://www.futureofthebook.org/blog/archives/2005/02/

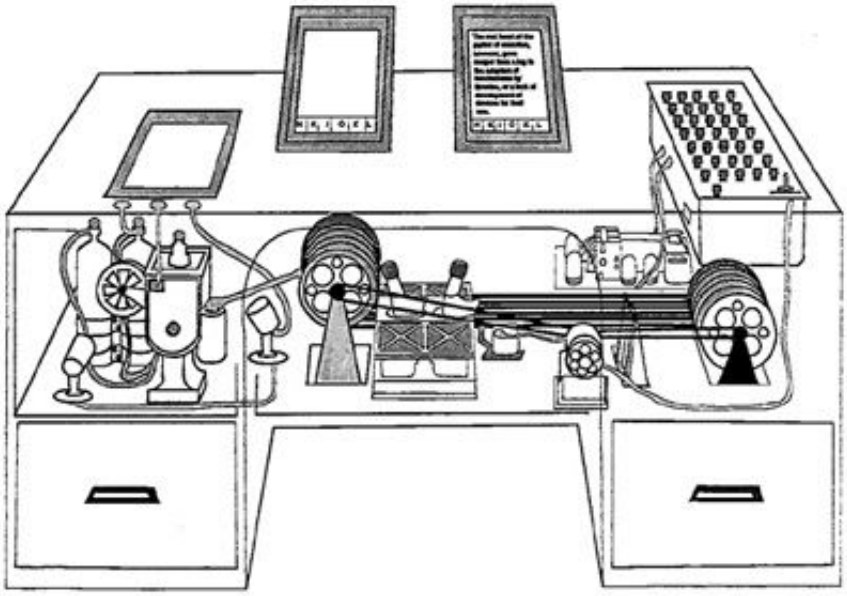

FIG. 4.1 – Le système Memex.

- **La technologie de l'ordinateur** : sans aucun doute, le développement de la technologie de l'ordinateur est très important dans le développement des bibliothèques numériques. Selon la loi de Moore, le nombre de transistors qu'on peut mettre dans un circuit intégré est doublé environ tous les 18-24 mois. On peut imaginer que la puissance de calcul des ordinateurs augmente avec une vitesse correspondante. La technologie de mémoire et de stockage a des améliorations similaires. Le prix des systèmes de stockage a fortement baissé. Le premier disque magnétique, fabriqué par IBM en 1956, pouvait stocker 4,5 Mo de données et son prix était de 40.000$; aujourd'hui nous pouvons acheter un disque dur qui peut contenir plus de 100 Go de donnés pour moins de 100$. De plus, les ordinateurs personnels deviennent de plus en plus populaire. Au fur et à mesure, les hommes prennent l'habitude de lire directement les documents électroniques sur l'écran de l'ordinateur au lieu de chercher une version papier de ces documents.

- **Internet et WWW** : l'Internet et le WWW jouent également un rôle important pour les bibliothèques numériques. De plus, il y a de plus en plus de personnes qui utilisent des connexions haut débit. Grâce à ces deux derniers, nous pouvons maintenant nous connecter aux bibliothèques numériques de façon rapide.

4.1.3 Architecture d'une bibliothèque numérique moderne

En 1995, Kahn et Wilensky [62] ont proposé une architecture importante pour les bibliothèques numériques. Puis elle a été étendue par Arms et al. [3]. Brièvement, les principaux composants de cette architecture sont les suivants :

- **Objets numériques :** les éléments dans une bibliothèques sont appelés des objets

numériques[6]. Un objet numérique contient deux composants : les méta-données qui sont des informations de gestion de cet objet et les données de contenu de cet objet. Les méta-données contiennent un *handle*, c'est un identifiant unique de chaque objet numérique.

- **Dépôt :** un dépôt[7] est un système de stockage d'objets numériques. On peut accéder au dépôt en utilisant un protocole d'accès (*Repository Access Protocol* - RAP). Ce protocole fournit des opérations sur les objets (accéder, ajouter, modifier etc.).

- **Système d'identification :** le système d'identification a pour but de gérer les *handles* des objets. Ce système stocke les handles et les informations qui ont pour but de localiser les objets.

- **Interface utilisateurs :** les interfaces utilisateurs permettent aux utilisateurs d'utiliser les services de bibliothèque telles que la navigation, la recherche d'information, la visualisation etc. Il existe deux types d'interface utilisateurs dans une bibliothèque numérique : l'un pour les usagers de la bibliothèque, l'autre pour les bibliothécaires et les administrateurs du système.

- **Système de recherche d'information** : le système de RI dans une bibliothèque numérique permet de rechercher pour découvrir les informations avant de les charger à partir d'un dépôt.

A côté de ces composants, d'autres composants et services sont souvent intégrés à une bibliothèque numérique : service de sécurité pour l'identification et la gestion des droits des utilisateurs ; service de journalisation de transactions (*transaction logs*) ; service de miroir ou de cache de données pour accélérer l'accès aux données etc. Une illustration de cette architecture est montrée dans la figure 4.2.

Paepcke et al. [99] montrent que l'interopérabilité est aussi un sujet important pour les bibliothèques numériques parce que les dépôts d'information et les services peuvent être fournis par plusieurs organisations indépendantes autour du monde. Selon eux, les cinq fonctionnalités principales d'une bibliothèque numérique sont : la gestion de données (stockage, organisation, recherche d'information), la présentation de l'information pour les utilisateurs, la communication entre les éléments du système, l'initiation et le contrôle des actions du système, et la protection pour les utilisateurs, leurs propriétés, et les ressources d'information. L'interopérabilité concerne non seulement la communication entre les différents composants de différents systèmes mais aussi entre les différentes fonctionnalités d'un système. Pour traiter le problème de l'interopérabilité, le protocole PMH (*Protocol for Metadata Harvesting*) a été développé par l'OAI (*Open Archives Initiative*). Ce protocole permet aux bibliothèques numériques d'échanger les méta-données décrivant les collections. Ce protocole est actuellement supporté par plusieurs bibliothèques numériques[8] [39].

[6] *digital object* en anglais.
[7] *repository* en anglais.
[8] http://www.openarchives.org/OAI/openarchivesprotocol.html

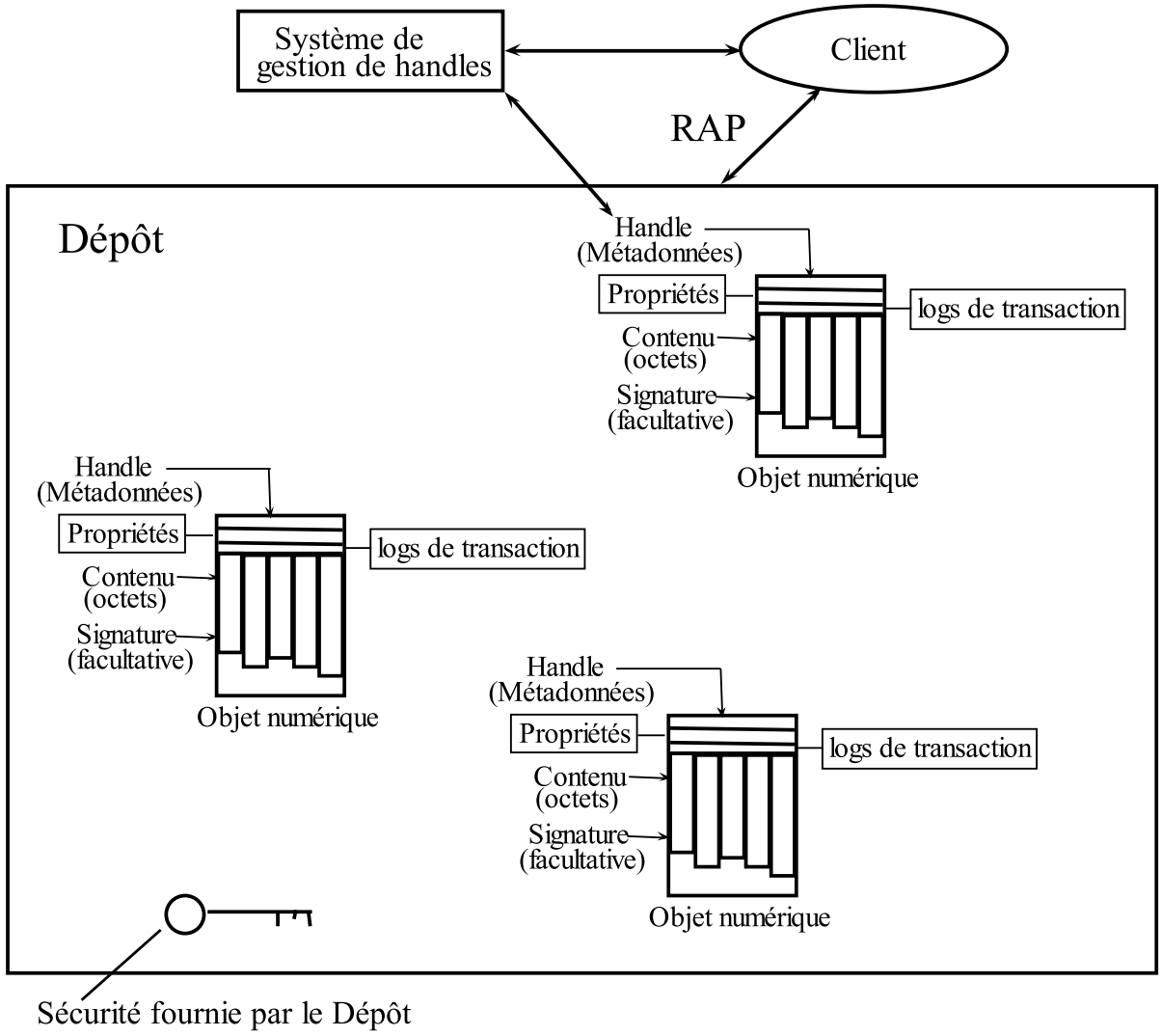

FIG. 4.2 – Objets numériques, handles, et dépôt [4].

4.1.4 Recherche d'information dans les bibliothèques numériques

Dans le système Memex, Vannevar Bush n'a pas prévu un mécanisme de recherche d'information automatique. Dans son système, toutes les informations sont organisées et indexées par l'homme. Cependant maintenant la RI est un service indispensable des bibliothèques numériques. Comme nous l'avons mentionné dans la section 4.1.3, à l'aide des systèmes informatiques, la recherche de documents est un des avantages d'une bibliothèque numérique par rapport aux bibliothèques traditionnelles. Les critères de recherche sont plus nombreux et flexibles : nous pouvons rechercher sur le contenu, sur le titre, sur le nom des auteurs, sur l'année de publication, sur la conférence ou la revue, sur les bibliographies des documents, ...

Dans l'article [117], Schatz retrace les efforts pour construire des systèmes de RI dans les bibliothèques numériques depuis les années 1960, à commencer par le besoin de recherche de citations des articles scientifiques dans les bases de données bibliographiques (MEDLINE, Inspec, etc.). A cause des limites techniques à cette époque, les premiers systèmes ne font qu'un appariement simple des mots dans les champs des articles tels que titre, noms des auteurs, mots clés, nom de la revue (modèle booléen). Puis on a pu rechercher également dans les résumés des articles. Et après, quand les réseaux deviennent plus rapides et les capacités de stockage deviennent plus larges, on a pu rechercher dans tout le contenu des articles.

Depuis les années 1980, les ordinateurs personnels sont devenus de plus en plus populaires et la technologie de multimédia s'est développée, les interfaces graphiques se sont démocratisées. Les documents peuvent maintenant contenir des images, des vidéos etc. Les utilisateurs peuvent combiner recherche et navigation pour découvrir les documents de façon plus efficace. De plus, avec le développement de la technologie des réseaux, plusieurs collections de documents peuvent être stockées à plusieurs endroits mais les utilisateurs n'ont pas à savoir où. Pour eux, les résultats de la recherche sont affichés de manière cohérente comme s'ils venaient d'une seule collection.

Maintenant, une des pistes importantes pour les nouvelles bibliothèques numériques est de construire les bibliothèques personnalisées qui peuvent s'adapter aux intérêts et préférences spécifiques des utilisateurs. Par exemple, les interfaces adaptatives, la recherche d'information personnalisée [114] ou la visualisation personnalisée des résultats de recherche [128], la recommandation automatique [138] des documents qui sont potentiellement intéressant etc. Nous aborderons en plus détail ce problème dans la section 4.3.

4.1.5 Avantages et inconvénients des bibliothèques numériques

Une bibliothèque numérique possède plusieurs avantages par rapport à une bibliothèque traditionnelle. Nous listons quelques avantages (de façon non exhaustive) ci-dessous :

- **Accès distant et immédiat :** Comme les bibliothèques numériques sont souvent accessibles sur l'Internet, les utilisateurs peuvent les accéder à partir de n'importe où et n'importe quand dès qu'ils ont une connexion Internet.

- **Partage de ressources :** Dans une bibliothèque numérique, plusieurs utilisateurs

peuvent accéder aux mêmes ressources en même temps.

- **Facilité de la navigation et de la recherche des documents :** Avec l'aide des systèmes informatiques, la navigation et la recherche des documents dans les bibliothèques numériques devient beaucoup plus facile que dans les bibliothèques traditionnelles.

- **Préservation et conservation de données :** Comme les informations numériques peuvent être copiées sans erreur, il est plus facile de préserver et de conserver ces objets sans qu'une utilisation intensive n'apporte de détérioration[9].

- **Mise à jour/publication facile et rapide** : grâce à l'Internet et les systèmes informatiques, la mise à jour des bibliothèques numériques est très facile et rapide. Dans la chaîne de publication traditionnelle, avant qu'un article puisse être lu par les lecteurs, il doit passer plusieurs étapes intermédiaires (éditeur, imprimeur, distributeur ...). Cependant, s'il est publié dans les bibliothèques numériques, on n'a pas besoin des imprimeurs et des intermédiaires.

Néanmoins, bien que les bibliothèques numériques aient beaucoup d'avantages par rapport aux bibliothèques traditionnelles, elles ne peuvent pas encore totalement les remplacer. Ci-dessous sont quelques inconvénients de ces bibliothèques [82] :

- **Coût de remplacement :** Pour remplacer une bibliothèque traditionnelle en cours d'utilisation, il faut investir dans un système pour convertir les documents papiers en documents numériques (par exemple, un système de reconnaissance optique de caractères - ROC). Probablement, il faut payer également les droits d'auteur pour ces documents. De plus, comme les documents numériques sont très faciles à copier, il faut aussi prendre des mesures pour prévenir cette éventualité.

- **Préférence des lecteurs :** Malgré qu'il y ait de plus en plus de personnes qui utilisent des ordinateurs pour lire des documents, d'autres ne veulent pas changer leurs habitudes et veulent les lire sans leur forme traditionnelle.

- **Matériel :** Accéder aux bibliothèques numériques demande un ordinateur et une connexion Internet, qui ne sont pas toujours disponible pour tous le monde.

Cependant, la situation est en train de changer très vite. Les coûts des systèmes et des matériels sont en train de baisser, les habitudes des personnes sont en train de changer, et nous pouvons croire qu'un jour les bibliothèques numériques joueront un rôle encore plus déterminant.

[9] Cependant, est-ce que les bibliothèques numériques sont un moyen de conservation à long-terme ou non est une question à débat.

4.2 Quelques exemples de bibliothèques numériques

Il existe des bibliothèques « généralistes » qui contiennent des documents sur plusieurs domaines ou des bibliothèques « spécialisées » qui contiennent seulement des documents sur un domaine spécifique. Dans cette section nous présentons quelques bibliothèques numériques de ces types.

4.2.1 Projet Gutenberg

Le projet Gutenberg[10] a été commencé en 1971 par Michael Hart, un étudiant à l'Université de l'Illinois aux Etats-Unix. Il est considéré comme la première bibliothèque numérique. Le projet a été nommé selon l'imprimeur allemand Gutenberg, l'inventeur de l'imprimerie typographique en Europe. C'est un projet pour collecter, numériser et distribuer des livres du domaine public — c'est-à-dire les livres qui ne sont plus protégés par les lois sur la propriété intellectuelle. En 2008, selon son site Web, le projet contient plus de 25000 livres électroniques. La plupart des livres sont en anglais, mais d'autres langues sont représentés.

4.2.2 Gallica

Gallica[11], la bibliothèque numérique de la bibliothèque nationale de France, est une bibliothèque patrimoniale et encyclopédique. Elle couvre plusieurs domaines tels que l'histoire, la littérature, les sciences, la philosophie, le droit, l'économie et la science politique etc. Depuis 2007, elle numérise environ 100 000 nouveaux documents chaque année en mode texte et en mode image. Elle a l'intention de numériser 27 journaux de presse du XIX^e siècle à 1944 avec un coût global de 3,5 millions d'euros. Les journaux en cours de numérisation sont « La Croix », « Le Temps », « Le Monde Diplomatique », « Le Figaro », « L'Humanité » etc.
Actuellement, une nouvelle version de Gallica – Gallica 2 – est en train d'être construite.

4.2.3 Google recherche de livres

« Google recherche de livre »[12] est un service de recherche de livre de la société Google. Ce service numérise et stocke les livres de plusieurs grandes bibliothèques. Ce service permet aux utilisateurs de rechercher sur la collection de ces livres et le système affiche les informations générales sur les livres dans le résultat. Cependant, dans plusieurs cas l'utilisateur ne peut pas consulter tous les contenus de ces livres mais il peut seulement lire des courts extraits ou un nombre limité de pages. Si un livre est dans le domaine public, alors l'utilisateur peut consulter et télécharger le livre entier. Le service propose toujours des liens vers les éditeurs et les vendeurs de ces livres ou vers les bibliothèques où l'utilisateur peut emprunter un exemplaire.

[10] http://www.gutenberg.org
[11] http://gallica.bnf.fr/
[12] http://books.google.com/

4.2.4 CiteSeer

La bibliothèque CiteSeer[13] [46] est un représentant d'une nouvelle génération des bibliothèques numériques : celles qui peuvent automatiquement collecter et organiser leurs documents sans l'intervention de l'homme. En 2008, elle permet l'accès à plus de 700 000 articles scientifiques, notamment en informatique et en science de l'information. CiteSeer utilise des moteurs de recherche sur le Web et des heuristiques pour localiser des articles scientifiques. Par exemple, elle recherche des pages qui contiennent des mots tels que « publications », « papers », « postscript » etc. (Il existe des travaux sur le problème de collecter automatiquement des documents appartenant à un domaine donné sur le Web en utilisant des robots d'indexation dédié à ce domaine[14] [9, 101].) Puis elle télécharge les fichiers des articles, repère les duplicatas et extrait des informations utiles à partir de ces fichiers. Les informations extraites sont les informations suivantes :

- *URL :* Le lien où le fichier a été téléchargé.
- *Header :* Le titre de l'article et les noms des auteurs.
- *Abtract :* Le résumé de l'article.
- *Introduction :* La partie d'introduction de l'article.
- *Citations :* Les références de l'article. CiteSeer peut utiliser ces information pour construire un graphe de citations de ses articles.
- *Citation context :* Le contexte où les références sont citées dans l'article.
- *Full text :* Tout le contenu textuel de l'article est indexé.

CiteSeer n'est pas seulement une bibliothèque numérique, elle est aussi une base de citations (cf. le chapitre 5) avec un mécanisme d'indexation de citations autonome (*Autonomous Citation Indexing*). En utilisant le graphe de citation de ses articles, CiteSeer peut fournir des fonctionnalités basées sur les citations à côté des fonctionnalités basées sur le contenu textuel des documents. Par exemple, il permet de trouver des articles similaires qui partagent des références communes. Une liste complète des fonctionnalités disponibles de CiteSeer peut être trouvée dans son site Web[15].

4.2.5 Bibliothèque CODESNET

L'action CODESNET[16] (*COllaborative DEmand and Supply NETworks*) a pour vocation de fédérer une communauté s'intéressant au renforcement des structures d'organisation des entreprises en réseau. Un aspect important de cette communauté rassemblant industriels, chercheurs, enseignants et apprenants est le partage de savoirs. La bibliothèque numérique CODESNET a pour but de créer une environnement d'échange de documents et de connaissances dans cette communauté.

4.3 Bibliothèques numériques personnalisées

Comme nous l'avons mentionné dans la section 4.1.4, la recherche d'information est un service indispensable d'une bibliothèque numérique. Actuellement, comme plusieurs

[13] http://citeseer.ist.psu.edu/
[14] *focused crawler* ou *topical crawler* en anglais.
[15] http://citeseer.ist.psu.edu/citeseer.html
[16] http://www.codesnet.polito.it/

systèmes de recherche et de filtrage d'information, certaines bibliothèques numériques fournissent des services personnalisés pour s'adapter aux besoins d'informations spécifiques de leurs utilisateurs. Toutes les techniques de personnalisation abordées dans le chapitre précédent peuvent être appliquées dans le cas des bibliothèques numériques. De plus, une bibliothèque numérique est souvent rattachée à une institution ou un organisme dont l'utilisateur est membre. Chaque utilisateur a déjà un statut dans cette institution, ce qui permet de mieux le connaître (par exemple, son domaine de recherche) et donc de faciliter le processus de profilage.

4.3.1 Les approches de personnalisation

Dans [41], les auteurs classifient les approches de personnalisation dans les bibliothèques numériques. Selon eux, il y a trois types de services personnalisés basiques fournis par une bibliothèque numérique :

- **Personnalisation de contenu :** Ce mécanisme permet à chaque utilisateur de créer sa bibliothèque personnelle qui contient seulement des informations qui sont intéressantes et pertinentes pour lui.
- **Personnalisation pour aider à la navigation :** Ce mécanisme facilite les interactions de l'utilisateur avec la bibliothèque numérique. Par exemple, il permet à l'utilisateur de choisir les couleurs de texte, de liens, d'arrière-plan, d'ordonner et de réarranger les répertoires etc.
- **Personnalisation de la recherche d'information et du filtrage d'information :** Ces services permettent à l'utilisateur de rechercher et de filtrer de manière efficace.

Dans [96], les auteurs font une autre classification plus détaillée des méthodes de personnalisation dans les bibliothèques numériques (voir la figure 4.3). Selon eux, il y a deux grandes catégories de personnalisation :

- **Personnalisation de service :** La personnalisation de services se compose de deux sous-catégories : i) les services spéciaux comme le service de notification, des agents personnels ii) la personnalisation des propriétés des services comme la visualisation personnalisée ou la configuration individuelle des services du système.
- **Personnalisation de contenu :** La catégorie de personnalisation de contenu est divisée dans trois sous-catégories : i) l'enrichissement d'information qui utilise les recommandations, les annotations, les votes pour faciliter la décision sur la sélection ou l'utilisation de contenu ii) la sélection de contenu contient des méthodes personnalisées de filtrage et de recherche d'information iii) la structuration de contenu est la création de points d'entrée additionnels aux informations d'une bibliothèque numérique (par exemple, les structures supplémentaires de navigation personnelles basées sur les *patterns* de navigation fréquents de l'utilisateur).

4.3.2 Quelques systèmes actuels

Nous décrivons ci-dessous quelques approches pour personnaliser les bibliothèques numériques, ce sont des cas particuliers des approches générales que nous avons mentionnées dans la section 4.3.1 :

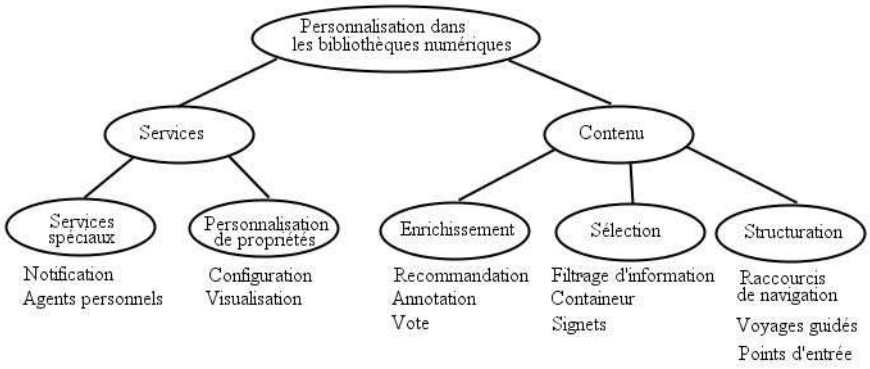

FIG. 4.3 – Classification des méthodes de personnalisation dans les bibliothèques numériques [96].

Interfaces adaptatives : Dans [119], les auteurs proposent d'utiliser des interfaces adaptatives pour les utilisateurs. En utilisant des techniques d'apprentissage automatique, le système classifie les utilisateurs dans trois classes différentes selon sa familiarité avec le système : *Novice*, *Expert*, et *Enseignant*[17]. Un type d'interface différent est proposé pour chaque classe d'utilisateur. Les utilisateurs dans la classe *Novice* utilisent l'interface de « carte de sujets » (*topic map interface*) qui fournit une vue globale du contenu sémantique de l'ensemble des documents de la bibliothèque numérique. Les utilisateurs dans la classe *Expert* utilisent l'interface basée sur un arbre (*tree-based interface*). En utilisant cette interface, l'utilisateur navigue dans la bibliothèque numérique à travers une structure d'arbre. Les utilisateurs dans la classe *Enseignant* se voient proposer une interface classique basée sur un formulaire (*form-based interface*). Cette interface donne des champs de saisie (*text box*) pour recevoir les requêtes correspondant avec les champs titre, abstract, article, nom, etc.

Service de notification : Un des services populaires est le *service de notification* (recommandation des documents) [1, 38] (cf. la section 3.1.2.3) : la bibliothèque notifie automatiquement un utilisateur chaque fois qu'il y a un nouveau document correspondant à son profil.

La bibliothèque numérique CiteSeer [13] (cf. la section 4.2.4) qui contient des articles scientifiques utilise un profil hétérogène pour représenter les intérêts d'un utilisateur. S'il y a un nouvel article, CiteSeer va calculer la similarité entre cet article et le profil utilisateur pour décider s'il va recommander cet article à l'utilisateur ou non. CiteSeer utilise deux méthodes pour déterminer si l'article est intéressant pour l'utilisateur : i) jeu de contrainte (*constraint matching*) et ii) similarité de propriétés (*feature relatedness*). Dans la première méthode, l'utilisateur peut décrire les caractéristiques qu'un article doit posséder pour qu'il lui soit présenté. Par exemple, il peut donner un mot-clé et si un nouvel article contient ce mot-clé, cet article sera recommandé à l'utilisateur. Il peut aussi donner une source du Web par son URL, si un nouvel article est lié à partir de cette source, il sera recommandé à l'utilisateur. Dans la deuxième méthode, l'utilisateur va spécifier un ensemble d'articles intéressants et CiteSeer va trouver des articles qui sont

[17] *Novice*, *Expert*, et *Teacher* en anglais.

reliés à cet ensemble. Les mesures de similarité que cette méthode utilise sont la mesure basée sur le contenu textuel TF-IDF et la mesure basée sur les citations CC-IDF. CC-IDF est partiellement similaire à TF-IDF, dans cette mesure la similarité entre deux articles scientifiques est basée sur le nombre de co-références entre eux (c'est aussi une variante de la méthode du *couplage bibliographique* qui est présentée dans le chapitre 5).

Torres et al. [139] présentent plusieurs algorithmes pour recommander des articles scientifiques : les méthodes collaboratives, les méthodes basées sur le contenu textuel et les méthodes hybrides. Dans leurs travaux, un profil utilisateur représente des intérêts à court terme et se compose d'un seul article. Les auteurs ont fait plusieurs expérimentations *offline* et *online* pour comparer les performances de ces méthodes et trouvent plusieurs résultats intéressants. Par exemple, les algorithmes hybrides peuvent générer des bonnes recommandations ; les différents utilisateurs ont différents niveaux de satisfaction avec les recommandations, les professionnels semblent être moins « satisfaits » que les étudiants.

Re-classement des résultats de recherche : Dans [114], les auteurs proposent quelques approches pour re-trier les résultats de recherche dans une bibliothèque numérique qui contient des livres numérisés. Dans leurs travaux, les auteurs considèrent deux types de recherche : recherche de livre en interrogeant sur les méta-données des livres (*Metadata Search* ou MS) et recherche dans le contenu textuel des pages des livres en utilisant des mots clés (*Content Search* ou CS). Ils utilisent deux types de profils utilisateurs correspondant à ces deux types de recherche : les profils MS et les profils CS. Un profil MS est construit à partir des votes sur les livres que l'utilisateur a fait explicitement. Un profil CS est construit à partir du contenu des pages que l'utilisateur a jugé pertinentes. Les résultats de recherche (MS et CS) sont triés à nouveau en utilisant ces profils.

Bibliothèques numériques comme un environnement collaboratif : Dans [110], les auteurs présentent un prototype d'une bibliothèque numérique collaborative personnalisée. Dans cet environnement, les utilisateurs peuvent organiser leurs espaces d'information dans leurs propres répertoires personnels. Il peuvent collaborer avec les autres utilisateurs en partageant leurs répertoires et peuvent recevoir des recommandations basées sur leurs préférences.

4.4 Bilan

Bien qu'il y ait eu plusieurs efforts pour personnaliser les bibliothèques numériques, les résultats obtenus sont encore limités. Comme nous avons abordé dans le chapitre 3 et dans ce chapitre, la plupart des systèmes personnalisés non-collaboratifs actuels utilise le contenu des éléments auxquels l'utilisateur s'est intéressé pour construire son profil et pour calculer les similarités entre profils et documents.

Nos travaux concernent la personnalisation des bibliothèques numériques qui contiennent des articles scientifiques. Dans ces bibliothèques, il y a d'autres informations utiles qui sont utilisables pour ces buts. Ce sont les relations bibliographiques entre les documents. Dans notre contexte sur les bibliothèques numériques, le profil de l'utilisateur ne seront

pas représentés seulement par le contenu des documents intéressants mais aussi par les citations de ces documents. En utilisant cette représentation, l'appariement profil-document peut-être calculé par les méthodes basées sur l'analyse de citations/liens. Les méthodes basées sur les citations ont une longue histoire d'être utilisées dans le domaine de bibliométrie. Dans le chapitre suivant, nous présentons les méthodes d'analyse de citations/liens et leurs applications dans la RI.

Chapitre 5

Analyse de liens et de citations et applications dans la recherche d'information

Plusieurs études sur l'analyse de citations ont été faites depuis longtemps dans le domaine de la *bibliométrie*. Avec l'apparition du Web, ces méthodes ont été utilisées également dans ce nouvel environnement en prenant en compte la similarité entre les liens Web (hyperliens) et les citations scientifiques. Le succès des moteurs de recherche sur le Web utilisant les méthodes d'analyse de liens est une preuve de l'utilité de ces méthodes à côté des méthodes basées sur l'analyse de contenu textuel traditionnelle.

5.1 Bibliométrie

Selon Borgman et al. [14], la bibliométrie[1] concerne la mesure spécifique des propriétés des documents et des processus associés aux documents[2]. Ce mot a son origine en langue grecque ancienne, « biblos » signifie « livre ». Bien que le mot français « bibliométrie » ait été utilisé en 1934 par Paul Otlet dans son « Traité de Documentation », il faut attendre jusqu'à l'année 1969 quand Pritchard donne une définition en anglais du mot « bibliometrics » pour que ce mot devienne largement connu et utilisé. Selon Pritchard, la bibliométrie est « l'application des méthodes mathématiques et statistiques aux livres et les autres moyens de communication » [3] [57].

La bibliométrie utilise tout un ensemble de méthodes mais ces méthodes peuvent être classifiées dans deux catégories principales : les méthodes basées sur l'analyse de citations[4] et les méthodes basées sur l'analyse de contenu textuel[5]. Comme exemples de méthodes d'analyse de citations, nous pouvons citer la méthode du couplage bibliographique, la méthode des co-citations, les réseaux de citations, la théorie de citation etc. Les applications de l'analyse de citations concernent les évaluations qualitatives et quantitatives

[1] *bibliometrics* en anglais.

[2] *The field whose concern is with the measurement specifically of properties of documents and of document-related processes, is known as bibliometrics.*

[3] *The application of mathematics and statistical methods to books and other media of communication.*

[4] *citation analysis* en anglais.

[5] *content analysis* en anglais.

des scientifiques, des publications et des institutions, la modélisation du développement historique de la science et de la technologie, la recherche d'information [36]. Du côté des méthodes d'analyse de contenu textuel sont l'analyse des mots associés[6], l'analyse de co-auteurs[7] l'analyse de fréquence de mots, etc. Cependant, comme nous nous intéressons aux méthodes basées sur les citations, nous n'allons pas trop aborder les méthodes basées sur l'analyse de contenu textuel dans ce chapitre.

Les recherches de la bibliométrie utilisent souvent des bases de données bibliographiques[8] (ou bases de données de citations) comme sources de données. Une base de données bibliographique est un système qui contient des notices bibliographiques des publications (articles, livres ...) et plusieurs d'autres informations concernant ces publications. Actuellement, la base de données bibliographique *Thomson ISI* est la base de données bibliographique la plus utilisée et souvent considérée comme une source de données standard pour les recherche de la bibliométrie. Nous allons présenter cette base de données en détail dans le chapitre suivant.

5.1.1 Références et citations

Isaac Newton a dit « si j'ai vu plus loin que les autres, c'est parce que j'ai été porté par des épaules de géants ». Cette expression reflète le fait que les travaux scientifiques sont basés sur les travaux antérieurs ou développés à partir de ces travaux. C'est une des principales raisons pour laquelle les articles scientifiques contiennent des références vers d'autres articles. Les citations qu'un travail reçoit est une mesure de l'importance de ce travail. Dans plusieurs cas on utilise les mots « citation » et « référence » de manière interchangeable. Cependant, on peut distinguer ces mots pour qu'ils soient plus clairs. Selon de Solla Price [28], « si l'article R contient une note bibliographique utilisant et décrivant l'article C, alors R contient une référence à C, et C a une citation de R ».

5.1.1.1 Motivation des citations

Depuis longtemps, il y a beaucoup de recherches sur le sujet de la motivation des « citeurs ». En fait, il y a beaucoup de raisons pour qu'un article cite un autre article. Garfield [42] a dénombré quinze raisons pour citer un article :

1. rendre hommage aux pionniers
2. approuver des travaux reliés
3. identifier des méthodologies, équipements, etc.
4. donner des fondements
5. corriger son propre travail
6. corriger les travaux des autres personnes
7. apporter des critiques aux travaux antérieurs
8. prouver des affirmations
9. donner des pistes travaux à venir

[6] *co-word analysis* en anglais.
[7] *co-author analysis* en anglais.
[8] *bibliographic database* ou *citation database* en anglais.

10. donner des liens vers les travaux mal diffusés, non indexés, ou non cités
11. authentifier les données et les faits, par exemples, les constantes physiques
12. identifier des publications originales dans lesquelles une idée ou un concept est discuté
13. identifier des publications originales ou les autres travaux décrivant un concept éponyme ou terme. Par exemple, maladie de Hodgkin, loi de Pareto etc.
14. démentir d'autres travaux ou idées
15. contester des affirmations de paternité des autres

A côté de cette liste, il y a d'autres travaux qui se concentrent sur les différentes motivations de citation. Ces travaux sont présentés dans [47, 36].

5.1.1.2 Graphe de citations

Dans une bibliothèque numérique ou une collection de documents, à partir des bibliographies des documents, nous pouvons construire un *graphe de citation* des documents dans une bibliothèque : chaque sommet dans le graphe représente un document, et chaque arc représente une citation d'un document vers un autre. Différents algorithmes peuvent être mis en œuvre pour calculer un poids pour chaque sommet qui reflète son importance (par exemple, PageRank [100]). Un tel graphe de citation a les caractéristiques suivantes [32] :

- C'est un graphe orienté et unidirectionnel.
- Il n'y a pas de cycle dans le graphe, parce qu'un document peut seulement contenir des références vers des documents publiés précédemment[9].
- Les sommets ne sont pas également distribué dans le graphe. Il y a des ensembles de sommets fortement connectés qui font des sous-graphes (chaque sous-graphe représente un sous-domaine).

Une fois que le graphe de citation a été construit, il est possible de l'utiliser pour trouver un sous-graphe correspondant avec un article, rechercher des articles importants, montrer des sujets d'actualité ... dans la bibliothèque.

5.1.2 Lois de la bibliométrie

Dans cette section nous abordons quelques lois connues de la bibliométrie [37] : la loi de Zipf concernant la distribution des mots dans les textes, la loi de Lotka concernant le nombre de publications des scientifiques, et la loi de Bradford concernant la répartition des revues en fonction de leur nombre d'articles.

Loi de Zipf : Selon cette loi, la fréquence d'occurrence d'un mot et son rang dans les textes sont liés par la formule : $fréquence \cdot rang = C$, où C est une constante. Cette liaison est indépendante des langues.

[9] Cette caractéristique n'est pas vérifiée dans le cas des pages Web.

Loi de Lotka : La loi de Lotka est un peu similaire à la loi de Zipf, mais cette loi concerne les publications des scientifiques. Selon cette loi, le nombre des auteurs ayant fait n publications est égale $\frac{1}{n^a}$ du nombre des auteurs ayant fait une publication. a est dépendant du domaine mais généralement a est aux environs 2.

Loi de Bradford : Cette loi a été décrite par Samuel C. Bradford en 1934 et concerne la répartition des revues en fonction de leur nombre d'articles. Il a trouvé que pour un sujet donné, un grand nombre d'articles appartiennent à un petit nombre de revues. Selon cette loi, « si les revues scientifiques sont rangés par ordre décroissant de leur productivité sur un sujet donné, ils peuvent être divisés en un noyau de revues qui sont plus particulièrement reliés au sujet et en plusieurs groupes ou zones contenant le même nombre d'articles que le noyau quand les nombres de revues dans le noyau et dans les zones successives serons : $1 : n : n^2 ...$ »[10]. L'étude de Bradford est importante pour les gestionnaires de bibliothèques, elle leur permet d'optimiser le nombre de leurs abonnements aux revues selon les centres d'intérêts de leur clientèle [55].

5.1.3 Les applications de la bibliométrie

De nos jours, la bibliométrie est un des rares domaines qui concerne presque toutes les autres domaines de recherche. Dans [136, 14], les auteurs présentent deux applications importantes de la bibliométrie qui utilisent des approches basées sur les citations : i) la bibliométrie évaluative concerne le jugement de l'impact des travaux scientifiques, par exemple la comparaison des contributions scientifiques de groupes ou d'individus différents pour répondre à la question « la recherche de qui est plus importante que celle de qui ? » ii) la bibliométrie relationnelle concerne la relation entre les documents, les revues, les groupes, organisations, ou nations etc. pour répondre à la question « qui est relié à qui ? ». Dans les sections suivantes nous présentons seulement les méthodes basées sur les citations/liens de ces applications bien que ces applications puissent utiliser également des méthodes lexicales.

5.1.3.1 Bibliométrie évaluative

Dans ce sous-domaine, le nombre de citations est souvent utilisé pour mesurer l'importance, la qualité, l'influence, ou la performance des documents, des personnes, des groupes, des domaines, ou des nations.

Évaluation des documents L'évaluation des documents est principalement basée sur le nombre de citations que les documents ont reçues. En 1990, Garfield [44] utilise la base de données bibliographique ISI pour analyser 175 millions de citations et donne une liste de 100 articles les plus cités. Il trouve que parmi 33 millions de publications citées (articles, livres, brevets ...), 500.000 documents (c'est-à-dire 2%) reçoivent plus de 50 citations,

[10] Bradford a déclaré : « *if scientific journals are arranged in order of decreasing productivity on a given subject, they may be divided into a nucleus of journals more particularly devoted to the subject and several groups or zones containing the same number of articles as the nucleus when the numbers of periodicals in the nucleus and the succeeding zones will be as* $1 : b : b^2$ *...* ».

et le nombre de documents qui reçoivent plus de 1000 citations chacun est seulement d'environ 1400 tandis que l'article le plus cité reçoit 187.652 citations.

Brooks [14] analyse 28 articles qui reçoivent le prix annuel pour le meilleur article de l'année de la revue *Journal of the American Society for Infomation Science* de 1969 à 1996. Il trouve que ces articles reçoivent un grand nombre de citations par rapport aux autres articles de cette revue, ce qui est probablement une évidence de la corrélation entre le nombre de citations et les jugements humains (par contre, on peut supposer que ces articles ont reçu beaucoup de citations parce qu'ils ont reçu le prix).

Évaluation des revues Depuis plus plusieurs années, *Thomson ISI* publie un rapport annuel qui s'appelle *Journal citation reports (JCR)* pour évaluer, classer, et comparer des revues scientifiques. Le facteur d'impact d'une revue[11] [43, 47] est une mesure importante utilisée dans ce rapport. Le facteur d'impact d'une revue dans l'année n est calculée de manière suivante :

$$IF_n(J) = \frac{c_n}{p_{n-1} + p_{n-2}}$$

Dans cette formule, c_n est le nombre de citations que les articles de la revue J publiés dans les années $n-1$ et $n-2$ ont reçues dans l'année n, $p_{n-1} + p_{n-2}$ est le nombre total des articles de la revue J dans les deux années $n-1$ et $n-2$. Par exemple, le facteur d'impact d'une revue dans l'année 2007 est le nombre des citations que les articles de cette revue publié dans les années 2005 et 2006 ont reçues dans l'année 2007, divisé par le nombre total des articles de cette revue dans les années 2005 et 2006. Le facteur d'impact n'est pas constant, il change chaque année.

Avec cette approche, le facteur d'impact peut éviter des biais causés par l'approche basée sur la fréquence absolue de citations. Par exemple, l'approche basée sur la fréquence absolue favorise les revues contenant plus d'articles, les revues qui paraissent plus fréquemment que celles qui paraissent moins fréquemment, ou les anciennes revues que les nouvelles revues. Cependant, il y a aussi des critiques avec cette approche [47] :

- Les facteurs d'impact ne prennent pas en compte les pratiques et traditions de références différentes selon les domaines de recherche.
- Il n'y a pas de distinction selon la nature et les mérites des revues qui font des citations.
- Les facteurs d'impact favorisent les revues ayant des longs articles. Par exemple, les revues ayant des articles de synthèse peuvent avoir de meilleurs facteurs d'impact.
- Le temps moyen entre la date de publication d'un article jusqu'à l'année où il reçoit le maximum de citations n'est pas toujours de deux ans. Selon Garfield, le fondateur de *ISI*, si l'on change la période pour calculer les facteurs d'impact, quelque revues auront de meilleurs facteurs d'impact.
- Le calcul des facteurs d'impact n'est pas toujours précis pour toutes les revues (mauvaise identification de certaines revues dans la base de données).

Évaluation des chercheurs Une des approches évidentes pour évaluer les chercheurs est de calculer le nombre de publications. La loi de Lotka représente la répartition des

[11] *Journal Impact Factor* en anglais.

auteurs en fonction de leur nombre de publication. Cependant, un article a normalement plusieurs co-auteurs. Quatre procédures de comptabilisation de la productivité des auteurs ont été présentées dans [55] :

1. Comptage normal (normal count) : il donne un crédit équivalent à tous les auteurs d'une même publication ; il y a donc comptabilisation pour un auteur de tous les articles dont il est signataire.
2. Paternité fractionnée (authorship fractional) : la contribution de l'auteur est pondérée par le nombre d'auteurs de l'article. La productivité de l'auteur est alors la somme de toutes ses participations aux publications.
3. Comptage direct (straight count) : seul le premier auteur reçoit la paternité de la publication.
4. Comptage direct modifié (modified straight count) : chaque publication est attribuée à un seul auteur, celui qui a la plus forte productivité.

Cependant, le nombre de publications n'est pas tout. Une publication d'Albert Einstein dans son année miracle 1905 est aussi importante qu'une centaine d'autres publications d'autres auteurs moins connus en physique. Une autre approche est de prendre en compte les citations que les chercheurs ont reçus pour leurs travaux. A côté des listes sur les articles qui sont les plus cités, ISI publie également les listes des auteurs les plus cités. On peut même utiliser ces listes pour prévoir les futurs lauréats du prix Nobel [14].

5.1.3.2 Bibliométrie relationnelle

Comme nous l'avons mentionné, la bibliométrie relationnelle concerne la question « qui est relié à qui ? ». Dans cette section, nous considérons deux méthodes très connues pour ce but : la méthode du couplage bibliographique et la méthode des co-citations.

Méthode du couplage bibliographique En 1963 Kessler [65] a proposé la méthode du *couplage bibliographique*[12]. Dans cette méthode, la similarité entre deux articles est basée sur leur nombre de co-références. Il a supposé que si deux articles ont des références communes, ils ont probablement un même sujet. Kessler a définit deux critères du couplage pour grouper les articles :

- Critère A : un ensemble d'articles constituent un groupe G_A si chaque membre de ce groupe a au moins une co-référence avec une article fixé P_o. La similarité (*coupling strength*) entre P_o et un membre de G_A est mesurée par le nombre de couplages (c'est-à-dire, co-références). G_{A^n} est le sous-ensemble de G_A qui contient des articles ayant n co-références avec P_o.
- Critère B : un ensemble d'articles constituent un groupe G_B si chaque membre de ce groupe a au moins une co-référence avec toutes les autres membres du groupe.

Kessler a considéré cette méthode comme un outil pour aider à la recherche de documents. Avec un article P_o donné, un système peut retrouver tous les articles qui partage des co-références avec P_o. Cette méthode a les propriétés suivantes :

1. Cette méthode est indépendante des mots et des langues.

[12] *bibliographic coupling* en anglais

2. On n'a pas besoin d'experts pour lire ou juger les documents. On n'a même pas besoin du texte des documents.
3. Le groupe des documents G_A peut être élargi. Par exemple, dans le futur, quand il y a plus de documents qui citent l'article P_o.
4. La méthode ne produit pas une classification statique pour un article donné. Le groupement évolue avec les changement de l'utilisation et les intérêts de la communauté scientifique.
5. Les membres du groupe G_A peuvent être considérés comme des « références logiques » de P_o. Cependant, Kessler n'a pas confirmé l'importance de cette stratégie.

La méthode du couplage bibliographique est illustrée dans la figure 5.1.

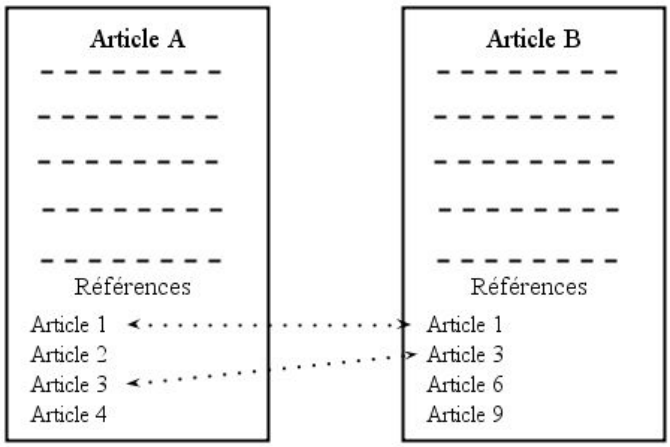

FIG. 5.1 – Illustration de la méthode du couplage bibliographique. Dans ce cas, deux articles A et B partagent deux co-références : l'article 1 et l'article 3.

Dans [36], Egghe et al. ont cité quelques critiques de cette méthode. Par exemple, une co-référence n'est pas une bonne mesure de la relation entre deux articles parce qu'ils peuvent citer la même article tandis qu'ils n'ont pas un même sujet. De plus, dans cette méthode la similarité de deux articles est fixée car elle est calculée en utilisant le nombre de co-références de ces articles qui ne peut pas changer après leur date de publication. Cet inconvénient n'existe pas dans la méthode des co-citations que nous allons aborder ci-dessous.

Méthode des co-citations En 1973, Marshakova [86] et Small [129] ont indépendamment proposé la *méthode des co-citations*. Dans cette méthode, la similarité entre deux articles est basée sur leur nombre de *co-citations* : c'est-à-dire le nombre de fois où ils sont co-cités. Deux articles sont co-cités s'ils apparaissent ensemble dans la bibliographie d'un autre article.

La méthode des co-citations est illustrée dans la figure 5.2.

Dans la méthode des co-citations, la similarité entre deux articles n'est pas fixée parce qu'avec le temps, deux articles similaires peuvent recevoir de plus en plus de co-citations. Cependant, cette méthode n'est pas applicable immédiatement après que deux articles sont publiés. Il faut attendre quelque temps pour que ces articles puissent recevoir des

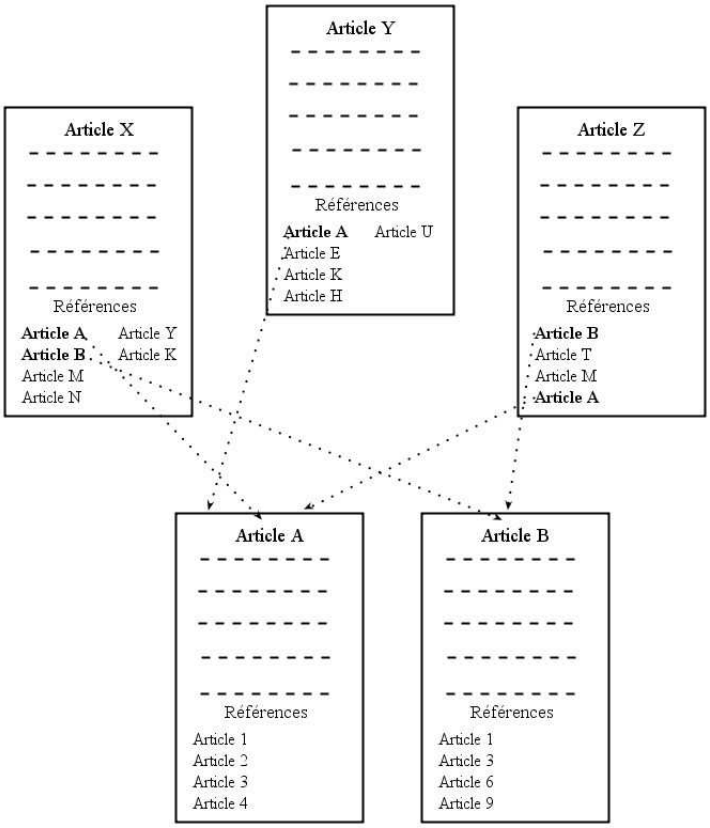

FIG. 5.2 – Illustration de la méthode des co-citations. Dans ce cas, deux articles A et B sont co-cités 2 fois par les articles X et Z tandis que l'article Y ne cite que l'article A.

citations. De plus, comme dans le cas de la méthode du couplage bibliographique, une co-citation de deux articles ne signifie pas toujours qu'ils ont un même sujet (partiellement ou entièrement).

Dans son article, Small a utilisé la méthode des co-citations pour montrer la relation entre les idées « clés » dans un domaine scientifique. Il a également montré quelques applications possibles de cette méthode dans la recherche d'information ainsi que l'utilisation de cette méthode pour surveiller le développement des domaines de recherche et pour évaluer le degré de relation entre les domaines.

5.2 Webométrie

Récemment, avec le développement de WWW, les chercheurs commencent à appliquer les méthodes de la bibliométrie et des sciences concernées sur ce nouveau domaine. Un nouveau domaine de recherche est né qui s'appelle *webométrie*. La webométrie est définie par Björneborn et al. [12] comme « les études des aspects quantitatifs de la construction et de l'utilisation des ressources d'information, structures et technologies sur le Web en héritant des approches bibliométriques et infométriques »[13]. Ils ont classifié les recherches de la webométrie dans quatre catégories principales suivantes :

- Analyse de contenu des pages Web.
- Analyse de structure des liens Web.
- Analyse de l'utilisation de Web (fichiers *log* et les habitudes de navigation des utilisateurs).
- Analyse des technologie Web (inclut la performance des moteurs de recherche).

Dans une recherche plus récente, Thelwall [136] classifie les recherches dans ce domaine dans cinq catégories :

- **Analyse des liens :** L'analyse des liens est l'étude quantitative des liens entre les pages Web, par exemple pour calculer des Facteurs d'Impact du Web[14] [59]. Les Facteurs d'Impact du Web sont similaires aux Facteurs d'Impact de Revues (cf. la section 5.1.3.1) mais appliqués pour les sites Web, pour les domaines Web, ou pour les pays. Le Facteur d'Impact d'un site Web ou d'un pays est le nombre des pages Web qui pointent vers ce pays ou ce site (incluant le nombre des pages internes mais contient au moins un lien vers le site ou le pays), divisé par le nombre des pages Web de ce site ou de ce pays.

- **Analyse des citations Web :** A côté de l'analyse des liens, un autre axe de recherche de la webométrie est d'analyser des citations Web en utilisant le Web pour estimer le nombre de fois où les articles de revues sont cités [70, 148, 147, 71]. Nous allons revenir sur ce problème en détail dans le chapitre suivant (cf. la section 6.3).

- **Evaluation des moteurs de recherche :** Un nombre important de recherches de la webométrie ont pour but d'évaluer les moteurs de recherche. Par exemple, pour estimer la couverture Web et l'exactitude de ces moteurs de recherche. Cependant,

[13] *The study of the quantitative aspects of the construction and use of information resources, structures and technologies on the Web drawing on bibliometric and informetric approaches.*

[14] *Web Impact Factors* en anglais

les recherches sur les algorithmes de recherche d'information et comment les moteurs de recherche sont utilisés n'appartiennent pas à la webométrie.

- **Description de Web :** Il y a des recherches qui sont purement descriptive dans la webométrie. Par exemple, pour estimer la taille moyenne des pages Web, le nombre moyen et les types des méta-balises, le nombre des utilisateurs etc. En 2000, Broder et al. [18] ont analysé environ 200 millions de pages Web et ont trouvé un modèle pour le Web. Selon ce modèle, le Web se compose de cinq composants (figure 5.3). Le premier composant est un coeur de 56 millions de pages (*SCC - Strongly Connected Component*) dans laquelle une page Web est connectée avec n'importe quelle autre page Web dans le même composant. Le diamètre de ce composant est de 28 liens. Le deuxième composant est l'ensemble des pages Web entrantes (*IN*) qui peuvent atteindre le *SCC* mais ne peuvent pas être atteintes à partir des pages du *SCC*. Le troisième composant est l'ensemble des pages Web sortantes (*OUT*) qui sont accessibles à partir des pages du *SCC*, mais ne peuvent pas atteindre le *SCC*. Le quatrième composant est noté *TENDRILS*. Ce sont des pages Web qui ne peuvent pas atteindre les pages Web du *SCC* et ne sont pas non plus accessibles à partir du *SCC*. Les trois composants *IN*, *OUT* et *TENDRILS* contiennent environ le même nombre de pages Web (44 millions pour chaque composant). Le dernier composant se compose d'environ 16 millions pages Web déconnectées.

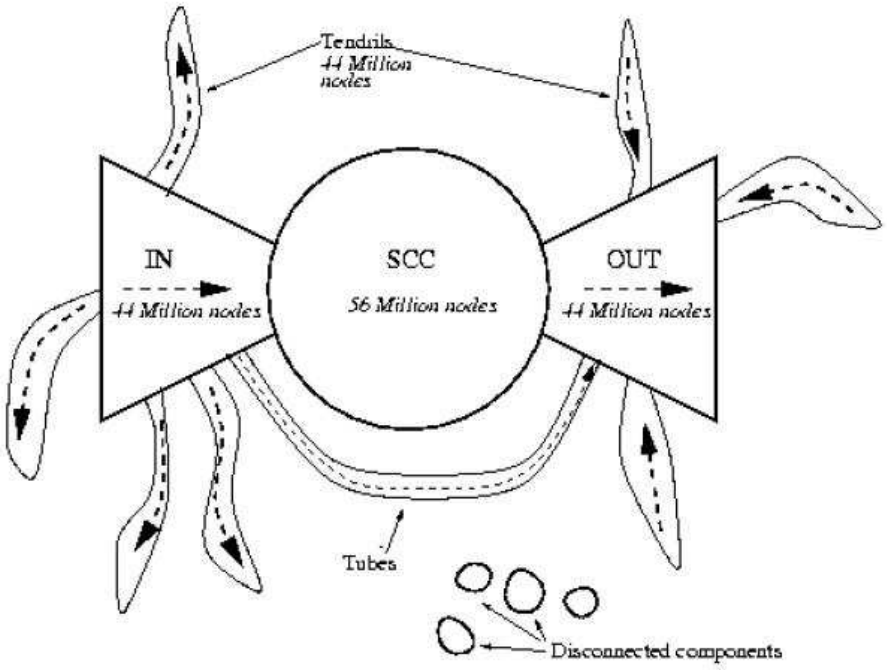

FIG. 5.3 – Structure du Web [18].

- **Mesure du Web 2.0 :** Le terme Web 2.0 désigne une technologie pour enrichir la création, le partage d'information et la collaboration entre les utilisateurs. Nous pouvons citer à titre d'exemples quelques applications du Web 2.0 comme les blogs, le Wikipédia, les sites de réseaux sociaux etc. La mesure du Web 2.0 signifie la fouille

de données sur ces pages Web pour extraire des « patterns » utiles ou la description et l'explication de comportement des utilisateurs dans cet environnement. Par exemple, Gruhl et al. [50] proposent d'utiliser le volume de discussions sur les blogs pour prévoir les ventes des livres.

5.2.1 Similarités et différences entre liens Web et citations scientifiques

Les liens Web sont assez similaires aux citations scientifiques. Dans [116], les auteurs ont listé quelques similarités entre ces entités :

- Les liens Web sont une forme de connexion entre documents, comme les citations
- De plus, les liens Web dans les articles des revues en-lignes sont plus analogues aux citations que les autres liens généraux.
- Un article peut être mentionné dans des pages Web qui ne sont pas elles-mêmes des articles de revues. Ce type de citation est appelé « citation Web »[15].
- Les citations normales peuvent être trouvé en ligne dans les sections de références des articles des revues électroniques (*e-journal*) et dans les copies numériques des revues académiques ou des articles de conférences.

Cependant, il y a aussi plusieurs différences entre les liens Web et les citations scientifiques. Une des plus grandes différences est que le Web n'est pas le produit d'un processus de contrôle de qualité comme les publications scientifiques. Presque tout le monde peut publier des pages Web et créer des liens entre elles. De ce fait la nature des liens Web est plus compliquée que celle des citations scientifiques. Le graphe de citations scientifiques est un graphe acyclique parce qu'un article ne peut citer que des articles publiés auparavant. Cependant, dans le graphe des liens Web, deux pages Web peuvent mutuellement se citer.

5.2.2 Bibliométrie, scientométrie, infométrie, cybermétrie, et webométrie

Il y a souvent des confusions entre les termes bibliométrie, scientométrie, infométrie, cybermétrie, et webométrie. En fait, ce sont des domaines de recherche différents. Nous citerons ci-dessous quelques définitions concernant la scientométrie et l'infométrie extraites de [57, 11].

Définition de la cybermétrie

- Définition de Björneborn : la cybermétrie est l'étude des aspects quantitatifs de la construction et de l'utilisation des ressources d'information, structures et technologies sur l'ensemble de l'Internet, en héritant des approches bibliométriques et infométriques[16].

[15] *Web citation* en anglais.

[16] *The study of the quantitative aspects of the construction and use of information resources, structures and technologies on the whole Internet, drawing on bibliometric and informetric approaches.*

Définitions de la scientométrie

- Définition de Tague-Sutcliffe : la scientométrie est l'étude des aspects quantitatifs de la science en tant que discipline ou activité économique. Elle est une part de la sociologie de la science et s'applique à l'élaboration des politiques scientifiques. Elle concerne les études quantitatives des activités scientifiques, incluant, parmi d'autres, la publication, et donc partage quelques aspects avec la bibliométrie[17].

Définitions de l'infométrie

- Définition de Tague-Sutcliffe : l'infométrie est l'étude des aspects quantitatifs de l'information sous n'importe quelle forme, pas seulement les notices ou les bibliographies, et dans n'importe quel groupe social, et pas seulement les scientifiques[18].

- Définition de Ingwersen et Christensen : le terme infométrie désigne une extension récente des analyses bibliométriques traditionnelles pour couvrir également les communautés non scientifique dans les quelles l'information est produite, transmise, et utilisée[19].

Comme nous l'avons mentionné au-dessus, la bibliométrie concerne notamment les publications scientifiques, la webométrie concerne l'utilisation des méthodes de la bibliométrie dans l'environnement Web. La cybermétrie couvre la webométrie ; c'est un domaine plus large que la webométrie parce que « certaines activités dans le cyber-espace ne sont normalement pas enregistrées, mais elles sont assurées de manière synchrone, comme dans les salons de discussion (*chat room*) »[20] [137]. La scientométrie concerne les aspects de la science et de la technologie en général, et pas seulement les publications scientifiques. L'infométrie est le plus large domaine parmi ces domaines qui concerne toutes les formes d'information. La relation entre ces domaines est illustrée dans la figure 5.4 [12].

5.3 Applications des méthodes d'analyse de citations et de liens

Dans cette section, nous abordons quelques applications des méthodes d'analyse de citations et de liens dans le cadre de la recherche d'information et les domaines apparentés.

5.3.1 Classification et regroupement

Les méthodes des co-citations et du couplage bibliographique sont utilisées fréquemment pour la catégorisation des documents. Il existe deux types de catégorisation : i) la

[17] *Scientometrics is the study of the quantitative aspects of science as a discipline or economic activity. It is part of the sociology of science and has application to science policy-making. It involves quantitative studies of scientific activities, including, among others, publication, and so overlaps bibliometrics to some extent.*

[18] *Informetrics is the study of the quantitative aspects of information in any form, not just records or bibliographies, and in any social group, not just scientists.*

[19] *The term informetrics designates a recent extension of the traditional bibliometric analyses also to cover non-scholarly communities in which information is produced, communicated, and used.*

[20] some activities in cyberspace are not normally recorded, but communicated synchronolously, as in chat rooms.

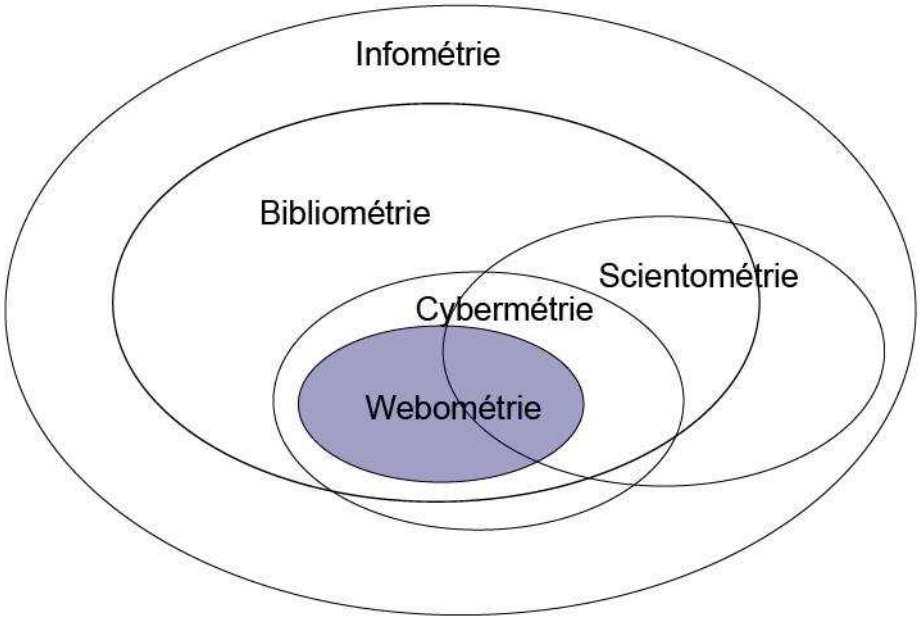

FIG. 5.4 – Infométrie, bibliométrie, scientométrie, cybermétrie, et webométrie [12].

catégorisation dans des classes prédéfinies, ou classification et ii) catégorisation quand il n'y a pas des classes prédéfinies, ou regroupement[21]. Par exemple, dans [75], la méthode des co-citations est utilisée pour un système de classification de brevets. D'autres travaux [105, 108] utilisent cette méthode pour regrouper des pages Web.

5.3.2 Algorithme PageRank

Actuellement, le moteur de recherche Google[22] est le système dominant pour la RI sur l'Internet. Une des raisons principales pour le succès de ce système par rapport aux autres moteurs de recherche de l'époque est qu'il a appliqué un nouvel algorithme pour calculer l'« importance » des pages Web. Cet algorithme est basée sur l'analyse de liens et il s'appelle *PageRank* [17, 100]. Bien que *PageRank* ne soit pas le seul critère utilisé pour trier des résultats, il joue un rôle très important pour la performance de Google.

Intuitivement, dans *PageRank*, une page Web est considérée comme importante si : i) elle est pointée par plusieurs autres pages Web ou ii) elle est pointée par une (ou plusieurs) pages Web importantes. Dans une forme simple, le *PageRank* d'une page A est calculé par la formule suivante :

$$PR(A) = (1-d) + d(\frac{T_1}{C(T_1)} + ... + \frac{T_n}{C(T_n)})$$

Dans cette formule, d est un facteur *damping* qui est dans l'intervalle de 0 et 1. $T_1...T_n$ est l'ensemble des pages Web qui pointent vers A. $C(T_i)$ est le nombre de liens émises/sortantes de la page T_i. Le calcul est un processus récursif et il est itéré jusqu'à la convergence.

[21] *clustering* en anglais.
[22] http://www.google.com

Selon les auteurs de *PageRank*, il est un modèle du comportement de l'utilisateur. Le *PageRank* d'une page Web peut être considéré comme la probabilité pour qu'un utilisateur la visite en naviguant aléatoirement sur le Web, le facteur *damping* est la probabilité pour que à chaque page cet utilisateur devienne ennuyé et demande aléatoirement une autre page Web.

5.3.3 Algorithme HITS

L'algorithme *HITS*[23] proposé par Kleinberg [67] est un autre algorithme connu qui est basé sur l'analyse des liens pour la recherche d'information sur le Web. Cet algorithme calcule deux scores pour chaque page Web : un score *hub* et un score *authority*. Une page avec un bon score *authority* est une page qui contient de bonnes informations concernant la requête. Une page avec un bon *hub* est une page qui pointe vers les bonnes pages *authorities* ; à l'inverse, une bonne page *authority* est pointée par plusieurs bonnes pages *hubs*. La relation entre ces deux types de pages Web est une relation mutuelle. Cependant, au contraire de l'algorithme *PageRank*, cet algorithme n'est pas appliqué sur toutes les pages Web dans le graphe mais pour seulement un sous-ensemble des pages Web. Les étapes principales de cet algorithme peuvent être résumé comme suit :

1. Envoyer une requête à un moteur de recherche et obtenir un ensemble R_σ qui contient les t premiers résultats.
2. Élargir R_σ en ajoutant des pages qui sont pointées par R_σ et au maximum d pages Web qui pointent vers R_σ. Le graphe de ces pages Web est noté $G[\sigma]$. On enlève tous les liens qui connectent les pages ayant le même nom de domaine dans $G[\sigma]$ et on obtient un autre graphe G_σ.
3. Exécuter la procédure *Itérer* sur cet ensemble étendu pour calculer le score *hub* et le score *authority* pour chaque page. Cette procédure est décrite dans la figure 5.5.
4. Renvoyer les bonnes pages *authorities* et *hubs*

Les deux algorithmes *HITS* et *PageRank* sont tout deux basés sur l'analyse des liens Web. Cependant, il y a des différences entre ces algorithmes :

- *HITS* est appliqué sur un sous-ensemble de pages Web tandis que *PageRank* est appliqué sur tout le graphe Web.
- *HITS* est exécuté au moment du traitement de la requête tandis que *PageRank* est exécuté auparavant.
- *HITS* calcule deux scores (*hub* et *authority*) pour une page Web, tandis que *PageRank* calcule seulement un score.
- Les scores de *hub* et *authority* pour chaque page Web sont dépendants de la requête de l'utilisateur, mais le score *PageRank* est indépendant de la requête de l'utilisateur.

5.3.4 Utilisation de contexte de citations

Quand un auteur fait une citation vers un autre article, il doit décrire (de manière brève) cet article dans le contexte de son article. Cette description peut servir pour mieux

[23] *Hyperlink-Induced Topic Search* en anglais.

Itérer(G, k)
G : une collection de n pages liées
$(p, q) \in E$ signifie qu'il existe un lien de p à q
$x^{<p>}$, $y^{<p>}$ sont des scores *authority* et *hub* de page p
x est le vecteur représentant des scores $\{x^{<p>}\}$
y est le vecteur représentant des scores $\{y^{<p>}\}$
k : nombre entier
z : vecteur (1,1, ... ,1) $\in R^n$
$x_0 := z$
$y_0 := z$
For i = 1,2 ..., k
$x_i^{<p>} \leftarrow \sum_{q:(q,p)\in E} y_{i-1}^{<q>}$
$y_i^{<p>} \leftarrow \sum_{q:(p,q)\in E} x_i^{<q>}$
Normaliser les vecteurs x_i et y_i
End
Return (x_k, y_k)

FIG. 5.5 – Calcul de *hubs* et de *authorities*.

comprendre l'article cité. Dans [15], Bradshaw et al. ont indexé les documents d'une bibliothèque numérique en utilisant le contexte de citations des articles. C'est-à-dire utiliser les textes autour des citations que les autres articles ont fait vers un article pour indexer cet article. Selon les auteurs, le contexte de références est utile pour l'indexation dans un système d'information parce qu'il est concis et contient des informations riches qui décrivent l'article cité.

Similairement, dans le système Google [17], les textes d'ancre autour des hyperliens d'une page Web sont propagés vers les pages Web citées par cette page Web. Les auteurs considèrent que le « texte d'ancre fournit souvent des descriptions plus exactes pour les pages Web que le contenu lui-même »[24]. Cependant, dans quelques cas, cette approche peut causer des problèmes. Par exemple, pour un liens dans le contexte suivante : « le Président de la République est en train de visiter les États-Unix. Pour en savoir plus, cliquer ici ». Dans ce cas là, le texte d'ancre « cliquer ici » ne peut être considéré comme utile pour décrire la page citée.

A côté de ces systèmes, Craswell et al. [26] utilisent les textes d'ancre pour trouver le point d'entrée[25] des sites Web et ils trouvent que les textes d'ancres sont plus efficace que le contenu entier des documents dans cette tâche. La bibliothèque numérique CiteSeer présente également des articles avec le contexte dans lequel ces articles sont cités pour permettre de mieux comprendre ces articles.

5.3.5 Stratégies de recherche basée sur les citations

A côté des stratégies de recherche d'information basées sur le contenu textuel, il y a des stratégies de recherche basées sur les citations. Par exemple, dans les bibliothèques

[24] anchors often provide more accurate descriptions of web pages than the pages themselves.
[25] *main entry point/home page* en anglais

numériques ou les bases de données bibliographique, il y a souvent des fonctionnalités de recherche des articles reliés par rapport avec un article donné. Dans [77], l'auteur a résumé une liste des stratégies de recherche basée sur les citations :

- *Backward chaining* : retrouver les références dans la bibliographie d'un document donné.
- *Forward chaining* : retrouver les documents qui citent un document pertinent donné. Le document pertinent est appelé le *seed document* (en anglais) et on n'a pas besoin d'accéder à son contenu.
- *Citation cycling* : le processus commence à partir d'un ensemble des références des documents connus retrouvés par le processus *backward chaining*. Ces références sont ensuite utilisés dans un processus de *forward chaining*.
- *Uncontrolled subject search* : d'abord on utilise les mots dans les titres (ou les abstracts, ou la liste des mots-clés) des articles pour retrouver les autres documents. Puis, ces documents sont utilisé dans un processus de *citation cycling*.
- *Controlled subject search* : d'abord on retrouve quelques documents en utilisant des descripteurs à partir de l'index d'un domaine. Puis, ces documents sont utilisé dans un processus de *citation cycling*.
- *Highly cited document search* : retrouver des documents ayant un nombre important de citations basé sur un certain critère. Par exemple, les documents qui sont cités fréquemment dans un domaine spécifique.
- *Bibliographical coupling search* : retrouver des documents qui ont des co-références avec un document pertinent.
- *Co-citation search* : retrouver des documents qui sont co-cités avec les documents pertinents connus.

5.3.6 Quelques autres applications

Les méthodes basées sur les citations/liens sont également utiles pour trouver des documents similaires, c'est-à-dire des documents qui traitent du même sujet. Par exemple, Dean et al. [29] utilisent la méthode des co-citations pour trouver des pages Web similaires. Dans [88], cette méthode est utilisée pour recommander des articles scientifiques. La bibliothèque numérique CiteSeer utilise les méthodes du couplage bibliographique et des co-citations pour trouver des articles scientifiques reliés.

Dans [35], Efron utilise la méthode des co-citations pour estimer l'« orientation politique » des pages Web, c'est-à-dire la probabilité qu'une page Web appartient à la gauche ou à la droite. Son modèle estime cette probabilité en calculant les co-citations de cette page Web avec deux ensembles des pages Web de la gauche et de la droite.

La découverte de communautés d'intérêts sur le Web[26] utilisant les méthodes d'analyse de liens est aussi un axe intéressant de la Webométrie. Ce sont des groupes de pages Web de personnes qui partagent les mêmes intérêts. Il y a plusieurs raisons pour découvrir ces communautés [74]. D'abord, ces communautés peuvent fournir des ressources importantes, fiables et mises à jour pour les personnes qui s'intéressent à ces sujets. De plus, elles représentent la sociologie du Web, étudier ces communautés peut nous donner des aperçus dans l'évolution intellectuelle du Web. Enfin, la découverte de ces communautés peut

[26] *cyber-communities* en anglais

servir à la publicité d'une manière précise (bien qu'il y ait des personnes qui n'aiment pas ça). Parmi les travaux dans cette axe nous pouvons citer à titre d'exemples [109, 74].

5.4 Les méthodes basées sur les citations/liens sont-elles toujours bonnes ?

Bien que les méthodes basées sur les citations aient été utilisées depuis longtemps et le succès des moteurs de recherche qui utilisent l'analyse de liens comme Google, ces méthodes ne sont pas toujours efficaces.

5.4.1 Dérive de sujet

Dans [58], les auteurs présentent un phénomène courant qui peut dégrader la performance des méthodes basées sur les citations/liens. Ce phénomène s'appelle « dérive de sujet »[27] : à cause de différents buts de citations, dans un article les citations ne sont pas concentrées dans le même sujet. Par exemple, dans la figure 5.6, les articles 3 et 4 sont co-cités et donc liés. Les articles 5, 6, et 7 sont cités par 3 et 4, donc on peut supposer que ces articles sont liés. Cependant, dans ce cas seulement les articles 5 et 6 sont liés, et l'article 7 a un sujet différent par rapport aux articles 5 et 6. Les auteurs proposent également une méthode qui permet de résoudre ce problème en prenant en compte le contexte de citations.

5.4.2 Caractéristiques des collections de documents

Dans [150], les auteurs examinent la performance de plusieurs méthodes basées sur le contenu et sur les liens (l'algorithme *HITS*) sur la collection WT10g de TREC. Ils trouvent que les méthodes basées sur le contenu textuel sont bien meilleures que les méthodes basées sur les liens. De plus, les méthodes de fusion ne donnent pas beaucoup d'amélioration. Ils constatent également que plusieurs autres méthodes utilisant cette collection qui combine les méthodes basées sur le contenu et sur les liens ne sont pas aussi bonnes qu'espéré. Selon les auteurs, l'échec des méthodes de fusion peut être lié avec les caractéristiques de la collection WT10g, la faiblesse de l'analyse de liens ou des formules de fusion, ou une combinaison de ces raisons. Dans [26], les auteurs montrent que dans plusieurs expérimentations de TREC, surtout dans les tâches Web de TREC-8, les méthodes de liens ne sont pas meilleures que les méthodes basées sur le contenu textuel. Dans [51], les auteurs essaient d'améliorer les résultats les méthodes basées sur le liens en utilisant un sous-ensemble de la collection WT10g. Ce sous-ensemble contient des pages Web ayant une densité de liens plus élevé que dans la collection originale. Les auteurs montrent que les méthodes basées sur les liens utilisant ce sous-ensemble peuvent donner quelques améliorations par rapport aux méthodes basées sur le contenu textuel.

Dans [25], les auteurs utilisent les méthodes des co-citations et du couplage bibliographique pour classifier des pages Web et des articles scientifiques. Ils trouvent que la méthode des co-citations est très efficace pour classifier les pages Web mais elle ne fonctionne pas aussi bien quand elle est utilisée pour classifier des articles scientifiques. Par

[27] *topic drift* en anglais

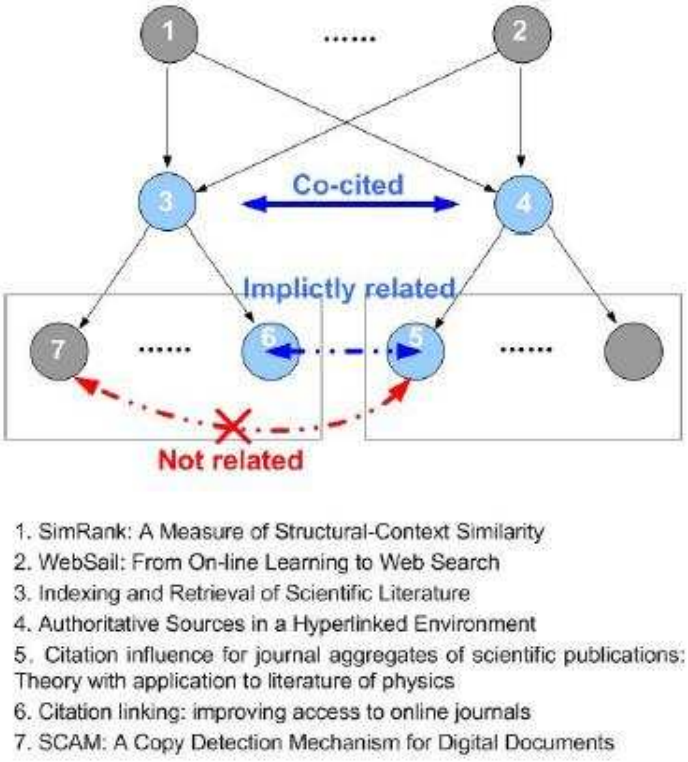

FIG. 5.6 – Dérive de topic [58].

contre, la méthode du couplage bibliographique semble être meilleure que la méthode des co-citations pour la classification des articles. Selon les auteurs, ce phénomène peut être expliquée par les caractéristiques des collections. Dans la collection des articles scientifiques, les articles citent plusieurs autres articles à l'intérieur ou à l'extérieur de la collection. Cependant, seulement des citations vers les documents externes sont disponibles tandis que les citations à partir d'autres documents externes vers les articles dans cette collection ne sont pas disponibles. Le nombre des articles internes qui sont citées par plusieurs autres articles sont rares, et donc les co-références semblent être plus nombreuses que les co-citations, surtout quand la couverture de cette collection n'est pas grande. Par ailleurs, dans la collection de pages Web, beaucoup de pages Web n'ont pas de liens sortants tandis que peu de pages n'ont pas de liens entrants. Cette collection est un sous-ensemble d'une base de données (index) d'un grand moteur de recherche et donc les informations concernant ces pages Web peuvent être extraites à partir de cette base de données. De plus, comme cette collection est issu d'un répertoire Web connu alors les pages Web de cette collection sont plus « visibles » et donc susceptibles de recevoir plus de citations. Dans ce cas le nombre des co-citations est plus élevé que le nombre de co-références entre les pages Web. Ce fait explique pourquoi la méthode des co-citations est meilleure que la méthode du couplage bibliographique dans cette collection.

5.5 Bilan

Dans ce chapitre, nous venons de présenter les méthodes basées sur les citations qui ont été utilisées depuis longtemps dans la bibliométrie et – depuis les années récentes – dans la webométrie. Nous présentons également quelques applications de ces méthodes dans la recherche d'information et les domaines apparentés. De plus, nous montrerons des problèmes potentiels avec ces méthodes.

La plupart des systèmes personnalisés actuels n'utilisent que des méthodes et approches basées sur le contenu textuel pour représenter les documents et les profils utilisateurs et pour calculer les similarités entre eux. Pour nos travaux, nous nous intéressons également aux méthodes de la bibliométrie relationnelle pour trouver la relation entre les articles scientifiques. Elles sont la méthode du couplage bibliographique et la méthode des co-citations que nous avons présentées dans la section 5.1.3.2. Dans le chapitre suivant, nous allons présenter notre approche pour personnaliser le service de RI des bibliothèques numériques et l'utilisation de ces méthodes dans nos travaux pour calculer les similarités documents-profils.

Chapitre 6

Notre approche pour la recherche d'information personnalisée dans les bibliothèques numériques

Dans ce chapitre nous présentons nos travaux concernant la personnalisation de la RI dans les bibliothèques numériques. Tout d'abord, dans la section 6.1 nous présentons l'idée de base et la procédure de personnalisation. Dans les deux sections suivantes nous abordons les matières et méthodes nécessaires pour nos travaux : les bases de données bibliographiques et la méthode des co-citations sur le Web. Enfin nous présentons les méthodes de combinaison pour combiner les scores afin de re-trier les résultats.

6.1 Introduction à notre approche

Comme nous l'avons mentionné dans les chapitres 3 et 4, il y a plusieurs manières pour personnaliser les systèmes d'information : re-classement des résultats de recherche (recherche d'information personnalisée), filtrage de documents (service d'alerte dans les bibliothèques numériques), visualisation personnalisée de documents etc. Nos travaux se concentrent sur la recherche d'information personnalisée dans les bibliothèques numériques qui contiennent des articles scientifiques. Dans cette approche, après que l'utilisateur ait soumis une requête à un moteur de recherche, la liste des premiers résultats retournés par le moteur de recherche sont re-triés en prenant en compte leurs similarités avec le profil utilisateur. Nous choisissons d'étudier le problème de recherche d'information personnalisée parce que c'est un sujet important dans les bibliothèques numériques. De plus, notre apport concerne principalement l'utilisation des méthodes de citations pour calculer la similarité document-profil. Ces méthodes peuvent être appliquées facilement dans d'autres applications de personnalisation comme dans la construction d'un service d'alerte dans les bibliothèques.

Les étapes principales utilisées dans nos travaux pour la recherche d'information personnalisée sont décrites ci-dessous (voir la figure 6.1) :

1. Une requête utilisateur est soumise au moteur de recherche de la bibliothèque.

2. Le moteur de recherche renvoie des résultats avec les scores correspondants.

3. Les n premiers documents sont sélectionnés pour être re-triés. Dans nos expérimentations, $n = 300$.

4. Le système calcule les similarités de ces documents avec le profil utilisateur en utilisant différentes méthodes. Supposons qu'un utilisateur a un domaine d'intérêt, le profil utilisateur est un ensemble d'articles intéressants pour cette utilisateur[1]. La similarité document-profil entre un document d à re-trier et un profil p est la somme des similarités entre ce document et les autres documents d' dans le profil utilisateur :

$$similarité(d, p) = \sum_{d' \in p} similarité(d, d') \tag{6.1}$$

   La $similarité(d, d')$ peut être calculée par une des trois méthodes suivante : une méthode basée sur le contenu textuel (cosinus) et deux méthodes basées sur les citations (co-citations et couplage bibliographique). Les bases de données ISI Web of Science et le Web sont utilisées dans la méthode des co-citations. Notons que dans la méthode des co-citations et la méthode du couplage bibliographique, nous pouvons calculer seulement la similarité entre deux documents. C'est une raison pourquoi nous avons choisi de représenter le profil utilisateur par un ensemble d'articles pour faciliter ce calcul (cf. la formule 6.1).

5. Le score original d'un document calculé par le moteur de recherche est combiné avec les similarités document-profil. Nous utilisons plusieurs méthodes de combinaison dans cette étape : combinaison produit, combinaison linéaire, et la combinaison basée sur la théorie Dempster-Shafer. Le score final obtenu après cette étape est utilisé pour re-trier les documents.

6. Enfin, la liste re-triée de documents est présentée à l'utilisateur.

Comme nous l'avons mentionné dans le chapitre 3, il y a plusieurs étapes différentes pour construire des systèmes personnalisés : sélection de modèle de représentation de profils utilisateurs, acquisition de données d'utilisateurs, construction et mise à jour de profil utilisateur, et utilisation de profil utilisateur pour différents buts. Dans nos travaux, nous nous concentrons plutôt sur le dernier aspect. Nous représentons le profil d'un utilisateur par un ensemble d'articles que cet utilisateur a trouvé intéressants. Cette représentation est similaire à l'approche utilisée dans le système CiteSeer [13].

L'originalité de nos travaux consiste à l'utilisation des méthodes basées sur les citations pour calculer la similarité entre les documents à re-trier et le profil utilisateur à côté des méthodes basées sur le contenu textuel. Ce sont la méthode du couplage bibliographique

[1] Cette approche peut être élargie facilement pour les utilisateurs ayant plusieurs domaines d'intérêt différents. Par exemple, pour les utilisateurs ayant plusieurs domaines d'intérêt, nous pouvons appliquer des méthodes de regroupement/classification pour diviser l'ensemble d'articles intéressants de cet utilisateur en plusieurs groupes/catégories différentes qui représentent différents domaines d'intérêt d'utilisateur.

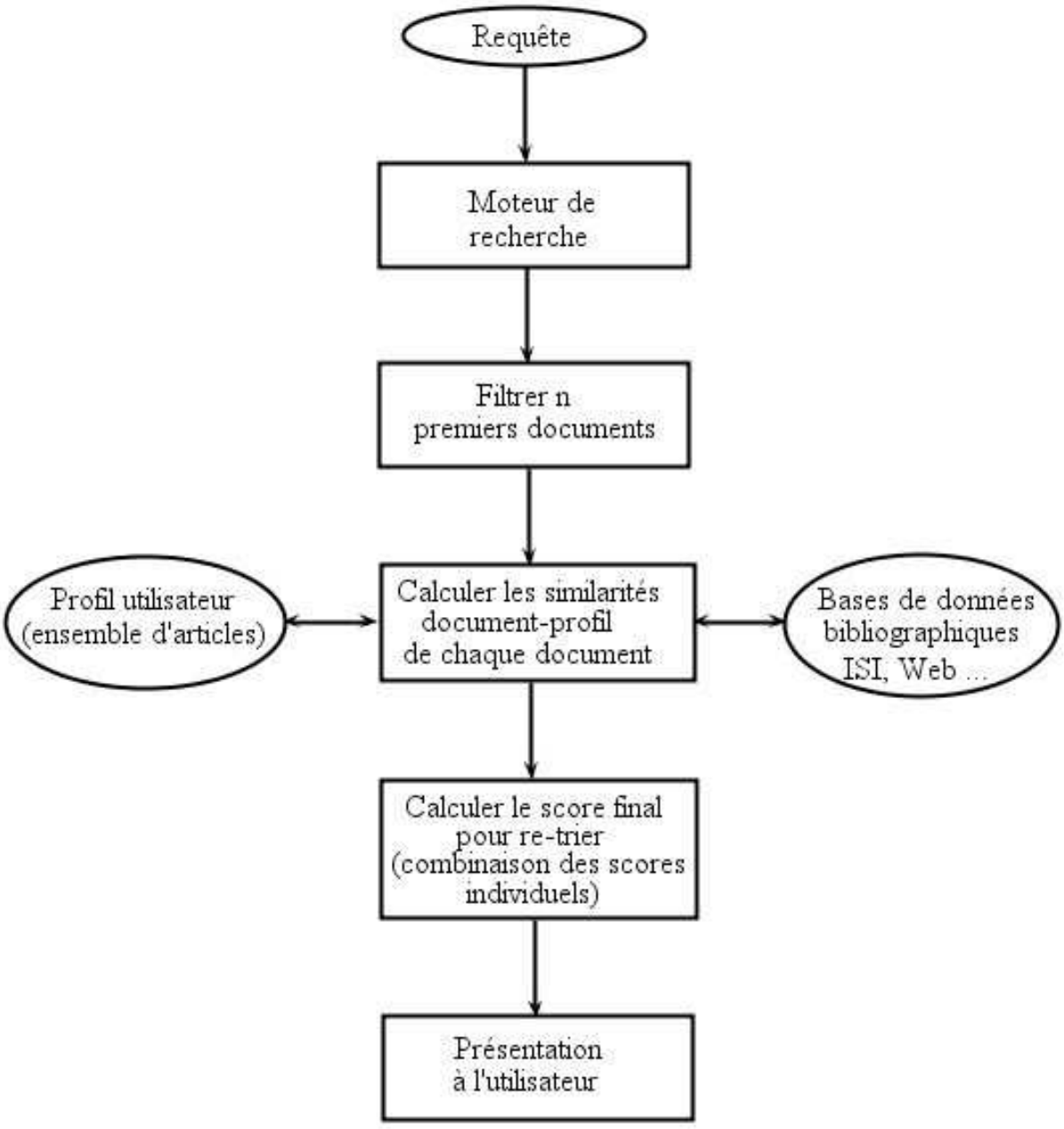

FIG. 6.1 – Recherche d'information personnalisée.

et la méthode des co-citations que nous avons présentées dans le chapitre précédent (cf. la section 5.1.3.2).

Dans la méthode du couplage bibliographique, la similarité entre deux documents (c'est-à-dire la $similarité(d, d')$ dans la formule 6.1 ci-dessus, d est un document à retrier et d' est un document dans le profil utilisateur) est basée sur le nombre de co-références entre ces deux articles. On peut extraire les références des articles à partir de son texte et trouver les références communes. Cependant, dans la méthode des co-citations, la similarité entre eux est basée sur le nombre des co-citations de ces articles. Pour calculer ce nombre, il faut d'abord connaître les articles qui citent chacun de ces deux articles, puis trouver les articles qui co-citent ces deux articles ensemble. Pour faire ça, il faut extraire le graphe de citations de la collection de documents de la bibliothèque ou utiliser une base de données bibliographique pour avoir ces informations. Comme nous avons discuté dans la section 5.4.2, l'utilisation du graphe de citation interne de la bibliothèque numérique peut être inefficace si la collection n'est pas suffisamment grande. Alors nous proposons d'utiliser les bases de données bibliographiques externes dans la méthode des co-citations pour trouver la relation entre les articles. Cependant, pour les grandes bibliothèques numériques (par exemple CiteSeer), l'extraction du graphe de citation interne peut être suffisante. Actuellement, la base de données bibliographique la plus utilisée dans les études bibliométriques est la base de données de Thomson ISI. Nous utilisons donc cette base de données dans nos travaux. De plus, nous proposons également d'utiliser le Web comme une base de données bibliographique. La méthode des co-citations qui utilise le Web comme base de données de bibliographique pour calculer la similarité entre deux articles s'appelle la méthode des co-citations sur le Web.

Le reste de ce chapitre est organisé comme suit. Nous présentons les bases de données bibliographiques, surtout la base de données de Thomson ISI, dans la section 6.2. Ensuite, dans la section 6.3 nous abordons la méthode des co-citations sur le Web. Les formules de mesure de similarité sont présentées dans la section 6.4. Enfin nous présentons les méthodes de combinaison dans la section 6.5. Ces méthodes sont utilisées pour combiner les différents scores : le score original calculé par le moteur de recherche, les similarité document-profil calculés par les méthodes du couplage bibliographique et des co-citations.

6.2 Bases de données bibliographiques

Comme nous l'avons présenté brièvement dans le chapitre 5, une base de données bibliographique est une collection d'informations organisées concernant les articles de revues, les articles de conférences, ou les livres etc. Les informations stockées sont diverses : titre de l'article, résumé, références, informations sur les auteurs, date de publication etc. Une base de données bibliographique peut contenir ou non le contenu de ces publications. Par exemple, le système CiteSeer fournit à la fois le contenu des articles scientifiques et les autres informations concernant les articles. Les bases de données bibliographiques sont utilisées pour rechercher des publications/travaux. Les bases de données bibliographiques connues comme celle de Thomson ISI[2] sont également utilisées pour tracer le développement de la science, pour évaluer ou comparer les domaines, les travaux, les équipes de recherche etc. (cf. la section 5.1.3).

[2] http://www.isiwebofknowledge.com/

Il y a des bases de données bibliographiques spécialisées, comme *MEDLINE* pour les sciences biologiques et biomédicales, *INSPEC* dans le domaine de la physique, de l'électronique, du génie électrique, et de l'informatique. Il y a également des bases de données bibliographiques générales qui couvrent plusieurs domaines, comme Thomson ISI ou Scopus[3].

Indexation de citations et indexation automatique de citations L'indexation des citations (*citation indexing*) est une mission importante des bases de données bibliographiques. Elle permet de naviguer et tracer les publications scientifiques (en arrière en suivant les références ou vers l'avant en suivant les citations). Selon Lawrence et al. [78], l'indexation de citations peut améliorer les communications scientifiques par :

- montrer les relations entre les articles ;
- attirer l'attention aux corrections ou rétractions importantes des travaux publiés ;
- identifier les améliorations ou critiques des travaux précédents ;
- éviter la duplication des recherches précédentes.

Normalement, l'indexation de citations demande un travail manuel. Récemment, un autre type d'indexation de citations a apparu : ce sont des systèmes d'indexation de citation autonome (*Autonomous Citation Indexing* - ACI). Le système CiteSeer est un représentant typique de ces systèmes. Il cherche les articles sur l'Internet, extrait les références de ces articles, et construit le graphe de ces articles de manière automatique. Cette approche réduit nettement les efforts nécessaires pour construire des systèmes d'indexation de citations.

6.2.1 *Web of Science* de Thomson ISI

Une des bases de données bibliographiques que nous utilisons est la base de données bibliographique de Thomson ISI. L'Institute for Scientific Information (ISI) a été créé en 1960 par Eugene Garfield, un pionnier dans les domaines de la bibliométrie et de la scientométrie. Il a été acquis par le groupe Thomson en 1992. Plus précisément, dans nos expérimentations nous utilisons le Web of Science (WoS) qui est la version Web des index principaux de Thomson ISI. Chaque index est une base de données bibliographique spécialisée dans un grand domaine. Les bases de données bibliographiques couverts par le WoS sont : *Science Citation Index Expanded* (SCIE, 1900-présent) qui est une extension de *Science Citation Index* (SCI), *Social Science Citation Index* (SSCI, 1956-présent), *et Arts & Humanities Citation Index* (AHCI, 1975-présent). Récemment, le WoS inclut également deux autres bases de données : Index Chemicus (1993-présent) et Current Chemical Reactions (1986-présent).

Le WoS est une base de données importante qui a été largement utilisée dans les études bibliométriques [60, 89, 55, 47]. Elle permet aux utilisateurs de rechercher des informations à propos de plusieurs disciplines dans plus de 8700 revues scientifiques les plus prestigieuses et influentes du monde. Le SCIE couvre environ 5900 revues (SCI couvre 3500 revues), le SSCI couvre 1700 revues, et l'AHCI couvre 1100 revues [47]. Le WoS fournit plus de 1,1 million d'articles et plus de 23 millions de références citées par an. Selon Peter Jacso [60], en 2005 le WoS contient environ 35 millions d'enregistrements.

[3] http://www.scopus.com

Le WoS est une part d'un autre plus large base de données *Web of Knowledge* (WoK). Le WoK couvre toutes les bases de données du WoS ainsi que plusieurs autres bases de données comme *ISI Proceedings*, *Derwent Innovations Index*, *Current Contents Connect* etc.

Le WoS fournit en ensemble de fonctionnalités pour l'accès et la recherche dans cette base de données[4], par exemple :

- Analyser les résultats d'une recherche (l'outil *Analyze*).
- Définir des alertes de citation.
- Accéder aux références (et connaître leur nombre) pour un article.
- Accéder au contenu.
- Fonctions de personnalisation avancée pour la sélection ou l'affichage.

De plus, les fonctionnalités de WoS permettent aux utilisateurs de naviguer sur le graphe de citations des articles :

- Remonter dans le temps en utilisant des références citées dans un article.
- Avancer dans le temps avec *Times Cited* pour trouver les articles qui citent un article.
- Découvrir des relations « cachées » entre des articles sans liens apparents pouvant être ignorés par une recherche traditionnelle par sujet.

La fonctionnalité la plus importante pour nos travaux est que les utilisateurs peuvent rechercher les articles citant un article donné dans cette base. Comme nous l'avons mentionné au-dessus, c'est nécessaire pour calculer le nombre de co-citations de deux articles.

6.2.1.1 Accès au *Web of Science*

Le WoS fournit une API qui permet l'accès au WoS sans l'utilisation d'un navigateur[5]. Un article dans le WoS est identifié par un identifiant unique appelé **ut**. Quelques opérations importantes de l'API du WoS sont décrites dans le tableau 6.1. La figure 6.2 est un exemple d'un enregistrement dans WoS.

Opération	**Description**
searchRetrieve	Exécution d'une recherche pour obtenir les enregistrements des articles et leurs identifiants **ut**.
citingArticles	Recherche des articles qui citent un article prédéfini qui est identifié par son identifiant **ut**.

TAB. 6.1 – Deux opérations importantes de l'API du WoS.

Grâce au service de recherche du WoS, à partir des informations concernant un article (titre, année de parution ...), il est possible de rechercher l'identifiant **ut** correspondante en utilisant la fonction *searchRetrieve*. Puis en utilisant cet identifiant comme un paramètre pour la fonction *citingArticles* on identifie les articles qui le citent. Avec ces informations, nous pouvons savoir le nombre de citations d'un article ou le nombre de co-citations de deux articles dans la base de données WoS. Dans nos travaux, nous utilisons l'API du WoS au lieu d'utiliser un navigateur Web pour accéder au WoS.

[4] http://www.thomsonscientific.com/frwok/wosdetails/

[5] http://scientific.thomson.com/support/faq/webservices

```
<RECORDS>
<REC inst_id="22" recid="111295780" hot="yes" sortkey="3236110423"
timescited="5" sharedrefs="0" inpi="false">
<ut>000081993000012</ut>
<source_title>IEEE TRANSACTIONS ON PATTERN ANALYSIS
AND MACHINE INTELLIGENCE</source_title>
<item_title>Multiprimitive segmentation of planar curves - A
two-level breakpoint classification and tuning approach</item_title>
<bib_id>21 (8) : 791-797 AUG 1999</bib_id>
<bib_issue year="1999" vol="21"/>
<authors count="2">
<primaryauthor>Sheu, HT</primaryauthor>
<author key="3149832">Hu, WC</author>
</authors>
<abstract avail="Y" count="1">
<p>A breakpoint classification and tuning approach is proposed for
the multiprimitive segmentation of planar curves, and cockhead-like
graph is suggested to evaluate the multiprimitive segmentation
algorithms. The breakpoints are divided into corners and smooth
joints and the types of the segments on both sides of a breakpoint
are identified. Then, a joint tuning procedure is exercised to
merge/split segments and adjust the joint locations. The carefully
designed cockhead-like graph includes all possible combinations and
parameters of line and are segments and serves as a benchmark to
test the algorithms. The proposed scheme is simple, fast, threshold-free
and robust to quantization and preprocessing errors, thus allowing it
to be employed in a variety of applications such as matching and
recognition. Test against the suggested benchmark and comparison with
those in the literature assures the superiority of the method suggested
herein.</p>
</abstract>
<categories count="2">
<category>Computer Science, Artificial Intelligence</category>
<category>Engineering, Electrical & Electronic</category>
</categories>
<headings count="1">
<heading>Multidisciplinary Science & Technology</heading>
</headings>
</REC>
</RECORDS>
```

FIG. 6.2 – Exemple d'un enregistrement du WoS.

6.2.1.2 Avantages et inconvénients du *Web of Science*

Parmi les trois bases de données du WoS, la base de données la plus importante c'est la SCI/SCIE. Selon Glänzer [47], cette base de données a comme avantages :

- *Multidisciplinarité :* Tous les domaines de recherche dans les sciences de la vie, les sciences naturelles, les mathématiques et l'ingénierie sont représentés.
- *Sélectivité :* Les revues couvertes par la SCI sont choisis sur la base de critères quantitatifs (mesures d'impact), et la sélection est généralement confortée par les avis des experts.
- *Couverture complète :* Tous les articles publiés dans des revues couvertes par la SCI sont enregistrées.
- *Complétion des adresses :* Les adresses de tous les auteurs sont indiquées, ce qui permet l'analyse de la collaboration scientifique et le comptage des publications.
- *Références bibliographiques :* Les références des documents sont également traitées et incluses dans SCI.
- *Disponibilité :* Le SCI est disponible sous plusieurs formes : en édition imprimée, sous forme électronique, ou sur des CD-ROM.

Cependant, il y a également des inconvénients avec le WoS [54] :

- Dans quelques cas, Web of Science peut fournir une sous-estimation des citations vers les travaux scientifiques. Par exemple, dans une étude récente, Nisonger [97] trouve que le WoS capture seulement 29,8 % des citations vers ses travaux. Les raisons principales concernent la couverture du WoS, par exemple le WoS ne couvre pas beaucoup les revues qui ne sont pas en anglais.
- Le WoS couvre peu de publications en sciences humaines et sociales.
- Le WoS est un système commercial, contrairement aux systèmes gratuits comme CiteSeer ou Google Scholar.

6.2.2 Quelques autres bases de données bibliographique

A côté des bases de données bibliographiques que nous venons de présenter, il y a beaucoup d'autres bases de données comme Scopus[6] et des bibliothèques numériques comme CiteSeer ou ACM[7] qui peuvent fournir également des informations bibliographiques sur des articles scientifiques. La base de données Scopus déclare sur son site Web[8], être « la plus grande base de données de citation et d'abstracts des publications « peer-reviewed ». Actuellement, Scopus couvre plus de 15.000 revues « peer-reviewed » de la science, technologie, médecine, et sciences sociales de plus de 4000 éditeurs internationaux. Cependant, jusqu'à maintenant, la base de données ISI est encore la base de données dominante pour les recherches dans le domaine de la bibliométrie. C'est pourquoi nous l'utilisons pour nos travaux actuels.

[6] http://www.scopus.com/scopus/home.url
[7] http://portal.acm.org/portal.cfm
[8] http://www.info.scopus.com/faq/

6.3 La méthode des co-citations sur le Web

Avec l'explosion du WWW, les moteurs de recherche sur le Web deviennent de plus en plus complets pour satisfaire les besoins d'information des utilisateurs. Par exemple, en 2005 le moteur de recherche Google a indexé environ 8 milliards de documents Web. Avec leurs grands index, les moteurs de recherche sur le Web peuvent devenir des bons outils pour plusieurs tâches de fouille de données. Par exemple, Turney et al. [141] ont utilisé le moteur de recherche AltaVista pour trouver l'orientation sémantique des mots. Une orientation sémantique positive implique le désir (par exemple, « beauté », « honnête » etc.) tandis qu'une orientation sémantique négative implique une indifférence (par exemple, « superflue », « absurde » etc.). La méthode de Turney est basée sur une méthode d'information mutuelle (*Pointwise Mutual Information* - PMI). Dans la méthode PMI de base, l'information mutuelle entre deux mots mot_1 et mot_2 est définie de la manière suivante :

$$PMI(mot_1, mot_2) = \log_2 \left(\frac{p(mot_1 \& mot_2)}{p(mot_1)p(mot_2)} \right)$$

Dans cette formule, $p(mot_1 \& mot_2)$ est la probabilité que ces deux mots apparaissent ensemble. Cette fonction mesure la dépendance entre ces deux mots. Si deux mots sont statistiquement indépendants, alors $p(mot_1 \& mot_2) = p(mot_1)p(mot_2)$.

Dans le travail de Turney et al., l'orientation sémantique d'un mot est calculée à partir de son niveau d'association avec un ensemble des mots « positifs » (*good, nice, excellent, positive, fortunate, correct, superior*), diminué par son niveau d'association avec un ensemble des mots « négatifs » (*bad, nasty, poor, negative, unfortunate, wrong, and inferior*) :

$$\begin{aligned} Orientation_sémantique(mot) \quad = \quad & Association(mot, \{paradigme\ positif\}) \\ & - Association(mot, \{paradigme\ négatif\}) \end{aligned}$$

Dans cette formule, le paradigme positif représente sept mots positifs et le paradigme négatif représente sept mots négatifs. Ces niveaux d'association sont calculés à partir des nombres de co-occurrences des mots. L'algorithme de Turney est l'algorithme PMI-IR (*Pointwise Mutual Information and Information Retrieval*). Cet algorithme estime l'information mutuelle entre les mots en envoyant les requêtes à un moteur de recherche et note le nombre de résultats renvoyés. Turney estime orientation sémantique d'un mot par la formule suivante :

$$Orientation_sémantique(mot) = \log_2 \left(\frac{hits(mot\ NEAR\ p_query) hits(n_query)}{hits(mot\ NEAR\ n_query) hits(p_query)} \right)$$

Dans cette formule, *hits* est le nombre de résultats renvoyés pour une requête ; $p_query = (good\ OR\ nice\ OR\ ...\ OR\ superior)$; $n_query = (bad\ OR\ nasty\ OR\ ...\ OR\ inferior)$. Un mot est considéré comme avoir une orientation sémantique positive si la valeur calculée par cette formule est positive, et il aura une orientation sémantique négative si la valeur calculée par cette formule est négative.

6.3.1 Citations sur le Web

Récemment, une nouvelle méthode pour l'analyse des citations des articles scientifiques appelée *citations sur le Web*[9] [147, 148] commence à attirer l'attention de la communauté de bibliométrie. Cette méthode utilise les moteurs de recherche pour analyser les citations des articles scientifiques au lieu d'utiliser les bases de données bibliographiques traditionnelles comme Thomson ISI. La méthode des *citations sur le Web* permet de trouver les citations d'un article de la manière suivante : on envoie une requête contenant le titre de cet article (recherche d'une expression exacte en utilisant des guillemets) à un moteur de recherche sur le Web et on analyse les pages retournées. Les citations sur le Web et les hyperliens Web diffèrent parce qu'une citation Web est une mention d'une publication dans une page Web tandis qu'un hyperlien est créé pour faciliter la navigation sur le Web. Puisqu'un moteur de recherche sur le Web peut indexer les documents de formats et de types différents, la notion de « citation » ici est une « relaxation » par rapport à la notion traditionnelle.

Vaughan et al. [147] ont utilisé cette méthode avec le moteur de recherche Google et comparé avec la méthode traditionnelle qui utilise la base de données de citation Web of Science. Deux collections d'articles sont utilisées dans leurs expérimentations : la première collection se compose de 1209 article de recherche publiés en 1997 qui appartiennent à 46 revues du domaine des sciences de l'information et des bibliothèques (*information and library science*) ; la deuxième collection se compose de 554 articles publié en 1992 qui appartiennent à 15 revues avec de bons facteurs d'impact. Les pages Web citant ces articles sont classifiées selon leurs pays, leurs types de domaine Web (.com, .org, .edu, et « inconnu »), et leur classe de *source de citation*. Cependant, pour la deuxième collection, on applique seulement la classification selon les *source de citation*. Les pays et les types de domaine sont identifié à partir des URLs de ces pages Web. Les *source de citation* ce ces pages Web sont classifiées sous 7 catégories différentes :

- Revue : le site Web de la revue de cet article, ou le site de l'éditeur de cette revue.
- Auteur : les sites Web de l'auteur, des co-auteurs, ou de leurs employeurs. On prend en compte également le CV de l'auteur ou le site Web de son institution.
- Service : un service bibliographique Web qui liste cet article ; par exemple DBLP ou ResearchIndex.
- Classe : liste de lecture/bibliographie d'un cours.
- Article : un article scientifique citant cet article qui est disponible sur le Web.
- Conférence : annonce, rapport, description d'une conférence.
- Autres : les autre types de page Web.

Dans leurs travaux, Vaughan et al. ont amené plusieurs conclusions intéressantes :

- Il y a une corrélation entre citation bibliographique du WoS et citation sur le Web.
- Le nombre de citations sur le Web est beaucoup plus élevé que le nombre de citations du WoS. Par exemple, le nombre moyen de citations sur le Web d'un article JASIS[10] dans la première collection (1997) est 34, tandis que le nombre moyen de citations de l'ISI est seulement 5.
- Il y a une corrélation entre les facteurs d'impact de revues (*Journal Impact Facteurs*) et les nombres moyens des citations sur le Web de ces revues (le nombre total de

[9] *Web citations* en anglais.

[10] *Journal of the American Society for Information Science*

citations sur le Web qu'une revue reçoit divisé par le nombre d'articles de cette revue).
- Les services bibliographiques et les citations par les articles disponible sur le Web sont les deux principales sources de citations.

Kousha et al. [71] utilisent une stratégie similaire qui s'appelle *citations par URL*[11] pour trouver les citations vers les articles des revues d'accès libre. Cependant, dans leurs travaux, la citation d'une page Web est la mention de son hyperlien dans le texte d'une autre page Web au lieu de son titre, c'est peut-être un hyperlien vers l'article si son URL apparaît dans le texte d'ancre ou l'inclusion de cet URL dans la page, même si ce n'est pas un hyperlien. L'exemple ci-dessous illustre la recherche des citations par URL vers un article de la revue *Cybermetrics* :

Titre de l'article : *LOTKA : A program to fit a power law distribution to observed frequency data*
URL de l'article : *http://www.cindoc.csic.es/cybermetrics/articles/v4i1p4.html*
Requête Google : *http://www.cindoc.csic.es/cybermetrics/articles/v4i1p4.html -site:www.cindoc.csic.es*

Il est nécessaire d'utiliser la commande *-site*: pour exclure les hyperliens qui viennent du même domaine que l'article parce que la plupart de ces hyperliens sont créés dans un but de navigation. Cette approche est différente de l'approche qui utilise la commande *link*: des moteurs de recherche pour trouver les pages Web qui pointent vers une autre page Web. Elle exclut les pages Web où l'URL n'est pas explicitement mentionné. Cette méthode a le potentiel d'identifier les communications scientifiques formelles parce que dans la plupart des citations formelles, l'URL de l'article cité est apparu dans le texte de l'hyperlien. Comme Vaughan et al., les auteurs classifient les citations vers un article dans quatre grandes catégories :

- URLs qui sont équivalents aux citations académiques comme les articles scientifiques, les rapports de recherche, les thèses/mémoires, les livres etc.
- URLs qui sont reliés aux activités académiques mais non formelles comme les listes de lecture d'une classe, CV, enregistrements dans les services bibliographiques etc.
- URLs qui ont pour but la navigation.
- Les autres types d'URLs.

Les auteurs trouvent que le premier type de citations représente un taux important de 43% de toutes les citations. Les auteurs comparent également cette approche avec l'approche d'analyse de citations en utilisant la base de données Web of Science. Comme Vaughan et al., ils trouvent également qu'il y a une corrélation entre le nombre des citations par URL et le nombre des citations dans la base de données WoS. De plus, les citations par URL du premier type sont beaucoup plus nombreuses que les citations du WoS.

[11] *URL citation* en anglais.

6.3.2 Co-citations sur le Web

Comme dans les études de citations sur le Web, nous avons l'intention d'utiliser les moteurs de recherche pour analyser les relations bibliographiques entre les articles scientifiques. Précisément, nous voulons utiliser ces moteurs pour estimer le nombre de co-citations des articles scientifiques sur le Web. Dans notre méthode des co-citations sur le Web, nous calculons la similarité de co-citation entre deux articles scientifiques par le nombre de fois où ils sont « co-cités » sur le Web en utilisant le moteur de recherche Google pour trouver le nombre de pages Web qui mentionnent les deux articles ensemble.

La notion de « co-citation » dans ce contexte est aussi une « relaxation » par rapport à la définition traditionnelle. Si le document Web qui mentionne les deux articles est aussi un article scientifique, alors ces deux articles sont normalement co-cités. Cependant, si c'est le programme d'une conférence, nous pouvons dire que ces deux articles sont co-cités et ils ont une relation parce qu'une conférence a normalement une thématique à laquelle se rapporte les communications qui y sont présentées. Si deux articles sont apparus dans la même conférence, ils ont probablement le même sujet général. Similairement, si deux articles sont dans la liste de lecture d'un même cours, ils ont probablement une relation avec le sujet général du cours. En bref, si deux articles sont mentionnés dans le même document Web, nous pouvons supposer qu'ils ont une relation, bien que cette relation puisse être faible ou forte.

Le moteur de recherche que nous utilisons dans nos expérimentations est Google. Google a été choisi parce qu'il est sans aucun doute le système dominant pour la recherche d'information sur le Web. Pour trouver la fréquence à laquelle un article est « cité » par Google, nous envoyons le titre de cet article (recherche d'une expression exacte en utilisant des guillemets) à Google et notons le nombre de documents retournés. Similairement, pour trouver le nombre de fois où deux article sont « co-cités », nous envoyons les titres de ces deux articles (dans une même requête) à Google et notons le nombre de documents retournés. Cette idée est illustrée dans la figure 6.3. Dans cet exemple, le nombre de co-citations de deux articles est 11. Dans nos expérimentations, nous utilisons un script pour interroger automatiquement Google au lieu d'utiliser manuellement un navigateur Web.

FIG. 6.3 – Illustration de la méthode des co-citations sur le Web avec Google

6.3.3 Avantages et inconvénients de la méthode des co-citations sur le Web

Le plus grand avantage de la méthode des co-citations sur le Web est lié à sa couverture. Le Web est une « base de données » énorme qui contient des milliards de pages Web de plusieurs types différents. Dans les travaux que nous avons présentés dans la section 6.3.1, les citations sur le Web sont beaucoup plus nombreuses que les citations traditionnelles dans la base de données WoS. Nous pouvons donc supposer que l'utilisation du Web peut être plus efficace pour la découverte de la relation entre les articles dans la méthode des co-citations, c'est-à-dire trouver plus de documents Web qui « co-citent » les deux articles.

Par contre, l'inconvénient principal de cette méthode est que la nature des citations Web est très variée, ce qui peut (potentiellement) causer des problèmes de bruits pour cette méthode. Comme nous l'avons mentionné, les notions de « citation » et de «co-citations» sur le Web sont des « relaxations » par rapport aux définitions traditionnelles de ces notions. Cependant, nous supposons que si deux articles sont « co-cités » sur le Web, ces deux articles ont un certain niveau de relation, bien que dans quelques cas, cette relation puisse être moins « forte » que si ces articles sont co-cités par un autre article scientifique comme dans les approches traditionnelles.

Google Scholar Google Scholar[12] est un moteur de recherche spécialisé de Google pour les travaux universitaires. Actuellement, il est encore en version bêta. Google Scholar couvre plusieurs types de documents (articles de revues/conférences, thèses/mémoires, rapports ...) qui viennent de plusieurs sources (éditeurs scientifiques, sociétés savantes, Web ...). Google Scholar fournit les fonctionnalités principales suivantes[13] :

- Rechercher différentes sources à partir d'une interface unique.
- Trouver des articles, des résumés analytiques et des citations.
- Localiser un article complet.
- Consulter les articles « clés » dans n'importe quel domaine de recherche.

Bien qu'il y ait eu des critiques sur Google Scholar dans le passé [60], c'est un outil potentiel pour les études bibliométriques [7, 70, 89]. Bauer et al. [7] montrent que pour les articles de la revue JASIST publiée en 2000, Google Scholar fournit un nombre de citations qui est beaucoup plus élevé que celui fourni par Web of Science ou Scopus. De plus, contrairement à la méthode de citations sur le Web, la plupart des travaux couverts par Google Scholar sont des publications scientifiques. Nous avons eu l'intention de l'utiliser dans nos travaux mais ce système a un mécanisme très strict contre les interrogations multiples à partir d'un même endroit. Alors nous réservons l'étude avec Google Scholar pour les travaux futurs.

6.4 Mesures de similarité

Comme nous l'avons mentionné au-dessus, un profil utilisateur p est un ensemble d'articles intéressants pour l'utilisateur, la similarité document-profil entre un document

[12] http://scholar.google.com/
[13] http://scholar.google.fr/intl/fr/scholar/about.html

d et un profil p est la somme des similarités entre ce document et les autres documents d' dans le profil utilisateur :

$$similarité(d,p) = \sum_{d' \in p} similarité(d,d') \tag{6.2}$$

La $similarité(d,d')$ est calculée par trois méthodes : la méthode des co-citations, la méthode du couplage bibliographique, et la méthode cosinus. Pour calculer la similarité entre d et d' dans la méthode cosinus, nous utilisons le moteur de rechercher Zettair (cf. la section 2.5.2) pour indexer la collection de document qui contient d' et puis on envoie d comme une requête[14] à Zettair et note la similarité retournée.

Pour calculer la similarité entre d et d' dans la méthode des co-citations, après une comparaison préliminaire entre plusieurs formules, nous proposons d'utiliser une variante de la formule présentée dans [108] :

$$cocitation_sim(d,d') = ln\left(\frac{cocitation(d,d')^2}{citation(d) \times citation(d')}\right) \tag{6.3}$$

Dans cette formule, $cocitation(d,d')$ est le nombre de co-citations de d et d' ; $citation(d)$ et $citation(d')$ sont respectivement les nombres de citations de d et d'. Dans la méthode des co-citations sur le Web, nous avons trouvé que les nombres des citations Web sont beaucoup plus grands que le nombre des co-citations Web, nous modifions cette formule de la manière suivante :

$$cocitation_sim(d,d') = ln\left(\frac{cocitation(d,d')^2}{citation(d) + citation(d')}\right) \tag{6.4}$$

Similairement, dans la méthode du couplage bibliographique, la similarité entre deux documents d et d' est calculée par :

$$bibcoupling_sim(d,d') = ln\left(\frac{coreference(d,d')^2}{reference(d) \times reference(d')}\right) \tag{6.5}$$

Dans la formule 6.5, $coreference(d,d')$ est le nombre de co-références entre d et d' ; $reference(d)$ and $reference(d')$ sont respectivement les nombres de références de d et d'.

Dans les méthodes des co-citations et du couplage bibliographique, la similarité document-profil (cf. la formule 6.2) a une valeur négative. Alors nous la convertissons en une valeur positive afin de pouvoir la combiner avec d'autres scores en utilisant la formule suivante :

$$similarité'(d,p) = \frac{1}{|similarité(d,p)|} \tag{6.6}$$

Dans la formule 6.6, $|similarité(d,p)|$ signifie la valeur absolue de $similarité(d,p)$. La conversion dans la formule 6.6 garantit que si $similarité(d_1,p) > similarité(d_2,p)$ alors nous avons également $similarité'(d_1,p) > similarité'(d_2,p)$.

[14] Zettair permet de changer la taille maximale d'une requête.

6.5 Combinaison des scores

Comme nous l'avons mentionné, les scores finaux pour re-trier les résultats de recherche sont une combinaison de plusieurs scores différentes : le score original calculé par le moteur de recherche pour les documents dans la liste des n premiers documents et les similarités document-profil calculées par différentes méthodes (cosinus, couplage bibliographique, co-citations). Dans nos travaux, nous utilisons les fonctions de combinaisons suivantes pour combiner les scores : combinaison produit, combinaison linéaire, et combinaison basée sur la théorie de Dempster-Shafer. Ces fonctions ont été présentées dans la section 2.4.

Combinaison produit

$$score_final(d) = \prod_i score_i(d) \tag{6.7}$$

Dans cette méthode, le score final est calculé en multipliant les scores individuels. Dans notre première expérimentation, nous utilisons seulement cette méthode pour combiner les scores (cf. la section 7.3).

Combinaison linéaire Dans cette méthode, le score final est la somme des scores individuels normalisés et pondérés par des coefficients.

$$score_final(d) = \sum_i \beta_i \times score_i(d) \tag{6.8}$$

Les scores individuels sont normalisés en divisant ces scores par la valeur maximale correspondante. Dans la formule 6.8, les β_i sont les coefficients positifs qui satisfont la condition $\sum_i \beta_i = 1$. Une approche pour apprendre ces coefficients serait d'utiliser les techniques d'apprentissage par machine, cependant dans nos travaux actuels nous essayons plusieurs combinaisons différentes pour trouver les meilleurs coefficients et l'apprentissage automatique est réservé pour les travaux futurs. Par exemple, pour combiner deux scores avec un pas de discrétisation de $0,05$, nous essayons 21 paires de coefficients correspondants : $\{0,0; 1,0\}$, $\{0,05; 0,95\}$, ... $\{0,95; 0,05\}$, $\{1,0; 0,0\}$. La meilleur paire de coefficients est la paire qui donne la meilleure valeur MAP (*Mean Average Precision*).

La combinaison basée sur la théorie Dempster-Shafer Comme nous avons abordé dans la section 2.4.3, la théorie Dempster-Shafer a été utilisée depuis longtemps pour combiner des sources d'information. Dans cette théorie, un domaine de problème est représenté par un cadre de discernement (*frame of discernment*) qui est un ensemble exclusif et exhaustif Θ des états (ou hypothèses/étiquettes/propositions). Dans notre problème, pour chaque requête utilisateur, un cadre de discernement correspondant est l'ensemble de n documents à re-trier $\{d_1, d_2..., d_n\}$. Différentes méthodes qui calculent les scores pour ces documents correspondent aux différentes fonctions de masse. Supposons que le score du document k calculé par la méthode i est $score_{k,i}$, le degré de confiance de cette méthode est $1 - \alpha_i$. Nous allons normaliser ces scores pour qu'ils puissent satisfaire la condition de la fonction de masse dans l'équation 2.4 :

$$\begin{aligned} m_i(d_k) &= (1-\alpha_i)\left(\frac{score_{k,i}}{\sum_{k=1}^{n} score_{k,i}}\right) \\ m_i(\Theta) &= \alpha_i \end{aligned}$$

Tous les autres $m(A)$ ($A \neq \Theta$ et A n'est pas un singleton d_k) sont nuls. Dans ce cas, la condition *bpa* est satisfaite : $\sum_k m_i(d_k) + m_i(\Theta) = 1$. Pour combiner deux scores $m_i(d_k)$ et $m_j(d_k)$ du document k correspondant à deux méthodes i et j afin d'obtenir un score combiné $m_{i,j}(d_k)$, nous utilisons la règle de combinaison de Dempster dans l'équation 2.9 :

$$\begin{aligned} m_{i,j}(d_k) &= m_i(d_k) \oplus m_j(d_k) \\ &= \frac{m_i(d_k)m_j(d_k) + m_i(d_k)m_j(\Theta) + m_j(d_k)m_i(\Theta)}{1 - \sum_{l \neq h} m_i(d_l)m_j(d_h)} \end{aligned}$$

Le nouveau $m_{i,j}(\Theta)$ satisfait la condition $\sum_k m_{i,j}(d_k) + m_{i,j}(\Theta) = 1$. S'il y a plus de deux scores à combiner, nous combinons ces scores séquentiellement de la manière suivante : $m_i \oplus m_j ... \oplus m_n$. Le résultat final n'est pas influencé par l'ordre de la combinaison.

Comme nous l'avons décrit dans la section 2.4, dans la théorie de Dempster-Shafer, il y a trois fonctions qui peuvent calculer les scores pour les documents : la fonction de masse *bpa*, la fonction de croyance *Bel*, et la fonction de plausibilité *Pl*. Il est facile à trouver que dans notre cas, la fonction de croyance donne les mêmes scores que la fonction de masse *bpa*. De plus, le score calculé par la fonction *Pl* est la somme des scores calculés par la fonction de masse et l'ignorance $m(\Theta)$ ($Pl(d_k) = m(d_k) + m(\Theta)$). Parce que $m(\Theta)$ est identique pour tous les documents, alors cette fonction donne le même classement que la fonction de masse. Finalement, ces trois fonctions donnent le même classement pour les documents. Alors, pour raison de simplicité, nous utilisons les scores calculés par la fonction de masse comme scores finaux pour classer les documents.

Comme dans le cas de la combinaison linéaire, un problème avec cette approche est de décider les degrés de confiance qui sont assignés pour chaque méthode. Bien que ces degrés puissent être obtenus lors d'un processus d'apprentissage (par exemple, [79]) ou par des heuristiques (par exemple, [149]), dans nos travaux nous essayons de nombreuses configurations de valeurs possibles et l'apprentissage automatique est réservé pour les travaux futurs. Contrairement au cas de la combinaison linéaire où les coefficients doivent satisfaire la condition $\sum_i \beta_i = 1$, dans ce cas il n'y a pas de contraintes explicites entre les degrés de confiance aux différentes sources d'évidences, ainsi le nombre des valeurs possibles est beaucoup plus grand. De ce fait, dans nos expérimentations nous élargissons les pas de discrétisation afin de réduire le nombre de combinaisons dans des limites acceptables en terme de temps de calcul.

6.6 Bilan

Dans ce chapitre, nous avons présenté notre approche pour personnaliser la recherche d'information dans une bibliothèque numérique scientifique. En particulier, nous nous concentrons sur les sujets suivants :

1. Utilisation des méthodes basées sur les citations comme la méthode du couplage bibliographique et la méthode des co-citations à côté des méthodes basées sur le contenu textuel (cosinus) pour calculer les similarités documents-profils afin de re-trier les documents. Comme nous avons présenté dans les premiers chapitres, il y a d'autres approches pour personnaliser les résultats de recherche (par exemple, en étendant la requête originale) mais elles ne sont pas notre but dans ce travail.
2. Utilisation des bases de données bibliographiques dans la méthodes des co-citations. Nous utilisons la base de données Web of Science de Thomson ISI qui est une base de données dominante pour les études bibliométriques. De plus, nous proposons également d'utiliser le Web comme une base de données bibliographique. La méthode des co-citations qui utilise le Web s'appelle la méthode des co-citations sur le Web.
3. Les fonctions de combinaisons pour combiner des scores calculés par différentes méthodes afin d'obtenir les scores finaux qui sont utilisés pour re-trier des documents : la combinaison produit, la combinaison linéaire, et la combinaison basée sur la théorie Dempster-Shafer. Il y a certainement d'autres méthodes de combinaison possibles, mais dans le cadre de nos travaux nous utilisons ces méthodes de combinaison à cause de deux raisons suivantes : i) elles sont des méthodes de combinaisons bien connues et elles ont été largement utiliser pour les tâches de combinaison et ii) dans le cadre de nos travaux, elles fonctionne assez bien et donnent des bons résultats.

Dans le chapitre suivant, nous allons décrire les expérimentations qui ont été conduites pour valider notre approche.

Chapitre 7

Expérimentations et résultats

Dans ce chapitre nous décrivons nos expérimentations et les résultats qui ont été publiés dans [142, 144, 145, 143]. Premièrement, nous abordons dans la section 7.1 les méthodes d'évaluation possibles pour évaluer les systèmes personnalisés, nous insistons sur la méthode de simulation qui est l'approche que nous avons retenue. Puis, dans la section 7.2 nous présentons la collection de test utilisée dans nos travaux. Par la suite, dans les sections 7.3 et 7.4 nous décrivons et discutons en détail les expérimentations et les résultats obtenus.

7.1 Méthodes d'évaluation

Les expérimentations jouent un rôle très important pour la recherche d'information. Cependant, il y a plusieurs façon d'évaluer un travail dans ce domaine. Pour évaluer les systèmes personnalisés, il y a trois approches possibles :

1. Conduite d'expérimentations avec de vrais utilisateurs.
2. Utilisation des jeux de données standards (*dataset*).
3. Simulation.

Chaque approche a des avantages et aussi des inconvénients. Nous présentons ces approches ainsi que l'approche que nous avons retenue dans la suite.

7.1.1 Conduite des expérimentations avec les vrais utilisateurs

La première approche est de conduire des expérimentations avec de vrais utilisateurs et de vraies données (par exemple, [23, 135, 121, 139]). L'avantage de cette approche est qu'elle est une bonne approximation des actions des utilisateurs dans la réalité. Cependant, il y a également plusieurs problèmes avec cette approche : ce n'est pas toujours facile de trouver des volontaires ou d'employer des participants dans les expérimentations. De plus, tous les paramètres doivent être fixés avant conduire les expérimentations. Si nous voulions changer quelques paramètres après les expérimentations, peut-être nous devrions re-faire toutes les expérimentations (c'est-à-dire il faudrait trouver à nouveau des volontaires ou employer de nouveaux participants).

La variabilité des expérimentations est aussi une limite parce que normalement nous pouvons avoir seulement un nombre limité des participants variable selon la capacité d'un

laboratoire ou d'une équipe de recherche. Un autre problème avec cette approche est qu'il est difficile de comparer les travaux de personnes/équipes différentes parce qu'elles n'ont pas les mêmes participants. Différents participants peuvent avoir différents comportements dans une même situation tandis que le nombre de participants n'est généralement pas suffisant pour que les résultats obtenus puissent converger vers une valeur représentative.

7.1.2 Construction des jeux de données standards

La deuxième approche possible est de construire des jeux de données (*dataset*) standards, comme les collections de test dans les campagnes d'évaluation TREC ou INEX. Cependant, bien qu'il y ait eu plusieurs recherches sur la personnalisation dans le passé, il manque toujours un cadre de travail standard pour évaluer les méthodes personnalisées. Peut-être la raison principale concerne le respect des informations privées des personnes qui participent dans la construction des jeux de données. Une autre raison est que les méthodes de personnalisation sont très différentes dans leur nature, ce qui rend difficile la construction de jeux de données communs.

7.1.3 Simulation

La troisième approche est de simuler les comportements des utilisateurs en construisant des « pseudo-profils ». Nous présentons par la suite quelques travaux de personnalisation qui utilisent cette approche.

Tâche de filtrage du TREC : Dans la tâche de filtrage (*filtering track*) de la campagne d'évaluation TREC 2002 [112], on utilise des documents pertinents dans les collections de test pour construire des profils utilisateurs. Cette tâche « simule les applications de filtrage de texte en-ligne à temps critique, où la valeur d'un document est réduite rapidement avec le temps ». C'est-à-dire le document qui est potentiellement intéressant doit être présenté immédiatement à l'utilisateur[1]. Cette tâche se compose de trois sous-tâches : filtrage adaptatif (*adaptive filtering*), filtrage par lot (*batch filtering*), et routage (*routing*). Dans la sous-tâche de filtrage adaptatif, le système commence à filtrer les documents avec un profil utilisateur original qui se compose de quelques exemples positifs (documents pertinents). Chaque fois qu'un document est retrouvé (*retrieved*), le jugement de pertinence de ce document est disponible immédiatement au système et le système utilise cette information pour mettre à jour le profil. Ce modèle de filtrage est basée sur l'hypothèse que les utilisateurs examinent périodiquement les documents pour donner des jugements de pertinence de manière interactive. Dans les sous-tâche de filtrage par lot et de routage, tous les exemples de documents (*training documents*) et leurs jugements de pertinence sont disponibles à l'avance et le système peut utiliser ces informations pour construire des profils utilisateurs. La différence entre ces sous-tâche est que, dans le filtrage par lot le système doit décider d'accepter ou de rejeter chaque document, mais dans la sous-tâche de routage le système doit retourner une liste triée des documents.

[1] *The TREC filtering track tries to simulate on-line time-critical text filtering applications, where the value of a document decays rapidly with time. This means that potentially relevant documents must be presented immediately to the user [112].*

Simulation d'un profil ontologie : Dans [125], les auteurs utilisent un modèle ontologie (cf. section 3.1.2.2) pour représenter les profils utilisateurs. L'ontologie de référence est celle du projet *ODP* (*Open Directory Project*). C'est une hiérarchie de sujets (concepts) avec des pages Web appartenant à ces concepts. Pour simuler une expérimentation de re-classement personnalisé de résultats de recherche, ils utilisent 563 concepts et 10 226 documents qui sont indexés sous ces concepts. Les documents sont divisés en trois ensembles séparés : un ensemble d'entraînement (*training set*), un ensemble de test (*test set*), et un ensemble de profilage (*profile set*). L'ensemble d'entraînement contient 5041 documents et il est utilisé pour construire des vecteurs représentant les concepts de l'ontologie. Chaque document dans cet ensemble est représenté par un vecteur de termes pondérés $\overrightarrow{d} = (w_1, w_2, ..., w_k)$. Supposons que $Docs(n)$ est l'ensemble des documents indexés sous le concept n et des documents indexés sous les sous-concepts de n. Alors le vecteur de termes pondérés qui représente le concept n est le vecteur centroïde de ces documents :

$$\overrightarrow{n} = \left(\sum_{d \in Docs(n)} \overrightarrow{d} \right) / |Docs(n)|$$

L'ensemble de profilage contient 2118 documents. Il est ensuite utilisé pour calculer les scores d'intérêt des concepts. Le score d'intérêt d'un concept c représente l'intérêt de l'utilisateur pour ce concept. Nous le noterons $IS(c)$. Les expérimentations considèrent un certain nombre de concepts, construisent une requête pour chaque concept considéré puis re-trient les résultats d'un moteur de recherche en prenant en compte les scores d'intérêt. Les scores d'intérêt sont re-calculés pour chaque requêtes. Pour cela, les scores d'intérêt sont initialisés à un et les documents dans l'ensemble de profilage qui sont associé à ce concept est utilisé pour mettre à jour les score d'intérêt.

Enfin, après avoir calculé les scores d'intérêt correspondant à la requête actuelle, les documents dans l'ensemble de test seront utilisés dans l'expérimentation. L'ensemble de test contient 3067 documents. Selon la requête q, un document qui est originalement indexé/non indexé sous le concept correspondant ou ses sous-concepts sera considéré comme pertinent/non pertinent. Un algorithme va re-trier la liste de résultats de cette requête. Pour un document d_i dans la liste, il faut calculer la similarité de ce document avec chaque concept dans l'ontologie pour trouver le concept c le plus similaire avec ce document. Puis, on calcule le score final pour re-trier ce document :

$$rankScore(d_i) = \begin{cases} IS(c) \times \alpha \times similarité(d_i, q) \times similarité(q, c) & si\ IS(c) > 1 \\ IS(c) \times similarité(d_i, q) \times similarité(q, c) & sinon \end{cases}$$

Dans cette formule, $IS(c)$ est le score d'intérêt du concept c, $similarité(d_i, q)$ est la similarité entre le document d_i et la requête q, $similarité(q, c)$ est la similarité entre la requête q et le concept c, et $similarité(q, c)$ est la similarité entre la requête q et le concept c.

Simulation par un domaine de requête : Dans [27], les utilisateurs simulent les expérimentations de re-classement des résultats de recherche en utilisant la collection de

test TREC (disques 1 et 2 liée à la tâche ad hoc). Comme les requêtes (topic) de cette collection sont décrites par un champ particulier qui spécifie leurs domaines respectifs (environnement, finance, économie internationale, ...), ils supposent qu'un domaine de requête correspond à un centre d'intérêt pour l'utilisateur. Chaque domaine est associé à un nombre de requêtes, alors ils simulent les centres d'intérêt de manière suivante :

1. Pour chaque domaine de requête de la collection, on sélectionne un ensemble de requêtes associées.

2. A partir de cet ensemble, un processus automatique récupère une liste des vecteurs associés aux 30 documents pertinents et non pertinents à chaque requête.

3. Un centre d'intérêt est construit à partir des vecteurs de ces documents. Un centre d'intérêt est représenté par un vecteur de termes pondérés c_k où le poids d'un terme t_i dans le centre c_k est calculé en appliquant la formule suivante :

$$wtc(i,k) = \log \frac{(r+0.5)/(R-r+0.5)}{(n-r+0.5)/(N-n-R-r+0.5)}$$

Dans cette formule, R est l'ensemble des documents pertinents à la requête et appartenant au centre c_k, r est le nombre de documents pertinents contenant le terme t_i, n est le nombre de documents contenant le terme t_i, N est le nombre total de documents de la collection.

L'ensemble simulé des centres d'intérêts sera utilisé pour re-trier des résultats de recherche.

Utilisation des informations de *logs* : Dans [33], les auteurs utilisent les *logs* du moteur de recherche MSN pour simuler et évaluer les méthodes personnalisées de re-classement. Dans ces *logs*, chaque utilisateur est identifié par *cookie*. Pour chaque requête, les pages Web cliquées et leurs rangs sont enregistrés dans les *logs*. Les auteurs extraient 10 000 utilisateurs et leurs informations qui sont enregistrées pendant 12 jours pour construire les jeux de données. Les *logs* des 11 premiers jours sont utilisés comme les exemples pour construire des profils et les *logs* du dernier jours sont utilisé pour évaluer les stratégies de re-classement. Les étapes d'évaluation sont :

1. Pour chaque requête utilisée pour évaluer, on télécharge les 50 premiers résultats à partir du moteur de recherche MSN. L'ensemble de ces résultats est noté U et la liste de rangs de ces résultats est notée τ_1.

2. Pour chaque page Web dans U, on calcule un score personnalisé en utilisant l'algorithme de personnalisation et on génère une nouvelle liste de rangs τ_2. Le score personnalisé est la similarité entre la page Web et le profil. Les auteurs considèrent deux type de profils : un profil à long terme et un profil à court terme. En utilisant le profil à long terme, le score personnalisé de l'utilisateur u pour une page p avec

la requête q est calculé de manière suivante :

$$S^{L-Profile}(q,p,u) = \frac{c_l(u) \cdot c(p)}{\|c_l(u)\|\|c(p)\|}$$

Dans cette formule, $c(p)$ est le vecteur de la page p, $c_l(u)$ est le vecteur profil, calculé par :

$$c_l(u) = \sum_{p \in P(u)} P(p|u)w(p)c(p)$$

$P(u)$ est la collection des pages Web visités par l'utilisateur u dans le passé (dans les 11 premiers jours) ; $P(p|u)$ est la probabilité que la page p soit cliquée par l'utilisateur u ; $w(p)$ est l'impact de pondération pour la page p quand on génère les profils utilisateurs. $P(p|u)$ et $w(p)$ sont estimés en utilisant les données historiques des 11 premiers jours.
Par contre, un profil à court terme est calculé de la manière suivante :

$$c_s(u) = \frac{1}{|P_s(q)|} \sum_{p \in P_s(q)} c(p)$$

$P_s(q)$ est la collection des pages visitées dans la session actuelle de l'utilisateur (pour les autres requêtes). En utilisant le profil à court terme, le score personnalisé d'une page p est calculé par :

$$S^{S-Profile}(q,p,u) = \frac{c_s(u) \cdot c(p)}{\|c_s(u)\|\|c(p)\|}$$

Les auteurs proposent également de fusionner $S^{L-Profile}(q,p,u)$ avec $S^{S-Profile}(q,p,u)$ pour avoir un score combiné :

$$S^{LS-Profile}(q,p,u) = \theta S^{L-Profile}(q,p,u) + (1-\theta)S^{S-Profile}(q,p,u)$$

Dans cette formule $S^{LS-Profile}(q,p,u)$ est le score personnalisé combiné.

3. Combiner les rangs dans τ_1 et τ_2 et re-trier les pages Web en utilisant les rangs combinés. Dans ces travaux, on combine des rangs au lieu des scores des pages Web. La liste finale de rangs est notée τ.

4. Évaluer les performances de différentes méthodes de re-classement en utilisant deux métriques : *Rank Scoring* et *Average Rank*.

7.1.4 Approche retenue

Dans nos travaux, nous utilisons une approche de simulation pour les expérimentations. Parce que nous n'avons pas une bibliothèque réelle de taille suffisante pour conduire

des expérimentations, nous utilisons une collection de test qui contient des articles scientifiques. C'est la collection de test utilisée dans la campagne d'évaluation INEX 2005 (cf. section 2.3.5.2). Dans cette collection de test, il y a un corpus de documents, des topics représentant des besoins d'information, et des jugements de pertinence.

Nous supposons qu'un topic représente un besoin d'information/domaine d'intérêt d'une personne. Les différents topics peuvent représenter différents besoins d'information/domaines d'intérêt de différentes personnes ou d'une même personne. Comme nous avons mentionné dans le chapitre 6, nos travaux concernent le re-classement des résultats de recherche en utilisant des profils utilisateurs. Pour simuler le profil de cette personne, nous utilisons une part des documents jugés pertinents pour construire un « pseudo-profil » de cette personne ; les autres documents pertinents seront utilisés dans les expérimentations de re-classement. Nous pensons que cette approche est raisonnable à cause de raison suivante. Normalement, les informations dans un profil utilisateur reflètent l'intérêt de l'utilisateur. Elles correspondent donc avec les besoins d'information de cet utilisateur. Si on considère qu'un topic représente un domaine d'intérêt d'une personne, alors les documents jugés comme pertinents pour un topic satisfont une condition importante pour pouvoir être considérés comme le profil de cette personne.

Nous utilisons deux stratégies pour sélectionner des documents pertinents afin de construire des profils utilisateurs. Ces deux stratégies correspondent à deux scénarios d'un vrai utilisateur dans la vie réelle :

1. La première stratégie et de sélectionner manuellement les documents pertinents pour construire les profils. Cette stratégie correspond à un scénario dans lequel l'utilisateur va préciser explicitement les documents qui représentent ses intérêts.

2. La deuxième stratégie et de sélectionner de manière « aléatoire » un ensemble de documents à partir des documents pertinents pour construire les profils. Cette stratégie correspond à un scénario dans lequel l'utilisateur ne précise pas explicitement les documents qui représentent ses intérêts mais le système doit le faire en utilisant les techniques de construction de profils utilisateurs (cf. chapitre 3).

Nous avons conduit deux expérimentations qui correspondent à ces deux scénarios. Les expérimentations seront présentées dans les sections suivantes. Tout d'abord, nous présentons la collection de test INEX que nous utilisons dans nos expérimentations dans la section 7.2.

7.2 Collection de test INEX

Nous avons examiné les différentes collections de test disponibles pour choisir une collection pour nos expérimentations. Pour qu'une collection soit utilisable dans nos expérimentations, elle doit satisfaire les trois conditions suivantes :

- C'est une collection qui contient des articles scientifiques, car nous nous concentrons sur les bibliothèques numériques d'articles scientifiques.
- La taille de cette collection doit être assez importante pour qu'elle puisse représenter une bibliothèque numérique.

– Il faut pouvoir obtenir des informations bibliographiques comme des citations, des références de cette collection, qui sont nécessaires pour les méthodes basées sur les citations comme la méthode du couplage bibliographique ou la méthode des co-citations.

En fait, il y a assez peu de collections de test qui satisfont ces conditions. Les collections de test traditionnelles comme CACM ou CISI contiennent des articles scientifiques mais elles sont petites et obsolètes. Réciproquement, dans les collections récentes peu contiennent des articles scientifiques car les campagnes d'évaluation se concentrent sur d'autres types de collections plutôt que sur les collections d'articles scientifiques. Après avoir considéré plusieurs collections de test différentes, nous choisissons la collection utilisée dans la campagne d'évaluation INEX 2005[2]. Cette collection satisfait toutes les conditions nécessaires pour nos expérimentations. Par la suite, nous allons présenter les caractéristiques de cette collection.

7.2.1 Corpus de documents

Le corpus de la collection INEX 2005 contient environ 17000 documents formatés en XML extraits de 24 revues de *IEEE Computer Society* dans la période de 1995-2004. Cette collection est une expansion des corpus utilisés dans les années 2002-2004 dans les campagnes d'évaluation d'INEX.

La taille de cette collection est 735 Mo (568 Mo sans les balises XML). La structure d'un article typique se compose d'un *front matter* (*<fm>*), un *body* (*<bdy>*), et un *back matter* (*<bm>*). Le *front matter* contient les méta-données d'un article, comme titre, auteurs, abstract etc. Le *body* (*<bdy>*) d'un article contient le contenu de cet article. Cette partie contient des sections (*<sec>*), sous-sections (*<ss1>*), paragraphes (*<p>*) etc. La partie *back matter* (*<bm>*) contient les références (*<bb>*) et les informations supplémentaires concernant les auteurs. Les références contiennent les informations comme les noms des auteurs, le titre de l'article, la revue/conférence, l'année de publication etc. La figure 7.1 illustre la structure d'un article typique de la collection.

Les documents de la collection INEX ne sont pas seulement des articles scientifiques mais aussi d'autres documents comme des critiques de livres, des éditoriaux, le courrier des lecteurs, etc. Donc dans la première étape nous essayons d'éliminer ces autres documents pour ne garder que les articles. Nous avons constaté que ces documents soit ne contiennent pas de champ *title*, soit les titres de ces documents sont des phrases simples comme *News*, *About this Issue*, *Article summaries* etc. Après cette étape, la collection contient 14237 documents (par rapport à 17000 documents dans la collection originale). Cette collection peut être utilisée comme une bibliothèque numérique de taille moyenne en informatique. Puis nous extrayons toutes les informations nécessaires pour les expérimentations comme le titre, le nom de la revue, l'année de parution, les références etc. De plus, la collection contient des documents formatés en XML mais dans nos expérimentations nous ne nous intéressons pas à l'aspect de la structure de ces documents. Alors nous enlevons donc toutes les balises XML de ces documents et ne gardons que le contenu textuel de ces documents.

[2]http://inex.is.informatik.uni-duisburg.de/2005/index.html

```
<article>
    <fm>
        ...
        <ti>IEEE Transactions on ...</ti>
        <atl>Construction of ...</atl>
        <au>
            <fnm>John</fnm>
            <snm>Smith</snm>
            <aff>University of ...</aff>
        </au>
        <au>...</au>
        ...
    </fm>
    <bdy>
    <sec>
        <st>Introduction</st>
        <p>...</p>
        ...
    </sec>
    <sec>
        <st>...</st>
        ...
        <ss1>...</ss1>
        <ss1>...</ss1>
        ...
    </sec>
    ...
    </bdy>
    <bm>
        <bib>
            <bb>
                <au>...</au><ti>...</ti>
                ...
            </bb>
            ...
        </bib>
    </bm>
</article>
```

FIG. 7.1 – La structure d'un article typique de l'INEX [52].

```
<!– Topic definition –>
<inex_topic topic_id="202" query_type="CO+S" ct_no="1">
<InitialTopicStatement>I'm interested in knowing how ontologies are used to
encode knowledge in real world scenarios. I'm writing a report on the use of
ontologies. I'm particularly interested in knowing what sort or concepts and
relations people use in their ontologies. </InitialTopicStatement>
<title>ontologies case study</title>
<castitle>//article[about(., ontologies)]//sec[about(., ontologies case study)]</castitle>
<description>Case studies in the use of ontologies</description>
<narrative>I'm writing a report on the use of ontologies. I'm interested in
knowing how ontologies are used to encode knowledge in real world scenarios.
I'm particularly interested in knowing what sort or concepts and relations
people use in their ontologies. I'm not interested in general ontology frameworks
or technical details about tools for ontology creation or management. An example
relevant result contains a description of the real world phenomena described by
the ontology and also lists some of the concepts used and relations between concepts.
</narrative>
</inex_topic>
```

FIG. 7.2 – Exemple d'un topic CO+S.

7.2.2 Les topics dans la collection INEX

INEX fournit également des besoins d'informations (topics) avec la collection et aussi des jugements pour chaque topic. Il existe deux types de topics dans cette collection [76] :

- Les topics CAS (*Content-And-Structure*) qui contiennent explicitement des informations concernant la structure des réponses souhaitées. La figure 7.3 illustre un exemple d'un topic CAS de INEX.
- Les topics CO (*Content-Only*) qui ne précisent pas la structure des documents cherchés. Dans INEX 2005, les topics CO+S (*Content-Only + Structure*) ont été introduits. Ce sont des variantes de ces topics. Ces topics contiennent également un champ facultatif *<castitle>* qui inclut les contraintes structurelles comme dans les topics CAS. Les topics CA+S permettent une comparaison d'un système de recherche d'information XML selon deux scénarios sur un même topic quand les contraintes structurelles sont prises en compte (+S) et quand elles sont ignorées (CO). Dans nos expérimentations, nous n'utilisons que les topics CO+S. La figure 7.2 illustre un exemple d'un topic CO+S de INEX.

7.2.2.1 Format des topics

Dans un topic CO+S, il y a plusieurs parties, chaque partie représente le même besoin d'information [127].

- **<narrative> :** description détaillée du besoin d'information qui explique les raisons pour que un élément soit jugé pertinent ou non pertinent.
- **<description> :** description courte du besoin d'information.

```
<!– Topic definition –>
<inex_topic topic_id="253" query_type="CAS" ct_no="39">
<InitialTopicStatement>We have developed an environment for the evaluation of
retrieval systems that allows both usability and retrieval experiments to be performed.
We are now looking for alternative evaluation measures and methodologies that are
used for usability testing in digital libraries.</InitialTopicStatement>
<title></title>
<castitle>//article[about(.//abs,evaluation "usability experiment" "digital libraries")]
//sec[about(., evaluation methodology measures "usability testing")]</castitle>
<description>Retrieve information about evaluation methodology or evaluation
measures used for usability testing (experiments) in digital libraries.</description>
<narrative>We have developed an environment for the evaluation of retrieval systems
that allows both usability and retrieval experiments to be performed. We are now
looking for alternative evaluation measures and methodologies that are used for
usability testing in digital libraries. We expect to find relevant information in articles
with an abstract discussing the evaluation of digital libraries through a usability
experiment. We want to retrieve sections that explain the used methodology or evaluation
measures for usability testing.</narrative>
</inex_topic>
```

FIG. 7.3 – Exemple d'un topic CA+S.

- **<title> :** Les mots clés du besoin d'information. Le champ *<title>* est présent seulement dans les topics CO+S.
- **<castitle> :** combinaison des mots clés et des contraintes structurelles du besoin d'information. Ce champ est obligatoire dans les topics CAS mais il est facultatif dans les topics CO+S.

7.2.2.2 Préparation des requêtes

Nous construisons manuellement les requêtes à partir des topics. Nous voulons simuler les scénarios de vrais utilisateurs dans la vie réelle qui ont des difficultés à formuler des requêtes précises pour exprimer leurs besoins d'information. Nous construisons donc des requêtes courtes qui contiennent seulement deux ou trois mots clés afin de valider l'efficacité des méthodes de re-classement de résultats de recherche en utilisant les profils utilisateurs. Rappelons que dans la recherche d'information sur le Web, une requête typique contient le plus souvent entre un et trois mots [134].

7.2.3 Jugement de pertinence

Comme nous l'avons présenté brièvement dans la section 2.3.2.1, dans la campagne d'évaluation INEX, les éléments retournés par les SRI ne sont pas seulement des documents entiers mais aussi des composants XML. De ce fait les jugements de pertinence ont été faits au niveau des éléments (y compris le niveau de document). INEX utilise les deux mesures suivantes pour calculer la pertinence d'un élément :

- *exhaustivité (e)* : cette mesure décrit à quel niveau cet élément traite le topic de la requête. Dans INEX 2005, il y a 3+1 niveaux avec cette mesure : très exhaustif (*highly exhaustive*, *e = 2)*, signifie que cet élément traite la plupart ou tous les aspects du topic ; assez exhaustif (*somewhat exhaustive*, *e = 1*), signifie que cet élément traite quelques aspects du topic ; trop petit (*too small*, *e = ?*), signifie que cet élément contient quelques information pertinente mais il est trop petit ; dans tous les autres cas, l'élément est jugé comme non exhaustif (*not exhaustive*, *e = 0*). Généralement, l'exhaustivité d'un élément père est toujours égal ou supérieur à celui de ses fils. Par exemple, si un élément fils est jugé comme « très exhaustif », alors l'élément père de cet élément peut être considéré comme « très exhaustif » parce que son élément fils traite déjà la plupart ou tous aspects du topic. Les personnes qui participent dans la construction des collections de test de l'INEX doivent juger explicitement le niveau d'exhaustivité de tous les éléments qui contiennent des informations pertinentes [127].

- *spécificité (s)* : cette mesure décrit à quel niveau cet élément se concentre sur le topic de la requête. Dans INEX 2005, cette mesure est représentée par une valeur réelle dans l'intervalle [0,1]. La valeur $s = 1$ correspond à un élément qui ne contient que des informations pertinentes. Pour juger un document, un participant doit lire attentivement tout le document et surligner (*highlight*) les parties pertinentes. La spécificité d'un élément est définie comme un ratio entre la taille de texte surlignée (*rsize*) et la taille totale de l'élément (*size*).

INEX propose plusieurs fonctions pour combiner ces deux mesures en une valeur de pertinence unique (cf. la section 2.3.2.1) afin de pouvoir utiliser une évaluation multivaluée pour les documents. Cependant, dans nos expérimentations nous utilisons une évaluation binaire traditionnelle « pertinent/non pertinent » pour les documents avec l'outil *trec_ eval*. Une évaluation multivaluée pourrait être intéressante à étudier mais elle est réservée pour les travaux futurs. Dans nos expérimentations, nous définissons un document pertinent comme un document avec une exhaustivité $e \neq 0$ et une spécificité $s \neq 0$, quelque soit les niveaux de e et de s. Rappelons que dans la campagne d'évaluation TREC[3], un document est jugé comme pertinent si une part de ce document est pertinent, quelque soit sa taille par rapport au reste du document[4]. La figure 7.4 est un extrait d'un fichier de jugement de pertinence de INEX. Nous appliquons une transformation sur les fichiers de jugement de pertinence pour que ces jugements puissent être utilisés avec l'outil *trec_ eval*.

[3] http://trec.nist.gov/data/reljudge_eng.html

[4] *A document is judged relevant if any piece of it is relevant (regardless of how small the piece is in relation to the rest of the document).*

```
<file collection="ieee" name="ex/1996/x6030">
</file>
<file collection="ieee" name="co/1996/r7057">
...
<element path="/article[1]/bdy[1]" exhaustivity="2" size="19563" rsize="1890"/>
<element path="/article[1]" exhaustivity="2" size="25307" rsize="1890"/>
<element path="/article[1]/bdy[1]/sec[3]/st[1]" exhaustivity="?" size="22" rsize="22"/>
<element path="/article[1]/bdy[1]/sec[3]/p[1]" exhaustivity="1" size="540" rsize="540"/>
...
</file>
<file collection="ieee" name="so/1995/s1004">
</file>
<file collection="ieee" name="co/1997/r2033">
</file>
```

FIG. 7.4 – Jugement de pertinence de l'INEX.

7.3 Première expérimentation

7.3.1 Procédure d'évaluation

Dans notre première expérimentation, nous validons la première stratégie de simulation que nous avons mentionnée dans la section 7.1.4. Cette stratégie correspond à un scénario dans lequel l'utilisateur va préciser explicitement les documents qui représentent ses intérêts. Comme nous l'avons mentionné, nos expérimentations sont des simulations de recherches personnalisées en utilisant des profils utilisateurs. Les différents topics peuvent représenter différents besoins d'information/domaine d'intérêt de différentes personnes ou d'une même personne. Pour chaque topic, nous sélectionnons quelques documents pertinents (5 en moyenne) pour former un « pseudo-profil » de l'utilisateur à la base de ce topic. Les documents sélectionnés sont les documents pertinents « importants » qui reçoivent plusieurs citations. Nous pensons que cette approche est raisonnable parce que dans la réalité si l'utilisateur d'une bibliothèque numérique doit déclarer son profil, il choisira probablement des articles importants dans son domaine de recherche ; et si on construit le profil à partir des documents que l'utilisateur a lus, le choix doit porter sur des articles importants. Les articles qui sont inclus dans les profils seront exclus des résultat pour éviter un biais dans l'évaluation.

Comme nous l'avons présenté dans la section 6.1, après l'étape de préparation, nous utilisons le moteur de recherche Zettair pour indexer la collection INEX. Le modèle par défaut utilisé dans Zettair est le modèle *Dirichlet-smoothed* [104]. Il y a 29 topics CO+S originaux. Cependant, certains topics contiennent peu de documents pertinents et comme nous devons utiliser un certain nombre de ces documents pour former un « pseudo-profil » d'un topic, nous ne considérons que les topics ayant plus de 30 documents pertinents, soit 20 topics. Ce sont les topics : 206 207 208 209 210 212 213 216 217 218 221 222 223 227 228 229 234 235 236 237. Nous envoyons les 20 requêtes construites à partir de ces topics à Zettair. Avec chaque requête nous re-trions 300 premiers documents en utilisant

les « pseudo profils » correspondants.

Le score final d'un document est la combinaison du score original calculé par Zettair et la similarité document-profil (cf. la formule 6.2). Dans cette première expérimentation, on combine seulement deux scores : i) soit le score de Zettair avec la similarité document-profil calculé par la méthode du couplage bibliographique ; ii) soit le score de Zettair avec la similarité document-profil calculé par la méthode des co-citations utilisant la base de données ISI Web of Science iii) soit le score de Zettair avec la similarité document-profil calculé par la méthode des co-citations sur le Web. De plus, dans cette première expérimentation, nous utilisons seulement la combinaison produit (les autres fonctions de combinaison sont utilisées dans la deuxième expérimentation).

7.3.2 Résultats

Les mesures de performance utilisées dans cette expérimentations sont la mesure précision/rappel et les précisions à n avec $n = 5, 10, 15, 20, 30$ (cf. la section 2.3.4). Les résultats des expérimentations sont présentés dans figure 7.5 (précision/rappel) et tableau 7.1 (précisions à 5, 10, 15, 20, 30 premiers documents). Nous nous intéressons particulièrement aux premiers niveaux de précisions parce que c'est une mesure très importante liée au fait que les utilisateurs ne consulte le plus souvent qu'environ les 30 premiers documents retournés par les moteurs de recherche (c'est-à-dire, la première ou la deuxième page de résultats). Un document pertinent mais classé au 100ème rang ne sera quasiment jamais regardé. Dans la figure 7.5, *Zettair* correspond au résultat original de Zettair, *Cocitations_ WoS* correspond au résultat de la méthode de re-classement qui combine le score original de Zettair et la similarité document-profil calculée par la méthode des co-citations avec la base de données ISI Web of Science, *Cocitations_ Google* correspond au résultat de la méthode de re-classement qui combine le score original de Zettair et la similarité document-profil calculée par la méthode des co-citations sur le Web, *Couplage_ Bibliographique* correspond au résultat de la méthode de re-classement qui combine le score original de Zettair et la similarité document-profil calculée par la méthode du couplage bibliographique.

Comme nous l'avons mentionné dans la section 2.3.4, la courbe précision/rappel donne la précision du système aux différents niveaux de rappel pour chaque requête et la moyenne pour l'ensemble des requête. La précision à n documents est le nombre de documents pertinents parmi les n premiers documents retournés par le systèmes de RI.

7.3.3 Discussion sur le résultat

Précision à n : A partir des résultats, nous pouvons constater que la méthode des co-citations avec WoS ne donne aucune amélioration ; elle cause une dégradation de performance par rapport au résultat original de Zettair. La méthode du couplage bibliographique est un peu meilleure, mais l'amélioration n'est pas très claire. Pour la précision à 5 documents, cette méthode donne la meilleure performance (+10,61% d'amélioration par rapport au résultat original de Zettair). Cependant, pour les autres niveaux de précisions, elle ne donne pas beaucoup d'amélioration. La méthode des co-citations sur le Web est la meilleure, elle donne des améliorations à tous les niveaux de précision. Pour la précision à 30 document, cette méthode donne 15,06% d'amélioration de performance.

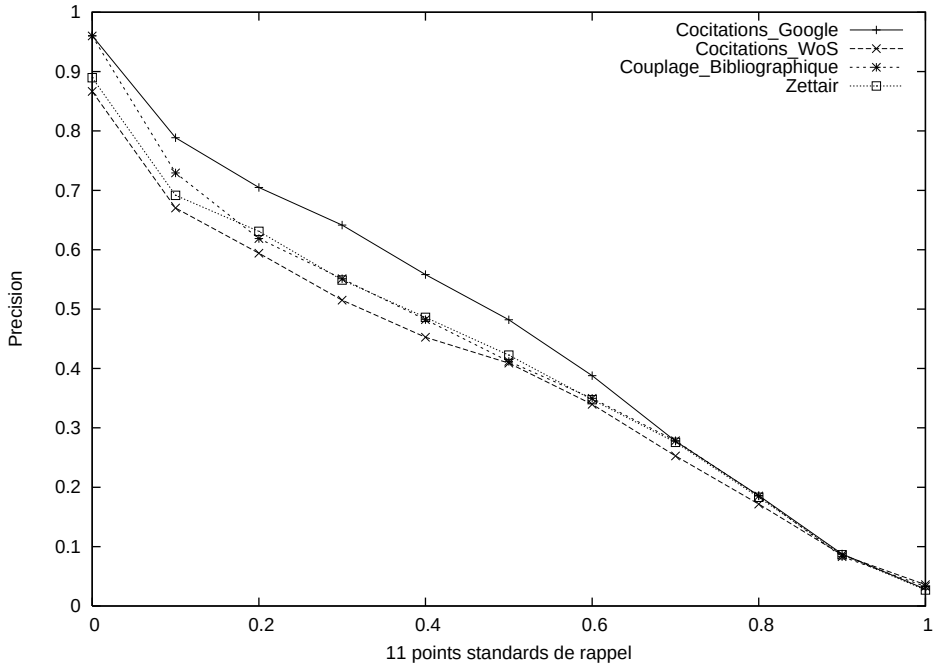

FIG. 7.5 – Re-classement des résultats de recherche de Zettair avec différentes méthodes de citations

Méthode	5 docs	10 docs	15 docs	20 docs	30 docs
Résultat original de Zettair	0,6600	0,6150	0,5633	0,5375	0,4867
Couplage bibliographique	0,7300 +10,61%	0,6050 -1,65%	0,5767 +2,38%	0,5600 +4,19%	0,4883 +0,33%
Co-citations avec WoS	0,6300 -4,55%	0,5900 -4,07%	0,5533 -1,78%	0,5150 -4,19%	0,4567 -6,16%
Co-citations avec Google	0,7100 +7,58%	0,6800 +10,57%	0,6400 +13,62%	0,6025 +12,09%	0,5600 +15,06%

TAB. 7.1 – Précisions à 5, 10, 15, 20, 30 documents

Courbe précision/rappel : Conformément à la mesure de précisions, la courbe précision/rappel de la méthode des co-citations avec WoS est moins bonne que celle du résultat original de Zettair. La méthode du couplage bibliographique donne quelques améliorations pour les premiers niveaux de rappel mais se dégrade dans les autres niveaux. Par contre, la méthode des co-citations sur le Web donne une amélioration nette par rapport aux autres méthodes. On trouve également que la différence entre les méthodes sont plus nettes pour les niveaux de rappel plus faibles.

Maintenant nous allons analyser les données d'expérimentations pour expliquer ces résultats. Pour calculer la similarité entre des documents et des « profils » pour le re-classement, nous devons calculer le nombre de co-citations (ou co-références) de 25497 paires de documents (chaque paire se compose d'un document à re-classer et d'un document dans un « profil utilisateur »). Il y a deux facteurs importants que nous devons considérer avec chaque méthode : i) Le nombre de paires qui sont co-citées (dans la méthode des co-citations) ou partagent des co-références (dans la méthode du couplage bibliographique) et ii) le nombre moyen des co-citations ou des co-références de chaque paire.

	Méthode du couplage bibliographique	Méthode des co-citations avec WoS	Méthode des co-citations sur le Web
Nombre des paires de documents ayant co-citations ou co-références	1126	213	4845
Nombre moyen de co-citations ou co-références de chaque paire	1,69	1,94	4,84

TAB. 7.2 – Analyse des données expérimentales

Comme nous pouvons le voir dans le tableau 7.2, dans la méthode des co-citations avec WoS, seulement 213 paires de documents sont co-citées et le nombre moyen des co-citations de chaque paire est de 1,94. Ce très petit nombre de paires co-cités est la raison pour laquelle elle ne peut pas donner des améliorations mais devenir une source de bruit qui cause des mauvais effets sur le résultat final. Plusieurs facteurs peuvent influencer sur la performance de la méthode des co-citations. Le plus important est la couverture de la base de données de citations que nous utilisons. Nous savons que le WoS fournit des informations de citations surtout pour des revues, mais en informatique les conférences jouent un rôle important, plus que dans d'autres domaines. Dans une estimation de CiteSeer[5] (voir la figure 7.6), parmi les 10 événements les plus connus dans l'informatique (estimé par le nombre moyen de citations reçues), seulement 2 événements sont des revues (8ème et 10ème places) et les autres 8 événements sont des conférences et symposiums. Le manque d'informations concernant les publications dans les conférences informatiques peut fortement influencer la couverture de WoS dans ce domaine. (Cependant, à la fin d'année 2008, la base de données *ISI Proceedings* qui donne l'accès sur le Web aux comptes-rendus des conférences, séminaires, ateliers ... est intégrée au WoS.) Les articles qui sont sélectionnés comme « profils » sont aussi déterminants : plus importants

[5] http://citeseer.ist.psu.edu/impact.html

1. OSDI : 3.31 (top 0.08%)
2. USENIX Symposium on Internet Technologies and Systems : 3.23 (top 0.16%)
3. PLDI : 2.89 (top 0.24%)
4. SIGCOMM : 2.79 (top 0.32%)
5. MOBICOM : 2.76 (top 0.40%)
6. ASPLOS : 2.70 (top 0.49%)
7. USENIX Annual Technical Conference : 2.64 (top 0.57%)
8. TOCS : 2.56 (top 0.65%)
9. SIGGRAPH : 2.53 (top 0.73%)
10. JAIR : 2.45 (top 0.81%)
11. SOSP : 2.41 (top 0.90%)
12. MICRO : 2.31 (top 0.98%)
13. POPL : 2.26 (top 1.06%)
14. PPOPP : 2.22 (top 1.14%)
15. ...

FIG. 7.6 – Liste des événements importants de CiteSeer. Parmi 10 premiers événements, seulement deux sont des revues.

ils sont, plus de citations ils peuvent recevoir, et la probabilité qu'ils soient co-cités avec les autres articles sera plus élevée. Même si nous avons essayé de sélectionner des articles importants dans la collection, il n'y a aucune garantie qu'ils soient les plus importants dans leur domaine.

Dans la méthode du couplage bibliographique, il y a 1126 paires de documents ayant des co-références et le nombre moyen des co-références de chaque paire est de 1,69. Ce nombre plus élevé de documents concernés explique la petite amélioration de cette méthode. Les références des articles sont extraits à partir du contenu des articles ; ils ne sont donc pas dépendants de la base de données de citations utilisée.

Dans la méthode des co-citations sur le Web avec Google, il y a 4845 paires de documents qui sont co-citées. Le nombre moyen des co-citations de chaque paire est de 4,84. C'est bien meilleur que les deux premiers cas. C'est pourquoi elle obtient la meilleure performance. Bien que la nature des co-citations sur le Web soit très variée comme nous l'avons expliqué dans le chapitre précédent, la grande couverture du Web peut compenser cet inconvénient.

7.4 Deuxième expérimentation

Dans notre première expérimentation, nous validons la deuxième stratégie de simulation que nous avons mentionnée dans la section 7.1.4. Cette stratégie correspond à un scénario dans lequel l'utilisateur ne précise pas explicitement les documents qui représentent ses intérêts mais le système doit le faire en utilisant les techniques de construction de profils utilisateurs (cf. chapitre 3). Dans la première expérimentation, pour chaque topic nous avons sélectionné manuellement quelques documents pertinents pour former un « pseudo-profil ». Dans cette nouvelle expérimentation, nous choisissons aléatoirement les documents pertinents pour former le profil. Le principe de validation de cette deuxième

expérimentation est basée sur la méthode de validation croisée à **k** blocs. Contrairement à la première expérimentation qui a été conduite une seule fois, la deuxième expérimentation est répétée plusieurs fois pour donner un résultat plus certain.

Par la suite, nous présentons brièvement les méthodes de validations et la méthode de validation croisée à **k** blocs qui a été appliquée dans nos travaux.

7.4.1 Les méthodes de validation

Les méthodes de validation [68] ont pour but d'estimer les performances de différents modèles. La méthode la plus simple est la méthode *holdout* (voir la figure 7.7). Ces méthodes partitionnent l'ensemble de données en deux sous-ensembles. Le premier constitue les données d'entraînement (*training data*) et l'autre constitue des données de test (*testing data*). Dans nos premières expérimentations, nous avons utilisé une approche similaire : parmi les documents pertinents d'un topic, on sélectionne quelques documents pertinents pour construire le profil utilisateur (cet ensemble joue le rôle des données d'entraînement) et les autres sont utilisés dans l'expérimentation de re-classement (cet ensemble joue le rôle des données de test).

L'avantage de cette méthode est qu'elle est simple. Cependant, ce n'est pas toujours facile de décider les portions de données utilisées dans chaque ensemble. De plus, parce que l'expérimentation est faite une seule fois, les performances des modèles peuvent être inexactes (trop faibles ou trop élevées par rapport aux vraies performances des systèmes).

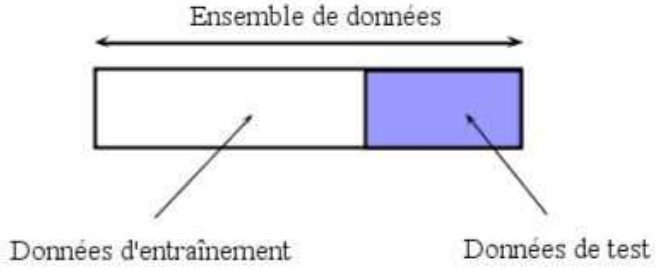

FIG. 7.7 – Méthode *Holdout*.

Les méthodes de validation croisée sont assez similaires à la méthode *holdout*. Cependant, dans ces méthodes, les expérimentations sont répétées plusieurs fois avec différents ensembles de données d'entraînement et différents ensembles de données de test. Nous abordons par la suite les méthodes de validation croisée connues :

- **Méthode de validation croisée à k blocs**[6] **:** Dans cette méthode, l'ensemble de données est divisé en **k** sous-ensembles (blocs). L'expérimentation est répétée **k** fois, à chaque fois un bloc est utilisé comme données de test et les autres $\mathbf{k}-1$ blocs sont utilisés comme données d'entraînement pour le modèle. Les $\mathbf{k}-1$ résultats obtenus lors de $\mathbf{k}-1$ expérimentations peuvent être moyennés ou combinés pour produire un résultat final. Dans ce cas, tous les éléments dans l'ensemble de données seront utilisés dans le test une fois et dans l'ensemble d'entraînement $\mathbf{k}-1$ fois. L'inconvénient de cette approche par rapport à la méthode *holdout* est qu'il faut répéter

[6] *k-fold cross validation* en anglais.

l'expérimentation **k** fois, ce qui est beaucoup plus coûteux que la méthode *holdout*. Cette méthode est illustrée dans la figure 7.8.

- **Méthode Leave-one-out :** Cette méthode est une variante de la méthode de validation croisée à **k** blocs. Dans cette méthode, $\mathbf{k} = \mathbf{N}$ où **N** est le nombre d'éléments dans l'ensemble de données. C'est-à-dire les données d'entraînement se composent de tout l'ensemble de données, sauf un seul élément qui est utilisé dans le test.

7.4.2 Procédure d'évaluation

Nous utilisons une autre méthode d'évaluation qui est basée sur le principe de la méthode de validation croisée à **k** blocs. Les étapes suivantes sont appliquées :

- Pour chaque topic, nous partitionnons aléatoirement l'ensemble des documents pertinents en **k** blocs (dans nos expérimentations, $\mathbf{k} = 5$).

- Les documents dans un bloc sont utilisés comme documents de test et les documents dans les **k** - 1 autres blocs sont utilisés comme le « pseudo-profil » de ce topic (un topic représente un domaine d'intérêt d'une personne).

- L'expérimentation est répétée **k** fois, chaque fois avec un bloc différent contenant les documents de test.

- Le résultat final sera la valeur moyenne des **k** résultats des **k** expérimentations. Dans ces expérimentations, nous moyennons les valeurs de MAP et de précisions à n des expérimentations.

Avec cette approche, chaque document pertinent sera utilisé comme document de test 1 fois et dans le « profil » **k** - 1 fois. Parmi 29 topics CO+S originaux, on utilise 26 topics pour former 26 requêtes. Les topics ayant trop peu de documents pertinents sont ignorés. Les topics suivants sont utilisés dans la deuxième expérimentations : 202 203 205 206 207 208 209 210 212 213 216 217 218 221 222 223 227 228 229 230 232 235 236 237 239 241. Comme dans la première expérimentation, nous re-trions 300 premiers documents de chaque requête en prenant en compte les similarités de ces documents avec le profil utilisateur. Les articles qui sont inclus dans les profils seront exclus des résultat pour éviter un biais dans l'évaluation. Comme nous l'avons mentionné au-dessus, l'expérimentation est répétée **k** fois, c'est-à-dire les résultats de chaque requête sont re-triés **k** fois, chaque fois avec les nouveaux documents de tests.

Par contre, dans l'expérimentation précédente, la similarité document-profil avait été calculée avec plusieurs approches basées sur les citations (co-citations sur le Web, co-citations avec Web of Science, couplage bibliographique). Dans les nouvelles expérimentations, on utilise seulement la méthode des co-citations sur le Web et la méthode du couplage bibliographique qui ont donné les meilleures performances dans l'expérimentation précédente comme approche basée sur les citations. Par ailleurs, on ajoute la méthode basée sur le contenu textuel pour calculer cette similarité. Dans l'approche basée sur le

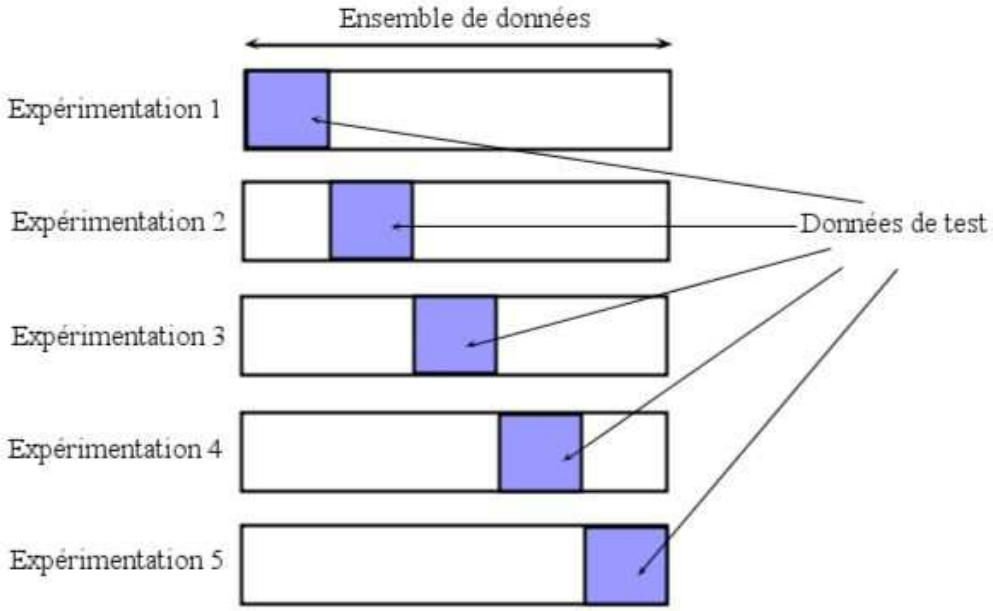

FIG. 7.8 – Validation croisée à k blocs.

contenu textuel, la similarité document-profil est calculée par le modèle vectoriel (cosinus) en utilisant le logiciel Zettair. Contrairement à la première expérimentation où le score final est une combinaison entre seulement deux scores, dans la deuxième expérimentation le score final d'un document sera une combinaison entre quatre scores suivants : i) score original calculé par Zettair ii) similarité document-profil calculée par la méthode des co-citations sur le Web iii) similarité document-profil calculée par la méthode du couplage bibliographique et iv) similarité document-profil basée sur le contenu textuel. Le nombre de scores qui sont pris en compte est varié de 2 à 4. Dans cette nouvelle expérimentation, nous utilisons trois méthodes de combinaison : méthode de combinaison produit, méthode de combinaison linéaire, et méthode de combinaison basée sur la théorie de Dempster-Shafer. Ces trois méthodes ont été présentées dans la section 6.5.

7.4.3 Résultats

Pour évaluer la performance de différentes méthodes, nous utilisons les métriques de précision à n document (avec $n = 5\ 10\ 15\ 20\ 30$) et la précision moyenne (MAP). Le logiciel **trec_eval** est utilisé pour l'évaluation. Puisque nous utilisons l'approche de validation à **k** blocs, nous obtenons **k** valeurs de précision et **k** valeurs de MAP que nous moyennons :

$$Moyenne_des_Précisions = \frac{\sum_{i=1}^{k} précision_i}{k} \tag{7.1}$$

$$Moyenne_de_MAPs = \frac{\sum_{i=1}^{k} MAP_i}{k} \tag{7.2}$$

Les valeurs de MAP sont présentées dans le tableau 7.3, les valeurs de précisions sont présentées dans le tableau 7.4. Dans les tableaux de résultats, la deuxième ligne présente le résultat original du moteur de recherche Zettair. Les résultats obtenus par les différentes

méthodes de re-classement sont présentés dans les autres lignes (de la troisième ligne à la huitième ligne). Les scores prises en compte pour calculer le score final d'un document sont :

1. score original calculé par Zettair (**Zet.**).
2. similarité document-profil calculée par la méthode des co-citations sur le Web (**Co-cit.**).
3. similarité document-profil calculée par la méthode du couplage bibliographique (**Bib.**).
4. similarité document-profil calculée par la méthode basée sur le contenu textuel (**Contenu**), dans notre travail c'est la méthode cosinus.

En combinant ces scores, nous considérons six méthodes de re-classements suivantes :

1. **Zet. + Co-cit.** (troisième ligne) : Cette méthode combine deux scores – le score original calculé par Zettair et la similarité document-profil calculée par la méthode des co-citations sur le Web.
2. **Zet. + Bib.** (quatrième ligne) : Cette méthode combine deux scores – le score original calculé par Zettair et la similarité document-profil calculée par la méthode du couplage bibliographique.
3. **Zet. + Contenu** (cinquième ligne) : Cette méthode combine deux scores – le score original calculé par Zettair et la similarité document-profil calculée par la méthode cosinus.
4. **Zet. + Co-cit. + Contenu** (sixième ligne) : Cette méthode combine trois scores – le score original calculé par Zettair, la similarité document-profil calculée par la méthode des co-citations sur le Web, et la similarité document-profil calculée par la méthode cosinus.
5. **Zet. + Bib. + Contenu** (septième ligne) : Cette méthode combine trois score – le score original calculé par Zettair, la similarité document-profil calculée par la méthode du couplage bibliographique, et la similarité document-profil calculée par la méthode cosinus.
6. **Zet. + Co-cit. + Contenu + Bib.** (huitième ligne) : Cette méthode combine quatre scores – le score original calculé par Zettair, la similarité document-profil calculée par la méthode des co-citations sur le Web, la similarité document-profil calculée par la méthode cosinus, et la similarité document-profil calculée par la méthode du couplage bibliographique.

Pour chaque méthode, « **p** » signifie la combinaison produit (cf. la section 6.5), « **l** » signifie la combinaison linéaire (cf. la section 6.5), et « **DS** » signifie la combinaison basée sur la théorie de Dempster-Shafer (cf. la section 6.5).

La troisième colonne du tableau 7.3 présente les paramètres utilisés dans chaque méthode. Ce sont les coefficients utilisés dans la combinaison linéaire (β_i), ou les degrés de confiance utilisés dans la théorie Dempster-Shafer ($1 - \alpha_i$). Comme nous avons expliqué dans la section 6.5, dans nos travaux actuels nous essayons plusieurs combinaisons différentes pour trouver les meilleures valeurs de paramètres (coefficients dans la combinaison linéaire ou degrés de confiance dans la combinaison Dempster-Shafer). Pour chaque méthode de re-classement, les meilleures valeurs de paramètres sont celles qui donnent les

meilleures valeurs MAP (*Mean Average Precision*). Tous ces paramètres sont placés selon l'ordre correspondant des scores utilisés dans les méthodes. Par exemple, dans la méthode de re-classement **Zet. + Co-cit. + Contenu** (la sixième ligne), les coefficients du score original de Zettair, de la similarité document-profil calculée par la méthode des co-citations, et la similarité calculé par la méthode basée sur le contenu textuel sont respectivement $0,2$, $0,15$, $0,65$ (dans la combinaison linéaire), et leurs degrés de confiance sont respectivement $0,3$, $0,3$, $0,7$ (dans la méthode de combinaison basée sur la théorie de Dempster-Shafer).

Similairement, les résultats présentés dans le tableau 7.4 sont les moyens de précisions à 5, 10, 15, 20, 30 de ces méthodes. Les meilleures valeurs (MAP ou précision à n) dans ces tableaux sont écrites en valeurs grasses.

7.4.4 Discussion sur le résultat

Performance générale : A partir de ces tableaux, nous pouvons voir que les méthodes de re-classement fonctionnent bien. Toutes ces méthodes peuvent améliorer le résultat original du moteur de recherche (en terme de MAP et de précisions à n). Généralement, l'amélioration pour les précisions à 5, 10, 15 est plus nette que celle pour les précisions à 20 et à 30. Particulièrement, l'amélioration pour la précision à 15 semble être la plus nette par rapport aux autres niveaux de précision.

L'apport de la similarité basée sur le contenu textuel semble meilleur que ceux des similarités basées sur les citations dans le processus de re-classement. En effet le résultat présenté dans la cinquième colonne (**Zet. + Contenu**) est meilleur que les résultats dans les résultats présentés dans la troisième colonne (**Zet. + Co-cit.**) et dans la quatrième colonne (**Zet. + Bib.**), sauf dans la combinaison produit. C'est probablement dû à ce que les articles scientifiques, contrairement aux pages Web, contiennent normalement un contenu textuel riche, ce qui permet de trouver facilement la relation entre eux en utilisant les méthodes basées sur le contenu.

Par contre, l'utilisation combinée de plusieurs méthodes montre sa performance. Plus nous utilisons de scores, plus nous obtenons des améliorations. Les méthodes de re-classement utilisant trois scores dans les sixième et septième lignes surpassent celles qui utilisent seulement deux scores, en terme de MAP et de précisions à n. La méthode de re-classement utilisant quatre scores présentée dans la huitième colonne obtient la meilleure valeur de MAP par rapport à toutes les autres méthodes et elle donne de très bonnes précisions à 5 documents. Cependant, la différence totale entre cette méthode et les deux méthodes qui utilisent trois scores n'est pas aussi nette que la différence entre les méthodes utilisant trois scores avec les méthodes utilisant deux scores. Par exemple, l'amélioration de performance en terme de MAP de cette méthode par rapport à la méthode **Zet. + Co-cit. + Contenu** n'est que 2,89% pour la combinaison linéaire et 2,68% pour la combinaison Dempster-Shafer. Peut-être quand on a plus de scores, il devient plus difficile de trouver des paramètres appropriés pour les combiner (coefficients dans la combinaison linéaire ou degrés de confiance dans la combinaison de Dempster-Shafer). Une autre raison possible est que la nouvelle source n'apporte pas beaucoup de nouvelles informations par rapport aux trois autres sources. C'est peut-être également une combinaison de ces deux raisons.

L'impact des méthodes de combinaison : Parmi les trois méthodes de combinaison, la combinaison linéaire et la combinaison basée sur la théorie de Dempster-Shafer semblent être meilleures que la combinaison produit. La combinaison linéaire, malgré sa simplicité, fonctionne très bien et elle est un peu meilleure que la combinaison basée sur la théorie de Dempster-Shafer dans quatre sur six des méthodes de re-classement en terme de MAP. Ce sont des méthodes de combinaisons suivantes : **Zet. + Co-cit.**, **Zet. + Bib.**, **Zet. + Co-cit. + Contenu** , et **Zet. + Co-cit. + Contenu + Bib.**. Dans deux autres cas (**Zet. + Contenu**, **Zet. + Bib. + Contenu**), la combinaison Dempster-Shafer est la meilleure.

Pour la métrique de précisions à n, la combinaison linéaire est la meilleure pour la précision à 10 (**Zet. + Co-cit. + Contenu**), à 20 (**Zet. + Bib. + Contenu**), et à 30 (**Zet. + Bib. + Contenu**). La combinaison Dempster-Shafer est la meilleure à la précision à 15 (**Zet. + Co-cit. + Contenu + Bib.**). Cependant, la combinaison linéaire et la combinaison basée sur la théorie Dempster-Shafer sont comparables. La différence de performance entre ces deux méthodes n'est pas grande comme celle entre ces méthodes avec la combinaison produit.

Comme nous l'avons mentionné dans la section 6.5, contrairement au cas de la combinaison linéaire où les coefficients doivent satisfaire la condition $\sum_i \beta_i = 1$, dans ce cas de la combinaison Dempster-Shafer il n'y a pas de contraintes explicites entre les degrés de confiance de différentes sources d'évidences $1-\alpha_i$, et le nombre des valeurs possibles à envisager est beaucoup grand. Dans cette méthode nous élargissons les pas de discrétisation afin de réduire le nombre de combinaisons dans des limites calculables. Cependant, en réduisant le nombre de combinaisons, on réduit également la chance de trouver les paramètres optimaux pour cette méthode. C'est peut-être une raison pourquoi la combinaison linéaire est un peu meilleure que la méthode de combinaison Dempster-Shafer.

7.5 Bilan

Nous venons de présenter les expérimentations que nous avons conduites pour valider notre approche. Nous simulons différents scénarios de recherche d'information personnalisée dans une bibliothèque numérique. Dans le premier scénario, l'utilisateur doit préciser explicitement les documents qui représentent ses intérêts. Dans le deuxième scénario, l'utilisateur ne précise pas explicitement les documents qui représentent ses intérêts mais le système doit le faire en utilisant les techniques de construction de profils utilisateurs.

Les expérimentations confirment notre hypothèse : utilisation des méthodes basées sur les citations à côté des méthodes basées sur le contenu textuel traditionnelles peut améliorer la performance des systèmes de RI personnalisés des bibliothèques numériques scientifiques. Cependant, la performance de la méthode des co-citations est influencée par la base de données bibliographique utilisée. De plus, combinaison de plusieurs méthodes pour calculer les similarités documents-profils peut donner plus de performance, mais il est plus difficile de trouver les paramètres optimaux quand le nombre de méthodes à combiner augmente. Dans le chapitre suivant, nous allons conclure et discuter sur quelques perspectives.

TAB. 7.3 – Moyennes de MAP

Méthode	MAP	Paramètres (β_i / $1-\alpha_i$)
Zet.	0,2631	
Zet. + Co-cit.	0,3076 (p) (+16,89%)	
	0,3098 (l) (+17,73%)	0,45 0,55
	0,3076 (DS) (+16,89%)	1,0 1,0
Zet. + Bib.	0,2959 (p) (+12,44%)	
	0,3020 (l) (+14,77%)	0,35 0,65
	0,2993 (DS) (+13,73%)	0,8 0,9
Zet. + Contenu	0,2938 (p) (+11,67%)	
	0,3208 (l) (+21,91%)	0,25 0,75
	0,3254 (DS) (+23,66%)	0,75 0,95
Zet. + Co-cit. + Contenu	0,3213 (p) (+22,12%)	
	0,3421 (l) (+30,0%)	0,2 0,15 0,65
	0,3404 (DS) (+29,35%)	0,3 0,3 0,7
Zet. + Bib. + Contenu	0,3202 (p) (+21,70%)	
	0,3366 (l) (+27,91%)	0,2 0,1 0,7
	0,3372 (DS) (+28,13%)	0,2 0,2 0,6
Zet. + Co-cit. + Contenu + Bib.	**0,3402** (p) (+29,28%)	
	0,3497 (l) (+32,89%)	0,2 0,2 0,3 0,3
	0,3474 (DS) (+32,03%)	0,3 0,3 0,7 0,3

TAB. 7.4 – Moyennes des précisions à 5, 10, 15, 20, 30 documents

Méthode	5 docs	10 docs	15 docs	20 docs	30 docs
Zet.	0,2892	0,2123	0,1672	0,1473	0,1154
Zet. + Co-cit.	0,3292 (p) +13,84%	0,2446 (p) +15,20%	0,1974 (p) +18,11%	0,1669 (p) +13,32%	0,1233 (p) +6,88%
	0,3354 (l) +15,96%	0,2431 (l) +14,49%	0,2000 (l) +19,66%	0,165 (l) +12,02%	0,1221 (l) +5,77%
	0,3292 (DS) +13,84%	0,2446 (DS) +15,20%	0,1974 (DS) +18,11%	0,1669 (DS) +13,32%	0,1233 (DS) +6,88%
Zet. + Bib.	0,3246 (p) +12,24%	0,2446 (p) +15,21%	0,1887 (p) +12,91%	0,1596 (p) +8,36%	0,1223 (p) +5,98%
	0,3400 (l) +17,56%	0,2423 (l) +14,13%	0,1928 (l) +15,36%	0,1608 (l) +9,15%	0,1210 (l) +4,87%
	0,3292 (DS) +13,84%	0,2477 (DS) +16,66%	0,1938 (DS) +15,96%	0,1635 (DS) +10,98%	0,1238 (DS) +7,31%
Zet. + Contenu	0,3185 (p) +10,11%	0,2362 (p) +11,24%	0,1959 (p) +17,19%	0,1677 (p) +13,85%	0,1274 (p) 10,40%
	0,3462 (l) +19,69%	0,2715 (l) +27,89%	0,2174 (l) +30,08%	0,1815 (l) +23,25%	0,1374 (l) +19,10%
	0,3539 (DS) +22,35%	0,2677 (DS) +26,07%	0,2185 (DS) +30,69%	0,1788 (DS) +21,41%	0,1356 (DS) +17,54%
Zet. + Co-cit. + Contenu	0,3446 (p) +19,15%	0,2615 (p) +23,18%	0,2164 (p) +29,47%	0,1777 (p) +20,62%	0,1333 (p) +15,53%
	0,3661 (l) +26,60%	**0,2877** (l) +35,49%	0,2185 (l) +30,70%	0,1823 (l) +23,76%	0,1377 (l) +19,31%
	0,3569 (DS) +23,41%	0,2823 (DS) +32,96%	0,2190 (DS) +31,00%	0,1808 (DS) +22,72%	0,1382 (DS) +19,77%
Zet. + Bib. + Contenu	0,3400 (p) +17,56%	0,2592 (p) +22,10%	0,2046 (p) +22,41%	0,1750 (p) +18,79%	0,1315 (p) +13,97%
	0,3616 (l) +25,01%	0,2731 (l) +28,61%	0,221 (l) +32,21%	**0,1839** (l) +24,82%	**0,1387** (l) +20,21%
	0,3585 (DS) +23,95%	0,2762 (DS) +30,07%	0,2195 (DS) +31,31%	0,1831 (DS) +24,29%	0,1377 (DS) +19,32%
Zet. + Co-cit. + Contenu + Bib.	0,3723 (p) +28,73%	0,2731 (p) +28,62%	0,2159 (p) +29,16%	0,1785 (p) +21,17%	0,1341 (p) +16,20%
	0,3769 (l) +30,32%	0,2684 (l) +26,44%	0,2149 (l) +28,54%	0,1804 (l) +22,47%	0,1382 (l) +19,76%
	0,3769 (DS) +30,32%	0,2815 (DS) +32,59%	**0,2205** (DS) +31,92%	0,1823 (DS) +23,77%	0,1382 (DS) +19,77%

Chapitre 8

Conclusions et perspectives

8.1 Conclusions générales

Dans les années récentes, le problème de personnalisation de systèmes de recherche d'information est devenu de plus en plus important. Il s'agit de systèmes « intelligents » qui peuvent s'adapter aux besoins d'informations spécifiques des utilisateurs. Il y a plusieurs façons pour personnaliser un système de recherche d'information, mais tous ces approches sont basées sur l'utilisation de profils utilisateurs qui sont les informations décrivant les intérêts et/ou préférences des utilisateurs.

De plus, avec le développement de technologies, les bibliothèques numériques se sont développées avec une grande vitesse. Une bibliothèque numérique est beaucoup plus qu'une collection de documents numériques. Ceux-ci sont sélectionnés, organisés, préservés et la bibliothèque fournit un ensemble des services utiles pour les utilisateurs. Parmi ces services, le service de recherche d'information est un service indispensable de ces bibliothèques. C'est un atout des bibliothèques numériques par rapport aux bibliothèques traditionnelles. Dans cette thèse, nous abordons un problème important pour les bibliothèques numériques scientifiques : la personnalisation du service de recherche d'information de ces bibliothèques.

Le contexte de la thèse concerne plusieurs sujets différents : la recherche d'information, les systèmes personnalisés, les bibliothèques numériques, les méthodes de citations dans le domaine de la bibliométrie. Nous avons commencé par des études sur tous ces domaines. Plus particulièrement, nous nous concentrons sur le problème de re-classement de résultats d'un moteur de recherche en prenant en compte les similarités de ces documents avec le profil utilisateur. Bien qu'il y ait déjà plusieurs travaux sur les systèmes de recherche d'information personnalisée, mais la plupart de ces systèmes n'utilisent que les approches basées sur le contenu textuel pour représenter les documents et les profils utilisateurs et pour calculer les similarités entre eux dans la phase de re-classement des résultats de recherche.

Dans nos travaux, nous nous concentrons sur l'aspect de calcul la similarité entre document-profil dans la phase de re-classement de résultats de recherche. L'originalité de nos travaux est d'étudier et d'appliquer les méthodes basées sur les citations pour ce but à côté des méthodes traditionnelles basées sur le contenu textuel. Les méthodes basées sur les citations que nous utilisons sont la méthode des co-citations et la méthode du couplage bibliographique. Nous utilisons la base de données Thomson ISI pour trouver la

relation entre les articles scientifiques dans la méthodes des co-citations. Nous proposons également la méthode des co-citations sur le Web qui utilise le moteur de recherche Google pour ce but. Nous utilisons également différentes méthodes de combinaison pour comparer les performances de ces méthodes quand elles sont utilisées pour combiner des scores individuels.

Pour valider nos approches, nous utilisons une approche de simulation avec une collection de test de l'INEX. Les expérimentations simulent différents scénarios d'un système de recherche d'information personnalisée dans lesquels les utilisateurs peuvent sélectionner manuellement les documents pertinents pour construire les profils ou le système sélectionne automatiquement ces documents.

Les résultats des expérimentations confirment l'efficacité de nos approches. L'application des méthodes basées sur les citations à côté des méthodes basées sur le contenu textuel peut améliorer les performances des systèmes de recherche d'information des bibliothèques numériques scientifiques. Cependant, l'efficacité de la méthode des co-citations est largement dépendante de la base de données bibliographique utilisée. Si la couverture de la base de données n'est pas suffisante, cette méthode ne peut pas faire valoir son efficacité. De plus, la combinaison de plusieurs méthodes pour calculer la similarité document-profil peut amener à de meilleurs résultats par rapport à l'utilisation d'une seule méthode. Dans nos expérimentations, plus nous utilisons de scores, plus nous obtenons des améliorations.

8.2 Perspectives

Il y a plusieurs pistes que nous pouvons continuer pour améliorer nos travaux dans le futur :

Combiner plusieurs bases de données bibliographiques : Dans une étude récente [90], Meho et al. ont conduit une recherche pour analyser les citations vers les publications des membres de *School of Library and Information Science* de l'université Indiana. Les bases de données bibliographiques utilisées sont : Web of Science, Scopus, Google Scholar. Les auteurs ont trouvé que le chevauchement des citations de ces bases de données est assez faible. Le chevauchement de citations entre WoS et Scopus est seulement 58.2%. Google Scholar identifie plus de 53% de citations par rapport à l'union de WoS et Scopus. Google Scholar identifie 4181 citations tandis que l'union de WoS et scopus identifie 2733 citations . Le chevauchement entre Google Scholar et l'union de WoS et Scopus est considérablement faible : le nombre de citations communes est 1629 parmi 5285 citations uniques (c'est-à-dire 30,8%).

A partir de ce fait, nous pouvons supposer que si nous combinons plusieurs bases de données bibliographiques ensemble, c'est peut-être meilleur que l'utilisation d'une seule base de données pour trouver la similarité entre les articles scientifiques. Une idée similaire ont été appliquée pour les méta-moteurs de recherche. Il y a des études qui montrent que les différents moteurs de recherche retournent différents documents pour une même requête [118]. Alors si on utilise un seul moteur de recherche, on peut manquer plusieurs résultats pertinents. La raison est que chaque moteur dispose d'une collection de documents et d'un algorithme de classement très différent. Alors et les méta-moteurs de recherche comme

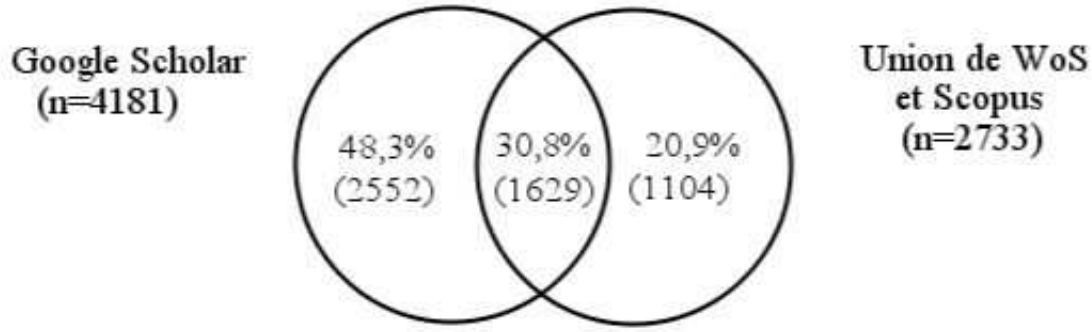

FIG. 8.1 – Chevauchement de citations entre WoS, Scopus, et Google Scholar [90].

metacrawler[1] ont utilisé cette caractéristiques pour combiner les résultats de plusieurs moteurs de recherche afin de donner une meilleure performance.

Faire des expérimentations sur les autres collections : Jusqu'à maintenant, on utilise seulement la collection de test INEX pour faire des expérimentations. Bien que ce ne soit pas facile de trouver d'autres collections qui satisfont toutes les conditions nécessaires pour nos travaux (cf. section 7.2), faire d'autres expérimentations sur d'autres collections de test permettrait de mieux valider nos approches. Ce serait encore mieux si on peut trouver une façon d'évaluer nos approches sur une vraie bibliothèque numérique avec les vrais utilisateurs.

De plus, nous savons qu'il y a des similarités entre les citations et les hyperliens Web. La recherche d'information sur le Web est aussi un axe importante. Alors c'est peut-être intéressant si on peut conduire une expérimentations de recherche d'information personnalisée sur les collections Web en appliquant les méthodes des co-citations et du couplage bibliographique.

Utilisation des méthodes d'apprentissage automatique : Dans les méthodes de combinaison linéaire et Dempster-Shafer, nous utilisons une approche de type « brute-force » pour générer les combinaisons des valeurs des paramètres nécessaires (coefficients dans la méthode linéaire ou degrés de confiance dans la méthode Dempster-Shafer), ce qui requiert beaucoup de temps pour essayer les combinaisons et trouver celle qui donne les meilleurs performances. Une des approches possibles pour améliorer ce problème est d'utiliser des méthodes d'apprentissage automatique ou des heuristiques pour trouver ces paramètres.

[1] http://www.metacrawler.com

Bibliographie

[1] Giuseppe Amato and Umberto Straccia. User profile modeling and applications to digital libraries. In *ECDL '99 : Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries*, pages 184–197, London, UK, 1999. Springer-Verlag.

[2] William Y. Arms. *Digital Libraries*. MIT Press, 2000.

[3] William Y. Arms, C. Blanchi, and E. A. Overly. An Architecture for Information in Digital Libraries. *D-Lib Magazine*, 3(2), 1997.

[4] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[5] Marko Balabanovic. An adaptive web page recommendation service. In *AGENTS '97 : Proceedings of the first international conference on Autonomous agents*, pages 378–385, New York, NY, USA, 1997. ACM Press.

[6] Marko Balabanovic and Yoav Shoham. Fab : content-based, collaborative recommendation. *Commun. ACM*, 40(3) :66–72, 1997.

[7] Kathleen Bauer and Nisa Bakkalbasi. An examination of citation counts in a new scholarly communication environment. *D-Lib Magazine*, September 2005.

[8] Nicholas J. Belkin and W. Bruce Croft. Information filtering and information retrieval : two sides of the same coin ? *Commun. ACM*, 35(12) :29–38, 1992.

[9] Donna Bergmark. Collection synthesis. In *Proceedings of the Second ACM/IEEECS Joint Conference on Digital Libraries (Portland OR, 2002)*, 2002.

[10] Daniel Billsus and Michael J. Pazzani. A hybrid user model for news story classification. In *UM '99 : Proceedings of the seventh international conference on User modeling*, pages 99–108, Secaucus, NJ, USA, 1999. Springer-Verlag New York, Inc.

[11] Lennart Bjorneborn. *Small-world link structures across an academic web space : a library and information science approach. PhD dissertation.* PhD thesis, Royal School of Library and Information Science, Copenhagen, 2004.

[12] Lennart Björneborn and Peter Ingwersen. Toward a basic framework for webometrics. *J. Am. Soc. Inf. Sci. Technol.*, 55(14) :1216–1227, 2004.

[13] Kurt Bollacker, Steve Lawrence, and C. Lee Giles. A system for automatic personalized tracking of scientific literature on the web. In *Digital Libraries 99 - The Fourth ACM Conference on Digital Libraries*, pages 105–113, New York, 1999. ACM Press.

[14] Christine L. Borgman and Jonathan Furner. Scholarly communication and bibliometrics. *Annual Review of Information Science and Technology*, 36 :3–72, 2002.

[15] Shannon Bradshaw, Andrei Scheinkman, and Kristian Hammond. Guiding people to information : providing an interface to a digital library using reference as a basis for indexing. In *IUI '00 : Proceedings of the 5th international conference on Intelligent user interfaces*, pages 37–43, New York, NY, USA, 2000. ACM Press.

[16] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, July 1998.

[17] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *WWW7 : Proceedings of the seventh international conference on World Wide Web 7*, pages 107–117, Amsterdam, The Netherlands, The Netherlands, 1998. Elsevier Science Publishers B. V.

[18] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener. Graph structure in the web. In *Proceedings of the 9th international World Wide Web conference on Computer networks : the international journal of computer and telecommunications netowrking*, pages 309–320, Amsterdam, The Netherlands, The Netherlands, 2000. North-Holland Publishing Co.

[19] Jay Budzik and Kristian J. Hammond. User interactions with everyday applications as context for just-in-time information access. In *IUI '00 : Proceedings of the 5th international conference on Intelligent user interfaces*, pages 44–51, New York, NY, USA, 2000. ACM Press.

[20] Vannevar Bush. As we may think. *Atlantic Monthly*, 176 :101–108, 1945.

[21] Vishnu Kanth Reddy Challam. Contextual information retrieval using ontology based user profiles. Master's thesis, University of Kansas, 2004.

[22] Liren Chen and Katia Sycara. Webmate : a personal agent for browsing and searching. In *AGENTS '98 : Proceedings of the second international conference on Autonomous agents*, pages 132–139, New York, NY, USA, 1998. ACM Press.

[23] Mark Claypool, Phong Le, Makoto Wased, and David Brown. Implicit interest indicators. In *IUI '01 : Proceedings of the 6th international conference on Intelligent user interfaces*, pages 33–40, New York, NY, USA, 2001. ACM Press.

[24] C. W. Cleverdon, J. Mills, and M. Keen. *Factors Determining the Performance of Indexing Systems Vol. 1 Design Vol.II Test Results*. ASLIB Cranfield Project, 1966.

[25] Thierson Couto, Marco Cristo, Marcos André Goncalves, Pável Calado, Nivio Ziviani, Edleno Silva de Moura, and Berthier A. Ribeiro-Neto. A comparative study of citations and links in document classification. In *JCDL '06*, 2006.

[26] Nick Craswell, David Hawking, and Stephen Robertson. Effective site finding using link anchor information. In *SIGIR '01 : Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 250–257, New York, NY, USA, 2001. ACM.

[27] Mariam Daoud, Lynda Tamine-Lechani, Mohand Boughanem, and Bilal Chebaro. Construction des profils utilisateurs à base d'ontologie pour une recherche d'information personnalisée. In *5ème Conférence en Recherche d'Information et Applications (CORIA)*, mars 2008.

[28] D.J. de Solla Price. Citation measures of hard science, soft science, technology, and non-science. In C.E. Nelson and D.K. Pollock, editors, *Communication among Scientists and Engineers*, pages 3–22, 1970.

[29] Jeffrey Dean and Monika R. Henzinger. Finding related pages in the world wide web. In *WWW '99 : Proceeding of the eighth international conference on World Wide Web*, pages 1467–1479, New York, NY, USA, 1999. Elsevier North-Holland, Inc.

[30] Lyes Dekar and Hamamache Kheddouci. A cluster based mobility prediction scheme for ad hoc networks. *Ad Hoc Netw.*, 6(2) :168–194, 2008.

[31] A.P. Dempster. A generalization of bayesian inference. *Journal of Royal Statistical Society*, 30(2) :205–247, 1968.

[32] Chen Ding, Chi-Hung Chi, Jing Deng, and Chun-Lei Dong. Citation retrieval in digital libraries. In *In Proceeding of IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 1999.

[33] Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. A large-scale evaluation and analysis of personalized search strategies. In *WWW '07 : Proceedings of the 16th international conference on World Wide Web*, pages 581–590, New York, NY, USA, 2007. ACM.

[34] Daniel Dreilinger and Adele E. Howe. Experiences with selecting search engines using metasearch. *ACM Trans. Inf. Syst.*, 15(3) :195–222, 1997.

[35] Miles Efron. The liberal media and right-wing conspiracies : using cocitation information to estimate political orientation in web documents. In *CIKM '04 : Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 390–398, New York, NY, USA, 2004. ACM Press.

[36] Leo Egghe and Ronald Rousseau. *Introduction to Informetrics : quantitative methods in library, documentation and information science.* Elsevier Science, 1990.

[37] Aydin Erar. Bibliometrics or informetrics : Displaying regularity in scientific patterns by using statistical distributions. *Hacettepe Journal of Mathematics and Statistics*, 31 :113–125, 2002.

[38] Daniel Faensen, Lukas Faulstich, Heinz Schweppe, Annika Hinze, and Alexander Steidinger. Hermes : a notification service for digital libraries. In *ACM/IEEE Joint Conference on Digital Libraries*, pages 373–380, 2001.

[39] Ed Fox, Lillian Cassel, Hussein Suleman, and Devika Madalli. Digital libraries. In Munindar P. Singh, editor, *Practical Handbook of Internet Computing.* Chapman Hall & CRC Press, Baton Rouge, 2004.

[40] Edward A. Fox and Joeseph A. Shaw. Combination of multiple searches. In *The Second Text REtrieval Conference (TREC-2)*, volume 500-215 of *NIST Special Publication*, pages 243–252. NIST, 1994.

[41] E. Frias-Martinez, G. Magoulas, S. Chen, and R. Macredie. Automated user modeling for personalized digital libraries. *International Journal of Information Management*, 26(3) :234–248, June 2006.

[42] Eugene Garfield. Can citation indexing be automated ? In *Statistical association methods for mechanized documentation : Symposium proceedings*, 1965.

[43] Eugene Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178 :471–479, 1972.

[44] Eugene Garfield. The most-cited papers of all time, SCI 1945–1988. Part 1A. The SCI top 100—will the Lowry method ever be obliterated ? *Current Contents*, 13(7) :3–14, 1990.

[45] Susan Gauch, Jeason Chaffee, and Alaxander Pretschner. Ontology-based personalized search and browsing. *Web Intelli. and Agent Sys.*, 1(3-4) :219–234, 2003.

[46] C. Lee Giles, Kurt D. Bollacker, and Steve Lawrence. Citeseer : an automatic citation indexing system. In *DL '98 : Proceedings of the third ACM conference on Digital libraries*, pages 89–98, New York, NY, USA, 1998. ACM Press.

[47] W. Glanzel. Bibliometrics as a research field, 2003. Course Handouts.

[48] Miha Grcar, Hunja Mladenic, and Marko Grobelnik. User profiling for interest-focused browsing history. In *Proceedings of Workshop on User Aspects of the Semantic Web*, 2005.

[49] David A. Grossman and Ophir Frieder. *Information Retrieval : Algorithms and Heuristics*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[50] Daniel Gruhl, R. Guha, Ravi Kumar, Jasmine Novak, and Andrew Tomkins. The predictive power of online chatter. In *KDD '05 : Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 78–87, New York, NY, USA, 2005. ACM.

[51] Cathal Gurrin and Alan F. Smeaton. Improving the evaluation of web search systems. In *ECIR*, pages 25–40, 2003.

[52] Norbert Gövert and Gabriella Kazai. Overview of the initiative for the evaluation of xml retrieval (inex) 2002. In *In Fuhr et al*, pages 1–17. ERCIM, 2003.

[53] Donna Harman. Overview of trec-1. In *HLT '93 : Proceedings of the workshop on Human Language Technology*, pages 61–65, Morristown, NJ, USA, 1993. Association for Computational Linguistics.

[54] Anne-Wil Harzing. Google scholar - a new data source for citation analysis. Document link : http ://www.harzing.com/pop_gs.htm, February 2008.

[55] Rostaing Hervé. *La bibliométrie et ses techniques*. Sciences de la société, Toulouse, France, 1996.

[56] Charles R. Hildreth. Accounting for users' inflated assessments of on-line catalogue search performance and usefulness : an experimental study. *Information Research*, 6(2), 2001.

[57] William W. Hood and Concepción S. Wilson. The literature of bibliometrics, scientometrics, and informetrics. *Scientometrics*, 52(2) :291–314, 2001.

[58] Shen Huang, Gui-Rong Xue, Ben-Yu Zhang, Zheng Chen, Yong Yu, and Wei-Ying Ma. Tssp : A reinforcement algorithm to find related papers. In *WI '04 : Proceedings of the Web Intelligence, IEEE/WIC/ACM International Conference on (WI'04)*, pages 117–123, Washington, DC, USA, 2004. IEEE Computer Society.

[59] Peter Ingwersen. The calculation of web impact factors. *Journal of Documentation*, 54(2) :236–243, March 1998.

[60] Peter Jacso. As we may search : Comparison of major features of the web of science, scopus, and google scholar citation-based and citation-enhanced databases. *Current Science*, 89(9) :1537–1547, 2005.

[61] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR '05 : Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 154–161, New York, NY, USA, 2005. ACM Press.

[62] Robert Kahn and Robert Wilensky. A framework for distributed digital object services. Technical Report cnri.dlib/tn95-01, CNRI, 1995.

[63] Gabriella Kazai and Mounia Lalmas. INEX 2005 evaluation measures. In *4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005*, pages 16–29, 2005.

[64] Diane Kelly and Jaime Teevan. Implicit feedback for inferring user preference : a bibliography. *SIGIR Forum*, 37(2) :18–28, 2003.

[65] M. M. Kessler. Bibliographic coupling between scientific papers. *American Documentation*, 14(1) :10–25, 1963.

[66] Hyoung R. Kim and Philip K. Chan. Learning implicit user interest hierarchy for context in personalization. In *IUI '03 : Proceedings of the 8th international conference on Intelligent user interfaces*, pages 101–108, New York, NY, USA, 2003. ACM Press.

[67] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5) :604–632, 1999.

[68] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, pages 1137–1145, 1995.

[69] Joseph A. Konstan, Bradley N. Miller, David Maltz, Jonathan L. Herlocker, Lee R. Gordon, and John Riedl. Grouplens : applying collaborative filtering to usenet news. *Commun. ACM*, 40(3) :77–87, 1997.

[70] Kayvan Kousha and Mike Thelwall. Google scholar citations and google weburl citations : A multi-discipline exploratory analysis. In *Proceedings International Workshop on Webometrics, Informetrics and Scientometrics & Seventh COLLNET Meeting*, 2006.

[71] Kayvan Kousha1 and Mike Thelwall. Motivations for url citations to open access library and information science articles. *Scientometrics*, 68(3) :501–517, 2006.

[72] Gerald Kowalski. *Information Retrieval Systems : Theory and Implementation.* Kluwer Academic Publishers, Norwell, MA, USA, 1997.

[73] Bruce Krulwich. Lifestyle finder : Intelligent user profiling using large-scale demographic data. *AI Magazine*, 18(2) :37–45, 1997.

[74] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. Trawling the web for emerging cyber-communities. In *WWW '99 : Proceeding of the eighth international conference on World Wide Web*, pages 1481–1493, New York, NY, USA, 1999. Elsevier North-Holland, Inc.

[75] Kuei-Kuei Lai and Shiao-Jun Wu. Using the patent co-citation approach to establish a new patent classification system. *Information Processing and Management*, 41(2) :313–330, 2005.

[76] Mounia Lalmas and Anastasios Tombros. Inex 2002 - 2006 : Understanding xml retrieval evaluation. In Costantino Thanos, Francesca Borri, and Leonardo Candela, editors, *DELOS Conference*, volume 4877 of *Lecture Notes in Computer Science*, pages 187–196. Springer, 2007.

[77] Birger Larsen. *References and citations in automatic indexing and retrieval systems : experiments with the boomerang effect.* PhD thesis, Royal School of Library and Information Science, Copenhagen, 2004.

[78] Steve Lawrence, C. Lee Giles, and Kurt Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6) :67–71, 1999.

[79] Cuong Anh Le, Van-Nam Huynh, and Akira Shimazu. An evidential reasoning approach to weighted combination of classifiers for word sense disambiguation. In Petra Perner and Atsushi Imiya, editors, *MLDM 2005*, volume 3587 of *Lecture Notes in Computer Science*, pages 516–525. Springer, 2005.

[80] Joon Ho Lee. Analyses of multiple evidence combination. In *SIGIR '97 : Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–276, New York, NY, USA, 1997. ACM.

[81] Philippe Lefèvre. *La recherche d'informations : du texte intégral au thésaurus.* Hermès, 2000.

[82] Michael Lesk. *Practical digital libraries : books, bytes, and bucks.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1997.

[83] Joseph Carl Robnett Licklider. *Libraries of the Future.* The MIT Press, 1965.

[84] H. Lieberman. Letizia : An agent that assists web browsing. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Montreal, August 1995.

[85] Saadia Malik, Gabriella Kazai, Mounia Lalmas, and Norbert Fuhr. Overview of INEX 2005. In *INEX*, pages 1–15, 2005.

[86] I.V. Marshakova. System of document connections based on references. *Nauchno-Tekhnicheskaya Informatsiya Seriya 2 – Informatsionnye Protsessy i Sistemy*, pages 3–8, 1973.

[87] Andrew McCallum and K. Nigam. A comparison of event models for naive bayes text classification. In *Proc. AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[88] Sean M. McNee, Istvan Albert, Dan Cosley, Prateep Gopalkrishnan, Shyong K. Lam, Al Mamunur Rashid, Joseph A. Konstan, and John Riedl. On the recommending of citations for research papers. In *CSCW '02 : Proceedings of the 2002 ACM conference on Computer supported cooperative work*, pages 116–125, New York, NY, USA, 2002. ACM.

[89] Lokman I. Meho and Kiduk Yang. Multi-faceted approach to citation-based quality assessment for knowledge management. In *World Library and Information Congress : 72nd IFLA General Conference and Council*, 2006.

[90] Lokman I. Meho and Kiduk Yang. A new era in citation and bibliometric analyses : Web of science, scopus, and google scholar. *Journal of the American Society for Information Science and Technology*, 58(13) :2105–2125, 2007.

[91] Stuart E. Middleton, David C. De Roure, and Nigel R. Shadbolt. Capturing knowledge of user preferences : ontologies in recommender systems. In *K-CAP '01 : Proceedings of the 1st international conference on Knowledge capture*, pages 100–107, New York, NY, USA, 2001. ACM Press.

[92] Stuart E. Middleton, Nigel R. Shadbolt, and David C. De Roure. Capturing interest through inference and visualization : ontological user profiling in recommender systems. In *K-CAP '03 : Proceedings of the 2nd international conference on Knowledge capture*, pages 62–69, New York, NY, USA, 2003. ACM Press.

[93] Stefano Mizzaro. How many relevances in information retrieval ? *Interacting with Computers*, 10(3) :303–320, 1998.

[94] Dunja Mladenic. Text-learning and intelligent agents. Technical Report Technical Report IJS-DP-7948, J. Stefan Institute, Department for Intelligent Systems, Ljubljana, 1998.

[95] Miquel Montaner, Beatriz López, and Josep Lluís De La Rosa. A taxonomy of recommender agents on theinternet. *Artif. Intell. Rev.*, 19(4) :285–330, 2003.

[96] Erich J. Neuhold, Claudia Niederée, and Avare Stewart. Personalization in digital libraries - an extended view. In *ICADL*, pages 1–16, 2003.

[97] Thomas E. Nisonger. Citation autobiography : An investigation of isi database coverage in determining author citedness. *College & Research Libraries*, 65(2) :152–163, 2004.

[98] D. Oard and J. Kim. Modeling information content using observable behavior. In *In Proceedings of the 64 Annual Meeting of the American Society for Information Science and Technology, USA*, 2001.

[99] Andreas Paepcke, Chen-Chuan K. Chang, Terry Winograd, and Héctor García-Molina. Interoperability for digital libraries worldwide. *Commun. ACM*, 41(4) :33–42, 1998.

[100] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking : Bringing order to the Web. Technical report, Stanford Digital Library Technologies Project, 1998.

[101] Gautam Pant, Kostas Tsioutsiouliklis, Judy Johnson, and C. Lee Giles. Panorama : extending digital libraries with topical crawlers. In *JCDL '04 : Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, pages 142–150, New York, NY, USA, 2004. ACM.

[102] M. Pazzani, J. Muramatsu, and D. Billsus. Syskill & Webert : Identifying interesting web sites. In *Proc. AAAI Spring Symposium on Machine Learning in Information Access*, 1996.

[103] Michael Pazzani and Daniel Billsus. Learning and revising user profiles : The identification ofinteresting web sites. *Mach. Learn.*, 27(3) :313–331, 1997.

[104] Jovan Pehcevski, James A. Thom, and Seyed M. M. Tahaghoghi. RMIT university at INEX 2005 : Ad hoc track. In *INEX*, 2005.

[105] James Pitkow and Peter Pirolli. Life, death, and lawfulness on the electronic frontier. In *CHI '97 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 383–390, New York, NY, USA, 1997. ACM Press.

[106] M.F. Porter. An algorithm for suffix stripping. *Program*, 14(3) :130–137, July 1980.

[107] A. Pretschner and S. Gauch. Personalization on the web. Technical Report ITTC-FY2000-TR-13591-01, University of Kansas, 1999.

[108] Camille Prime-Claverie, Michael Beigbeder, and Thierry Lafouge. Transposition of the cocitation method with a view to classifying web pages. *J. Am. Soc. Inf. Sci. Technol.*, 55(14) :1282–1289, 2004.

[109] P.Krishna Reddy and Masaru Kitsuregawa. Inferring web communities through relaxed cocitation and dense bipartite graphs. In *Data Base Engineering Workshop*, 2001.

[110] M. Elena Renda and Umberto Straccia. A personalized collaborative digital library environment. In *ICADL*, pages 262–274, 2002.

[111] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens : an open architecture for collaborative filtering of netnews. In *CSCW '94 : Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186, New York, NY, USA, 1994. ACM Press.

[112] Stephen E. Robertson and Ian Soboroff. The trec 2002 filtering track report. In *TREC*, 2002.

[113] J. Rocchio. *Relevance Feedback in Information Retrieval.* G. Salton (editor), The SMART Retrieval System : Experiments in Automatic Document Processing. Prentice–Hall, Inc., Englewood Cliffs, NJ, 1971.

[114] U. Rohini and Vamshi Ambati. A collaborative filtering based re-ranking strategy for search in digital libraries. In *ICADL*, pages 194–203, 2005.

[115] Jacques Savoy. A stemming procedure and stopword list for general french corpora. *J. Am. Soc. Inf. Sci.*, 50(10) :944–952, 1999.

[116] Andrea SCHARNHORST and Mike THELWALL. Citation and hyperlink networks. *Current science*, 89(9) :1518–1523, November 2005.

[117] B. R. Schatz. Information retrieval in digital libraries : Bringing search to the Net. *Science*, 275 :327–334, 1997.

[118] E. Selberg and O. Etzioni. The MetaCrawler architecture for resource aggregation on the Web. *IEEE Expert*, 12(1) :11–14, 1997.

[119] G. Semeraro, M.F. Costabile, F. Esposito, N. Fanizzi, and S. Ferilli. Machine learning techniques for adaptive user interfaces in a corporate digital library service. In *ACAI-99 Workshop on Machine Learning in User Modeling*, 1999.

[120] Kari Sentz and Scott Ferson. Combination of evidence in dempster-shafer theory. Technical Report SAND 2002-0835, SAND, 2002.

[121] Young-Woo Seo and Byoung-Tak Zhang. A reinforcement learning agent for personalized information filtering. In *IUI '00 : Proceedings of the 5th international conference on Intelligent user interfaces*, pages 248–251, New York, NY, USA, 2000. ACM Press.

[122] G. Shafer. *A mathematical theory of evidence*. Princeton university press, 1976.

[123] Upendra Shardanand and Pattie Maes. Social information filtering : algorithms for automating ẅord of mouth.̈ In *CHI '95 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[124] Xuehua Shen, Bin Tan, and ChengXiang Zhai. Implicit user modeling for personalized search. In *CIKM '05 : Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 824–831, New York, NY, USA, 2005. ACM Press.

[125] Ahu Sieg, Bamshad Mobasher, and Robin Burke. Web search personalization with ontological user profiles. In *CIKM '07 : Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 525–534, New York, NY, USA, 2007. ACM.

[126] Ahu Sieg, Bamshad Mobasher, and Robin D. Burke. Inferring user's information context : Integrating user profiles and concept hierarchies. In *In Proceedings of the 2004 Meeting of the International Federation of Classification Societies*, 2004.

[127] Börkur Sigurbjörnsson, Andrew Trotman, Shlomo Geva, Mounia Lalmas, Birger Larsen, and Saadia Malik. INEX 2005 guidelines for topic development. http ://inex.is.informatik.uni-duisburg.de/2005/internal/pdf/TD05.pdf, 2005.

[128] A. Singh and K. Nakata. Hierarchical classification of web search results using personalized ontologies. In *Proceedings of HCI International 2005*, Las Vegas, 2005.

[129] H. G. Small. Co-citation in the scientific literature : A new measure of the relationship between two documents. *Journal of American Society for Information Science*, 24(4) :265–269, 1973.

[130] Gabriel L. Somlo and Adele E. Howe. Incremental clustering for profile maintenance in information gathering web agents. In *AGENTS '01 : Proceedings of the fifth international conference on Autonomous agents*, pages 262–269, New York, NY, USA, 2001. ACM Press.

[131] Fei Song and W. Bruce Croft. A general language model for information retrieval. In *CIKM '99 : Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321, New York, NY, USA, 1999. ACM.

[132] Mirco Speretta. Personalizing search based on user search histories. Master's thesis, University of Kansas, 2004.

[133] Mirco Speretta and Susan Gauch. Personalizing search based on user search histories. In *Thirteenth International Conference on Information and Knowledge Management (CIKM)*, 2004.

[134] Amanda Spink, Seda Ozmutlu, Huseyin C. Ozmutlu, and Bernard J Jansen. U.s. versus european web searching trends. *SIGIR Forum*, 36(2) :32–38, 2002.

[135] Kazunari Sugiyama, Kenji Hatano, and Masatoshi Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW '04 : Proceedings of the 13th international conference on World Wide Web*, pages 675–684, New York, NY, USA, 2004. ACM Press.

[136] Mike Thelwall. Bibliometrics to webometrics. *Journal of Information Science*, 34(4) :1–18, 2007.

[137] Mike Thelwall, Liwen Vaughan, and Lennart Björneborn. *Annual Review of Information Science and Technology 39*, chapter Webometrics, pages 81–135. 2005.

[138] Roberto Torres, Sean M. McNee, Mara Abel, Joseph A. Konstan, and John Riedl. Enhancing digital libraries with techlens+. In *JCDL '04 : Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, pages 228–236, New York, NY, USA, 2004. ACM.

[139] Roberto Torres, Sean M. McNee, Mara Abel, Joseph A. Konstan, and John Riedl. Enhancing digital libraries with techlens+. In *JCDL '04 : Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, pages 228–236, New York, NY, USA, 2004. ACM Press.

[140] Theodora Tsikrika and Mounia Lalmas. Merging techniques for performing data fusion on the web. In *CIKM '01 : Proceedings of the tenth international conference on Information and knowledge management*, pages 127–134, New York, NY, USA, 2001. ACM.

[141] Peter D. Turney and Michael L. Littman. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. *CoRR*, cs.LG/0212012, 2002.

[142] Thanh-Trung Van and Michel Beigbeder. Web co-citation : Discovering relatedness between scientific papers. In Katarzyna M. Wegrzyn-Wolska and Piotr S. Szczepaniak, editors, *5th International Atlantic Web Intelligence Conference, AWIC 2007, Fontainebleau, France, Juin 25-27, 2007*, volume 43 of *Advances in Intelligent Web Mastering*. Springer, 2007.

[143] Thanh-Trung Van and Michel Beigbeder. A comparison of re-ranking methods in digital libraries using user profiles. In *2008 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2008, 9-12 December 2008, Sydney, Australia, Main Conference Proceedings*, pages 751–754. IEEE Computer Society, December 2008.

[144] Thanh-Trung Van and Michel Beigbeder. Hybrid method for personalized search in digital libraries. In Craig Macdonald, Iadh Ounis, Vassilis Plachouras, Ian Ruthven, and Ryen W. White, editors, *Advances in Information Retrieval , 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008, poster session*, volume 4956 of *Lecture Notes in Computer Science*, pages 647–651. Springer, 2008.

[145] Thanh-Trung Van and Michel Beigbeder. Hybrid method for personalized search in scientific digital libraries. In Alexander F. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing, 9th International Conference, CICLing 2008, Haifa, Israel, February 17-23, 2008*, volume 4919 of *Lecture Notes in Computer Science*, pages 512–521. Springer, 2008.

[146] Bas van Gils and Eric D. Schabell. User-profiles for information retrieval. In *BNAIC'03 : Proceedings of the 15th Belgian-Dutch Conference on Artificial Ingelligence*, 2003.

[147] Liwen Vaughan and Debora Shaw. Bibliographic and web citations : what is the difference ? *J. Am. Soc. Inf. Sci. Technol.*, 54(14) :1313–1322, 2003.

[148] Liwen Vaughan and Debora Shaw. Web citation data for impact assessment : A comparison of four science disciplines. *J. Am. Soc. Inf. Sci. Technol.*, 56(10) :1075–1087, 2005.

[149] Peter Wilkins, Paul Ferguson, and Alan F. Smeaton. Using score distributions for query-time fusion in multimedia retrieval. In *MIR '06 : Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 51–60, New York, NY, USA, 2006. ACM.

[150] Kiduk Yang. Combining Text- and Link-based Retrieval Methods for Web IR. In *TREC*, 2001.

[151] Nesrine Zemirli, Lynda Tamine, and Mohand Boughanem. Accès personnalisé à l'information : Proposition d'un profil utilisateur multidimensionnel. In *International Symposium On Programming Systems (ISPS)*, Alger, 2005.

[152] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR '01 : Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342, New York, NY, USA, 2001. ACM.

[153] Justin Zobel and Alistair Moffat. Inverted files for text search engines. *ACM Computing Survey*, 38(2) :6, 2006.

École Nationale Supérieure des Mines
de Saint-Étienne

N° d'ordre : 501 I

Thanh-Trung VAN

Title : Using users' profiles for accessing a digital library

Speciality : Computer Science

Keyword : Personalized information retrieval, re-ranking search results, user profile, digital library, co-citation, bibliographic coupling, combination function

Abstract :
Nowadays, digital libraries are becoming popular. These libraries provide many services for their users. The information retrieval service is an important service of the digital libraries. Personalizing this service in order to better satisfy users' requirements is an approach that attract much attention of the scientific community. Many present personalized information retrieval systems re-rank results of a search engine by taking into account the similarities between these results and the user profiles to return more relevant results. However, most of these systems only use content-based approaches for this purpose. In our work, we propose to use also citation-based methods like co-citation method and bibliographic coupling method to compute document-profile similarities. We study the performance of the co-citation method with different bibliographic databases. We also use many different combination functions to combine individual scores. The proposed approaches were validated by experiments with the test collection used in INEX 2005.

École Nationale Supérieure des Mines
de Saint-Étienne

N° d'ordre : 501 I

Thanh-Trung VAN

Titre : Utilisation de profils utilisateurs pour l'accès à une bibliothèque numérique

Spécialité : Informatique

Mots clefs : Recherche d'information personnalisée, re-classement de résultats de recherche, profil utilisateur, bibliothèque numérique, co-citations, couplage bibliographique, fonction de combinaison

Résumé :
Aujourd'hui, les bibliothèques numériques deviennent de plus en plus populaires. Ces bibliothèques fournissent plusieurs services pour leurs utilisateurs. Le service de recherche d'information est un service indispensable pour ces bibliothèques. La personnalisation de ce service pour mieux répondre aux exigences des utilisateurs est une approche qui attire beaucoup d'attention de la communauté scientifique. Plusieurs systèmes de recherche d'information personnalisée actuels ont choisi de re-trier les résultats d'un moteur de recherche en prenant en compte les similarités entre ces résultats et le profil utilisateur afin de rendre des résultats plus pertinents pour les utilisateurs. Cependant, la plupart de ces systèmes n'utilise que les approches basées sur le contenu textuel pour ce but. Dans nos travaux, nous proposons d'utiliser également des méthodes basées sur les citations telles que la méthode des co-citations et la méthode du couplage bibliographique pour calculer les similarités document-profil. Nous étudions la performance de la méthode des co-citations avec différentes bases de données bibliographiques. Nous utilisons également différentes fonctions de combinaison pour combiner les scores individuels. Les approches proposées ont été validées par des expérimentations sur une collection de test utilisée dans INEX 2005.