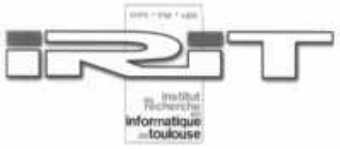
Modèle d'accès personnalisé à l'information : Structuration et Exploitation

Pascaline Tchienehom



Modèle d'accès personnalisé à l'information : Structuration et Exploitation

Rapport Interne

Réalisé par : Pascaline Tchienehom

Sous la direction de : Chantal Soulé-Dupuy
Et le co-encadrement de : Max Chevalier

Janvier 2005

Laboratoire Irit (Institut de Recherche en Informatique de Toulouse)
118, route de Narbonne, 31062 Toulouse cedex 4 - France
Équipe de recherche Sig (Systèmes d'Informations Généralisés)

Résumé

La pertinence de l'information est au centre des problématiques des techniques d'accès à l'information. Une solution à l'amélioration de cette pertinence est la personnalisation des réponses fournies aux utilisateurs. L'hétérogénéité des informations mises à disposition et des caractéristiques des usagers de ces informations a contribué à l'augmentation de l'intérêt des chercheurs pour l'amélioration des techniques de personnalisation. Ces techniques de personnalisation conditionnent, aujourd'hui, l'efficacité et la robustesse des techniques d'accès à l'information.

Nos travaux de recherche proposent dans un premier temps des modèles de structuration, de l'information mise à disposition et des usagers de cette information, à travers un cadre générique pouvant servir de base homogène pour la conception de tout type d'applications en vue d'un accès personnalisée à l'information. Ce cadre générique comprend :

1 *une architecture de recherche et de recommandation* dans laquelle tout élément (ou composant) de l'architecture est décrit de façon détaillée par un profil. L'originalité de cette architecture se situe au niveau de son aspect générique et des nombreuses possibilités d'interactions entre profils complémentaires, qu'elle offre ;

2 *un modèle générique de profil* pour la description de tout type de profil de notre architecture. Ces profils peuvent être : ré-utilisables, adaptables, multi-facettes et évolutifs.

Par la suite, nous nous sommes intéressés à l'exploitation des modèles proposés à travers la définition de méthodes d'appariement et de transformation de structure et de contenu de profils. La méthode d'appariement de profils proposée permet de calculer un degré de ressemblance d'une information à un utilisateur selon un ensemble de critères donné. La méthode de transformation de profils, quant à elle, se base sur l'usage de profils existants pour déduire une structure arborescente de profil plus simple (réduction du nombre de nœuds et du nombre de feuilles). Pour cela, on exploite à la fois la structure et le contenu des profils. Le but de la transformation est de faciliter l'appariement de profils de structure complexe (contenant plusieurs nœuds ou feuilles identiques). Cette transformation implique aussi à une modification du contenu du profil à transformer dans le profil résultant.

Table des matières

Table des figures

Liste des tableaux

Introduction générale

Le développement que connaît Internet et en particulier le World Wide Web a conduit à la mise à disposition d'une masse sans cesse croissante d'informations aux thématiques diverses, pour des utilisateurs chaque jour plus nombreux et en quête d'informations variées. Les informations sont de nature variée et complexe (bases de documents semi-structurés, mono et multimédia) et proviennent de différentes sources (bases de connaissances issues d'internet et du Web, d'intranets, de Workflows, etc.). De même, on observe également une population d'utilisateurs très hétérogène tant au niveau de l'expression de leurs besoins que de leurs caractéristiques propres (objectifs, préférences, connaissances, etc.). Cette hétérogénéité (des informations mises à disposition et des usagers de ces dernières) contribue à complexifier l'accès à l'information.

Afin de résoudre ce problème d'hétérogénéité, il est nécessaire de décrire avec le plus de détails possibles aussi bien les informations mises à disposition que les usagers de ces informations. La combinaison de ces deux descriptions va permettre la restitution d'informations adaptées à chaque usager. Pour ce faire, de nombreuses techniques d'accès à l'information ont été développées. Le but de ces processus est de fournir une information qui corresponde à un individu ou groupe d'individu donné : on parle de personnalisation. La personnalisation de l'accès à l'information va donc permettre la restitution personnalisée d'informations. Ainsi, d'un usager à un autre on n'aura pas forcément le même ensemble de résultats pour un même besoin en information, chaque usager ayant ses objectifs spécifiques, ses habitudes, ses préférences, ses pré-requis et autres dont on doit tenir compte si l'on veut répondre au mieux à ses attentes.

Aujourd'hui, les utilisateurs disposent d'un certain nombre de moyens pour rechercher, retrouver et stocker une information utile (ou lien vers cette information). Dans le but d'améliorer ces moyens et aider les utilisateurs en quête d'information, la tendance est à la modélisation aussi bien des informations mises à disposition que des usagers de ces informations. Cette modélisation conduit à la définition de profils d'informations ou d'usagers qui décrivent les caractéristiques de chaque information (mots clés, langue, taille, date, etc.) ou usager (thèmes d'intérêt, préférences, données démographiques, etc.) afin de canaliser et personnaliser les recherches. Ces profils peuvent

être construits par des processus d'annotations (de signets, de documents, de fragments de documents, etc.), de classification (de signets, de documents, de fragments de documents, etc.), de création de métadonnées, etc. Par exemple, pour construire un profil utilisateur ou un profil de groupe d'utilisateurs, on peut analyser : les collections d'informations ciblées par ces derniers (signets mémorisés, des résultats validés d'une recherche antérieure, des pages ou fragments de documents importés, etc.), les requêtes formulées par chaque usager ou groupe d'usagers, leurs parcours et habitudes de navigation, etc.

Les moyens de personnalisation existants sont variés et ont pour but d'aider tout utilisateur ou groupe d'utilisateurs à rechercher, retrouver, stocker et exploiter l'information utile, mais également à partager le résultat de ses efforts. L'objet des travaux entrepris est d'étendre les modèles et techniques d'accès à l'information (recherche et filtrage d'information) par des moyens de structuration et d'exploitation des profils, afin d'homogénéiser l'accès à l'information. Pour ce faire, nous proposons un cadre générique pour l'accès personnalisé à l'information et des méthodes d'exploitation de ce cadre, afin d'améliorer les processus de personnalisation.

Dans le chapitre 1, nous présentons une étude bibliographique sur les différentes techniques d'accès personnalisé à l'information et sur l'utilisation de profils dans ces techniques. De plus, à travers une étude comparative de systèmes de personnalisation nous définissons nos axes de recherche et positionnons nos travaux par rapport à la revue de littérature effectuée.

Dans le chapitre 2, nous présentons nos différentes contributions (cadre générique d'accès à l'information, méthodes d'appariement de profils et de transformation de structure et de contenu de profils) qui nous permettent de définir une nouvelle approche d'accès personnalisé à l'information afin de fournir des réponses plus abouties aux utilisateurs. Enfin, nous terminons par une analyse de nos contributions et une présentation de nos perspectives de recherche.

Chapitre 1

Étude bibliographique : Accès personnalisé à l'information

1.1 Introduction

Plusieurs outils d'accès à l'information ont été développés pour aider l'utilisateur à retrouver ce qu'il recherche. L'évaluation de la pertinence d'une information est au coeur de la mise en œuvre de ces différents outils. Les systèmes de personnalisation pose le postulat selon lequel la pertinence est une notion relative à chaque usager ou groupe d'usagers. Ils visent donc à évaluer la pertinence relativement à un usager ou groupe d'usagers afin d'améliorer une *pertinence système* qui serait évaluée de la même façon pour tous les utilisateurs. Le but est de se rapprocher le plus possible de la pertinence personnelle (ou attentes) des utilisateurs.

Le développement du Web a créé un besoin de techniques nouvelles pour aider les utilisateurs à trouver ce qu'ils recherchent mais aussi pour faire savoir qu'une information existe. Dans ce chapitre, nous présentons une synthèse des techniques d'accès personnalisé à l'information et des modèles de représentation associés à travers la notion de profil. Nous effectuons également une étude comparative de différents systèmes de personnalisation pour situer nos travaux de recherche par rapport à la revue de littérature effectuée.

1.2 Techniques d'accès à l'information

Les techniques d'accès à l'information permettent à un individu d'obtenir des informations répondant à ses besoins. Nous pouvons les regrouper en deux grands groupes :

– celles qui reposent sur une approche *service au comptoir* ou *pull* qui consistent à renvoyer des informations répondant à une demande explicite d'un individu. C'est le cas de la Recherche d'Information (RI) ;

– celles qui reposent sur une approche *service à domicile* ou *push* qui consistent à renvoyer automatiquement à un individu des informations qui pourraient l'intéresser, sans qu'il n'en ait fait explicitement la demande. C'est le cas du Filtrage (ou Recommandation) d'Information (FI).

Les sections 1.2.1 et 1.2.2 présentent différentes techniques d'accès à l'information au travers des processus de recherche (pull) et de filtrage d'information (push).

1.2.1 Recherche d'Information

Le processus de Recherche d'Information (RI) repose sur l'expression du besoin d'un individu au travers d'une requête formulée dans un langage libre plus ou moins structuré. En réponse à cette requête, un appariement est réalisé entre les termes (ou mots-clés) d'indexation de la requête et ceux des informations pré-indexées par le système. La recherche d'information est principalement basée sur le principe d'un appariement optimal, de type vectoriel (*cf.* section 1.3.3.2) ou probabiliste (*cf.* section 1.3.3.3) [Rij79] [BYRN99]. Enfin, le système propose traditionnellement à l'individu les informations pertinentes sous forme d'une liste ordonnée selon leur degré de pertinence décroissant.

Cependant en Recherche d'Information, l'intention réelle de l'utilisateur n'est pas toujours évidente dans sa manière de formuler sa requête et cela peut générer des ambiguïtés au niveau du sens des mots qu'elle contient. De nombreuses solutions existent pour préciser le sens d'une requête et on peut citer en particulier :

– les techniques d'expansion de requêtes via des *thésaurus* [XC96] ou des *ontologies* [Voo94] [BAGB03] ;
– les techniques de reformulation de requêtes dans des *processus de personnalisation de recherche* à travers une recherche personnalisée individuelle ou collaborative.

La *recherche individuelle personnalisée* va consister à : utiliser des jugements de pertinence (ou non-pertinence) d'un utilisateur sur un ensemble d'informations pour reformuler sa requête et affiner ainsi la recherche. C'est la méthode de réinjection de pertinence ou relevance feedback [Roc71] [BCSD99] [KV02] ; utiliser la notion de profil long terme des besoins (ou centres d'intérêt) de l'usager et la notion de profil court terme (ou contexte) de ses besoins (*cf.* section 1.3.1.2), pour aider à l'interprétation de ses requêtes afin de réévaluer et de réordonnancer les résultats d'une recherche [BBB03a] [BBB04] ; utiliser la notion de contextualisation et d'individualisation pour la personnalisation de la recherche [PSC+02] ; etc.

La *recherche collaborative* [KGB98] quant à elle va consister à utiliser la notion de groupe pour répondre aux besoins des utilisateurs. Ainsi, on va

pouvoir reformuler la requête d'un utilisateur avec les termes des documents validés par des utilisateurs de profils similaires au sien, lesquels documents ont été obtenus suite à des requêtes (ou situations de recherche) similaires [JRP01].

1.2.2 Filtrage d'Information

Alors que la Recherche d'Information (RI) est une tâche très interactive, celle du Filtrage d'Information (FI) est relativement passive [BC92] car l'utilisateur ne formule pas explicitement ses besoins au travers d'une requête (ou expression d'un besoin ponctuel) comme c'est le cas en RI. En Filtrage d'Information, on utilise plutôt une représentation de l'utilisateur appelé *profil utilisateur* pour lui envoyer des informations. Ces informations proviennent généralement d'un flux dynamique ou sont obtenues grâce à un agent. Elles sont ensuite comparées aux différents profils disponibles pour déterminer ceux auxquels elles correspondent. Il existe plusieurs méthodes de filtrage [MLR03] :

- *le filtrage cognitif ou basé sur le contenu* qui utilise la description du contenu des informations pour déterminer à quels profils utilisateurs elles correspondent [Lie95] [Mla96] [PMB96]. Le profil utilisateur, en filtrage cognitif, décrit les centres d'intérêt durables ou récurrents de l'individu qui sont représentés communément par une liste de mots-clés pondérés [Kor97]. Ce profil est obtenu manuellement ou automatiquement en indexant (*cf.* section 1.3.2.1), par exemple, les informations sauvegardées par l'utilisateur lors de ses sessions de recherche ;
- *le filtrage social ou collaboratif* qui utilise les jugements (ou feedback) d'un ensemble d'utilisateurs concernant un ensemble d'informations pour effectuer des recommandations. On utilise une mesure de similarité entre jugements d'individus pour déterminer si une information correspond à un individu donné [GNOT92] [KMM+97] [RP97]. La description du contenu réel des informations est ignorée. Le tableau Tab. 1.1 représente des exemples de profils de jugements utilisateurs en filtrage collaboratif. Le «+» signifie que l'information intéresse l'utilisateur, le «-» qu'elle ne l'intéresse pas et le «?» que l'information n'a pas encore été jugée par un utilisateur et pourrait donc être recommandée à ce dernier. Les utilisateurs *utilisateur1* et *utilisateur3* peuvent être considérés comme similaires car ils ont effectué les mêmes jugements. On peut donc recommander le document *document4* à l'utilisateur *utilisateur3* car l'utilisateur *utilisateur1* l'a déjà jugé comme étant intéressant ;
- *le filtrage démographique* qui utilise les données démographiques des utilisateurs (sexe, âge, profession, ville d'origine, etc.) pour les regrouper par groupes [Kru97] et leur faire des recommandations. Pour cela, on se base sur une catégorisation des informations en fonction des don-

Utilisateurs	Jugements de documents			
	document1	document2	document3	document4
utilisateur1	+	-	+	+
utilisateur2	-	-	-	?
utilisateur3	+	-	+	?

TAB. 1.1 – *Exemples de profils de jugements utilisateurs en filtrage collaboratif*

nées démographiques des individus. Cette catégorisation permet de déterminer quel type d'information est appréciée par un type d'utilisateur (relativement à leur données démographiques) particulier. Pour cela, on peut procéder à une catégorisation manuelle ou on peut se baser, par exemple, sur les jugements des utilisateurs pour déduire le type d'individu (groupe) auquel correspond une information [Paz99]. Le tableau TAB. 1.2 représente des exemples de profils utilisateurs en filtrage démographique. Le «+» signifie que l'information intéresse l'utilisateur, le «-» qu'elle ne l'intéresse pas et le «?» que l'information n'a pas encore été jugée par un utilisateur et pourrait donc être recommandée à ce dernier. On peut déduire du tableau TAB. 1.2, trois groupes de personnes du fait de la similarité de leurs jugements : les femmes de moins de 18 ans, les femmes de plus de 25 ans et les hommes de moins de 18 ans. On peut donc faire des recommandations relativement aux jugements effectués dans ces groupes.

Utilisateurs	Données démographiques		Jugements de documents		
	Sexe	Âge	document1	document2	document3
utilisateur1	F	15	+	-	+
utilisateur2	F	18	+	-	?
utilisateur3	F	25	-	-	+
utilisateur4	F	28	-	-	?
utilisateur5	M	16	+	-	+
utilisateur6	M	18	+	-	?
utilisateur7	M	25	-	-	-
utilisateur8	M	32	+	+	?

TAB. 1.2 – *Exemples de profils utilisateurs en filtrage démographique*

Ces approches ne sont pas exclusives et différentes méthodes hybrides, combinant ces différents types de filtrage, ont été développées [GSK+99] [Paz99]. L'utilisation des approches hybrides permet d'améliorer la pertinence des résultats des systèmes de filtrage en palliant certaines limites des

types de filtrage présentés précédemment [BS97] comme : la sur-spécialisation en filtrage basé sur le contenu ; l'obtention des jugements qui est une tâche coûteuse pour les utilisateurs, etc.

Comme exemples d'approches de filtrage hybride, on peut citer :

- *le filtrage collaboratif via le contenu* [Paz99] qui va permettre de déterminer des similarités entre utilisateurs via leur profil de besoins (centres d'intérêt), construit à partir du contenu des informations qu'ils ont jugées. Ainsi, pour identifier des groupes d'utilisateurs on ne se basera plus uniquement sur une mesure de similarité entre jugements utilisateurs. L'intérêt particulier de ce type de filtrage hybride est qu'il va permettre de faire des recommandations à un nouvel utilisateur, en l'affectant à un groupe via son profil des besoins. En filtrage collaboratif pur, il aurait fallut attendre que cet utilisateur ait effectué des jugements (sur des informations) et qu'avec ces jugements on puisse l'associer à d'autres utilisateurs pour pouvoir lui faire des recommandations. Cela nécessite en général un certain temps : c'est le problème de l'*entonoir* (ou boîte noire) qui se pose généralement pour le démarrage d'un filtrage collaboratif ;
- *les approches réclusives* [Yag02] qui sont basées sur la recherche d'une similarité entre objets en comparant leur description respective (ou contenu respectif). Ainsi, on pourra recommander une information si sa description est similaire à une autre information qui elle a déjà été validée (c'est-à-dire jugée intéressante) par l'utilisateur. L'intérêt de cette approche hybride est que l'on va pouvoir recommander une information qui n'a pas encore été jugée. En filtrage collaboratif pur, il faut attendre qu'une information soit jugée par au moins un utilisateur pour pouvoir la recommander ;
- etc.

Pour résumer, les différentes techniques d'accès à l'information partagent le même objectif qui est d'aider l'utilisateur à obtenir les informations qu'il recherche. Pour cela, on doit décrire les informations manipulées par les processus de recherche et de recommandation d'information. Cette description des informations est désignée sous le nom de profil (ou modèle ou représentation). L'appariement (ou mesure de similarité) entre ces profils va permettre de décider de la restitution ou non des informations aux usagers.

Le filtrage cognitif ou basé sur le contenu peut-être considéré comme le processus dual de la recherche d'information individuelle comme l'illustre la figure FIG. 1.1, décrivant les *modèles en U* de la RI et du FI. Cependant, quand on est dans un *contexte collaboratif* ou plusieurs utilisateurs concourent à la restitution d'un résultat donné, les appariements ou comparaisons de profils ne se font plus uniquement entre les informations mises à disposition et les besoins des usagers de ces informations mais aussi entre informations, entre usagers et jugements d'usagers. C'est le cas typique de la

recherche collaborative, du filtrage collaboratif, du filtrage démographique et des approches hybrides d'accès à l'information qui utilisent ces techniques.

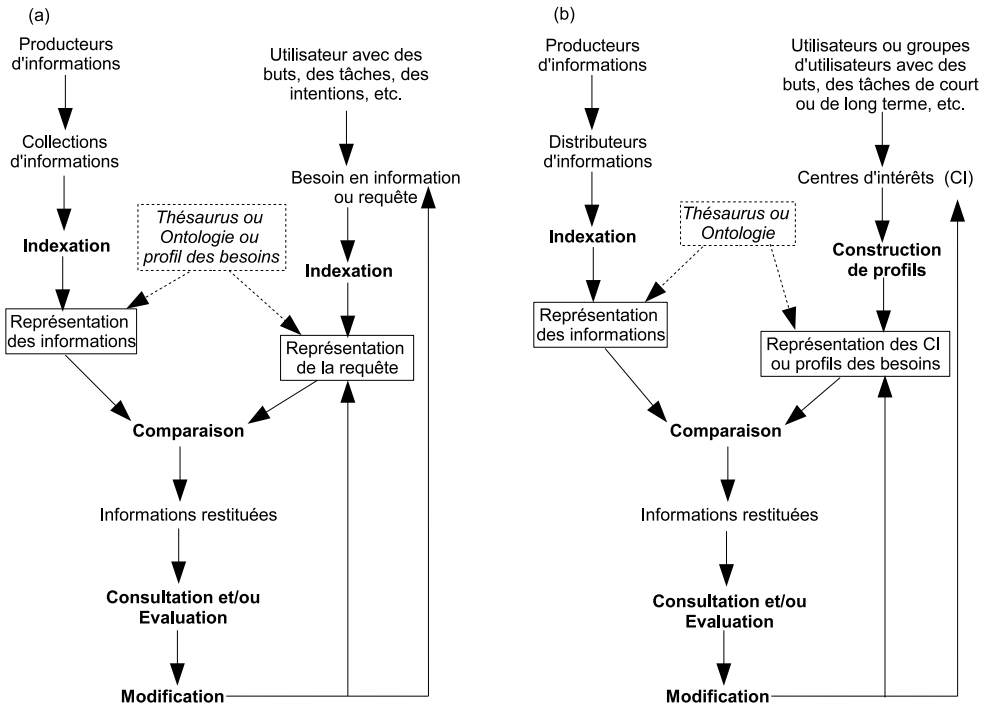


FIG. 1.1 – *(a) Modèle général en U de la Recherche d'Information individuelle (b) Modèle général en U du Filtrage d'Information basé sur le contenu*

Ainsi, on va distinguer, dans les techniques d'accès à l'information, différentes combinaisons de profils pour effectuer des appariements :

- représentation (ou profil) de la requête (reformulée ou pas) et représentation (ou profil) de l'information à restituer ;
- représentation d'une requête et représentation d'une autre requête ;
- profil des besoins d'un utilisateur et profil des besoins d'un autre utilisateur ;
- profil des besoins d'un utilisateur et profil d'une information mise à disposition ;
- profil d'une information et profil d'une autre information mise à disposition ;
- profil des jugements d'un utilisateur et profil des jugements d'un autre utilisateur ;
- profil des données démographiques d'un utilisateur et profil des données démographiques d'un autre utilisateur.

Le tableau TAB. 1.3 représente pour chaque technique d'accès à l'information, les combinaisons de profils possible pour les appariements.

Techniques d'accès	Combinaisons de profils pour les appariements
Recherche individuelle	1. représentation de requête (reformulée ou pas) et profils des informations mises à disposition.
Recherche collaborative	1. représentation de requête et profils des informations mises à disposition 2. représentation de requête d'un utilisateur et représentations de requêtes d'autres usagers ; 3. profil des besoins d'un utilisateur et profils des besoins d'autres usagers.
Filtrage cognitif	1. profil des besoins d'un utilisateur et profils des informations mises à disposition.
Filtrage collaboratif	1. profil des jugements d'un utilisateur et profils des jugements d'autres usagers.
Filtrage démographique	1. profil des données démographiques d'un utilisateur et profils des données démographiques d'autres usagers ; 2. profil des jugements d'un utilisateur et profils des jugements d'autres usagers.
Filtrage hybride	1. profil des besoins d'un utilisateur et profils des besoins d'autres usagers ; 2. profil des jugements d'un utilisateur et profils des jugements d'autres usagers ; 3. profil d'une information et profils d'autres informations mises à disposition.

TAB. 1.3 – *Techniques d'accès à l'information et combinaisons de profils associées pour les appariements*

Dans la section suivante, nous présentons avec plus de détails la notion de profil telle qu'elle est utilisée dans les différentes techniques d'accès à l'information.

1.3 Profils : types, méthodes de construction, modèles de représentation et appariements

De façon générale, le profil d'un objet est un ensemble de caractéristiques permettant de l'identifier ou de le représenter. Nous avons étudié les profils dans les techniques d'accès à l'information sous différents angles : types, méthodes de construction, modèles de représentation et appariements de profils.

1.3.1 Types de profils

Les profils utilisés dans les techniques d'accès à l'information sont de nature très variée et on peut les classifier en deux grands groupes :

- ceux relatifs aux informations mises à disposition ;
- ceux relatifs aux utilisateurs de ces informations.

1.3.1.1 Profils relatifs aux informations mises à disposition

Le profil des informations mises à disposition correspond à la description de ces dernières qui est souvent réduite, en RI ou FI, à une liste de mots-clés pondérés décrivant le contenu (sémantique) de ces informations. Plusieurs travaux permettent actuellement de décrire les informations en utilisant également d'autres critères que ceux liés à leur contenu effectif. On peut citer par exemple les métadonnées du Dublin Core [1], pour la description de ressources. Nous pouvons également citer les travaux de Lainé-Cruzel [LC99] qui permettent de définir des propriétés liées à l'ensemble d'un document (profession de l'auteur, type de document, etc.) ainsi que celles relatives à des parties de documents (type d'unité documentaire, forme discursive, style, etc.) afin de restreindre les documents pertinents (du point de vue du sujet dont ils traitent) aux seuls documents exploitables et réellement utilisables. De même, une liste non exhaustive de métadonnées pour l'annotation qualitative de documents est donnée par Berti-Equille [BE02] [BE03] dans le contexte de la recommandation multi-critères.

Notons que les informations à restituer par les processus de RI ou FI peuvent être de différents niveaux de granularité : *collections* de documents [GGMT99], *documents* et *granules ou parties* de document [INEX] [2], [TREC] [3]. De plus, les profils de ces informations peuvent être composés soit uniquement de mots-clés pondérés décrivant leur contenu, soit de mots-clés pondérés et de métadonnées (*cf.* Fig. 1.2).

1.3.1.2 Profils relatifs aux utilisateurs

Le profil utilisateur est une banque de données qui regroupe les différents sujets ou thèmes susceptibles d'intéresser un utilisateur donné [BMRM96]. Il peut également être vu comme une collection d'informations diverses sur l'utilisateur (*cf.* Fig. 1.3). Cette collection va permettre d'illustrer un ensemble de caractéristiques avec des valeurs associées [Mar02] contenant par exemple ce que l'utilisateur préfère, ce qu'il est capable de faire, l'historique de ses actions dans le temps, ses données démographiques, etc.

1. *cf.* http://dublincore.org/documents/dces/
2. *cf.* http://qmir.dcs.qmw.ac.uk/INEX/index.html
3. *cf.* http://icl.pku.edu.cn/icl_groups/iregroup/trec/trec.nist.gov/

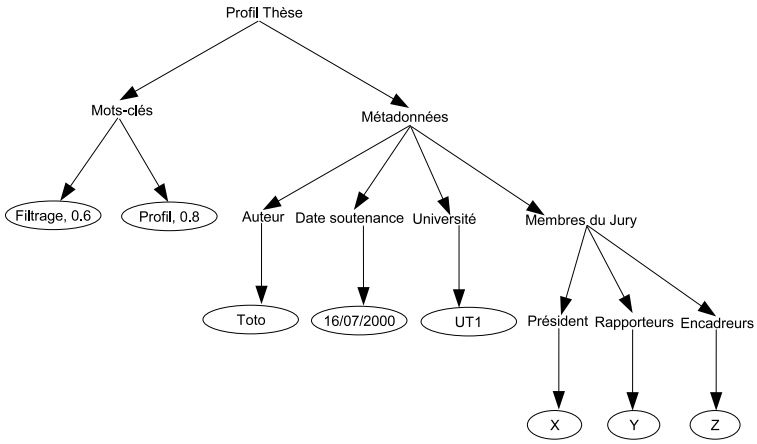


FIG. 1.2 – *Exemple de profil de thèse*

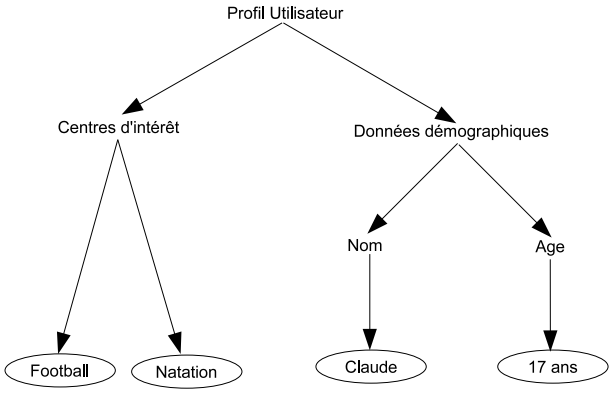


FIG. 1.3 – *Exemple de profil utilisateur*

Les utilisateurs peuvent être étudiés également selon différents niveaux de granularité [PSC+02] : *individu*, *groupe* d'utilisateurs ou *population* représentant tous les utilisateurs. Les profils utilisateurs peuvent donc décrire des individus ou des groupes d'individus. Ils peuvent également être de différents types, chacun décrivant une facette (ou vue) de l'utilisateur comme :

– *les profils de court terme ou de long terme* qui sont liés au temps d'apprentissage considéré pour l'obtention des informations du profil [WIY99] [MT02a] [MT02b]. Le profil court terme peut être, par exemple, le profil de l'utilisateur durant une session de recherche. Il peut être assimilé au contexte de recherche de ce dernier. Le profil long terme peut correspondre, quant à lui, au profil (description) de l'utilisateur construit sur plusieurs sessions de recherche. Ainsi, le profil court terme va permettre de préciser l'objectif à court terme d'un

utilisateur tandis que le profil long terme permet de préciser l'objectif à priori de l'utilisateur indépendamment de sa session de recherche. Le profil court terme est très important car il permet de se rendre compte des changements de centres d'intérêt ou de préférences d'un utilisateur pour mieux s'adapter à celui-ci. En résumé, les profils long terme sont obtenus après un temps d'apprentissage important contrairement aux profils court terme ;

– *les profils positif ou négatif* qui permettent de préciser ce que l'utilisateur aime et ce qu'il n'aime pas [HKNH00]. La notion de profil négatif est né du fait que la plupart des systèmes de filtrage d'information emploient des valeurs de seuil assez élevées pour éviter de sélectionner des documents non pertinents. Cette approche engendre dans certains cas, la non-sélection de documents pertinents dont la valeur de similarité avec le profil est inférieur au seuil. Pour résoudre ce problème, Hoashi [HKNH00] introduit la notion de profil négatif. Pour cela, il effectue d'abord un premier filtrage avec le profil positif de l'utilisateur et par la suite, un second filtrage avec le profil négatif de l'utilisateur.

En résumé, la structure d'un profil quelconque, en RI ou FI, peut être composée :

– *d'un seul critère* qui est lié au contenu des informations à savoir : *mots-clés* pour les profils d'informations et *centres d'intérêt* pour les profils utilisateurs. Pour ce dernier, on parle généralement de *profils des besoins utilisateurs* ;
– *de plusieurs critères.* Dans ce cas, on a un profil étendu via des métadonnées par exemple (*cf.* FIG. 1.2 et FIG. 1.3).

La typologie structurelle d'un profil peut donc être mono-critère [PMB96] [Amm03] ou multi-critères [LC99] [BE02].

Par ailleurs, selon ce que les profils représentent au niveau sémantique, on va distinguer également dans la littérature différents types de profils. Les tableaux TAB. 1.4 et TAB. 1.5 représentent des typologies sémantiques de profils.

Types de profils des informations mises à disposition	Sémantique
profil du contenu de l'information	description du contenu de l'information (mots-clés ou termes d'indexation)
profil étendu de l'information	description du contenu de l'information augmenter de métadonnées : langue, taille, ...

TAB. 1.4 – *Typologie sémantique de profils d'informations mises à disposition*

Notons que les différents *types sémantiques de profils d'informations mises à disposition*, du tableau TAB. 1.4, peuvent être utilisés pour décrire soit des collections de documents (collections CLEF 2002, etc.), soit des documents, soit des granules ou parties de documents (paragraphes, sections, etc.).

Types de profils utilisateurs	Sémantique
profil des besoins	centres d'intérêt
profil des jugements	types de jugements : pertinence, non-pertinence, ...
profil étendu	combinaison de différents critères de description : centres d'intérêts, jugements, ...
profil court terme	profil construit sur une période courte : deux heures, une session de recherche, ...
profil long terme	profil construit sur une période assez longue : plusieurs sessions de recherche, ...
profil positif	profil représentant ce qui est recherché
profil négatif	profil représentant ce qui n'est pas recherché

TAB. 1.5 – *Typologie sémantique de profils utilisateurs*

De même, les différents *types sémantiques de profils utilisateurs*, du tableau TAB. 1.5, peuvent être utilisés pour décrire soit un individu, soit un groupe d'individus.

Dans la section suivante, nous présentons les principales méthodes utilisées pour construire des profils en RI et FI.

1.3.2 Méthodes de construction de profils

En général, on distingue deux groupes de méthodes de construction de profils :

- *les méthodes manuelles* où l'humain renseigne lui-même les valeurs de certains critères descriptifs de profils ;
- *les méthodes automatiques ou semi-automatiques* de construction (ou d'apprentissage) de profils comme : l'indexation automatique, le profiling, les approches par stéréotypes, etc.

1.3.2.1 Indexation

Les mots-clés *significatifs* décrivant le contenu d'une information et leurs poids sont obtenus en général par une opération d'indexation [Rij79] [SM83].

L'indexation va permettre de déterminer les mots-clés décrivant le contenu des informations. Elle est applicable à différents niveaux de granularité des informations (document, granules de document, collections de documents). L'indexation est également utilisé pour déduire les centres d'intérêts (ou mots-clés décrivant leurs besoins) des utilisateurs à partir des informations d'usage (sites visités, informations jugées ou sauvegardées, etc.) de ces derniers durant des sessions de recherche.

Les différentes étapes généralement suivies pour l'indexation sont :

1. *le nettoyage* qui consiste à supprimer tout ce qui peut avoir des répercussions inattendues sur les traitements qui vont suivre. Par exemple, dans le cas de documents textuels on devra supprimer : les images, les formules mathématiques, etc. ;

2. *la segmentation* qui consiste à découper le texte en unités lexicales ou *termes* (mots simples ou groupes de mots) ;

3. *la suppression des mots outils* qui consiste à éliminer les mots qui n'ont pas de sens propre comme les articles, les adverbes, etc. Pour cela, on utilise des listes pré-définies de ces mots outils ou mots vides [4] ;

4. *l'analyse morphologique des termes* qui recouvre deux notions à savoir [Dai98] [5] :

- *la morphologie flexionnelle* où la forme du mot change du fait des accords grammaticaux ou de la conjugaison (genre, nombre, personne, temps, mode). Ce type d'analyse morphologique, encore appelé *lemmatisation*, permet de déterminer la forme neutre d'un mot ou *lemme*. Ainsi, le lemme du mot *amies* est : *ami*. Comme exemple de lemmatiseur on peut citer FLEMM [6] pour le français et pour l'anglais on peut utiliser l'étiqueteur morpho-syntaxique de BRILL [7] ;
- *la morphologie dérivationnelle* qui est liée à des règles de suffixation et de préfixation. Ainsi, du nom *volcan*, on pourra dériver l'adjectif *volcanique*.

La *radicalisation* est l'analyse morphologique qui va permettre de déterminer le *radical* ou *racine* d'un mot en effectuant des analyses morphologiques et en éliminant les *flexions* ou *dérivations* qu'elles engendrent. Ainsi, le mot *volcaniques* aura pour lemme *volcanique* et pour radical *volcan*. Pour déterminer les racines des mots en anglais on utilise généralement l'algorithme de Porter [Por80]. Des versions françaises de cet

4. *cf.* http://bll.epnet.com/help/ehost/Stop_Words.htm
5. *cf.* http://www.sciences.univ-nantes.fr/info/perso/permanents/daille/
6. *cf.* http://www.univ-nancy2.fr/pers/namer/Telecharger_Flemm.htm
7. *cf.* http://www.cs.jhu.edu/ brill/

algorithme ont été également proposées [PFL+02]. Notons qu'il existe, dans les pré-traitements d'information textuelles, d'autres types d'analyses à savoir : l'*analyse syntaxique* qui cherche à structurer la phrase en dépendances syntaxiques du genre *«sujet verbe complément»* ; l'*analyse sémantique* qui consiste à enrichir le texte avec d'autres informations comme par exemple la classe sémantique de chaque mot qui peut-être : objet, humain, etc. ;

5. *la pondération et la sélection des termes significatifs* : les termes significatifs sont des termes de fréquence d'apparition intermédiare. Ils sont obtenus en appliquant la loi de Luhn [Luh58] et de Zipf [Zip49] (*cf.* FIG. 1.4) :

- selon Zipf, si les termes sont rangés dans l'ordre décroissant de leur fréquence d'apparition, le produit de cette fréquence par le rang est quasiment constant : $Frequence.Rang \simeq Constante$;
- selon Luhn, les termes peuvent être regroupés en trois classes, en fonction de leur fréquence d'apparition, par la définition d'un seuil minimal S_m et d'un seuil maximal S_M de signification. Ces seuils permettent de caractériser les termes très fréquents et les termes très rares.

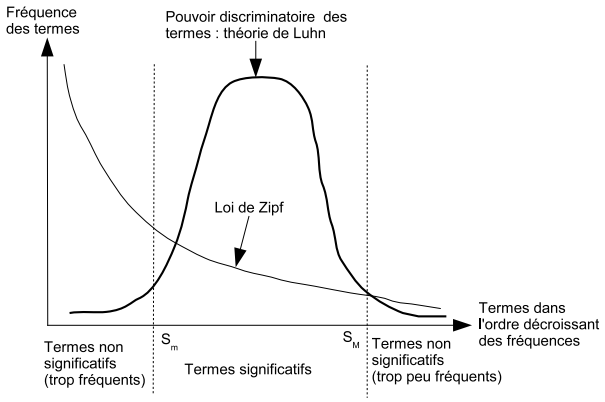


FIG. 1.4 – *Pouvoir discriminatoire des termes d'indexation*

En déduction de la loi de Zipf et de la théorie de Luhn, un terme d'indexation (radicaliser ou pas) représente le contenu des documents dans lequel il apparaît (*pouvoir sémantique*) et en même temps il distingue le contenu d'un document du reste de la collection (*pouvoir discriminatoire*), d'où les deux fréquences :

- *fréquence absolue* (notée df_i) ou fréquence d'apparition d'un terme t_i dans une collection de documents ;
- *fréquence relative* (notée tf_{ij}) ou fréquence d'apparition d'un terme t_i dans un document d_j.

Le poids sémantique (noté w_{t_i,d_j}) d'un terme t_i dans un document d_j est proportionnel à sa fréquence relative tf_{ij} et inversement proportionnel à sa fréquence absolue df_i. L'inverse de df_i est notée idf_i.

En conséquence, la fréquence relative tf_{ij}, l'inverse de la fréquence absolue idf_i et le poids sémantique w_{t_i,d_j} d'un terme peuvent être calculés comme suit :

$$tf_{ij} = \frac{f(t_i,d_j)}{Max[f(t,d_j)]} \tag{1.1}$$

où :

$f(t_i,d_j)$ est la fréquence d'apparition du terme t_i dans le document d_j ;
$Max[f(t,d_j)]$ est la fréquence maximale des termes dans le document d_j ;
$tf \in [0,1]$.

$$idf_i = \log(N/n) + 1 \tag{1.2}$$

où :

N est le nombre de documents de la collection ;
n est le nombre de documents contenant t_i ;
$idf \in \Re_+^*$.

$$w_{ti,d_j} = tf_{ij} \cdot idf_i \tag{1.3}$$

D'autres formules pour calculer w_{t_i,d_j} ont également été proposées afin d'améliorer la formule de base (*cf.* formule 1.3) de K. Sparck Jones [Jon72]. Ces formules sont explicitées dans de nombreux ouvrages [FY92] [BYRN99] [SD01]. Parmi les variantes du calcul du poids sémantique (w_{t_i,d_j}) d'un terme, on peut citer :

– la formule de *Salton et Buckley 90* [SB90] :

$$w_{t_i,d_j} = \frac{\log(tf_{ij}) + 1}{\sqrt{\sum_{i=1}^{N}(\log(tf_{ij}) + 1)}} \tag{1.4}$$

où :

tf_{ij} est la fréquence d'apparition du terme t_i dans le document d_j ;
N est le nombre total de documents dans la collection.
Notons que $\log(tf_{ij}) + 1$ permet de réduire l'impact des termes trop fréquents lors du processus de comparaison des requêtes et des documents.

– la formule de *Singhal et al. 97* [SMB97] :

$$w_{t_i,d_j} = \frac{\log(tf_{ij}) + 1}{0{,}7 + 0{,}3 \times \frac{l_j}{L_M}} \tag{1.5}$$

où :

tf_{ij} est la fréquence d'apparition du terme t_i dans le document d_j ;

l_j est la longueur du document d_j ;

L_M est la longueur moyenne des documents de la collection.

– la formule *Okapi-BM25* de *Robertson et al. 99* [RWB99]

$$w_{t_i,d_j} = \sum_{T \in Q} w^{(1)} \cdot \frac{(k_1+1)tf_{ij}}{K + tf_{ij}} \cdot \frac{(k_3+1)qtf_i}{k_3 + qtf_i} + k_2 \cdot |Q| \cdot \frac{avdl - dl_j}{avdl + dl_j} \quad (1.6)$$

où :

Q est une requête ;

T est l'ensemble des termes de la requête Q ;

$w^{(1)}$ est soit la formule, ci-dessous, de poids de *Robertson et Sparck Jones 76* [RJ76] des termes T de la requête Q

$$\log \frac{(r+0{,}5)/(R-r+0{,}5)}{(n-r+0{,}5)/(N-n-R+r+0{,}5)} \quad (1.7)$$

soit une version plus générale (*cf.* formule ci-dessous) qui prend en compte aussi bien la pertinence que la non-pertinence des informations [RW97]

$$\frac{k_5}{k_5+\sqrt{R}}(k_4 + \log\frac{N}{N-n}) + \frac{\sqrt{R}}{k_5+\sqrt{R}} \log \frac{r+0{,}5}{R-r+0{,}5}$$

$$-\frac{k_6}{k_6+\sqrt{S}} \log \frac{n}{N-n} - \frac{\sqrt{S}}{k_6+\sqrt{S}} \log \frac{s+0{,}5}{S-s+0{,}5} \quad (1.8)$$

N est le nombre de documents de la collection ;

n est le nombre de documents contenant le terme t_i ;

R est le nombre de documents connus comme étant pertinents pour une requête spécifique ;

r est le nombre de documents pertinents contenant le terme t_i ;

S est le nombre de documents connus comme étant non-pertinents pour une requête spécifique ;

s est le nombre de documents non-pertinents contenant le terme t_i ;

tf_{ij} est la fréquence d'apparition du terme t_i dans le document d_j ;

qtf_{ij} est la fréquence d'apparition du terme t_i dans la requête Q ;

dl_j est la longueur du document d_j ;

$avdl$ est la longueur moyenne des documents de la collection ;

$K = k_1((1-b) + b.dl_j)/avdl$;

k_1, k_2, k_3, b, k_4, k_5 et k_6 sont des paramètres dépendant de la nature des requêtes et éventuellement de la collection. Les valeurs de ces paramètres ont été obtenues par des séries d'expérimentations sur des collections de tests.

– la formule de *Boughanem et al. 00* utilisé dans le système connexionniste *Mercure* [BJMSD00]. Cette formule a été inspirée des travaux de Robertson sur le projet Okapi [RWB99]. Sa formule est la suivante :

$$w_{t_i,d_j} = \frac{(\log(tf_{ij}) + 1) \cdot (h_1 + h_2 \cdot \log(\frac{M}{n_i}))}{h_3 + h_4 \cdot \frac{dl_j}{\Delta}} \tag{1.9}$$

où :

tf_{ij} est la fréquence d'apparition du terme t_i dans le document d_j ;

M est le nombre de documents dans la collection ;

n_i est le nombre de documents contenant le terme t_i ;

dl_j est le nombre de termes d'indexation du document d_j ;

Δ est le nombre moyen de termes dans un document (longueur moyenne des documents de la collection).

Notons que les formules qui prennent en compte la longueur des documents [SMB97] [RWB99] [BJMSD00] essayent de compenser les grandes fréquences de termes dans le cas de documents longs conduisant généralement à la restitution des documents les plus longs au détriment des documents les plus courts. Ces mesures sont de ce fait bien adaptées à des collections de documents hétérogènes en volume ou longueur (nombre de termes, etc.).

– la formule de *Crampes 80* pour la pondération de textes courts. La pondération des termes extraits de textes courts de type *titre* ou *résumé* est un cas particulier. La fréquence relative dans ce contexte est généralement binaire. Des formules spécifiques ont alors été proposées dont celle de *Crampes 80* [Cra80], définie comme suit :

$$w_{t_i,d_j} = (1 - \frac{n_i}{N})^{\sqrt{N}} \tag{1.10}$$

où :

N est le nombre de documents (textes courts) dans la collection ;

n_i est le nombre de documents (textes courts) contenant le terme t_i.

1.3.2.2 Clustering et Approches par stéréotypes

Le *clustering* ou *classification* [LC96] [Dun00] identifie et classe les objets sur la base de la similarité des caractéristiques qu'ils possèdent. Le clustering cherche à minimiser la variance à l'intérieur d'un groupe et à maximiser la

variance entre les groupes. Le résultat du clustering devrait être un nombre de groupes hétérogènes avec des contenus homogènes.

Ainsi, avec des profils individuels d'utilisateurs on peut créer des groupes d'utilisateurs par classification en regroupant, par exemple, des utilisateurs de centres d'intérêts communs. De même, pour les informations mises à disposition, on va pouvoir les regrouper par classes sur la base, par exemple, de la similarité entre leur contenu.

L'*approche par stéréotype* [SSH97] est une approche particulière de clustering qui est fondée sur l'identification (généralement manuelle) de groupes appelés stéréotypes et sur la détermination des caractéristiques clés de chaque groupe. Les groupes et les caractéristiques de chaque groupe sont pré-définis et les différents objets (utilisateurs ou informations) à classer sont affectés à ces groupes en fonction du degré de ressemblance de leur profil individuel aux différents stéréotypes.

1.3.2.3 Apprentissage de profils utilisateurs par profiling

L'apprentissage de profils utilisateurs peut se faire en «scrutant» les habitudes des utilisateurs et en analysant leurs réactions vis à vis des documents qui leur sont présentés : c'est du profiling. Le système détermine le besoin que l'utilisateur semble exprimer en notant, par exemple, la fréquence d'apparition de certains termes dans les requêtes ou documents visités.

Le profiling [CKK02] [BBB03b] consiste donc à scruter, enregistrer et analyser les actions et successions d'actions d'un utilisateur durant des sessions de recherche pour déterminer son profil. Pour ce faire, un sous-système de modélisation ou un *agent* observe l'utilisateur au travers d'une interface et apprend le profil de ce dernier à partir de ses actions (visite d'un site Internet de manière récurrente, achat d'un produit en ligne, sauvegarde de documents, etc.).

Il s'agit ici de ce que l'on appelle *profil implicite*. Il est souvent difficile pour un utilisateur d'exprimer clairement ses besoins, mais il lui est plus facile d'identifier ses besoins en partant des documents susceptibles d'y répondre. On peut donc utiliser le jugement d'un utilisateur sur des documents qui lui sont proposés (résultats d'une recherche par exemple) pour déterminer ses besoins en informations.

Le principe du profiling est d'enregistrer et d'analyser les comportements et actions des visiteurs pour déduire par apprentissage leurs besoins. Pour cela, il est possible d'utiliser les différentes informations que l'utilisateur examine pour essayer d'en extraire ses thèmes de recherche. On peut, pour cela, se baser sur :

- l'usage fait d'une information : lue ou ignorée ;
- le temps d'étude d'une information : une information non pertinente ne devrait pas être étudiée longtemps par l'utilisateur, au contraire des informations pertinentes sur lesquelles l'utilisateur va s'attarder ;

– les différentes commandes exécutées à la suite de l'étude d'une information : si l'utilisateur sauvegarde, par exemple, l'information sur son ordinateur ou répond à ses mails on peut alors supposer que l'information ou le mail en question répond à ses besoins. Dans le même ordre d'idées, certains systèmes comptabilisent le nombre de «clicks» effectués par l'utilisateur lors de la consultation d'une information pour déduire l'intérêt que l'usager porte à cette information ;
– les annotations sur des informations : jugements, etc.

Le profiling permet également d'opérer des regroupements entre des profils similaires et de produire des offres personnalisées aux membres de chaque groupe.

En résumé, il existe aujourd'hui une multitude d'approches qui permettent la construction automatique de profils. La plupart d'entre elles réalisent un apprentissage basé sur des techniques neuro-mimétiques [CP43], sur des algorithmes génétiques [Hol75], sur des processus de classification [Mit97], etc. Cependant, toutes ces méthodes ne sont pas forcément exclusives et peuvent être combinées pour obtenir un résultat optimal [Cal98] [BCSD99] [BHM01].

Dans la section suivante, nous présentons les principaux modèles de représentation de profils en RI et FI et nous expliquons comment ces modèles sont utilisés pour apparier des profils ou mesurer la similarité (respectivement dissimilarité) entre profils.

1.3.3 Modèles de représentation et appariements de profils

Il existe plusieurs modèles de représentation de profils en RI/FI et chacun d'entre eux permet de définir un principe d'appariement de profils pour mesurer leur ressemblance. Parmi ces modèles, les plus utilisés sont : le modèle booléen, le modèle vectoriel et le modèle probabiliste. Nous illustrons ces différents modèles de représentation à travers un *modèle de document* et un *modèle de requête* mais cela peut-être généralisé à tout type de profils.

1.3.3.1 Modèle booléen et booléen étendu

Le modèle booléen est basé sur la présence ou l'absence des termes de la requête dans les représentations des documents (termes d'indexation associés). Il permet d'effectuer une recherche stricte réalisée à partir d'une comparaison exacte entre le besoin en information, décrit dans la requête à l'aide d'opérateurs logiques (ET, OU, NON ou SAUF) et les termes représentant les documents d'une collection. Par exemple, soit la requête booléenne suivante : *«initiation ET informatique ET (algorithmique OU programmation) ET NON gestion»*, les documents restitués devront comporter :

– les trois termes *initiation*, *informatique* et *algorithmique* mais pas le terme *gestion* ;

– les trois termes *initiation*, *informatique* et *programmation* mais pas le terme *gestion* ;
– les quatre termes *initiation*, *informatique*, *algorithmique* et *programmation* mais toujours pas le terme *gestion* ;

Soit Q l'ensemble des requêtes booléennes, D l'ensemble des documents d'une collection, T l'ensemble des termes d'indexation et F la fonction de similitude telle que :

$$F : D \times Q \rightarrow \{0,1\}$$

$$(d,q) \rightarrow \{0,1\}$$

La ressemblance $F(d_k,q)$ entre un document d_k et une requête q dépend de l'équation de cette requête qui peut avoir les formes basiques suivantes :

$$F(d_k,t_i) = \begin{cases} 1 & si\ t_i \in d_k \\ 0 & sinon \end{cases}$$

$$F(d_k,t_i ET t_j) = MIN(F(d_k,t_i),F(d_k,t_j)) = F(d_k,t_i) * F(d_k,t_j)$$

$$F(d_k,t_i OU t_j) = MAX(F(d_k,t_i),F(d_k,t_j))$$
$$= F(d_k,t_i) + F(d_k,t_j) - F(d_k,t_i) * F(d_k,t_j)$$

$$F(d_k,NON t_i) = 1 - F(d_k,t_i)$$

Le modèle booléen a l'avantage d'être simple à mettre en œuvre mais engendre plusieurs inconvénients, notamment le fait de ne pas pouvoir ordonner les documents restitués. De plus, l'écriture de la requête peut-être complexe et une mauvaise formulation peut engendrer des résultats erronés.

Une extension du modèle booléen a été introduite pour essayer de pallier le premier inconvénient en s'appuyant sur la théorie des ensembles flous proposée par Zadeh. Dans ce modèle, la fonction de similitude retourne une valeur dans un intervalle [0,1] :

$$F : D \times Q \rightarrow [0,1]$$

$$(d,q) \rightarrow [0,1]$$

Le modèle *booléen étendu* prend en compte, en plus de l'absence ou de la présence des termes de la requête dans le document, les poids sémantiques de chacun de ces termes. L'inconvénient majeur de cette méthode de recherche est que l'évaluation peut proposer des valeurs différentes pour des expressions équivalentes.

1.3.3.2 Modèle vectoriel

Le modèle vectoriel préconise une représentation commune des unités d'informations et de la requête dans un même espace vectoriel engendré par les termes d'indexation [SM83]. Ainsi, si on pose $\overrightarrow{q_i}$ le vecteur représentant la requête q_i et $\overrightarrow{d_j}$ le vecteur représentant le document d_j :

$$\overrightarrow{q_i} = (w_{t_1,q_i}, w_{t_2,q_i}, w_{t_3,q_i}, \ldots, w_{t_n,q_i})$$

$$\overrightarrow{d_j} = (w_{t_1,d_j}, w_{t_2,d_j}, w_{t_3,d_j}, \ldots, w_{t_n,d_j})$$

où :

n est le nombre de termes d'indexation de la base ;

w_{t_n,q_i} est le poids du terme t_n dans la requête q_i ;

w_{t_n,d_j} est le poids du terme t_n dans le document d_j.

La figure FIG. 1.5 illustre, par exemple, la représentation vectorielle d'un document dans un espace à deux dimensions.

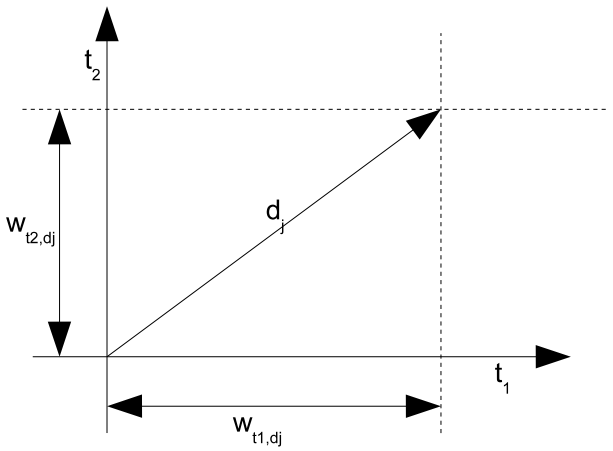


FIG. 1.5 – *Représentation vectorielle d'un document*

L'hypothèse de l'indépendance des termes (et donc de l'orthogonalité des vecteurs termes) est généralement faîte. Cette hypothèse permet de simplifier toutes les opérations de représentation et de recherche d'information à partir des termes d'indexation : calcul de poids, calcul de similarité, etc.

La majorité des modèles de recherche utilisés en informatique documentaire sont basés sur cette représentation.

Afin de pouvoir apparier ou comparer les vecteurs de requête et les vecteurs de documents, des mesures de similitude ont été élaborées : *produit scalaire*, *mesure du cosinus*, *distance métrique*.

Le *produit scalaire* est la fonction mathématique classique utilisant les coordonnées des vecteurs (poids sémantique des termes). La mesure de similarité entre le vecteur $\overrightarrow{q_i}$ et le vecteur $\overrightarrow{d_j}$, notée $sim(q_i,d_j)$ correspond au produit des vecteurs :

$$sim(q_i,d_j) = \sum_{k=1}^{n} w_{t_k,q_i} \cdot w_{t_k,d_j} \tag{1.11}$$

où :

n est le nombre de termes d'indexation de la base ;

w_{t_k,q_i} est le poids du terme t_k dans la requête q_i ;

w_{t_k,d_j} est le poids du terme t_k dans le document d_j.

La *mesure du cosinus* est la plus répandue dans ce type de modèle pour évaluer la ressemblance des documents et de la requête. Elle mesure l'angle entre les vecteurs. Elle est équivalente au produit scalaire des vecteurs normalisés. La mesure du cosinus est donnée comme suit [SM83] :

$$sim(q_i,d_j) = \frac{\sum_{k=1}^{n} w_{t_k,q_i} \cdot w_{t_k,d_j}}{(\sum_{k=1}^{n} w_{t_k,q_i}^2)^{1/2} (\sum_{k=1}^{n} w_{t_k,d_j}^2)^{1/2}} \tag{1.12}$$

où les poids des termes de la requête w_{t_k,q_i} et du document w_{t_k,d_j} sont ceux définis précédemment (*cf.* équation 1.11).

La pertinence d'une information par rapport à une requête est alors rattachée à la mesure de similarité des vecteurs correspondants. Le cosinus de deux vecteurs appartient toujours à l'intervalle [0,1] et plus les vecteurs sont proches, plus leur similarité (valeur du cosinus de l'angle entre les vecteurs) est grande. Ainsi, *si* $sim(q_i,d_j) = 1$ *alors* $\overrightarrow{q_i} = \overrightarrow{d_j}$. A chaque information ou document, il va correspondre une mesure de similarité ou degré de ressemblance qui va permettre de sélectionner les informations à restituer selon un seuil donné. Les informations sélectionnées peuvent ensuite être classées selon un degré de pertinence décroissant.

Un autre type de mesure de similarité utilisée dans le modèle vectoriel est la *distance métrique* entre les vecteurs à apparier :

$$dist(q_i,d_j) = ||q_i,d_j|| = \Big(\sum_{k=1}^{n} |w_{t_k,q_i} - w_{t_k,d_j}| \Big)^{1/p} \quad avec \quad p \geq 1 \tag{1.13}$$

Dans ce cas, plus la distance est grande, plus les vecteurs sont différents. Ainsi, *si* $dist(q_i,d_j) = 0$ *alors* $q_i = d_j$.

L'inconvénient majeur du modèle vectoriel est qu'il ne permet pas de modéliser les associations entre termes d'indexation : chaque terme est considéré comme indépendant des autres.

1.3.3.3 Modèle probabiliste

Dans le modèle probabiliste, on suppose que lorsque les représentations de la requête et d'un document sont suffisamment similaires, la probabilité correspondante de pertinence est suffisante pour restituer le document en réponse à la requête.

Pour faire intervenir le processus d'indexation, deux probabilités conditionnelles sont utilisées :

- $P(t_i/Pert)$: probabilité que le terme t_i apparaisse dans un document donné sachant que ce document est pertinent pour la requête ;
- $P(t_i/NonPert)$: probabilité que le terme t_i apparaisse dans un document donné sachant que ce document n'est pas pertinent pour la requête.

En utilisant une formule établie par Bayes (probabilités conditionnelles) et en supposant l'indépendance des variables *document pertinent* et *document non pertinent*, la fonction de recherche peut-être obtenue en calculant la probabilité de pertinence $P(Pert/d_j)$ d'un document d_j donné. Soit le document $d_j = (t_1,t_2,t_3,\ldots,t_n)$ où :

$t_i = 1$ si le terme t_i indexe le document d_j ;

$t_i = 0$ sinon.

La probabilité de pertinence d'un document $P(Pert/d_j)$ sachant sa description et la probabilité de non pertinence $P(NonPert/d_j)$ de ce document se calculent comme suit :

$$P(Pert/d_j) = \frac{P(d_j/Pert) \cdot P(Pert)}{P(d_j)} \tag{1.14}$$

$$P(NonPert/d_j) = \frac{P(d_j/NonPert) \cdot P(NonPert)}{P(d_j)} \tag{1.15}$$

Pour les équations 1.14 et 1.15 on a :

$P(d_j) = P(d_j/Pert).P(Pert) + P(d_j/NonPert).P(NonPert)$;

$P(d_j/pert)$ (respectivement $P(d_j/pert)$) est la probabilité de restituer le document d_j sachant qu'il est pertinent (respectivement non pertinent) ;

$P(Pert)$ (respectivement $P(NonPert)$) est la probabilité à priori pour qu'un document soit pertinent (respectivement non pertinent).

Si on considère l'indépendance des termes :

$$P(d_j/Pert) = \prod_{i=1}^{n} P(t_i/Pert) \tag{1.16}$$

$$P(d_j/NonPert) = \prod_{i=1}^{n} P(t_i/NonPert) \tag{1.17}$$

Pour les équations 1.16 et 1.17 on a :

$P(t_i/Pert) = \frac{r_i}{R}$;

$P(t_i/NonPert) = \frac{n_i - r_i}{N - R}$;

r_i est le nombre de documents pertinents dans lesquels le terme t_i apparaît ;

R est le nombre de documents pertinents pour la requête ;

n_i est le nombre de documents non pertinents dans lesquels le terme t_i apparaît ;

N est le nombre de documents non pertinents pour la requête (dans la collection).

Pour caractériser l'occurrence des termes d'indexation dans les documents, on utilise un loi de distribution (comme la loi de poisson). Cette occurrence est déduite d'un échantillon de documents.

Pour la restitution, les documents sont classés en fonction de $P(Pert/D)$. Le principe d'ordonnancement probabiliste stipule que cet ordonnancement est optimal en ce sens que, quel que soit le pourcentage de documents pertinents qui sont restitués, le pourcentage de documents restitués qui sont effectivement pertinents est maximisé.

1.3.3.4 Autres modèles de représentation

Les modèles de représentation booléen, vectoriel et probabiliste sont les plus utilisés dans les techniques d'accès à l'information. Notons cependant qu'il existe également d'autres modèles de représentation parmi lesquels nous pouvons citer :

- *le Latent Semantic Analysis ou Indexing (LSA ou LSI)* qui est une variante du modèle vectoriel. Il convertit un exemple représentatif de documents en une matrice de terme-par-document dans laquelle chaque cellule indique la fréquence avec laquelle chaque terme (lignes) apparaît dans chaque document (colonnes). Ainsi, un document devient un vecteur colonne et peut être comparé à une requête utilisateur représentée comme un vecteur de même dimension. Le LSA ou LSI étend le modèle vectoriel en modélisant les relations terme-document via une approximation réduite de l'espace ligne et colonne. Cette approximation est calculée par une *décomposition en valeurs singulières* ou *SVD pour Singular Value Decomposition* de la matrice de terme-par-document [DDF90] [Dum94]. Dans l'approche LSI, on va essayer de tenir compte du contenu sémantique (mots-clés) et conceptuel des documents (relations terme-document déduite par *SVD*) ;
- *les réseaux inférentiels Bayésiens* qui permettent de prendre en compte la dépendance entre les termes contrairement au modèle vectoriel [TC91] ;
- *les approches connexionnistes* [CP43], [BJMSD00] ;

– *les graphes conceptuels* [Sow84] ;
– *les algorithmes génétiques* [Hol75].

Dans la section suivante, nous présentons des définitions des termes *personnalisation* et *adaptation* afin de préciser l'aspect de la personnalisation que nous allons aborder dans nos travaux de recherche. En effet, ces deux concepts sont assez souvent liés dans des applications.

1.4 Personnalisation et adaptation des processus

De façon générale, personnaliser l'information signifie s'adapter aux buts, préférences et capacités de chaque utilisateur :

– *l'adaptation aux buts* consiste à prendre en considération le but que cherche à atteindre l'utilisateur, l'objectif de sa recherche. Cette prise en compte du but de l'utilisateur (et par conséquent, l'adaptation à la tâche de l'utilisateur) est une des préoccupations fondamentales en personnalisation ;
– *l'adaptation aux préférences* de l'utilisateur existe de façon répandue dans les interfaces adaptables et paramétrables par l'utilisateur, grâce à un ensemble d'options de menus de type «personnaliser». Un exemple d'une forme plus «automatisée» d'adaptation est donnée par la configuration d'un environnement graphique particulier et l'exécution de certaines applications utiles à l'utilisateur, lors de sa connexion à un système en s'identifiant ;
– *l'adaptation aux capacités* de l'utilisateur consiste à lui délivrer de l'information selon une forme et dans des délais acceptables (utilisables) par lui. Par exemple, cela peut consister à limiter l'attente de l'utilisateur avant de récupérer une information. Cela peut se concrétiser également par l'adaptation aux moyens matériels et/ou logiciels dont disposent l'utilisateur pour récupérer ou visualiser l'information fournie (exemple : taille d'écran, débit du réseau, etc.)

Aujourd'hui avec Internet, la personnalisation est devenue une approche incontournable. En effet, ce concept est très utilisé dans le domaine du multimédia interactif comme les applications de publications électroniques du Web [CB02]. La personnalisation s'est imposée car elle facilite la gestion de la richesse et de la complexité croissantes des contenus et des usagers de ces contenus.

Il existe différentes techniques de personnalisation que l'on peut regrouper en deux groupes :

– les plus simples qui permettent à l'utilisateur de personnaliser, par exemple, une page Web ou l'interface d'une application en définissant manuellement des couleurs, des polices de caractères, etc. On parle aussi de *custumisation* ;

– les plus complexes qui se basent sur une modélisation automatique ou semi-automatique de l'utilisateur appelé *profil utilisateur*. Parmi ces techniques on peut citer :

1. *les tuteurs intelligents* [Bru95] : qui prennent en compte le niveau des apprenants ainsi que leurs connaissances pour leur dispenser un cours de façon adaptée à chacun d'entre eux ;
2. *les interfaces adaptatives* [Che02] : qui permettent d'assister visuellement l'utilisateur à travers un module de recommandation, par exemple, lors de la navigation. Ce module de recommandation pourra mettre en évidence : l'importance d'un document par rapport à la navigation en cours, la liste des répertoires de l'utilisateur qui contiennent un signet pointant vers ce document, etc.
3. *les systèmes basés sur les actions des usagers en phase de recherche* [CKK02] [Sha00] : qui construisent un modèle utilisateur à partir d'un certains nombre d'actions ou usages pré-définis (sauvegardes, jugements, clicks, etc.). Ces systèmes permettent de construire un profil utilisateur de façon transparente à l'usager (profil implicite) ;
4. *les systèmes de personnalisation à base d'annotations* [CB02] : qui s'intéressent particulièrement au partage d'annotations pour faciliter les tâches collaboratives au sein d'une communauté d'utilisateurs. Une nouvelle annotation d'un utilisateur peut être intéressante pour les autres usagers de profil similaire au sien ;
5. *les systèmes de personnalisation de services Web* comme le courrier électronique [GNOT92], les listes de discussions (Usenet News) [KMM+97], les guides de programmes télévisés personnalisés qui sont basés sur le Web [CS00], etc. ;
6. *les systèmes de personnalisation pour les dispositifs mobiles* comme les téléphones portables, les PDA ou *Personnal Digital Assistant* (fonctionnant avec Pocket PC ou Palm OS), etc. Ces dispositifs sont caractérisés par une taille d'affichage réduite, une bande passante limitée et une mémoire restreinte. Les techniques de personnalisation doivent donc prendre en compte ces particularités afin de s'adapter à ce type d'oultils qui sont de plus en plus utilisés pour accéder à l'information via le WAP ou *Wireless Application Protocol*. Dans ce contexte, la personnalisation s'avère particulièrement indispensable pour réduire la quantité d'informations à renvoyer aux usagers. Les contenus Web sont affichés comme des *pages WML*. Le WML[8] ou Wireless Markup Language est l'équivalent du HTML pour les systèmes compatibles WAP. Comme

8. *cf.* http://thewml.org/docs/

exemple de systèmes de ce type, on peut citer : les guides de programmes télévisés personnalisés qui sont basés sur le WAP [CS00]. Notons que pour les téléphones mobiles (ou fixes), en particulier, il existe également des *systèmes d'envois de messages SMS (Short Message Service ou Service de messages courts) personnalisés* pour des annonces diverses : offres, publicités, soldes de comptes bancaires, etc. De plus, on peut aujourd'hui coupler les pages Web qui sont régies par le protocole HTTP, les e-mails qui fonctionnent avec les protocoles POP et SMTP, les pages WML qui sont régies par le protocole WAP et les SMS [9].

Il est à noter que des combinaisons des techniques citées précédemment sont possibles. Notons également que la personnalisation ne peut pas être regardée du seul point de vue d'un individu. Elle s'applique aussi pour un groupe d'individus ayant une ou plusieurs caractéristiques communes. Ainsi, la personnalisation doit permettre de fournir à chaque utilisateur ou groupe d'utilisateurs des informations qui correspondent à son profil.

L'adaptation, quant à elle, va permettre de modifier le comportement d'un système. Elle recouvre deux notions qui sont : *l'adaptabilité et l'adaptativité*. L'*adaptabilité* est la capacité d'un système à être modifié par des usagers. L'*adaptativité* est la capacité du système à modifier son comportement sans intervention explicite de l'usager en apprenant des interactions de ce dernier. Ainsi, l'adaptativité est liée à l'évolution automatique dans le temps, du contenu des profils des usagers [Tma02].

Un système d'accès à l'information peut être :

- *personnalisable uniquement* c'est à dire qu'il prend en compte le profil de chaque utilisateur pour lui renvoyer de l'information ;
- *adaptatif uniquement* c'est à dire qu'il modifie son comportement en fonction des déductions qu'il fait des interactions des usagers. Cependant, il ne fait pas de distinction entre un usager ou un groupe d'usagers particulier. Ici, l'usager représente l'ensemble de la population des utilisateurs (c'est-à-dire tout le monde) ;
- *personnalisable et adaptatif* c'est à dire qu'il identifie chaque usager et chaque groupe d'usagers de façon unique. Il définit alors une stratégie d'accès à l'information qui correspond à chaque utilisateur ou groupe d'utilisateurs et qui évolue en fonction des interactions de chacun (individu ou groupe) ;
- *non personnalisable et non adaptatif.*

Le tableau Tab. 1.6 (voir aussi [TJK99]) illustre des exemples de systèmes personnalisables et/ou adaptatifs.

Dans nos travaux de recherche, nous nous intéressons aux aspects relatifs à la personnalisation et à l'adaptativité de l'accès à l'information.

9. *cf.* http://www.mctel.fr/

Systèmes	Personnalisation	adaptativité
Walden Path [a]	non	non
Alexa [b]	non	oui
Lainé-Cruzel 99 [LC99]	oui	non
Berti-Equille 02 [BE02]	oui	non
Letizia [Lie95]	oui	oui
Personal WebWatcher [Mla96]	oui	oui
Syskill & Webert [PMB96]	oui	oui
Broadway V1 [Jac98]	oui	oui
Easy-Dor [Che02]	oui	oui

TAB. 1.6 – *Exemples de systèmes personnalisables et/ou adaptatifs*

[a] http://www.csdl.tamu.edu/walden/
[b] http://http://www.alexa.com/

Dans la section suivante, nous allons faire une étude comparative de différents systèmes de personnalisation dans le domaine de l'accès à l'information (*cf.* section 1.2) afin de positionner nos travaux de recherches et d'en montrer l'intérêt.

1.5 Étude comparative de systèmes de personnalisation pour l'accès à l'information

Le tableau TAB. 1.7 illustre une étude comparative entre systèmes de personnalisation pour les critères :

a. *contexte d'étude* ou type d'accès à l'information ;

b. *typologie sémantique des profils utilisés* (*cf.* section 1.3.1, TAB. 1.4 et TAB. 1.5) ;

c. *typologie structurelle des profils* (*cf.* section 1.3.1) ;

d. *méthode d'évaluation de la pertinence utilisateur* pour restituer de façon personnalisée des informations à chaque usager (individu ou groupe).

Le tableau TAB. 1.7, nous permet de mettre en évidence la diversité qui existe au niveau des systèmes de personnalisation de l'accès à l'information pour les différents critères de comparaison que nous avons choisis. Dans nos travaux de recherche, nos contributions portent sur ces différents aspects.

1. Les trois premiers critères montrent que chaque application définit la typologie sémantique ou structurelle de ses profils selon l'objectif qu'elle veut atteindre. C'est pour cela que nous nous sommes fixés comme premier objectif, celui de définir un *cadre générique pour la conception de tout type d'applications d'accès à l'information. Ce cadre doit fournir une structure de base homogène à partir de laquelle on doit*

pouvoir concevoir tout type d'applications de recherche ou de filtrage d'information. Le cadre générique que nous proposons comprend :

- *une architecture générique de recherche et de recommandation à base de profils* pour tout type d'applications de RI et/ou FI ;
- *un modèle générique de profil* pour la description de tout type (structure et sémantique) de profils.

Plusieurs travaux ont proposés des modèles génériques d'utilisateurs [Kay95] [Sha00] [Kob01]. Dans nos travaux, nous ne nous intéressons pas qu'aux usagers mais également aux informations mises à disposition ainsi qu'aux interactions entre modèles des usagers et modèles des informations. *La spécificité de notre architecture se situe au niveau des nombreuses possibilités de combinaisons de profils complémentaires décrivant les informations mises à disposition et les usagers de ces informations pour une restitution personnalisée de ces dernières.*

Par la suite, nous nous sommes intéressés aussi aux méthodes d'utilisation des profils pour améliorer l'accès à l'information à travers des *méthodes d'appariement et de transformation de structure et de contenu de profils.*

2. Le but de *l'appariement de profils* en personnalisation est la restitution des informations adaptées aux attentes des utilisateurs. Parmi ces méthodes d'appariement on peut citer :

- *la correspondance aux besoins de l'utilisateur* [PMB96] [Paz99] [BBB03a] qui mesure un degré de similarité entre le profil des besoins de l'utilisateur ou sa requête (éventuellement reformulée) et une information mise à disposition ;
- *la sélection multi-critères* qui permet de restituer à l'utilisateur les informations qui correspondent à un sous ensemble de critères prédéfinis (langue, taille, date, etc.) ;
- *l'interrogation à relance* [Ame01] qui consiste à évaluer dans un premier temps la correspondance aux besoins de l'usager, puis à effectuer une sélection multi-critères sur les résultats obtenus.

Les méthodes d'évaluation de la pertinence utilisateur lorsque qu'elles veulent prendre en compte plusieurs critères sont basées généralement sur des appariements de type booléen ou de type base de données. Dans ce cas, on ne renvoie à l'utilisateur que les informations qui correspondent aux critères choisis et qui respectent les contraintes imposées sur ces critères. Cette approche pose certains problèmes :

- il y a un risque d'avoir un ensemble vide de résultats, si aucune information ne correspond exactement aux critères définis ;
- il y a également le risque d'augmenter *le silence* informationnel en ne restituant pas des informations qui seraient intéressantes pour un sous-ensemble de critères de l'ensemble pré-définis ;

– cette méthode donne la même importance aux différents critères, ce qui ne correspond pas toujours aux objectifs de l'utilisateur. Ainsi, certaines informations qui seraient intéressantes pour un critère très important pour l'utilisateur et moins intéressantes pour les autres critères choisis risque de ne pas être restituées.

Pour pallier cet handicap, il est nécessaire de proposer des méthodes d'évaluation de la pertinence utilisateur basées sur plusieurs critères. L'approche que nous proposons s'inscrit dans cette problématique qui est aussi celle des méthodes de recherche de solutions optimales pour les problèmes de décision basés sur des attributs multiples [ZP04]. *La particularité de notre méthode d'appariement de profils est qu'elle va calculer un degré de ressemblance d'une information à l'utilisateur, relativement à un ensemble de critères pré-définis et pondérés pour/par ce dernier.*

3. *La méthode de transformation de structure et contenu de profils* que nous proposons ici a pour but de faciliter l'appariement de profils de structure complexe (arbre contenant plusieurs nœuds ou feuilles identiques) en les transformant en profils de structure plus simple (réduction du nombre de nœuds ou feuilles de l'arbre). Cette méthode est basée sur la notion de ré-utilisabilité de profils existants pour la construction d'autres profils. De plus, la méthode proposée va pouvoir également être utilisée afin d'assurer l'évolutivité de nos profils (modification de profils long terme à partir de plusieurs profils court terme). *Notons que notre méthode de transformation de structure et de contenu de profils est basée à la fois sur l'usage de la structure et du contenu des profils qui décrivent le profil à transformer.*

1.6 Conclusion

La personnalisation de l'accès à l'information conduit nécessairement à modéliser l'utilisateur du système d'information. Personnaliser signifie s'adapter aux exigences de chaque utilisateur ou groupe d'utilisateurs. De nombreux travaux, présentés dans ce chapitre, ont été menés dans ce sens. Ils sont très variés et ont tous pour préoccupation la satisfaction des usagers.

L'étude comparative effectuée entre différents systèmes de personnalisation (*cf.* section 1.5) nous permet de souligner l'intérêt qu'il y a à définir un cadre générique de recherche ou de filtrage d'information, ainsi que l'intérêt de la définition de méthodes d'appariement et de transformation de structure et contenu de profils pour optimiser la personnalisation de l'accès à l'information. Nos travaux de recherche s'inscrivent dans ce cadre et nous présentons, dans le chapitre suivant, une nouvelle approche pour un accès personnalisé à l'information. L'objectif est d'améliorer la qualité des résultats renvoyés aux utilisateurs.

Systèmes de personnalisation	Contexte	Typologie sémantique de profils	Typologie structurelle de profils	Évaluation de la pertinence utilisateur
Syskill & Webert 96 [PMB96]	Filtrage cognitif	1. profil du contenu de document 2. profil individuel des besoins	mono-critère	mesure la correspondance au profil des besoins de l'utilisateur
Pazzani 99 [Paz99]	Filtrage hybride	1. profil du contenu de document 2. profil individuel des besoins, des jugements, des données démographiques 3. profil de groupe des besoins, des jugements, des données démographiques	mono-critère	Combinaison des cinq premiers résultats des trois types d'approches de filtrage
Amerouali 01 [Ame01]	Recherche	1. profil étendu de document 2. profil individuel étendu	multi-critères	Interrogation à relance
Berti-Equille 02 [BE02]	Recommandation	1. profil étendu de document 2. profil individuel étendu	multi-critères	Recommandation multi-critères
Pitkow 02 [PSC+02]	Recherche	1. profil étendu de document 2. profil individuel étendu 3. profil de groupe étendu	multi-critères	Reformulation de requête en pré-recherche en utilisant le profil des besoins et ordonnancement multi-critères des résultats
Bottraud et al. 03 [BBB03a]	Recherche	1. profil du contenu de document 2. profil individuel des besoins	mono-critère	Reformulation de la requête en post-recherche, réévaluation des résultats et réordonnancement

TAB. 1.7 – *Étude comparative de systèmes de personnalisation pour l'accès à l'information*

Chapitre 2

Une approche à base de profils pour un accès personnalisé à l'information

2.1 Introduction

De nombreux outils d'accès à l'information (moteurs de recherche, systèmes de recommandation) ont été développés pour aider l'utilisateur à rechercher, retrouver et exploiter une information. A travers ces outils, la question de la pertinence système (ou degré de similitude entre le contenu effectif d'un document et les besoins, centres d'intérêt ou requête, en information de l'utilisateur) des résultats restitués a fait l'objet d'une réflexion très approfondie. Cependant, une autre question qui a été beaucoup moins approfondie est celle qui consiste à évaluer si ces résultats sont réellement adaptés à l'utilisateur, relativement à un certain nombre de critères. Il s'agit de s'assurer entre autre que les résultats obtenus sont compréhensibles par l'utilisateur, qu'ils correspondent aux préférences de ce dernier, qu'ils sont compatibles avec son environnement logiciel et matériel, etc. L'objectif est d'améliorer la pertinence en essayant de rapprocher le plus possible la pertinence système de la pertinence personnelle (ou attentes) de chaque utilisateur ou groupe d'utilisateurs en analysant le contexte propre de ces derniers. Pour cela, il faut décrire avec le plus de détails possibles les éléments amenés à interagir au travers de processus d'accès à l'information. Ces éléments sont relatifs aux utilisateurs mais également aux informations recherchées et leurs descriptions sont appelées profils. L'objet des travaux entrepris est de proposer une architecture de recherche et de recommandation pour un accès flexible à l'information. Cette architecture se base sur des profils multi-facettes, réutilisables, adaptables à différents contextes et évolutifs. Cette architecture exploite également la complémentarité entre profils pour améliorer la restitution personnalisée d'informations à travers des appariements de profils et des

transformations de structure et contenu de profils.

Dans ce chapitre, nous présentons nos propositions pour la définition d'une nouvelle approche de personnalisation. Dans un premier temps, nous décrivons un *cadre générique pour l'accès à l'information* à travers une architecture de recherche et de recommandation à base de profils et un modèle générique de profil qui fournissent une structure de base homogène et adaptable à différents contextes de RI et/ou FI. Ensuite, nous présentons nos contributions sur l'utilisation de profils pour l'accès à l'information notamment l'*appariement de profils* suivant un ensemble de critères donné qui permet de mesurer un degré de correspondance d'une information à un usager (ou groupe d'usagers) et la *transformation de profils* de structure complexe (arbre contenant plusieurs nœuds ou feuilles identiques) en profils de structure plus simple (réduction du nombre de nœuds ou feuilles de l'arbre) pouvant conduire à des modifications de contenu, ceci dans le but de faciliter l'appariement des profils de structure complexe.

2.2 Cadre générique pour l'accès personnalisé à l'information

Dans cette section, nous présentons notre architecture de recherche et de recommandation à base de profils. Nous décrivons également le modèle générique de profil à partir duquel nous pouvons dériver différents profils. Ensuite, nous expliquons les règles d'appariement de profils pour la restitution d'informations pertinentes et adaptées à chaque utilisateur ou groupe d'utilisateurs ainsi que les méthodes de construction ou de transformation de la structure et du contenu des profils. Nous illustrons nos différentes propositions par des exemples.

2.2.1 Architecture de recherche et de recommandation à base de profils

Le schéma de la figure Fig. 2.1 [Tch04b] présente l'architecture de recherche et de recommandation à base de profils que nous proposons. Elle résulte de l'analyse de la mise en œuvre de différents systèmes de recherche et de recommandation afin d'en déduire un modèle général. Les systèmes existants sont conçus pour atteindre des objectifs particuliers en fonction des spécificités propres de leur contexte : recommandation de pages web en fonction des signets [RP97], filtrage de mails [GNOT92], commerce électronique [CKK02], etc. Contrairement à ces systèmes, notre architecture est assez générale pour servir de modèle à différents types applications.

Les figures Fig. 2.2 et Fig. 2.3 instancient des exemples d'architectures dérivées de la figure Fig. 2.1. Elles illustrent les différents types de granularité qui peuvent exister tant au niveau des usagers que des informations

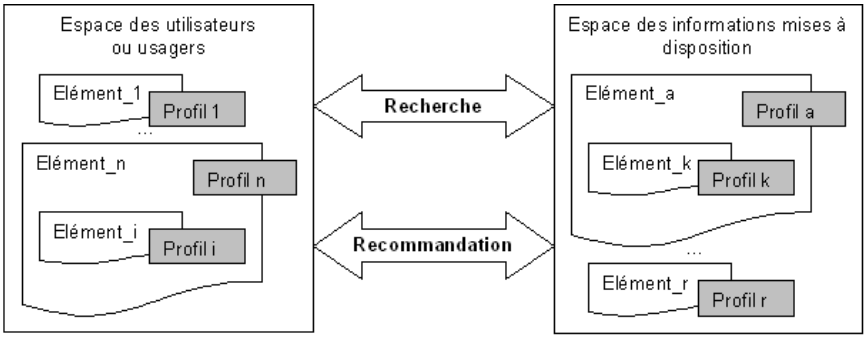


FIG. 2.1 – *Architecture générale de recherche et de recommandation à base de profils*

mises à dispositions. Cette granularité est traduite schématiquement par la composition ou l'imbrication des éléments de l'architecture. Cependant on note également la possibilité d'une juxtaposition d'éléments différents pour un même niveau d'imbrication donné FIG. 2.3. Notre architecture va pouvoir se servir de cette hétérogénéité de structure pour assurer une exploitation maximale de la compléméntarité entre différents profils qui décrivent des éléments de l'architecture.

Notre architecture ne s'applique pas à un cadre pré-défini. Elle est constituée d'un ensemble d'éléments qui interviennent dans la mise en œuvre d'un système de recherche et/ou de recommandation. C'est à chaque application de sélectionner dans cette architecture les éléments qui l'intéressent. Notre architecture peut être utilisée comme point de départ pour toute construction de systèmes de recherche et/ou de recommandation.

Sont mis en évidence, dans cette architecture, les processus de recherche et de recommandation ainsi que la structure générique des éléments manipulés par ces derniers. Ces éléments sont regroupés en deux grands groupes : ceux qui sont relatifs à l'espace des utilisateurs et ceux relatifs à l'espace des informations mises à disposition. Notons qu'à chaque type d'élément on associe un profil qui le décrit de façon détaillée et qui est exploitable par les processus d'accès à l'information. De plus, ces éléments peuvent aussi être composés d'un ou de plusieurs sous éléments (granularité : *cf.* figure FIG. 2.3) eux-mêmes décrits par des profils.

Les éléments liés à l'espace des utilisateurs peuvent être, par exemple :

- *les informations de l'espace de travail des différents utilisateurs* : historique des informations d'usage (requêtes, sites visités, informations jugées, informations transférées, informations sauvegardées, etc.), structure et contenu d'informations diverses (signets, courriers, etc.) ;
- *les informations sur l'environnement de travail des utilisateurs* : environnement logiciel et matériel ;

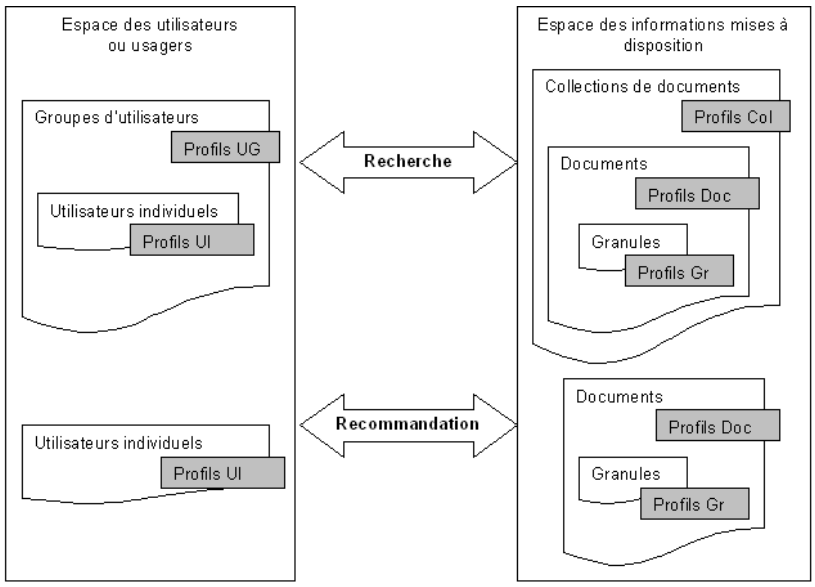


FIG. 2.2 – *Exemple d'architecture base de profils : granularité au niveau des usagers et des informations mises à disposition*

– *les informations sur les utilisateurs* : données démographiques, centres d'intérêt, préférences, etc. Ces informations sont obtenues soit manuellement, soit par des méthodes automatiques ou semi-automatiques (*cf.* 1.3.2.3 et 1.3.2.2).

Notons que les éléments de l'espace des utilisateurs peuvent être définis par individu ou par groupe d'individus. Les profils de ces éléments peuvent donc être combinés pour décrire des individus ou groupe d'individus afin de constituer leur profil. Un profil utilisateur peut être : de court terme ou de long terme [WIY99] (*cf.* section 1.3.1.2), positif ou négatif [HKNH00] (*cf.* section 1.3.1.2), etc.

De même, les éléments liés à l'espace des informations mises à disposition peuvent être, par exemple :

– *des informations* comme : des documents, des parties ou granules de documents (chapitres, paragraphes, sections, etc.), des collections de documents, des sites ou pages Web, des articles de journaux, des résumés d'articles scientifiques comme dans la base Medline, etc. Ces informations sont éventuellement annotées par des utilisateurs ou par des experts ou auteurs ;
– *des ressources physiques* comme : des supports d'information, des serveurs, etc.

L'objectif de l'architecture proposée est la restitution personnalisée d'informations à travers l'usage de différents profils. Ces profils sont issus d'un

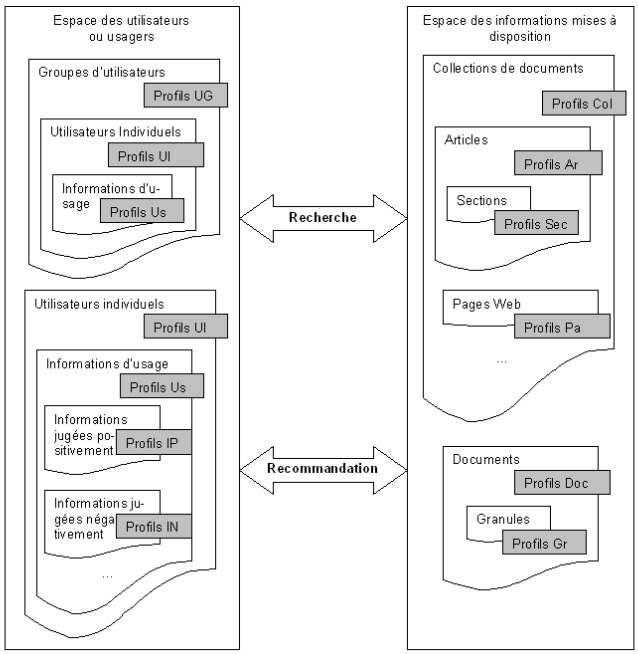


FIG. 2.3 – *Exemple d'architecture à base de profils : granularité au niveau des composants des éléments de l'architecture*

modèle générique que nous présentons dans la section suivante.

2.2.2 Modèle générique de profil

Afin de pouvoir définir différents profils qui soient ré-utilisables, multi-facettes, adaptables et évolutifs, nous avons défini un modèle générique de profil.

Le modèle générique de profil de la figure FIG. 2.4 [CSDT04] présente la structure générale d'un profil. Cette structure est sous la forme d'une hiérarchie de catégories de critères permettant de caractériser un profil. Cette hiérarchie est une forêt ou un ensemble d'arbres dont les nœuds sont des catégories ou classes de critères et les feuilles sont tout simplement des critères auxquels on peut affecter des valeurs. Un profil peut donc être soit une forêt, soit un arbre, soit une végétation (ou liste) de critères. Ainsi, si P est un profil : *Structure(P)={forêt, arbre, végétation}*.

Les profils dérivés de ce modèle générique peuvent avoir les caractéristiques suivantes :

– *profils ré-utilisables* : l'agrégation réflexive sur la classe «profil» traduit le fait qu'un sous-arbre d'un profil peut avoir la structure d'un autre profil existant. Ainsi, la structure de certains profils peut être

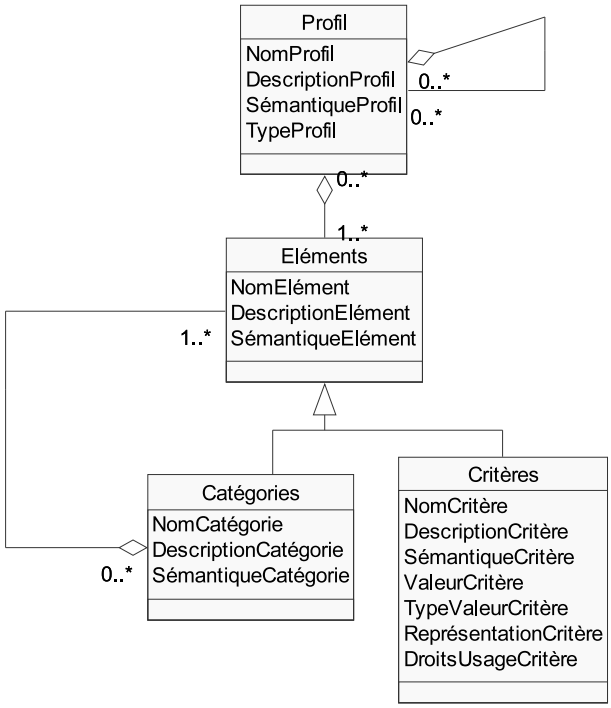


FIG. 2.4 – *Modèle générique de profil*

ré-utilisable dans d'autres profils. Par exemple, un profil utilisateur peut-être composé : de ses différents profils d'usage (ou profils court terme) et/ou d'un profil d'environnement existant. De plus, une liste de métadonnées pour des informations textuelles ou pour des images peut-être ré-utilisée pour décrire des profils de documents ou parties de documents. Dans ce cas, ces structures (comme la liste de métadonnées de MPEG 7) peuvent être considérées comme des *profils abstraits* permettant la définition de la structure ou partie de la structure d'autres profils ;

– *profils multi-facettes* : ce sont des profils qui peuvent être analysés sous différents angles (critères, sous-profils, etc.). Ainsi, un profil utilisateur peut être composé par son profil positif et son profil négatif ou encore par ses différents profils court terme ou une combinaison de ces différents profils. De même, on peut retrouver dans un profil utilisateur des pointeurs vers les profils de groupes auxquels il appartient. Chaque profil ou combinaison de profils peut constituer une facette de l'utilisateur. La figure FIG. 2.5 illustre des exemples de facettes utilisateurs qui représentent des vues différentes du profil d'un même utilisateur ;

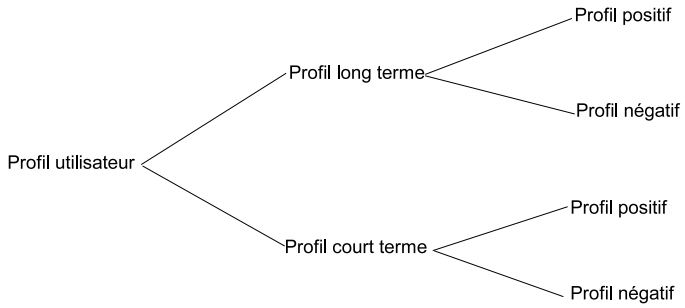


FIG. 2.5 – *Exemples de facettes d'un profil utilisateur*

– *profils adaptables* : tous les critères d'un profil donné ne sont pas forcément renseignés. De plus, en fonction de l'application qui l'utilise tous les critères d'un profil ne sont pas forcément pris en compte. Chaque application choisit les éléments à considérer dans un profil en fonction de l'objectif qu'elle veut atteindre et peut également rajouter des éléments selon le besoin. La généralisation de la structure d'un profil quelconque constitué de plusieurs critères (critères de type feuille) pas toujours tous renseignés, permet de garder le même profil pour un élément donné de l'architecture. Un profil peut être partagé et enrichi par différentes applications : ce sont des profils adaptables. Par exemple, un profil utilisateur peut être considéré comme une «carte d'identité» de la personne. Quelle que soit l'application, l'utilisateur peut être reconnu avec le même profil (critères de type feuille et contenu) ;
– *profils évolutifs* : nos profils peuvent être modifiés et peuvent évoluer dans le temps tant au niveau de leur structure que de leur contenu. Au niveau de la structure, il est question de la modification de la structure d'un profil pour des besoins applicatifs (appariements de profils, etc.). Au niveau du contenu, il s'agit de décrire les conditions de mises à jour : des profils d'informations (qui peuvent être dues à une modification de contenu) et des profils utilisateurs (qui peuvent être liées au fait que plusieurs profils court terme soient différents du profil long terme de l'usager).

D'autre part, l'organisation des différents critères par catégorie permet de regrouper les critères similaires dans une même classe et de définir ainsi une nomenclature (ou taxinomie) des critères. Cette taxinomie est une façon de définir la sémantique ou du moins une partie de la sémantique de certains critères. A partir du modèle générique de profil, nous pouvons dériver la structure de différents profils en appliquant des règles de décomposition sur des catégories de critères. Les figures FIG. 2.6, FIG. 2.7, FIG. 2.8 et FIG. 2.9 présentent, respectivement, des exemples de structures de profils pour : un profil individuel d'utilisateur, un profil d'information, un profil de groupe

d'utilisateurs et un profil de collection de documents.

Plus particulièrement, les figures FIG. 2.6, FIG. 2.7 instancient des exemples de profils dérivés de la figure FIG. 2.4 en mettant en évidence les parties : structure logique (ou taxinomie) et contenu des profils.

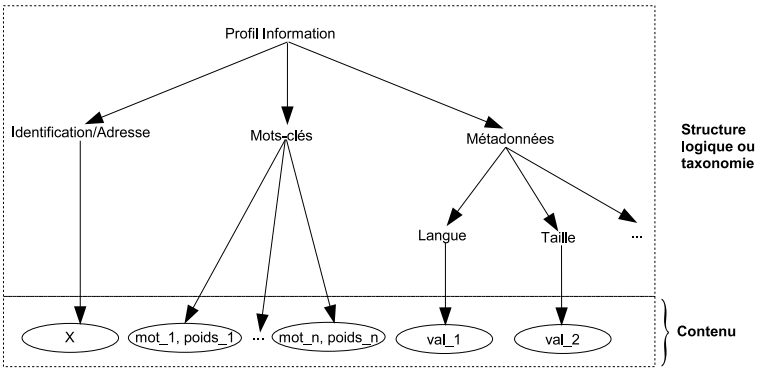


FIG. 2.6 – *Exemple de profil d'une information mise à disposition : structure et contenu*

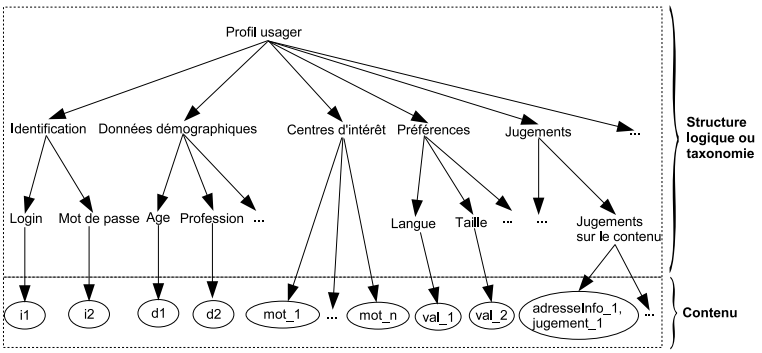


FIG. 2.7 – *Exemple de profil usager : structure et contenu*

Les figures FIG. 2.8 et FIG. 2.9 instancient des exemples de profils dérivés également de la figure FIG. 2.4 qui sont, dans ce cas, composés d'autres profils. Elles illustrent la granularité qui peut exister au sein de ces différents types de profils. La granularité est matérialisée par la profondeur des profils de l'arbre et donc par la composition de profils. Notons que différents profils peuvent appartenir à la même profondeur d'une arborescence donnée.

Dans ces figures (*cf.* FIG. 2.8 et FIG. 2.9), la composition concerne à la fois la structure et le contenu des profils. On pourrait également avoir une composition qui soit uniquement structurelle. Dans ce cas, on peut imaginer que l'on a des structures logiques abstraites de profils que l'on peut ré-utiliser

pour décrire d'autres profils. A titre d'exemple, on peut citer la structure des métadonnées de la norme MPEG-7 qui permet de décrire des informations multi-média.

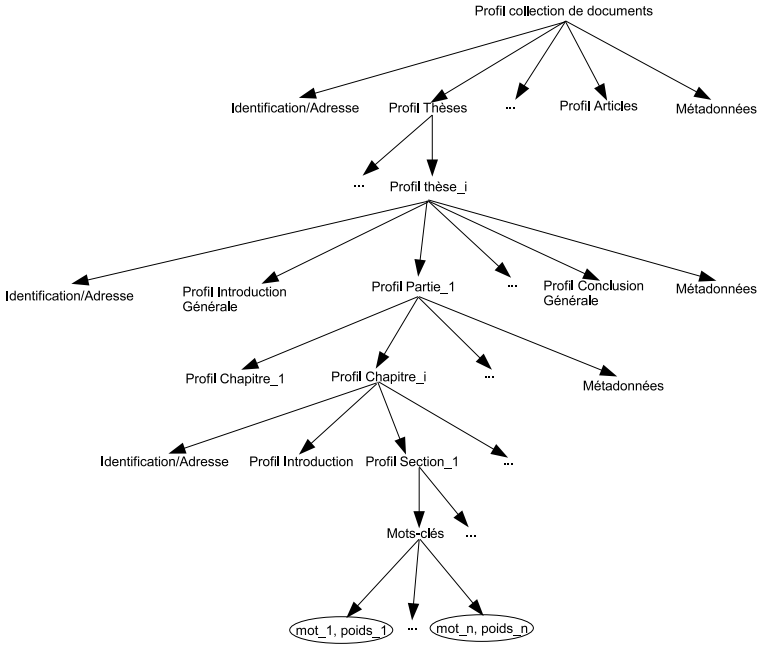


FIG. 2.8 – *Illustration de la composition de profils pour la description des informations mises à disposition*

L'intérêt de l'utilisation d'un profil générique pour définir un type de profil donné est que la structure de base qu'il propose peut être utilisée par tout type d'application afin de définir tout type de profils. Ici, nous ne nous intéressons pas à des critères particuliers mais plutôt à la mise en place d'un cadre pour la définition de ces critères. Pour chaque critère, il faut définir un certain nombre de propriétés (du critère) pour faciliter son utilisation par les processus d'accès à l'information : nom, valeur, type, représentation, sémantique, sécurité, etc.

Le modèle générique de profil de la figure FIG. 2.4, va nous permettre de dériver la structure de différents profils sous la forme d'une hiérarchie de catégories ou classes de critères. Dans le contexte de la recherche et de la recommandation d'informations textuelles nous pouvons envisager, par exemple, les modèles de profils illustrés par le tableau TAB. 2.1 et décrivant une taxinomie pour les profils individuels d'utilisateurs, de groupes d'utilisateurs, de documents et de sources ou collections de documents. Dans le tableau TAB. 2.1, chaque catégorie est composée d'une ou de plusieurs sous catégories. Ces listes ne sont pas exhaustives et on peut ajouter de nouvelles catégories ou sous catégories selon le besoin. On peut également effectuer

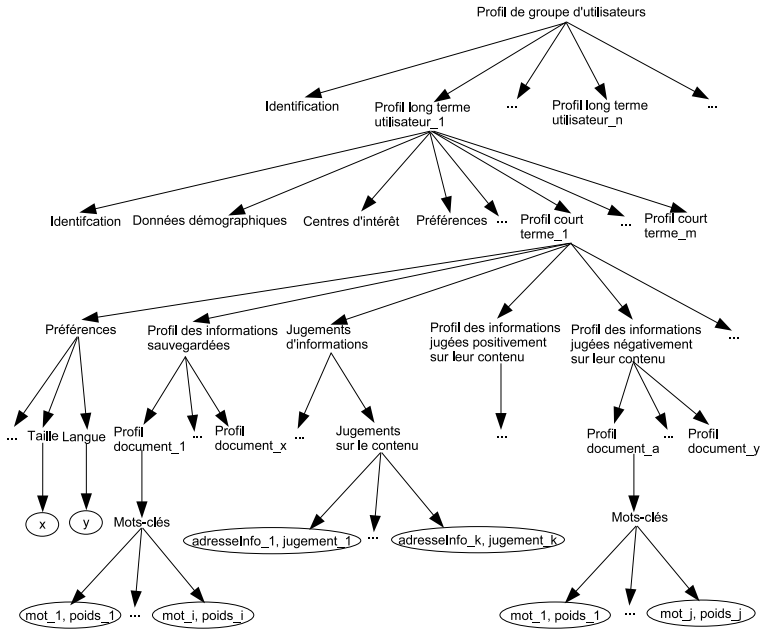


FIG. 2.9 – *Illustration de la composition de profils pour la description des usagers*

d'autres décompositions sur les sous catégories. La structuration des profils en hiérarchie de catégories permet de modéliser les informations ou éléments que l'on souhaite décrire. L'objectif est d'aider à trouver une information qui corresponde à l'utilisateur ou à faire savoir qu'elle existe.

Le tableau TAB. 2.1 (voir aussi [Tch04a]) illustre des taxinomies conjointes des profils relatifs aux informations mises à disposition et aux utilisateurs.

Différentes structures de profils peuvent décrire un même élément. Ainsi, un profil peut être décrit uniquement par une taxinomie simple (c'est-à-dire que sa structure est composée uniquement de catégories et de critères *cf.* FIG. 2.6 et 2.7). Il peut également être composé par un ensemble de profils (*cf.* FIG. 2.8 et 2.9) qui eux-mêmes sont décrits par des taxinomies. Cette composition de profils peut conduire à des structures de profils complexes où plusieurs nœuds ou feuilles de l'arbre sont identiques. Il pourrait être intéressant de pouvoir déduire un profil décrit uniquement par des catégories et des critères (profil de structure simple) à partir d'une structure de profil composée d'autres profils, ceci dans le but de faciliter l'exploitation du profil par les processus d'accès à l'information. Pour ce faire, des règles de transformation de structure et de contenu doivent être définies (*cf.* section 2.3.2).

L'architecture et le modèle de profil proposés vont nous permettre de définir des profils (ré-utilisables, multi-facettes, adaptables et évolutifs) pour

Catégories de critères	Profil individuel utilisateur	Profil de groupe d'utilisateurs	Profil de granules de document	Profil de document
Identification	Login, Mot de passe	Identifiant	Adresse	Adresse
Sujet	Centres d'intérêt	Centres d'intérêt	Mots clés	Mots clés
Préférences Ou Métadonnées	Langue, Taille, Popularité	Langue, Taille	Langue, Taille	Langue, Taille, Popularité
Jugements	Contenu, Utilisabilité			
Données démographiques	Âge, Profession			Âge cible, Profession cible
Environnement	Matériels, Logiciels	Matériels, Logiciels		
Critères de gestion				Accessibilité, Disponibilité
...				

TAB. 2.1 – *Taxinomies conjointes des profils relatifs aux informations mises à disposition et aux utilisateurs*

un accès flexible à l'information. La combinaison de différents profils va permettre une exploitation optimale de la complémentarité entre profils au travers de méthodes d'appariement et de transformation de structure et de contenu de profils.

2.3 Utilisation de la structure et du contenu des profils pour l'accès à l'information

Dans la section 2.2 nous nous sommes intéressés à la représentation et à la structuration des informations qui sont manipulées par les processus de recherche et de recommandation en proposant un cadre générique qui offre de nombreuses possibilités d'interactions entre profils complémentaires. Dans cette section, nous allons proposer des méthodes d'utilisation de ces profils pour l' accès à l'information à savoir : une méthode d'appariement de profils basée sur un ensemble de critères et une méthode de transformation de structure et de contenu de profil à partir d'autres profils existants.

2.3.1 Appariement de profils pour l'accès à l'information

L'appariement de profils est lié aux appariements des critères décrivant ces profils. Nous avons identifié différents appariements de critères de profils. La combinaison de certains de ces appariements va permettre de sélectionner les résultats correspondant aux utilisateurs ou tout simplement de réordonnancer ces résultats. Pour apparier des critères de profils différents il faut que ces critères aient la même sémantique (type des valeurs des critères, modèle de représentation, caractéristique représentée, etc.).

Nous avons classifié les appariements de critères de profils comme suit :

1. *appariement de type booléen ou de type base de données* : cet appariement est utilisé lorsque les critères à apparier sont mono-valués et non-pondérés ;

Un exemple de ce type d'appariement est *l'évaluation de la compatibilité aux préférences mono-valués de l'usager* : cet appariement consiste à effectuer des comparaisons entre les critères mono-valués des préférences de l'utilisateur et les critères correspondant de description des informations (comme la popularité, la taille, la profession cible, l'âge cible, etc.). On peut comparer, par exemple, l'âge cible pour un document à l'âge de l'usager. Le résultat de ce type d'appariement est binaire ;

2. *appariement de type RI* : ce type d'appariement est utilisé lorsque au moins un des critères à apparier est multi-valué et/ou pondéré. Dans ce cas, on représente les différents critères à apparier dans un même espace vectoriel dont la dimensionnalité est donnée par la taille du vocabulaire. A chaque vecteur de termes, noté par exemple $d = (t_1,t_2,\ldots,t_n)$, est associé un vecteur de poids réel ou booléen, noté $p_d = (w_{d,t_1},w_{d,t_2},\ldots,w_{d,t_n})$, qui permettra de calculer un degré de similarité entre critères à apparier ;

Comme exemples de ce type d'appariement on peut citer :

- *la correspondance aux besoins de l'usager* : il s'agit d'une mesure de similarité entre le vecteur des poids des termes représentant les besoins (requête éventuellement reformulée ou centres d'intérêt) de l'utilisateur et celui des termes représentant le contenu d'une information (document, granule de document, collection de documents, etc.). Les poids, dans ce cas, sont généralement calculés avec les formules de *tf* (*cf.* formule 1.1) ou *tf.idf* (*cf.* formule 1.3) et la similarité avec la formule du cosinus (*cf.* formule 1.12) ;
- *la compatibilité aux préférences multi-valuées de l'utilisateur* : il s'agit de mesurer la similarité d'une information pour un critère donné (langue, format, etc.) aux préférences de l'utilisateur pour ce critère. Le tableau TAB. 2.2 illustre un exemple d'évaluation de cet appariement pour le critère langue ;

Critère Langue f	Anglais (t_1)	Français (t_2)	Espagnol (t_3)	Similarité
Poids du document pour le critère Langue p_d	$w_{d,t_1} = 0$	$w_{d,t_2} = 1$	$w_{d,t_3} = 0$	$sim(p_d,p_u)$ $= \sum_{i=1}^{i=3} w_{d,t_i} \cdot w_{u,t_i}$ $= v_{d,u,f} = 0.5$
Préférences utilisateur en langue p_u	$w_{u,t_1} = 1$	$w_{u,t_2} = 0.5$	$w_{u,t_3} = 0.25$	

TAB. 2.2 – *Exemple de calcul de la compatibilité aux préférences en langue de l'utilisateur*

- *la compatibilité à l'environnement de l'utilisateur* : il s'agit de garantir que l'utilisateur dispose de matériels et de logiciels nécessaires à l'exploitation du document qui lui est renvoyé ;
- etc.

Le problème qui se pose est de pouvoir évaluer la similarité entre deux profils lorsque ces derniers sont décrits par plusieurs critères. Nous proposons, pour cela, une méthode qui combine différents appariements effectués sur les critères de ces profils.

2.3.1.1 Méthode de combinaison d'appariements

Pour combiner différents types d'appariements, ces derniers doivent tout d'abord être effectués séparément. Ainsi, chaque résultat d'appariement (ou combinaison de résultats d'appariements) va représenter un facteur potentiel de sélection ou d'ordonnancement des informations. On peut donc décrire les informations recherchées sous forme de listes de facteurs (ou appariements) $a = (f_1,f_2,\ldots,f_n)$ auxquels vont correspondre des vecteurs de résultats de ces appariements effectués entre des couples de critères appariables de profils. Ainsi, soit u et d des profils à apparier, le vecteur des résultats des appariements entre ces profils est noté $p_{d,u} = (v_{d,u,f_1},v_{d,u,f_2},\ldots,v_{d,u,f_n})$. Une sous-liste de a peut-être utilisée pour la sélection (on la note a_s) et une autre (ou la même) pour l'ordonnancement (on la note a_o) des informations. Un exemple de liste de facteurs de sélection ou d'ordonnancement des informations peut être : *correspondance aux besoins de l'usager (pertinence du document, des granules, de la collection), compatibilité aux préférences en langue de l'usager, compatibilité à l'environnement de l'usager, etc.*

De plus, à chaque utilisateur ou groupe d'utilisateurs ou pour l'ensemble de la population des usagers, on va associer un vecteur de poids pour une liste de facteurs donnée a' (a' pouvant être a_s ou a_o). Ce vecteur de poids est noté $p_{a',x} = (w_{f_1},w_{f_2},\ldots,w_{f_n})$ et décrit le pouvoir discriminant (ou l'im-

portance) des facteurs les uns par rapport aux autres. Ainsi, w_{f_j} est le poids ou l'importance du facteur f_j. Afin de calculer les valeurs des w_{f_j}, des ordres de préférences doivent être donnés pour tous les éléments du vecteur $p_{a',x}$. Considérons les ordres de préférences d'un utilisateur, pour une liste de facteurs donnée, définit dans le tableau TAB. 2.3. La méthode de calcul des éléments du vecteur $p_{a',x}$ est donnée par :

$$\alpha_i = \beta \Sigma_{j>i} \alpha_j \tag{2.1}$$

où α_1 et β sont pré-définis et α_i représente le poids des facteurs d'ordre de préférence i (car plusieurs facteurs peuvent avoir le même ordre de préférence). Ainsi, si on a k ordres de préférences, on aura $(k-1)$ équations à $(k-1)$ inconnues à résoudre :

$$\begin{cases} \alpha_1 = \beta(\alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \cdots + \alpha_{k-1} + \alpha_k) \\ \alpha_2 = \beta(\alpha_3 + \alpha_4 + \alpha_5 + \cdots + \alpha_{k-1} + \alpha_k) \\ \alpha_3 = \beta(\alpha_4 + \alpha_5 + \cdots + \alpha_{k-1} + \alpha_k) \\ \cdots \\ \alpha_{k-2} = \beta(\alpha_{k-1} + \alpha_k) \\ \alpha_{k-1} = \beta\alpha_k \end{cases}$$

Pour déterminer les α_i on peut utiliser la méthode du pivot de Gauss. Notons que si on souhaite que les $\alpha_i \in [0,1]$, α_1 doit être fixé à 1.

`vecteur de facteurs` a'	f_1	f_2	f_3	f_4	f_5	f_6	$\ldots$	f_n
`Ordres de préférences` i	1	1	2	3	3	4	...	k
`vecteur de poids des facteurs` $p_{a',x}$	$w_{f_1} = \alpha_1$	$w_{f_2} = \alpha_1$	$w_{f_3} = \alpha_2$	$w_{f_4} = \alpha_3$	$w_{f_5} = \alpha_3$	$w_{f_6} = \alpha_4$	$\ldots$	$w_{f_n} = \alpha_k$

TAB. 2.3 – *Ordres de préférences et poids des facteurs de sélection ou d'ordonnancement des informations*

On peut donc calculer un *poids de sélection* p_s et/ou un *poids d'ordonnancement* p_o pour chaque information qui sera une fonction de $p_{d,u}$ et $p_{a',x}$. Ces poids peuvent être évalués à l'aide de la formule de la moyenne pondérée qui est définie, dans ce contexte, comme suit :

$$f(p_{d,u}, p_{a',x}) = \frac{\sum_j v_{d,u,f_j} \cdot w_{f_j}}{\sum_j w_{f_j}} \tag{2.2}$$

L'utilisation d'une moyenne pondérée, plutôt que celle d'une somme pondérée, permet de normaliser la valeur des poids de sélection ou d'ordonnancement à l'intervalle des valeurs possibles pour les résultats d'appariements v_{d,u,f_j}.

Pour la sélection des informations, il sera nécessaire de définir un seuil pour décider si la correspondance d'une information à un utilisateur est assez significative. On peut utiliser, pour cela, des méthodes basées sur la distribution des scores [AH01] [BTT04]. Les scores, dans ce cas, sont les poids de sélection ou d'ordonnancement des informations.

Notons que la liste complète de tous les appariements possibles a est généralement déterminée en fonction de l'ensemble des critères de tous les profils définis pour une application donnée. Les sous-listes a_s et a_o et les ordres de préférences de leurs facteurs sont déterminés soit manuellement par les utilisateurs, soit par l'application qui les fixe pour les différents usagers.

L'algorithme du tableau TAB. 2.4 résume les étapes à suivre pour la restitution d'informations adaptées à chaque usager.

`Algorithme de restitution d'informations à un utilisateur`
1. Choix des profils à utiliser qui décrivent des éléments de l'architecture
2. Détermination des différents appariements en fonction de la liste de critères de description des profils sélectionnés : $a = (f_1,f_2,\ldots,f_n)$
3. Effectuer les différents appariements : $p_{d,u} = (v_{d,u,f_1},v_{d,u,f_2},\ldots,v_{d,u,f_n})$
4. Combiner les résultats des différents appariements
4.1 Détermination de sous-listes pour la sélection (a_s) et/ou l'ordonnancement (a_o) des résultats à partir de la liste a
4.2 Détermination des ordres de préférences pour chaque liste
4.3 Détermination des poids des facteurs de chaque liste : $p_{a_s,x} = (w_{f_1},w_{f_2},\ldots,w_{f_l})$ et $p_{a_o,x} = (w_{f_1},w_{f_2},\ldots,w_{f_n})$
4.4 Calcul des poids de sélection p_s et des poids d'ordonnancement p_o de chaque information pour chaque usager
5. Restituer les informations

TAB. 2.4 – *Algorithme pour un accès flexible et personnalisé à l'information*

Étant donné que tous les critères descriptifs d'un profil ne sont pas toujours renseignés, il va se poser le problème des valeurs nulles et de leur gestion dans la restitution d'informations adaptées aux usagers.

2.3.1.2 Illustration de la combinaison d'appariements et influence des valeurs nulles

Dans cette section, nous illustrons la méthode proposée au travers d'un exemple. Nous allons utiliser deux types de profils : des profils de documents représentés dans le tableau TAB. 2.5 et des profils utilisateurs représentés dans le tableau TAB. 2.6.

Le choix des critères utilisables pour les appariements va permettre de définir la liste de facteurs $a = (f_1,f_2,f_3,f_4)$ pour représenter les documents. Les f_j représentent respectivement et par ordre de pouvoir discriminant :

Profils de documents	Adresse	(mots clés, poids)	Langue	Taille	Popularité
D1	X1	(landscape, 0.73) ; (nature, 0.59) ; (flower, 0.62)	anglais	normal	0.86
D2	X2	(paysage, 0.28) ; (architecture, 0.65)	français	court	0.42
D3	X3	(paysage, 0.51) ; (nature, 0.72) ; (faune, 0.63)	français	long	0.73
D4	X4	(paisaje, 0.32) ; (naturaleza, 0.23) ; (ciudad, 0.54)	espagnol	normal	0.35

TAB. 2.5 – *Exemples de profils de documents*

Profils utilisateurs	(login, mot de passe)	Centres d'intérêt	Préférences en Langue : (langue, poids)	Préférences en Taille : (taille, poids)
U1	(toto, ****)	nature, paysage	(anglais, 1) ; (français, 0.5) ; (espagnol, 0.25)	
U2	(titi, ****)	nature, paysage		(long, 0.25) ; (normal, 0.5) ; (court, 1)

TAB. 2.6 – *Exemples de profils utilisateurs*

la correspondance aux besoins ou centres d'intérêt (ordre 1), la compatibilité aux préférences en langue (ordre 1), la compatibilité aux préférences en taille (ordre 2), la popularité (ordre 3). Pour chaque document et pour chaque utilisateur on aura un vecteur de résultats d'appariements $p_{d,u} = (v_{d,u,f_1}, v_{d,u,f_2}, v_{d,u,f_3}, v_{d,u,f_4})$. Dans cet exemple, le principe de restitution des informations consiste à sélectionner les documents selon les facteurs f_1 et f_2 et à ordonnancer les résultats sélectionnés selon tous les facteurs de a. La liste a est donc égale à la liste a_o. Dans cet exemple, les vecteurs $p_{a_s,x}$ et $p_{a_o,x}$ sont les mêmes pour tous les utilisateurs. Les éléments de $p_{a_s,x}$ ont le même ordre de préférence. α_1 est fixé à 1, donc ils ont tous les deux (f_1 et f_2) la valeur 1. Le calcul des éléments de $p_{a_o,x}$ se fait à partir du système d'équations suivant :

$$\begin{cases} \alpha_1 = \beta(\alpha_2 + \alpha_3) \\ \alpha_2 = \beta\alpha_3 \end{cases}$$

Avec $\alpha_1 = 1$ et $\beta = 2$, on obtient $\alpha_2 = 0.333$ et $\alpha_3 = 0.165$. Le calcul des poids de sélection (p_s) et des poids d'ordonnancement (p_o) des documents pour les utilisateurs U1 et U2 est décrit dans le tableau TAB. 2.7.

Facteurs	f_1 formule du cosinus	f_2 produit scalaire	f_3 produit scalaire	f_4	$p_s = (\sum_j v_{d,u,f_j} \cdot w_{f_j}) / \sum_j w_{f_j},\ j < 3$ $p_o = (\sum_j v_{d,u,f_j} \cdot w_{f_j}) / \sum_j w_{f_j},\ j < 5$
$p_{a_o,x}$	$w_{f_1} = 1$	$w_{f_2} = 1$	$w_{f_3} =$ 0.333	$w_{f_4} =$ 0.165	
p_{d_1,u_1}	0,829745715	1	nul	0.86	$p_s =$ **0,914872858** $p_o =$ 0,910690861
p_{d_2,u_1}	0,27974834	0.5	nul	0.42	$p_s =$ **0,38987417** $p_o =$ 0,39217013
p_{d_3,u_1}	0,802226992	0.5	nul	0.73	$p_s =$ **0,651113496** $p_o =$ 0,657125631
p_{d_4,u_1}	0,581758201	0.25	nul	0.35	$p_s =$ **0,415879101** $p_o =$ 0,410858291
p_{d_1,u_2}	0,829745715	nul	0.5	0.86	$p_s =$ **0,829745715** $p_o =$ 0,759776846
p_{d_2,u_2}	0,27974834	nul	1	0.42	$p_s =$ **0,27974834** $p_o =$ 0,45530597
p_{d_3,u_2}	0,802226992	nul	0.25	0.73	$p_s =$ **0,802226992** $p_o =$ 0,671513346
p_{d_4,u_2}	0,581758201	nul	0.5	0.35	$p_s =$ **0,581758201** $p_o =$ 0,538056209

TAB. 2.7 – *Calcul des poids de sélection et d'ordonnancement en tenant compte des valeurs nulles*

1. *Discussion sur les valeurs nulles et commentaires sur le tableau* TAB. *2.7*:
En cas de valeur nulle, deux choix s'offrent à nous :
- considérer la valeur nulle comme un zéro (ce qui correspond au pire des cas) et dans ce cas, le dénominateur des formules de poids reste le même ;
- ne tenir compte que des valeurs renseignées et dans ce cas, le dénominateur de la formule du poids ne prend pas en compte le poids des facteurs pour lesquels la valeur (résultat d'appariement) est nulle.

Si p_p est le poids obtenu en remplaçant la valeur nulle par la valeur 0 et p_n le poids obtenu en tenant compte de la valeur nulle, on aura :

$$\begin{cases} \text{numérateur}(p_p) = \text{numérateur}(p_n) \\ et \\ \text{dénominateur}(p_p) > \text{dénominateur}(p_n) \end{cases}$$

On aura donc toujours $p_p < p_n$. Ceci signifie qu'en considérant la valeur nulle, on augmente les chances de restituer un document dont certains critères ne seraient pas renseignés. Dans notre tableau TAB. 2.7, nous n'avons pas remplacé les valeurs nulles par des zéros. De ce fait, si on a une valeur nulle pour un facteur donné, le dénominateur de la formule du poids de sélection ou du poids d'ordonnancement ne prend pas en compte le poids de ce facteur ;

2. *Définition du seuil de sélection* : la définition d'un seuil optimal de sélection pour chaque usager est très importante, dans ce contexte, car pour un même besoin en information et pour un même ensemble d'informations on n'aura pas toujours les mêmes scores. D'un utilisateur à un autre, le poids d'un document peut être très différent. Afin, de pouvoir définir pour chaque utilisateur un seuil optimal, nous préconisons d'utiliser des méthodes basées sur la distribution des scores [BTT04].

En conclusion, les résultats restitués aux différents utilisateurs, dans l'ordre, sont :

– l'utilisateur U1 reçoit les documents D1 et D3 ;

– l'utilisateur U2 reçoit les documents D1, D3 et D4.

Le document D4 est sélectionné pour l'utilisateur U2 parce que l'usage de la valeur nulle a permis d'obtenir un poids de sélection plus important que si on avait remplacé la valeur nulle par un zéro.

Par cet exemple, nous avons illustré le fait que la combinaison de différents appariements permet de restituer des réponses adaptées à chaque usager. Les deux utilisateurs ont le même besoin en information mais leurs ensembles de résultats sont différents. De plus, nous montrons que l'usage de valeurs nulles permet d'augmenter la probabilité de restitution d'un document par rapport *au pire des cas* qui correspond à l'utilisation du zéro à la place de la valeur nulle.

Dans la section suivante, nous présentons les méthodes que nous avons définies pour la transformation de profils afin de faciliter leur appariement dans les processus d'accès à l'information.

2.3.2 Transformation de profils

La composition de profils peut créer la confrontation de plusieurs taxinomies où plusieurs catégories ou critères ont une même sémantique dans une même arborescence. Ceci peut rendre difficile l'appariement entre deux

profils ayant ce type de structure(*cf.* FIG. 2.8 et FIG. 2.9). Pour résoudre ce problème, nous définissons une méthode permettant de transformer ces profils et de les ramener à une structure ne contenant qu'un seul profil où chaque catégorie ou critère a une sémantique qui lui est propre.

La méthode de transformation de profils que nous proposons va se baser principalement sur l'usage des profils existants pour la construction d'autres profils. Dans cette section, nous allons définir comment construire un profil de structure simple (*cf.* FIG. 2.6 ou FIG. 2.7) à partir d'une description de ce profil qui serait composée de plusieurs autres profils. Il s'agit de pouvoir passer d'une structure du type FIG. 2.8 ou FIG. 2.9 à une structure de type FIG. 2.6 ou FIG. 2.7. Cette approche permet d'exploiter la *ré-utilisabilité* des profils pour en construire d'autres.

Nous allons nous inspirer des approches comme l'indexation, le profiling, les approches par stéréotypes pour définir notre approche de transformation. Cependant, nous allons analyser directement des profils existants au lieu d'analyser des informations brutes. La particularité de notre approche est qu'elle est basée sur l'exploitation de la structure et du contenu des profils. Avant de décrire cette approche, nous avons défini un certain nombre de concepts relatifs aux arborescences de profils et que nous allons utiliser.

Ainsi, soit A une arborescence avec $N = \{\eta_1, \ldots, \eta_n\}$ l'ensemble des nœuds de A, $C = \{\varsigma_1, \ldots, \varsigma_n\}$ l'ensemble des nœuds de type *feuille* (ou *critère*) de A et $P = \{\rho_1, \ldots, \rho_n\}$ l'ensemble des nœuds de type *profil* de A, ces concepts sont les suivants :

1. $\forall \varsigma_i \in C$, $\varsigma_i \in N$;

2. $\forall \rho_i \in P$, $\rho_i \in N$;

3. $A[\eta]$ est la sous-arborescence de A ayant pour racine le nœud η de A ;

4. $fils[\eta]$ est la liste des fils du nœud η ;

5. $pere[\eta]$ est un nœud représentant le père du nœud η ;

6. $nom[\eta]$ est le nom associé au nœud η ;

7. $anc[\eta,A]$ est l'ensemble des ancêtres du nœud η dans A, η non inclu ;

8. $chm[\eta,A]$ est la liste des nœuds qui composent le chemin allant de la racine de A vers le nœud η de A, η inclu. Ainsi, $chm[\eta,A] = \eta + anc[\eta,A]$;

9. $val[\varsigma,A]$ est la valeur du critère ς de l'arborescence A ;

10. $typeNoeud[\eta]$ est le type du nœud η qui peut être : catégorie, critère ou profil ;

11. $typeProfil[\rho]$ est le type du profil ρ qui peut être : profil d'information (document, collection ou partie de document, etc.), profil négatif, profil positif, profil court terme, profil long terme, profil individuel, profil de groupe ;

12. $profondeur[\eta,A]$ est la profondeur du nœud η dans l'arbre A.

La transformation de l'arborescence d'un profil ρ composé d'autres profils en une arborescence composée uniquement de catégories et de critères

est utile, du fait de la simplicité de la structure résultante, pour faciliter l'usage de profils par les processus d'accès à l'information. L'appariement de profils se fait toujours sur les critères car ce sont ces critères qui contiennent le contenu des profils. Ainsi, si on veut comparer un profil ρ_a composé de profils à un autre profil ρ_b éventuellement aussi composé de profils, cette comparaison serait plus facile si on a une représentation de ρ_a et de ρ_b sous forme de catégories et de critères uniquement. Dans le cas contraire, il faudrait définir une stratégie de comparaison qui tienne compte des structures particulières de ρ_a et ρ_b ce qui peut s'avérer très difficile (*cf.* FIG. 2.8 et FIG. 2.9) si on a plusieurs nœuds ou feuilles de même sémantique. *Nous partons du postulat selon lequel, deux nœuds ou feuilles ont la même sémantique s'ils ont le même nom et les mêmes ancêtres de type «catégorie»*. Notons cependant que nous considérons les critères *mots clés* et *centres d'intérêt* comme étant synonymes.

Pour transformer la structure et modifier le contenu de profils composés d'autres profils, nous avons défini un algorithme que nous présentons dans la section suivante.

2.3.2.1 Méthode de transformation de profils composés d'autres profils

L'algorithme de transformation de la structure et de modification du contenu de profils composé d'autres profils est défini dans le tableau TAB. 2.8. Cet algorithme (*cf.* TAB. 2.8) analyse chaque critère ou feuille d'un arbre. La transformation se fait de façon ascendante. On recherche d'abord la profondeur maximale (notée m) des profils de la copie A de l'arborescence à transformer. Ensuite, pour chaque profil de cette profondeur, on vérifie son type et celui du profil père et on applique le cas correspondant. Pour un cas donné et pour chaque critère, les étapes *recherche de chemins* et *comparaisons de chemins* sont d'abord effectuées :

- *recherche de chemins* : on va rechercher le chemin du critère ς_i dans $A[\rho_i]$ noté $chm(\varsigma_i, A[\rho_i])$, ainsi que le chemin de ce critère dans l'arborescence $A[pere[\rho_i]]$ noté $chm(\varsigma_i, A[pere[\rho_i]])$. Il s'agit d'obtenir les chemins menant au critère ς_i dans les deux arborescences : $A[pere[\rho_i]]$ et $A[\rho_i]$. Ceci permet également de s'assurer que le critère ς_i, dans les deux arborescences, a la même sémantique ;
- *comparaison des chemins* : on compare les chemins de ς_i dans $A[pere[\rho_i]])$ avec celui dans $A[\rho_i]$ de la façon suivante :

$$(chm(\varsigma_i, A[\rho_i]) - \rho_i) = (chm(\varsigma_i, A[pere[\rho_i]]) - pere[\rho_i])$$

S'il n'existe pas de chemin remplissant cette contrainte alors on procède à la *création du chemin* $(chm(\varsigma_i, A[\rho_i]) - \rho_i)$, en partant du nœud $pere[\rho_i]$.

Ensuite, des traitements spécifiques sont appliqués pour chaque critère (mots clés, langue, etc.) d'un profil de l'arborescence. Si pour un critère donné le principe à suivre n'est pas détaillé, le critère disparaît dans la structure résultante.

Dans cet algorithme (*cf.* TAB. 2.8), nous avons identifié différents cas particuliers de profils à traiter spécifiquement :

Cas 1. *profil quelconque composé de profils d'informations* : On définit l'ensemble P_{info} des profils d'informations de même profondeur m donnée. Ensuite, des traitements spécifiques sont effectués pour chaque critère de l'ensemble des profils de P_{info} :

a. $nom[\varsigma_i]$ = *mots clés* : le critère «mots clés» étant généralement multi-valué, on va d'abord *modifier la valeur du poids* de chaque mot clé de $A[\rho_i]$. Soit $w_{m_{t,\rho_i}}$ ce nouveau poids, il est obtenu en divisant le poids w_{t,ρ_i}, du mot clé t dans le profil ρ_i, par le rapport entre la taille du profil $A[\rho_i]$ (notée $taille_{\rho_i}$) et la taille moyenne des profils de même profondeur que ρ_i (notée $taille_{moy_{\rho_i}}$). Le but est de rendre le poids d'un mot clé inversement proportionnel à la taille normalisée de l'élément qu'il décrit afin de ne pas favoriser les poids des mots clés issus de profils de grande taille par rapport à ceux issus de profils de petite taille. Ainsi, on a :

$$w_{m_{t,\rho_i}} = \frac{w_{t,\rho_i}}{taille_{\rho_i}/taille_{moy_{\rho_i}}}$$

Ensuite, on va procéder à l'*insertion* des différents mots-clés de $A[\rho_i]$ dans les mots clés de $A[pere[\rho_i]]$. Le principe de l'insertion pour chaque mot clé, noté t, de $A[\rho_i]$ est le suivant :

- si le mot clé t n'existe pas dans le critère «mots clés» de $A[pere[\rho_i]]$, alors on ajoute t avec son nouveau poids ;
- si le mot clé t existe dans le critère «mots clés» de $A[pere[\rho_i]]$ alors le poids du mot clé t dans $A[pere[\rho_i]]$ est remplacé par la somme entre le poids de t dans $A[pere[\rho_i]]$ et le poids modifié de t dans $A[\rho_i]$, noté $w_{m_{t,\rho_i}}$. Ainsi, le nouveau poids $w_{n_{t,pere[\rho_i]}}$ du mot clé t dans $A[pere[\rho_i]]$ est donc calculé comme suit :

$$w_{n_{t,pere[\rho_i]}} = w_{t,pere[\rho_i]} + w_{m_{t,\rho_i}} = w_{t,pere[\rho_i]} + \frac{w_{t,\rho_i}}{\frac{taille_{\rho_i}}{taille_{moy_{\rho_i}}}} \quad (2.3)$$

Où :

$taille_{\rho_i}$ est la taille du profil ρ_i ;

$taille_{moy_{\rho_i}}$ est la taille moyenne des profils de même profondeur que ρ_i ;

w_{t,ρ_i} est le poids du mot clé t dans le profil ρ_i ;

$w_{t,pere[\rho_i]}$ est le poids du mot clé t dans le profil $pere[\rho_i]$.

Le poids w_{t,ρ_i} est divisé par le rapport $\frac{taille_{\rho_i}}{taille_{moy_{\rho_i}}}$, et non pas par $taille_{\rho_i}$, afin d'éviter une trop grande distorsion entre le poids initial du terme t et son nouveau poids. Le rapport $\frac{taille_{\rho_i}}{taille_{moy_{\rho_i}}}$ permet de normaliser les tailles de profils, d'une profondeur donnée, dans l'intervalle $[\frac{taille_{min_{\rho_i}}}{taille_{moy_{\rho_i}}}, \frac{taille_{max_{\rho_i}}}{taille_{moy_{\rho_i}}}]$ où $taille_{min_{\rho_i}}$ est la taille minimale des profils de même profondeur que ρ_i et $taille_{max_{\rho_i}}$ la taille maximale de ces profils. Ainsi, $w_{m_{t,\rho_i}}$ est calculé en fonction de la *taille normalisée* des profils de même profondeur dans l'arborescence considérée.

b. $nom[\varsigma_i]$=*taille* et $val[\varsigma_i, pere[\rho_i]]$=*0*: dans $val[\varsigma_i, A[pere[\rho_i]]$, on met la somme des valeurs du critère «taille» de tous les profils de même profondeur que ρ_i ;

c. $nom[\varsigma_i]$=*langue*: dans $val[\varsigma_i, pere[\rho_i]]$, on met la valeur de la langue qui décrit le plus grand nombre de profils de même profondeur que ρ_i.

Cas 2. *profil utilisateur quelconque composé de profils positifs ou négatifs*: des traitements spécifiques sont effectués pour chaque critère dans ce cas. Ainsi, pour le critère *mots clés (ou centres d'intérêts)* par exemple, on considère les poids des mots clés des profils négatifs comme étant négatif et ceux des profils positifs comme étant positifs et on les insère dans le profil père. Le principe d'insertion est le suivant :

- si le mot clé n'existe pas dans le profil père, on l'y insère avec son poids positif ou négatif ;
- si le mot clé existe déjà dans le profil père, on ajoute le poids positif ou négatif du mot clé dans le profil ρ_i au poids du même mot clé dans le profil père. Les différents poids du mot clé t dans ρ_i et $pere[\rho_i]$ peuvent être modulés par des paramètres α et β qui traduisent l'importance de chaque poids. Cette opération va permettre d'augmenter ou de diminuer le poids du mot clé considéré dans le profil père. Notons que α and β sont déterminés par expérimentations.

Cas 3. *profil long terme composé de profils court terme*: On définit l'ensemble P_{ct} des profils court terme qui composent le profil long terme. Afin de pouvoir appliquer la transformation, il faut vérifier que le nombre de profils court terme composant le profil long terme est supérieur ou égal à une valeur entière n fixée par l'application. Si c'est le cas, on peut définir un sous-ensemble de P_{ct} que l'on note P'_{ct} qui contient les n derniers profils court terme de l'usager. Ensuite, des traitements spécifiques sont effectués pour chaque critère de l'ensemble des profils de P'_{ct}.

Ainsi, pour le critère *«mots clés» (ou centres d'intérêts)* par exemple, on détermine tout d'abord l'ensemble des mots clés des profils court terme de P'_{ct}. Pour chaque mot clé de cet ensemble, on calcule un poids w_{m_t} qui décrit à quel point un terme t est récurrent dans l'ensemble des profils de P'_{ct}. Ce poids peut-être calculé comme suit :

$$w_{m_t} = \frac{\sum_{i=1}^{cardinal(P'_{ct})} w_{t,\rho_i}}{cardinal(P'_{ct})}$$

Si la valeur absolue de ce poids w_{m_t} est supérieure à un seuil donné alors le mot clé est inséré dans le profil long terme (profil père). Le principe d'insertion est le suivant :

- si le mot clé n'existe pas dans le profil père, on l'y insère avec son poids w_{m_t} (qui peut être positif ou négatif) ;
- si le mot clé existe déjà dans le profil père, on ajoute au poids du même mot clé dans le profil père, le poids w_{m_t}. Le nouveau poids w_{n_t} du mot clé t dans le profil long terme, se calcule comme suit :

$$w_{n_{t,pere[\rho_i]}} = w_{t,pere[\rho_i]} + \frac{\sum_{i=1}^{cardinal(P'_{ct})} w_{t,\rho_i}}{cardinal(P'_{ct})} \tag{2.4}$$

Où :

$cardinal(P'_{ct})$ est le nombre de profils de l'ensemble P'_{ct} ;

w_{t,ρ_i} est le poids du mot clé t dans le profil ρ_i ;

$w_{t,pere[\rho_i]}$ est le poids du mot clé t dans le profil $pere[\rho_i]$.

Notons qu'un mot clé non pondéré est considéré comme ayant un poids de 1. De plus, le seuil de sélection d'un mot clé afin de l'intégrer au profil long terme, peut être déterminé par expérimentations en analysant la distribution des poids w_{m_t} des mots clés de P'_{ct}.

Cas 4. *profil de groupe d'utilisateurs composé de profils individuels d'utilisateurs* : On définit l'ensemble P_{ind} des profils individuels qui composent le profil de groupe. Ensuite, des traitements spécifiques sont effectués pour chaque critère de l'ensemble des profils de P_{ind}.
Ainsi, pour le critère *mots clés (ou centres d'intérêts)* par exemple, on détermine tout d'abord l'ensemble des mots clés des profils individuels qui composent le profil de groupe. Pour chaque mot clé t de cet ensemble, on calcule un poids w_{m_t} qui décrit à quel point un mot clé est commun à l'ensemble des profils de P_{ind}. Ce poids peut-être calculé comme suit :

$$w_{m_t} = \frac{\sum_{i=1}^{cardinal(P_{ind})} w_{t,\rho_i}}{cardinal(P_{ind})}$$

Par la suite, on définit un seuil (que l'on détermine en analysant l'ensemble des poids w_{m_t} des mots clés ainsi obtenus) et uniquement les

mots-clés dont la valeur absolue du poids w_{m_t} est supérieure au seuil fixé, sont retenus pour le profil de groupe. Le principe d'insertion dans le profil de groupe, des mots clés sélectionnés, est le même que pour les profils long terme. Notons qu'il peut arriver qu'aucun mot clé ne soit retenu pour le profil de groupe. Ceci arrive lorsque les profils individuels sont très différents.

Notons qu'à la fin des différentes transformations, on va procéder à la suppression des *mots clés* ou *centres d'intérêt* de la racine de A, dont le poids est inférieur à un seuil θ donné. θ est déterminé par expérimentations, en analysant la distribution des poids des mots clés ou centres d'intérêt résultants des différentes transformations. Le but de cette opération est d'éliminer les termes qui auront un poids trop faible ;

Par ailleurs, la méthode de transformation de profil proposée va également permettre l'*évolutivité* du contenu des profils utilisateurs dans le temps. Ceci va se faire à travers la modification du profil long terme d'un utilisateur à partir des profils court terme de ce dernier.

2.3.2.2 Illustration de la méthode de transformation de profils

Les figures Fig 2.10 et Fig. 2.11 illustrent des exemples d'exécution de l'algorithme du tableau Tab. 2.8 pour la transformation de profils d'informations mises à disposition et de profils d'usagers respectivement.

2.4 Conclusion

Dans ce chapitre, nous présentons une architecture de recherche et de recommandation à base de profils ré-utlisables, multi-facettes, adaptables à différents contextes et évolutifs qui sont dérivés d'un modèle générique. La généricité de l'approche proposée garantit une coopération et une complémentarité maximales entre tout élément interagissant dans le cadre d'un même processus. Nous montrons également que la combinaison de différents appariements entre profils, au travers d'une méthode que nous définissons, permet d'améliorer théoriquement la qualité des résultats renvoyés à un individu. Enfin, nous proposons également une méthode de transformation de la structure et du contenu des profils composés d'autres profils afin de faciliter leur appariement.

L'architecture que nous proposons peut servir de base pour toute conception d'applications d'accès personnalisé à l'information : Recherche d'Information, Filtrage d'Information, aide à la découverte d'informations, etc. L'utilisation de différents appariements permet de fournir des réponses adaptées à chaque usager ou groupe d'usagers. L'objectif en premier lieu a été de travailler sur la personnalisation et l'adaptativité dans le cadre de la re-

cherche et de la recommandation d'information. Il reste néanmoins à vérifier, par expérimentations, l'impact réel de celles-ci sur les résultats restitués.

`Algorithme de transformation de profil`

1. Détermination de l'ensemble P, d'une copie A, de l'arbre à transformer
2. Construction ascendante
   2.1 Recherche de la profondeur maximale des profils de A : notée m
   2.2 Passage des profils de profondeur m aux profils de profondeur $m-1$
   Répéter
      Pour chaque profil ρ_i de P de profondeur m faire
         Cas ρ_i
         2.2.1 Cas 1
            $typeProfil[\rho_i]$=*profil d'information :*
               Pour chaque critère ς_i de $A[\rho_i]$ faire
                  Traitements particuliers pour chaque critère dans ce cas
               Fin pour
         2.2.2 Cas 2
            $typeProfil[\rho_i]$=*profil négatif :*
               Pour chaque critère ς_i de $A[\rho_i]$ faire
                  Traitements particuliers pour chaque critère dans ce cas
               Fin pour ;
         2.2.3 Cas 3
            $typeProfil[\rho_i]$=*profil court terme*
            ET $typeProfil[pere[\rho_i]]$=*profil long terme* :
               Détermination de l'ensemble P_{ct} composé de profils ρ tels que :
               ($profondeur[\rho,A] = profondeur[\rho_i,A]$ *ET* $pere[\rho] = pere[\rho_i]$
               ET $typeProfil[\rho]$=*profil court terme*)
               Si $card[P_{ct}] > \theta$ alors
                  Pour chaque critère ς_i de $A[\rho_i]$ faire
                     Traitements particuliers pour chaque critère dans ce cas
                  Fin pour
               Fin si
               Suppression des sous-arbres $A[\rho]$ de A, tels que : $\rho \in P_{ct}$
               ET suppression des profils de P_{ct} dans l'ensemble P ;
         2.2.4 Cas 4
            $typeProfil[\rho_i]$=*profil individuel*
            ET $typeProfil[pere[\rho_i]]$=*profil de groupe :*
               Pour chaque critère ς_i de $A[\rho_i]$ faire
                  Traitements particuliers pour chaque critère dans ce cas
               Fin pour
               Suppression des sous-arbres $A[\rho]$ de A, tels que :
               ($profondeur[\rho,A] = profondeur[\rho_i,A]$ *ET* $pere[\rho] = pere[\rho_i]$
               ET $typeProfil[\rho]$=*profil individuel*)
               ET suppression de ces profils ρ dans l'ensemble P ;
         Fin cas
         Suppression de $A[\rho_i]$ dans l'arbre A
      Fin pour
      $m \leftarrow m-1$
   Jusqu'à $(m = 0)$

TAB. 2.8 – *Algorithme de transformation d'un profil composé d'autres profils*

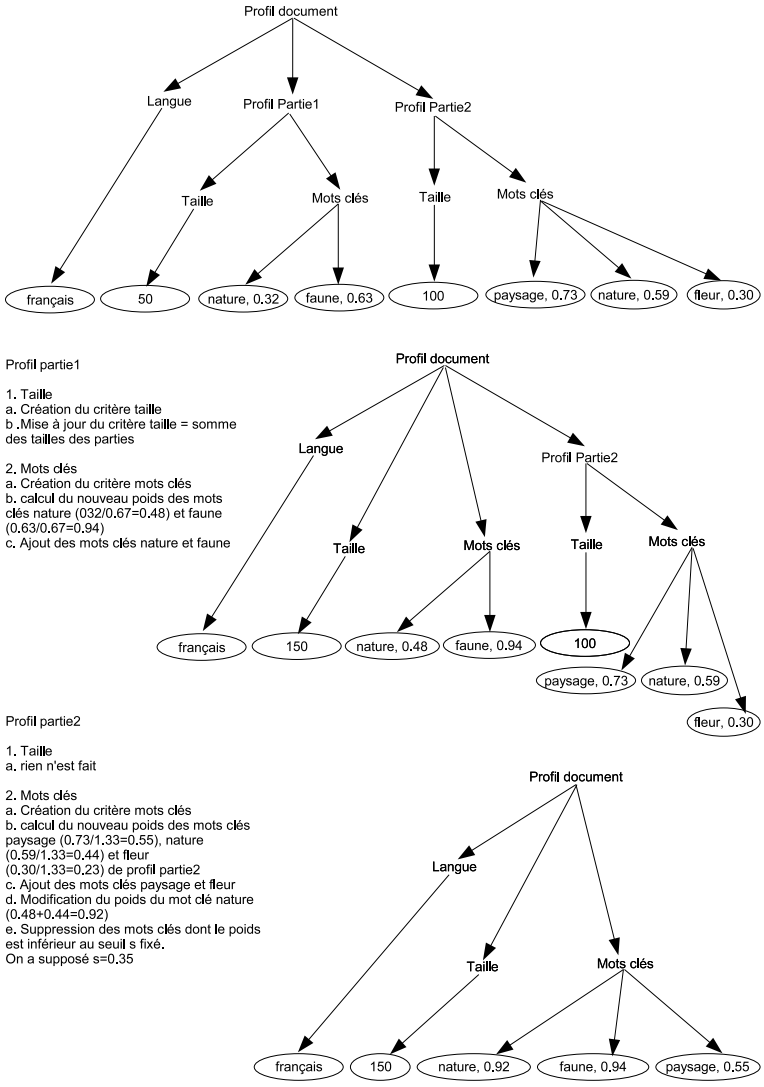


FIG. 2.10 – *Illustration de l'algorithme de transformation des profils des informations mises à disposition*

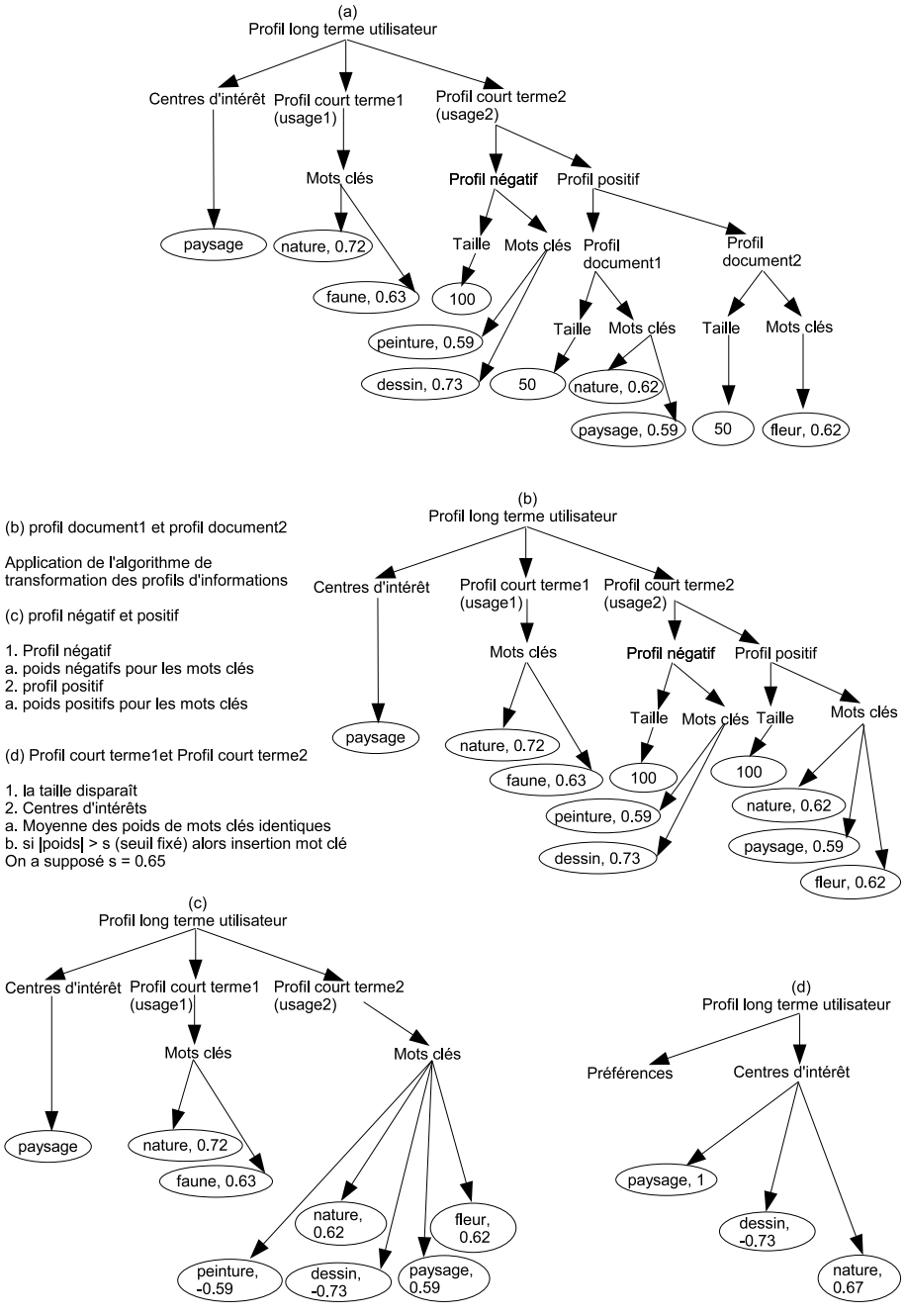


FIG. 2.11 – *Illustration de l'algorithme de transformation de profils d'usagers*

Conclusion générale et perspectives

L'augmentation continue du volume d'information mis à disposition rend crucial le développement d'outils pour aider les utilisateurs à retrouver ce qu'ils recherchent. L'efficacité de ces outils se mesurent à leur capacité à s'adapter aux particularités de chaque usager pour satisfaire ses besoins. Nous avons proposé, à cette fin, une approche à base de profils pour améliorer la restitution personnalisée de résultats. Notre approche comprend trois aspects : *un cadre générique* composé d'une architecture de recherche et de recommandation permettant une exploitation optimale de la complémentarité entre différents profils qui interviennent dans les processus d'accès à l'information et d'un modèle générique de profil pour la description de tout type de profils ; *une méthode d'appariement de profils* qui combine différents appariements afin de calculer un degré de ressemblance d'une information à un usager pour un ensemble de critères donné ; *un algorithme de transformation de profils* qui utilisent à la fois la structure et le contenu de ces profils afin de faciliter l'appariement de profils de structure complexe (contenant plusieurs nœuds ou feuilles identiques).

En terme de perspectives à notre travail nous comptons :

1. *valider nos propositions par des expérimentations et des tests* sur une application de recherche et/ou de recommandation. Cette validation sera basée sur une évaluation qualitative : de la méthode d'appariement de profils définie et de la méthode de transformation de profils proposée. Pour l'appariement de profils, il faudra évaluer l'influence de l'usage de différents profils et de différents critères de profils sur l'ensemble des résultats renvoyés aux usagers. Pour la transformation de profils, on devra faire un étude comparative entre le profil construit à partir des informations brutes disponibles et le profil construit à partir d'autres profils existants afin de mesurer la qualité du profil obtenu après transformation. On devra également comparer les résultats (de recherche par exemple) obtenus avec des profils construits à partir des informations brutes à ceux obtenus avec des profils transformés. Dans un premier temps, nous allons effectuer nos tests sur la collection CLEF.

Cette collection est : multilingue, composée de plusieurs collections. Nous utiliserons quatre types de profils : profils de collection, profils de document, profils de granules de document et profils d'utilisateur individuel. Ces profils seront décrit par les critères : mots-clés ou centres d'intérêts, langue, taille et date.

Par la suite, nous comptons mettre sur pied une plate-forme de recherche et de recommandation à base de profils pour une communauté d'utilisateurs (comme une équipe de recherche). Nous définirons plusieurs usagers et groupe d'usagers, des collections de documents très variées avec des profils ayant une taxinomie assez riche. Les actions des usagers pendant leurs phases de recherches (jugements, sauvegardes, etc.) seront enregistrées et utilisées pour leur renvoyer des informations. On pourra sur cette plate-forme mettre en œuvre différents processus d'accès à l'information ;

2. *proposer un modèle de profil intégrant les aspects : sémantique des critères de profils et ontologie* [Sow00a] [Sow00b]. Si les profils doivent pouvoir être partagés entre différentes applications, il se posera le problème de correspondance entre profils décrits par des taxinomies différentes. Il est donc nécessaire de définir pour chaque critère une sémantique non ambiguë d'autant plus que les méthodes proposées, pour l'appariement et la transformation de profils, s'appliquent critères par critères. De plus, les critères de profils différents qui sont utilisés pour l'appariement ou la transformation de structure et de contenu doivent représenter la même unité sémantique ou doivent avoir une sémantique compatible. Notons que des critères de noms différents peuvent représenter la même unité sémantique comme : volume et taille. De même, des critères de même nom peuvent avoir une sémantique non compatible : par exemple, une taille représentée en nombre de termes et une taille représentée par catégories (court, normal, long). Afin, de pouvoir gérer ce type de problème on a besoin d'effectuer un certains nombre de déductions basées sur la sémantique de chaque critère. Avec une *approche orientée base de données* comme celle que nous avons utilisée jusqu'à présent (UML), cela s'avère assez difficile. Nous comptons donc, pour cela, étendre notre approche aux *approches orientées logique de description* qui offrent un pouvoir de raisonnement plus important [CPSV03] comme OWL [1] (*Ontology Web Language*). Avec une approche orientée logique de description, on pourra déduire plus facilement (à partir de règles pré-définies) les critères de même sémantique et donc compatibles pour effectuer un appariement dans notre méthode d'appariement de profils ou utilisables pour déduire la valeur d'autres critères dans nos méthodes de transformation de structure et de contenu de profils ;

1. *cf.* http://www.w3.org/2004/OWL/

3. *proposer un modèle de profil pour le respect de données privées*: le problème du respect de la vie privée de l'utilisateur est très important surtout lorsque l'on traite des aspects de personnalisation. Aujourd'hui, il y a une véritable dérive de la divulgation des données privées. Lorsqu'un internaute visite plusieurs sites Internet, ses actions sont suivies à son insu, par exemple au travers de *cookies*[2] où des informations privées le concernant peuvent être recueillies. Par ailleurs, il est nécessaire de connaître les données privées des utilisateurs pour faire de la personnalisation. Il y a donc un conflit entre les besoins de personnalisation, technique qui utilise des données privées de utilisateur, et le besoin de confidentialité de cet utilisateur. Un autre problème sous-jacent au respect des données privées est de pouvoir déterminer ce qu'est une information privée. Selon la sensibilité de l'utilisateur, la barrière ne se place pas au même niveau. Une des solutions consisterait alors à considérer que toute information concernant l'utilisateur est privée. Dans ce cas, il faudrait déterminer si on peut divulguer une information ou pas. La configuration idéale serait de concevoir des systèmes de personnalisation qui tiennent compte des soucis d'intimité des usagers et des lois de sécurité qui sont appliquées dans différents pays [Kob02].

En définitive, notre sujet de thèse propose des modèles de structuration d'informations manipulées (informations mises à disposition et usagers de ces dernières) par les processus d'accès à l'information. De plus, nous nous intéressons aussi aux méthodes d'exploitation (appariement et transformation de structure et de contenu) de ces modèles pour la restitution personnalisée d'informations.

2. un cookie est ensemble d'informations sur un utilisateur (nom, mot de passe, parties du site visitées, etc.) qu'un serveur Web donne à un navigateur Web. Ces informations sont sauvegardées sous forme de fichiers texte, sur le serveur et sur la machine de l'utilisateur

Bibliographie

[AH01] A.T. Arampatzis and A. Van Hameren. The score-distributional threshold optimization for adaptive binary classification tasks. *In proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

[Ame01] Y. Amerouali. Métadonnées basées sur l'association d'éléments de description de ressources et d'éléments de profil d'utilisateur. *Thèse de Doctorat en sciences de l'information et de la communication*, Mai 2001. Université Claude Bernard Lyon I - Enssib (France).

[Amm03] A. Ben Ammar. Profils en recherche d'information : définition, exploitation et adaptation. *Thèse de Doctorat en Informatique*, Avril 2003. Université Paul Sabatier Toulouse 3 - Laboratoire Irit (France).

[BAGB03] M. Baziz, N. Aussenac-Gilles, and M. Boughanem. Exploitation de liens sémantiques pour l'expansion de requêtes dans un système de recherche d'information. *21ième Congrès Inforsid*, pages 121–134, 2003.

[BBB03a] J.C. Bottraud, G. Bisson, and M.F. Bruandet. An adaptative information research personnal assistant. *In proceedings of Workshop AI2IA (Artificial Intelligence, Information Access and Mobile Computing) IJCAI'03*, 2003.

[BBB03b] J.C. Bottraud, G. Bisson, and M.F. Bruandet. Apprentissage de profils pour un agent de recherche d'information. *Conférence d'Apprentissage (CAp03)*, 2003.

[BBB04] J.C. Bottraud, G. Bisson, and M.F. Bruandet. Expansion de requêtes par apprentissage automatique dans un assistant pour la recherche d'information. *COnférence en Recherche d'Information et Applications (CORIA'04)*, pages 89–105, 2004.

[BC92] N.J. Belkin and W.B. Croft. Information filtering and information retrieval : Two sides of the same coin? *Communications of the ACM, Information Filtering*, 35(12):29–38, 1992.

[BCSD99] M. Boughanem, C. Chrisment, and C. Soulé-Dupuy. Query modification based on relevance backpropagation in adhoc environment. elsevier science. *Information Processing & Management Journal*, 35:121–139, 1999.

[BE02] L. Berti-Equille. Annotation et recommandation collaboratives de documents selon leur qualité. *Ingénierie des Systèmes d'Information, Recherche et Filtrage d'information. RSTI serie ISI-NIS*, 7(2):125–155, 2002.

[BE03] L. Berti-Equille. Quality-based recommendation of xml documents. *Journal of Digital Information Management*, 1(3):117–128, 2003.

[BHM01] A. Benammar, G. Hubert, and J. Mothe. Reformulation automatique des profils utilisant un ensemble local de documents. *Actes de la Conférence Veille Stratégique, Scientifique et Technologique (VSST'01)*, pages 321–329, 2001.

[BJMSD00] M. Boughanem, C. Julien, J. Mothe, and C. Soulé-Dupuy. Mercure at TREC8 : adhoc, filtering, Web and CLIR tasks. *In Harman D.K (ed.) NIST Gaithersburg Proceedings of the eight International Conference on Text REtrieval TREC8*, November 2000. Maryland (USA).

[BMRM96] E. Bloedorn, I. Mani, T. Richard, and McMillan. Representational issues in machine learning of user profiles. *Notes of the AAAI Spring Symposium on Machine Learning in Information Access*, 1996. Stanford University.

[Bru95] P. Brusilovsky. Systèmes tuteurs intelligents pour le World-Wide Web. *In R. Holzapfel (ed.) Proceedings of Third International WWW Conference*, pages 42–45, 1995.

[BS97] M. Balabanovic and Y. Shoham. Fab : Content-based, collaborative recommendations. *Communications of the ACM*, 40(3):66–72, 1997.

[BTT04] M. Boughanem, M. Tmar, and H. Tebri. Filtrage d'information. *Méthodes avancées pour les systèmes de recherche d'informations, Hermes Sciences Publication, Lavoisier*, pages 137–161, 2004. Chapitre 7.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. First edition, Addison Wesley, ISBN 0-201-39829-X, 1999.

[Cal98] J. Callan. Learning while filtering documents. *In Proceedings of the twenty first Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 224–231, 1998.

[CB02] N. Correia and M. Boavida. Towards an integrated personalization framework : A taxonomy and work proposals. *In Procee-*

dings of the Workshop on Personalization Techniques in Electronic Publishing on the Web : Trends and Perspectives, May 2002. Malaga (Spain).

[Che02] M. Chevalier. Interface intelligente pour l'aide à la recherche d'information sur le web. *Thèse de Doctorat en Informatique*, Décembre 2002. Université Paul Sabatier Toulouse 3 - Laboratoire IRIT (France).

[CKK02] Y.H. Cho, J. Kyeong, and S.H. Kim. A personalized recommender system based on web usage mining and decision tree induction. *Expert System with Applications*, 23(3):329–342, 2002.

[CP43] W.S Mc Cullogh and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, (5):115–133, 1943.

[CPSV03] N. Cullot, C. Parent, S. Spaccapietra, and C. Vangenot. Des SIG aux ontologies géographiques. *Revue internationale de géomatique*, 2003.

[Cra80] J.B. Crampes. Aide à l'interrogation d'un dictionnaire de données. *RAIRO Informatique/Computer Science*, 14(1):86–95, 1980.

[CS00] P. Cotter and B. Smith. PTV : Intelligent personalised TV guides. *American Association for Artificial Intelligence (AAAI)*, 2000. www.aaai.org.

[CSDT04] M. Chevalier, C. Soulé-Dupuy, and P. L. Tchienehom. A profile-based architecture for a flexible and personalized information access. *IADIS International Conference*, October 2004. to appear.

[Dai98] B. Daille. Traitement des corpus, application à l'extraction automatique de terminologie. *La structure du lexique, Ateliers en morphologie. Paul Boucher (ed.) les ateliers* COLEX *(Centre Ouest Lexique)*, pages 159–164, 1998. Université de Nantes.

[DDF90] S. Deerwester, S. Dumais, and G.W. Furnas. Indexing by latent semantic indexing. *Journal of American Society for Information Science*, 41:391–407, 1990.

[Dum94] S.T. Dumais. Latent Semantic Indexing (LSI) : TREC-3 report. *In Proceedings of the Third Text Retrieval Conference (TREC-3). NIST Special Publication*, 1994.

[Dun00] M.D. Dunlop. Development and evaluation of clustering techniques for finding people. *In Proceedings of the Third International Conference on Practical Aspects of Knowledge Management (PAKM2000)*, October 2000. Basel (Switzerland).

[FY92] W.B Frakes and R.B. Yates. *Information Retrieval : Data Structures & Algorithms*. Ed. Addison Wesley Publishing Company, ISBN 0-134-63837-9, 1992.

[GGMT99] L. Gravano, H. Garcia-Molina, and A. Tomasic. Gloss : Text-source discovery over the internet. *ACM transactions on Database systems*, 24(2):229–264, 1999.

[GNOT92] D. Goldberg, D. Nichols, B.M. Oki, and D. Terry. Using collaborative filtering to weave an information tapestry. *Communications of the ACM, Information Filtering*, 35(12):61–70, 1992.

[GSK+99] N. Good, J. Schafer, J. Konstan, A. Borchers, B. Sarwar, J. Herlocker, and J. Riedl. Combining collaborative filtering with personal agents for better recommendations. *In Proceedings of AAAI. AAAI Press*, 35:439–446, 1999.

[HKNH00] K. Hoashi, M. Kazunori, I. Naomi, and K. Hashimoto. Document filtering method using non relevant information profile. *In Proceedings of the twenty third Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Distributed Retrieval*, pages 176–183, 2000.

[Hol75] J. Holland. Adaptation in natural and artificial systems. *Ann Arbor. University of Michingan Press*, 1975. Michingan.

[Jac98] M. Jaczynski. Modèle et plate-forme à objets pour l'indexation des cas par situations comportementales : application à l'assistance à la navigation sur le Web. *Thèse de doctorat en Informatique, Université de Nice*, 1998.

[Jon72] K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–20, 1972.

[JRP01] L. Jeribi, B. Rumpler, and J.M. Pinon. Système d'aide à la recherche et à l'interrogation de bases documentaires, fondé sur la réutilisation d'expériences. *19ième Congrès d'Inforsid*, pages 443–463, 2001.

[Kay95] J. Kay. The um toolkit for reusable, long term user models. *User Modeling and User-Adapted Interaction*, 4(3):149–196, 1995.

[KGB98] C. Kurzke, M. Galle, and M. Bathelt. WebAssist : a user profiles specific information retrieval assistant. *Computer Networks*, 30(1-7):654–655, 1998.

[KMM+97] J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, and J. Riedl. Grouplens : Applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, March 1997.

[Kob01] A. Kobsa. Generic user modeling systems. *User Modeling and User-Adapted Interaction, Kluwer Academic Publishers*, 11:49–63, 2001. Printed in the Netherlands.

[Kob02] A. Kobsa. Personalized hypermedia and international privacy. *Communications of the ACM, Special Isuue on Adaptive Web-Based Systems and Adaptive Hypermedia, ACM Press*, 2002.

[Kor97] R.R. Korfhage. *Information storage and retrieval.* Wiley computer publishing, ISBN 0-471-14-338-3, 1997.

[Kru97] B. Krulwich. Lifestyle finder : Intelligent user profiling using large-scale demographic data. *AI Magazine*, 18(2):37–45, 1997.

[KV02] M. Karamuftuoglu and F. Venuti. Okapi in tips : The changing context of information retrieval. *In Proceedings of the Workshop on Personalization Techniques in Electronic Publishing on the Web : Trends and Perspectives*, May 2002. Malaga (Spain).

[LC96] A. V. Leouski and W. B. Croft. An evaluation of techniques for clustering search results. *Technical Report IR-76*, 1996.

[LC99] S. Lainé-Cruzel. Profildoc : Filtrer une information exploitable. *Bulletin des Bibliothèques de France*, (5):60–65, 1999. http://www.enssib.fr/bbf/bbf-99-5/10_lainecruzel.pdf.

[Lie95] H. Lieberman. Letizia : An agent that assists web browsing. *In Proceedings of the IJCAI'95*, pages 924–929, 1995.

[Luh58] H.P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[Mar02] S. Marel. L'appariement de profils. *Diplôme de Recherche Technologique (DRT)*, Décembre 2002. Université Paul Sabatier Toulouse 3 - Laboratoire IRIT et entité Hewlett Packard Laboratoire de Grenoble (France).

[Mit97] T. Mitchell. *Machine Learning.* McGraw Hill, 1997.

[Mla96] D. Mladenic. Personal WebWatcher : design and implementation. *Technical Report IJS-DP-7472*, 1996. Department of Intelligent Systems, Slovenia : J. Stefan Institute.

[MLR03] M. Montaner, B. Lopez, and J.L.D.L Rosa. A taxonomy of recommender agents on the internet. *Artificial Intelligence Review. Kluwer Academic Publishers*, 19:285–330, 2003.

[MT02a] S. Mizzaro and C. Tasso. Ephemeral and persistent personalization in adaptive information access to scholarly publications on the Web. *In Paul De Bra, Peter Brusilovsky, and Ricardo Conejo editors, Adaptive Hypermedia and Adaptive Web-Based Systems, Second International Conference AH 2002, LNCS 2347*, pages 306–316, May 2002. ISBN 3-540-43737-1.

[MT02b] S. Mizzaro and C. Tasso. Personalization techniques in the tips project : The cognitive filtering module and the information retrieval assistant. *In Proceedings of the Workshop on Personalization Techniques in Electronic Publishing on the Web : Trends and Perspectives*, May 2002. Malaga (Spain).

[Paz99] M. Pazzani. A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, 1999.

[PFL+02] M. Paternostre, P. Francq, J. Lamoral, D. Wartel, and M. Saerens. Carry, un algorithme de désuffixation pour le français. *Projet Galilei*, Juillet 2002. http://www.galilei.ulb.ac.be/rd/pu_publi/carryfinal.pdf.

[PMB96] M. Pazzani, J. Muramatsu, and D. Billsus. Syskill & Webert : Identifying interesting web sites. *In Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 54–61, 1996.

[Por80] M.F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[PSC+02] J. Pitkow, N. Schutze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search : A contextual computing approach may prove a breakthrough in personalized search efficiency. *Communications of the ACM*, 45(9):50–55, 2002.

[Rij79] C.J V. Rijsbergen. *Information Retrieval*. Second edition, Butterworths, 1979.

[RJ76] S. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, pages 129–146, May-June 1976.

[Roc71] J.J. Rocchio. Relevance feedback in information retrieval. *In Salton Editor, The SMART Retrieval System - Experiments in Automatic Document Processing. Prentice Hall Inc. Englewood Cliffs (NJ)*, pages 313–323, 1971.

[RP97] J. Rucker and M.J. Polanco. Siteseer : Personalized navigation for the web. *Communications of the ACM*, 40(3):73–75, 1997.

[RW97] S. Robertson and S. Walker. On relevance weights with little relevance information. *In J. Belkin, A. Desai Narasimhalu and Peter Willet (ed.) proceedings of the 20th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press*, pages 16–24, May-June 1997.

[RWB99] S. Robertson, S. Walker, and M. Beaulieu. Okapi at trec-7 : Automatic adhoc filtering VLC and interactive track. *Information Processing & Management Journal, Elsevier Science*, pages 130–137, 1999. Gaithersburg (Maryland, USA).

[SB90] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of American Society for Information Science*, 41:288–297, 1990.

[SD01] C. Soulé-Dupuy. Bases d'informations textuelles : des modèles aux applications. *Mémoire d'Habilitation à Diriger des Recherches (HDR) en Informatique*, Décembre 2001. Université Paul Sabatier Toulouse 3 - Laboratoire IRIT (France).

[Sha00] A. Sharma. A generic architecture for user modelling systems and adaptive web services. *Delhi College of Engineering*, 2000. New Delhi.

[SM83] G. Salton and M.J McGill. *Introduction to Modern Information Retrieval*. Second Edition, Mc Graw Hill International Book Company, ISBN 0-07-Y66526-53, 1983.

[SMB97] A. Singhal, M. Mitra, and C. Buckley. Learning routing queries in a query zone. *ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–32, 1997. Philadelphia (USA).

[Sow84] J. Sowa. *Conceptual Structures : information processing in mind and machine*. The System Programming Series, Addison Wesley Publishing Company, 1984.

[Sow00a] J. Sowa. Knowledge representation : Logical, Philosophical and Computational Foundations. *Editor Brooks Cole*, 2000. Pacific Grove (CA).

[Sow00b] J. Sowa. Ontology, Metadata and Semiotics. *In proceedings of the International Conference on Conceptual Structures (ICCS'00)*, 2000. Darmstadt (Germany), http://www.bestweb.net/∼sowa/peirce/ontometa.htm.

[SSH97] B. Shapira, P. Shoval, and U. Hanani. Stereotypes in information filtering systems. *Information Processing & Management*, 33(3):273–287, 1997.

[TC91] H.R Turtle and W.B Croft. Evaluation of inference network-based retrieval model. *ACM Transact. Inf. Syst.*, 9(3):187–222, July 1991.

[Tch04a] P. Tchienehom. Accès à une information pertinente et personnalisée : Approche à base de profils. *Colloque des doctorants Edit'04*, pages 100–104, Février 2004.

[Tch04b] P. Tchienehom. Architecture de recherche et de recommandation d'information à base de profils : définitions, acquisitions et usages de profils. *22ième Congrès National Inforsid'04*, pages 143–159, Mai 2004.

[TJK99] B. Trousse, M. Jaczynski, and R. Kanawati. Une approche fondée sur le raisonnement à partir de cas pour l'aide à la navigation dans un hypermédia. *In Proceedings of Hypertexte & Hypermedia: Products, Tools and Methods (H2PTM'99)*, 1999.

[Tma02] M. Tmar. Modèle auto-adaptatif de filtrage d'information : apprentissage incrémental du profil et de la fonction de décision. *Thèse de Doctorat en Informatique*, 2002. Université Paul Sabatier Toulouse 3 - Laboratoire IRIT (France).

[Voo94] E. Voorhees. Query expansion using lexical-semantic relations. *In Proceedings of the 17th Annual Internatinal ACM-SIGIR*

Conference on Research and Development in Information Retrieval, pages 61–69, 1994. Dublin (Ireland).

[WIY99] D.H. Widyantoro, T.R. Ioerger, and J. Yen. An adaptative algorithm for learning changes in user interests. *In Proceedings of the Eighth International Conference on Information and Knowledge Management (CIKM'99). ACM Press*, pages 405–412, 1999. New York (USA).

[XC96] J. Xu and W.B. Croft. Query expansion using local an global document analysis. *In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, August 1996. Zurich (Switzerland).

[Yag02] R.R. Yager. Fuzzy logic methods in recommender systems. *Fuzzy Sets and Systems. Elsevier*, 136:133–149, 2002.

[Zip49] G.K. Zipf. Human behavior and the principle of least effort. *Ed. Addison Wesley Publishing*, 1949.

[ZP04] J. Zhang and P. Pu. Survey of solving multi-attribute decision problems. *EPFL Technical Report No : IC/2004/54*, June 2004.

Index

J

[Jac98], 29
[JRP01], 5

K

[KV02], 4
[Kay95], 30
[Kob01], 30
[Kob02], 63
[KMM+97], 5, 27
[Kor97], 5
[Kru97], 5
[KGB98], 4

L

[LC99], 10, 12, 29
[LC96], 18
[Lie95], 5, 29
[Luh58], 15

M

[Mar02], 10
[CP43], 20, 25
[Mit97], 20
[MT02b], 11
[MT02a], 11
[Mla96], 5, 29
[MLR03], 5

P

[PFL+02], 15
[Paz99], 6, 7, 30, 32
[PMB96], 5, 12, 29, 30, 32
[PSC+02], 4, 11, 32
[Por80], 14

R

[Rij79], 4, 14
[RJ76], 17
[RW97], 17
[RWB99], 17, 18
[Roc71], 4
[RP97], 5, 34

S

[SD01], 16
[SM83], 14, 22, 23
[SB90], 16
[SSH97], 19
[Sha00], 27, 30
[SMB97], 16, 18
[Sow00a], 62
[Sow00b], 62
[Sow84], 26
[Jon72], 16

T

[Tch04b], 34
[Tch04a], 42
[Tma02], 28
[TJK99], 28
[TC91], 25

V

[Voo94], 4

W

[WIY99], 11, 36

X

[XC96], 4

Y

[Yag02], 7

Z

[ZP04], 31
[Zip49], 15