Recherche d'Information Collaborative

Razan Taher

Université Joseph Fourrier – Grenoble I

Ecole Doctorale : Institut d'Informatique et Mathématiques Appliquées de Grenoble, IMAG

Doctorat de l'Université Joseph Fourrier

Discipline : Informatique

Razan TAHER

Recherche d'Information Collaborative

Directeur : Mme. Marie-France BRUANDET

5-Mars-2004

Jury

Président :	M. Jean CAELEN
Rapporteurs :	M. Jean-Marie PINON
	Mme. Josiane MOTHE
Examinateur :	M. Michel LEONARD
Invité :	M. Xavier ROUSSET DE PINA

Je tiens à remercier :

M. Jean CAELEN professeur de l'université Joseph Fourrier à Grenoble et directeur du laboratoire CLIPS, qui m'a fait l'honneur de présider ce jury.

Mme. Josiane MOTHE professeur à l'université IUFM de Toulouse, d'avoir bien voulu rapporter ce travail et pour ses remarques pertinentes.

M. Jean-Marie PINON directeur du département d'informatique et professeur de l'INSA à Lyon pour l'intérêt qu'il a manifesté pour ce travail et sa volonté de le rapporter.

M. Michel LEONARD professeur à l'université de Genève pour son aimable participation à ce jury.

M. Xavier ROUSSET DE PINA professeur de l'INRIA à Grenoble pour avoir accepté de participer au jury et pour ses conseils, ses aides et son encouragement qui m'ont été très précieux.

Mme. Marie-France BRUANDET, professeur de l'université Joseph Fourrier à Grenoble de diriger ce travail et de m'avoir encadré durant ces années de thèse, pour ses conseils directives, et ses aides à achever ce travail dans un milieu plein de respect.

M. Jean-Christophe BOTTRAUD pour sa gentillesse et sa générosité de partager ses connaissances et pour son aide qui m'ont permis de surmonter certaines difficultés rencontrées.

M. Quoc HO-BAO avec qui j'ai partagé le bureau durant la thèse pour son amitié et sa gentillesse.

Toutes les personnes qui ont participé aux expérimentations pour leur disponibilité, ainsi les membres de l'équipe MRIM.

La très chère Mme. Claudine qui était toujours là pour m'aider.

Tous les amis Amir MURANI, Alcinda et Alexis MARIN, et Marie-Thérèse et Jean-jacques Capponi pour leur gentillesse et hospitalité, qui ont apaisé mon dépaysement…

Je voudrais dédier cette thèse à :

Ma mère Raïfet AL-MAGOUT

Mon père Sadek TAHER

Mes deux sœurs Siba et Bana

Table des Matières

Introduction

1. Problématique générale de la Recherche d'Information

1.1. *Architecture Générale*

Aujourd'hui, une masse considérable d'informations textuelles est saisie et mémorisée sur des supports électroniques par les entreprises ou les administrations. L'écrit continue de constituer le mode privilégié de représentation riche et dense des connaissances humaines. Il est donc nécessaire de pouvoir exploiter efficacement l'information contenue dans de tel réservoir de connaissances. En particulier, pour y retrouver les documents qui sont pertinents avec une requête donnée. Un système de recherche d'information est l'outil utilisé pour atteindre ce but. La Figure 1 trace les grandes lignes de l'implantation d'un tel système [Savoy 1994a].

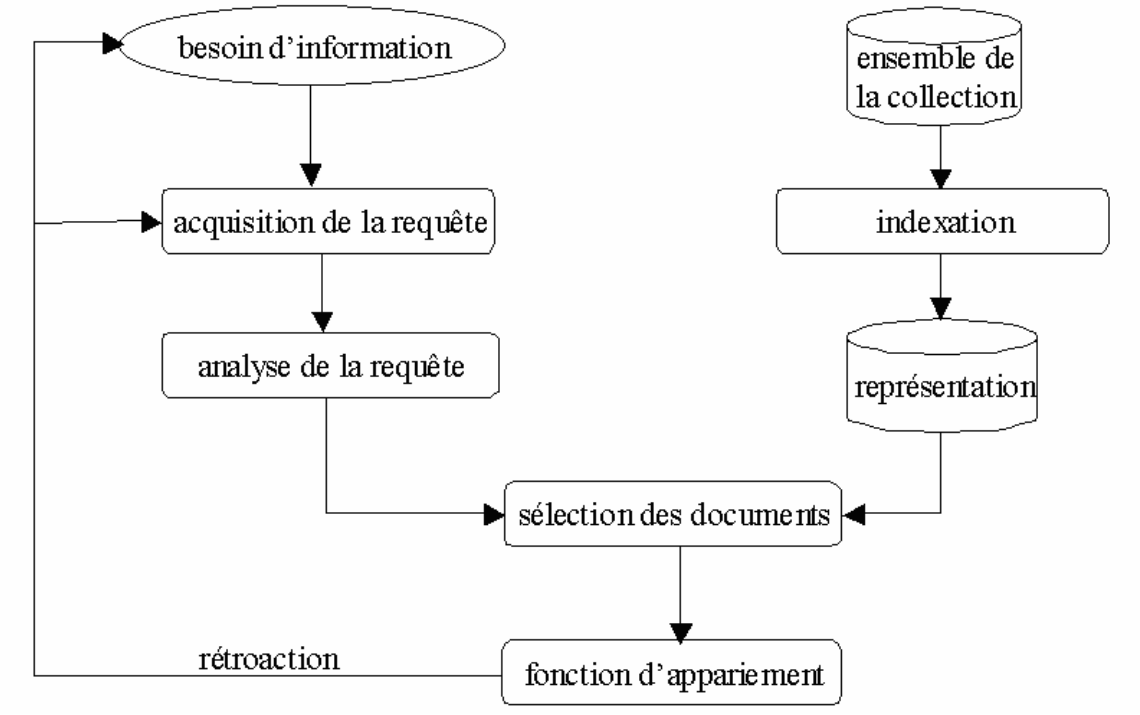

Figure 1. *Architecture générale d'un système de recherche d'information.*

Retrouver de l'information dans des documents suppose :

- Que l'ordinateur dispose des documents sur support électronique. Le système de recherche d'information construit un représentant de chacun des documents. Ce représentant se limite souvent à une liste de mots-clés. Le but poursuivi est d'obtenir une

représentation synthétique de la sémantique des documents. L'ensemble complet de ces représentations est mémorisé sur disque.

- Que l'utilisateur dispose d'outils informatiques pour décrire et exprimer ce qu'il désire sous forme de requêtes écrites. Ces dernières sont analysées par le système afin d'en tirer des représentations qui soient compatibles avec celles des documents.

Disposant d'une représentation interne des requêtes et des documents, le système effectue un appariement afin de déterminer les documents qu'il juge pertinents (« pertinence système ») avec chacune des requêtes. Une fois l'appariement réalisé, le système sélectionne les documents les plus « prometteurs » et les présente à l'utilisateur. En fait, un système de recherche d'informations ne visualise souvent qu'une référence au document et pas le document lui-même.

Sur la base de cette première réponse du système, l'usager peut indiquer au système les documents qu'il juge réellement importants (« pertinence utilisateur ») et ceux qui ne présentent aucun intérêt pour lui. À l'aide de ces informations, le système est capable de construire automatiquement une nouvelle requête (bouclage de pertinence ou rétroaction *relevance feedback* [Salton 1990]) afin de présenter une nouvelle liste de références. Ce processus, qui se révèle bénéfique et efficace, indique clairement que la recherche d'information doit toujours être vue comme un processus itératif.

Chaque modèle de recherche d'informations comprend donc, trois fonctions :

- L'indexation, qui effectue un traitement préalable des documents, en leur associant une représentation de leur contenu sémantique.
- L'interrogation, qui prend en entrée une requête à appliquer aux documents indexés, de façon à sélectionner le sous-ensemble du corpus de documents qui est pertinent pour cette requête.
- L'appariement d'une requête avec des documents qui permet l'extraction de la base des objets jugés pertinents.

Parmi les approches dites classiques, on a trois grandes familles : les modèles booléens, vectoriels [Salton 1971] et probabilistes [Turtle 1991].

1.2. *Limites et Solutions*

Afin de diminuer la différence entre la pertinence système et la pertinence utilisateur l'exécution du processus de recherche a été rendue itérative. Cela demande à l'utilisateur non seulement de **dépenser beaucoup de temps** pour arriver à un résultat satisfaisant, mais lui demande surtout **un gros effort cognitif** pour exprimer son besoin d'information. Le temps et les efforts cognitifs qui sont demandés à l'utilisateur dépendent de son niveau d'expertise, c'est-à-dire qu'ils sont fonction de son expérience concernant, d'une part, l'exécution du processus de recherche d'information, et d'autre part, du domaine d'information ciblé et du contenu de la collection. C'est pour ces raisons que les études dans le domaine de la recherche d'information ont comme objectif principal d'améliorer l'efficacité de la recherche en termes de qualité du résultat, de temps et d'efforts demandés à l'utilisateur.

En1995 Croft a listé les dix principaux problèmes liés à la recherche d'information [Croft 1995][1] : la souplesse des processus de recherche et d'indexation ; les solutions intégrées ; la recherche distribuée ; l'expansion du vocabulaire ; les interfaces et la navigation ; le filtrage ; l'efficacité du processus de recherche ; la recherche multimédia ; l'extraction d'information ; enfin, le bouclage de pertinence. Ces problèmes ont été largement étudiés afin d'améliorer le processus de recherche, à titre d'exemples on peut citer :

- La souplesse de la recherche. L'approche Cabri-n (Case Based Retrieval of Information-Nancy) proposée par [Smail 1994] est un système de recherche flexible où une stratégie de recherche générique s'adapte selon une typologie des besoins d'information de l'utilisateur.
- L'interface et visualisation du processus de recherche. Ceci s'obtient en fournissant aux utilisateurs des interfaces plus faciles à manipuler et à comprendre[2]. Cela répond au double but d'économiser le temps de l'utilisateur ainsi que son niveau d'expertise requis. Aroyo [Aroyo 1999] proposent un système AIMS (Agent-based Information Management System) qui visualise le résultat de la recherche sous une forme graphique non seulement facile d'accès, mais aussi facile à sélectionner et à comprendre. Elle fournit à l'utilisateur une information complémentaire sur le contexte des résultats retrouvés. Le graphe est présenté comme une carte de concepts (termes) qui décrit explicitement la structure de domaine et tous les documents sont rattachés aux concepts. Cela permet à l'utilisateur de traiter rapidement un plus grand nombre de documents retrouvés et d'en avoir une vision synthétique. Ce type de visualisation provoque l'imagination de l'utilisateur en lui fournissant une connaissance complémentaire sur le contexte et la signification de l'information employée.
- L'expansion des requêtes par des méthodes de bouclage de pertinence [Rocchio 1971] ou par l'utilisation de thesaurus [Bruandet 2003].
- L'intégration des résultats de plusieurs systèmes et de plusieurs collections de recherche[3] afin d'obtenir des résultats plus satisfaisants. En effet, chaque outil de recherche trouve les références pertinentes selon son propre point de vue. De ce fait, le résultat obtenu par plusieurs techniques de recherche est probablement meilleur que celui qui est obtenu par une seule technique. De telles intégrations ont été étudiées dans [Bartell 1994], [Vogt 1997], [Lee 1997], [Smeaton 1998] et [Vogt 1999]. Une autre forme d'intégration

---

[1] Flexible Indexing and Retrieval, Integrated Solutions, Distributed Information Retrieval, Efficient, Vocabulary Expansion, Interfaces and Browsing, Routing and Filtering, Effective Retrieval, Multimedia Retrieval, Information Extraction, Relevance Feedback.

[2] « *The interface is a major part of how a system is evaluated, and as the retrieval and routing algorithms become more complex to improve recall and precision, more stress is placed on the design of interfaces that make the system easy to use and understandable. Interfaces must support a range of functions including query formulation, presentation of retrieved information, feedback, and browsing.* » [Croft 1995].

[3] « *The more general problems are locating the best databases to search in a distributed environment that may contain hundreds or even thousands of databases, and merging the results that come back from the distributed search. The results must be merged in order to produce the overall ranking of retrieved items, instead of a collection of individual rankings. The difficulty of doing this effectively comes from the fact that the individual rankings may be incompatible in the sense that the numbers used to produce these rankings may not be directly comparable (they may even come from different IR systems). Research addressing these issues has begun to appear in the major conferences.* » [Croft 1995]

concerne la combinaison des résultats obtenus à partir de différentes collections de recherche. Ainsi, des documents pertinents pour une recherche d'information peuvent être issus de plusieurs collections. Il peut être intéressant d'interroger plusieurs corpus, pour cela il faut déterminer le nombre de documents à récupérer de chaque collection, et la façon de les fusionner et de les ordonner ce qui a été étudié dans [Voorhees 1994] et [Towell 1995].

- Mesure de l'efficacité de la recherche. Cela fait partie intégrante de la conception et du développement des systèmes de recherche d'information. Les travaux sur l'évaluation et les mesures de l'efficacité de la recherche comme ceux décrits dans [Dunlop 1997], [Mizzaro 1998a], [Mizzaro 1998b], [Mizzaro 2001] et [Yao 1995] sont nécessaires. Ils permettent, en effet, de déterminer et de mesurer les limites du système ; ce qui permet de comprendre les raisons de ces limites et donc, de chercher le moyen de les faire reculer. Cela permet aussi de comprendre quels sont les critères que l'utilisateur emploie pour juger les documents…

Comme beaucoup d'autres domaines de recherche de l'informatique, le domaine de la recherche d'information s'est d'abord intéressé au processus de recherche considéré comme l'activité d'un seul utilisateur. Ainsi, les techniques sophistiquées mises au point pour assister et améliorer le processus de recherche ont fait, pendant longtemps, l'hypothèse d'un contexte d'utilisation mono-utilisateur. Cependant l'arrivée des réseaux et la grande diffusion d'Internet grâce au *World Wide Web* ont ouvert de nouvelles perspectives dirigées vers la distribution, le partage et la collaboration.

1.3. *Collaboration et Recherche d'Information*

L'arrivée du « World Wide Web » et la grande utilisation d'Internet, se sont donc accompagnées d'une augmentation correspondante de la demande de systèmes de recherche d'informations fonctionnant pour des groupes d'utilisateurs. Cette demande vient, le plus souvent, d'applications coopératives ou « groupware », on peut citer l'exemple de *Lotus Notes*, qui a facilité la création rapide de bases de données ou d'information pour une organisation.

Du fait de la richesse apportée par la diversité des connaissances et des compétences des divers individus, la collaboration doit permettre un accès à l'information plus efficace, en terme de la qualité des informations obtenues et du temps mis à les obtenir. En effet, les activités humaines d'accès à l'information présentent généralement une dimension collective et collaborative très prononcée. L'information collaborative résulte des relations s'établissant entre les gens et les objets [Xiong 1998]. Par exemple, les objets peuvent être des sujets de discussion comme c'est le cas des forums de discussion USENET, ou des livres intéressants, ou des pages Web, … Dans le cas d'un livre il est possible de ne s'intéresser qu'aux personnes qui achètent ce livre. Cette information est généralement utilisée par des techniques de filtrage collaboratif pour recommander des livres à d'autres utilisateurs, (cf. ce qui est fait sur site Amazon.com[4]).

Notre travail de thèse s'intéresse principalement aux moyens à fournir pour tirer le meilleur parti possible de l'approche collaborative dans la recherche d'information.

[4] Web Site : http://www.amazon.com.

2. Présentation de Notre Travail

2.1. *Motivations*

Le fait d'élargir le contexte des activités de recherche d'information d'un utilisateur en le regardant en termes d'activités menées par une communauté d'utilisateurs coopérants, présente des avantages à deux niveaux celui de l'utilisateur et celui du processus de recherche.

- Pour l'utilisateur :
  - Il n'a plus le sentiment d'être isolé, puisqu'il a conscience que d'autres cherchent la même chose.
  - Il est davantage motivé pour exploiter au mieux son temps, afin de conduire de façon intelligente le processus de recherche.
  - Il a conscience que les autres peuvent l'aider en cas de besoin pour continuer ou approfondir le processus de recherche. Ainsi, il peut espérer résoudre les problèmes de construction de requêtes ou de validation de résultats avec les autres.
- Pour le processus de recherche, on obtient :
  - Une meilleure qualité des résultats obtenus de la recherche. En effet, le système peut trouver des informations qui n'auraient pas pu l'être si chaque utilisateur avait travaillé isolément. Cela permet, de plus, un recouvrement plus grand de l'espace d'information recherché.
  - Une meilleure efficacité de la recherche qui est conduite. En effet, quand les utilisateurs partagent « une conscience sociale » ils ne poursuivent pas leurs activités de recherche seules mais réutilisent les résultats des autres ; par exemple, en ne posant pas les mêmes requêtes. Ceci peut conduire à un gain de temps significatif pour les utilisateurs.

Cependant si l'on veut tirer parti de la collaboration, il faut intégrer aux systèmes de recherche d'informations, des techniques qui leur permettent d'organiser la collaboration entre divers utilisateurs. Ce qui introduit donc dans le domaine de la recherche d'information, des problématiques qui ont été abordées dans d'autres contextes d'applications collaboratives.

2.2. *Contexte et Problématique*

Plus précisément, dans cette étude, nous envisageons la situation présentée dans la Figure 2 où *plusieurs* utilisateurs travaillant dans des *lieux différents* cherchent des informations sur le *même sujet* de recherche au *même moment* ou à des *moments différents.* Les requêtes soumises dans leur recherche ou les résultats obtenus alimentent une mémoire collaborative. À un moment donné, l'utilisateur peut avoir besoin d'aide. Dans une situation d'utilisation isolée d'un système de recherche, en cas d'insuccès ou de difficulté, l'utilisateur, en général, demande l'aide à une personne qui peut-être un collègue ou à un spécialiste (un ou une bibliothécaire dans le cas de recherche dans une bibliothèque)… Dans le contexte que nous envisageons - la recherche d'information depuis son bureau ou son domicile - trouver une telle aide n'est pas toujours possible. L'approche collaborative permet à chaque utilisateur de bénéficier de l'aide des autres et de tirer profits des expériences qu'ils ont déjà effectuées.

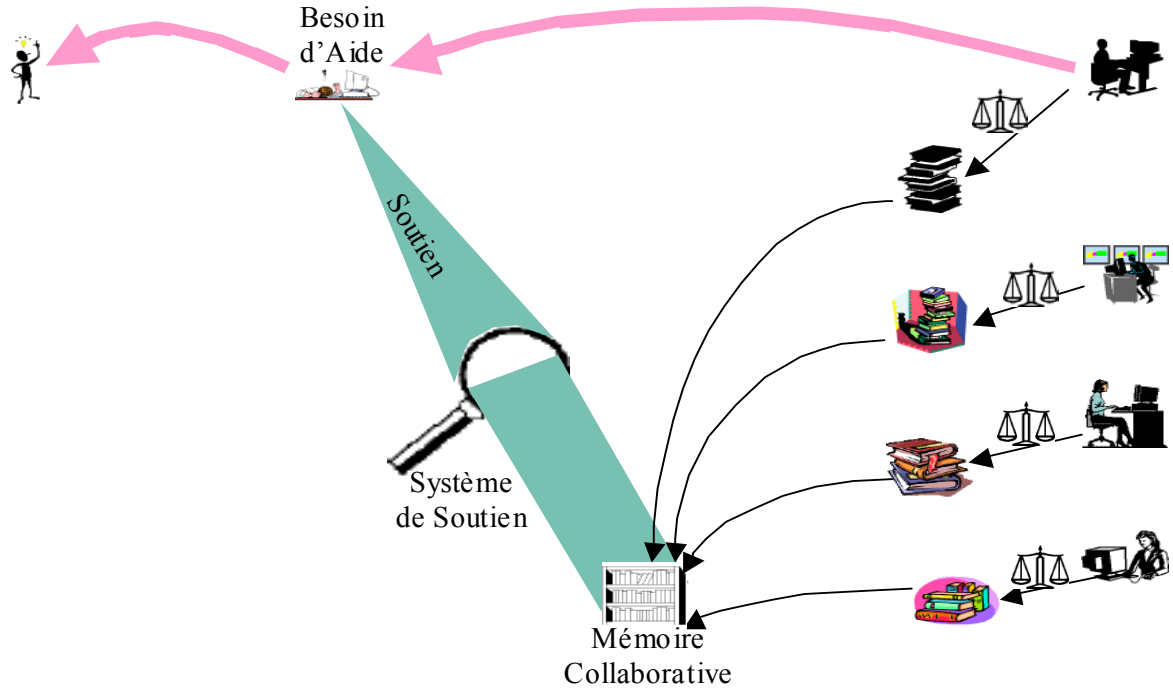

Figure 2. *Contexte de recherche collaborative.*

Le groupe effectue une recherche collaborative sur un sujet donné. Une telle recherche consiste en un ensemble de session individuelle effectuée par chaque membre du groupe.

Nous ne nous préoccupons pas de la formation du groupe. Ainsi nous supposons que nous disposons d'une interface collaborative comme celle proposée par Laurillau [Laurillau 1999] qui fournit toutes les fonctions nécessaires pour créer, organiser et maintenir un groupe d'utilisateurs qui s'intéressent au même sujet de recherche. Nous avons centré notre travail sur le fonctionnement du soutien dans la recherche d'information. Nous avons étudié un ensemble de fonctionnalités permettant d'enrichir l'interface collaborative en lui permettant un accès souple et efficace à des systèmes de recherche d'information. Notre système de soutien interagit donc avec des systèmes de recherche d'information, mais il n'intervient pas dans leurs fonctions internes. Ce travail comprend implicitement la construction et la gestion d'une mémoire collaborative. Cette mémoire est conçue et utilisée pour assurer le fonctionnement du soutien.

A partir de cette description du contexte fixé à notre travail, nous analysons les situations auxquelles notre système d'aide et de soutien à l'utilisateur devra faire face s'il veut permettre de partager les connaissances sur la technique ou le domaine de recherche, d'échanger avec les autres soit pour les aider et soit pour se faire aider sans pour autant y consacrer beaucoup de temps.

2.3. *Situations et Objectifs*

Nous examinons donc maintenant les situations dans lesquelles un utilisateur d'un système de recherche d'information pourrait avoir besoin d'aide :

- *Blocage* et *Désorientation* : l'utilisateur n'a pas ou plus d'idées pour formuler sa requête : soit il ne sait comment la formuler, soit il a formulé une requête, le résultat obtenu n'est pas satisfaisant et il ne sait plus ce qu'il doit faire. L'utilisateur a alors besoin de connaissances supplémentaires sur la technique de recherche et d'une meilleure compréhension du système. En effet, comme le note Twidale [Twidale 1995][5] la recherche d'information collaborative doit permettre l'apprentissage de la connaissance du domaine sur lequel s'applique la recherche et le moyen utilisé pour l'effectuer.
- *Doute* : l'utilisateur a des doutes, il a obtenu un résultat satisfaisant, mais il n'est pas sûr de sa qualité intrinsèque, et ne sait pas s'il a réellement exploré tous les aspects du sujet de recherche qui l'intéressent. L'utilisateur a donc besoin de se positionner et de s'évaluer par rapport aux autres utilisateurs.
- *Curiosité* : l'utilisateur n'a pas de problème particulier, mais il a besoin d'avoir une vision globale ou panoramique du travail des autres.
- *Plan* : l'utilisateur a le plan d'un article, il veut construire cet article à partir des documents retrouvés et arrangés selon le plan. En fait, l'utilisateur veut rédiger un article ou un rapport, il a un plan de rédaction. Par exemple, le plan d'un article bibliographique ou un *survey* sur « *la recherche d'information collaborative* », peut consister des éléments suivants : une présentation du domaine de collecticiels, un état de l'art concernant le collecticiels, une présentation du domaine de recherche d'information, un état de l'art concernant ce domaine, et enfin un état de l'art sur les travaux qui incluent les deux domaines. Alors, il est intéressant de proposer à cet utilisateur une organisation des références retrouvées (ou des parties de la référence) selon les éléments du plan où chaque élément du plan est défini, par exemple, par un ensemble de mots clés. En utilisant des mesures de « similarité sémantique » on peut choisir les références qui concernent tel ou tel élément du plan ou même choisir un « morceau de donnée » dans la référence (un paragraphe de texte, une image, …) et les classer selon le plan.

Dans cette thèse nous nous focalisons sur les trois premières situations. Notre objectif est donc de fournir un environnement collaboratif à un groupe d'utilisateurs désirant effectuer ensemble des recherches d'information sur un sujet commun. Ce système doit fournir à chacun de ses utilisateurs une aide interactive (avec le système) et personnalisée afin qu'il puisse faire face aux situations précédentes.

2.4. *Hypothèses*

L'hypothèse sous-jacente à ce travail est que les utilisateurs recherchent de l'information sur un *sujet de recherche commun*. Si nous voulons définir correctement un environnement collaboratif, il faut nous intéresser au fonctionnement d'un groupe et étudier ses impacts sur l'environnement collaboratif. Pour cela, nous faisons certaines hypothèses supplémentaires sur le comportement des utilisateurs. Ainsi, les utilisateurs du système doivent fournir :

---

[5] « *Information searching is an interesting context to investigate collaborative learning because it involves two processes: learning about the domain in question (say, Psychology) and learning about how to locate information. The two processes are inextricably linked - information skills can't be effectively taught in an abstract manner. In other words, (as librarians themselves have noted) it is difficult to learn about searching for information without actually searching for some real information.* »

- L'évaluation des documents retrouvés. Nous supposons que les utilisateurs donnent leur opinion, que nous appellerons dans la suite *jugement,* sur les références retrouvées par le système. Les utilisateurs ne devraient pas rechigner à fournir un jugement des références reçues car, contrairement aux systèmes de recherche d'information utilisant le bouclage de pertinence basé sur le jugement (*relevance feedback*), ces jugements ne sont pas destinés à « une machine » mais à des personnes qui les utilisent immédiatement. De plus, ils savent que ces jugements sont importants pour améliorer la qualité des résultats finaux et pour collaborer avec les autres.
- L'évaluation des utilisateurs. L'utilisateur exprime ses *préférence*s relativement aux autres utilisateurs, c'est-à-dire la confiance qu'il leur accorde. La préférence introduit une dimension sociale au soutien. Il est normal que la recherche collaborative permette aux usagers du groupe d'exprimer la connaissance qu'ils ont des autres. Nous verrons ultérieurement quelles implications sur l'organisation du groupe peut avoir cette hypothèse.

2.5. *Contributions*

Nous fournissons un soutien à l'utilisateur, lorsqu'il le demande, dans différentes tâches de recherche : formulation de requête, visualisation du résultat et reformulation de requête. Nous aidons l'utilisateur non seulement par la présentation d'une visualisation « intelligente » du processus de recherche mais aussi en lui suggérant de nouvelles requêtes construites collaborativement à partir des requêtes et/ou des résultats du groupe.

Nous avons défini un modèle de soutien souple et adapté selon :

- La typologie de groupe. Dans l'objectif de soutenir un groupe d'utilisateurs, nous avons analysé ce qu'était la collaboration dans une tâche de recherche, ce qui nous a amené à considérer la structure du groupe, et à définir une typologie du groupe selon leur degré de collaboration. Nous avons défini des stratégies de soutien différentes selon le type de groupe où la ou les valeur(s) de chacun des paramètres du modèle du soutien sont déterminée(s) selon le type de groupe. Ainsi, selon ce type, les utilisateurs n'auront pas les mêmes possibilités.
- La typologie de l'utilisateur. L'efficacité d'un utilisateur donné pendant la session de recherche dépend de sa capacité à exprimer correctement son besoin d'information c'est-à-dire sa compétence dans le domaine de recherche, et sa capacité à utiliser le mieux possible le système de recherche, c'est-à-dire sa compétence technique. Le niveau de l'utilisateur dans ces deux compétences définit la typologie de l'utilisateur. Nous avons défini une stratégie de soutien selon cette typologie. Ainsi le type de l'utilisateur qui a besoin d'aide va nous servir pour choisir le(s) utilisateur(s) du groupe qui ont des qualités complémentaires en terme d'expertise sur le domaine et sur la technique. Cet ou ces utilisateur(s) sont considéré(s) comme le(s) plus apte(s) à lui apporter de l'aide.

A partir de ce modèle de soutien adapté, l'utilisateur qui a besoin d'aide peut interrompre la boucle de recherche et s'intéresser au soutien en faisant une demande dans laquelle il exprime ses souhaits et ses désirs au moyen, entre autres, de critères de personnalisation de soutien. Ces critères sont utilisés afin de personnaliser l'aide à l'utilisateur qui la demande.

Donc, nous avons construit une aide « à la demande » de l'utilisateur, cette aide n'est pas prise en compte toujours de la même manière, elle dépend de ses désirs et souhaits mais aussi de son contexte personnel et du groupe.

Notre démarche est un peu différente de celle adoptée par les systèmes C-TROI [Hoppe 1994], MOO-Gopher [Masinter 1993], VR-VIBE [Glance 1999], qui supportent et facilitent les interactions personnelles, et la visualisation de ces communications pendant le processus de recherche. Nous voulons que les utilisateurs tirent profit de la collaboration sans pour autant investir dans les interactions. Nous fournissons donc conseils et aides à l'utilisateur sans qu'il ait à communiquer directement. Nous avons construit un outil qui peut servir d'intermédiaire de communication entre les utilisateurs.

Comme dans les systèmes Brodway [Trousse 1999] ou Brodway-QR [Kanawati 1999], nous nous intéressons au profil de l'utilisateur pour effectuer le partage du produit de recherche au moyen du filtrage collaboratif. Le partage que nous mettons en œuvre prend en compte, outre le profil de l'utilisateur, l'implication d'organisation de groupe sur la recherche ; et notre notion du profil de l'utilisateur et notre manière de le prendre en compte sont différentes. Dans notre approche le profil de l'utilisateur n'est pas précisé en termes de ses requêtes et de ses résultats évalués mais plutôt en terme de la façon dont l'utilisateur réagit pendant la recherche face au système et aux résultats, ce qui nous permet de mesurer ses compétences dans le domaine et la technique. De plus dans notre approche, le partage ne se base pas sur la notion de similarité des profils entre les utilisateurs, mais plutôt sur la notion de complémentarité des compétences entre eux. Un autre point de différence avec le filtrage collaboratif est la participation demandée à l'utilisateur. En effet, dans notre système l'utilisateur intervient dans l'opération de soutien en la demandant, en précisant quelle aide il désire et sur quels critères la personnaliser. Nos expérimentations ont montré que les utilisateurs apprécient ces possibilités d'intervention.

Comme dans les systèmes ARIADNE [Twidale 1996], CIRE [Romano 1999], Collaborative Spider [Chau 2003] nous permettons le partage et la visualisation du processus de recherche. Cependant contrairement à ces systèmes, la visualisation que nous fournissons est synthétique et personnalisée.

2.6. *Démarche*

La suite de la présentation de notre travail se divise en deux parties : l'une constituée des chapitres 1 à 3 présente un état de l'art des concepts et des techniques que nous avons utilisées pour élaborer une solution au problème qui nous était posé ; l'autre constituée des chapitres de 4 à 6 développe nos propositions, nos réalisations, ainsi qu'une évaluation critique. Plus précisément :

Dans le CHAPITRE.1, nous nous intéressons dans ce chapitre aux collecticiels en général. Après en avoir proposé une classification, nous présentons quelques approches comme la recommandation, le filtrage collaboratif ou les interfaces collaboratives qui sont en synergie avec le domaine de la recherche d'information. Chacune des approches présentées est située dans la classification temporelle « synchrone » ou « asynchrone » des collecticiels.

Dans le CHAPITRE.2, nous nous intéressons dans ce chapitre à l'ensemble des concepts et des notions proposés dans littérature pour juger la pertinence d'un document ceci afin de

déterminer quels sont les concepts et les mesures à mettre en oeuvre pour permettre l'élaboration du jugement de pertinence que nous adoptons dans cette étude, et l'évaluation de l'efficacité de recherche.

Dans le CHAPITRE.3, les techniques de fusions telles qu'elles sont élaborées en recherche d'information sont présentée, car nous devons permettre à plusieurs utilisateurs de fusionner leurs efforts.

Le CHAPITRE.4 présente l'environnement collaboratif de soutien que nous proposons. Pour cela nous définissons les différents éléments de l'environnement, les données qui sont pris en compte, le soutien en terme de tâches, de dimensions, …, le contexte et la typologie de l'utilisateur, la typologie de groupe, et les stratégies de soutien selon ces deux typologies proposées.

Le CHAPITRE.5 présente comment nous personnalisons le soutien. Nous présentons, tout d'abord, les modèles des requêtes et des documents utilisés, puis nous définissons les critères de personnalisation de soutien, ensuite, la procédure de recherche et d'extraction de la mémoire collaborative selon les critères de personnalisation, des éléments nécessaires pour la réalisation du soutien demandé.

Le CHAPITRE.6 présente l'implantation de notre proposition de soutien, nous présentons d'abord l'architecture du système, puis le prototype implanté, ensuite afin de clarifier notre approche nous développons un exemple complet. Enfin nous présentons les expérimentations que nous avons effectuées.

Finalement, la conclusion synthétise notre démarche et les différentes conclusions auxquelles nous sommes arrivées à l'issu de notre travail. Nous tentons d'ouvrir différentes perspectives pour la continuation de ce travail.

CHAPITRE.1. Collecticiels et Recherche d'Information Collaborative

Dans ce travail, nous avons une approche visant à favoriser une approche collaborative de la recherche d'information au moyen d'un système permettant à différents utilisateurs de collaborer et de combiner au mieux leurs efforts. Afin de concevoir un tel système, nous tentons de mettre en évidence tous les problèmes liés à cette approche. Tout d'abord, nous présentons différents aspects intervenant dans la définition et la classification des collecticiels (paragraphe 1.1) afin de situer notre travail relativement à ce domaine. Ensuite, nous examinons les différentes approches de la recherche d'information collaborative, afin de mettre en évidence les principes de la collaboration dans ce domaine particulier qu'est la recherche d'information (paragraphe 1.2). Enfin, nous conclurons dans le paragraphe 1.3 en classant les différentes approches de recherche collaborative recensées relativement à une classification temporelle des collecticiels en deux groupes principaux « synchrone » ou « asynchrone ».

1.1. Collecticiels ou « Groupware »

Le groupware, néologisme anglo-saxon récent, fait référence à la technologie informatique (-ware) appliquée au travail de groupe. Les termes français de « collecticiel » voire de « synergicicel » ont été proposés comme des traductions possibles, mais l'anglicisme « groupware » a résisté. Parmi les différentes définitions du groupware qui sont proposées, nous retiendrons la définition suivante [Courbon 1997] : « *Le groupware est l'ensemble des technologies et des méthodes de travail associées qui, par l'intermédiaire de la communication électronique, permettent le partage de l'information, sur un support numérique, à un groupe engagé dans un travail collaboratif.* »

Dans ce qui suit, nous nous intéressons tout d'abord à l'environnement du groupe et aux situations de travail (paragraphe 1.1.1). Puis, à une classification des fonctions que doit fournir le groupware afin d'apporter l'aide propre à rendre efficace le travail de groupe (paragraphe 1.1.2). Nous nous appuyons essentiellement sur le travail de Marc Favier [Favier 1998] pour cette partie de notre travail sur le groupware.

1.1.1. *Situations de Travail*

Les configurations de travail d'un groupe sont multiples, elles peuvent se décliner selon la localisation des participants, le moment de leurs interventions, leur nombre (la taille des groupes), leur durée de vie ou, encore, les résultats attendus. C'est ce que nous étudions maintenant et dont les conclusions sont résumées dans le Tableau 1.1.

1. *Lieu* et *Temps* : c'est la classification la plus classique des configurations de travail d'un groupe, elle consiste à identifier quatre situations selon que les participants du groupe travaillent :
   - Dans un même lieu ou en des lieux différents.
   - Au même moment, on parle alors de groupware « synchrone », ou à des moments différents, on parle dans ce cas de groupware « asynchrone ».

   La Figure 1.1 résume ces quatre situations donnant des exemples pour chacune d'elle.

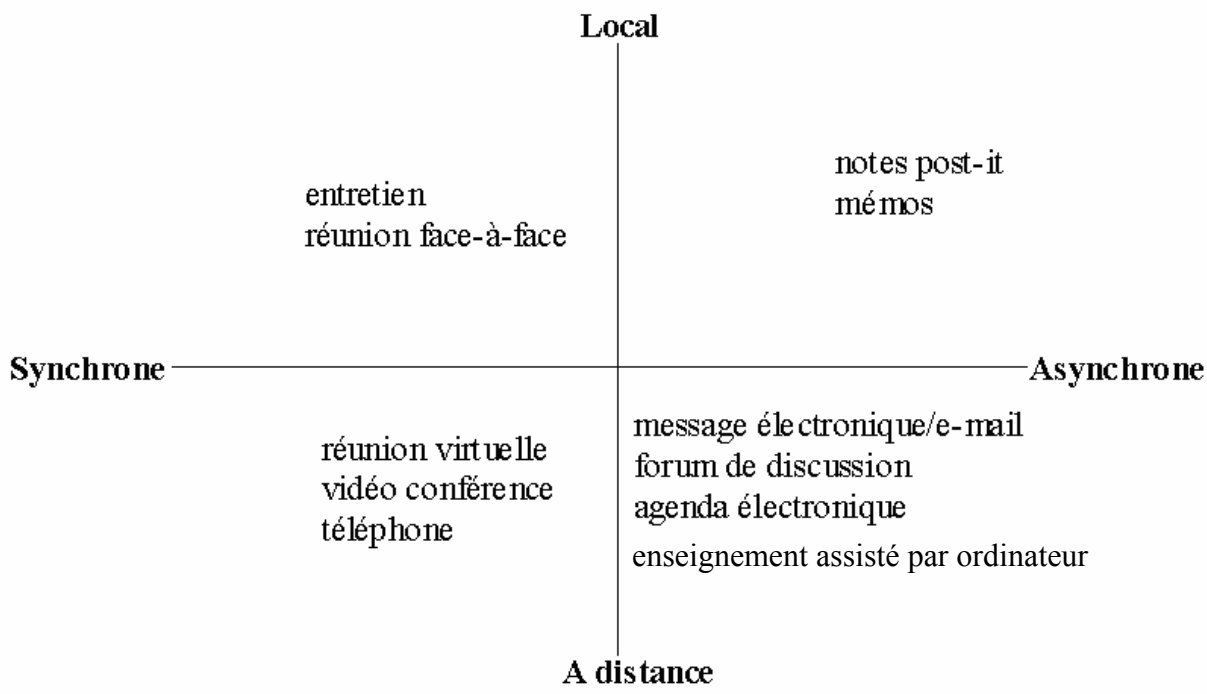

Figure 1.1. *Classification temps-lieu des situations de travail.*

2. *Durée de vie et Taille d'un groupe* : les situations de travail des groupes peuvent aussi être vues selon une autre perspective, celle de la durée de vie du groupe et de sa taille. En effet, le groupe peut être plus ou moins important, et constitué de gens travaillant ensemble en permanence ou seulement pour une durée limitée. La réunion et la discussion entre les membres d'un groupe pour résoudre un problème ou effectuer un travail est un exemple de groupe *temporaire restreint*, un exemple de groupe *large* mais *temporaire* est l'ensemble des personnes sollicitées pour répondre à un questionnaire ou à une enquête. Des experts mettant en commun leur savoir-faire à l'occasion d'un projet (service d'étude par exemple) illustrent ce qu'est un groupe *restreint* mais *stable* dans le temps. Enfin, un groupe *large* et *permanent* peut, à la limite, être illustré par l'ensemble du personnel d'une entreprise.
3. *Types des résultats du travail effectué* : ces types sont différents selon que le travail du groupe débouche sur une décision, l'expression d'une demande, l'exécution d'une mission, ou bien une assistance ou une prospection.

Classifications	Situation de Travail
Lieu et Temps	• local synchrone. • local asynchrone. • à distance synchrone. • à distance asynchrone.
Durée de vie et Taille de groupe	• temporaire restreint. • temporaire large. • permanent restreint. • permanent large.
Types des résultats du travail effectué	• décision. • expression d'une demande. • exécution d'une mission. • assistance. • prospection.

Tableau 1.1. *Classifications des situations de travail.*

1.1.2. *Fonctions du Groupware*

En considérant les classifications que nous venons de définir dans le paragraphe précédent, nous nous proposons, maintenant, de définir les fonctions que l'on doit attendre du groupware. Traditionnellement, elle sont désignées par « 3C », à savoir Communication, Coordination et Collaboration. Deux autres aspects supplémentaires sont également mis en évidence, ce sont la mémorisation et la circulation de l'information. La Figure 1.2 synthétise les cinq fonctions caractéristiques du groupware et leurs dépendances [Favier 1998].

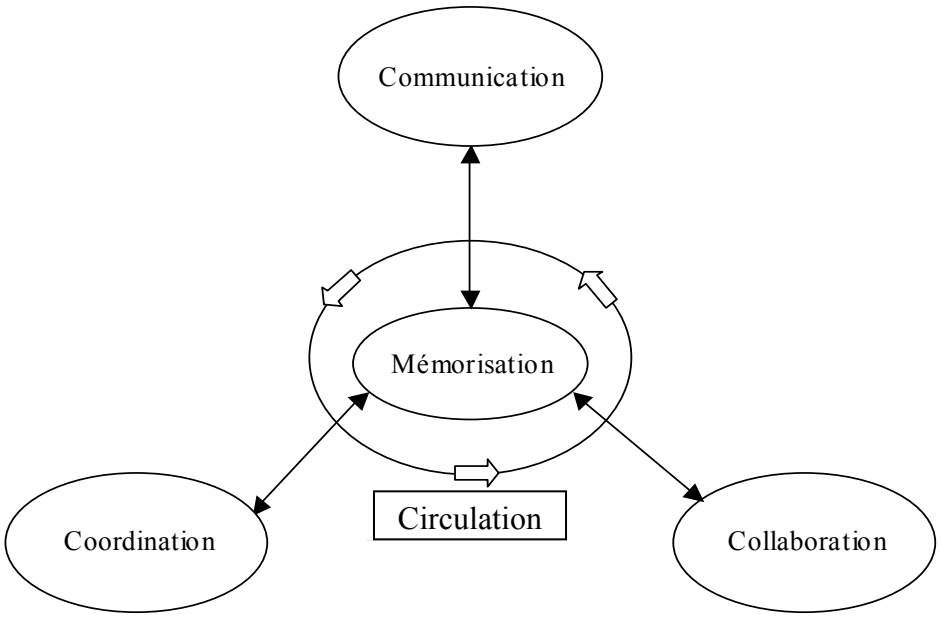

Figure 1.2. *Les fonctions du groupware.*

1. Communication interpersonnelle : elle peut prendre plusieurs formes :
   - *Un à un* : il s'agit en général d'échanges de messages électroniques.
   - *Un à plusieurs* : c'est le cas, par exemple, des messages envoyés à une liste de diffusion. Les nouvelles technologies issues de la *vidéoconférence* permettent à une personne à partir d'un poste informatique de diffuser des informations en temps réel à un ensemble de participants.
   - *Plusieurs à plusieurs* : on parle de discussion, comme dans les réunions électroniques ou les forums de discussion.
2. Coordination des activités : le groupe de travail doit coordonner les tâches confiées à chacun afin que le travail soit effectué selon l'agenda et les spécifications définis. La coordination suppose des rôles définis, des dates prévues d'achèvement, des tâches constituant le travail entrepris, une assignation de ces tâches à des personnes et, éventuellement, l'ordonnancement de ces tâches. Les fonctions que l'on attend du groupware en la matière se déclinent donc sur plusieurs dimensions : le calendrier des disponibilités, le suivi des tâches et les arbitrages entre les différentes contraintes.
3. Collaboration : cette notion fait référence à deux visions du travail en commun. La première s'intéresse aux relations entre personnes dans lesquelles il y a un « demandeur » définissant un résultat attendu à un « exécutant » chargé de le réaliser. Dans la seconde vision, les rôles sont moins hiérarchiquement définis. Les personnes travaillent ensemble sur un travail et tentent de le réaliser en combinant leurs différents points de vue, leurs compétences et leurs aptitudes.

Fonctions	Explication
Communication interpersonnelle	• un à un, • un à plusieurs, • plusieurs à plusieurs.
Coordination des activités	• définir des rôles, • prévoir des dates d'achèvement des tâches, • assigner les tâches à des personnes, • définir un séquencement des tâches.
Collaboration	• relation demandeur-exécutant, • relation équivalant combinant les efforts.
Mémoire du groupe	• les informations organisationnelles, • les informations sociales, • l'expression de travail collectif effectué.
Circulation de l'information	• avertissements, • notifications, • demandes, • formulaires électroniques à compléter, • accords, • …etc.

Tableau 1.2. *Résumé des fonctions attendues.*

4. Mémoire du groupe : la mémoire comporte toutes les informations de nature organisationnelle et sociale du groupe et la trace du travail collectif effectué. Ainsi elle comporte aussi bien les messages échangés avec un contenu textuel et informel évident, qu'une trace des relations de travail comme les informations concernant des personnes représentées par une adresse, mais aussi par les rôles qu'elles jouent ou ont joué, les interactions entre les personnes, les liens entre les documents …
5. Circulation de l'information : la communication, la coordination et la collaboration impliquent implicitement un processus de circulation d'information. Il s'agit d'avertissements, de notifications, de demandes, de formulaires électroniques à compléter, d'accords, etc. Pour réaliser cela, le groupware doit fournir des mécanismes plus ou moins sophistiqués de transmission, de manière automatique ou grâce à des moyens de programmation adaptés à cette activité. Une forme extrême de la circulation de l'information est le *workflow* ou automatisation de processus administratifs.

Le Tableau 1.2 résume les fonctions attendues d'un groupware.

1.2. Soutien dans la Recherche d'Information Collaborative

Le soutien par ordinateur peut intervenir dans diverses formes d'activités de recherche collaborative, ainsi dans [Twidale 1997] ; l'auteur met en évidence l'assistance pour :

1. Les interactions inter personnelles qui peuvent se produire pendant le processus de recherche lui-même. Entre dans cette catégorie les services, opérant probablement en mode synchrone, permettant d'identifier les personnes pouvant conseiller sur la recherche. Par exemple, des experts ou des utilisateurs ayant précédemment conduit une recherche sur un thème semblable (paragraphe 1.2.1).
2. Le partage du produit de recherche par les services de recommandation, d'annotation, et d'évaluation (paragraphe 1.2.2).
3. Le partage du processus de recherche et le produit final. Le processus de recherche et le produit final sont enregistrés afin d'être présentés et visualisés (paragraphe 1.2.3).

Dans la suite, nous discutons des différentes approches de la recherche d'information collaborative selon ces formes de soutien afin de déterminer les lacunes de ces approches et de mettre en évidence certaines de leurs limites.

1.2.1. *Soutien des Interactions Pendant le Processus de Recherche*

Nous présentons trois interfaces qui soutiennent les interactions entre les utilisateurs pendant la recherche. La première, « MOO-Gopher » est une interface textuelle interactive qui permet à plusieurs utilisateurs d'explorer collaborativement l'espace d'information et d'en discuter (paragraphe 1.2.1.1). La seconde, « C-TORI » est une interface orientée tâche qui fournit des facilités pour la recherche d'information collaborative synchrone. Ces facilités consistent à permettre le partage des historiques de recherche, la navigation coopérative dans les résultats et la formulation coopérative des requêtes (paragraphe 1.2.1.2). La troisième interface, « VR-VIBE » est une interface Multi-Utilisateurs 3D qui représente explicitement les utilisateurs dans le processus de recherche et qui soutient les activités collaboratives au sein de communauté network (paragraphe 1.2.1.3).

1.2.1.1. MOO-Gopher Environnement de Réalité Virtuelle pour la Recherche

Dans [Masinter 1993] le système « MOO-Gopher » est décrit comme l'introduction d'un outil de recherche d'information « Gopher » dans un environnement social de réseaux « MOO » de réalité virtuelle basée sur le texte. On va présenter les composantes « MOO » et « Gopher », puis le système « MOO-Gopher » et ce qu'il apporte de positif.

Les environnements de réalité virtuelle basés sur le texte permettent de représenter et de décrire textuellement les emplacements, les objets, les utilisateurs et leurs interactions. MUDs (Multi-User Dungeon) ou MOOs qui est un MUD orienté objet (MUD Object Oriented) sont des exemples typiques de telles interfaces textuelles. Il s'agit de programmes qui acceptent les connexions de multiples utilisateurs à travers différents types de réseaux (par exemple : le téléphone ou Internet), et offrent à chaque utilisateur la possibilité d'accéder à une base de données partagée. L'utilisateur connecté est situé dans une « pièce » (room), décrite textuellement, qui contient d'autres objets et d'autres utilisateurs. L'utilisateur navigue et manipule la base de donnée depuis « l'intérieur » d'une pièce qui limite la visibilité de l'utilisateur aux objets qu'elle contient. L'utilisateur peut aller d'une pièce à une autre via les « sorties » qui les relient. Le concept de pièce est donc une métaphore pour définir des ensembles de visibilité sur les objets de la base, les « sorties » d'une pièce agissent comme des liens entre ces emplacements.

« Gopher » est construit suivant un modèle client/serveur. Les échanges entre client et serveur reposent sur le principe simple suivant : un client contacte un serveur via une connexion « telnet » et lui envoie une ligne de texte identifiant les données qu'il désire consulter. Le serveur Gopher présente l'information dont il dispose sous une forme hiérarchique. Il permet de diffuser une grande variété de types de document (texte, image, son, répertoire, ...).

Les gens qui utilisent « MOO-Gopher », pour l'exploration collaborative d'espace de « Gopher », ont généralement l'intention de résoudre un problème ou de trouver la ou les réponses à une question. Plusieurs parmi ces utilisateurs étaient déjà des utilisateurs de « Gopher », mais ils utilisent « MOO-Gopher », parce que « MOO » fournit un moyen d'expression collaborative pour la recherche d'information, par opposition à l'isolement ressenti en utilisant « Gopher ». Les utilisateurs qui ont fait cette expérience trouvent que « MOO-Gopher » :

- favorise l'échange d'idées sur comment trouver une information ; les utilisateurs peuvent prendre d'autre(s) idée(s) et les développer selon différentes directions, ce qui n'est pas facile lors d'une recherche individuelle indépendante.
- facilite, avec son environnement semi-réaliste, la communication de résultats d'un utilisateur à l'autre.

Bien sûr, ces avantages existent également pour les membres d'un groupe travaillant ensemble dans un même espace physique. Mais, grâce à son environnement semi-réaliste, « MOO-Gopher » permet aux utilisateurs de chercher ensemble à partir de lieux différents.

De l'étude présentée dans [Masinter 1993], en dehors du fait qu'elle décrit une étape dans l'histoire de la recherche d'information collaborative, l'on retiendra deux choses :

- Elle fournit une approche collaborative pour la recherche d'information via une interface interactive.
- Elle confirme à travers l'opinion des utilisateurs l'importance de l'aspect collaboratif et à quel point il est souhaitable et désiré dans le domaine de recherche d'information. Elle a montré la motivation des gens pour collaborer en recherche d'information.

1.2.1.2. C-TORI Interface pour la Recherche Collaborative

Hoppe et Zhao [Hoppe 1994] proposent une interface pour la recherche coopérative dans les bases de données C-TORI (Cooperative-TORI), une version coopérative du prototype TORI (Task-Oriented database Retrieval Interface). La collaboration entre les utilisateurs dans C-TORI passe par :

1. La formulation collaborative de requête : cela peut être réalisé au moyen de différentes opérations :
   - *Coupler* : les utilisateurs peuvent spécifier en même temps la requête. Un mécanisme de verrouillage automatique pour éviter les conflits est mis en place. Chaque action (spécifier une requête, visualiser et évaluer son résultat) faite par un utilisateur est visualisée/affichée et re-exécutée dans l'environnement des autres utilisateurs, c'est-à-dire What You See Is What I See WYSIWIS.
   - *Copier* : l'utilisateur peut copier une requête de l'environnement d'un autre utilisateur dans son propre environnement.
   - *Fusionner* : il peut fusionner la requête d'un autre utilisateur avec sa requête.
2. Le partage des histoires de requêtes : l'*histoire d'une requête* est un ensemble ordonné des requêtes précédentes avec leurs sous-ensembles de résultat potentiellement sélectionnés. Ainsi, chaque requête émise par un utilisateur est seulement enregistrée (sans être exécutée) avec le sous-ensemble de son résultat dans les environnements de tous les utilisateurs couplés. Cela permet de partager la « mémoire » entre les utilisateurs.
3. Le parcours collaboratif des résultats : les utilisateurs peuvent naviguer collaborativement à travers les résultats. La formulation collaborative de requête le permet par défaut. La différence ici est qu'il n'y a pas de spécification jointive de la requête et que la requête n'est pas exécutée dans l'environnement de chacun. Cela permet alors aux utilisateurs de partager les résultats sans accéder à nouveau à la base de données.

1.2.1.3. VR-VIBE Environnement 3D Virtuel pour des Communautés Network

Le travail dans [Glance 1999] focalise l'étude du soutien des activités collaboratives sur deux types de communautés qui ont des caractéristiques bien différentes :

- Les communautés de cadre de travail *workplace*, c'est-à-dire les communautés partageant un même lieu de travail. Ces communautés sont, en général, stables. Elles consistent en un groupe fermé de personnes où chacun connaît les tâches des autres et où les personnes ont l'habitude de collaborer pour effectuer leur travail.
- La communauté des utilisateurs des services qui facilitent l'accès commun aux bibliothèques, les utilisateurs ici construisent une communauté seulement parce qu'ils utilisent les mêmes services, et parce qu'ils sont au même endroit en même temps. La

collaboration entre eux émerge seulement s'ils se rencontrent. Cette collaboration consiste essentiellement à faire partager son expérience d'utilisation de la bibliothèque.

Dans ce travail, les auteurs explorent deux approches axées sur les interfaces :

- VR-VIBE, pour une coopération étroite, c'est une interface Multi-Utilisateurs 3D pour la recherche de collections de documents. Les informations de navigation et les utilisateurs navigants sont représentés dans un environnement 3D virtuel. En « VR-VIBE », les utilisateurs peuvent itérativement construire une recherche de collection de documents et feuilleter les documents retrouvés par la recherche. En plus de la représentation des documents et des utilisateurs, les utilisateurs peuvent se voir et communiquer via des canaux audio en temps réel ou par des dialogues textuels « *chat* ».
- Ecran de visualisation pour une coopération relâchée. De grands écrans situés dans des lieux fréquentés par les membres de la communauté, fournissent une conscience périphérique des activités en cours dans la communauté. Cette approche focalise sur une collaboration couplée de façon plus lâche et elle a pour objectif de promouvoir une conscience des éléments qui intéressent à l'instant présent la communauté. Le logiciel affiche sur des écrans situés dans des lieux importants pour la communauté, les éléments intéressants ainsi que les commentaires et les évaluations des utilisateurs.

La première interface permet un couplage fort entre les utilisateurs qui sont en communication directe en permanence, et qui peuvent suivre par eux-mêmes les activités des autres. Ainsi cette interface soutient, en les permettant, les interactions entre les utilisateurs durant le processus de recherche. Tandis que, la deuxième interface, à la différence de la première, permet un couplage faible entre les utilisateurs en leur permettant d'avoir conscience des autres et de la communauté. Cette dernière interface synthétise de façon radicale les activités de l'ensemble de la communauté dans le but d'informer toute la communauté, sans s'intéresser en particulier aux activités et aux centres d'intérêt de chacun. Ainsi cette deuxième interface soutient le partage du produit de recherche.

Ce travail insiste sur l'importance de prendre conscience des autres. Cette idée nous semble très importante. Aussi, baserons nous notre approche sur l'idée d'avoir conscience du travail de ceux qui cherchent des informations qui sont probablement intéressantes pour tous.

1.2.2. *Soutien par le Filtrage Collaboratif ou la Recommandation*

Nous allons tout d'abord donner une brève introduction sur ce qu'est le filtrage d'information (paragraphe 1.2.2.1). Nous présentons ensuite une étude qui suggère de considérer le filtrage collaboratif comme un groupware (paragraphe 1.2.2.2). Il y a plusieurs approches du filtrage collaboratif, nous avons choisi de présenter celles qui sont proches de la notre, c'est-à-dire celles qui tiennent compte du comportement de l'utilisateur. Ainsi, nous présentons l'approche Brodway pour l'aide à la navigation sur le Web qui recommande à l'utilisateur les pages qui ont satisfait des utilisateurs ayant navigué de la même manière (paragraphe 1.2.2.3), et l'approche Brodway-QR. Cette dernière s'inspire de la précédente pour aider au raffinement de requête pour un méta-moteur. On recommande à un utilisateur donné la requête d'un autre utilisateur ayant déposé des requêtes semblables (paragraphe 1.2.2.4). Nous discutons (paragraphe 1.2.2.5) des points intéressants apportés par les approches Broadway et Broadway-QR. Enfin, la dernière approche que nous présentons ne concerne pas directement le filtrage collaboratif. Mais, c'est une approche qui aide les

utilisateurs à comprendre les contextes sociaux d'environnements d'interactions comme « Usenet », ce qui peut être utilisé pour le filtrage (paragraphe 1.2.2.6).

1.2.2.1. Introduction

Mizzaro et Tasso [Mizzaro 2002] distinguent deux modes d'accès aux informations (voir Tableau 1.3) :

- La recherche d'information (RI). Elle est caractérisée par une base statique de documents, un besoin d'information à court terme, et une requête composée par quelques (souvent moins de deux) termes. Les moteurs de recherche sur le Web sont l'exemple le plus connu des systèmes RI.
- Le filtrage d'information (FI). Il est caractérisé par une base dynamique de données (réellement un flux entrant de documents) et un besoin d'information statique et à long terme. Les utilisateurs des systèmes FI sont donc plus motivés pour exprimer plus précisément l'information dont ils ont besoin. Ils sont prêts à la décrire plus précisément et plus complètement puisque ces descriptions sont supposées rester valides longtemps. Ces descriptions s'appellent habituellement des *profils d'utilisateur* (ou *modèles*), et se composent de beaucoup de données : les concepts, les relations entre eux, les poids, ….

La problématique des systèmes de filtrage d'information n'est qu'un cas particulier de celle des systèmes de recherche d'information[1].

	Recherche d'Information	**Filtrage d'Information**
base de documents	statique	dynamique
besoin d'information	court terme	long terme
représentation des besoins d'information	requête (contient peu (2 à 3) de termes)	profil/modèle (description longue et précise)

Tableau 1.3. *Résumé de la distinction entre recherche et filtrage d'information.*

Les systèmes de filtrage d'information sont, à leur tour, partagés traditionnellement entre deux familles principales [Kanawati 1999] :

- Filtrage basé sur le contenu. On recommande à l'utilisateur des références ou des actions selon l'évaluation de ses actions passées.
- Filtrage collaboratif. On recommande à l'utilisateur les références et les actions qui ont satisfait d'autres utilisateurs ayant un profil similaire au sien.

[1] « *Several of the systems referred to in this section use the term information filtering rather than IR, but most of the issues which appear at first to be unique to information filtering, are really specializations of IR problems*. » [Twidale 1997]

1.2.2.2. Filtrage Collaboratif et Groupware

Lueg [Lueg 1998] discute les ressemblances et les différences entre la problématique du filtrage collaboratif actif et celle du groupware. Il suggère de considérer le filtrage collaboratif comme un groupware afin d'expliquer certaines difficultés et d'apprendre des expériences du groupware. Dans la suite, nous présentons le filtrage collaboratif actif puis les points de ressemblance et de différence avec le groupware.

Le fait de qualifier le filtrage collaboratif par l'un des adjectifs « actif » ou « passif » signifie qu'il y a ou non intention de la part de la personne, ayant localisé une information particulière de la partager avec d'autres. Dans cette intention, un outil doit être fournit aux utilisateurs afin de leur permettre de distribuer l'information serait-ce au prix d'un petit effort. Ainsi, le système CORA (Collaboratif Recommender Agent) permet à l'utilisateur de recommander des URLs (localisations de pages Web) à des destinataires d'un groupe bien défini. L'URL peut être envoyé en cliquant sur l'icône correspondante, les URL recommandés par les autres sont inscrits dans la fenêtre qui est utilisée pour l'envoi des recommandations.

Le filtrage collaboratif actif et le groupware partagent les problèmes suivants :

- Le besoin d'acceptation par les utilisateurs. Le succès de l'application du filtrage collaboratif ou du groupware dépend de leur acceptation par les utilisateurs. L'utilisation de telles applications doit faire émerger des besoins.
- Le bénéfice retiré dépend de l'utilisation. En effet, le bénéfice résultant du filtrage collaboratif et/ou du groupware existe seulement si l'application utilisant ces mécanismes est employée effectivement par plusieurs personnes.
- Le problème du « démarrage à froid ». Tant que l'application de filtrage collaboratif ou de groupware n'a pas été employée par plusieurs utilisateurs elle ne peut fournir des profits en particulier au démarrage.

Cependant, le filtrage collaboratif actif et le groupware présentent les différences suivantes :

- Le groupe ciblé. Le groupe ciblé par le groupware est habituellement défini par la structure d'une organisation et par les tâches allouées à ses membres. Au contraire, le groupe ciblé par un outil de partage d'information est défini par le désir et/ou la volonté de ses membres de partager des intérêts.
- La quantité des participations actives nécessaire. Le filtrage collaboratif actif fournit des bénéfices même si seulement quelques utilisateurs participent activement alors que les autres consomment l'information passivement. Donc, il ressemble au système de communication Usenet News avec sa significative asymétrie (l'absence significative de symétrie) entre quelques participants actifs (ceux qui postent des articles) et beaucoup de participants passifs (ceux qui les lisent). En revanche l'efficacité d'un groupware, comme la gestion d'agenda, dépend largement de son utilisation active par la plupart des membres d'une organisation.

Le Tableau 1.4 résume les points de ressemblance et de différence entre le groupware et le filtrage collaboratif actif.

		Groupware	Filtrage Collaboratif Actif
ressemblance		besoin d'acceptation d'utilisateur. bénéfice dépend de l'utilisation. problème de « démarrage à froid ».	
différence	groupe ciblé	membres d'une organisation	ceux désirants partager des intérêts
	quantité des participations active nécessaire	grandes participations actives	quelques participations actives

Tableau 1.4. *Résumé de la comparaison entre groupeware et filtrage collaboratif actif.*

Selon Lueg, le fait de voir le filtrage collaboratif actif comme une des catégories d'application groupware permet de profiter de la grande masse de littérature sur les expériences groupware réussites ou ayant échoué. C'est particulièrement intéressant puisque les expériences avec les systèmes de filtrage collaboratif ont tendance à se concentrer plutôt sur les avantages retirés par le système plutôt que sur la considération des relations sociales au sein de l'organisation.

1.2.2.3. Broadway Aide à la Navigation

Un type particulier du filtrage collaboratif est celui de l'approche Broadway (BROwsing ADvisor reusing pathWAYs) [Trousse 1999] qui s'appuie sur un filtrage basé sur la réutilisation des comportements de l'utilisateur et/ou d'un groupe d'utilisateurs pour l'aide à la navigation sur le Web.

Cette approche repose sur l'utilisation des techniques de raisonnement à partir de cas (RàPC) afin de recommander à un utilisateur des pages qui ont satisfait des utilisateurs ayant navigué d'une manière semblable.

Le raisonnement à partir de cas est une approche de résolution de problèmes basée sur la réutilisation par analogie d'expériences passées appelées *cas* au cours d'un cycle de raisonnement afin de résoudre (dans le futur) plus efficacement des problèmes et d'éviter les échecs passés. Un cas est composé de deux parties décrivant respectivement un *problème* et la *solution* qui lui a été appliquée. Dans Broadway les cas sont principalement composés de :

- *Problème/situation comportementale*. La description du comportement d'un utilisateur à un instant de référence dans sa navigation est constituée notamment de la séquence des pages sélectionnées pour leur pertinence et de leur contenu.
- *Solution*. C'est une liste de pages évaluées et recommandées pour le comportement décrit.

Ainsi, pour l'utilisateur qui est en situation comportementale similaire, c'est-à-dire que sa liste des pages sélectionnées est similaire à celle d'un ou de cas précédent(s) (à celui d'autre(s) utilisateur(s)), le système lui recommande la partie solution du cas qui donne une liste des pages pertinentes pour cet utilisateur.

1.2.2.4. Broadway-QR Aide au Raffinement de la Requête

L'approche Broadway-QR [Kanawati 1999] repose sur l'approche Broadway précédente pour opérer un raffinement des requêtes (QR Query Refinement) dans le contexte d'un méta-moteur CBKB (Constraint Based Knowledge Brokers). Les sessions passées de raffinement de requêtes sont réutilisées et on cherche à éviter les échecs qui se sont déjà produits.

L'apprentissage des expériences est fait selon le raisonnement à partir de cas. Le cas dans cette approche Broadway-QR est composé de :

- *problème/situation comportementale*. C'est la description du comportement d'un utilisateur à un instant de référence dans sa session recherche. Le comportement est constitué essentiellement des dernières spécifications des requêtes, les ensembles de documents (et leurs contenus) retrouvés, les évaluations que l'utilisateur donne de ces documents ainsi que l'estimation du système de la satisfaction globale de l'utilisateur pour chaque requête et enfin, pour chaque requête, deux listes de termes. L'une contient les termes qui apparaissent dans les documents pertinents, l'autre contient les termes qui apparaissent dans les documents non-pertinents.
- *Solution*. C'est une recommandation positive ou négative. La positive recommande les requêtes qui, selon l'expérience, amélioreraient la satisfaction de l'utilisateur ; tandis que la négative montre une requête qui a mené à un échec. Dans les deux cas, une recommandation se compose d'une configuration de requête. Une recommandation négative est la première requête négativement évaluée après l'instant de référence. Tandis qu'une recommandation positive est la dernière requête positivement évaluée dont tous les prédécesseurs ont été également positivement évalués.

Ainsi, on réutilise les solutions positives. Les solutions négatives sont utilisées de façon à être sûr que la solution proposée par le système ne va pas conduire à un échec. Donc, on recommande à l'utilisateur une requête (solution positive). Dans le cas où plusieurs requêtes (solutions positives) sont possibles, celle qui est la plus proche de sa dernière requête est recommandée, afin de ne pas disperser l'utilisateur par une requête trop différente de la sienne.

1.2.2.5. Discussion

Des deux approches Broadway et Broadway-QR que nous venons de présenter, plusieurs idées intéressantes pour la suite de notre travail peuvent être dégagées :

- La recommandation faite à l'utilisateur est basée sur son comportement qui est évalué par les requêtes soumises et/ou les références évaluées.
- Le fait de recommander une requête dans Broadway-QR, devrait amener l'utilisateur à l'utiliser, une redondance de requête est donc possible et même probable. On peut l'éviter si l'utilisateur a accès aux résultats obtenus par cette requête.
- Dans l'aide à la navigation, des références sont recommandées et dans le raffinement de requête c'est une requête qui est recommandée. Il est intéressant d'aider l'utilisateur dans les deux cas.
- Dans le raffinement de requête, c'est une requête qui a obtenu un bon résultat et qui est similaire à la requête émise qui est recommandée à l'utilisateur. On peut, donc, considérer

cela comme une proposition de requête selon deux critères : son efficacité et sa ressemblance à la requête de l'utilisateur. Nous pensons que d'autres critères de choix sont possibles et qu'ils peuvent s'avérer intéressants.

- Dans le raffinement de requête des variables sont définies pour trouver des sessions similaires afin de déduire que les objectifs de ces sessions sont les mêmes ou similaires. Pour ce qui nous concerne ce problème n'existe pas car nous ne nous intéressons qu'aux utilisateurs ayant le même sujet de recherche.

1.2.2.6. Visualisation du Contexte Social d'Environnement d'Interactions Usenet

Xiong **[**Xiong 1998] et ses collègues proposent des visualisations qui représentent le contexte des connexions sociales entre des gens à travers les liens d'interactions réseau (comme le Web, Usenet, les groupes de discussion, les listes de courriel …). En utilisant des méthodes graphiques, ces liens sont représentés par quatre diagrammes différents (Objet à Objet, Objet à Personne, Personne à Personne, et Personne à Objet) qui montrent les intercommunications entre les personnes et les objets.

Les auteurs ont appliqué ces diagrammes à Usenet (groupes/forums de discussion). Dans la suite, on présente chacune de ces visions avec son application sur Usenet :

- Diagramme Objet à Objet. Il permet de voir les rapports entre les objets créés par les activités des utilisateurs. L'application de ce diagramme sur Usenet donne deux visions :
  - La vision inter-groupe (*Inter-Group View*) montre tous les groupes de discussion liés (par des messages envoyés à plusieurs groupes par exemple) à un groupe de discussion sélectionné, comme l'illustre le diagramme de la Figure 1.3.a.
  - La vision inter-fils de discussion (*Inter-Thread View*) où deux fils de discussion sont proches s'ils partagent plusieurs messages entre eux, cette vision montre quels sont les sujets de discussion qui sont liés.
- Diagramme Objets à Personnes. Il permet de visualiser quelles sont les personnes liées aux objets. La Figure 1.3.b donne la vision fils de discussion à expéditeurs (*Thread-to-Poster View*). Cette vue permet de trouver les personnes qui s'intéressent au même sujet et qui partagent le même centre d'intérêt.
- Diagramme Personnes à Objets. Il visualise les objets liés aux personnes.
- Diagramme Personnes à Personne. Il permet de visualiser comment les gens sont liés entre eux en se basant sur les objets. Ainsi, la vision inter-expéditeurs (*Inter-Poster View*) qui montre tous les expéditeurs qui partagent les mêmes fils de discussion. Cette vue montre qui répond à qui. Ceux qui sont connectés avec beaucoup d'autres personnes et, donc, qui interagissent avec beaucoup de personnes, font probablement partie du noyau de groupe.

De telles visualisations aident les utilisateurs à avoir conscience de la structure d'un groupe/forum de discussion et à comprendre leurs centres d'intérêt, et leurs relations. Donc, ces visions peuvent mieux éclairer les utilisateurs sur le contexte d'interaction, et rendre les intercommunications entre les groupes plus riches.

Les auteurs suggèrent d'appliquer cette technique sur différentes sortes d'information collaborative comme celle résultant du filtrage collaboratif, ainsi le diagramme Objet à Objet peut montrer quelles ressources sont proches en se basant sur les préférences des utilisateurs,

le diagramme Personnes à Objets peut également permettre d'effectuer un filtrage collaboratif *visuel* en montrant directement les ressources préférées par des autres personnes.

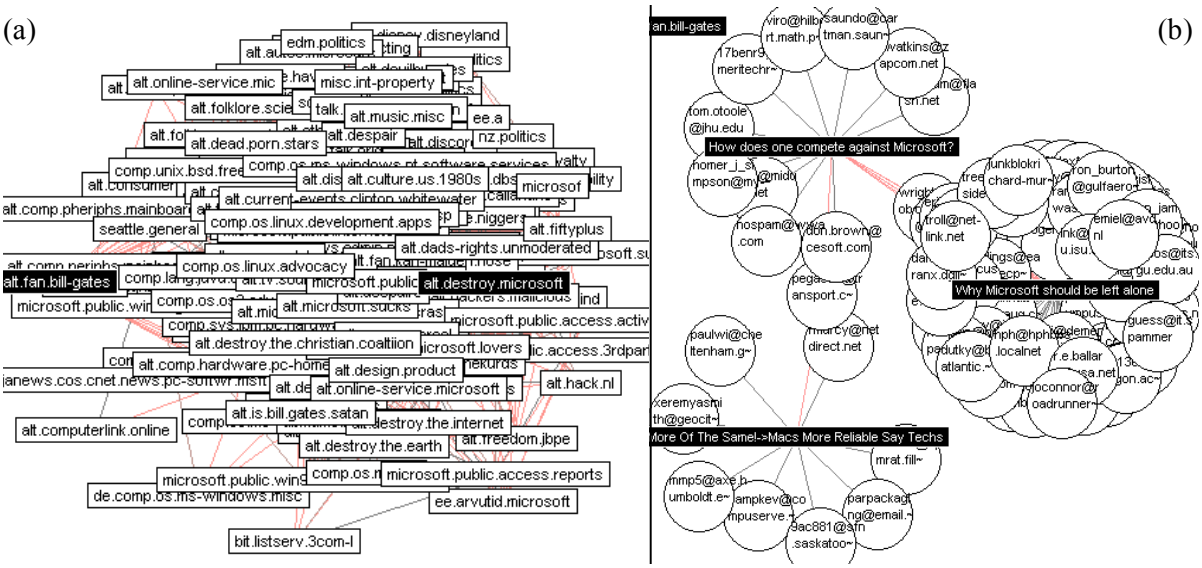

Figure 1.3. *(a) Vision entre-groupes. (b) Vision fils_de_discussion-à-expéditeurs.*

Ce travail s'intéresse à l'aspect collaboratif dans le contexte social d'interaction en ligne, aux relations selon deux dimensions les personnes et les objets à l'importance d'avoir conscience du contexte d'interaction produit par les intercommunications entre les personnes. Nous pensons qu'une telle interface serait intéressante dans le contexte de la recherche collaborative pour aider davantage l'utilisateur par une visualisation adaptée à ce contexte : nous définissons des relations entre les objets et les personnes, dans ce cas un objet peut être une référence retrouvée, une requête soumise, un terme utilisé dans une requête ou d'autres objets faisant partie des éléments de la recherche. Ainsi, nous pouvons utiliser les différents diagrammes pour visualiser le processus de recherche. Par exemple, un diagramme termes-utilisateurs qui montre les utilisateurs qui ont utilisé le même terme dans leurs requêtes, un diagramme requête-requête qui montre les requêtes qui sont similaires ou différentes en se basant sur les termes qui y apparaissent ou en se basant sur les documents qui apparaissent dans leurs résultats, un diagramme utilisateurs références qui montre toutes les références retrouvées et jugées pertinentes par chaque utilisateur. Nous pouvons imaginer d'autres diagrammes qui permettent la visualisation de processus de recherche selon différents critères, ces critères définissant les liens représentés dans les diagrammes afin d'aider l'utilisateur dans sa tâche de recherche.

1.2.3. *Soutien pour le Partage et la Visualisation du Processus de Recherche*

Le partage, la visualisation et la sauvegarde du processus de recherche sont pris en compte dans les trois approches que nous allons présenter. Nous présentons d'abord l'interface graphique ARIADNE qui s'intéresse à une meilleure visualisation des requêtes et de leurs résultats dans le dessin de faciliter la compréhension et la navigation dans une base de données (paragraphe 1.2.3.1). Nous présentons ensuite, un environnement de recherche d'information collaborative CIRE qui permet le partage des requêtes et des résultats au sein d'une équipe (paragraphe 1.2.3.2). Enfin, nous présentons le système Collaborative Spider qui

permet non seulement le partage et l'annotation des sessions de recherche précédentes mais aussi le partage de l'analyse linguistique des sites Web retrouvés (paragraphe 1.2.3.3).

1.2.3.1. ARIADNE

Le système ARIADNE (Annotatable Retrieval of Information And Database Navigation Environment.) [Twidale 1996] est une interface graphique pour la visualisation du processus de recherche. Il fournit un soutien automatisé par ordinateur pour la collaboration entre des utilisateurs qui cherchent dans une base de données (bibliothèque numérique/ électronique). Le soutien collaboratif passe par la représentation du processus de recherche. L'objectif du système est d'aider l'acquisition collaborative de compétences pour la navigation dans une base de données.

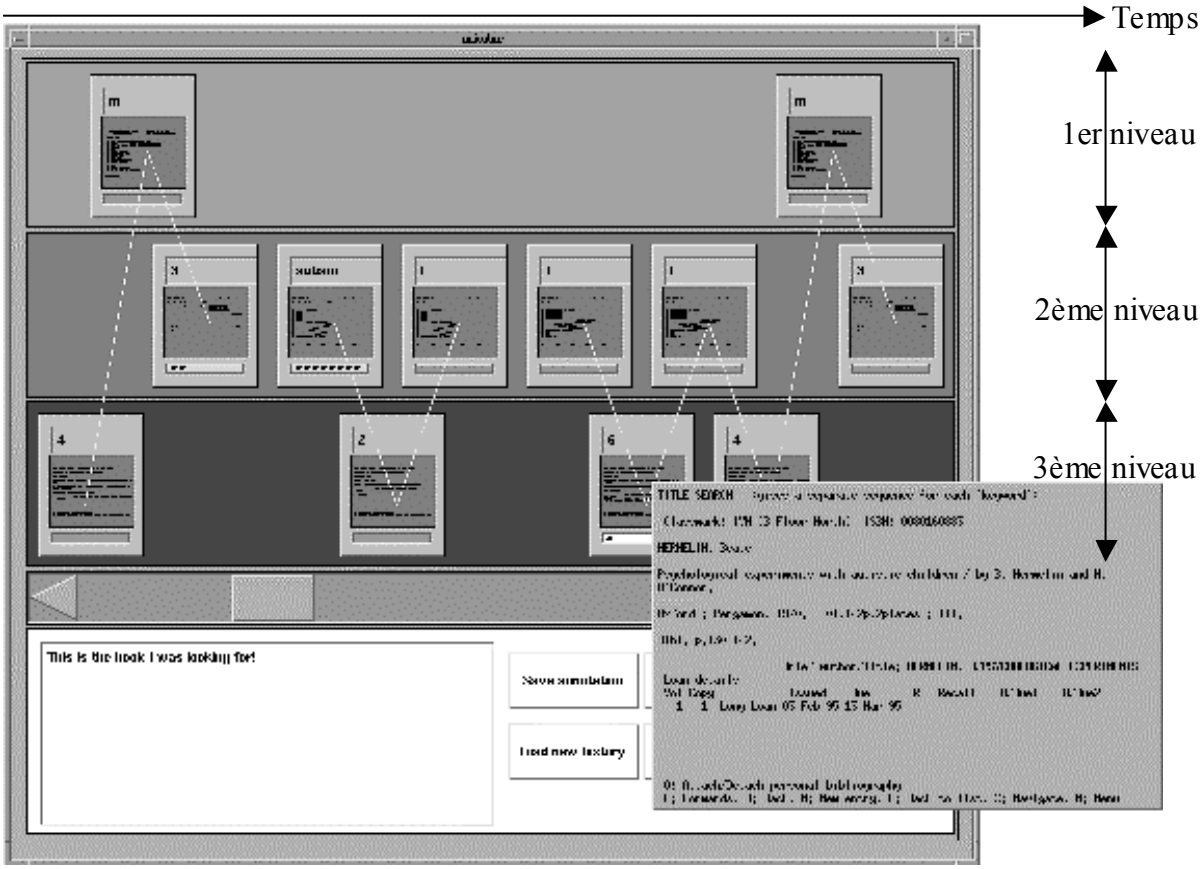

Figure 1.4. *L'interface de ARIADNE, une visualisation de recherche dans ARIADNE.*

Le système enregistre les requêtes et leurs résultats, puis produit une visualisation du processus de recherche qui permet aux utilisateurs de réfléchir, et de partager des techniques sophistiquées de recherche. La visualisation du processus de recherche est illustrée dans la Figure 1.4. Elle est composée d'une séquence de cartes qui donnent une description rapide des commandes de recherche en entrée, et des écrans de résultats en sortie. Les cartes peuvent être expliquées en cliquant sur elles. Le défilement de séquence de cartes, lisibles de gauche à droite permet de consulter un historique dans le temps, des activités menées pendant la session de recherche. Des informations sur la sémantique des actions sont données par la position verticale des cartes : au premier niveau se trouvent les choix du menu principal, les

requêtes de recherche sont placées au niveau intermédiaire, et les actions concernant la visualisation de résultats sont au dernier niveau. Nous soulignons les points suivants :

- L'interface dans ce système soutient la navigation dans une base de données, ce que nous visons est un espace de recherche beaucoup plus large puisqu'il s'agit des informations disponibles sur Internet. De plus, cette interface a pour but unique d'apprendre à naviguer tandis que l'apprentissage n'est qu'un de nos objectifs parmi d'autres comme, par exemple, l'amélioration de l'efficacité de la recherche que nous avons mentionnés dans l'introduction.
- Notre approche est cependant proche de l'interface proposée, si on se place du point de vue de la visualisation de la recherche des autres. Néanmoins, la visualisation fournie par ARIADNE est trop détaillée et de ce fait les utilisateurs doivent prendre beaucoup de temps pour regarder les cartes et obtenir une idée générale du processus. Nous visons à fournir une présentation plus synthétique qui soit dédiée à un utilisateur particulier dans une situation particulière.

1.2.3.2. CIRE Partage d'Environnement de Recherche d'Information Collaborative

Romano et ses coauteurs [Romano 1999] ont observé des expériences d'utilisation de systèmes de recherche d'information (SRI) et de systèmes de support de groupe (GSS). Ils montrent comment ces observations permettent une autre vision de la recherche collaborative. Cela les a amené à développer un prototype qui fusionne les paradigmes de ces deux domaines SRI et GSS dans un environnement de recherche d'information collaborative CIRE.

Ils ont conçu le système CIRE qui fournit aux membres d'une équipe les mêmes fonctions que les systèmes RI individuels et étend celles-ci avec des fonctions de recherche collaboratives. Plus précisément CIRE fournit les fonctions suivantes :

- La fonction de recherche. L'utilisateur soumet une requête, le système interagit avec un moteur de recherche commercial d'Internet (Alta-Vista) et présente le résultat de cette requête à l'utilisateur, ce dernier peut commenter et évaluer les pages retrouvées. Toutes ces informations sont enregistrées dans une base de données.
- La fonction de re-exécution des requêtes stockées. Cette fonction met à jour un ensemble de nouvelles références retrouvées. C'est-à-dire qu'elle permet aux utilisateurs de savoir quelles sont les pages que la requête retrouve mais qui n'avaient pas été trouvées lors de la dernière soumission de la requête. Les utilisateurs ont la possibilité d'intervenir sur ces requêtes et de les modifier en éditant simplement leurs chaînes, avant de les soumettre.
- Les utilisateurs peuvent voir les requêtes de l'équipe, si les requêtes sont annotées, les annotations sont également présentées. Ils peuvent aussi demander à voir la liste des pages que l'équipe a visitées pendant la navigation dans les résultats de la recherche. Les utilisateurs peuvent soumettre des annotations, répondre aux annotations des autres, ou soumettre une évaluation de la page en terme de pertinence avec la tâche de recherche.

La conception du système, et son amélioration (raffinement) sont basées sur des expériences d'utilisateurs avec le prototype. Ainsi, les expériences d'utilisateurs avec le premier prototype ont conduit à des améliorations et à l'ajout de certaines fonctions qui ne pouvaient pas être trouvées sans disposer d'un système et de son utilisation par des

utilisateurs[2]. En se basant sur les expériences d'utilisateur avec le prototype final les auteurs estiment que le système est très utile potentiellement pour la recherche d'information collaborative, et que les équipes deviennent plus performantes en l'employant.

1.2.3.3. Collaborative Spider Partage de Recherche et d'Analyse Linguistique

Chau et les autres [Chau 2003] relatent le développement d'un système multi-agents appelé Collaborative Spider. Il fournit une analyse des références retrouvées (*post-retrieval analysis*). Il permet une collaboration basée sur le partage des sessions de recherche entre les utilisateurs et la possibilité de les annoter afin d'améliorer l'efficacité de la recherche.

L'architecture du système Collaborative Spider s'articule autour de trois agents. Chaque utilisateur a son propre agent personnalisé *agent utilisateur* (*User Agent*). Il est principalement responsable de la recherche de pages sur le Web, de leur analyse, et des interactions avec l'utilisateur. Les utilisateurs d'un même groupe (par exemple, tous les participants d'un projet de recherche ou tous les membres d'une équipe de conception d'un nouveau produit) partagent deux agents : l'*agent collaborateur* (*Collaborator Agent*) qui facilite le partage d'information entre différents agents utilisateur, et l'*agent répartiteur* (*Scheduler Agent*) qui effectue une surveillance afin d'éviter d'avoir simultanément plusieurs recherches et analyses sur le même site. Cet agent peut également, à la demande des utilisateurs, effectuer des visites régulières à certains sites Web, et y effectuer de nouvelles recherches afin d'être au courant des nouveautés.

Nous nous intéressons dans ce qui suit à la façon dont sont mis en œuvre dans ce système les mécanismes de recherche et d'analyse ainsi que ceux de collaboration.

- **Recherche et Analyse** L'utilisateur commence une session en choisissant son ou ses centre(s) d'intérêt qui définissent son profil parmi ceux proposés. Ensuite, il fournit des URLs de départ/lancement (*starting URLs*) et des mots clés, que le système utilise afin d'effectuer une recherche en largeur (*breadth-first search*) sur les sites Web indiqués par l'utilisateur. Les pages Web retrouvées sont présentées à l'utilisateur selon une structure hiérarchique. Elles sont analysées afin d'extraire de chacune les groupes nominaux que l'utilisateur peut visualiser selon la fréquence de leurs occurrences. Si l'utilisateur le désire, les pages peuvent être automatiquement organisées et catégorisées selon la fréquence des occurrences de ces groupes nominaux. Une fois qu'une session de recherche est accomplie, elle est stockée afin d'être partagée avec d'autres utilisateurs.
- **Collaboration**. L'utilisateur peut accéder aux sessions d'autres utilisateurs dont le(s) centre(s) d'intérêt est/sont le(s) même(s) que le(s) sien(s). Il peut naviguer dans leurs sessions et visualiser les résultats de recherche détaillés et leurs analyses. Il peut réutiliser

[2] Par exemple, dans le premier prototype il y avait une fenêtre intermédiaire pour l'évaluation, puis le prototype a été amélioré en intégrant cette fenêtre de façon que l'utilisateur puisse voir le contenu de la page, et la commenter et évaluer dans la même fenêtre. Le premier prototype enregistre seulement les pages qui sont listées dans le résultat de moteur de recherche, puis, une description complète des pages visitées a été ajoutée en enregistrant tous les pages visitées et non seulement celles qui sont listées dans le résultat de moteur de recherche.

certains URLs de départ ou certains mots clés des sessions passées, ou même charger la session complète d'un autre utilisateur. Il a aussi la possibilité d'ajouter des commentaires à sa propre session ou à celle d'un autre. Ces annotations restent accessibles aux autres utilisateurs.

Les expérimentations du système montrent que l'efficacité de la recherche des utilisateurs qui ont accès à un nombre limité des sessions précédentes (une ou deux) est moins bonne si on la compare à celle d'utilisateurs qui n'ont aucun accès à d'autres sessions. Cependant, l'efficacité devient significativement meilleure quand les utilisateurs ont accès à un plus grand nombre de sessions de recherche. Ce qui indique que le gain de la collaboration dans l'analyse et la recherche sur le Web n'est acquis qu'après avoir partagé un certain nombre de sessions.

Une autre approche qui s'intéresse aussi à l'analyse de l'espace d'information doit être mentionnée, c'est celle proposée dans [Neches 1998]. Les auteurs ont développé des outils d'analyse collaborative de l'espace d'information au sein de projet DASHER (Defense Acquisition Services for High Performance Electronic Commerce). Ces outils aident les utilisateurs à structurer, catégoriser et caractériser les contenus et les thèmes de l'ensemble des documents obtenus par la navigation ou par la recherche d'information. Ainsi les utilisateurs partagent un espace d'information et ils peuvent collaborer à distance afin de le consulter, de l'organiser et de l'analyser. Cette approche est une aide au partage du produit de la recherche parce qu'elle permet la classification des résultats.

1.3. Conclusion

Les considérations sur le travail de groupe et sur les fonctions que le groupware peut apporter, nous permettent de positionner notre approche dans la taxinomie du groupware selon les critères que constituent la classification de la situation de travail et la classification fonctionnelle des collecticiels dans le paragraphe 6.1.3. En ce qui concerne les approches collaboratives pour la recherche d'information que nous avons étudiées dans ce chapitre le Tableau 1.5 en fournit un classement selon qu'ils sont synchrones ou asynchrones.

Synchrone	Asynchrone
• MOO-Gopher • C-TORI • VR-VIBE • écran de visualisation. • ARIADNE	• CORA • Brodway • Brodway-QR • CIRE • Collaborative Spider • DASHER

Tableau 1.5. *La position de différentes approches dans la classification temporelle.*

Nous nous sommes intéressés dans ce chapitre à différents types de soutien offert par un certain nombre de systèmes. Nous reprenons dans la suite les différentes conclusions critiques auxquelles nous sommes arrivés en présentant les différents types de soutien fournis par les systèmes que nous avons analysés afin d'en tirer des perspectives pour notre propre travail.

- Dans le soutien pour les interactions pendant le processus de recherche, l'utilisateur passe du temps à interagir avec les autres, donc il y a une perte de temps afin de tirer profits de la collaboration.
- Dans le soutien pour le partage de produit de recherche par la recommandation, l'utilisateur est considéré en terme de son profil ou de son comportement. Ce profil n'est basé que sur les requêtes et les références sélectionnées alors que d'autres critères comme la vitesse et la compétence pourraient être intéressants. De plus, le comportement de l'utilisateur est exploité afin de trouver d'autres utilisateurs qui ont un comportement semblable. Mais, il pourrait être aussi intéressant de considérer le comportement de l'utilisateur afin de trouver les utilisateurs qui complètent les manques, que l'utilisateur qu'on veut aider peut avoir, dans ses compétences afin de le faire profiter de leurs expériences.
- Dans le soutien pour le partage du processus de recherche l'utilisateur a un accès aux recherches des autres, et c'est à lui seul (sans aide du système) de tirer profit du travail des autres auquel il a accès et de faire l'effort cognitif pour prendre ce qu'il peut lui servir. De plus, la visualisation du processus ne prend pas en compte l'utilisateur.

A cause de ces limites nous orientons notre travail sur le soutien des utilisateurs pour le partage du processus de recherche et de sa visualisation. Mais nous voulons que la visualisation soit « intelligente », « synthétique » et « personnalisée » afin de pourvoir effectivement guider l'utilisateur dans sa recherche. D'un coté, nous voulons veiller à ce que l'utilisateur garde les avantages qu'on peut lui fournir en soutenant ses interactions personnelles mais sans pour autant qu'il perde du temps. D'un autre coté, nous voulons diminuer la charge cognitive qui lui est demandée dans le cas de soutien pour le partage de processus de recherche. Nous voulons également prendre en compte les utilisateurs non seulement en terme de profil et de comportement semblables comme c'est le cas dans la recommandation mais aussi, en utilisant d'autres facteurs comme par exemple, la variété de leurs compétences ou dans certains cas les relations sociales existant entre eux. Enfin, nous pensons qu'il est important de personnaliser le soutien selon le besoin de chaque utilisateur.

CHAPITRE.2. Pertinence

La notion de pertinence est centrale dans la recherche d'information et elle est très importante dans notre étude, parce que nous utilisons et exploitons cette notion pour aider les utilisateurs, et ce chapitre a pour but de la clarifier. Nous y expliquerons également le concept d'*évaluation de pertinence* ou *jugement de pertinence* ainsi que la notion de *qualité* ou *efficacité* dans la recherche d'information.

La notion de pertinence et son évaluation seront détaillées dans le paragraphe 2.1. Le paragraphe 2.2 présentera les différentes mesures utilisées et proposées pour évaluer l'efficacité de recherche. Le paragraphe 2.3 s'attachera à expliquer la pertinence dans un environnement collaboratif, en particulier, lorsque des références sont évaluées par plusieurs utilisateurs. Les questions suivantes se posent alors : comment évaluer l'accord ou le désaccord entre les juges ? Nous conclurons dans le paragraphe 2.4 les points intéressants pour notre travail.

2.1. Pertinence (Évaluation des Informations)

L'étude dans [Mizzaro 1998a] clarifie le concept de pertinence et précise qu'il y a plusieurs sortes de pertinence, ces différentes pertinences peuvent être classifiées dans un espace à quatre dimensions ou ensembles : l'*ensemble de problématique* (paragraphe 2.1.1), l'*ensemble d'information* (paragraphe 2.1.2), l'*ensemble de composants* (paragraphe 2.1.3) et l'*ensemble des points de temps* (paragraphe 2.1.4). Une telle classification est présentée dans le paragraphe 2.1.5. Le paragraphe 2.1.6 présente quelques critères selon lesquels la pertinence peut être évaluée ou jugée et les différentes expressions de ce jugement de pertinence.

2.1.1. *Ensemble de Problématique* (*Problem Set*)

Les interactions de l'utilisateur avec un système de recherche d'information sont schématisées dans la Figure 2.1.

L'ensemble de la problématique *EP* est composé des quatre éléments suivants [Mizzaro 1998a] :

$EP = \{ P, BI, RU, RS \}$

- La problématique *P* : l'utilisateur qui interagit avec le système de recherche d'information a une problématique à résoudre, ou un objectif à atteindre.
- Le besoin d'Information *BI* : l'utilisateur connaît sa problématique, et construit une représentation mentale de sa problématique. La « perception » est la transition qui permet de passer de la problématique à l'expression d'un besoin d'information. Cette perception est souvent difficile, parce que l'utilisateur cherche souvent des informations dans un

domaine qu'il ne connaît pas, donc, il est probable que le besoin d'information ne présente pas correctement ou complètement la problématique.

- La Requête-Usager *RU* : est l'expression du besoin d'information à l'aide d'un langage « humain », habituellement en langue naturelle. La « expression » comme la perception peut être aussi difficile, d'une part, parce que l'utilisateur exprime son besoin d'information à l'aide d'une requête brève et incomplète souvent en terme d'étiquettes, ou mots-clés et non en tant que déposition/déclaration/description complète. Or d'autre part, à cause de la grande variété des termes pour exprimer le même concept. Ainsi, il y a une grande variété entre les termes utilisés dans les documents et les termes utilisés dans la requête, cela est dû à l'existence de termes ambigus et des synonymes dans la langue naturelle.
- La Requête-Système *RS* est la traduction ou la formalisation de la requête de l'utilisateur sous une forme compréhensible par le système de recherche d'information, c'est-à-dire en langage « système ». La requête-système est le résultat de cette formalisation. La formalisation n'est pas simple puisque le langage « système » peut de ne pas être facile à comprendre par l'utilisateur, par exemple, la formalisation d'une requête booléenne révèle une difficulté.

Donc, la perception, l'expression et la formalisation sont des étapes qui demandent un effort cognitif à l'utilisateur.

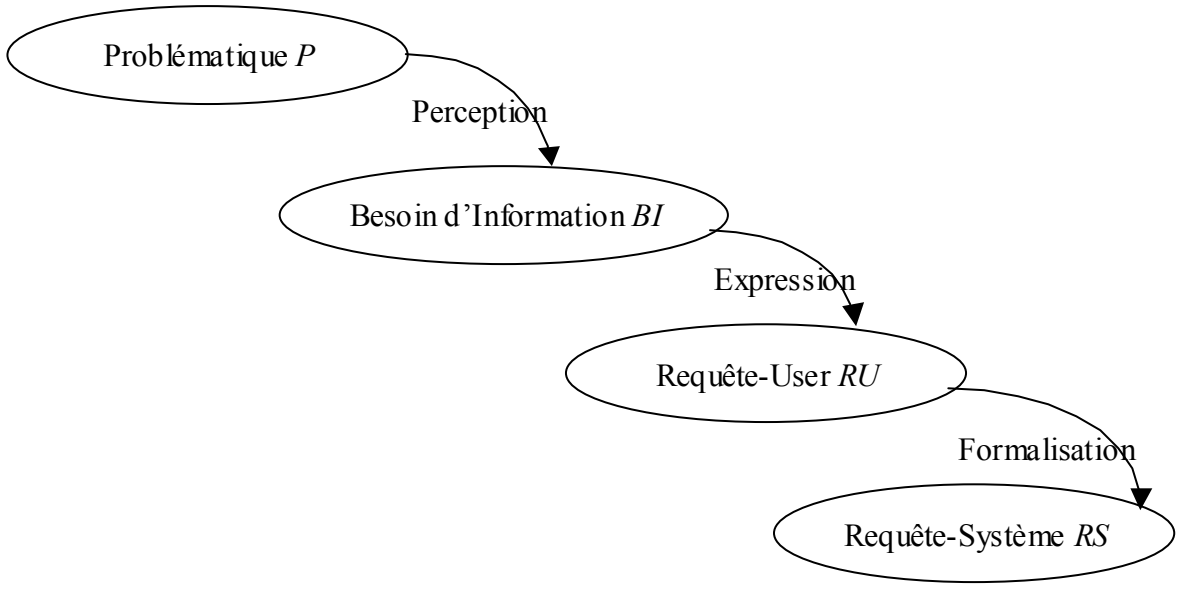

Figure 2.1. *Interactions entre un utilisateur et un système de recherche d'information.*

2.1.2. *Ensemble d'Information* (*Information Set*)

L'ensemble d'information *EI* est présenté par le triplet :

$EI = \{ S, D, I \}$

Où :

- Substitut (*Surrogate*) *S* : est la représentation d'un document pour un système de recherche d'information. Cette représentation peut contenir un ou plusieurs des composants : titre, résumé, liste de mots clés ou d'identificateurs, une citation bibliographique qui normalement contient le(s) nom(s) d'auteur(s), la date et le lieu de la publication, des extraits de document, typiquement la première et la dernière phrase ; etc.
- Document *D* : est l'entité physique que l'utilisateur d'un système de recherche d'information obtiendra après sa recherche.
- Information *I* : est l'entité (non physique) que l'utilisateur perçoit ou crée en lisant un document.

La pertinence peut être vue comme une relation entre deux entités, l'une appartenant à l'ensemble de problématique *EP* et l'autre à l'ensemble d'information *EI*. Par exemple, la pertinence d'un substitut *S* avec une requête-système *RS*, ou la pertinence d'information *I* reçue par l'utilisateur au besoin de l'information *BI* sont deux exemples de ces relations. Par conséquent, une pertinence peut être assimilée à un point dans un espace bidimensionnel. Ce ne sont cependant pas toutes les pertinences possibles, puisque deux dimensions ou ensembles supplémentaires doivent être pris en considération : les composants et le temps.

2.1.3. *Ensemble de Composants* (*Components Set*)

Les ensembles *EP* et *EI* mentionnés ci-dessus peuvent être décomposés en trois entités :

- Thème ou sujet T_h : c'est le domaine auquel l'utilisateur s'intéresse.
- Tâche T_a : c'est l'activité que l'utilisateur va exécuter avec les documents retrouvés, par exemple, écrire un état d'art.
- Contexte *C* : il inclut tout ce qui ne concerne pas le sujet et la tâche, il contient la manière dont la recherche et l'évaluation des résultats sont faites, le temps et/ou l'argent disponibles pour la recherche, etc.

Par conséquent, un substitut (un document, de l'information) est approprié à une requête-système (requête-usager, besoin de l'information, problématique) en ce qui concerne un ou plusieurs de ces composants.

La pertinence peut prendre en compte non seulement un seul composant mais plusieurs, alors cet ensemble n'est plus défini par $\{T_h, T_a, C\}$ mais par l'ensemble de composants C_o défini ainsi :

$$C_o = \big\{\{T_h\}, \{T_a\}, \{C\}, \{T_h, T_a\}, \{T_h, C\}, \{T_a, C\}, \{T_h, T_a, C\}\big\}$$

2.1.4. *Ensemble des Points de Temps* (*Time Points Set*)

L'auteur dans [Mizzaro 1998a] a remarqué que :

1. souvent, la première requête-système ne donne pas le résultat désiré, à cause de la difficulté rencontrée par l'utilisateur lors des étapes mises en évidence précédemment : « perception », « expression » et « formalisation ». D'autres requêtes-système sont alors nécessaires.

2. la requête-usager peut être modifiée, par exemple, quand l'utilisateur réalise que la première expression de son besoin d'information n'est pas correcte ou est incomplète.
3. le besoin d'information peut changer durant l'interaction avec le système de recherche d'information, parce que l'utilisateur peut percevoir sa problématique de façon différente, au fur et à mesure qu'il examine les réponses du système. Il y a donc un aspect dynamique qui intervient d'où l'aspect temporel.

La Figure 2.2 illustre la transformation dynamique de *P*, *BI*, *RU*, et *RS*, ces éléments étant représentés par quatre ellipses. Au moment $t_{P0,}$ l'utilisateur a une problématique P_0, il formule le besoin initial de l'information b_0 au moment t_{b0}, il l'exprime, et il obtient la requête-usager initiale r_0 au moment t_{r0}, et il la formalise, ce qui donne la requête-système initiale q_0 au moment t_{q0}. Alors, une révision a lieu : la requête-système initiale peut être modifiée, on obtient q_1 au moment t_{q1}, la même chose peut se produire pour la requête-usager et le besoin de l'information, jusqu'à obtenir, finalement, le besoin d'information b_p au moment t_{bp}, la requête-usager r_m au moment t_{rm}, et la requête-système q_n au moment t_{qn}. Ce schéma montre les différentes évaluations pouvant intervenir dans une session de recherche aussi bien pour le besoin d'information que pour la formulation des requêtes systèmes.

En prenant en compte la dépendance temporelle, on définit l'ensemble des points de temps *T* qui contient les instants signifiants, où t_f représente le moment de résolution de la problématique *P* (t_f est le moment final).

$T = \{ t_{p0}, t_{n0}, t_{r0}, t_{q0}, t_{q1}, t_{r1}, t_{q2}, \ldots , t_{bp}, \ldots, t_{rm}, \ldots, t_{qn}, t_f \}$

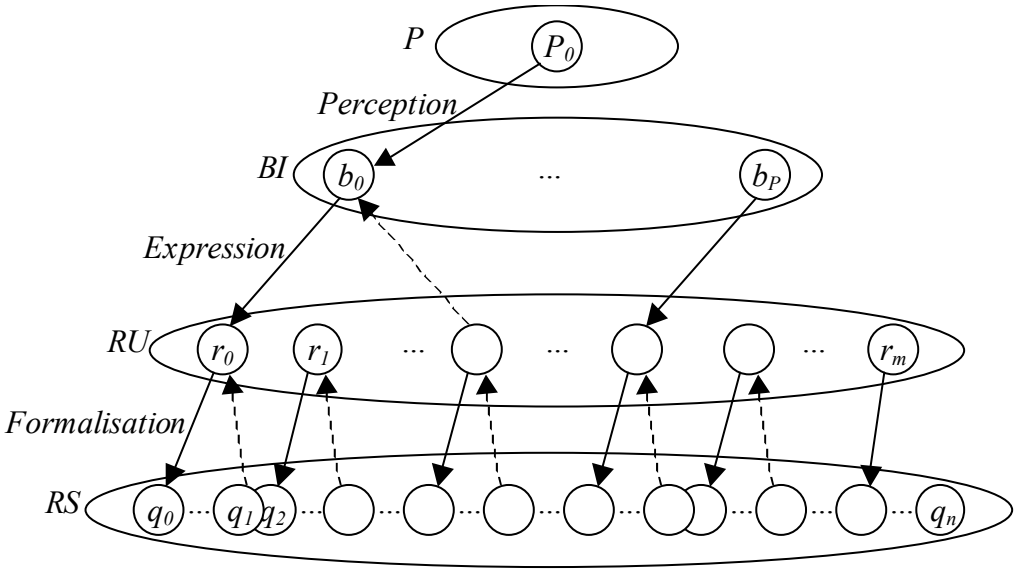

Figure 2.2. *Transformation dynamique de l'ensemble de la problématique.*

La pertinence donc est un phénomène dynamique. Pour un utilisateur, un document peut être pertinent à un instant donné et non pertinent plus tard, ou vice-versa. Il s'agit d'évaluer la « consistance » du jugement. Selon Mizzaro dans [Mizzaro 1998b] les études sur la nature dynamique du jugement de la pertinence ont mis en exergue les éléments suivants :

- le jugement de pertinence dépend du temps.
- l'état mental de l'utilisateur et la pertinence changent pendant que l'utilisateur lit le document et le jugement réel de la pertinence n'est effectué que lorsque l'utilisateur a lu le document en entier.
- l'ordre de présentation des documents peut affecter la préférence de l'utilisateur. Une étude expérimentale a montré que lorsque le système de recherche d'information retourne moins de 15 documents, l'ordre de présentation des documents n'a aucune influence, et que son importance apparaît quand l'utilisateur a plus de 15 documents à examiner.

Ces études ont des conséquences importantes pour la construction des systèmes de recherche d'information. En fait, elles mettent en évidence le besoin de systèmes de recherche d'information itératifs et interactifs, c'est à dire des systèmes qui simulent une communication d'information bidirectionnelle itérative entre l'utilisateur et le système, et qui réalisent une interaction efficace avec l'utilisateur.

2.1.5. *Classification de la Pertinence*

Les quatre ensembles (*EP*, *EI*, C_o et *T*) présentés dans les paragraphes précédents sont utilisés pour définir les différentes sortes de pertinence Figure 2.3. Il y a une pertinence pour chaque combinaison des éléments des quatre ensembles : chaque pertinence peut être vue comme un point dans un espace de quatre dimensions. Formellement, l'ensemble *Per* de toutes les sortes de pertinence peut être défini comme suit :

$$Per = EP \times EI \times C_o \times T$$

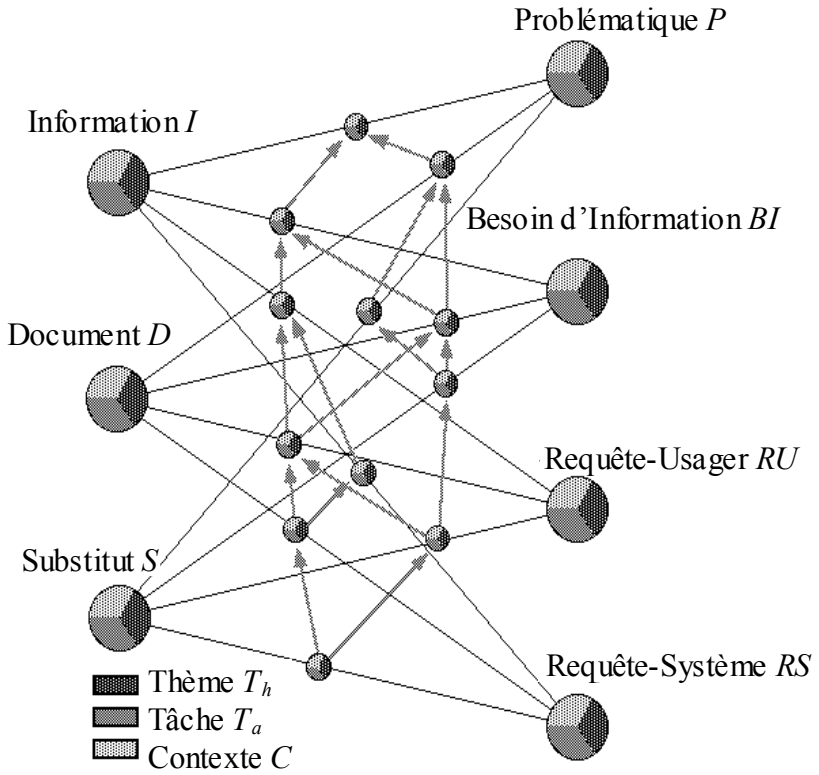

Figure 2.3. *Plusieurs sortes de pertinence.*

2.1.6. *Jugement de l'Utilisateur*

Un jugement de pertinence est le fait qu'un utilisateur affecte à un certain moment une valeur de pertinence. Voici quelques exemples tirés de [Mizzaro 1998b] des critères adoptés par les utilisateurs pour juger de la pertinence :

- Classification des critères en 7 catégories : les informations sur le contenu des documents, l'expérience de l'utilisateur, la préférence et la conviction de l'utilisateur, d'autres informations et sources sur l'environnement, les sources des documents, le document comme une entité physique, et la situation de recherche de l'utilisateur.
- Identification des critères suivants : les références, la réputation, la confiance, la compétence, les considérations financières, les considérations de temps, le style, le niveau de difficulté du document, la crédibilité, l'exactitude ou la précision, la spécificité, la fiabilité, l'accessibilité, la vérifiabilité à travers un autre, … etc., qui sont des critères sur l'évaluation de la qualité d'un document
- Classification des critères qui interviennent dans le jugement de pertinence, et leur regroupement en trois catégories :
  - le contexte interne : il contient les critères appartenant à des expériences précédentes de l'utilisateur (compétence dans le domaine, niveau d'éducation).
  - le contexte externe : c'est l'ensemble des facteurs concernant la recherche qui est faite (objectif de recherche, stade de la recherche).
  - le contexte de satisfaction : il représente les motivations et l'utilisation intentionnelle de l'information (obtenir les définitions de quelque chose).

Redman [Redman 1998] explore les dimensions de la qualité des données dans le domaine des bases de données, selon trois vues :

- les dimensions qualité à la vue conceptuelle : au niveau des items individuels des données la vue conceptuelle fournit le contexte dans lequel l'item de données doit être interprété, au niveau application, cette vue correspond au modèle du monde réel capturé par les données. Les dimensions qualité sont le contenu, la portée (qui fait référence à la capacité d'une vue à renfermer assez des données pour satisfaire les besoins de toutes les applications), le niveau de détail, la composition (qui fait référence à la structure interne par exemple l'identification : chaque entité est facilement identifiée et de manière unique), la cohérence sémantique et structurelle des divers composants, et la réaction au changement (c'est-à-dire l'aptitude à s'accommoder des changements comme l'ajout d'une valeur, la modifier…).
- les dimensions qualité des valeurs des données comprennent l'exactitude, la complétude (qui fait référence au degré de présence des valeurs dans les collections de données), l'actualisation et la dimension corrélées, et la cohérence des valeurs.
- les dimensions qualité de la présentation des données consistent de l'adéquation, l'interprétabilité, la portabilité, la précision des formats, la flexibilité, l'aptitude à représenter les valeurs nulles, l'utilisation efficace des médias d'enregistrement, la cohérence des représentations.

Le jugement d'un document peut s'effectuer de deux façons [Mizzaro 1999] :

1. soit par l'affectation d'une valeur numérique indiquant la pertinence (score de jugement) (paragraphe 2.1.6.1).
2. soit en ordonnant ou en classant les documents (paragraphe 2.1.6.2).

Détaillons maintenant deux façons de juger de la pertinence d'un document.

2.1.6.1. Score de Jugement

L'utilisateur que l'on appelle également le juge assigne une valeur, ou un score à chaque document de l'ensemble obtenu. Il y a trois types de notation possibles :

- *Une estimation binaire* (*standard dichotomous*) : le juge décide si le document est pertinent ou non-pertinent (oui/non, 1/0).
- *Une estimation selon une échelle de valeurs* (*category rating scales*) : où le jugement de la pertinence est exprimé par une valeur prise d'une échelle qui contient de 3 à 11 valeurs, par exemple (non-pertinent, partiellement pertinent, pertinent). Les juges préfèrent avoir un grand nombre des catégories parmi lesquelles choisir, afin de pouvoir nuancer leur jugement.
- *Une estimation pondérée* (*magnitude estimation*) : dans laquelle chaque nombre positif et rationnel peut être utilisé. Il y a plusieurs façons d'exprimer physiquement le jugement :
  - *Par une estimation numérique* (*numeric estimation*) : où un nombre élevé indique une pertinence élevée. Habituellement, le juge assigne une valeur qui est un nombre réel dans l'intervalle [0, 1].
  - *Par un curseur* (*line-length*) : où la longueur de curseur dessinée par le juge indique la pertinence.
  - *Par la force de poignée* (*force hand grip*) : où la force est mesurée par un dynamomètre, qui indique la pertinence.

Le jugement *binaire* est un cas particulier du jugement selon une *échelle de valeur* qui est à son tour un cas particulier du jugement *pondéré*. L'estimation *pondéré*e est donc une méthode plus générale.

2.1.6.2. Jugement par Ordonnancement

Le juge n'affecte pas une valeur, mais ordonne les documents, il peut construire :

- Un ordre total : chaque document est comparable aux autres, en terme mathématique, la relation d'ordre doit être totale (voir la Figure 2.4.a). Dans ce cas le juge ne peut pas exprimer le fait qu'il ne sait pas comparer deux documents. Il y a deux cas :
  - sans égalité : la relation d'ordre est < ; aucun document ne peut avoir une pertinence égale à un autre.
  - avec égalité : les relations d'ordre sont < et = ; certains documents peuvent avoir une pertinence égale à d'autres documents.
- Un ordre partiel : il n'est pas nécessaire que chaque document soit comparable aux autres. La relation d'ordre est partielle (voir la Figure 2.4.b). Prenons deux documents, le juge

peut dire qu'il ne sait pas comment établir un rapport entre eux. On a également ici deux cas comme précédemment ordre partiel sans et avec égalité.

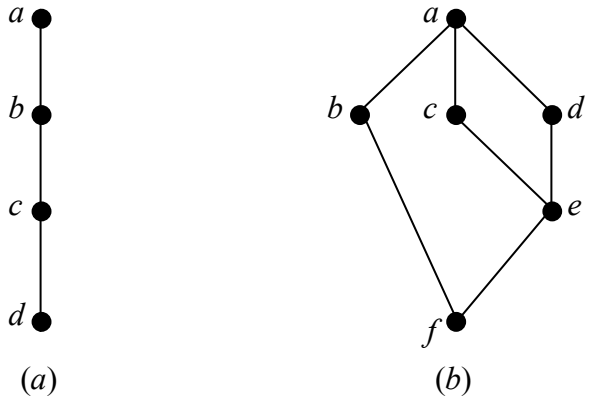

Figure 2.4. *Représentation graphique pour un ordre total (a) et un ordre partiel (b).*

2.2. Évaluation de Système de Recherche d'Information

Plusieurs méthodes et mesures sont proposées et testées pour évaluer la recherche d'information : les mesures traditionnelles comme le rappel, la précision,… etc. (paragraphe 2.2.1), et d'autres qui sont inspirées de ces mesures traditionnelles afin d'évaluer les moteurs de recherche comme la précision complète, la meilleure précision, la précision différentielle,… (paragraphe 2.2.2). et encore pour évaluer cette efficacité il y a : la mesure de durée prévue de recherche (paragraphe 2.2.3), et deux autres mesures, la première est la mesure de distance moyenne qui mesure cette efficacité en se basant sur les valeurs de pertinences données par l'utilisateur et par le système (paragraphe 2.2.4), tandis que la deuxième est la mesure de performance basée-distance qui mesure l'efficacité par rapport à la distance entre les deux ordonnancements celui de l'utilisateur et celui du système (paragraphe 2.2.5).

2.2.1. *Mesures Traditionnelles*

Dans le domaine de la recherche d'information, les mesures de l'efficacité des systèmes de recherche d'informations sont basées sur la pertinence binaire (un document est pertinent pour une requête ou non) et sur la recherche binaire (un document est retrouvé ou non), (voir la Figure 2.5) mais la pertinence n'est pas toujours binaire, et le système de recherche d'information ordonne en générale les documents retrouvés.

Selon cette notion binaire de la pertinence et la recherche, on a défini les mesures suivantes : le *rappel* et la *précision*. La *précision* (*precision*) est le ratio des documents pertinents retrouvés sur le nombre total des documents retrouvés *R*, tandis que le *rappel* (*recall*) est défini comme le ratio du nombre des documents pertinents retrouvés sur le nombre total des documents pertinents *P* dans toute la collection de documents [Berrut 1997].

$Précision = |R \cap P| / |R|$

$Rappel = |R \cap P| / |P|$

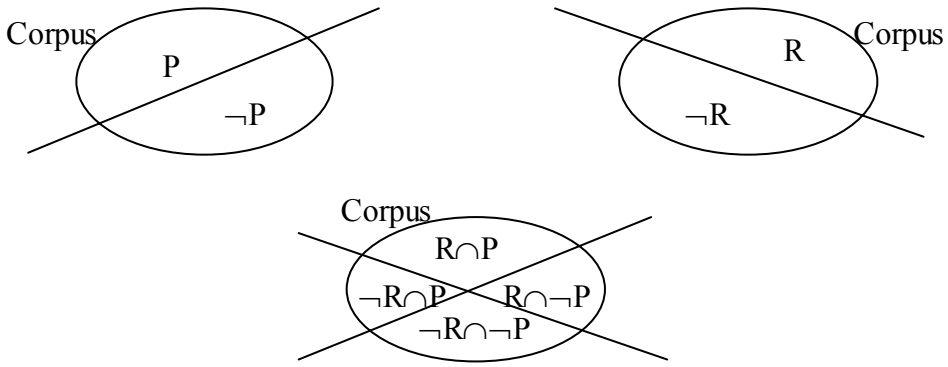

Figure 2.5. *Mesures traditionnelles* : *rappel, précision, élimination et hallucination.*

Ces deux mesures évaluent la pertinence des réponses pour l'utilisateur, ce sont des mesures dites orientées utilisateur. Il y a deux autres mesures qui évaluent l'efficacité du système à retrouver les documents pertinents et à éliminer ceux qui sont non-pertinents des réponses, on les appelle mesures orientées système : *élimination* et *hallucination*. L'*élimination* (*discrimination*) est la proportion de documents non-pertinents parmi ceux non-retrouvés. L'*hallucination* (*fallout*) est la proportion de documents non-pertinents retrouvés parmi ceux qui sont non-pertinents [Berrut 1997].

$Elimination = |\neg R \cap \neg P| \,/\, |\neg P|$

$Hallucination = |R \cap \neg P| \,/\, |\neg P|$

Afin d'effectuer l'évaluation de l'efficacité de système de recherche d'information via la précision et le rappel, il est nécessaire de disposer des trois ensembles de données suivants :

- un ensemble de documents dans lequel le système va chercher.
- un ensemble de requêtes.
- un ensemble de jugements de pertinence, c'est à dire quels documents sont pertinents pour chaque requête.

Il y des collections de test standards qui sont créées comme par exemple CACM, TREC. [Leloup 1998]. « L'organisation, TREC (Text REtrieval Conference), est soutenue par le NIST (National Institute Standards Technology) et le DARPA (Defense Advanced Research Projects Agency), deux organisations américaines. Le but est d'encourager la recherche et le développement, en fournissant des bases textuelles de grande taille (plusieurs giga-octets), des procédures communes d'analyse des taux de rappel et pertinence, et en organisant un forum permettant aux organisations de présenter et de comparer leurs résultats. Les sujets traités sont de deux types : les techniques de recherche générique et les techniques spécifiques. Les participants sont des instituts de recherche et des universités ainsi que des éditeurs logiciels. »

L'article [Reid 2000] décrit une nouvelle méthodologie d'évaluation de système de recherche d'information. Cette méthodologie basée sur le concept de collection de test orientée-tâche combine les avantages de test traditionnel non-interactif avec un accent mis sur l'aspect « orienté utilisateur ». Le principe de la collection de test orientée-tâche est d'adopter

la tâche, plutôt que la requête, comme une unité primaire d'évaluation de jugement de pertinence. La collection de test orientée-tâche consiste en :

- une collection de documents.
- un ensemble des descriptions de tâches : le Tableau 2.1 montre un exemple d'une description de tâche.
- un ensemble de requêtes par tâche.
- un ensemble de jugements de pertinence par tâche.

Selon l'auteur, la collection de test orientée-tâche est une méthode d'évaluation de systèmes de recherche d'information plus réaliste que la collection-test traditionnelle parce qu'elle reconnaît l'importance de la tâche dans la motivation de l'utilisateur.

La collection de test orientée-tâche se situe entre l'évaluation de recherche d'information orientée-système et celle orientée-usager.

information sur le rôle de la tâche à accomplir/exécuter
vous êtes un étudiant en médecine de la première année. Votre travail écrit pendant l'année comporte 5 dissertations, que vous devez passer avec la mention (50%) afin de réussir le passage à la deuxième année.
information sur le résultat de tâche (le format et le contenu demandés du résultat)
écrivez une dissertation jusqu'à 1000 mots sur les forces et les faiblesses principales de la « conception » du corps humain. La dissertation obtiendra une note, et elle sera évaluée selon le contenu et la présentation.
information sur l'accomplissement/achèvement de tâche (conseils et recommandations donnés pour achever la tâche)
Servez-vous de la bibliothèque de médicine et du Web pour trouver toutes les ressources et les références utiles.

Tableau 2.1. *Exemple d'une description de tâche.*

2.2.2. *Mesures de l'Evaluation des Moteurs de Recherche*

Les auteurs dans [Chignell 1999] essaient de répondre aux questions suivantes : comment peut-on mesurer la performance des moteurs de recherche ? la mesure de performance a-t-elle un impact sur la comparaison des moteurs ?

Ainsi, ils ont effectué deux expériences afin d'étudier l'efficacité relative de différents moteurs de recherche dans différentes conditions. Dans la première expérience, ils ont examiné les effets suivants sur le temps d'interrogation : l'effet du temps dans le jour et le jour de la semaine, et l'effet de la stratégie de la requête (générale, précise, obtention d'un rappel élevé) sur la durée de traitement de la requête, et la précision.

Dans la deuxième expérimentation, ils ont utilisé un ensemble de mesures afin d'estimer la qualité des résultats obtenus. Nous présentons ces mesures dans ce qui suit, mais tout d'abord, nous mentionnons que les auteurs n'ont considéré que les 20 premières références retrouvées par les moteurs de recherche.

2.2.2.1. Précision

a) précision « complète » (full) : cette mesure prend entièrement en compte le score subjectif affecté à chaque référence. Son calcul est représenté par l'équation suivante :

$$PrecComp\ (1, NRef_{20}) = \frac{\sum_{i=1}^{N\mathrm{Re}f_{20}} Score_i}{N\mathrm{Re}f_{20} * \max\mathrm{Re}fScore} \qquad [2.1]$$

où $Score_i$ est le score affecté à la *i*ème référence, $NRef_{20}$ = min (20, *Ref-Retourné*), *Ref-Retourné* est le nombre total des références retournées, *maxRefScore* est le score maximal qui peut être affecté à une référence. *PrecComp* est défini pour *Ref-Retourné* > 0.

b) « meilleure » précision (best) : cette mesure prend en compte seulement la référence la plus pertinente (les références ayant obtenu *Scorei* = *maxRefScore*) :

$$PrecBon\ (1, NRef_{20}) = \frac{\sum_{i=1}^{N\mathrm{Re}f_{20}} Score_i}{Score_{=\max\mathrm{Re}fScore} \cdot N\mathrm{Re}f_{20}} \qquad [2.2]$$

2.2.2.2. Précision Différentielle

La *précision différentielle* (differential precision) mesure la différence de précision entre les 10 premières références et les 10 suivantes. La mesure de précision différentielle positionne les références pertinentes à l'intérieur de 20 premières références obtenues. Il est préférable pour l'utilisateur qu'il y ait une concentration de références pertinentes dans les 10 premières références, puisque cela lui permet de trouver plus vite les informations pertinentes. La précision différentielle entre les 10 premières et les 10 secondes références est calculée comme suit :

$pd\ (1, NRef_{20}) = Prec\ (1, NRef_{10})$ - $Prec\ (NRef_{11}, NRef_{20})$

Prec peut être la précision « complète » calculé par la formule [2.1] ou la meilleure précision calculée par la formule [2.2].

- $pd > 0 \Rightarrow$ il y a plus de documents pertinents dans les 10 premières références que dans les 10 suivantes.
- $pd = 0 \Rightarrow$ le nombre de documents pertinents parmi les 10 premières références et parmi les 10 suivantes est le même.
- $pd < 0 \Rightarrow$ il y a moins de documents pertinents dans les 10 premières références que dans les 10 suivantes.

2.2.2.3. Effort Cognitif de l'Utilisateur

Ce critère permet de mesurer l'effort cognitif de l'utilisateur (user effort or search length), c'est-à-dire mesurer combien, en moyenne, de documents non-pertinents l'utilisateur doit examiner afin qu'il trouve un nombre donné de documents pertinents.

La mesure est basée sur le nombre des références :

$$FEffort_i = 1 - \frac{\max Effort_i - Effort_i}{\max Effort_i - BonEffort_i}$$

Où $maxEffort_i$ est l'effort cognitif maximal pour i références pertinentes à l'intérieure de n références retournées, $BonEffort_i$ est le meilleur (le plus petit) effort possible pour i références, $Effort_i$ l'effort cognitif pour i références les plus pertinentes. Le domaine de la fonction $FEffort_i$ est [0-1], où 0 est le meilleure, ou l'effort cognitif de recherche le plus petit.

2.2.3. *Mesures de Durée Prévue de Recherche*

Pour Dunlop [Dunlop 1997], les études dans la recherche d'information peuvent être effectuées selon deux aspects :

a) l' « aspect utilisateur » est de plus en plus pris en compte, que ce soit pour la conception d'interface d'utilisateur, pour les systèmes basés sur la navigation, ou pour comprendre la signification des termes *pertinence*, *besoin d'information* comme on a vu ci-dessus dans les études de [Mizzaro 1998a, b].

b) l' « aspect système », la rapidité et l'efficacité de l'algorithme de correspondance.

Il introduit un modèle d'évaluation qui inclut les aspects de la conception d'interface et ceux du moteur de recherche. Ainsi l'article propose une nouvelle approche d'évaluation appelée la *durée prévue de recherche* (Expected Search Duration), cette mesure permet d'analyser la performance du moteur et de l'interface. Pour cela, il a introduit deux courbes et les a prises en compte afin d'obtenir la mesure de la *durée prévue de recherche*, ces deux courbes sont :

- la courbe du nombre à voir (Number To View graph) NTV qui est le nombre de références pertinentes r contre le nombre des références que l'utilisateur doit voir en moyenne m_r pour rencontrer ces r références pertinentes.
- la courbe du temps pour voir (Time To View graph) TTV qui prédit le temps qu'il faut pour retrouver un nombre donné de références pertinentes.

2.2.4. *Mesure de Distance Moyenne MDM*

L'auteur dans [Mizzaro 2001] propose une nouvelle mesure de l'efficacité de recherche nommée mesure de distance moyenne *MDM* (Average Distance Mesure *ADM*), qui mesure simplement la distance moyenne - ou la différence - entre l'*Estimation de Pertinence par l'Utilisateur EPU* et l'*Estimation de Pertinence par le Système EPS*. Cela est traduit de façon formelle ainsi : pour une requête q, on a deux pertinences pour chaque document d_i dans la base de données D :

a) $EPSq\ (d_i)$: le *EPS* pour d_i en respect à q.

b) $EPUq\ (d_i)$: le *EPU* pour d_i en respect à q.

MDM est alors définit comme la distance moyenne entre $EPS_q\ (d_i)$, $EPU_q\ (d_i)$:

$$MDM_q = 1 - \frac{1}{|D|} \Sigma_{di \in D} |EPS_q(d_i) - EPU_q(d_i)| \quad [2.3]$$

où $|D|$ est le nombre de documents dans la base de données D. *MDM* est une valeur dans l'intervalle [0, 1] avec 0 qui représente la plus mauvaise performance. La moyenne de MDM_q de quelques requêtes donne l'efficacité du système de recherche d'information, pour l'utilisateur.

L'auteur pense que *MDM* est plus général, que le *rappel* et la *précision* :

- le *rappel* et la *précision* ne prennent pas en compte la partie $\neg R \cap \neg P$ de document non-pertinents et non-retrouvés, alors que le fait que le système n'a pas retrouvé ces documents, veut dire qu'il a estimé correctement leurs pertinences.
- le *rappel* et la *précision* sont sensibles aux seuils pris pour distinguer les différents secteurs (entre pertinent et non-pertinent, entre retrouvé et non-retrouvé).
- des petites différences entre les *EPS* peuvent conduire à une *précision* et un *rappel* très différents, tandis que des petites différences n'affectent pas *MDM*. Des grandes différences entre les *EPS* peuvent conduire à une *précision* et un *rappel* similaires, tandis que des grandes différences n'affectent pas *MDM*.

Le problème de cette nouvelle mesure est d'obtenir *EPU*, l'auteur suggère de résoudre ce problème en demandant aux utilisateurs d'exprimer leur jugement. Mais le problème le plus difficile, en se basant sur la formule [2.3], est d'avoir les valeurs *EPU* pour tous les documents dans la base de données, pour résoudre cela, il suggère aussi de demander à juger seulement les documents retrouvés ou de sélectionner au hasard les documents à juger… Ces problèmes sont similaires aux problèmes rencontrés pour calculer la *précision* et le *rappel*.

2.2.5. *Mesures de Performance Basée-Distance mpd*

Yao dans [Yao 1995], présente une autre mesure de la performance des systèmes de recherche basée sur la distance entre l'ordonnancement de l'utilisateur et celui du système. Pour cela il définit une mesure de distance entre les ordonnancements (paragraphe 2.2.5.1) puis, la mesure de la qualité de l'ordonnancement (paragraphe 2.2.5.2).

Cette mesure est différente de la mesure précédente de distance moyenne MDM. En effet, la mesure d'Yao est basée sur la relation d'ordre c'est-à-dire la distance entre deux ordonnancements des documents celui de l'utilisateur et celui du système ; Alors que MDM est basée sur les valeurs de pertinence c'est-à-dire la distance entre les valeurs de pertinence des documents celles qui sont données par l'utilisateur et celles qui sont données par le système.

2.2.5.1. Mesure de Distance entre Ordonnancements

Définissons la relation d'ordre suivante sur un ensemble de documents D :

$\succ (D) = \{(d, d') \mid d \succ d' \text{ et } d, d' \in D\}$

Soient deux relations d'ordre $\succ_1$, $\succ_2$ définies sur l'ensemble des documents D, et deux documents $d, d' \in D$. Ces deux relations de ordre $\succ_1$, $\succ_2$ peuvent être :

a) *d'accord* si toutes les deux ordonnent d, d' dans le même ordre c'est à dire :

$$\succ_1 (\{d, d'\}) = \succ_2 (\{d, d'\})$$

b) *contradictoire* si l'une met d en haut et l'autre met d' en haut.

c) *compatible* si l'une met d ou d' en haut et l'autre met d' et d en égalité.

Prenons $\Gamma(D)$ l'ensemble de toutes les relations d'ordre possibles sur l'ensemble de document D. La mesure de distance entre les ordonnancements est une fonction de valeur réelle de R :

$$B : \Gamma(D) \times \Gamma(D) \rightarrow R$$

On définit, par exemple, la distance δ entre deux ordonnancements $\succ_1$, $\succ_2$ pour un couple de documents $d, d' \in D$ ainsi :

$\delta_{\succ_1, \succ_2}(d, d') = 0$ si $\succ_1$, $\succ_2$ sont *d'accord* à propos de d, d'.

$= 1$ si $\succ_1$, $\succ_2$ sont *compatibles* à propos de d, d'.

$= 2$ si $\succ_1$, $\succ_2$ sont *contradictoires* à propos de d, d'.

La distance totale entre deux ordonnancements est calculée par :

$$\beta (\succ_1, \succ_2) = \sum_{d, d'} \delta_{\succ_1, \succ_2} (d, d') \qquad [2.4]$$

2.2.5.2. Mesure de la Qualité de l'Ordonnancement

Habituellement, le résultat du processus de recherche est une liste de documents ordonnés. Prenons $\succ_u$ l'ordonnancement de l'utilisateur, et $\succ_s$ l'ordonnancement du système. Dans la situation idéale, les deux doivent être identiques, c'est-à-dire $\succ_s = \succ_u$. Cette exigence est connue comme le critère d'*ordonnancement parfait.* Un critère moins exigeant peut être utilisé, le critère d'*ordonnancement acceptable*, selon lequel le système doit placer les documents préférés avant les documents non-préférés. Selon le critère adopté, la performance du système peut être mesurée en terme de *divergence* de $\succ_s$ ou *proximité* de $\succ_u$.

La mesure de performance peut être dérivée par l'utilisation de la distance entre $\succ_s$ et un ordonnancement acceptable de $\succ_u$. Il y a plusieurs ordonnancements acceptables en respect à $\succ_u$. Du point de vue efficacité, tous ces ordonnancements acceptables sont équivalents. Mais, afin de définir une mesure juste, on doit choisir un ordonnancement acceptable qui est le plus proche de $\succ_s$.

L'auteur suggère la Mesure de Performance basée-Distance *mpd* (Distance-based Performance Measure) suivante :

$mpd\ (\succ_u, \succ_s) = MIN_{\succ \in \Gamma u(D)}\ \beta\ (\succ, \succ_s)$

où $\Gamma_u(D)$ est l'ensemble de tous les ordonnancements acceptables de $\succ_u$, la fonction β est celle définie par la formule [2.4].

Si $\succ_s$ est un ordonnancement acceptable, alors $mpd\ (\succ_u, \succ_s) = 0$

La mesure précédente est appropriée pour comparer plusieurs systèmes par rapport à l'ordonnancement du résultat en prenant la même requête, mais il n'évalue pas de façon équivalente la performance pour plusieurs requêtes, pour cela l'auteur a défini une mesure de performance normalisée *mpdn* en terme de distance relative à la distance maximale :

$mpdn\ (\succ_u, \succ_s) = mpd\ (\succ_u, \succ_s) / MAX_{\succ \in \Gamma(D)}\ mpd\ (\succ_u, \succ)$

où $MAX_{\succ \in \Gamma(D)}\ mpd\ (\succ_u, \succ)$ est la distance maximale entre $\succ_u$ et tous les ordonnancements. La valeur *mpdn* est comprise entre 0 et 1.

2.3. Collaboration et Jugement de Pertinence

La pertinence est *subjective*, les utilisateurs (les juges) peuvent exprimer différents jugements de pertinence. Alors il est important de se poser la question suivante : les jugements de pertinence exprimés par différents juges (ou groupe de juges) sont-ils cohérents ?

Selon Mizzaro dans [Mizzaro 1998b] les études qui analysent le comportement des juges lors de l'évaluation de la pertinence ont trouvé que :

- les caractéristiques du juge affectent le jugement de la pertinence, et elles expliquent beaucoup la différence des jugements de pertinence. Par exemple, on a examiné différents groupes de juges regroupés selon les paramètres suivants : type (chercheur ou étudiants), niveau (sénior ou junior) et spécialité (bio-médicine ou science sociale, …). Le paramètre le plus important est la spécialité et il est suivi par le niveau d'étude puis le type.
- la discussion entre juges peut changer les jugements de pertinence de chacun d'entre eux et elle peut résoudre les désaccords.
- la subjectivité du jugement de pertinence est systématique et dépend de deux variables :
  - la compétence du juge et ses intérêts dans le domaine de la recherche.
  - l'ouverture du juge à l'information, c'est-à-dire l'aptitude du juge à percevoir des messages en tant que des messages informatifs.

La subjectivité de la pertinence, fait que les différents utilisateurs qui jugent la pertinence peuvent être d'accord et désaccord entre eux. Cela a amené à mesurer cet accord ou ce désaccord comme dans l'étude suivante.

2.3.1. *Mesure de l'Accord ou du Désaccord entre les Juges de Pertinence DES*

Dans l'article [Mizzaro 1999], l'auteur essaie de définir les notions d'*accord* et de *désaccord* entre les juges. Pour définir cette mesure, les différentes expressions du jugement sont prises en compte, à savoir le score de jugement, et le jugement par ordonnancement.

Dans la suite, nous présentons la mesure du *désaccord*, entre deux juges (paragraphe 2.3.1.1) et entre un groupe de juges (paragraphe 2.3.1.2). Les notations utilisées sont les suivantes :

- On indique les jugements de chaque juge par j_1, j_2, …, j_k,… En effet, un jugement ne concerne pas nécessairement un seul document (parce qu'ordonner un seul document n'a pas de sens), mais un ensemble de documents. Le nombre de documents jugés est appelé la *cardinalité* de jugement $|j_k|$.
- En ce qui concerne le score de jugement, j_k est un vecteur de jugement, $j_k(i)$ indique la valeur donnée au *i*ème document de l'ensemble de documents jugés par le juge *k*. Un jugement est dans l'intervalle [0, 1]. Par exemple, pour un ensemble de documents {*a*, *b*, *c*}, $i = 1$ indique le document *a*, un jugement peut être j_k = [0.9, 0.2, 0.7], et $j_k(1) = 0.9$ indique le jugement du document *a* par le juge *k*.
- En ce qui concerne le jugement par ordonnance, j_k contient les documents et les relations entre eux. Par exemple, pour un ensemble de documents {*a*, *b*, *c*}, un jugement par ordonnancement du juge *k* peut être exprimé ainsi : j_k = [*bca*] (ordre total sans égalité) qui indique que *b* est moins pertinent que *c* qui est moins pertinent que *a*.

2.3.1.1. Mesure du Désaccord entre deux Juges

1. le score de jugement : le désaccord *dés* entre deux jugements effectués par les juges 1 et 2, soient j_1 et j_2, est calculé comme la différence moyenne de ces deux jugements :

$$dés(j_1, j_2) = \frac{1}{j_1} \sum_{i=1}^{j_1} |j_1(i) - j_2(i)|$$

2. le jugement par ordonnancement : l'idée de base est qu'un ordonnancement est une permutation d'éléments, alors on a à compter la distance entre deux permutations. On mesure le désaccord comme la distance normalisée par la distance maximale (qui est la distance entre un ordonnancement et son opposé). On compte le nombre de permutations.

   Par exemple, si on a cinq documents *a*, *b*, *c*, *d* et *e* la distance entre j_1 [*abcde*] et j_2 [*abedc*] est 3, parce qu'il faut appliquer trois changements entre les éléments adjacents du premier ordonnancement afin d'obtenir le deuxième :

   $[abcde] \rightarrow [abced] \rightarrow [abecd] \rightarrow [abedc]$

   La distance maximale est la distance entre [*abcde*] et son opposé [*edcba*], elle est égale ici à 10. Donc le désaccord entre j_1 et j_2 est $dés(j_1, j_2) = \frac{3}{10}$

2.3.1.2. Mesure du Désaccord d'un Groupe de Juges

Si nous avons un ensemble de jugements $J = \{j_1, j_2, \ldots, j_n\}$ donnés par n juges. Le désaccord du groupe peut être défini comme la moyenne des désaccords de chaque juge avec les autres juges.

$$DES(J) = \frac{\sum_{i=1}^{n} \frac{\sum_{k \neq i} dés(j_i, j_k)}{n-1}}{n}$$

où la fonction *dés* est le désaccord entre deux jugements défini dans le paragraphe précédent.

Le désaccord maximal pour un groupe de juges est réalisé lorsque aucun des juges n'a de consensus avec les autres c'est-à-dire chaque juge donne des jugements différents. Ce calcul de désaccord proposé ci-dessus dépend du nombre de juges dans le groupe. Cette valeur du désaccord est une valeur dans l'intervalle [0, 1[où on n'obtient jamais la valeur 1.

2.4. Conclusion

Ce chapitre a présenté le vocabulaire et quelques notions de base de la pertinence et son jugement ainsi que l'évaluation de la recherche d'information et l'aspect collaboration de la pertinence. Il justifie notre considération de jugement pour aider les utilisateurs comme on va voir plus loin (CHAPITRE.5). Nous tirons plusieurs points intéressants que nous résumons ainsi :

- Il y a plusieurs sortes de pertinence, la pertinence est de nature dynamique. Pour la juger les utilisateurs ont adopté différents critères, et ce jugement peut être exprimé par valeur ou par ordonnancement. Tout cela nous permet de préciser nos concepts de pertinence et son jugement et de nous situer par rapport à la pertinence dont nous allons parler ultérieurement au CHAPITRE.4.
- Les études insistent sur l'importance de la pertinence pour évaluer les systèmes de recherche d'information où plusieurs méthodes et mesures sont proposées et testées pour évaluer la recherche d'information. Nous pouvons, dans notre contexte réutiliser certaines de ces mesures, afin d'évaluer une étape de recherche, un utilisateur et une session de recherche individuelle ou collaborative et même pour évaluer le système de soutien proposé.
- La nature subjective du jugement de pertinence a amené à mesurer l'accord et le désaccord entre différents juges. Nous pouvons utiliser cette mesure afin de déterminer les utilisateurs dont les profils ou les centres d'intérêts sont proches ou similaires. De tels utilisateurs peuvent apporter une aide intéressante l'un à l'autre.
- La plupart des études se basent sur une évaluation de la pertinence pour estimer essentiellement la qualité des documents retrouvés. Peu d'études, à notre connaissance essaie d'évaluer l'utilisateur et son l'efficacité lors d'une session de recherche. Nous nous sommes intéressés à ce problème (CHAPITRE.4) car nous pensons que c'est un élément intéressant pour orienter l'aide apportée par le système dans un environnement collaboratif.

CHAPITRE.3. Fusion

Nous consacrons ce chapitre aux différents travaux effectués sur la fusion de données en recherche d'informations. L'idée maîtresse qui guide ces travaux est la prise en compte et l'intégration de différents points de vue dans la recherche. La fusion peut concerner, d'une part, différentes expressions du même besoin d'information (paragraphe 1.1) et, d'autre part, les différents résultats obtenus en faisant varier :

1. les requêtes (paragraphe 3.2).
2. les systèmes de recherche, (paragraphe 3.3).
3. les collections, (paragraphe 3.4).

Les études dans ce domaine ont essayé de répondre aux questions suivantes : pourquoi la fusion améliore-t-elle la performance mesurée en termes d'un ou plusieurs critères fixés à l'avance (par exemple précision, rappel, ordre des références) ? comment fusionner (les fonctions/méthodes de combinaison des requêtes ou des résultats) ? Enfin, dans quelles conditions et sous quelles contraintes l'amélioration due à la fusion est-elle atteinte ?

Dans les différentes approches, la fusion est effectuée selon un seul des axes énoncés tandis que les autres ne sont pas pris en compte et ne varient pas au cours de l'expérience. Ainsi, quand la fusion est une fusion de multiples requêtes, ces différentes requêtes sont appliquées au même système de recherche et sur les mêmes collections.

Nous nous intéressons enfin, dans le paragraphe 3.5 aux « méta-moteurs de recherche » qui interrogent, pour chaque requête, différents outils de recherche pour fournir une réponse unique à partir des différents résultats obtenus. On peut, en effet, les considérer comme une application commerciale de la fusion des résultats obtenus à partir de plusieurs outils de recherche. Le paragraphe 3.6 dégagent les résultats intéressants pour notre propre travail qui découlent des travaux présentés dans cet état d'art.

3.1. Fusion de Multiples Expressions d'un Besoin d'Information

L'hypothèse sous-jacente est que le processus de formulation d'un besoin d'information à l'aide d'une requête est difficile. De ce fait, quelles que soient les techniques de recherche utilisées, on ne capture qu'une partie de ce besoin : celle qui répond à la requête émise. En conséquence il semble légitime de penser qu'en utilisant plusieurs expressions du même besoin d'information, on couvrira plusieurs aspects du besoin et donc plus de documents pertinents devraient être trouvés.

C'est cette démarche qui a été adoptée et qui fait l'objet de l'article dont le premier signataire est Belkin [Belkin 1993]. La démarche que les auteurs ont suivie consiste à :

- Obtenir plusieurs requêtes pour un même besoin d'information en donnant à dix chercheurs volontaires dix « descriptions de thèmes ». Les descriptions de thèmes utilisées

sont prises parmi les 50 thèmes donnés pour les expériences de TREC/TIPSTER [Harman 1993], le Tableau 3.1 donne un exemple de thème dans TREC. Les chercheurs devaient utiliser toutes les informations pertinentes extraites des thèmes pour construire des requêtes booléennes. Chaque chercheur a créé une requête pour cinq thèmes différents pris parmi les dix, et a donc fourni cinq requêtes booléennes.

Exemple de Thème dans TREC

<top>
<head> Tipster Topic Description
<num> Number: 066
<dom> Domain: Science and Technology
<title> Topic: Natural Language Processing
<desc> Description: Document will identify a type of natural language processing technology which is being developed or marketed in the U.S.
<narr> Narrative: A relevant document will identify a company or institution developing or marketing a natural language processing technology, identify the technology, identify one or more features of the company's product.
<con> Concept(s):
1. natural language processing
2. translation, language, dictionary, font
3. software applications
<fac> Factor(s):
<nat> Nationality: U.S.
</fac>
<def> Definition(s):
</top>

Tableau 3.1. *Exemple de thème dans TREC donné dans [Harman 1993].*

- Construire cinq groupes de requêtes à partir des requêtes élaborées par les chercheurs. Dans chaque groupe, il y a une requête q_{ti} concernant chacun des dix thèmes t_i :

  groupe de requêtes : (q_{t1}, q_{t2}, …, q_{t10})

  Deux façons différentes ont été utilisées pour les construire :

  1. dans chaque groupe de dix requêtes quatre proviennent d'un même chercheur u_j, on les appelle les groupes g_i où i = 1, 2, …, 5. Par exemple les quatre requêtes q_{t1}, q_{t2}, q_{t6}, q_{t7} proviennent de u_1 :

     groupe de requêtes (q_{t1}, q_{t2}, q_{t3}, q_{t4}, q_{t5}, q_{t6}, q_{t7}, q_{t8}, q_{t9}, q_{t10})

     u_1 u_2 u_3 u_4 u_5 u_6 u_7

  2. chaque groupe contient une requête par chercheur, on les appelle les groupes $g'i$ où i = 1, 2, …, 5. Par exemple :

     groupe de requêtes (q_{t1}, q_{t2}, q_{t3}, q_{t4}, q_{t5}, q_{t6}, q_{t7}, q_{t8}, q_{t9}, q_{t10})

     u_1 u_2 u_3 u_4 u_5 u_6 u_7 u_8 u_9 u_{10}

- Combiner les requêtes en utilisant le moteur de recherche INQUERY (probabilistic inference network-based system) et appliquer les différentes requêtes sur la collection test TREC/TIPSTER. INQUERY permet de combiner de multiples expressions de requête. En effet, ce système possède un grand nombre d'opérateurs permettant de combiner des requêtes en langue naturelle et des requêtes booléennes. La combinaison des groupes de requêtes est fournie dans INQUERY en utilisant un opérateur de somme non-pondéré, (voir [Turtle 1991] pour une description plus détaillée).

  Pour leur expérience, ils ont d'abord appliqué chacun des groupes de requêtes séparément, puis la combinaison progressive de plusieurs groupes ; les groupes 1 et 2, puis 1, 2 et 3 et ainsi de suite.

Les principaux résultats obtenus sont les suivants :

- Il y a une influence importante du chercheur sur la performance des groupes de requêtes. Dans le cas des groupes qui n'ont qu'une requête par chercheur g'_i, on constate que les performances respectives des cinq groupes sont très variées. En revanche, les différences de performance ne sont pas significatives entre les cinq groupes qui contiennent chacun quatre requêtes du même chercheur g_i. Les auteurs ne s'expliquent pas cette dépendance[1].
- En général, pour les deux types de groupes g_i et g'_i la performance augmente avec le nombre de groupes combinés et celle de tous les groupes combinés est meilleure que celle de chaque groupe appliqué isolément[2].

Nous retenons de cette expérience que les utilisateurs influent sur la performance de la recherche, puisque les deux façons de construire des groupes de requêtes ont donné des performances différentes, bien que dans cette expérience on n'ait pas pu fournir d'explication à ce constat. Il n'en reste pas moins que cette expérimentation nous pousse à étudier plus avant les effets, en terme des requêtes effectuées, des utilisateurs sur la performance de recherche.

3.2. Fusion des Résultats de Multiples Requêtes

L'idée de base pour ce type de fusion est de combiner les résultats obtenus avec différentes expressions de requête. Nous présentons ici deux approches : la première combine les résultats de requêtes booléennes et vectorielles (paragraphe 3.2.1), tandis que l'autre combine les résultats des requêtes produites par l'application de différentes méthodes de bouclage de pertinence (paragraphe 3.2.2). Dans le paragraphe 3.2.3 nous discutons et comparaison les approches concernant la fusion de multiples expressions d'un besoin d'information d'une part et la fusion des résultats de multiples requêtes d'autre part.

---

[1] « *We do not yet understand the effect of individual searcher on performance, especially how it might affect the weight given to particular source of evidence.* »

[2] « *Progressive combination of query formulations leads to progressively improving retrieval performance.* »

3.2.1. *Fusion des Résultats des Requêtes Booléennes/P-norm et Vectorielle*

Fox et Shaw dans [Fox 1994] ont proposé et expérimenté différentes fonctions pour combiner les résultats de plusieurs sortes de requête. Ils ont utilisé le système de recherche d'information SMART et les sous-collections de TREC. Les requêtes ont été créées automatiquement à partir des descriptions de thèmes fournies par NIST, où deux types de requêtes sont testés :

1. Un ensemble de requêtes booléennes étendues P-norm (*extended boolean queries*), où chaque requête est constituée des termes liés par des opérateurs AND et OR. Une pondération uniforme a été associée aux termes, c'est-à-dire que tous les termes avaient le même poids, ce poids a pris les valeurs de 1, 1.5 et 2. Donc l'ensemble a été interprété avec des poids différents (P-norm/P-valeurs 1, 1.5 et 2).
2. Deux ensembles de requêtes vectorielles ont été construits automatiquement à partir de descriptions de thèmes de TREC (*TREC topic description*). Le premier ensemble ne prend pas en compte la partie narrative des descriptions de sujet de recherche (il prend en compte les sections « Title », « Description », « Concepts »,… voir le Tableau 3.1) qui est appelé ensemble de vecteurs courts de requêtes (SV) (*Short Vector*), le second ensemble la prend en compte et est appelé ensemble de vecteurs longs de requêtes (LV) (*Long Vector*).

3.2.1.1. Fonctions de Combinaison

Fox et Shaw ont testé six fonctions de fusion résumées dans le Tableau 3.2. Ces fonctions sont les suivantes :

- CombMAX : considère la valeur maximale des valeurs de pertinence individuelles, afin de minimiser la probabilité qu'un document pertinent soit classé en queue de liste de résultats.
- CombMIN : contrairement à la fonction précédente celle-ci prend la valeur minimale des valeurs de pertinence individuelles, pour minimiser la probabilité qu'un document non pertinent soit classé en tête de liste.
- CombMED : cette fonction est une approche simple qui prend en compte les deux raisonnements des fonctions précédentes CombMAX et CombMIN puisqu'elle calcule la valeur moyenne des valeurs de pertinence individuelles.
- CombSUM : prend la somme des valeurs de pertinence individuelles, ce qui favorise les documents apparus dans plusieurs listes de résultats.
- CombANZ : ignore les résultats qui ne contiennent pas le document, donc elle calcule la valeur moyenne des valeurs de pertinence non-nulles.
- CombMNZ : favorise les documents retrouvés dans plusieurs listes de résultats en leur donnant des poids élevés ainsi : CombSUM * Nb des valeurs non- nulles.

Les deux premières fonctions appelées CombMAX et CombMIN sélectionnent une seule valeur de pertinence parmi plusieurs tandis que les quatre suivantes CombMED, CombSUM, CombANZ, et CombMNZ considèrent toutes les valeurs de pertinence au lieu d'une seule.

Fonctions	Combinaison de Valeurs de Pertinence
CombMAX	MAX(pertinences individuelles)
CombMIN	MIN(pertinences individuelles
CombMED	MED(pertinences individuelles)
CombSUM	SUM (pertinences individuelles)
CombANZ	CombSUM / Nb des valeurs non-nulles
CombMNZ	CombSUM * Nb des valeurs non-nulles

Tableau 3.2. *Résumé des fonctions de combinaison.*

3.2.1.2. Expérimentation et Résultat

Dans cette expérimentation, Fox et Shaw ont appliqué séparément les cinq groupes de requêtes (*P-norm*$_1$, *P-norm*$_{1.5}$, *P-norm*$_2$, *LV* et *SV*), puis ils ont testé les fonctions de fusion mentionnées ci-dessus sur les résultats retournés où les valeurs de pertinence n'ont pas été normalisées. Les performances des différentes combinaisons ont été comparées avec la performance d'un seul groupe de requêtes et les résultats obtenus sont résumés dans les points suivants :

- La combinaison CombMAX et CombMIN donne quelquefois de meilleures performances que celles d'un seul groupe de requêtes. Mais, cela n'est pas vérifié dans tous les cas.
- Les combinaisons CombANZ et CombMNZ ont toutes les deux amélioré les performances, et la combinaison CombMNZ fournit parfois une performance légèrement meilleure que la combinaison CombANZ.
- La combinaison CombSUM réalise une performance significativement meilleure que le meilleur groupe de requêtes isolé, et de plus elle fournit une efficacité de recherche meilleure que CombMAX, CombMIN et CombANZ.
- CombMNZ est mieux que CombSUM parce qu'elle favorise les documents retrouvés par plusieurs groupes de requêtes.

3.2.2. *Fusion des Résultats des Requêtes issues des Bouclages de Pertinence*

Dans cette approche, on génère automatiquement de multiples expressions de la même requête de recherche d'information, en appliquant différentes méthodes de bouclage de pertinence. Puis on combine les résultats obtenus en appliquant ces différentes expressions.

Le processus de bouclage de pertinence est un processus automatique de reformulation de requêtes. Il consiste à choisir les termes importants dans les documents pertinents et à améliorer leur poids dans une nouvelle formulation de requête. De façon analogue, les termes inclus dans des documents non pertinents peuvent être éliminé dans la nouvelle formulation de requête. L'effet d'un tel processus est de « modifier » la requête dans la direction des documents pertinents et de l'éloigner de ceux qui ne sont pas pertinents.

Lee [Lee 1998] a étudié ce type de fusion en utilisant le système de recherche SMART et cinq différentes méthodes de bouclage de pertinence implantées par ce système. La collection de test était TREC D1&D2. 50 requêtes sur des thèmes de TREC 151-200 ont été utilisées. Le

processus général de la combinaison est illustré dans la Figure 1 de l'annexe A. La démarche adoptée par Lee a consisté à :

1. Construire un vecteur de requête initial pour un besoin donné d'information.
2. Exécuter la recherche initiale, et considérer que les 30 premiers documents retrouvés sont des documents pertinents.
3. Engendrer de façon complètement automatique de nouveaux vecteurs de requête par l'expansion du vecteur de requête initial avec différentes méthodes de bouclage de pertinence.
4. Normaliser les nouveaux vecteurs de requête avec la normalisation cosinus où chaque poids de terme est divisé par la norme euclidienne de la longueur de vecteur.
5. Exécuter la recherche avec les nouveaux vecteurs de requête.
6. Normaliser chaque valeur de pertinence *perSys* donnée par le système de recherche à un document pour une requête *q* dans les résultats de recherche :

$$perSysNorm = \frac{perSys - perSysMin}{perSysMax - perSysMin} \qquad [3.1]$$

où *perSysMin, perSysMax* sont respectivement les valeurs de pertinence maximale et minimale données par le système à un document dans le résultat retrouvé pour *q*. Normalement $perSysMin \leq perSys \leq perSysMax$, puisque les valeurs *perSysMin, perSysMax* varient dans chaque résultat alors la normalisation fait que $0 \leq perSysNorm \leq 1$.

7. Combiner les résultats rendus. Les résultats des travaux de [Fox 1994] et [Belkin 1993], cités plus haut, ont amené l'auteur à utiliser la fonction de somme pour combiner les résultats rendus par les différentes méthodes de bouclage de pertinences (*feedback*) :

CombSUM = $perSys_1 + perSys_2 + \ldots$

L'auteur [Lee 1998] a expérimenté la combinaison des résultats de 2, 3, 4, et des 5 méthodes de bouclage de pertinence. Il constate que la performance augmente de façon monotone chaque fois qu'une autre méthode de bouclage de pertinence est prise en compte. Ainsi, cette étude montre que la combinaison des résultats des requêtes issues de différentes méthodes de bouclage de pertinence peut conduire à une amélioration substantielle de l'efficacité de recherche.

3.2.3. *Discussion*

Nous comparons les trois approches précédentes [Belkin 1993], [Fox 1994] et [Lee 1998] que nous venons de décrire ainsi :

- L'approche de [Belkin 1993] utilise le système de recherche INQUERY, les deux autres approches utilisent le système de recherche SMART, mais toutes les trois utilisent la précision moyenne pour mesurer la performance de recherche et la combinaison
- l'approche [Belkin 1993] fusionne plusieurs requêtes d'un besoin d'information, et la valeur de pertinence d'un document *d* est calculée ainsi : $perSys(d) = \cos(\Sigma_i\, q_i, d)$

Les autres approches fusionnent les résultats de plusieurs requêtes :

sans normalisation pour [Fox 1994] : $perSys(d) = \Sigma_i \cos(q_i, d)$

ou avec normalisation pour [Lee 1998] : $perSys(d) = \Sigma_i$ [cos (q_i, d) normalisé]

3.3. Fusion des Résultats de Multiples Systèmes de Recherche

Il s'agit ici de tenir compte dans l'évaluation de la pertinence des documents retrouvés de plusieurs systèmes de recherche. Nous nous intéressons (paragraphe 3.3.1) d'abord à l'analyse de différentes fonctions permettant de combiner les valeurs de pertinence-système obtenus. Puis, nous analysons l'influence de l'indépendance des systèmes de recherche sur leur combinaison (paragraphe 3.3.2). Ensuite, nous présentons une technique consistant à combiner linéairement des valeurs de pertinence obtenues pour des documents dans chacun des différents systèmes, ainsi que quelques expériences effectuées en utilisant cette approche (paragraphe 3.3.3). Enfin, une discussion à propos de ces différentes approches est proposée dans le paragraphe 3.3.4.

La fusion des résultats obtenus en opérant sur différents systèmes de recherche permet d'obtenir des effets différents entre ces systèmes selon la façon dont elle est faite [Diamond 1996]. Ces différents effets peuvent être :

- *Ecrémage* (*Skimming Effect*) : se passe quand les approches de recherche amènent à des documents différents et donc retrouvent des références pertinentes différentes. Un modèle de combinaison possible prend les références les plus pertinentes de chaque approche de recherche (tête de liste, c'est-à-dire « la crème » des listes de résultat) cet écrémage peut, non seulement augmenter le rappel (dû aux différents documents pertinents) mais aussi la précision (en supposant que les systèmes ont une grande densité de documents pertinents en tête de leur liste de résultats).
- *Choeur* (Chorus Effect) : se passe quand différentes approches de recherche suggèrent qu'une même référence est pertinente pour une requête. Cette référence tend à avoir une plus forte importance pour la pertinence qu'une autre qui ne serait délivrée que par une seule approche. En exploitant cet effet, on obtient une combinaison où les documents sont rangés selon leur fréquence d'apparition dans les listes des documents retrouvés par les différents systèmes.
- *Cheval noir* (Dark Horse Effect) : se passe parce que l'on estime qu'une des approches de recherche produit une évaluation de pertinence plus intéressante que les autres.

3.3.1. *Analyse de Plusieurs Combinaisons de Multiples Techniques*

Lee [Lee 1997] a évalué une variété de fonctions de combinaison et il a envisagé la combinaison non seulement des valeurs de pertinence des documents dans le résultat mais aussi leurs valeurs d'ordonnancement. Il a trouvé un nouveau raisonnement pour la combinaison que différents systèmes retrouvent des ensembles semblables de documents pertinents, mais retrouvent des ensembles différents de documents non-pertinents[3].

[3] New rationale for evidence combination: « *Different runs might retrieve similar sets of relevant documents but retrieve different sets of nonrelevant documents.* »

Pour ses expériences, il a choisi six résultats de recherche fournis par TREC3. Les résultats sont combinés selon deux axes : l'utilisation de la valeur de pertinence et l'utilisation de la valeur d'ordonnancement.

1. **Utilisation de la valeur de pertinence :** puisque des systèmes de recherche différents distribuent aux documents retrouvés des valeurs de pertinence *perSys* différentes, une normalisation s'impose afin de pouvoir comparer les valeurs. Les valeurs sont normalisées selon la formule [3.1] (page 54). Pour la combinaison, il a commencé par appliquer les fonctions employées par Fox [Fox 1994] à savoir CombMAX, CombMIN, CombMED, CombSUM, CombANZ, et CombMNZ (voir le paragraphe 3.2.1.1). Ensuite, il a généralisé les deux fonctions CombMNZ et CombSUM par la fonction CombGMNZ suivante afin de favoriser davantage les documents retrouvés dans plusieurs systèmes ainsi :

   $$\text{CombGMNZ} = \text{CombSUM} \times (\text{Nb des valeurs de pertinence non-zéro})^{\gamma} \qquad \gamma \geq 1$$

   $$\text{CombGMNZ} = \begin{cases} \text{CombSUM} & \text{si } \gamma = 0 \\ \text{CombMNZ} & \text{si } \gamma = 1 \end{cases}$$

   Le paramètre γ indique le fait que l'on donne des poids plus hauts aux documents qui sont retrouvés par plusieurs systèmes de recherche.

   En appliquant toutes les fonctions de combinaisons mentionnées ci-dessus et la fonction CombGMNZ avec une variété des valeurs pour le paramètre γ, il a obtenu les résultats suivants :

   - Les fonctions de combinaison CombSUM et CombMNZ fournissent une meilleure performance que CombMAX, CombMIN, CombANZ, ce qui confirme le résultat donné par Fox [Fox 1994] (voir le paragraphe 3.2.1).
   - CombMNZ (CombGMNZ où $\gamma = 1$) travaille toujours légèrement mieux que CombSUM (CombGMNZ où $\gamma = 0$) indépendamment du nombre de documents retrouvés. Et, même en donnant la valeur dix au paramètre γ (pour donner à tous les documents communs classés en tête de liste plus d'importance que celle attribuée à n'importe quel document qui n'est pas commun), l'efficacité de la fonction CombGMNZ est toujours légèrement meilleure que celle de CombSUM.

   Donc, le fait de donner plus de poids aux documents communs améliore la performance, ce qui confirme son raisonnement de départ, c'est-à-dire que la combinaison est mieux pour les systèmes qui retrouvent des ensembles semblables de documents pertinents, mais qui retrouvent des ensembles différents de documents non-pertinents.

2. **Utilisation de la valeur d'ordonnancement :** puisque l'utilisation des valeurs de pertinence ne considère pas la performance générale des systèmes isolés, Lee suggère de fusionner les valeurs d'ordonnancement des documents *rangSys*. Pour cela, il applique d'abord la fonction *SimRange* sur la valeur d'ordonnancement des documents *rangSys.* Avec :

   $$SimRange(rangSys) = 1 - \frac{rangSys - 1}{NbDocRetrouvé}$$

Puis, il fusionne les valeurs résultantes *SimRange*(*rangSys*) des différents systèmes mentionnés ci-dessus en utilisant la fonction CombMNZ. Il constate alors que, si les systèmes engendrent des ordonnancements complètement différents, la combinaison des valeurs d'ordonnancement donne une efficacité de recherche meilleure que celle obtenue en utilisant les valeurs de pertinence. Sinon, c'est la combinaison des valeurs de pertinence qui fournit une efficacité de recherche légèrement meilleure.

Donc, l'utilisation de la valeur d'ordonnancement est un peu mieux que l'utilisation de la valeur de pertinence si les systèmes ordonnent les résultats de façon complètement différente.

3.3.2. *Effet de l'Indépendance des Systèmes sur la Fusion des Résultats*

Smeaton [Smeaton 1998] choisit de fusionner les résultats de systèmes de recherche d'information différents et indépendants les uns des autres. Afin d'expérimenter, il utilise neuf systèmes différents utilisés par différents groupes dans TREC-4.

Le regroupement des neuf systèmes selon leurs dépendances ou indépendances conceptuelles est subjectif, l'auteur s'est basé sur quelques indicateurs pour obtenir les quatre groupes suivants : le groupe des approches qui emploient des fonctions de pondération des termes similaires à travers le modèle vectoriel, le groupe des systèmes qui font l'étiquetage des documents et des requêtes et les représentent par les formes/classes des mots, le groupe des approches qui utilisent le modèle n-grammes (qui considère toutes les chaînes de longueur fixe de *n* caractères qui se trouvent dans le document et la requête), et le dernier groupe correspond aux approches qui emploient l'algorithme d'inférence dans les réseaux Bayésiens pour la recherche.

La technique de fusion qu'il utilise est de faire la somme des valeurs normalisées de pertinence de documents obtenus par l'utilisation de plusieurs systèmes de recherche d'information. Ainsi, les valeurs de pertinence assignées aux documents sont normalisées dans l'intervalle [0, 1], avec la valeur 1 assignée au document qui a la plus grande valeur de pertinence. Pour évaluer la performance d'une telle fusion, il utilise la valeur de la précision respectivement pour les 5, 10 et 20 premiers documents de chacune des listes de résultats comme une base de comparaison de la performance.

Les neuf systèmes de recherche sont combinés deux à deux (36 résultats), une amélioration a été observée dans dix combinaisons deux à deux, parmi ces dix combinaisons, sept combinaisons sont entre des systèmes qui appartiennent à des groupes différents.

Les résultats obtenus à l'issue de cette expérience montrent que la fusion entre des systèmes de recherche indépendants conceptuellement améliore la performance de recherche.

3.3.3. *Fusion Linaire des Résultats*

Nous présentons tout d'abord la technique de combinaison linéaire, puis nous présentons quelques expériences effectuées selon cette approche.

Dans le modèle de combinaison linaire (LC Linear Combination), l'estimation globale CombLIN(*d*) de la pertinence d'un document *d* pour une requête *q* est la somme pondérée

(par les paramètres Θ_i) des valeurs de pertinence obtenues $perSys_i(d)$ dans chaque système individuel sys_i.

$$\text{CombLIN}(d) = \Theta_1 perSys_1(d) + \Theta_2 perSys_2(d) + \ldots + \Theta_i perSys_i(d) + \ldots \quad [3.2]$$

Les difficultés de la combinaison linéaire résident dans la détermination :

- des types de systèmes avec lesquels le modèle marche le mieux,
- des valeurs à donner aux poids Θ_i afin de maximiser la performance.

Résoudre ces difficultés a fait l'objet de plusieurs travaux, que nous présentons dans ce qui suit, où les travaux de Bartell [Bartell 1994] et Vogt [Vogt 1997] déterminent les valeurs des paramètres Θ_i par l'apprentissage afin d'optimiser l'ordre des documents dans le résultat global (paragraphe 3.3.3.1). Le travail [Vogt 1999] propose un autre schéma de pondération Θ_i ainsi qu'une nouvelle mesure de performance *diff* afin de déterminer les valeurs des paramètres Θ_i (paragraphe 3.3.3.2). *diff* est la différence entre la valeur de pertinence moyenne des documents pertinents et celle des documents non pertinents.

3.3.3.1. Optimisation selon la Mesure de l'Ordonnance des Documents

Bartell et son groupe [Bartell 1994] ont utilisé des requêtes témoins (test) afin d'apprendre quelles sont les valeurs des paramètres Θ_i qui optimisent le mieux l'ordre des documents dans le résultat global. Il s'agit donc de comparer l'ordre produit par la fonction CombLIN à l'ordre fournit par l'utilisateur, pour mesurer la corrélation entre ces deux ordres, le critère J a été utilisé :

$$J(\text{CombLIN}) = \frac{1}{|Q|} \sum_{q \in Q} \frac{\sum_{d \succ_q d'} (\text{CombLIN}(q,d) - \text{CombLIN}(q,d'))}{\sum_{d \succ_q d'} |\text{CombLIN}(q,d) - \text{CombLIN}(q,d')|}$$

Où :

$|Q|$ le nombre des requêtes dans l'ensemble de requêtes Q.

$d \succ_q d'$ indique que l'utilisateur préfère le document d au document d' pour la requête q.

Le critère J a une valeur maximale 1 quand le numérateur et le dénominateur sont les mêmes, cela signifie que les deux ordres sont identiques, cela signifie aussi que CombLIN classe les documents exactement comme l'utilisateur le voudrait, il s'agit donc du cas idéal. La valeur minimum -1 est justement la situation inverse.

Donc, le critère J est une sorte de correspondance entre l'ordre fourni par le système et l'ordre que l'utilisateur effectue sur les documents.

Dans la suite nous présentons les différentes expérimentations (paragraphe 3.3.3.1.1) ainsi que leurs résultats (paragraphe 3.3.3.1.2) effectuées pour optimiser le critère J.

3.3.3.1.1. Expérimentations

Nous présentons différentes expérimentations qui ont comme but d'utiliser l'apprentissage des pondérations Θ_i afin d'optimiser J c'est-à-dire améliorer l'ordre des documents. Les deux premières expérimentations effectuées par Bartell [Bartell 1994] utilisent comme collection de test des portions de l'*Encylopaedia Britannica* (EB) tandis que la troisième est effectuée par Vogt [Vogt 1997] en utilisant la collection TREC (qui est plus grande que celle utilisée par Bartell). Les trois expérimentations diffèrent en termes des systèmes de recherche pris en compte pour la combinaison linéaire. Détaillons maintenant ces trois expérimentations :

1. La première consiste à combiner deux systèmes vectoriels, l'un est un système classique, appelé « terme », qui utilise la délimitation de termes ; l'autre, appelé « phrase », retrouve les documents qui contiennent les groupes nominaux qui apparaissent dans la requête. Le système « phrase » ne pourra jamais retrouver de document qui n'ait pas déjà été trouvé par le système « terme ». Ainsi l'utilité du système « phrase » dans la combinaison est d'améliorer l'ordonnancement des documents dans les résultats du système « terme ». La formule [3.2] (page 58) pour cette expérimentation devient :

   $$\text{CombLIN}(d) = \Theta_{phrase}\, perSys_{phrase}(d) + \Theta_{terme}\, perSys_{terme}(d)$$

2. La seconde consiste à combiner trois outils de recherche du système commercial *SmarTrieve* (développé par Compton's New Media, Inc), les trois outils sont des modèles vectoriels appelés l'outil vectoriel traditionnel (*vector expert*), l'outil compteur du nombre de termes de la requête dans le document (*count expert*), et un outil d'estimation du nombre de groupes nominaux de la requête apparaissant dans les documents (*phrase expert*). La formule [3.2] pour cette expérimentation devient alors :

   $$\text{CombLIN}(d) = \Theta_{phrase}\, perSys_{phrase}(d) + \Theta_{count}\, perSys_{count}(d) + \Theta_{vecteur}\, perSys_{vecteur}(d)$$

3. La troisième utilise d'autres systèmes de recherche où les auteurs dans [Vogt 1997] ont en effet combiné les trois systèmes suivants :

   - Les deux premiers sont basés sur le modèle vectoriel standard de SMART et utilisent deux schémas de pondération différents appelés respectivement *bnn* et *ltc* selon le code de SMART. *bnn* est avec une pondération binaire des termes ($w_{ij} = 1$) et *ltc* avec une pondération logarithmique des termes basée sur « tf-idf » [4].

[4] Le poids w_{ij} pour le terme t_j ($j = 1, 2, ..., m$) dans un document d_i est :

$$w_{ij} = \frac{(\log(tf_{ij})+1)*idf_j}{\sqrt{\sum_{k=1}^{m}((\log(tf_{ik})+1)*idf_k)^2}}$$

tf_{ij} : indique la fréquence d'occurrence du terme t_j dans le document d_i (ou dans la requête), df_j : le nombre de documents dans lesquels le terme t_j apparaît (fréquence documentaire), n représente le nombre de documents d_i dans la collection, et idf_j l'inverse de la fréquence documentaire ($idf_j = \log [n/df_j]$).

- Le troisième est basé sur l'analyse factorielle des correspondances (Latent Semantic Indexing) et sera noté *lsi*[5].

La formule [3.2] (page 58) des trois systèmes est alors :

$$\text{CombLIN}(d) = \Theta_{bnn}\, perSys_{bnn}(d) + \Theta_{ltc}\, perSys_{Itc}(d) + \Theta_{lsi}\, perSys_{lsi}(d)$$

3.3.3.1.2. Résultats d'Expérimentations

La démarche pour l'évaluation est ainsi : tout d'abord, la performance de chaque système isolée est évaluée. Puis, les poids optimaux Θ_i pour chaque outil a été appris en utilisant un ensemble de requêtes. Les résultats de ces expérimentations sont :

- Pour la première expérimentation, la combinaison optimale est obtenue en affectant au résultat du système « phrase » un poids légèrement plus fort que celui affecté au résultat du système « terme » (Θ_{phrase} = 0.738, Θ_{terme} = 0.675). La combinaison des résultats des systèmes « terme » et « phrase » a augmenté la précision de 12% par rapport à celle fournit par le meilleur des deux systèmes qui est le système « terme ».
- Pour la deuxième expérimentation l'outil « count » a eu un poids significativement plus grand que les deux autres. Le poids affecté à l'outil « phrase » vient ensuite et est plus grand que celui qui est affecté à l'outil « vecteur », (Θ_{phrase} = 0.305, Θ_{count} = 0.946, $\Theta_{vecteur}$ = 0.112). La combinaison des résultats des trois outils réalise une performance meilleure (de 47%) que celle du meilleur des outils (l'outil « count ») considéré isolément.
- Les auteurs dans [Bartell 1994] ont donc obtenu une amélioration de performance très encourageante à travers ces deux expérimentations. Cependant, ils pensent que cette possible amélioration dépend de la spécification de différents facteurs. Par exemple, les outils utilisés, les caractéristiques de la collection, en plus le nombre de documents inclus dans l'ensemble d'apprentissage, et la qualité des requêtes utilisées pour l'apprentissage, notamment à quel point ces requêtes représentent les requêtes typiques du système.
- Pour la troisième expérimentation, les auteurs dans [Vogt 1997] ont essayé d'utiliser d'abord les 15 premiers documents en tête de liste pour chaque requête afin optimiser *J*, comme c'était le cas dans les deux expérimentations de [Bartell 1994], puis ils ont pris les 100 premiers documents pour obtenir une combinaison plus raisonnable étant donné que la collection utilisée WSJ de TREC est beaucoup plus grande. Ils ont trouvé que la combinaison réalise une performance significativement meilleure que celle de n'importe lequel des systèmes pris isolément mais, dans certains cas seulement (pour les données d'expérimentations actuelles 100 documents en tête de liste). Ainsi, l'amélioration dépend de la performance des systèmes pris isolément et de la similarité des classements des différents documents dans le résultat de chacun d'eux.

[5] L'unité d'indexation est les contextes. *lsi* nécessite une étude de tout le texte pour en extraire des relations utiles entre les termes et les documents. Des techniques statistiques sont utilisées pour calculer et simuler ces associations. Le principe de la méthode consiste à construire une matrice (termes-documents), ensuite est réduite en lui appliquant la méthode de décomposition SVD (approximation par combinaisons linéaires).

3.3.3.2. Optimisation selon une Mesure de Performance *diff*

Dans [Vogt 1999] est présentée une analyse approfondie à la fois empirique et mathématique des capacités du modèle qui utilise une combinaison linéaire pour fusionner les systèmes de recherche d'informations. Les auteurs utilisent un très large ensemble de systèmes de recherche (61 TREC-5 *entries*), ils combinent toutes les paires possibles de systèmes, et ils analysent les résultats de ces combinaisons linéaires par comparaison avec le résultat obtenu à l'aide d'un seul système. Ils proposent un nouveau schéma de pondération pour la combinaison de deux systèmes. La formule [3.2] (page 58) devient :

$$\text{CombLIN}(d) = \sin(\Theta)\, perSys_1(d) + \cos(\Theta)\, perSys_2(d)$$

Où : $\Theta_1 = \sin(\Theta)$ et $\Theta_2 = \cos(\Theta)$

Cette formule tient compte de tous les rapports possibles des signes des deux poids (+ / +, +/-,-/+,-/-) et de tous les taux possibles Θ_1/Θ_2.

Ils proposent une mesure de performance *diff* qui permet le choix des poids Θ où *diff* est la différence entre la valeur de pertinence moyenne donnée par le système Sys_i aux documents pertinents $perSys_i$ et la valeur de pertinence moyenne $\overline{non\text{-}perSys_i}$ donnée par le système aux documents non pertinents.

Au terme de leur étude, ils arrivent à la conclusion que la combinaison linaire ne conduit pas toujours à une amélioration de la performance. Elle ne doit être employée que lorsque les systèmes impliqués ont en commun dans leurs résultats individuels un grand nombre de documents pertinents et un petit nombre de documents non pertinents. Cette conclusion les amène à conclure que le modèle linaire CombLIN n'exploite principalement que l'effet de *chœur*.

3.3.4. *Discussion*

Nous avons présenté différentes approches pour fusionner les résultats de multiples systèmes de recherche, nous les résumons et comparons ainsi :

- Lee [Lee 1997] a envisagé deux sortes de combinaison : la combinaison des valeurs de pertinence des documents et la combinaison des valeurs d'ordonnancement des documents. Smeaton [Smeaton 1998] a étudié l'effet de l'indépendance conceptuelle des systèmes sur la fusion. Bartell [Bartell 1994] et Vogt [Vogt 1997] [Vogt 1999] ont étudié la combinaison linéaire des valeurs de pertinence des documents issus de plusieurs systèmes.
- Lee [Lee 1997] a testé plusieurs fonctions de combinaison (CombMAX, CombMIN, CombMED, CombSUM, CombANZ, CombMNZ et CombLIN). Smeaton [Smeaton 1998] a utilisé la combinaison CombSUM. Les autres ont appliqué la combinaison linéaire CombLIN.
- La fusion des résultats de plusieurs systèmes a souvent été bénéfique pour la performance de recherche, et les différentes approches et expérimentations essayent de prévoir une telle amélioration.

- La combinaison CombSUM a donné de bons résultats et la combinaison CombLIN semble plus souple à utiliser que les autres. CombSUM peut être vue comme un cas particulier de CombLIN dans laquelle les poids de la combinaison sont égaux. Nous pouvons, dans notre contexte, réutiliser la fonction linéaire pour fusionner les résultats et les requêtes issus des utilisateurs.
- Nous pensons que les fonctions CombMAX, CombMIN exploitent l'effet de *cheval noir* (voir le paragraphe 3.3) parce qu'elles prennent en compte l'estimation de pertinence d'un seul système, tandis que les fonctions CombMED, CombSUM, CombANZ, CombMNZ, CombGMNZ exploitent plutôt l'effet de *chœur* qui favorise les documents retrouvés par plusieurs systèmes.

3.4. Fusion des Résultats issus de Plusieurs Collections

En général les systèmes de recherche d'information supposent que les documents sont dans une seule et monolithique collection, donc les documents retrouvés par une requête donnée sont ceux de la collection qui ont les meilleures valeurs de pertinence. Cependant, les documents intéressants peuvent être stockés dans plusieurs corpus. Par exemple, pour obtenir les meilleurs résultats d'une requête portant sur « les gens les mieux payés » on doit l'appliquer aussi bien sur des collections de « documents sportif » que sur des collections de « documents d'affaire ». Ainsi, des documents pertinents pour une recherche d'information peuvent être dans plusieurs collections. Il peut donc être intéressant, dans certains cas, d'interroger plusieurs corpus ; pour cela il faut déterminer le nombre de documents à récupérer dans chaque collection.

Nous rappelons brièvement (paragraphe 3.4.1) comment s'énonce le problème de la fusion sur des collections tel qu'il est formalisé dans les travaux de Towell [Towell 1995] et dans ceux de Voorhees [Voorhees 1994]. Puis, nous présentons les principales techniques de combinaison des résultats issus de multiples collections de recherche (paragraphe 3.4.2).

3.4.1. *Problématique de la Fusion sur des Collections*

La problématique de la fusion sur des collections est de déterminer le nombre de documents à récupérer dans chaque collection pour obtenir un meilleur résultat, c'est-à-dire déterminer l'impact et le critère de choix des collections, cette problématique est détaillée dans la suite.

Considérons plusieurs collections de documents 1, 2, …c. Quand une requête donnée q est appliquée, chaque collection i retourne une liste s de documents ordonnés selon leur similarité décroissante avec cette requête. Si on dénote par $F^i_q(s)$ la distribution des documents pertinents à travers l'ensemble s des documents retrouvés. La problématique de fusion sur des collections peut être maintenant formalisée de la façon suivante : trouver les valeurs λ_i (où λ_1, λ_2, …, λ_c) c'est-à-dire trouver le nombre de documents pertinents λ_i qu'il faut prendre de la liste s des documents retrouvés par chaque collection i, de façon que :

- $\sum_{i=1}^{c} \lambda_i = N$: le nombre total de documents pris de tous les listes s des collections est N.

- $\sum_{i=1}^{c} |F^i_q(\lambda_i)|$ est maximum, c'est-à-dire maximiser la distribution des documents pertinents à travers les N documents pris de toutes les collections.

Pratiquement, la fonction F^i_q n'est pas connue, il faut donc en trouver une approximation.

3.4.2. *Fusion sur des Collections*

Nous présentons, d'abord, deux stratégies simples pour trouver une approximation de la fonction F^i_q (paragraphe 3.4.2.1). Puis, nous présentons deux stratégies d'apprentissage pour la fusion sur des collections proposées et expérimentées par Towell [Towell 1995] et Voorhees [Voorhees 1994] ; les deux stratégies déterminent le nombre optimum des documents à retrouver dans chaque collection (paragraphe 3.4.2.2).

3.4.2.1. Deux Stratégies Naïves de Détermination de la Distribution

On peut distinguer les deux stratégies suivantes :

- La stratégie *uniforme* qui est basée sur l'hypothèse que toutes les collections ont le même nombre de documents pertinents et qu'elles ont des distributions identiques de documents pertinents pour chaque requête possible.

  A partir de cette hypothèse, la récupération d'un nombre égal de documents issu de chaque collection maximise en moyenne le nombre total de documents pertinents :

  $$\lambda_1 = \lambda_2 = \ldots = \lambda_c \qquad \Rightarrow \qquad \sum_{i=1}^{c} |F^i_q(\lambda_i)| \text{ est maximal}$$

  Dans la pratique, cela est une approche médiocre parce que les différentes collections ont des spécialités différentes et donc, n'ont pas le même nombre de documents pertinents pour une même requête. Un test de l'efficacité de cette stratégie montre qu'elle diminue la performance de 40% relativement à une recherche qui opèrerait sur une seule collection adaptée.

- La stratégie de *fusion d'ordonnancement* (merge-sort) qui suppose que les valeurs de pertinence données à un document pour une requête au travers des différentes collections sont comparables. Donc, elle fixe des niveaux (seuil-limite) des valeurs de pertinence, tels que les N documents avec les valeurs de pertinence les plus grandes à travers toutes les collections soient sélectionnés.

  Cette approche est valable si les mesures de pertinence sont vraiment comparables. Cependant, ces mesures dépendent de la fréquence des termes dans un document, donc chaque collection a des distribution des mots différentes selon leurs fréquences dans les documents de la collection, ce qui invalide cette hypothèse.

Dans la suite, nous présentons les stratégies d'apprentissage, qui tentent de remédier aux inconvénients de ces deux stratégies.

3.4.2.2. Stratégies d'Apprentissage

Le principe de ces stratégies consiste à employer au départ des requêtes pour l'apprentissage, puis d'utiliser ce qu'on a appris via ces requêtes pour la fusion appliquée à chaque nouvelle requête *q*. Ces stratégies ont utilisé le système SMART (le modèle vectoriel) pour la recherche sur différentes sous-collections de la collection TREC. Nous présentons une illustration de ces stratégies dans la Figure 2 et la Figure 3 de l'annexe A.

- La stratégie de *modélisation de distribution des documents pertinents* (MRDD Modeling Relevant Document Distributions) : qui prédit le nombre de documents pertinents λ_i à récupérer de chaque collection pour une nouvelle requête *q* en utilisant la stratégie suivante. On utilise les *k* requêtes[6] les plus semblables à *q* (la similarité de deux requêtes est déterminée comme le cosinus de l'angle formé par leur vecteur), afin de déterminer la distribution moyenne F^i_q des documents pertinents pour chaque collection. Puis, on calcule les valeurs λ_i pour chaque collection qui maximisent le nombre de documents pertinents retrouvés. Après avoir déterminé les valeurs λ_i, les λ_i documents de chaque collection sont unies de façon aléatoire. Mais, les collections ayant la plus grande valeur de λ_i sont favorisées.
- La stratégie de *groupement de requêtes* (QC Query Clustering). Cette approche ne modélise pas explicitement la distribution des documents pertinents d'une collection. Elle essaie de mesurer la qualité de recherche pour un domaine de thème particulier à la collection. Les domaines de thème sont représentés comme des *centroïdes* des groupes de requêtes. Cette stratégie commence par une phase d'apprentissage qui consiste à :

  1. Grouper pour chaque collection des ensembles des requêtes similaires. Le nombre de documents communs retrouvés par deux requêtes est utilisé ici comme mesure de similarité.
  2. Déterminer le centroïde de chaque groupe de requêtes. Le centroïde est le vecteur moyen des vecteurs associés aux requêtes contenues dans le groupe.
  3. Assigner à chaque groupe un poids qui reflète l'efficacité des requêtes dans le groupe. Ce poids est le nombre moyen de documents pertinents retrouvés parmi les *L* premiers documents retournés par chacune des requêtes du groupe.

  Pour une nouvelle requête *q*, on choisit dans chaque collection le centroïde le plus similaire au vecteur de la requête *q* et on retourne aussi le poids qui lui est associé. L'ensemble des poids w_i rendus par toutes les collections est employé pour répartir l'ensemble total de documents *N* pris de toutes les collections ainsi : $\lambda_i = \frac{w_i}{\sum_{i=1}^{C} w_i} * N$

Les résultats importants de l'évaluation de la précision des résultats des deux méthodes d'apprentissage décrites précédemment portent sur les deux points :

- Chacune des stratégies d'apprentissage est toujours significativement meilleure que la « stratégie uniforme ».

[6] Les expériences ont trouvé que la valeur $k = 8$ est aussi bien ou mieux que d'autres valeurs.

- Les deux stratégies d'apprentissage MRDD et QC sont bonnes pour la fusion sur des collections d'ailleurs leur précision est près de celle d'une « seule collection » (comme si l'on pouvait fusionner toutes les sous-collections en une seule, et monolithique collection et calculer la précision de cette seule collection). Elles diffèrent dans leurs conditions de mémoire, vitesse, et leur capacité de résoudre le problème. La stratégie MRDD est meilleur que la stratégie QC, mais elle demande plus de mémoire et elle est moins rapide.

3.5. Méta-Moteur

Les méta-moteurs sont des techniques de présentation des résultats issus de plusieurs outils.

On peut les considérer comme des applications commerciales des techniques de fusion. Ils sont une des méthodes de recherche d'information commercialisées et opèrent en interrogeant pour chaque requête qui leur est soumise différents outils de recherche, afin de lui fournir la réponse la plus exhaustive. Pour leur fonctionnement, certains indexent l'information contenue dans différents annuaires et moteurs, d'autres les interrogent simultanément de façon dynamique. Le problème n'est pas simple car chaque outil de recherche a ses particularités. S'ils sont capables de repérer l'outil disponible le plus « fourni » sur un sujet et ils permettent une recherche rapide. Ils génèrent, en revanche, trop de bruit (réponses non-pertinentes), et ils ne profitent pas des particularités fournies par chaque outil.

Nous proposons à partir des travaux de Lardy [Lardy 2002] et Largouet [Largouet 2000] une synthèse des points importants relatifs aux méta-moteurs (un tableau comparatif des méta-moteurs est proposé dans l'annexe A, le tableau.1).

- **Choix des outils** : certains méta-moteurs permettent de choisir les outils à interroger. Ainsi, DigiSearch[7] offre le choix d'outils pour rechercher sur le web, les groupes de news et des adresses de messagerie.
- **Requête** : le fait d'interroger des outils de recherche différents implique que le choix des mots clés et leur combinaison avec des opérateurs soient adaptés en fonction du type d'outil utilisé. Donc, la requête doit en principe être simple. Les méta-moteurs ne permettent pas tous de formuler des requêtes complexes (utilisant différents opérateurs et des expressions parenthésées). Cependant certains outils adaptent la requête à la syntaxe de chaque outil et tentent en partie de remédier à la diversité des possibilités d'interrogation des différents outils mais leur travail n'est pas parfait. Par exemple, DogPile[8] formule la requête pour chaque moteur interrogé et présente ensuite la façon dont la requête a été formulée à chaque moteur. On peut utiliser les opérateurs booléens et de proximité AND, OR, NEAR et NOT pour combiner des mots et des phrases. NEAR est remplacé par AND pour les outils ne le gérant pas. DogPile supprime l'opérateur NOT et les mots suivants pour les outils ne le gérant pas. L'opérateur AND est implicite. On peut utiliser les parenthèses et les guillemets et les opérateurs unaires comme « – » et « + ». Le résultat est classé uniquement par outil de recherche, sans traitement de doublons.
- **Résultats** : le traitement des résultats est très variable. Certains méta-moteurs limitent le nombre de réponses obtenues de chacun des différents outils interrogés (les 10 à 20

[7] Site Web : http://www.digiway.com/digisearch/.
[8] Site Web : http://www.dogpile.com/.

premières réponses fournies par chacun), ce qui devrait permettre, en principe, d'obtenir les réponses les plus pertinentes. Highway 61[9] permet de préciser le volume de résultats désirés. D'autres méta-moteurs ne synthétisent pas les résultats. Ainsi All4one[10] affiche les résultats des différents outils dans des fenêtres indépendantes. D'autres, encore, font un classement par outils de recherche (DogPile). Enfin, certains méta-moteurs après avoir récupéré les différents résultats, les fusionnent avec élimination des doublons (Ixquick[11]) ou les classent par pertinence (MetaCrawler[12]).

- **Temps** : les temps de réponse des moteurs de recherche utilisés pouvant être très variables, certains méta-moteurs (Ixquick) utilisent une valeur d'expiration ou *time out*. Pour d'autres, l'utilisateur a un contrôle total sur la durée de la recherche (DogPile, Highway 61).

3.6. Conclusion

Nous avons essayé de donner une vision synthétique de la fusion de données. Après avoir dégagé la problématique qui y était attachée, nous avons décrit des travaux s'attachant à comprendre les raisons qui font que la fusion permet ou non d'améliorer les performances d'une recherche d'information, à trouver des critères de fusion et les conditions qui améliorent sa performance.

Les études dans ce domaine sont encourageantes, même si la fusion implantée ne conduit pas toujours au succès espéré. Toutes ces études s'appuient sur l'idée que la combinaison de multiples sources de pertinence augmente la performance de la recherche quelles que soient ces sources : des formulations de requête, des systèmes de recherches différents, ou différentes collections d'information. Ces études ont conduit à appliquer certains principes de fusion dans la pratique comme l'illustre les méta-moteurs disponibles sur le Web.

L'objectif de notre travail est de faire collaborer un groupe d'utilisateurs effectuant ensemble une session de recherche d'informations. Lors de cette session collaborative chaque utilisateur peut être vu comme une « source de pertinence », et donc une mise en commun des efforts de recherche par la fusion des résultats et/ou des requêtes individuelles doit être envisagée. Ce qui nous conduit à résumer les points intéressants de ce chapitre ainsi :

- Les études qui ont été faites dans le domaine de la fusion n'ont pris à aucun moment en compte l'utilisateur, et ne font donc intervenir aucun critère concernant l'utilisateur. Il en résulte que nous voulons concentrer notre travail autour de l'utilisateur et orienter la fusion dans cette direction.
- Il y a certaine structure des requêtes dans les moteurs de recherche, souvent sous forme des mots liés par des opérateurs booléens, de proximité…etc, ces opérateurs ne sont pas tous toujours gérés par le moteur. Au cas où les utilisateurs dans notre contexte utiliseraient des moteurs de recherche différents, et si nous leurs proposons une requête, on peut procéder de façon similaire à celle de certains méta-moteurs en adaptant la requête au moteur de recherche de chaque utilisateur.

---

[9] Site Web : http://www.highway61.com/.
[10] Site Web : http://www.all4one.com/.
[11] Site Web : http://www.ixquick.com/.
[12] Site Web : http://www.metacrawler.com/.

- Le constat fait par Belkin [Belkin 1993] qu'il y a un effet de la personne qui effectue les requêtes sur la performance, nous pousse à étudier les effets des utilisateurs sur la performance de recherche non seulement en terme des requêtes effectuées mais aussi en terme des résultats obtenus et des différentes évaluations des résultats effectuées par les utilisateurs.
- En particulier, l'amélioration de la performance dans les cas de la fusion de plusieurs formulations de requête ou des résultats de plusieurs requêtes, nous amènent à penser que la performance peut être aussi améliorée en combinant les requêtes des différents utilisateurs, ou en combinant les résultats de ces requêtes.

CHAPITRE.4. Environnement de Soutien pour la Recherche d'Information Collaborative

Notre objectif est de soutenir un groupe d'utilisateurs dans sa tâche de recherche d'information, ce qui nous a conduit à construire un environnement collaboratif de soutien. Un tel environnement doit aider les utilisateurs dans leurs tâches de recherche d'information et créer une synergie entre les différents acteurs de recherche collaborative de façon à augmenter leur satisfaction individuelle et collective. Leur satisfaction peut être due :

- à une meilleure qualité des résultats obtenus.
- à un gain de temps pour obtenir des résultats satisfaisants.
- à une meilleure expression des besoins.

Si l'on reprend le schéma de la Figure 2.1 on doit aider l'utilisateur dans les différentes étapes : la définition de la problématique, l'expression de sa requête et sa formalisation liée au moteur de recherche utilisé.

Nous voulons que l'utilisateur puisse avoir lorsqu'il le désire des informations sur le fonctionnement de la session de recherche du groupe.

Nous avons adopté la démarche illustrée dans la Figure 4.1. *Démarche adoptée* pour la construction de l'environnement collaboratif de soutien, composée de trois modules. Le premier module présente l'ensemble d'informations stocké dans la mémoire collaborative à la disposition de l'environnement de soutien, ces informations sont les données primaires que l'environnement utilise pour effectuer l'aide. Le deuxième module définit une stratégie de soutien selon, d'une part, la typologie de l'utilisateur et, d'autre part, la typologie de groupe. Ces deux modules font l'objet de ce chapitre. Tandis que le troisième module clarifie la mise à jour et l'obtention du soutien personnalisé, ce dernier module fait l'objet du chapitre suivant.

Dans ce chapitre, nous définissons dans le paragraphe 4.1 les critères qui seront pris en compte ainsi qu'un certain nombre de mesures permettant d'évaluer le fonctionnement du groupe.

Nous donnons dans le paragraphe 4.2, une définition plus précise du soutien, en terme de tâches, de dimensions, et de données nécessaires pour obtenir un environnement de soutien. Nous faisons un paragraphe particulier 4.3 sur la typologie des utilisateurs lors d'une session de recherche, l'estimation de son efficacité et son comportement sont des éléments importants pour adapter le soutien (paragraphe 4.4).

Le paragraphe 4.5 propose un environnement collaboratif de soutien à partir d'une typologie de groupe et définit une stratégie de soutien selon cette typologie.

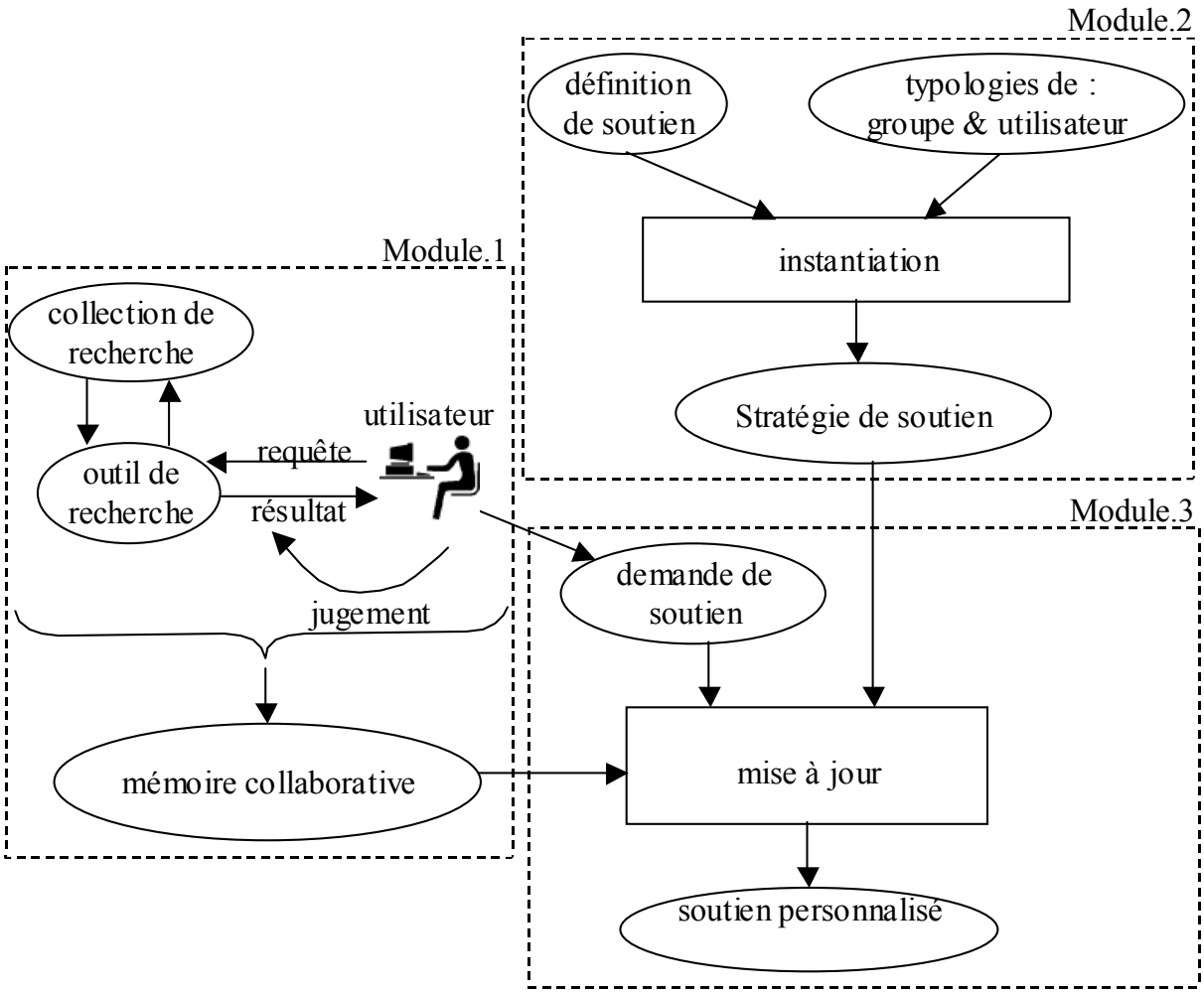

Figure 4.1. *Démarche adoptée.*

4.1. Modèle de Soutien

Lorsqu'un utilisateur veut obtenir de l'information pour un objectif de recherche donné, il va effectuer ce que nous appelons une session de recherche. En effet il est très rare que l'on obtient l'information recherchée en une seule étape c'est-à-dire en posant une seule requête. En général, l'on formule plusieurs requêtes afin d'essayer de trouver l'information que l'on estimera satisfaire notre objectif de recherche.

Nous pouvons donner les définitions suivantes :

- Définition d'une session individuelle de recherche : une session de recherche d'information est relative à un objectif de recherche donné effectuée avec un système donné. Elle est constituée d'un ensemble d'étapes.
- Définition d'une étape de recherche : une étape de recherche correspond à l'ensemble des documents évalués par l'utilisateur et retrouvés par le système en réponse à une requête donnée.

L'hypothèse restrictive à ces deux définition est que l'utilisateur au cours d'une session n'utilise qu'un seul moteur de recherche. S'il change de moteur de recherche, nous définissons une autre session de recherche.

Afin de pouvoir apporter un soutien personnalisé nous avons prévu des mesures permettant d'évaluer chaque session individuelle et chaque étape pour guider le processus de soutien, lors d'une session de recherche collaborative.

- Définition d'une session de recherche collaborative : une session de recherche collaborative est composée de l'ensemble des sessions individuelles des utilisateurs. L'hypothèse faite est qu'un groupe d'utilisateurs collabore pour trouver de l'information sur un sujet de recherche donné pouvant se décliner selon plusieurs objectifs.

Nous présentons la modélisation pour représenter et stocker les différents éléments dans la mémoire collaborative, ainsi que l'organisation de cette mémoire. Tout d'abord, nous présentons les principaux modèles de recherche collaborative : le modèle de session individuelle (paragraphe 4.1.1), et celui d'étape de recherche (paragraphe 4.1.2). Le modèle de session collaborative englobe naturellement les différentes sessions individuelles et il sera présenté ultérieurement (paragraphe 4.5.3.3).

4.1.1. *Session Individuelle*

Nous définissons une session individuelle (voir la Figure 4.2) de recherche d'information *Session-RII* de l'utilisateur *u*, de façon ensembliste, où l'écriture (..)* désigne un groupe multivalué d'élément ainsi :

Session-RII (*ID-utilisateur*, *Rôle-utilisateur*, *Objectif*, *InfoGénérale-RII*, *NbEtape*, *Durée-RII*, *Eval-RII, Etape-R* *)

Statique *Dynamique*

Il existe de nombreuses formalisations et notations tel que OMT (Object Modeling Technique) [Rumbaugh 1995]. Les différents modèles que nous proposons peuvent être décrits à l'aide de concepts empruntés au UML (Unified Modeling Language) [Muller 2000].

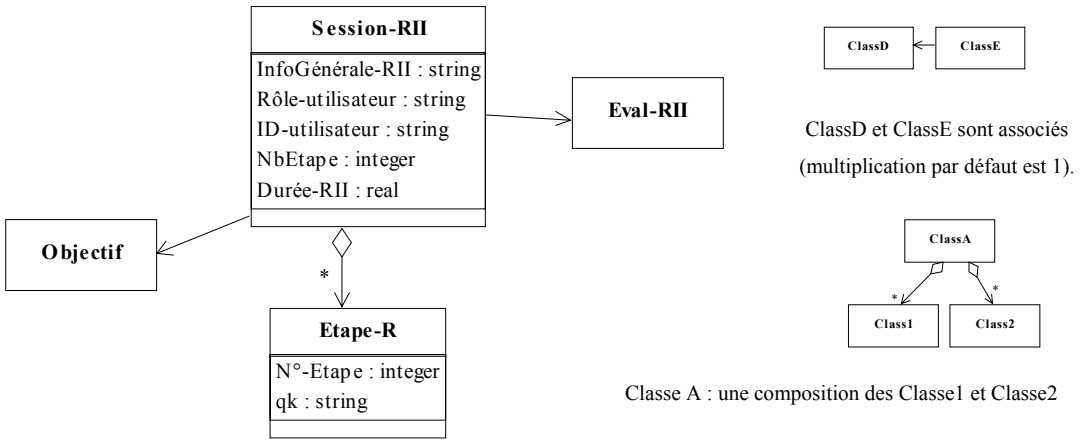

Figure 4.2. *Modèle d'une session individuelle.*

Le contenu de la session individuelle comporte deux types d'informations : des éléments statiques sur la session (paragraphe 4.1.1.1) et des éléments dynamiques et relatifs au déroulement de la session (paragraphe 4.1.1.2).

Cette distinction est nécessaire à cause des notions de base d'un environnement ou d'une aide synchrone, personnalisé et adaptatif. Le fait de préciser quels sont les éléments dynamiques conduit à détecter les changements de ces éléments chaque fois ils sont utilisés, cette détection permet de tenir compte de ces changements et par conséquence elle permet l'évolution de la personnalisation et de l'adaptation aux données les plus récentes dont le système dispose. Par contre les éléments statiques sont stockés au début de la session et ils seront référencés et consultés lorsque le système en a besoin.

4.1.1.1. Eléments Statiques

Les éléments statiques d'une session collaborative sont fixés au début de la session et ne changent pas jusqu'à sa fin. Il y a deux catégories d'éléments :

- *ID-utilisateur* qui identifie l'utilisateur qui effectue la session par son nom, son prénom et son adresse e-mail.
- *Rôle-utilisateur* qui définit le rôle de l'utilisateur dans la recherche collaborative ; ce rôle ne dépend pas directement de l'utilisateur mais il s'agit un paramètre de la recherche collaborative. Cet attribut ne fait pas partie des attributs de l'utilisateur, mais appartient au modèle de session individuelle. Par exemple, un utilisateur peut avoir le rôle de créateur de session pour une session collaborative *Session-RIC*$_x$, et le même utilisateur peut avoir le rôle d'un simple membre de groupe chargé d'un objectif précis de recherche dans une autre session collaborative *Session-RIC*$_y$. Le rôle peut être : soit *membre* soit *administrateur*.
- *Objectif* qui est l'objectif individuel de la recherche effectuée durant la session individuelle représenté par une chaîne de caractères qui décrit l'objectif.
- *InfoGénérale-RII* qui est défini par le triplet suivant :

  InfoGénérale-RII (*DateHeureDébut-RII*, *Info-Outil*, *Info-Collections*). Où :

  - *DateHeureDébut-RII* concerne le temps (date, et heure de début de la session).
  - *Info-Outil* est une description générale de l'outil de recherche que l'utilisateur a choisi pour sa session. Ce peut être un outil commercial par exemple un moteur de recherche sur le Web ou un outil spécialisé, comme un moteur interne. Nous supposons que les utilisateurs ne travaillent pas forcément tous avec le même moteur de recherche. Ainsi chacun peut choisir son moteur. La description de l'outil est composée de son *nom*, de la façon d'y *accéder* par exemple le nom de moteur de recherche est « google » et son adresse est http://www.google.fr/.
  - *Info-Collections* est une description générale de la collection de recherche (ou corpus) que l'utilisateur a choisi pour sa session. Nous supposons aussi que l'utilisateur peut

sélectionner la ou les collection(s) sur laquelle (ou lesquelles) il effectue sa recherche[1].

4.1.1.2. Eléments Dynamiques

Les éléments dynamiques sont des éléments temporaires et relatifs qui changent progressivement au fur et à mesure du déroulement de la session individuelle. Ces éléments sont :

- *NbEtape* qui est le nombre des étapes de recherche que l'utilisateur a effectuées à un moment donné au cours de sa session.
- *Durée-RII* qui est la durée de la session depuis son début jusqu'à un moment donné ou à la fin explicite de la session individuelle.
- *Eval-RII* qui représente l'estimation du succès de la session individuelle, la procédure de calcul de cette estimation fait l'objet du paragraphe 4.3.2.2.2.
- *Etape-R* * : une session individuelle est une itération de *NE* étapes de recherche (*NE* ≥ 1).

4.1.2. *Étape de Recherche*

Une étape de recherche (voir la Figure 4.3), va de la formulation d'une requête jusqu'à sa reformulation ou la fin de la session individuelle, s'il s'agit de la dernière étape. Les éléments d'une étape sont tous non-statiques, une fois l'étape finie, elle sera évaluée à l'aide des mesures que nous proposons dans le paragraphe 4.3.2.2.1.

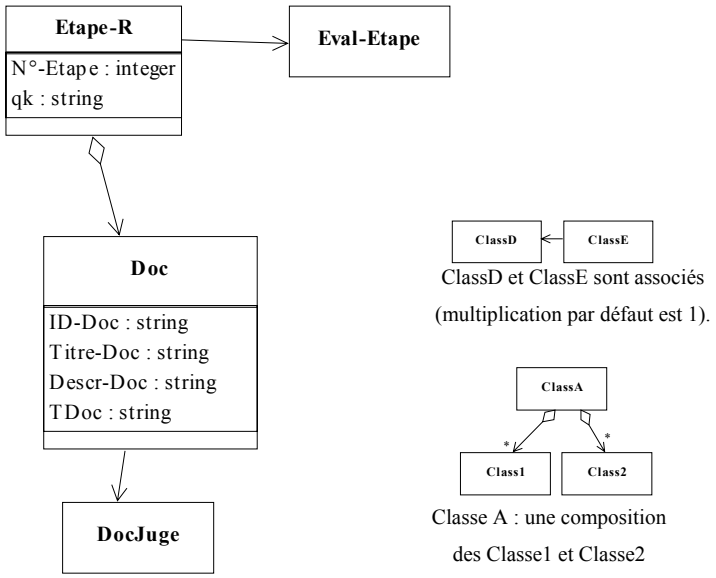

Figure 4.3. *Modèle d'une étape de recherche.*

[1] Généralement le choix d'outil de recherche implique le choix de(s) collection(s) de recherche.

Une étape de recherche *Etape-R* d'un utilisateur donné *u* est définie par le quadruplet suivant :

$$Etape\text{-}R\ (\underbrace{q_k, N°\text{-}Etape, (Doc, JugeDoc)^*, Eval\text{-}Etape_k}_{Dynamique})$$

où :

- q_k est la requête formulée par l'utilisateur. Le modèle de la requête varie selon le modèle de recherche d'information utilisé :
  - Pour le modèle vectoriel et le modèle probabiliste, la requête est un ensemble de mots clés éventuellement extraits d'une requête en langue naturelle.
  - Pour le modèle booléen, la requête est une expression booléenne construite avec des mots clé et les opérateurs (OU, ET, NON).
  - Pour un outil de recherche commercial, la requête est l'ensemble des mots clés combinés éventuellement par différents opérateurs qui varient selon l'outil commercial.
- *N°-Etape* est le numéro de l'étape (1, 2, …, *NE*).
- (*Doc*, *JugeDoc*)* est l'ensemble des doublets constitués des documents *Doc* sélectionnés par l'utilisateur parmi les documents retrouvés et de *JugeDoc* le jugement de qualité qui leur est associé.

  Un document possède les attributs suivants :

  Doc (*ID-Doc*, *Titre-Doc*, *Descr-Doc,* T_{Doc})

  où :
  - *ID-Doc* est l'identificateur du document.
  - *Titre-Doc* est le titre du document.
  - *Descr-Doc* est une description concernent le document comme le(s) *Auteur*(*s*), la méthode d'*accès* (son URL), sa *date*, son *format*, etc …
  - T_{Doc} l'ensemble des termes apparus dans le document si l'on dispose du résultat de l'indexation, ou dans les autres cas on ne prend que l'ensemble des termes apparus dans le titre *Titre-Doc*.
- $Eval\text{-}Etape_k$ évalue la qualité de l'étape en termes de résultats pertinents pour la requête *q* selon les mesures que nous proposons au paragraphe 4.3.2.2.1.

Le déroulement de la session individuelle peut se schématiser ainsi :

$$q_1 \xrightarrow{Etape\text{-}R_1} (Doc, JugeDoc)^*, q_2 \ldots, q_k \xrightarrow{Etape\text{-}R_k} (Doc, JugeDoc)^*, \ldots, q_{NE}$$

On notera qu'à une étape *i*, est attachée une expression de la requête q_i en début d'étape servant à rechercher des documents qui seront proposés au jugement de l'utilisateur.

4.2. Définition de Soutien

Afin de construire ce soutien, nous nous somme posés les questions suivantes : « Pour réaliser quelles *tâches* le soutien intervient-t-il (paragraphe 4.2.1) ? », « Comment le soutien sera-t-il effectué, c'est-à-dire selon quelles *dimensions* (paragraphe 4.2.2) ? » Enfin, « Comment obtenir les *données* nécessaires au soutien (paragraphes 4.2.3 et 4.2.4) ? » c'est-à-dire le jugement et la préférence.

Les paragraphes qui suivent sont consacrés à répondre à ces questions. Le schéma dans la Figure 4.4 ci-dessous permet d'avoir une vision globale des différents éléments.

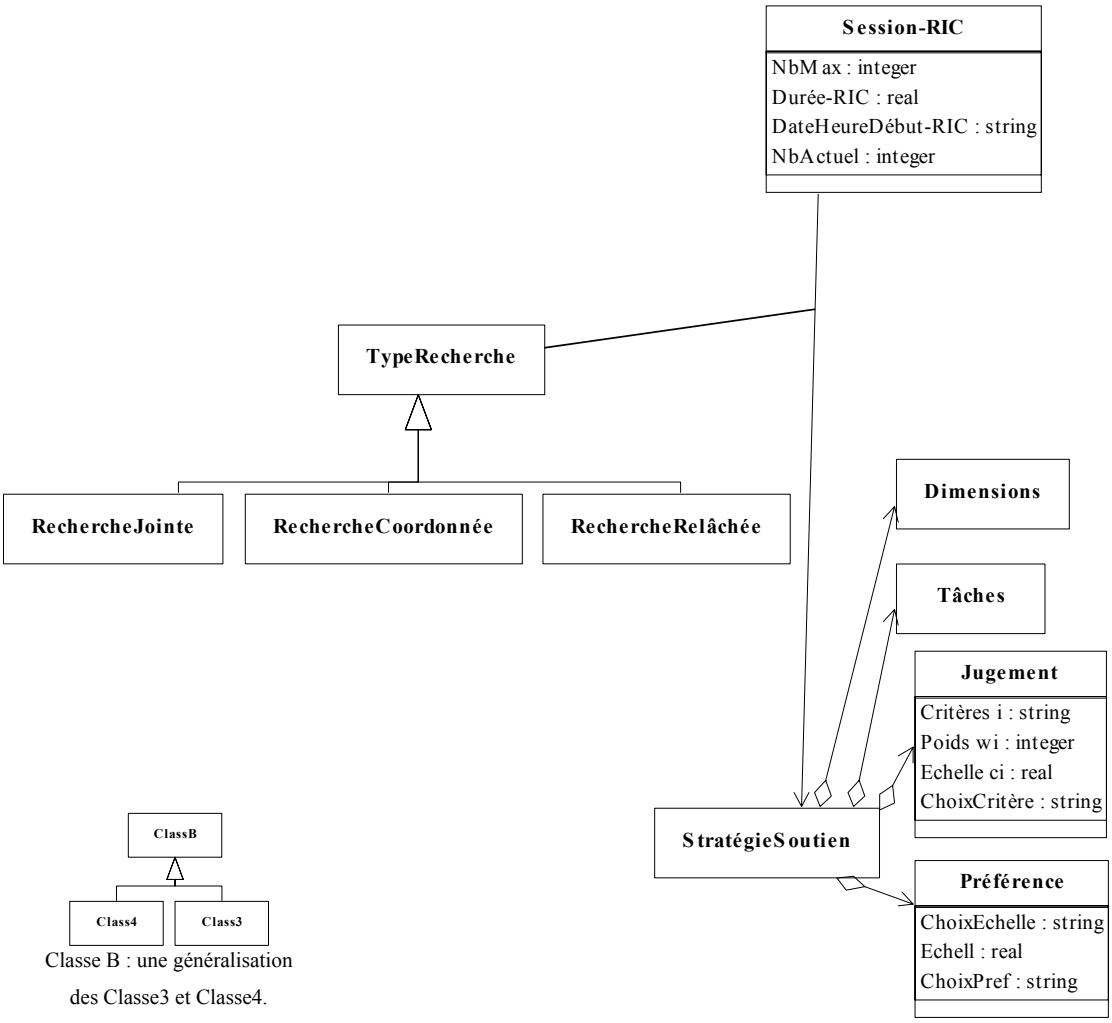

Figure 4.4. *Les éléments de soutien.*

4.2.1. *Tâches de Soutien*

Selon la définition[2] de Hoppe [Hoppe 1994] l'utilisateur en situation de recherche d'information, doit effectuer une série de tâches :

- créer une requête et la soumettre,
- visualiser ou parcourir le résultat de la requête et évaluer éventuellement ce résultat,
- raffiner la requête.

Un système de soutien pour le processus de recherche doit fournir à l'utilisateur en plus des fonctionnalités habituelles d'un système de recherche, des fonctionnalités l'aidant dans ces tâches. Ainsi, à chaque tâche, l'utilisateur peut choisir de travailler de façon individuelle ou utiliser le soutien et l'aide qu'un tel système de soutien pourrait lui fournir.

Le soutien dans un processus de recherche collaborative peut intervenir lors de différentes tâches :

- Lors de la formulation de requête : l'objectif est d'aider l'utilisateur à formuler la meilleure requête, c'est-à-dire exprimant au mieux son besoin d'information ; on peut pour cela lui présenter l'ensemble des requêtes du groupe (*présentation collaborative des requêtes*), ou lui construire une nouvelle requête à partir des requêtes de groupe (*requête collaborative*).
- Lors de la visualisation du résultat : on peut fournir aux utilisateurs un *résultat collectif* et synthétique des résultats obtenus par le groupe.
- Lors de la reformulation de requête : (ou *bouclage de pertinence collaboratif*) on utilise l'ensemble des requêtes, des résultats, et des « feedbacks » du groupe pour reformuler une requête.

Les valeurs possibles concernant le paramètre tâche sont résumées dans le Tableau 1.1.

Tâches	Types de Soutien
formulation de requête	• présentation des requêtes • construction d'une requête
visualisation de résultat	• présentation des résultats
bouclage de pertinence	• application d'une méthode de feedback sur des requêtes et des résultats jugés

Tableau 4.1. *Résumé des tâches.*

[2] « *Information search in databases should be a highly interactive process in which the steps of target recognition (i.e. result browsing) and query formulation are iterated to approximate an optimal result.* »

4.2.2. *Dimension de Soutien*

Nous distinguons plusieurs niveaux de soutien, selon les situations de recherche dans lesquelles se trouve l'utilisateur. Nous avons choisi de définir ces niveaux selon trois dimensions orthogonales : les deux dimensions temporelles : le *temps absolu* (paragraphe 4.2.2.1) et le *temps dans la session* (paragraphe 4.2.2.2) et la dimension *groupe* (paragraphe 4.2.2.3). Ces différentes dimensions vont nous permettre d'exprimer les différentes situations ou contextes de recherche (paragraphe 4.2.2.4).

4.2.2.1. Dimension Temps Absolu

Les recherches des différents membres du groupe peuvent être *synchrones* ou *asynchrones*. Cette notion de synchronisation intervient lorsque les utilisateurs cherchent en *même temps*. Tout utilisateur, pendant sa session de recherche, peut demander l'aide et donc être soutenu par le système en temps réel de façon presque synchrone. Par contre, on parle de recherches *asynchrones* dans le cas où chacun des utilisateurs effectue sa recherche à des instants différents.

Notre étude a montré que la distinction synchrone-asynchrone n'a pas une grande influence sur le système ; en effet la plupart des aides définies pour une recherche synchrone peuvent s'appliquer à une recherche asynchrone. Une discussion intéressante à ce sujet se trouve dans [Sire 1999] :

« La distinction suivant la dimension temporelle ne repose pas uniquement sur l'observation de l'activité des groupes. Elle résulte également des différentes techniques de transmission de données utilisées par les applications qui n'ont pas les mêmes contraintes dans les deux cas. Cependant, cette séparation peut apparaître artificielle si l'on considère l'activité d'une personne comme se déroulant sur plusieurs plans en même temps. En effet, une activité synchrone peut se transformer en activité asynchrone si elle passe dans l'arrière-plan de l'activité. Par exemple il est toujours possible de « geler » une conversation au cours d'un dialogue en ligne pour y revenir par la suite. Ce type de comportement est simplifié si l'application de dialogue conserve les derniers messages arrivés, de façon à retrouver les échanges des autres participants après une absence. Nous voyons ici que les contraintes assignées (synchrone) n'empêchent pas un usage asynchrone (récupération des messages passés). D'autres exemples suggèrent que la transition synchrone/asynchrone, qui peut être vue comme une transition entre le premier plan et l'arrière plan de l'activité, ne nécessite pas seulement de changer d'application mais nécessite également de recourir à quelques artefacts pour faciliter les transitions, comme l'historique des conversations dans le cas de dialogue en ligne. »

Alors, pour un groupe d'utilisateurs effectuant une recherche collaborative de façon synchrone avec une aide en temps réel, cela peut se transformer en recherche et soutien collaboratif asynchrones, si le système stocke toutes les informations d'une session une fois celle-ci terminée, il est donc possible de les utiliser de façon asynchrone. Par exemple, un nouvel utilisateur ou groupe peut consulter ultérieurement ces informations et être soutenu dans sa recherche de façon asynchrone par les informations de la session de recherche précédemment enregistrée.

4.2.2.2. Dimension Temps dans la Session

Cette dimension tient compte de l'*historique* de la session de recherche à savoir toutes les étapes de recherches (les requêtes soumises, les résultats évalués) depuis le début de la session jusqu'à l'étape présente. La notion de recherche *courante* signifie que l'on ne prend en compte que la « dernière » étape de recherche (c'est-à-dire la « dernière » requête soumise, et son résultat évalué).

4.2.2.3. Dimension Groupe

Selon cette dimension, on prend en compte la façon de travailler de l'utilisateur. Elle peut se décliner selon deux aspects : l'aspect *collaboratif* et l'aspect *individuel*. Ces deux aspects correspondent à la prise en compte pour le soutien des sessions de recherches de plusieurs utilisateurs, et de la session de recherche d'un seul utilisateur, respectivement.

4.2.2.4. Situations/Contextes de Recherche

Les valeurs possibles concernant les dimensions sont résumées dans le Tableau 4.2.

Dimensions	Valeurs
temps absolu	• synchrone • asynchrone
temps dans la session	• courant • historique
groupe	• collaboratif • individuel

Tableau 4.2. *Résumé des valeurs des dimensions.*

Les niveaux de soutien ne peuvent s'exprimer qu'en considérant les deux dimensions *groupe* et de *temps dans la session*, du fait de leur implication réciproque. Nous mettons ainsi en évidence différents niveaux de soutien (voir Figure 4.5) :

- sur le plan synchrone, il y a trois niveaux de soutien possibles numérotés (1, 2, 3) qui peuvent être fournis en temps réel.
  - dans un contexte individuel, nous n'envisageons qu'un seul niveau de soutien. En effet, nous pensons que la seule aide que l'on puisse apporter à l'utilisateur est une sorte de synthèse « intelligente » de sa session de recherche. Par exemple, on peut l'aider pour la formulation d'une requête en utilisant les différentes requêtes déjà soumises. Ainsi, ce niveau de soutien ne peut exister que lorsque l'utilisateur a procédé à plusieurs étapes de recherche et que l'on possède un historique de son travail. Ce soutien sera désigné par *Soutien Individuel Historique et Synchrone* (numéroté 1).
  - si nous considérons le contexte collaboratif, il y a deux niveaux de soutien possibles (numéroté 2, 3) : le *Soutien Collaboratif Historique et Synchrone* qui tient compte des historiques de plusieurs sessions de recherches individuelles, et le *Soutien Collaboratif Courant et Synchrone* qui prend en compte la dernière étape de chaque

session individuelle. Dans ce cas, le soutien prend en compte non plus des données issues d'une seule recherche individuelle mais des données issues de plusieurs recherches individuelles effectuées par des utilisateurs différents.

- sur le plan asynchrone, la notion d'étape courante n'existe plus puisque la session est déjà finie. Il y a deux niveaux de soutien numérotés (1', 2'), on parle alors de *Soutien Individuel Asynchrone*, et de *Soutien Collaboratif Asynchrone* qui sont toujours *Historiques*.

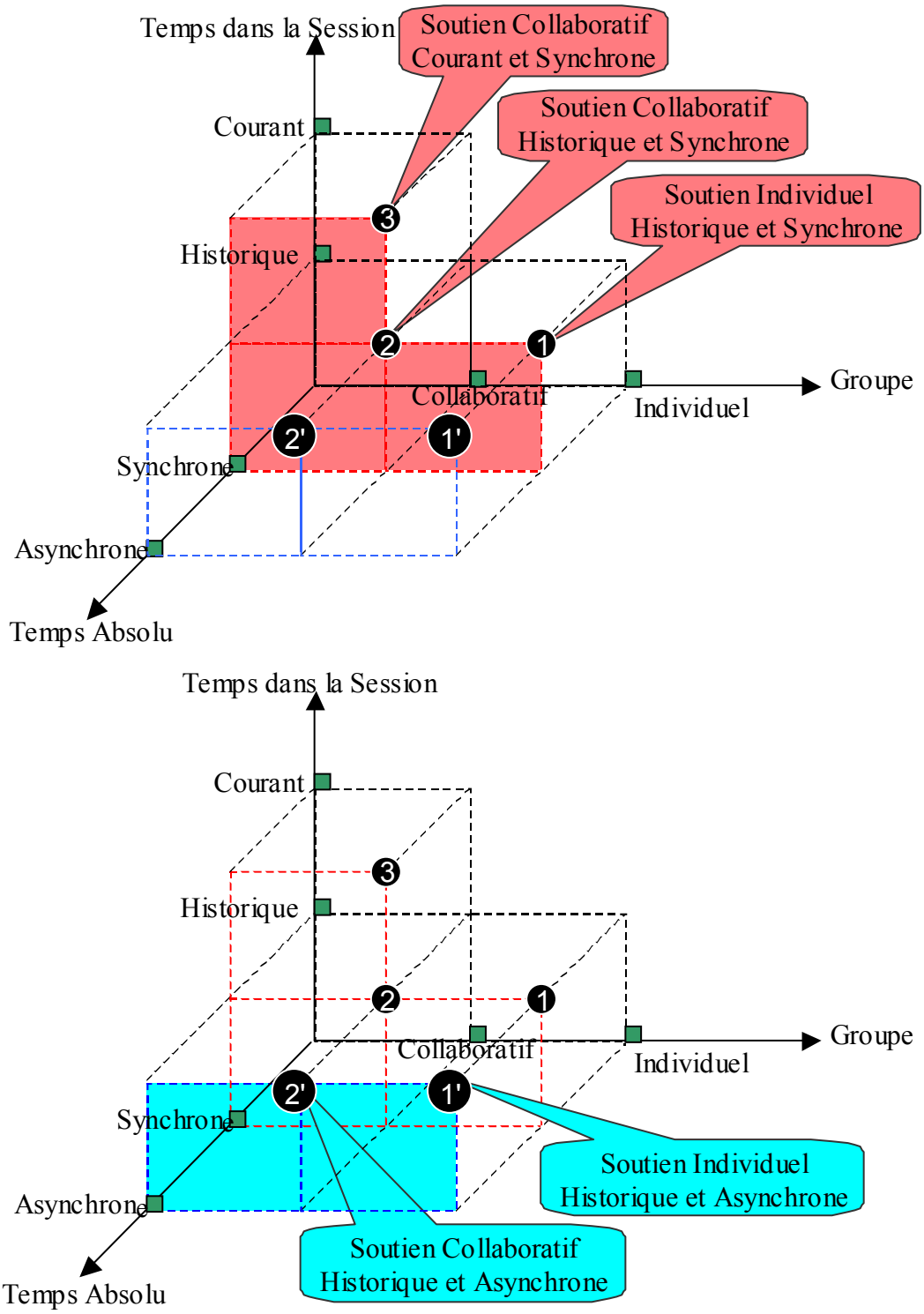

Figure 4.5. *Dimensions de soutien.*

Dans cette étude, nous concentrons sur le *Soutien Collaboratif Synchrone* selon l'axe *Courant* et l'axe *Historique* (numérotés 2, 3).

4.2.3. *Jugement*

L'utilisateur juge la pertinence d'un document par rapport à son besoin d'information. Pour que l'utilisateur puisse fournir ce *jugement*, il faut préciser les paramètres suivants :

- *critères* : l'ensemble des critères de qualité, selon lesquels les documents seront évalués, (paragraphe 4.2.3.1).
- *expression* : l'échelle de valeurs de jugement, (paragraphe 4.2.3.2).
- *poids* : le poids qui représente l'importance accordée à chaque critère, (paragraphe 4.2.3.3).
- *responsabilité* : il faut déterminer qui a la responsabilité de préciser les trois paramètres précédents, (paragraphe 4.2.3.4).

Finalement, dans le paragraphe 4.2.3.5 nous examinons un procédé d'évaluation de document.

4.2.3.1. Critères de Jugement

Le jugement attribué aux documents peut être donné selon un ou plusieurs critères de pertinence, qui relèvent de deux types :

- Critères de pertinence habituels concernant le document et sa qualité. Dans le paragraphe 2.1.6 nous avons présenté de nombreuses études qui se sont appliquées à évaluer cette qualité, à essayer de la mesurer, et à identifier l'ensemble des facteurs de pertinence. Pour juger la qualité d'un document, nous proposons de juger sa pertinence par rapport :
  - au contenu : le document traite-il le sujet qui intéresse l'utilisateur ?
  - à la date : le document propose-t-il une nouvelle étude ?
  - aux auteurs : les auteurs de document sont-ils compétents dans le domaine, réputés, … etc ?
- Critères de pertinence qui prennent en compte la dimension collective : l'utilisateur peut juger certains documents pertinents car ils sont intéressants pour un ou plusieurs membres du groupe. Le critère prend tout son sens si on considère que les utilisateurs s'intéressent à des aspects différents du sujet de recherche commun. Cela nous conduit à définir deux critères :
  - pertinence pour les autres : un document est intéressant pour tous.
  - pertinence pour quelqu'un en particulier : un document est intéressant par rapport au centre d'intérêt d'un utilisateur que l'on précise.

L'utilisateur dispose donc d'un ensemble *EC* de critères qui peut consister : d'un seul critère (le contenu), de plusieurs critères comme (le contenu, la date, et les auteurs), ou d'autres critères que le groupe a la liberté de choisir.

4.2.3.2. Expression de Jugement

Dans le paragraphe 2.1.6 nous avons présenté deux catégories d'expression de jugement : le score et l'ordonnancement. Nous considérons ici le premier, ainsi nous permettons aux utilisateurs d'exprimer leur jugement en affectant une valeur qui peut être : binaire (non-pertinente, pertinente), ou une échelle de valeurs, comme celle proposées par Maltz [Maltz 1994] qui contient quatre valeurs (mauvais, moyen, bien, très bien)[3].

4.2.3.3. Poids de Critères de Jugement

Les critères de jugement (leur nombre est désigné par *NC*) peuvent contribuer différemment au jugement global d'un document, où on peut considérer que les poids w_i des critères $i \in \{1, 2, \ldots, NC\}$ sont :

- égaux.
- différents.

4.2.3.4. Détermination de Paramètres de Jugement

Les critères de jugement, leur nombre, leur poids, et la façon d'exprimer leurs valeurs peuvent être déterminés de différentes façons par :

- le *groupe* : on laisse le groupe choisir les paramètres au début de la session collaborative, pour qu'ils soient adaptés au sujet de la recherche.
- le *système* : on prédéfinit ces paramètres lors de la conception du système.

4.2.3.5. Jugement du Document

L'utilisateur a un ensemble *EC* de *NC* critères de qualité *i*, cet ensemble lui permet de juger la pertinence du document obtenu en réponse à sa requête *q*. c_i est la valeur de pertinence du critère *i* et appartient à l'intervalle [0, 1], w_i l'importance de ce critère pour le jugement final. Les paramètres concernant les données de jugement sont résumés dans le Tableau 4.3.

Le jugement de chaque document est alors évalué par la moyen pondérée suivante :

$$JugeDoc = \frac{\sum_{i=1}^{NC} w_i * c_i}{\sum_{i=1}^{NC} w_i} \qquad i \in EC \; ; EC = \{1, 2, \ldots, NC\} \qquad [4.1]$$

L'évaluation précédente est une évaluation *globale* du document par rapport à tous les critères de jugement $i \in EC$. Mais une évaluation *partielle*, c'est-à-dire par rapport à certains critères de jugement, peut être intéressante pour un utilisateur qui s'intéresse aux documents selon des points de vue précis. Par exemple, supposons qu'un utilisateur veuille faire un état de l'art complet à propos de son sujet. Il s'intéresse donc à des documents pertinents quelle

[3] terrible, ok, good, or great.

que soit l'ancienneté de document. Il faut donc prendre en compte le point de vue du contenu, et ignorer celui de la date.

Ainsi, l'évaluation *partielle* d'un document pour un sous-ensemble *SEC* de l'ensemble *EC* de critères de jugement est calculée de la façon suivante :

$$JugeDoc = \frac{\sum_{i \in SEC} w_i * c_i}{\sum_{i \in SEC} w_i} \qquad i \in SEC \; ; \; SEC \subset EC \qquad [4.2]$$

Paramètres	Valeurs
critères de jugement i	• un seul : $NC = 1$, EC = {le contenu} • plusieurs : $NC > 1$, EC = {le contenu, la date, les auteurs} • plusieurs : $NC \geq 1$ EC = {critères}
poids d'importance des critères w_i	• même poids • poids différents
échelle de valeur c_i	• binaire • réelle
choix des critères, des poids et de l'échelle	• groupe • système

Tableau 4.3. *Résumé des paramètres du jugement.*

4.2.4. *Préférence*

Pour ce qui est de l'« *expression* » de la préférence, nous proposons une échelle numérique de N valeurs [0, ..., N], où N est la valeur maximale pour l'utilisateur auquel la plus grande préférence est donnée. La préférence doit être dans l'intervalle [0, 1], alors une normalisation sera appliquée, les valeurs de l'échelle numérique sont divisées par N. L'utilisateur peut donner le même degré de préférence à plusieurs utilisateurs, de même il peut s'attribuer à lui-même un degré de préférence. Cette échelle de valeurs peut être *déterminée* par le *groupe* au début de la session collaborative, ou par le *système* lors de sa conception.

Pour le jugement, ce sont les utilisateurs qui donnent leur opinion par rapport aux documents alors que pour la préférence, cela dépend de la connaissance qu'ils ont des membres du groupe. Dans le cas où ils se connaissent, la préférence donne des indications sur les liens sociaux entre les utilisateurs. L'utilisateur peut ainsi attribuer des valeurs de préférence élevées aux utilisateurs qui sont compétents ou qu'il juge comme tel ou qui ont le même comportement que lui (par exemple les utilisateurs qui sont aussi pressés que lui), … etc. Mais dans le cas où ils ne se connaissent pas ou peu, la préférence sera déterminée par le système, ou assistée par lui une explication plus détaillée est présentée ultérieurement dans le paragraphe 4.4.3.

Le type de collaboration, permet de définir plus précisément le choix des valeurs des paramètres de préférence. Les paramètres concernant les données de préférence sont résumés dans le Tableau 4.4.

Nous avons expliqué ci-dessus l'échelle de valeurs de préférence et qui a la responsabilité de déterminer cette échelle ainsi que la valeur de préférence elle-même.

Paramètres	Valeurs
échelle de valeur	• numérique
responsabilité du choix des valeurs de l'échelle	• le groupe ou • le système
responsabilité de la distribution de la préférence	• l'utilisateur ou • le système ou • l'utilisateur et le système

Tableau 4.4. *Résumé des paramètres de la préférence.*

La préférence étant liée à l'utilisateur, nous avons étudié les caractéristiques de l'utilisateur par rapport auxquelles la valeur de préférence peut être fournie c'est-à-dire établi une typologie de l'utilisateur.

4.3. Typologie de l'Utilisateur

4.3.1. *Introduction*

L'utilisateur, membre d'un groupe, peut être caractérisé par ses compétences et son comportement pendant la recherche. La préférence peut être donnée à l'utilisateur selon une ou plusieurs de ses caractéristiques, que nous expliquons dans la suite :

- *Organisation* : cette caractéristique concerne aussi bien l'organisation du lieu du travail, que son organisation *hiérarchique* et *temporaire* ; par exemple, les membres du même groupe de travail, le chef du groupe, les membres permanents ou les membres temporaires comme les stagiaires, … ont des caractéristiques différentes.
- *Compétence de l'utilisateur* : la compétence peut être variable en fonction de l'expérience de l'utilisateur concernant l'exécution du processus de recherche d'information d'une part, et/ou le domaine d'information ciblé d'autre part. Nous distinguons deux types de compétence :
  - la compétence technique : c'est-à-dire l'habitude qu'a l'utilisateur d'utiliser des systèmes de recherche d'informations.
  - la compétence concernant le domaine de recherche, à savoir le niveau d'expertise de l'utilisateur dans le domaine sur lequel il effectue sa recherche.

  Dans le projet IOTA [Defude 1986], on établit une typologie de l'utilisateur par rapport à ces deux types de compétence, se pose alors les questions suivantes :

  « Comment évaluer la connaissance que l'utilisateur a sur le système ? » L'hypothèse de travail était : Si l'utilisateur connaît bien le système, il utilise les « bons termes », c'est-à-dire les termes d'indexation connus par le système.

« Comment évaluer la connaissance que l'utilisateur a sur le domaine ? » Cela est fait via les spécificités des concepts employés dans un thésaurus.

Le Tableau 4.5 donne une typologie de l'utilisateur selon trois niveaux (*débutant*, *moyen* et *expert*).

- *Comportement de l'utilisateur pendant la recherche* : ces attributs concernent son attitude pendant la session de recherche. L'utilisateur peut, par exemple, être *pressé* d'obtenir un résultat, être *rapide*, ou être *efficace* dans son processus de recherche.

Compétence dans le Domaine	Compétence dans le Système	Type de l'Utilisateur
Débutant	Débutant	Débutant
Débutant	Moyen	Débutant
Débutant	Expert	Moyen
Moyen	Débutant	Moyen
Moyen	Moyen	Moyen
Moyen	Expert	Moyen
Expert	Débutant	Moyen
Expert	Moyen	Expert
Expert	Expert	Expert

Tableau 4.5. *Typologie de l'utilisateur selon ses compétences.*

Nous constatons que ces caractéristiques peuvent être *objectives* ou *subjectives*, *dépendantes* ou *indépendantes* des événements apparaissant lors d'une session de recherche, *permanentes* ou *temporaires*. Certaines de ces caractéristiques doivent être fournies *manuellement* au système au début de la recherche (c'est le cas, si l'utilisateur est pressé, par exemple). D'autres caractéristiques peuvent être déduites *automatiquement* par le système, comme la rapidité ou l'efficacité d'un utilisateur. Sa rapidité peut être évaluée par le nombre de requêtes effectuées dans un même laps de temps. Son efficacité peut être évaluée par le nombre de documents jugés pertinents. Le Tableau 4.6 suivant résume les attributs de l'utilisateur.

Organisation hiérarchique temporaire		objective	manuelle	indépendante	temporaire
Compétence technique, domaine de recherche		subjective	manuelle	indépendante	temporaire
Comportement	**pressé**	subjective	manuelle	indépendante	temporaire
	rapidité, efficacité	subjective	automatique	dépendante	temporaire

Tableau 4.6. *Caractéristiques de l'utilisateur.*

Nous allons dans le paragraphe suivant, définir des mesures nous permettant de qualifier chaque utilisateur du groupe.

4.3.2. *Evaluation de l'Utilisateur*

Afin de pouvoir aider correctement un utilisateur, nous pensons que le système doit pouvoir analyser son comportement et essayer d'estimer ses compétences. Peu d'étude ont été faites et la seule qui va dans ce sens est celle de Zhang où il essaie de qualifier un utilisateur par rapport aux autres membres du groupe en estimant sa contribution au consensus obtenue. Nous proposons tout d'abord cette approche (paragraphe 4.3.2.1), puis nous expliquons notre approche pour parvenir à cette estimation (paragraphe 4.3.2.2).

4.3.2.1. Méthode de Consensus de Groupe

Zhang [Zhang 2002], propose une mesure de la capacité d'un individu à retrouver les documents et, seulement ceux, qui sont dans l'ensemble de documents retrouvés par les autres utilisateurs du groupe. Nous exprimons la *méthode de consensus de groupe* de la manière suivante :

Pour une problématique/question de recherche donnée, la méthode construit l'*ensemble consensus* C pour cette question, cet ensemble contient les documents retrouvés par un nombre d'utilisateurs supérieur ou égal à un seuil[4] donné. A chaque document dans cet ensemble est assigné un poids w_i qui est le nombre des utilisateurs ayant retrouvé le $i$ème document, les documents qui ne sont pas dans cet ensemble ont le poids zéro. L'ensemble consensus est considéré comme l'ensemble des documents pertinents pour l'ensemble du groupe.

Prenons D_u comme l'ensemble de documents retrouvés par l'utilisateur u pour cette question. La mesure du score de pertinence $Eval(u)$ est définie ainsi :

$$Eval\,(u) = \frac{\sum_{i=1}^{|D_u|} w_{ui}}{|C| + \left(|C| - |D_p| \right) + |D_{np}|}$$

Où $|D_u|$ est le nombre de documents dans D_u ; $|C|$ est le nombre de documents dans l'ensemble consensus C ; w_{ui} est le poids assigné au $i$ème documents retrouvé par l'utilisateur u ; $|D_p|$ est le nombre de documents pertinents que l'utilisateur a retrouvé c'est-à-dire le nombre de documents que l'utilisateur a retrouvés et qui sont inclus dans l'ensemble de consensus $D_p \subset C$; et $|D_{np}|$ est le nombre de documents non-pertinents que l'utilisateur a retrouvés, c'est-à-dire le nombre de documents que l'utilisateur a retrouvés et qui n'apparaissent pas dans l'ensemble de consensus $D_p \not\subset C$.

$(|C| - |D_p|)$ calcule le nombre de documents pertinents qui ont échappé à l'utilisateur (ceux qui sont inclus dans l'ensemble de consensus mais que l'utilisateur n'a pas retrouvé).

La valeur maximale qu'un utilisateur peut obtenir est :

[4] Le seuil peut être déterminé par le système, selon le nombre des utilisateurs dans le groupe et le nombre des documents retrouvés.

$$Eval_{idéale}(u) = \frac{\sum_{i=1}^{|D_u|} w_{ui}}{|C|}$$

La situation idéale correspond au cas où l'utilisateur u a retrouvé seulement les documents pertinents ($D_u = C$). Nous remarquons que la valeur $Eval(u)$ est comprise entre 0 et $Eval_{idéale}(u)$.

Puisque nous voulons que la valeur de préférence soit dans l'intervalle [0, 1], nous normalisons $Eval(u)$ par l'évaluation maximale qu'un utilisateur a obtenu $Eval_{max}$. La valeur de préférence est :

$Pref(u) = \frac{Eval(u)}{Eval_{\max}}$: $Eval_{max} \geq Eval(u)$ pour tous les utilisateurs u de la session collaborative.

Nous ne retenons pas cette mesure pour plusieurs raisons, tout d'abord, la construction de l'ensemble consensus est basée sur le fait que les utilisateurs de groupe ont la même question de recherche à résoudre, cependant notre approche est plus générale parce que l'on voudrait évaluer les utilisateurs de groupes qui recherchent à propos d'un sujet même si leurs questions de recherche ou leurs objectifs sont différents. Avec l'application de cette mesure dans notre approche nous risquons de ne pas avoir un grand nombre de documents dans l'ensemble consensus à cause de la diversité des objectifs. De plus, cette mesure ne prend pas en compte le jugement des utilisateurs, tandis que nous disposons d'une telle information qui permet d'évaluer les utilisateurs plus précisément. Enfin, selon l'auteur, la fiabilité de la méthode a besoin d'être testée et expérimenté pour l'améliorer.

4.3.2.2. Evaluation de l'Efficacité de l'Utilisateur

La question qui se pose dans le contexte d'un environnement de soutien est d'évaluer le travail d'un utilisateur donné pendant la session de recherche.

Dans cet environnement, nous ne connaissons de l'utilisateur que sa session individuelle dont les résultats, c'est-à-dire requêtes et documents retrouvés et jugés, sont stockés dans la mémoire collaborative.

Nous allons définir deux mesures basées sur ces résultats afin d'évaluer : quelle est la capacité de l'utilisateur à exprimer correctement son besoin (formule [4.5]), et sa capacité à utiliser le mieux possible le système (formule [4.6]).

Nous examinons ici, dans un premier temps l'évaluation d'une étape de recherche (paragraphe 4.3.2.2.1) et ensuite l'évaluation d'une session individuelle (paragraphe 4.3.2.2.2).

4.3.2.2.1. Évaluation d'une Etape de Recherche

La requête q_u a retourné ND documents jugés par l'utilisateur, l'évaluation de cette étape k de recherche est définie par $Eval\text{-}Etape_k$. On distingue deux mesures pour cette évaluation : $EvalPrec\text{-}Etape_k$ et $EvalOrd\text{-}Etape_k$, qui ont pour objectif de mesurer la précision des réponses pour l'utilisateur :

- Evaluation de la requête par rapport au sujet de recherche :

  Cette mesure correspond à la validité de la requête pour le sujet de la recherche en cours, ou plus exactement la satisfaction exprimée par l'utilisateur sur les documents choisis.

$$EvalPrec\text{-}Etape_k = \frac{1}{ND_{20}} \sum_{j=1}^{ND_{20}} JugeDoc_j \qquad [4.3]$$

  Où : $ND_{20} = Min\ (20, ND)$. Nous avons considéré, à titre d'exemple, les 20 premiers documents retournés par le système de recherche d'information utilisé.

  Cette mesure nous indiquera l'efficacité de l'utilisateur pour formaliser son besoin d'information.

- Evaluation de la requête par rapport à la pertinence système :

  La mesure précédente ne nous renseigne pas complètement sur le succès de la requête, car les documents jugés pertinents par l'utilisateur peuvent se trouver dans n'importe quelle position parmi les résultats ordonnés selon la *pertinence système.* La question que l'on peut se poser est la suivante : « les documents jugés pertinents par l'utilisateur sont-ils ou non classés en tête par le système ? ». Nous utilisons une mesure inspirée de [Chignell 1999] (voir le paragraphe 2.2.2.2), appelée « *précision différentielle* » *PrecDiff*, qui peut affiner la mesure précédente et donner une meilleure évaluation de l'étape, donc de la requête *q* pour le système utilisé et le sujet de recherche

$$PrecDiff\ (1,ND_{20}) = [\frac{1}{ND_{10}} \sum_{j=1}^{ND_{10}} JugeDoc_j] - [\frac{1}{ND_{20}-ND_{10}} \sum_{j=ND_{11}}^{ND_{20}} JugeDoc_j]$$

  - *PrecDiff* > 0, signifie qu'il y a plus de documents pertinents parmi les 10 premiers documents que parmi les dix suivants.
  - *PrecDiff* = 0, signifie qu'il y a autant de documents pertinents parmi les 10 premiers que parmi les 10 suivants.
  - *PrecDiff* < 0, signifie qu'il y a moins de documents pertinents dans les 10 premiers documents que dans les 10 suivants.

  Alors $PrecDiff \in [-1, +1]$, afin que cette valeur soit dans l'intervalle [0, 1] Nous normalisons ainsi :

$$EvalOrd\text{-}Etape_k = ½\ [1+ PrecDiff(1,ND_{20})] \qquad [4.4]$$

  Donc : $0 \leq EvalOrd\text{-}Etape_k \leq 0.5$ si $-1 \leq PrecDiff \leq 0$
  $0.5 \leq EvalOrd\text{-}Etape_k \leq 1$ si $0 \leq PrecDiff \leq 1$

  Cette mesure nous donnera la performance de la requête q_u pour le système utilisé. Selon cette valeur on pourra en déduire le niveau de la compétence technique de l'utilisateur.

On peut évaluer $Eval\text{-}Etape_k$ en utilisant la formule [4.3] ou la formule [4.4].

L'exemple suivant illustre la différence entre ces deux mesures $EvalPrec\text{-}Etape_k$ et $EvalOrd\text{-}Etape_k$:

Supposons que l'outil de recherche a rendu 25 documents ($ND_{20} = Min(20, 25) = 20$) et que l'utilisateur a jugé les 20 premiers documents selon l'échelle (mauvais 0, moyen 0.3, bien 0.6, très bien 1) ainsi :

$$(\underbrace{0.6, 0.6, 1, 0.3, 0, 0.6, 0.3, 0.3, 0.3, 0.3}_{JugeDoc_1,\ldots, JugeDoc_{10}}, \underbrace{1, 1, 1, 0, 1, 1, 1, 1, 0.6, 0}_{JugeDoc_{11},\ldots, JugeDoc_{20}})$$

Selon la formule [4.3], l'évaluation de la requête par rapport au sujet de recherche est :

$$EvalPrec\text{-}Etape_k = \frac{1}{20}(0.6 + 0.6 + 1 + 0.3 + 0 + 0.6 + 0.3 + 0.3 + 0.3 + 0.3 + 1 + 1 + 1 + 0 + 1 + 1 + 1 + 1 + 0.6 + 0) = 0.6$$

Selon la formule [4.4], l'évaluation de la requête par rapport à la pertinence système est :

$$PrecDiff(1, 20) = [\underbrace{\frac{1}{10}(0.6 + 0.6 + 1 + 0.3 + 0 + 0.6 + 0.3 + 0.3 + 0.3 + 0.3)}_{0.4}] - [\underbrace{\frac{1}{10}(1 + 1 + 1 + 0 + 1 + 1 + 1 + 1 + 0.6 + 0)}_{0.8}]$$
$$= 0.4 - 0.8 = -0.4$$

$EvalOrd\text{-}Etape_k = ½ [1 + PrecDiff(1, ND_{20})] = ½ (1 - 0.4) = 0.3$

Nous remarquons dans cet exemple que :

- la requête est bonne par rapport au sujet de recherche parce qu'elle a retrouvé pas mal des documents pertinents ($EvalPrec\text{-}Etape_k = 0.6$).
- et pourtant elle est mauvaise par rapport à la pertinence système parce que les documents pertinents se sont mal positionnés ($EvalOrd\text{-}Etape_k = 0.3$).
- bien que le nombre des documents pertinents (9) dans les dix premiers documents soit plus grand que le nombre des documents pertinents (8) parmi les dix suivants, la mesure estime que les documents sont mal ordonnés parce que les documents les plus pertinents se trouvent dans la deuxième partie. C'est-à-dire que les huit documents qui se trouvent dans la deuxième partie ont obtenu des jugements de pertinence (0.8) plus grands que ceux obtenus pour les neuf documents dans la première partie (0.4).

Supposons que la plupart des requêtes de l'utilisateur soient de ce type c'est-à-dire qu'elles sont plutôt bonnes par rapport au sujet et mauvaises par rapport au système. Alors, nous déduisons que l'utilisateur est compétent dans le domaine tandis qu'il manque de compétence technique, il a donc besoin d'être aidé dans ce sens.

Ici aussi, l'évaluation de l'étape $Eval\text{-}Etape_k$, pour les deux mesures confondues, peut être une évaluation *globale* ou *partielle* selon que l'on prend en compte tous les critères de jugement ou seulement une partie, c'est-à-dire à l'instant du calcul on prend en compte l'évaluation *globale*, on utilise alors la formule [4.1] ou l'évaluation *partielle*, on utilise la formule [4.2].

4.3.2.2.2. Évaluation d'une Session Individuelle

Les formules précédentes ([4.3] et [4.4]) permettent d'évaluer la session individuelle *Eval-RII* de deux façons : par rapport au sujet de recherche *EvalPrec-RII* et par rapport à la pertinence système *EvalOrd-RII*

$$EvalPrec\text{-}RII = \frac{1}{NE}\sum_{k=1}^{NE} EvalPrec\text{-}Etape_k \quad [4.5]$$

où *NE* est le nombre d'étapes effectuées dans la session et *EvalPrec-RII* est l'évaluation de la session individuelle par rapport au sujet de recherche. On peut également définir l'évaluation de la session *Eval-RII* par la formule suivante :

$$EvalOrd\text{-}RII = \frac{1}{NE}\sum_{k=1}^{NE} EvalOrd\text{-}Etape_k \quad [4.6]$$

où *EvalOrd-RII* est l'évaluation de la session individuelle par rapport à la pertinence système.

Les deux valeurs *EvalPrec-RII* et *EvalOrd-RII* sont des valeurs dans l'intervalle [0, 1].

De la même manière, on parle aussi d'une évaluation *globale* et *partielle* de la session individuelle selon les critères de jugements pris en compte.

4.4. Stratégie de Soutien selon la Typologie de l'Utilisateur

Dans le paragraphe 4.2.4 nous avons mentionné que la valeur de préférence peut être déterminée par :

- l'*utilisateur* : chaque utilisateur u_i associe à chaque utilisateur u_j une valeur de préférence $Pref_{ui}(u_j)$.
- le *système* : qui calcule la valeur de préférence de chaque utilisateur. Nous avons expliqué dans le paragraphe 4.3.2.2.2 comment calculer cette préférence qui n'est que l'estimation de système $Eval\text{-}RII_{uj}$ de l'efficacité de la session individuelle de l'utilisateur u_j.
- l'*utilisateur* et le *système* : où le système assiste l'utilisateur u_i pour déterminer la valeur de préférence pour l'utilisateur u_j. Ainsi la valeur finale est la moyenne des deux valeurs données par l'utilisateur $Pref_{ui}(u_j)$ et par le système $Eval\text{-}RII_{uj}$:

  $Pref(u_j) = ½ (Pref_{ui}(u_j) + Eval\text{-}RII_{uj})$

Quand le système intervient dans la détermination de la valeur de préférence, c'est-à-dire dans les deux derniers cas, le système doit choisir les utilisateurs préférés selon leurs compétences, puis déterminer leur valeur $Eval\text{-}RII_{uj}$ il a alors deux possibilités :

1. utiliser la mesure de compétence de l'utilisateur dans le domaine alors :

   $Eval\text{-}RII_{uj} = EvalPrec\text{-}RII_{uj}$

2. utiliser la mesure de compétence de l'utilisateur dans le système de recherche donc :

   $Eval\text{-}RII_{uj} = EvalOrd\text{-}RII_{uj}$

Nous appelons une stratégie de soutien selon la typologie de l'utilisateur le fait de déterminer quand et dans quel contexte choisir les utilisateurs u_j qui peuvent apporter de l'aide ainsi leur valeur $Eval\text{-}RII_{uj}$. En effet, cela est basé sur le contexte de l'utilisateur u_i que le système veut soutenir, ce contexte ici correspond aux niveaux de ses compétences.

Afin de définir une stratégie de soutien selon la typologie de l'utilisateur, nous voulons tout d'abord concrétiser cette typologie par rapport à notre mesure de l'efficacité de l'utilisateur (paragraphe 4.4.1), puis nous présentons dans le paragraphe 4.4.2 le contexte de l'utilisateur u_i à soutenir et le choix des utilisateurs préférés u_j. Le paragraphe 4.4.3 présente la détermination de la valeur de préférence parmi les deux possibilités.

4.4.1. *Définition de la Typologie de l'Utilisateur*

Nous avons proposé, d'une part, une typologie de l'utilisateur selon ses compétences dans le domaine et sur le système, d'autre part, une mesure de l'efficacité de l'utilisateur par rapport aux documents pertinents obtenus et par rapport à leur ordonnancement.

Nous précision que l'évaluation de l'utilisateur par rapport à la pertinence de son résultat *EvalPrec-RII* est une mesure de sa compétence dans le domaine de recherche. Cette évaluation mesure la capacité de l'utilisateur à exprimer son besoin d'information. Tandis que l'évaluation de l'utilisateur par rapport à l'ordonnancement des documents dans le résultat *EvalOrd-RII* est une mesure de sa compétence dans le système de recherche.

Nous prenons plusieurs seuils ($\alpha_{Ord\text{-}d}$, $\alpha_{Prec\text{-}d}$, $\alpha_{Ord\text{-}m}$, $\alpha_{Prec\text{-}m}$) afin de déterminer les différents niveaux de compétence *débutant*, *moyen* et *expert* pour un utilisateur donné u_j comme il est illustré dans le Tableau 4.7.

Compétence dans le Domaine		Compétence dans le Système	
Débutant	$0 \leq EvalPrec\text{-}RII_{uj} < \alpha_{Prec\text{-}d}$	**Débutant**	$0 \leq EvalOrdc\text{-}RII_{uj} < \alpha_{Ord\text{-}d}$
Moyen	$\alpha_{Prec\text{-}d} \leq EvalPrec\text{-}RII_{uj} < \alpha_{Prec\text{-}m}$	**Moyen**	$\alpha_{Ord\text{-}d} \leq EvalOrd\text{-}RII_{uj} < \alpha_{Ord\text{-}m}$
Expert	$\alpha_{Prec\text{-}m} \leq EvalPrec\text{-}RII_{uj} \leq 1$	**Expert**	$\alpha_{Ord\text{-}m} \leq EvalOrd\text{-}RII_{uj} \leq 1$

Tableau 4.7. *Niveaux de compétences de l'utilisateur selon les évaluations de sa session.*

Cette typologie va nous servir pour choisir le(s) utilisateur(s) du groupe le(s) plus apte(s) à apporter de l'aide grâce à ses (leurs) résultats.

4.4.2. *Contexte de l'Utilisateur et Choix des Utilisateurs Préférés*

Nous faisons l'hypothèse que les utilisateurs, qu'il faudra choisir sont ceux qui ont des qualités complémentaires en terme d'expertise sur le domaine et sur le système.

Nous définissons les différents contextes possibles de l'utilisateur u_i, en terme de la relation entre sa compétence dans le domaine $EvalPrec\text{-}RII_{ui}$ et sa compétence dans le système $EvalOrd\text{-}RII_{ui}$. Soit u_i l'utilisateur qui a besoin d'aide et u_j les utilisateurs desquels u_i va obtenir de l'aide, il y donc trois possibilités :

Contexte.1. $EvalPrec\text{-}RII_{ui} < EvalOrd\text{-}RII_{ui}$: l'utilisateur u_i à soutenir a un niveau de compétence sur le domaine moins élevé que sur le système, alors u_i a besoin de l'aide des autres utilisateurs u_j qui sont experts dans le domaine.

Contexte.2. $EvalPrec\text{-}RII_{ui} > EvalOrd\text{-}RII_{ui}$: l'utilisateur u_i ici contrairement aux contextes précédents a un niveau de compétence dans le domaine plus élevé que dans le

système, alors u_i a besoin de l'aide des autres utilisateurs u_j qui sont experts dans le système.

Contexte.3. *EvalPrec-RII*$_{ui}$ = *EvalOrd-RII*$_{ui}$: l'utilisateur u_i a un niveau de compétence dans le domaine égal à celui dans le système. Dans le cas où l'utilisateur u_i serait *débutant* ou *moyen* dans les deux compétences, nous privilégions d'aider l'utilisateur dans le domaine, donc on l'aide à partir des autres utilisateurs u_j qui sont experts dans le domaine. Tandis que dans le cas où l'utilisateur serait *expert* dans les deux compétences, c'est-à-dire il n'a pas un besoin réel d'aide, alors le système peut l'aider en considérant les utilisateurs qui sont aussi experts que lui.

4.4.3. *Choix de la Valeur de Préférence*

Selon le contexte de l'utilisateur u_i la mesure *Eval-RII*$_{uj}$ choisie pour sélectionner les meilleurs u_j prend les valeurs suivantes :

Contexte.1. le système considère *Eval-RII*$_{uj}$ = *EvalPrec-RII*$_{uj}$.

Contexte.2. le système prend *Eval-RII*$_{uj}$ = *EvalOrd-RII*$_{uj}$.

Contexte.3. le système détermine *Eval-RII*$_{uj}$ = *EvalPrec-RII*$_{uj}$.

La stratégie de soutien est résumée dans Tableau 4.8, où les trois premières colonnes du tableau représentent les différents contextes possibles de l'utilisateur u_i, selon ses compétences, tandis que les trois premières lignes de tableaux présentent ceux de l'utilisateur u_j (à qui on va distribuer la valeur de préférence ou de qui on va obtenir de l'aide). Les autres cases du tableau déterminent le choix de la valeur *Eval-RII*$_{uj}$: *EvalPrec-RII*$_{uj}$ symbolisé par P ou *EvalOrd-RII*$_{uj}$ symbolisé par O tandis que l'ignorance de l'utilisateur est symbolisée par -.

			les utilisateurs u_j de qui le soutien est obtenu									
			débutant			**moyen**			**Expert**			**compétence du domaine**
			débutant	**moyen**	**expert**	**débutant**	**moyen**	**expert**	**débutant**	**moyen**	**expert**	**compétence technique**
1	**débutant**	**expert**	-	-	-	-	-	-	**P**	**P**	**P**	
1	**débutant**	**moyen**	-	-	-	-	-	-	**P**	**P**	**P**	
3	**débutant**	**débutant**	-	-	-	-	-	-	**P**	**P**	**P**	
1	**moyen**	**expert**	-	-		-	-	-	**P**	**P**	**P**	
3	**moyen**	**moyen**	-	-	-	-	-	-	**P**	**P**	**P**	
2	**moyen**	**débutant**	-	-	**O**	-	-	**O**	-	-	**O**	
3	**expert**	**expert**	-	-	-	-	-	-	-	-	**P**	
2	**expert**	**moyen**	-	-	**O**	-	-	**O**	-	-	**O**	
2	**expert**	**débutant**	-	-	**O**	-	-	**O**	-	-	**O**	
contexte	**compétence du domaine**	**compétence technique**										
u_i à soutenir/ à aider												

Tableau 4.8. *Stratégie de soutien selon la typologie de l'utilisateur.*

Dans le cas où la valeur de préférence est déterminée par l'*utilisateur* et le *système* c'est-à-dire *Pref*(u_j) = ½ (*Pref*$_{ui}$(u_j) + *Eval-RII*$_{uj}$) et que l'utilisateur u_j est négligé par le système (parce que *EvalPrec-RII*$_{uj}$ < $\alpha_{Prec\text{-}m}$ ou *EvalOrd-RII*$_{uj}$ < $\alpha_{Ord\text{-}m}$) alors la valeur de préférence de u_j est celle donnée par l'utilisateur u_i :*Pref*(u_j) = *Pref*$_{ui}$(u_j).

4.5. Recherche d'Information Collaborative

Un certain nombre d'études ont été faites sur les manières dont peuvent collaborer un groupe d'individus dans une tâche de recherche d'information. Nous en présentons deux. La première étude expérimentale a pour but d'analyser le comportement de groupes d'utilisateurs en situation de recherche de références bibliographiques au sein de la bibliothèque de l'Université de Lancaster (paragraphe 4.5.1). La deuxième étude caractérise quatre types de navigation collaborative synchrone pouvant aider le groupe (paragraphe 4.5.2).

Dans l'objectif de soutenir un groupe d'utilisateurs, nous avons été amené à analyser ce qu'était la collaboration dans une tâche de recherche ce qui implique une définition de la typologie du groupe selon leur degré de collaboration et des stratégies de soutien différentes (paragraphe 4.5.3). Le paragraphe 4.5.4 représente un environnement collaboratif de soutien.

4.5.1. *Observation de la Recherche d'Information Collaborative*

Les auteurs dans l'étude [Twidale 1997] ont observé la collaboration entre des utilisateurs dans la Bibliothèque d'Université Lancaster. Ils ont donné quelques scénarios représentatifs de collaboration entre ces utilisateurs :

- *Recherche Jointe* « *Joint Search* » : un groupe d'étudiants (de 2 à 4) travaille autour d'un seul terminal, discute leurs idées et planifie leurs actions. Ils sont impliqués ensemble à résoudre une tâche.
- *Recherche Coordonnée* « *Coordinated Search* » : un groupe travaille sur plusieurs terminaux adjacents, les membres du groupe discutent de ce qu'ils font, comparent leurs résultats, et il semble, que parfois ils rivalisent pour trouver l'information. Il arrive qu'un étudiant se penche sur un autre terminal, ou même de temps en temps, il arrive que tout le groupe soit autour d'un seul terminal, comme dans le cas précédent.
- *Question Opportune* « *Free Query* » : les étudiants travaillent sur des terminaux séparés, mais adjacents De temps en temps, ils demandent à leur voisin de l'aide (par exemple : « excuse-moi, comment je peux faire.... ? »).
- *Question Ciblée* « *Directed Query* » : les étudiants travaillent sur des terminaux séparés, et peuvent surveiller les activités des autres. De temps en temps, cela conduit à une question de la forme « Comment as-tu fait cela ? ».
- *Rencontre Informelle* « *Chance Contact* » : cette situation se produit quand les étudiants se croisent autour d'une ressource commune comme une imprimante ou une photocopieuse. Par exemple une personne imprime des résultats de recherche, il trouve une autre liste imprimée, alors il demande qui en est le propriétaire, afin d'en discuter avec lui.

4.5.2. *Définition d'une Interface*

A partir de plusieurs études dont la précédente, Laurillau [Laurillau 1999] a défini quatre types de navigation collaborative synchrone comme suit :

- *Navigation Coopérative* (correspond à la *Recherche Jointe*) : un responsable de groupe oriente la recherche. Il lui incombe la responsabilité de définir les objectifs et les stratégies à suivre, de manière autoritaire ou concertée, et de répartir, si cela est nécessaire, les sous-espaces à explorer. Il est le seul à décider si un nouveau venu peut en devenir membre. Il a de plus la possibilité de déplacer (« téléporter ») l'ensemble du groupe dans un endroit qu'il peut juger intéressant.
- *Navigation Coordonnée* (correspond à la *Recherche Coordonnée*) : les membres du groupe travaillent séparément, dans un espace restreint ou non, sur un sujet commun, mais sans forcément avoir exactement les mêmes buts. Tous ont les mêmes droits, et n'importe qui peut décider si un nouveau venu peut devenir ou non un membre du groupe.
- *Navigation Opportuniste* (correspond à la *Question Opportune*, la *Question Ciblée* et la *Rencontre Informelle*) : la rencontre est informelle, et chacun navigue indépendamment les uns des autres, mais dispose des résultats collectés par les autres membres du groupe. N'importe qui peut devenir un membre et, à n'importe quel moment, il est possible de donner le contrôle de la navigation à un autre.
- *Visite Guidée* : inspirée de la visite guidée d'un musée ou d'un site, les utilisateurs membres du groupe cèdent le contrôle de l'interaction à un guide qui les emmène découvrir l'espace d'information. A tout moment un utilisateur peut joindre ou quitter la visite guidée.

4.5.3. *Environnement de Recherche d'Information Collaborative*

Notre but étant de soutenir un groupe d'utilisateurs, nous procédons à une action préalable : nous définissons d'une part une typologie pour la recherche collaborative (paragraphe 4.5.3.1), et d'autre part un modèle de soutien (paragraphe 4.5.3.2). Une fois ces éléments définis, nous pouvons présenter le modèle d'une session collaborative (paragraphe 4.5.3.3).

Dans la suite, nous proposons un procédé d'exploitation flexible du modèle de soutien sous forme d'utilisation conjointe avec la typologie, afin d'obtenir des stratégies de soutien selon le type de collaboration. Cette première proposition est une stratégie restrictive. Alors qu'un environnement de recherche d'information plus exhaustif est proposé dans le paragraphe 4.5.4.

4.5.3.1. Typologie de Groupe

Nous avons repris ces études et les avons adaptées pour la recherche d'information, le Tableau 4.9 est une synthèse de ces trois approches.

Nous distinguons trois types de recherche : la *Recherche Joint*e, la *Recherche Coordonnée*, et la *Recherche Relâchée*. L'hypothèse sous-jacente à ce travail est que les membres du groupe ont tous le *même sujet de recherche.* La distinction entre ces types de recherche se fait selon :

- l'objectif de recherche de chaque membre.
- la relation sociale existant entre les membres.

- la planification de recherche de chacun.

[Twidale 1997]	[Laurillau 1999]	Types Proposés
Recherche Jointe	Navigation Coopérative	Recherche Jointe
Recherche Coordonnée	Navigation Coordonnée	Recherche Coordonnée
Question Opportune Question Ciblée Rencontre Informelle	Navigation Opportuniste	Recherche Relâchée
-	Visite Guidée	-

Tableau 4.9. *Synthèse de types de collaboration.*

Nous caractérisons les types de recherche de la façon suivante (voir le Tableau 4.10) :

1. *Recherche Jointe* : les utilisateurs ont le même objectif, ils se connaissent bien et définissent ensemble un plan de recherche. Ils peuvent alors travailler en parallèle sur des tâches similaires, ou bien de façon complémentaire. Cela peut être le cas pour plusieurs membres de la même équipe de recherche, qui construisent une bibliographie afin d'écrire un article ensemble.
2. *Recherche Coordonnée* : les membres du groupe ont des objectifs connexes. Ces utilisateurs se connaissent suffisamment pour se coordonner, chacun construit son plan de recherche, cependant certaines étapes peuvent être similaires. Cette collaboration peut se produire pour des chercheurs qui se sont déjà rencontrés au cours d'une conférence, ils veulent écrire un livre dont chacun est l'auteur d'un chapitre et cherchent ensemble des bibliographies sur le thème du livre.
3. *Recherche Relâchée* : il s'agit d'un groupe d'utilisateurs ayant des objectifs différents et ne se connaissent pas, chacun d'entre eux établit son propre plan de recherche. Supposons que des utilisateurs s'intéressent au sujet de recherche « *le cinéma dans les pays asiatiques* ». Selon ses centres d'intérêts et sa compétence, chaque utilisateur formulera différemment ce sujet. On peut imaginer que quelqu'un passionné par le Japon cherchera « les films japonais », qu'un autre plus spécialiste cherchera les films du metteur en scène « Romain Slocombe », qu'un autre cherchera plutôt un genre de films (policier, sentimental,…).

	Recherche Relâchée	**Recherche Coordonnée**	**Recherche Jointe**
planification de recherche	individuelle	individuelle, certaines étapes peuvent être les mêmes	collaborative
objectifs de recherche	différents	connexes	identiques
relation sociale entre les membres	pas de connaissance	connaissance suffisante	bonne connaissance

Tableau 4.10. *Synthèse de types de collaboration.*

4.5.3.2. Stratégie de Soutien selon la Typologie de Groupe

Nous avons défini d'une part le soutien en terme de tâches, de dimensions, et de données de jugement et de préférence (paragraphes 4.2), et d'autre part une typologie du groupe selon le degré de collaboration entre les membres (paragraphe 4.5.3.1).

Le Tableau 4.11 résume les différents paramètres permettant de définir le soutien ainsi que l'ensemble des valeurs possibles.

Nous appelons une stratégie de soutien selon la typologie de groupe le fait de déterminer la ou les valeur(s) de chacun des paramètres selon le type de groupe.

Ainsi, selon ce type, les utilisateurs n'auront pas les mêmes possibilités. Nous analysons le choix des valeurs pour ces paramètres ainsi :

Dimensions	temps absolu	• synchrone • asynchrone
	temps dans la session	• courant • historique
	groupe	• collaboratif • individuel
Tâches	formulation de requête	• présentation des requêtes • construction de requête
	visualisation de résultat	• présentation des résultats
	bouclage de pertinence	• application d'une méthode de feedback sur des requêtes et des résultats jugés
Jugement	critères de jugement	• un seul • plusieurs
	poids d'importance des critères	• même poids • poids différents
	échelle de valeur	• binaire • réelle
Préférence	responsabilité du choix des valeurs de l'échelle	• groupe • système
	responsabilité de la distribution de la préférence	• utilisateur • système • utilisateur & système

Tableau 4.11. *Résumé du paramètre stratégique.*

- *Dimensions* : l'utilisateur peut se retrouver dans différentes situations de recherche, ces situations sont définies à l'aide de trois dimensions : le temps absolu, le temps dans la session et le groupe. Dans cette étude, nous orientons vers les situations de recherche collaborative synchrone courante ou historique (correspondant respectivement à considérer la dernière étape de la session ou toute la session de recherche). Donc, pour les différents types de groupe, nous aidons les utilisateurs dans ces deux situations.

- *Tâches* : nous pouvons soutenir l'utilisateur dans les tâches suivantes : formulation de requête, visualisation de résultats et bouclage de pertinence. Pour les recherches relâchée et coordonnée, chacun des utilisateurs a son plan de recherche, dont les objectifs diffèrent. Le soutien intervient alors pour la formulation de requête et la visualisation du résultat. Pour la recherche jointe, comme les utilisateurs ont le même plan de la recherche et le même objectif, on leur donne en plus la possibilité de reformuler des requêtes via le bouclage de pertinence.
- *Jugement* : il y a plusieurs paramètres concernant le jugement : les critères de jugement (un seul ou plusieurs), leurs poids d'importance (même poids ou poids différents), leur échelle de valeurs (binaire ou réelle) et qui a la responsabilité de déterminer tous cela (groupe ou système). Pour la recherche relâchée, le jugement se fera selon un seul critère et de façon binaire. En effet, dans ce cas, les utilisateurs ont des objectifs différents et on veut les assister sans pour autant leur imposer une grande charge cognitive. Pour la recherche coordonnée, les objectifs des utilisateurs sont connexes, ce qui implique un jugement plus détaillé que dans le cas précédent, le système définit les critères de jugement (la pertinence de contenu, la date du document, et l'auteur). Ces critères auront des valeurs multiples et une importance égale. Enfin pour la recherche jointe les utilisateurs vont déterminer ensemble toutes les valeurs (critères, poids, échelle, etc.) de façon adaptée à leur objectif. Les choix des valeurs des paramètres concernant les données de jugement sont résumés dans le Tableau 4.12.

	Recherche Relâchée	**Recherche Coordonnée**	**Recherche Jointe**
critères de jugement	un seul critère	plusieurs critères	plusieurs critères
poids d'importance des critères	-	égal	différent
échelle de valeur	binaire	réelle	réelle
choix des critères, des poids et de l'échelle	système	système	groupe

Tableau 4.12. *Choix des paramètres de données jugement face à chaque type de recherche.*

- *Préférence* : il y a aussi plusieurs paramètres concernant la préférence : l'échelle de valeurs de la préférence (numérique), qui a la responsabilité de déterminer cette échelle (groupe ou système) et qui a la responsabilité de déterminer la préférence (utilisateur ou système ou utilisateur & système). Pour les trois types de recherche, la préférence prend des valeurs multiples. Le système choisit l'échelle de valeurs pour la recherche relâchée et la recherche coordonnée, et le groupe pour la recherche jointe. Le système estime la valeur de préférence pour la recherche relâchée à partir de leurs résultats évalués puisque les utilisateurs ne se connaissent pas. Pour la recherche jointe, les utilisateurs se connaissent bien, ils peuvent définir leurs préférences. La recherche coordonnée se situant entre les deux précédentes, la préférence sera déterminée en collaboration par l'utilisateur et par le système. Les choix des valeurs des paramètres concernant les données de préférence sont résumés dans le Tableau 4.13.

Le type du groupe et la stratégie du soutien adaptée à ce type font partie de la définition d'une session collaborative comme nous expliquons dans la suite.

	Recherche Relâchée	Recherche Coordonnée	Recherche Jointe
choix de l'échelle	système	système	groupe
évaluation de la préférence	système	utilisateur et système	utilisateur

Tableau 4.13. *Choix des paramètres de donnée préférence face à chaque type de recherche.*

4.5.3.3. Session Collaborative

Nous décrivons la session de recherche d'information collaborative *Session-RIC* de façon ensembliste, où l'écriture (..)* désigne un groupe multivalué d'éléments ainsi :

$$\underbrace{\textit{Session-RIC}\ (\textit{TypeRecherche}, \textit{StratégieSoutien}, \textit{Sujet}, \textit{DateHeureDébut-RIC}}_{\textit{Statique}}, \underbrace{\textit{NbMax, NbActuel}, \textit{Durée-RIC}, \textit{Sesssion-RII}\ *)}_{\textit{Dynamique}}$$

Le type de l'objet session collaborative est présenté sur la Figure 4.6.

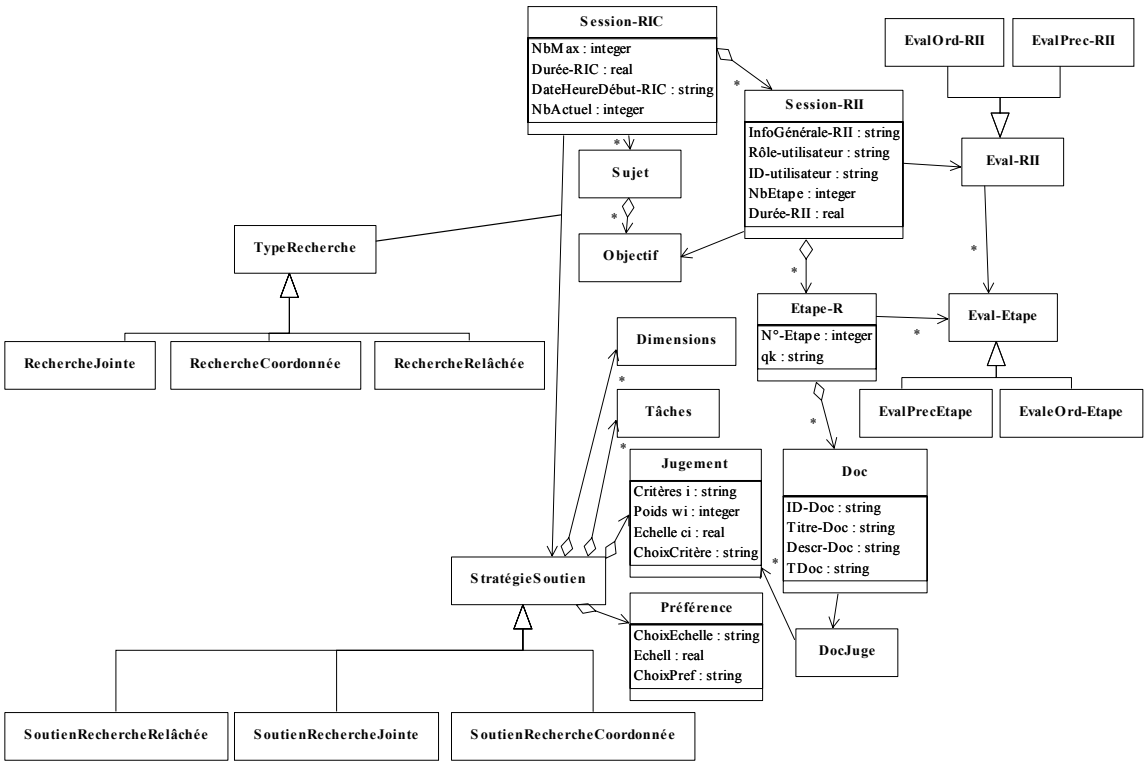

Figure 4.6. *Modèles décrits avec les concepts du modèle objet.*

Une session collaborative va de sa création explicite c'est-à-dire de la création de la première session individuelle dans cette session collaborative jusqu'à la fin de la dernière session individuelle.

Comme on peut le voir, le modèle de session individuelle *Session-RII* à son tour, comporte deux types d'informations statique (paragraphe 4.5.3.3.1) et dynamique (paragraphe 4.5.3.3.2).

4.5.3.3.1. Eléments Statiques

Les éléments statiques au cours de déroulement de la session collaborative sont :

- Des informations concernant le soutien et l'évaluation de ce soutien :
  - *TypeRecherche* : c'est le type de recherche choisi dès le départ par le groupe d'utilisateurs (voir paragraphe 4.5.3.1), cela peut être une recherche relâchée, coordonnée ou jointe, désignée respectivement par *RechercheRelâchée*, *RechercheCoordonnée*, ou *RechercheJointe*.
  - *StratégieSoutien* : la stratégie de soutien adopté pour le type de recherche choisi (voir paragraphe 4.5.3.2) peut être *SoutienRechercheRelâchée*, *SoutienRechercheCoordonnée*, ou *SoutienRechercheJointe*.

  $$TypeRecherche \xrightarrow[StratégieSoutien : (ps_1 = v_1 \ldots ps_e = v_e)]{Session\text{-}RIC}$$

  Où ps_i représente le *i*ème paramètre stratégique et v_i est la valeur du paramètre ps_i de la session de recherche collaborative. Ce paramètre stratégique ps_i concerne soit les *Tâches de Soutien*, les *Dimensions de Soutien,* le *Jugement*, et la *Préférence.*

- Des informations thématiques générales comme :
  - *Sujet* : il représente des informations sur le sujet de recherche effectuée durant la session. Il consiste en un *Titre* du sujet qui identifie la session collaborative et peut être éventuellement complétée par une *Description* du sujet.
  - *DateHeureDébut-RIC* concerne le temps, la date, et l'heure du début de la session.

4.5.3.3.2. Eléments Dynamiques

Les éléments progressifs relatifs au déroulement de la session collaborative sont :

- *NbMax* est le nombre maximal d'utilisateurs qui ont participé à la session collaborative. Cela permet de prendre en compte un utilisateur qui rejoint le groupe en cours de session ou qui quitte le groupe avant la fin.
- *NbActuel* est le nombre des utilisateurs qui effectuent des sessions individuelles à un moment donné au cours de la session collaborative.
- *Durée-RIC* est la durée de la session de sa création jusqu'à la fin de la dernière session individuelle dans cette session collaborative. Cette information sur la durée peut être un paramètre pour l'évaluation du système.
- *Sesssion-RII** représente les différentes sessions de recherche individuelles effectuées par les utilisateurs.

Nous définissons une session de recherche collaborative *Session-RIC* comme une agrégation de *NU* sessions individuelles *Session-RII*, où *NU* est le nombre des utilisateurs

dans le groupe. Chaque session individuelle est effectuée par un des membres du groupe, le modèle de session individuelle a été précédemment expliqué dans le paragraphe 4.1.1.

4.5.4. *Environnement de Recherche d'Information Collaborative Complet*

Si nous disposons de l'interface de navigation collaborative qui offre toutes les fonctionnalités nécessaires pour créer et organiser un groupe ; le système de soutien à la recherche d'information que nous avons défini peut aller dans la direction de l'étude de [Twidale 1997], ainsi les utilisateurs peuvent interagir selon les différents scénarios d'interactions :

- Recherche Jointe : le groupe est composé d'utilisateurs qui se connaissent bien, ils ont le même objectif de recherche. L'interface peut leur fournir les fonctionnalités nécessaires pour :
  - créer un groupe : l'interface fournit la fonctionnalité nécessaire pour communiquer et dialoguer, donc les utilisateurs dialoguent à propos de l'objectif et ils discutent d'un plan de recherche ainsi ils précisent des sous-objectifs de leur objectif principal, et la conséquence de ces sous-objectifs, puis ils peuvent choisir de travailler :
    1. de façon parallèle : ils travaillent simultanément sur chacun de ces sous-objectifs, ainsi ils planifient le temps consacré pour chaque sous-objectif, afin de synchroniser leur avancement d'un sous-objectif à l'autre ;
    2. de façon complémentaire : ils partagent/distribuent les sous-objectifs entre eux, ainsi chacun a la responsabilité d'un ou plusieurs sous-objectif(s), ils définissent le temps de finir chaque sous-objectif.
  - créer une session : une fois le groupe créé, les utilisateurs se mettent d'accord sur les paramètres de façon adaptée à leur objectif c'est-à-dire ils définissent les critères de jugement, leur échelle de valeurs, leur poids d'importance, et l'échelle de valeurs de préférence. Ils définissent l'heure de début et la durée de la recherche, ils définissent aussi qui sera chargé de démarrer et créer la session collaborative et de fournir tous ces paramètres.
  - déroulement de la recherche : chacun travaille sur son ou ses sous objectif(s). De temps en temps, les utilisateurs se rassemblent virtuellement afin de discuter de leur plan de recherche et probablement le changer et l'adapter selon les références trouvées et selon leurs nouvelles connaissances acquises pendant la recherche. Par exemple, ils visualisent une présentation des requêtes ou ils parcourent une synthèse des résultats. Ils peuvent ensemble discuter à propos des tels : est-ce que les résultats obtenus sont suffisants c'est-à-dire est-ce que le but est atteint ? pourquoi telle ou telle référence à été jugée ainsi ? est-ce que le groupe dérive de son objectif principal ? dans le cas de dérivation est-ce que cela va dans le bon sens ou est-ce que le groupe s'éloigne de la problématique de recherche ? etc.… puis chacun reprend sa recherche…
- Recherche Coordonnée : le groupe consiste en utilisateurs qui se connaissent assez, chacun a un objectif de recherche différent mais connexe aux autres, ils se mettent d'accord à propos d'un sujet général de session de recherche, des autres utilisateurs peuvent la joindre et chacun recherche tout seul. A partir d'une présentation des résultats ou des requêtes des autres un utilisateur peut se rende compte qu'il a le même profil que

lui ou plutôt qu'il a un ou plusieurs sous-objectif(s) identique(s). Ainsi ils peuvent en discuter.

- Recherche Relâchée : les utilisateurs ne se connaissent pas, un utilisateur démarre une session de recherche à propos d'un sujet de recherche et il définit son propre objectif. D'autres, qui s'intéressent au même sujet, se joignent à cette session, et chacun définit un objectif différent et son plan de recherche. Des interactions peuvent se passer ainsi :
  - *Question Opportune* : l'utilisateur est désorienté, il peut avoir un problème particulier, il se demande comment on fait cela ou comment on peut trouver telle ou telle information alors il va regarder les requêtes ou les résultats des autres, afin d'obtenir une réponse à sa question.
  - *Question Ciblée* : un utilisateur a des doutes, il peut avoir un problème mais qui n'est pas bien défini alors il va regarder la recherche des autres à ce moment là, il remarque quelque chose qui l'intéresse.
  - *Rencontre Informelle* : l'utilisateur n'a pas un problème particulier mais par curiosité va regarder la recherche des autres. Une requête ou un résultat évoque sa curiosité alors il va regarder comment on l'a obtenu…. Cela peut le conduire à discuter avec la personne.

L'environnement de soutien que nous décrivons dans le chapitre suivant permettra ces différentes interactions.

4.6. Conclusion

Les évaluations faites pour un utilisateur sont dynamiques et dépendantes de son travail lorsqu'il évolue dans sa propre session de recherche, et surtout des informations qu'il a obtenues du groupe.

En d'autres termes, le soutien qu'il demande, ne sera pas pris en compte toujours de la même manière, cela dépend de son contexte personnel, c'est-à-dire les requêtes qu'il a soumises, les résultats qu'il a évalués mais aussi des contextes de tous les membres du groupe.

Nous avons défini dans ce chapitre des mesures qui vont permettre d'identifier une typologie et le contexte de l'utilisateur, ainsi qu'une typologie de groupe, cela nous a permis de définir des stratégies selon ces deux typologies.

Cet environnement ne répond pas encore à notre objectif de départ, pour clarifier cela prenons l'exemple suivant où un utilisateur a besoin d'aide, il demande au système de l'aider par une présentation de requêtes courantes, il obtient une liste de toutes les requêtes soumises dernièrement par chacun des utilisateurs, un autre utilisateur peut demander la même aide, et ainsi obtenir la même liste de requêtes.

Nous constatons que les soutiens pour les deux utilisateurs sont identiques, c'est-à-dire le soutien n'est pas adapté à chacun. Un autre constat est que l'utilisateur peut ne pas être intéressé par toutes les requêtes à propos d'un sujet de recherche, mais seulement être intéressé par certaines requêtes, selon un ou plusieurs aspects.

Ces constats nous conduisent à définir des critères de personnalisation, à travers lesquels l'utilisateur peut exprimer ses désirs et ses souhaits, et à adapter l'aide selon ces critères, l'étude de cela fait l'objet du chapitre suivant.

CHAPITRE.5. Soutien Personnalisé

Nous avons défini dans le chapitre précédent l'environnement de soutien avec ses tâches, ses dimensions et ses données nécessaires à son fonctionnement et les stratégies d'aide selon la typologie de l'utilisateur et la typologie du groupe.

Nous avons expliqué dans le paragraphe 4.2.1 que nous aidons l'utilisateur dans différentes tâches de recherche : formulation de requête, visualisation du résultat et reformulation de requête, la Figure 5.1 illustre cela.

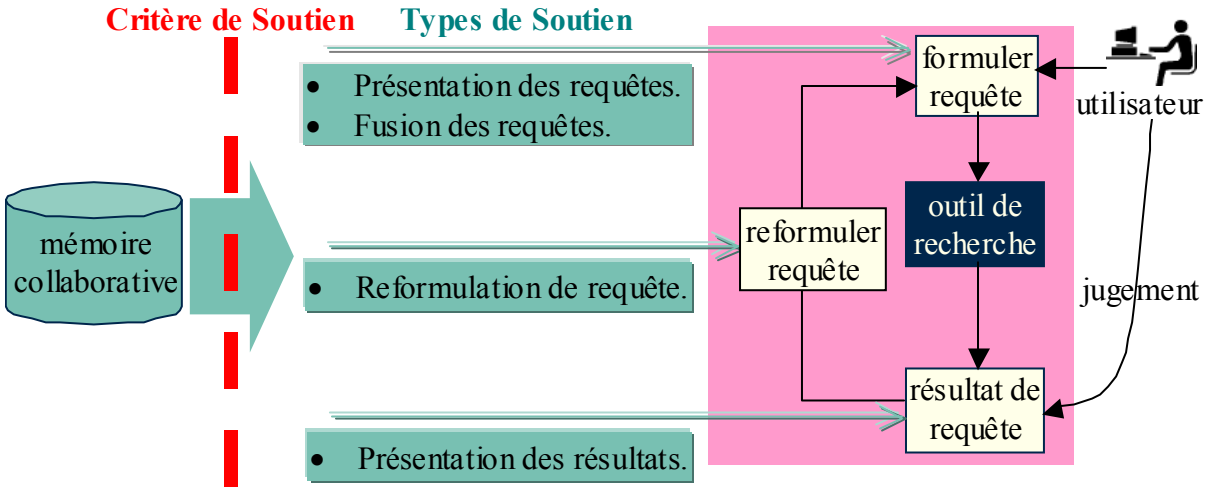

Figure 5.1. *Types, tâches, et critères de soutien.*

Nous disposons dans le mémoire collaborative des requêtes et des résultats jugés du groupe. A partir de ces données, l'aide est fournie à l'utilisateur selon un des types de soutien. Pour personnaliser et adapter cette aide aux besoins et aux souhaits de l'utilisateur, nous définissons un certain nombre de critères appelés *critères de soutien*. Ces critères présentés en forme de « filtre » sur la Figure 5.1 conditionnent, et trient, les requêtes et/ou les résultats que nous utilisons pour fournir l'aide.

Ce chapitre est organisé de la manière suivante : tout d'abord nous présentons les modèles des requêtes et des documents dans le paragraphe 5.1 que nous considérons afin de mieux formuler la personnalisation, puis, nous déterminons dans le paragraphe 5.2 les critères par rapport auxquels le soutien peut être personnalisé, nous définissons dans le paragraphe 5.3 la procédure d'extraction d'un sous-ensemble de requêtes et/ou de documents de la mémoire collaborative selon les critères de soutien, tandis que dans le paragraphe 5.4, nous appliquons des opérations sur les éléments extraits afin d'obtenir le soutien selon le type demandé. Enfin, le paragraphe 5.5 conclue ce chapitre.

5.1. Modèles de Recherche d'Information

Dans l'environnement du « World Wide Web », la notion de modèle de requête et de document n'est pas évidente. Il est donc plus difficile de traiter les requêtes et les documents selon tel ou tel modèle de recherche (vectoriel, booléen, probabiliste, etc.), simplement du fait que nous n'avons accès ni au modèle précis qu'implante le moteur de recherche, ni à l'indexation de sa base de données. Néanmoins, le modèle vectoriel est très utilisé expérimentalement, comme nous avons pu le remarquer dans les expérimentations sur la fusion (CHAPITRE.3), et le modèle booléen est actuellement le plus utilisé pour les moteurs de recherche du Web. Alors, nous présentons brièvement le modèle vectoriel (paragraphe 5.1.1) et le modèle booléen (paragraphe 5.1.2) ainsi que le modèle utilisé dans la suite de ce chapitre (paragraphe 5.1.3).

5.1.1. *Modèle Vectoriel*

Le modèle vectoriel (Vector Space Model [Salton 1971]) avec, en exemple, le système SMART représente les documents et les requêtes sous forme de vecteurs.

Tout document d'une collection peut être décrit par un ensemble de termes ou mots clés, un document Doc_i est représenté par un vecteur de dimension m :

$Doc_i = (w_{i1}, w_{i2}, \ldots, w_{im})$ pour $i = 1, 2, \ldots, n$.

Où w_{ij} est le poids (ou l'importance) du terme t_j dans le document Doc_i. n est le nombre de documents dans la collection, m est le nombre de termes dans les documents de la collection.

Une requête q_k comprend une phrase en langue naturelle ou une série de mots-clés jugés importants par l'utilisateur. Après avoir traité la phrase par l'ordinateur, la requête q_k est également représentée par un vecteur dans l'espace des termes.

$q_k = (w_{k1}, w_{k2}, \ldots, w_{km})$

Où w_{kj} est le poids de terme t_j dans la requête q_k.

L'opération de mise en correspondance peut alors être réalisée par une mesure de similarité entre vecteurs d'un espace. Une telle mesure, que nous noterons *sim*, peut fournir, pour chaque document Doc_i de la collection, un degré de similarité $sim(Doc_i, q_k)$ avec le vecteur requête. La valeur de $sim(Doc_i, q_k)$ devra être d'autant plus grande que le nombre et le poids des termes communs est important. Typiquement la mesure du cosinus de l'angle entre les deux vecteurs $cos(Doc_i, q_k)$ est une mesure de similarité en recherche d'information (Salton 1971 SMART). D'autres mesures de similarité ont été proposées, la mesure de Dice et le pseudo-cosinus en sont deux exemples.

Le modèle vectoriel réside dans sa simplicité conceptuelle et de mise en ouvre. Il permet de représenter de manière uniforme un vecteur document et une requête et intègre simplement des pondérations sur les mots-clés attachés à un document aussi bien qu'à une requête. Il offre aussi des moyens simples pour classer les résultats d'une recherche à travers une mesure de similarité en plaçant en tête les documents jugés les plus similaires.

5.1.2. *Modèle Booléen*

Le modèle booléen est basé sur l'algèbre de Boole, il y le modèle booléen simple qui s'appuie sur une indexation binaire ; un terme indexe ou non un document, et le modèle booléen étendu plus général, qui s'appuie sur une indexation pondérée.

Lors de la recherche, une requête booléenne représente les conditions logiques que doit respecter un document afin d'être retourné à l'usager. Alors, tout texte extrait par la machine répond strictement aux conditions de la requête. Ainsi la requête q est n'importe quelle combinaison booléenne des termes en utilisant les opérateurs ET, OU et NON. Normalement les opérateurs ont un ordre de priorité pour leur évaluation où le NON a la plus haute priorité, puis vient ET et OU. La formulation booléenne n'est pas toujours aisée à transcrire. Les documents sont indexés selon un ensemble de termes. Le degré de similarité requête-document peut se calculer selon la formule suivante :

$$P(Doc_i, q_k) = \sum_{k=1}^{q_i} w_{ik} \qquad \text{pour} \quad i = 1, 2, \ldots, n.$$

Dans laquelle w_{ik} indique le poids attribué au terme t_j pour le document Doc_i. Cette addition prend en compte tous les mots clé t_j, pour j = 1,2 … inclus dans la requête q_k. Plusieurs variantes de la fonction de correspondance entre le document et la requête ont déjà été proposées comme le recours à la logique flou, le modèle P-norme… etc.

5.1.3. *Modèles de Recherche Considérés*

Sur le Web on peut avoir une idée du modèle de la structure de la requête tandis que l'on ne peut pas connaître comment les documents sont indexés. Alors, nous prenons en compte le modèle vectoriel et le modèle booléen en ce qui concerne la requête. En ce qui concerne chaque document *Doc* est considéré comme un ensemble de termes *TDoc* apparus dans son titre. Le résultat R_{qj} retourné par le moteur de recherche comme réponse à q_j est présenté aussi étant un ensemble de documents ainsi :

R_{qj} = { *Doc* : *Doc* est retrouvé par q_j }

Puisque nous considérons le résultat de la requête comme un ensemble de documents, nous choisissons d'utiliser les opérations ensemblistes (intersection, union, différence…etc.) sur ces ensembles afin de définir entre ces ensembles des mesures de similarité, de dissimilarité, de fusion, etc. Nous avons donc choisi le modèle ensembliste.

Après cette présentation des différents modèles de requêtes souvent utilisés, et le modèle de document nous décrirons les critères de personnalisation.

5.2. Critères de Soutien

Nous introduisons, tout d'abord, les différents critères de soutien qui nous permettent d'analyser la mémoire collaborative, puis nous expliquons en détail dans les paragraphes qui suivent chacun de ces critères.

Nous définissons les critères de soutien suivants :

- *préférence c-pref* : ce premier critère prend en compte la valeur de préférence effectuée à chaque utilisateur du groupe. En l'absence de ce critère la mémoire collaborative est vue comme l'ensemble total des sessions individuelles des différents utilisateurs, tandis que le critère de préférence la divise en deux sous-ensembles : l'un contient les sessions individuelles des utilisateurs préférés et l'autre ceux des utilisateurs non préférés. C'est-à-dire qu'il sélectionne les utilisateurs dont on choisit les requêtes et/ou les résultats (paragraphe 5.2.1).
- *jugement c-juge* : le critère de jugement permet la sélection des éléments de la mémoire par rapport aux jugements des utilisateurs, et l'adaptation à la connaissance que l'on a de l'utilisateur dans un certain nombre de type d'aide, nous clarifierons cela dans le paragraphe 5.2.2.

Ces deux critères sont des critères numériques et ils sont sous la responsabilité du groupe, où le *jugement* est fourni par les utilisateurs, et la *préférence*, peut être fournie par l'utilisateur et/ou éventuellement par le système.

- *similarité de requêtes c-requête* : ce critère prend en compte la mesure de similarité d'une requête dans la mémoire avec une requête donnée de l'utilisateur (paragraphe 5.2.3). Cette similarité peut être : la similarité des requêtes elles-mêmes, ou la similarité entre leurs résultats respectifs.

D'autres critères peuvent être envisagés que nous appelons des *critères supplémentaires* :

- *contexte matériel c-contexte* : ce critère prend en compte le contexte matériel de recherche à savoir l'outil de recherche et la ou les collection(s) utilisés (paragraphe 5.2.4).
- *fréquence c-fréquence* : ce critère tient compte des statiques des mots clés et des documents ; par exemple, le nombre de fois qu'un mot clé ou un terme a été utilisé dans les requêtes, ou le nombre des fois qu'un document a été retrouvé (paragraphe 5.2.5).

Les trois derniers critères sont sous la responsabilité du système. Ils sont obtenus à partir des données de la session, c'est-à-dire des requêtes soumises, des résultats retrouvés, et des outils et des collections de recherche utilisés. L'aide peut être personnalisée selon ces critères même si les utilisateurs ne participent pas ; c'est-à-dire qu'ils ne jugent pas les résultats et qu'ils ne donnent pas de valeur au critère de préférence.

Bien que ces critères soient présentés séparément, l'utilisateur peut choisir simultanément plusieurs critères. Tous ces critères peuvent être appliqués aussi bien sur le contexte courant que sur le contexte historique. Ces critères sont présentés par l'ensemble *CritèreSoutien* :

CritèreSoutien = { *c-pref*, *c-juge*, *c-requête*, *c-contexte, c-fréquence* }

Ainsi, l'utilisateur qui a besoin d'aide peut interrompre sa boucle de recherche et s'intéresser au soutien en faisant une demande *DemandeSoutien* de la forme suivante :

DemandeSoutien (*t-soutien*, *TempsSession*, *C-Soutien*)

Il détermine dans sa demande non seulement le type de soutien *t-Soutien* (qui peut être une *présentation des requêtes*, une *fusion des requêtes*, une *présentation des résultats*, ou une *reformulation de requête*), et la dimension temporelle *TempsSession* (soit *courant* soit

historique), mais aussi un sous-ensemble *C-Soutien* de l'ensemble des critères de personnalisation *CritèreSoutien* possibles et proposés par le système de soutien (*C-Soutien* ⊂ *CritèreSoutien*).

La Figure 5.2 permet d'avoir une vision globale de la structure d'une demande de soutien ainsi que les critères de personnalisation possibles et leurs valeurs.

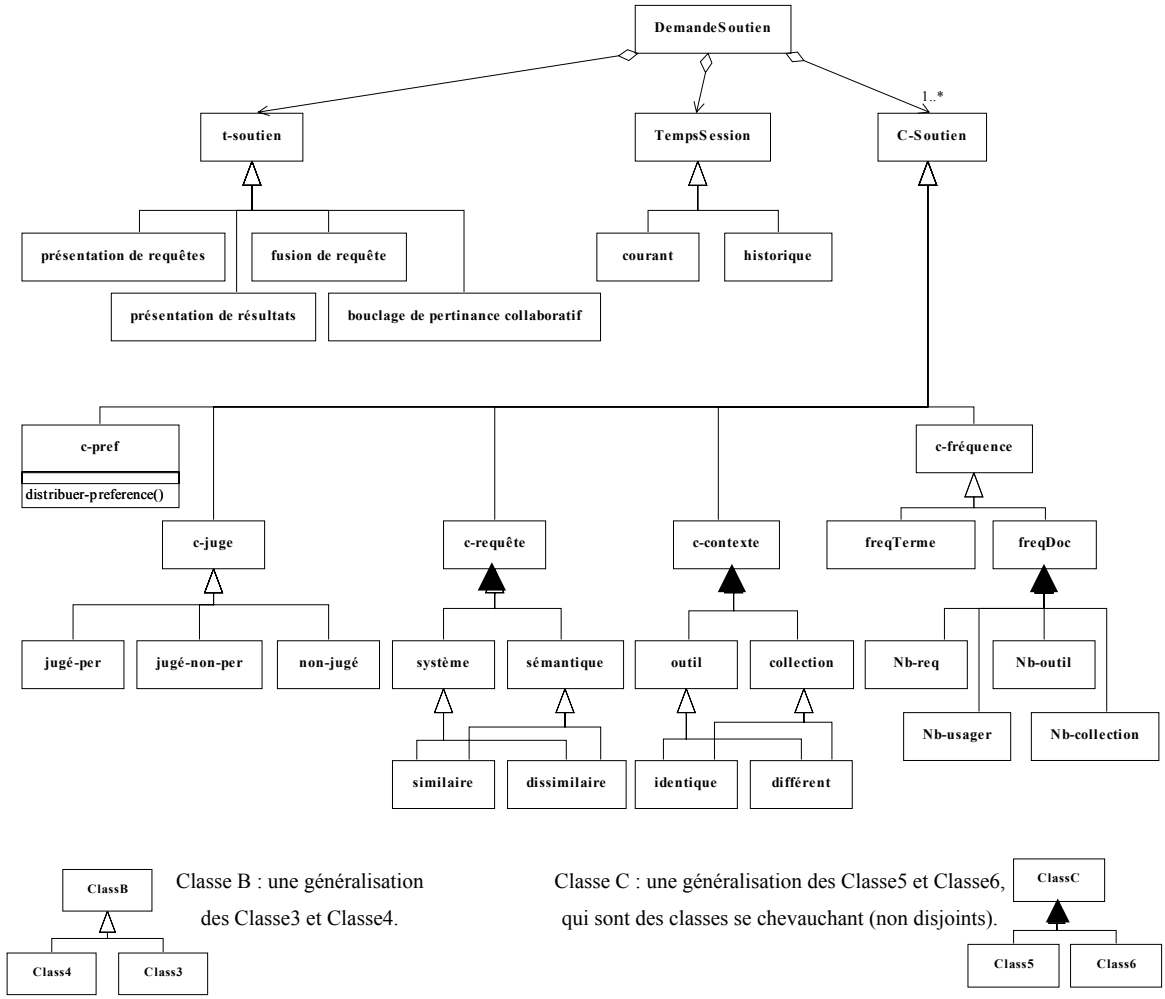

Figure 5.2. *Modèle d'une demande de soutien.*

5.2.1. *Préférence*

Ce critère tient compte des valeurs de préférence. Via ce critère, le soutien peut tenir compte soit de l'opinion de l'utilisateur intéressé, soit de l'évaluation du système ou des deux (voir le paragraphe 4.4).

Supposons que l'utilisateur u ait demandé de l'aide, et que la préférence $Pref_u(u_j)$ soit donnée pour chaque utilisateur u_j. L'utilisateur u_j a effectué la requête q_{uj}, les ensembles de requêtes *QPref* et de résultats *RPref* sont alors définis par :

$QPref(u) = \{ q_{uj} : Pref(u_j) \geq \alpha_{pref} \}$
$RPref(u) = \{ R_{quj} : q_{uj} \in QPref(u) \}$

La valeur $Pref(u_j)$ est dans l'intervalle [0, 1], α_{pref} est un seuil pour sélectionner les sessions de recherche des utilisateurs dont le degré de préférence est le plus élevé.

La valeur α_{pref} est déterminée de façon différente selon l'acteur qui détermine la valeur de préférence :

- l'*utilisateur* : $Pref(u_j) = Pref_{ui}(u_j)$ alors $\alpha_{pref} = 0$ et $QPref(u) = \{ q_{uj} : Pref(u_j) > \alpha_{pref} \}$
- le *système* : $Pref(u_j) = Eval\text{-}RII_{uj}$ alors $\alpha_{pref} = \alpha_{Prec\text{-}m}$ ou $\alpha_{pref} = \alpha_{Ord\text{-}m}$ selon que le système utilise $EvalPrec\text{-}RII_{uj}$ ou $EvalOrd\text{-}RII_{uj}$ pour l'utilisateur[1] u_j (voir le paragraphe 4.4).
- l'*utilisateur* et le *système* : $Pref(u_j) = ½ (Pref_{ui}(u_j) + Eval\text{-}RII_{uj})$ alors $\alpha_{pref} = \alpha_{Ord\text{-}m}$ ou $\alpha_{pref} = \alpha_{Prec\text{-}m}$. Dans le cas où l'utilisateur u_j est ignoré par le système parce que $EvalPrec\text{-}RII_{uj} < \alpha_{Prec\text{-}m}$ ou $EvalOrd\text{-}RII_{uj} < \alpha_{Ord\text{-}m}$ alors $Eval\text{-}RII_{uj} = 0$.

La valeur de préférence d'une requête (ou d'un document) est la valeur de préférence de l'utilisateur qui l'a soumise (ou l'a retrouvé).

Ce critère peut être intéressant pour :

- *apprendre sur le domaine de recherche* : c'est-à-dire, pour aider un utilisateur novice dans le domaine de recherche, on peut lui indiquer les requêtes de l'ensemble *QPref* formulées par les utilisateurs qu'il a « préféré » pour leur compétence dans le domaine, ou lui présenter les documents *RPref* de ces requêtes.
- *apprendre sur la technique de recherche* : pour aider un utilisateur novice dans la technique de recherche, on peut lui indiquer les requêtes d'autres utilisateurs sélectionnés pour leur compétence dans la technique.

Nous avons présenté une explication exhaustive de cela dans le paragraphe 4.4 où la stratégie du soutien selon la typologie de l'utilisateur prend justement en charge la détermination du contexte de l'utilisateur et son besoin d'apprendre que ce soit sur le domaine ou sur la technique de recherche. Ainsi, dans le cas où le système intervient pour déterminer la préférence, il choisit les utilisateurs qui peuvent aider l'utilisateur selon le contexte détecté pour celui-ci.

5.2.2. *Jugement*

Ce critère *c-juge* tient compte des jugements des utilisateurs sur les documents qu'ils ont obtenus. Il peut prendre les valeurs suivantes :

$c\text{-}juge \in \{ jugé\text{-}per, jugé\text{-}non\text{-}per, non\text{-}jugé \}$

---

[1] $\alpha_{Ord\text{-}m}$ et $\alpha_{Prec\text{-}m}$ sont des seuils qui déterminent les niveaux de compétence *expert* pour un utilisateur donné u_j :
si $\alpha_{Prec\text{-}m} \leq EvalPrec\text{-}RII_{uj} \leq 1$ alors u_j est expert dans le domaine de recherche.
si $\alpha_{Ord\text{-}m} \leq EvalOrd\text{-}RII_{uj} \leq 1$ alors u_j est expert dans la technique de recherche.

Afin de clarifier ce critère, nous rappelons que : *JugeDoc* est le jugement accordé au document *Doc* par l'utilisateur pour une requête donnée, dans une sessiondonnée, $EvalEtape_j$ est l'évaluation de l'étape de recherche lorsque la requête q_j a été posée. Ces valeurs *JugeDoc* et $EvalEtape_j$ sont dans l'intervalle [0, 1].

Nous avons présenté le calcul de la valeur *JugeDoc* dans le paragraphe 4.2.3.5, et celui de la valeur $EvalEtape_j$ dans le paragraphe 4.3.2.2.1, où nous avons distingué deux mesures pour cette dernière : $EvalPrec\text{-}Etape_k$ qui prend en compte la précision et $EvalOrd\text{-}Etape_k$ qui prend en compte l'ordonnancement des documents.

Afin de concrétiser l'efficacité ou non de la requête selon ces deux mesures, nous prenons les deux seuils α_{Prec}, α_{Ord} pour déterminer si la requête est efficace ou non pour le sujet et/ou pour le système, c'est-à-dire déterminer si la requête q_j est plutôt efficace parce qu'elle trouve beaucoup de documents pertinents, si elle est plutôt efficace parce qu'elle les ordonne bien, si elle est efficace pour les deux, ou pour aucune, comme il est illustré dans le Tableau 4.7.

Pour le Sujet		**Pour le Système**	
non efficace	$0 \leq EvalPrec\text{-}Etape_j < \alpha_{Prec}$	**non efficace**	$0 \leq EvalOrd\text{-}Etape_j < \alpha_{Ord}$
efficace	$\alpha_{Prec} \leq EvalPrec\text{-}Etape_j \leq 1$	**efficace**	$\alpha_{Ord} \leq EvalOrd\text{-}Etape_j \leq 1$

Tableau 5.1. *Efficacité de la requête pour le sujet et le système de recherche.*

5.2.2.1. Jugement Pertinent *jugé-per*

Cette valeur considère les meilleurs documents obtenus, c'est-à-dire ceux qui sont jugés pertinents par les utilisateurs. Elle détermine deux ensembles : *Qjugé-per* ensemble des requêtes ayant obtenu une meilleure évaluation, et *Rjugé-per* l'ensemble de documents jugés pertinents par l'utilisateur.

Quand le système sélectionne les requêtes selon leurs évaluations c'est-à-dire selon la valeur $EvalEtape_j$, il a deux possibilités : le choix de la valeur $EvalPrec\text{-}Etape_j$ ou le choix de la valeur $EvalOrd\text{-}Etape_j$. En effet, de façon semblable à la stratégie de soutien selon la typologie de l'utilisateur, le système considère l'une de ces deux valeurs, selon le contexte détecté de l'utilisateur.

Nous avons défini dans le paragraphe 4.4.2 trois contextes possibles de l'utilisateur u_i qui a besoin d'aide. Nous déterminons l'évaluation de la requête selon le contexte de l'utilisateur u_i de la façon suivante :

Contexte.1. $EvalPrec\text{-}RII_{ui} < EvalOrd\text{-}RII_{ui}$: l'utilisateur u_i a un niveau de compétence sur le domaine moins élevé que celui sur le système. Le système considère $Eval\text{-}Etape_j = EvalPrec\text{-}Etape_j$ pour les requêtes q_j.

Contexte.2. $EvalPrec\text{-}RII_{ui} > EvalOrd\text{-}RII_{ui}$: cependant si l'utilisateur u_i a un niveau de compétence dans le domaine plus élevé que dans le système. Le système prend $Eval\text{-}Etape_j = EvalOrd\text{-}Etape_j$.

Contexte.3. $EvalPrec\text{-}RII_{ui} = EvalOrd\text{-}RII_{ui}$: l'utilisateur u_i a un niveau de compétence dans le domaine égal à celui dans le système. Le système détermine $Eval\text{-}Etape_j = EvalPrec\text{-}Etape_j$.

Le choix de la valeur *Eval-Etape*$_j$ est résumé dans le Tableau 4.8, où les trois premières colonnes du tableau représentent les différents contextes possibles de l'utilisateur u_i selon ses compétences, tandis que la quatrième présente le choix de la valeur *Eval-Etape*$_j$.

L'utilisateur u_i à soutenir/ à aider			**les requêtes q_j**
contexte	**compétence du domaine**	**compétence technique**	
1	**débutant**	**expert**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
1	**débutant**	**moyen**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
3	**débutant**	**débutant**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
1	**moyen**	**expert**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
3	**moyen**	**moyen**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
2	**moyen**	**débutant**	*Eval-Etape*$_j$ = *EvalOrd-Etape*$_j$
3	**expert**	**expert**	*Eval-Etape*$_j$ = *EvalPrec-Etape*$_j$
2	**expert**	**moyen**	*Eval-Etape*$_j$ = *EvalOrd-Etape*$_j$
2	**expert**	**débutant**	*Eval-Etape*$_j$ = *EvalOrd-Etape*$_j$

Tableau 5.2. *Choix de valeur d'évaluation de requête selon la typologie de l'utilisateur.*

Maintenant nous définissons les deux ensembles des meilleures requêtes *Qjugé-per* et des meilleurs documents *Rjugé-per* ainsi :

$$Qjugé\text{-}per = \{\, q_j : EvalPrec\text{-}Etape_j \geq \alpha_{Prec} \quad \text{Si} \quad EvalPrec\text{-}RII_{ui} \geq EvalOrd\text{-}RII_{ui}$$
$$EvalOrd\text{-}Etape_j \geq \alpha_{Ord} \quad \text{Si} \quad EvalPrec\text{-}RII_{ui} < EvalOrd\text{-}RII_{ui} \,\}$$
$$Rjugé\text{-}per = \{\, Doc : JugeDoc > \alpha_{peres} \,\}$$

où α_{peres} sont des seuils qui déterminent la notion de document pertinent, ici[2] $\alpha_{peres} = 0$.

Remarque : l'évaluation de la requête *Eval-Etape*$_j$, ainsi que le jugement de document *JugeDoc* peuvent être *globaux* ou *partiels* selon que l'on prenne en compte tous les critères de jugement ou seulement une partie. L'utilisateur peut choisir dans sa demande de soutien les critères de jugement à prendre en compte, c'est-à-dire il peut être intéressé par l'ensemble des requêtes *Qjugé-per* ayant obtenu une meilleure évaluation, pour certains critères de jugement ou par l'ensemble des documents *Rjugé-per* jugés pertinents pour certains critères de jugement, alors dans ces deux cas les valeurs *Eval-Etape*$_j$ et *JugeDoc* sont prises comme les valeurs partielles selon les critères de jugements choisis.

Cette valeur *jugé-per* donne un résumé de la recherche collaborative selon un critère de qualité, et sert à :

- *apprendre sur la technique de recherche* : une façon d'aider un utilisateur novice dans la technique de recherche est de lui donner une idée de la façon de procéder. Par exemple, on peut lui suggérer une présentation de requêtes efficaces pour le système, ou une nouvelle requête via la fusion de celles-ci. Cela peut lui donner des idées pour formuler de bonnes requêtes.

[2] *JugeDoc* > 0 : le document *Doc* jugé pour au moins un critère *i* de qualité avec une valeur c_i supérieure à zéro ($c_i > 0$).

- *apprendre sur le domaine de recherche* : c'est-à-dire, pour aider un utilisateur novice dans le domaine de recherche, on peut lui indiquer les requêtes efficaces pour le sujet, ou une fusion de celles-ci.
- *avoir une vision positive sur le processus de recherche* : cette vision synthétise les meilleurs résultats ou requêtes de la recherche. Cela peut être fait via la présentation des documents pertinents de l'ensemble *Rjugé-per*.
- *étendre une requête* : on utilise une mécanisme de bouclage de pertinence qui considère les jugements fournis par les utilisateurs, cela fera l'objet du paragraphe 5.4.3.

5.2.2.2. Jugement Non Pertinent *jugé-non-per*

Cette valeur détermine deux ensembles, *Qjugé-non-per* celui de requêtes considérées comme non efficaces pour satisfaire le besoin d'information, et *Rjugé-non-per* celui de documents jugés non pertinents :

$Qjugé\text{-}non\text{-}per = \{ q_j : EvalPrec\text{-}Etape_j < \alpha_{Prec} \ \& \ EvalOrd\text{-}Etape_j < \alpha_{Ord} \}$
$Rjugé\text{-}non\text{-}per = \{ Doc : JugeDoc \leq \alpha_{peres} \}$

Cette valeur permet :

- de *vérifier et* de *s'assurer de la compréhension du sujet de recherche* : cela permettra à un utilisateur de vérifier si les autres utilisateurs ont bien compris le sujet de la recherche, ou s'il a lui-même bien compris le sujet de la recherche. Cela peut être réalisé via la présentation des documents de l'ensemble *Rjugé-non-per* ou la présentation de requêtes de l'ensemble *Qjugé-non-per*.
- *évaluer l'outil de recherche* : un chercheur peut s'intéresser à une présentation des documents retrouvés par l'outil, qui sont pertinents selon le système, mais jugés non pertinents par les utilisateurs. Ainsi, il peut étudier l'efficacité de cet outil de recherche en terme de pertinence-système et pertinence-utilisateur.

5.2.2.3. Sans Jugement *non-jugé*

Il y a certains documents retrouvés par les outils mais qui ne sont pas jugés par les utilisateurs, cette valeur à son tour détermine deux ensembles :

$Qnon\text{-}jugé = \{ q_j : Eval\text{-}Etape_j = nul \}$
$Rnon\text{-}jugé = \{ Doc : JugeDoc = nul \}$

Où la valeur *nul* exprime l'absence de jugement, elle peut servir à :

- *juger* : un utilisateur peut justement être intéressé par une présentation de ces documents non jugés et il peut donner son jugement, un document non jugé n'est pas forcement non-pertinent.
- *étudier les raisons d'absence de jugement* : un chercheur peut s'intéresser à une présentation des documents retrouvés et non-jugés, ou par une présentation des requêtes de l'ensemble *Qnon-jugé*, et ce afin d'étudier les raisons de l'absence de jugement. Est-ce l'outil qui ne le satisfait pas du tout ? l'utilisateur est-il novice et n'arrive-t-il pas à

formuler de bonnes requêtes ? Est-ce que l'utilisateur n'a pas conscience de l'impact et de l'intérêt de son jugement pour la collaboration ?...etc.

5.2.3. *Similarité de Requêtes*

La similarité des requêtes peut être envisagée selon deux points de vues :

1. du point de vue du système : les requêtes sont similaires si elles ont des termes communs (paragraphe 5.2.3.1).
2. du point de vue sémantique : si elles conduisent à obtenir des documents similaires même si elles n'ont peu ou pas de termes en commun (paragraphe 5.2.3.2).

Nous expliquons dans la suite ces deux points de vue, et dans le paragraphe 5.2.3.3. l'intérêt de ce critère de similarité

5.2.3.1. Point de Vue du Système

Afin de mesurer la similarité $SimReq(q_u, q_j)$ entre la requête q_u de l'utilisateur u et une autre requête q_j il faut prendre en compte les différents modèles des requêtes. Pour cela, nous présentons le calcul de la similarité entre deux requêtes vectorielles (paragraphe 5.2.3.1.1) et entre deux requêtes booléennes (paragraphe 5.2.3.1.2) et entre une vectorielle et autre booléenne (paragraphe 5.2.3.1.3), et enfin nous expliquons les ensembles de requêtes et de documents choisis selon ce critère (paragraphe 5.2.3.1.4).

5.2.3.1.1. Similarité des Requêtes Vectorielles

La similarité $SimReq(q_u, q_j)$ entre les deux requêtes vectorielles q_u et q_j est mesurée comme la cosinus de l'angle entre ces deux vecteurs ainsi :

$$SimReq\ (q_u, q_j) = cos\ (\overrightarrow{q_u}, \overrightarrow{q_j}) = \frac{\overrightarrow{q_u} \cdot \overrightarrow{q_j}}{\left\|\overrightarrow{q_u}\right\| \cdot \left\|\overrightarrow{q_j}\right\|} \qquad [5.1]$$

où || . || est la norme Euclidienne d'un vecteur.

5.2.3.1.2. Similarité des Requêtes Booléennes

Le calcul de la similarité entre deux requêtes booléennes est plus compliqué. Chaque requête q_u contient deux sous-ensembles :

1. $(T_{qu})^{ET/OU}$: un sous-ensemble de termes désirés (liés par ET/OU) dans le résultat de la requête q_u, et
2. $(T_{qu})^{NON}$: un sous-ensemble de termes non désirés (liés par NON) dans le résultat de la requête q_u.

Supposons que les termes communs entre les deux requêtes q_u et q_j appartiennent à l'ensemble $T_{qu} \cap T_{qj}$, cet ensemble d'intersection contient deux sous-ensembles de termes :

- le premier $(T_{qu} \cap T_{qj})^{\text{ACCORD}}$: est un sous-ensemble de termes désirés (liés par ET/OU) ou non désirés (liés par NON) dans le résultat des deux requêtes, c'est-à-dire les deux requêtes sont d'accord à propos de ce sous-ensemble.
- le deuxième $(T_{qu} \cap T_{qj})^{\text{DESACCORD}}$: est un sous-ensemble de termes désirés seulement dans le résultat d'une des requêtes (termes liés par ET/OU dans une requête et par NON dans l'autre), c'est-à-dire les deux requêtes sont en conflit ou en désaccord à propos de ce sous-ensemble :

$$(T_{qu} \cap T_{qj})^{\text{DESACCORD}} = [(T_{qu})^{\text{ET/OU}} \cap (T_{qj})^{\text{NON}}] \cup [(T_{qu})^{\text{NON}} \cap (T_{qu})^{\text{ET/OU}}]$$

$$T_{qu} \cap T_{qj} = (T_{qu} \cap T_{qj})^{\text{ACCORD}} \cup (T_{qu} \cap T_{qj})^{\text{DESACCORD}}$$

La similarité entre des requêtes booléennes est calculée en considérant le sous ensemble $(T_{qu} \cap T_{qj})^{\text{ACCORD}}$ ainsi :

$$SimReq\ (q_u, q_j) = \frac{\left|(T_{qu} \cap T_{qj})^{ACCORD}\right|}{\left|T_{qu} \cup T_{qj}\right|} \qquad [5.2]$$

L'exemple suivant clarifie cela, prenons les trois besoins d'information différents suivants avec les requêtes correspondantes :

1. besoin 1 : « on s'intéresse aux processus dans les systèmes d'exploitation et en particulier, dans le système Unix », alors la requête pour ce besoin est :

   q_1 : processus ET [(système ET exploitation) OU Unix]

2. besoin 2 : « on s'intéresse aux processus dans les systèmes d'exploitation autres que le système Unix » et la requête correspondant est :

   q_2 : processus ET [(système ET exploitation) NON Unix]

3. besoin 3 : « on s'intéresse aux processus dans les systèmes d'exploitation Unix » et la requête q_3 est :

   q_3 : processus ET [(système ET exploitation) ET Unix]

On remarque que l'ensemble de termes communs entre n'importe quelles deux requêtes q_u, q_j est le même :

$(T_{qu} \cap T_{qj})$ = {processus, Unix, système, exploitation}

tandis que la différence entre ces requêtes se nuance à propos des sous-ensembles $(T_{qu} \cap T_{qj})^{\text{ACCORD}}$ et $(T_{qu} \cap T_{qj})^{\text{DESACCORD}}$ ainsi :

- $(T_{q1} \cap T_{q2})^{\text{ACCORD}}$ = {processus, système, exploitation} et $(T_{q1} \cap T_{q2})^{\text{DESACCORD}}$= {Unix}

  où q_1 désire le terme « Unix » lié par OU tandis que q_2 l'exclue, il est précédé par NON.

  $SimReq\ (q_1, q_2)$ = ¾

- $(T_{q1} \cap T_{q3})^{\text{ACCORD}}$= {processus, système, exploitation, Unix} et $(T_{q1} \cap T_{q2})^{\text{DESACCORD}}$=∅

  où q_1 désire le terme « Unix » en le liant par OU et q_2 aussi en le liant par ET.

$SimReq\ (q_1, q_3) = 1$

5.2.3.1.3. Similarité des Requêtes Vectorielles et Booléennes

Quand on a deux requêtes une vectorielle et l'autre booléenne, deux solutions sont possibles pour calculer leur similarité :

1. transformer la requête booléenne en requête vectorielle : où les termes de la requête booléenne sont repris et les mots clés appartenant à une négation NON sont omis [Savoy 1994b].
2. transformer la requête vectorielle en requête booléenne : en considérant que les termes de la requête vectorielle sont liés par ET.

Après cette transformation les deux requêtes ont la même représentation et on peut mesurer leur similarité selon la formule [5.1] ou [5.2].

5.2.3.1.4. Choix des Requêtes et des Documents

Nous mentionnons que la valeur de la similarité des requêtes du point de vue du système *SimReq* est une valeur dans l'intervalle [0, 1] quels que soient les modèles des deux requêtes.

Ce critère de similarité de requêtes *c-requête* peut prendre deux valeurs : deux requêtes peuvent être semblables (proches) ou dissemblables (lointaines).

Deux ensembles $QSimReq(q_u)$ et $QDISimReq(q_u)$ sont déterminés selon les similarités et les dissimilarités d'une requête avec la requête q_u :

$QSimReq\ (q_u) = \{q_j : SimReq\ (q_u, q_j) \geq \alpha_{simreq}\ \}$
$QDISimReq\ (q_u) = \{q_j : SimReq\ (q_u, q_j) < \alpha_{simreq}\ \}$

où α_{simreq} est un seuil qui détermine la notion de requête proche ou lointaine, nous assumons $\alpha_{simreq} = 0.5$. La similarité de requête ne détermine pas seulement ces deux ensembles de requêtes mais aussi deux ensembles de résultats $RSimReq(q_u)$ et $RDISimReq(q_u)$ obtenus respectivement par des requêtes semblables ou dissemblables à la requête q_u :

$RSimReq\ (q_u) = \{\ R_{qj} : q_j \in QSimReq\ (q_u)\ \}$
$RDISimReq\ (q_u) = \{\ R_{qj} : q_j \in QDISimReq\ (q_u)\ \}$

5.2.3.2. Point de Vue Sémantique

La similarité $SimRes\ (q_u, q_j)$ entre les deux requêtes q_u et une autre requête q_j est considérée du point de vue des documents communs. Dans cette formule, le symbole R_{qu} représente l'ensemble des documents obtenus par l'utilisateur u en réponse à sa requête q_u :

$$SimRes\ (q_u, q_j) = \frac{\left|R_{q_u} \cap R_{q_j}\right|}{\left|R_{q_u} \cup R_{q_j}\right|}$$

SimRes est une valeur dans l'intervalle [0, 1].

Nous pouvons également définir des ensembles similaires à ceux qui sont définis par le point du vue précédent, en se focalisant sur la similarité des résultats des requêtes.

Deux ensembles de requêtes $QSimRes(q_u)$ et $QDISimRes(q_u)$ sont déterminés selon les similarités et les dissimilarités de résultat d'une requête avec le résultat de la requête q_u

$$QSimRes\ (q_u) = \{\ q_j : SimRes\ (R_{qu}, R_{qj}) \geq \alpha_{simres}\ \}$$
$$QDISimRes\ (q_u) = \{\ q_j : SimRes\ (R_{qu}, R_{qj}) < \alpha_{simres}\ \}$$

où α_{simres} est un seuil qui détermine la notion de requête proche ou éloignée du point de vue sémantique, nous prenons la valeur 0.5 pour ce seuil. Deux ensembles de documents $RSimRes(q_u)$ et $RDISimRes(q_u)$ sont déterminés ainsi :

$$RSimReq\ (q_u) = \{\ R_{qj} : q_j \in QSimRes\ (q_u)\ \}$$
$$RDISimReq\ (q_u) = \{\ R_{qj} : q_j \in QDISimRes\ (q_u)\ \}$$

5.2.3.3. Intérêt du Critère de Similarité de Requêtes

Nous résumons l'intérêt de ce critère de similarité dans les points suivants :

- *diminuer la redondance des requêtes* : le fait de savoir qu'une requête a déjà été posée est une information intéressante, qui peut augmenter l'efficacité du groupe (en terme de temps) en évitant la redondance. Cela peut être atteint via la présentation des requêtes qui sont proches de la requête de *u*, (critère : requêtes similaires du point de vue système).
- *comparer et s'évaluer* : dans un cas de blocage, donc lorsque l'utilisateur n'arrive pas à un résultat satisfaisant, il peut avoir besoin de s'évaluer par rapport aux autres. Par exemple, une présentation des résultats obtenus par des requêtes très différentes de la sienne du point de vue système lui permet d'évaluer si c'est sa propre requête qui n'est pas correcte ou si les collections ne contiennent pas de résultats intéressants, (critère : requêtes dissimilaires du point de vue système).
- *exploiter tous les aspects du sujet de recherche* : une aide pour l'utilisateur serait peut-être de lui fournir une présentation des requêtes dont les résultats sont lointains et bien évalués ou une présentation des résultats lointains du sien sans considérer la requête à l'origine de l'obtention des résultats. Il peut ainsi cerner d'autres aspects du sujet de recherche (critère : requêtes dissimilaires du point de vue sémantique).
- *obtenir un bon résultat* : dans [Belkin 1993], (voir le paragraphe 3.1) les auteurs ont montré que la combinaison des résultats des requêtes booléennes formulées par des experts de recherche pour le même besoin d'information a souvent amélioré la performance. Selon cette recherche, et si nous supposons que les requêtes similaires expriment le même besoin, nous pensons qu'une présentation des résultats dont les requêtes sont similaires peut donner une bonne performance (critère : requêtes similaires du point de vue système).

On pouvait aussi combiner les deux points de vue pour :

- *centrer la recherche* : où l'utilisateur peut sélectionner des requêtes différentes des siennes et qui ont, néanmoins, donné des documents communs. Supposons que

l'utilisateur ait soumis une bonne requête (l'utilisateur a obtenu un résultat satisfaisant) et qu'il veuille se concentrer sur le même sous-ensemble de documents renvoyés par le système de recherche comme résultat. Ainsi il peut obtenir cela par la fusion des requêtes dont les résultats sont similaires (critère : requêtes dissimilaires du point de vue système, et similaires du point de vue sémantique).

- *étendre une requête* : en ajoutant des termes synonymes ou d'autre variété morphologique, qui peuvent apparaître dans des requêtes similaires ou dissimilaires selon un ou les deux points de vue système ou sémantique. Nous revenons à cette technique dans le paragraphe 5.4.2.

5.2.4. *Contexte Matériel*

Il y a deux éléments qui concernent le contexte matériel de recherche : l'*outil* et la *collection* de recherche. Chacun de ces éléments peut prendre deux valeurs : *identique* ou *différent*. En général le choix de l'outil de recherche implique le choix de la collection.

L'utilisateur peut s'intéresser aux sessions de recherche des utilisateurs qui ont les mêmes outils de recherche que lui, ou au contraire, aux utilisateurs qui effectuent leurs recherches sur les mêmes collections mais avec des outils différents. Chacun de ces deux éléments (*outil/collection*) détermine pour chaque valeur (*identique/différent*) deux ensembles de requêtes et de résultats :

QOutilIdentique (u) = { q_j : q_j est soumise sur un outil identique à celui de u }
ROutilIdentique (u) = { R_{qj} : $q_j \in$ *QOutilIdentique* (u) }
QOutilDiff (u) = { q_j : q_j est soumise sur un outil différent de celui de u }
ROutilDiff (u) = { R_{qj} : $q_j \in$ *QOutilIdentique* (u) }
QCollectionIdentique(u) = {q_j : q_j est soumise sur une/des collection(s) identique(s) à celle(s) de u}
RCollectionIdentique (u) = { R_{qj} : $q_j \in$ *QCollectionIdentique* (u) }
QCollectionDiff(u) = {q_j : q_j est soumise sur une/des collection(s) différente(s) à celle(s) de u}
RCollectionDiff (u) = { R_{qj} : $q_j \in$ *QCollectionDiff* (u) }

Ce critère peut aider à :

- *couvrir l'espace d'information* (*la collection*) : cela est souhaitable car cela permet à l'utilisateur d'avoir une idée sur le contenu de la collection. Il peut savoir s'il a exploité « toutes » les collections de recherche et si elles contiennent d'autres facettes du sujet de recherche qu'il a ignoré. Pour cela, on lui indique les résultats obtenus à partir de la ou des même(s) collection(s). On pourrait aussi lui indiquer les requêtes qui ont permis d'obtenir ces résultats.
- *comparer et s'évaluer* : l'utilisateur peut évaluer sa propre méthode de recherche par confrontation avec les autres. Il peut ainsi savoir si avec le même outil ou la même collection, les autres sont arrivés à des résultats quasiment identiques aux siens ou non, et quelles requêtes ont été soumises.
- *obtenir un bon résultat* : dans différentes études présentées dans le chapitre de fusion (paragraphe 3.4) on a trouvé que la combinaison des résultats issus de différentes collections (fusion de collections) ou de différents outils (fusion des systèmes) améliorent la performance de recherche. Nous déduisons des résultats de ces études qu'une

présentation des résultats issus de différentes collections *RCollectionDiff*(*u*) ou outils *ROutilDiff*(*u*) devrait aboutir à de bons résultats.

- *étudier les outils et les collections de recherche* : un chercheur peut s'intéresser à comparer les performances des outils ou des collections en comparant leurs résultats. Il peut également s'intéresser à l'impact de la même requête sur des outils ou des collections différents.

5.2.5. *Fréquences*

Le critère de fréquence de recherche détermine deux ensembles de documents et de termes que nous allons expliquer : *Fréquence de Document* (paragraphe 5.2.5.1), *Fréquence de Terme* (paragraphe 5.2.5.2).

5.2.5.1. Fréquence de Document

Le document dans un contexte de recherche collaborative peut être retrouvé par : plusieurs outils/collections, et par plusieurs requêtes. Ces requêtes sont effectuées par un ou plusieurs utilisateurs (Figure 5.3). Cette fréquence peut prendre plusieurs valeurs :

FréqDoc (*Nb-usager, Nb-req, Nb-outil, Nb-collection*)

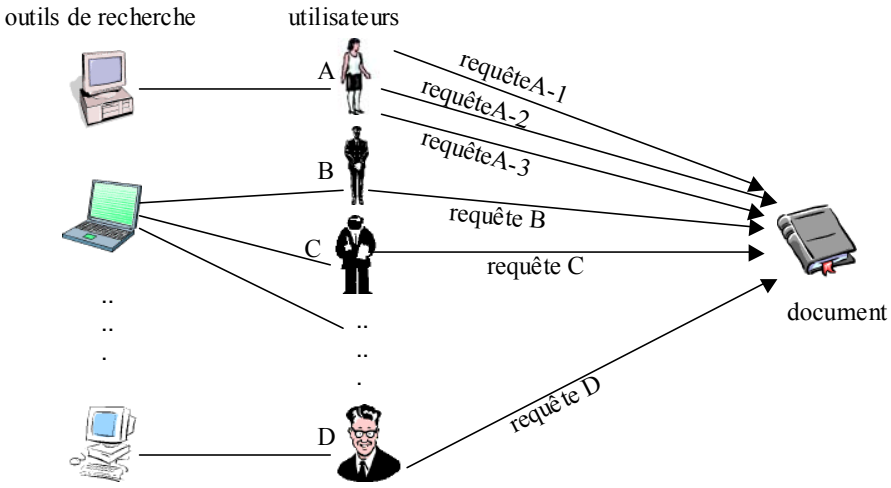

Figure 5.3. *Un document dans le contexte de recherche collaborative.*

On peut faire l'hypothèse qu'un document est plus intéressant, quand plus d'un utilisateur l'a trouvé et l'a jugé pertinent, ou s'il est trouvé par plusieurs outils. Cette hypothèse peut être prise en compte pour la présentation des documents dans le résultat collectif où on présente les documents fréquemment retrouvés et/ou jugés. Les valeurs du nombre d'utilisateurs, d'outils, etc… doivent alors être normalisées dans l'intervalle [0, 1] :

1. *Nbusager* (*Doc*) est le nombre d'utilisateurs qui ont obtenu *Doc*.

   $NbusagerNorm\ (Doc) = \frac{1}{NbTotalUsager}\ Nbusager\ (Doc)$

2. *NbreqDoc* est le nombre de requêtes via lesquelles *Doc* a été obtenu.

   $NbreqNorm\ (Doc) = \frac{1}{NbTotal\mathrm{Req}}\ Nbreq\ (Doc)$

3. *Nboutil* (*Doc*) est le nombre d'outils de recherche qui ont trouvé *Doc*.

   $NboutilNorm\ (Doc) = \frac{1}{NbTotalOutil}\ Nboutil\ (Doc)$

4. *Nbcollection* (*Doc*) est le nombre de collections de recherche qui ont trouvé *Doc*.

   $NbcollectionNorm\ (Doc) = \frac{1}{NbTotalCollection}\ Nbcollection\ (Doc)$

NbTotalUsager, NbTotalReq, NbTotalOutil, *NbTotalCollection* sont respectivement le nombre total d'utilisateurs, de requêtes soumises, d'outils et de collections utilisés dans la session collaborative. Les valeurs *NbusagerNorm* (*Doc*), *NbreqNorm* (*Doc*), *NboutilNorm* (*Doc*), *NbcollectionNorm* (*Doc*) sont des valeurs normalisées.

On peut combiner plus d'une valeur. Par exemple, les documents obtenus par plusieurs utilisateurs et plusieurs outils. Ceci veut dire qu'un document est plus intéressant quand le nombre d'outils et d'utilisateurs qui l'ont trouvé est grand. Les ensembles de documents sélectionnés selon ces différentes valeurs sont :

$RNbusager = \{ Doc : NbusagerNorm\ (Doc) > \alpha_{Nbusager} \}$
$RNbreq = \{ Doc : NbreqNorm\ (Doc) > \alpha_{Nbreq} \}$
$RNboutil = \{ Doc : NboutilNorm\ (Doc) > \alpha_{Nboutil} \}$
$RNbcollection = \{ Doc : NbcollectionNorm\ (Doc) > \alpha_{Nbcollection} \}$

Où $\alpha_{Nbusager}$, $\alpha_{Nboutil}$,$\alpha_{Nbcollection}$, α_{Nbreq} sont des seuils qui déterminent la valeur à partir de laquelle un document est pris en compte, (nous prenons ici $\alpha_{Nbusager}$, $\alpha_{Nboutil}$, $\alpha_{Nbcollection}$, $\alpha_{Nbreq} = 1$).

5.2.5.2. Fréquence de Terme

La *fréquence des termes de requêtes* est une sorte de résumé des différentes requêtes formulées pendant une session de recherche. Nous définissons la fréquence *Freq*(*t*) de terme *t* des requêtes comme le ratio entre le nombre de requêtes contenant ce terme $|Q_t|$ et le nombre total de requêtes soumises dans la session *NbTotalReq* :

$Freq\ (t) = \frac{|Q_t|}{NbTotal\mathrm{Req}} : Q_t = \{ q_j : t \in q_j \}$
$FreqTerme = \{ t : Freq\ (t) > \alpha_{freq} \}$

où α_{freq} est un seuil à partir duquel on parlera de terme fréquent, (nous prenons $\alpha_{freq} = 1$).

Cela permet à l'utilisateur de réutiliser les termes du domaine de recherche qui sont probablement plus précis pour exprimer son besoin d'information.

5.3. Procédure d'extraction des Requêtes et des Résultats de la Mémoire Collaborative (Recherche dans la mémoire collaborative)

Dans la mémoire collaborative sont enregistrées toutes les informations correspondant aux différentes sessions de recherche des membres du groupe. Nous utilisons les données de la mémoire collaborative pour fournir un soutien efficace à un utilisateur du groupe ; il faut que l'on soit capable d'extraire de cette mémoire un ensemble de requêtes ou de résultats pour aider ou satisfaire le besoin d'un utilisateur donné. En particulier, nous avons besoin de savoir si une requête proposée ou un document retrouvé par un autre utilisateur satisfait les critères précisés dans la demande de soutien. Pour cela, nous définissons une méthode d'extraction des requêtes ou des documents. Cela nous permet d'avoir une vision organisée de la mémoire collaborative.

Cette méthode est constituée de plusieurs étapes, nous donnons, tout d'abord, une description de ces étapes, puis, nous détaillons la description formelle et mathématique de chaque étape. Les étapes de la méthode d'extraction sont les suivantes :

1. *repérage des étapes* de recherche : la mémoire peut être vue soit de façon globale comme un ensemble historique de toutes les étapes, soit de façon partielle comme un sous-ensemble des étapes courantes. Selon la dimension temporelle (*historique* ou *courant*) déterminée dans la demande, l'ensemble historique ou le sous-ensemble courant fait l'objet des opérations suivantes (paragraphe 5.3.1.1).
2. *mise en commun* : à partir des étapes de recherche déterminées précédemment, nous mettons en commun les requêtes ou les documents qui réalisent les critères de la demande de soutien et seulement ceux là, cela veut dire que l'on obtient un sous-ensemble de requêtes ou de documents (paragraphe 5.3.1.2).
3. *calcul d'importance* : c'est le processus suivi pour associer une valeur unique à chacun des requêtes ou à chacun des documents mis en commun. Cette valeur est calculée comme la combinaison des valeurs associées aux critères de soutien. Ainsi, une requête ou un document qui satisfait au mieux les différents critères aura une valeur élevée, et une requête ou un document qui satisfait moins bien ces critères aura une valeur faible. Nous éliminons les requêtes ou les documents dupliqués, pour n'en garder qu'un seul exemplaire (paragraphe 5.3.1.3).
4. *sélection* (*nombre de seuil-limite*) : le nombre des requêtes ou des documents mis en commun peut être très grand, ce qui peut conduire à présenter à l'utilisateur une longue liste de requêtes ou de documents, ou lui suggérer de très longue requête. Pour éviter cela, les requêtes et les documents auxquels sont associées les valeurs calculées précédemment les plus élevées sont sélectionnés de façon à ce que le nombre de requêtes ou de documents sélectionnés soit inférieur ou égal à un seuil donné (paragraphe 5.3.1.4).

La formalisation de la méthode d'extraction est présentée dans la suite.

5.3.1.1. Repérage

A l'issue du *repérage des étapes* de recherche selon la dimension temporelle on obtient l'ensemble temporel d'étapes E_t :

$E_t = E_{historique}$ si *dimension temporelle = historique*
$= E_{courant}$ si *dimension temporelle = courant*

où : $E_{historique} = \{ Etape\text{-}R : Etape\text{-}R \in Session\text{-}RIC \}$
$E_{courant} = \{ Etape\text{-}R : Etape\text{-}R = Session\text{-}RII.Etape\text{-}R_{courant} \wedge Session\text{-}RII \in Session\text{-}RIC \}$

où l'écriture préfixée $objet_1.objet_2$ désigne l'élément $objet_2$ de l'$objet_1$. Par exemple, ici l'écriture $Session\text{-}RII.Etape\text{-}R_{courant}$ désigne l'étape courante $Etape\text{-}R_{courant}$ de la session individuelle *Session-RII*.

5.3.1.2. Mise en Commun

Le critère de soutien *c-soutien* peut prendre plusieurs valeurs *c-soutien* = val_1, *c-soutien* = val_2… Par exemple, le critère *c-requête* peut prendre les deux valeurs : val_1= *proche* et val_2= *lointaine*.

En général[3], chaque valeur val_j du critère de soutien *c-soutien* détermine deux ensembles :

1. l'ensemble des requêtes qui réalisent le critère *c-soutien* pour la valeur val_j :

   $Qc\text{-}soutien = \{q_k : (q_k \text{ réalise } c\text{-}soutien(q_k)) \wedge (c\text{-}soutien = val_j)\}$

2. l'ensemble des documents qui réalisent le critère *c-soutien* pour la valeur val_j :

   $Rc\text{-}soutien = \{Doc : (Doc \text{ réalise } c\text{-}soutien(Doc)) \wedge (c\text{-}soutien = val_j)\}$

Si nous reprenons notre exemple *c-requête = proche*, la similarité des requêtes du point de vue système, cette valeur détermine les deux ensembles :

$QSimReq\ (q_u) = \{ q_j : SimQ(q_u, q_j) \geq \alpha_{simreq} \}$
$RSimReq\ (q_u) = \{ R_{qj} : q_j \in QSim(q_u) \}$

où : q_j réalise *c-requête*(q_j) pour la valeur *proche* signifie que $SimQ(q_u, q_j) \geq \alpha_{simreq}$, et *Doc* réalise *c-requête*(*Doc*) pour la valeur *proche* signifie que $Doc \in R_{qj} : q_j \in QSim(q_u)$.

Nous venons d'expliquer comment les ensembles de requêtes et de documents sont déterminés pour un seul critère de soutien *c-soutien*. Nous avons déjà mentionné que nous donnons la possibilité à l'utilisateur de personnaliser le soutien pour plusieurs critères. Maintenant, nous expliquons comment déterminer les ensembles de requêtes et de documents pour tous les critères demandés *c-soutien* $\in$ *C-Soutien.* Les deux ensembles sont :

1. l'ensemble des requêtes qui réalisent tous les critères de *C-Soutien* :

   $QC\text{-}Soutien = \{q_k : q_k \in \cap_{c\text{-}soutien \in C\text{-}Soutien} Qc\text{-}soutien \}$

   où : $\cap_{c\text{-}soutien \in C\text{-}Soutien}$ *Qc-soutien* est l'intersection de tous les sous-ensembles de requêtes qui réalisent chacun des critères de *C-Soutien*.

[3] Sauf pour le critère *c-fréquence* qui détermine deux ensembles de documents et de termes.

2. l'ensemble des documents qui réalisent tous les critères de *C-Soutien* :

$$RC\text{-}Soutien = \{Doc : Doc \in \bigcap_{c\text{-}soutien \in C\text{-}Soutien} Rc\text{-}soutien \}$$

où : $\bigcap_{c\text{-}soutien \in C\text{-}Soutien} Rc\text{-}soutien$ est l'intersection de tous les sous-ensembles de documents qui réalisent chacun des critères de *C-Soutien*.

Ainsi les critères *C-Soutien* précisés dans la demande de soutien, déterminent deux ensembles de requêtes *QC-Soutien* et de documents *RC-Soutien*, qui réalisent tous les critères de la demande.

Maintenant, nous mettons en commun dans QE_{tc} les requêtes $Etape\text{-}R.q_k$, ou dans RE_{tc} les documents *Etape-R.Doc* des étapes de l'ensemble E_t, qui réalisent tous les critères *C-Soutien* ainsi :

- *mise en commun des requêtes* : qui réalisent tous les critères de *C-Soutien* :

$$QE_{tc} = \{q_k : Etape\text{-}R.q_k \in QC\text{-}Soutien \wedge Etape\text{-}R \in E_t \}$$

- *mise en commun des documents* : qui réalisent tous les critères de *C-Soutien* :

$$RE_{tc} = \{ Doc : Etape\text{-}R.Doc \in RC\text{-}Soutien \wedge Etape\text{-}R \in E_t \}$$

5.3.1.3. Calcul d'Importance

Nous définissons une mesure afin de déterminer l'importance d'une requête ou d'un document dans l'ensemble précédent. La façon la plus simple d'évaluer cette importance est la combinaison linaire.

Ainsi, une requête dans QE_{tc} ou un document dans RE_{tc} a son importance obtenue par la fonction *L* :

$$L(Etape\text{-}R.q_k) = \frac{1}{|C-Soutien|} \sum_{c\text{-}soutien \in C\text{-}Soutien} c\text{-}soutien(Etape\text{-}R.q_k),\ Etape\text{-}R.q_k \in QE_{tc}$$

$$L(Etape\text{-}R.Doc) = \frac{1}{|C-Soutien|} \sum_{c\text{-}soutien \in C\text{-}Soutien} c\text{-}soutien(Etape\text{-}R.Doc),\ Etape\text{-}R.Doc \in RE_{tc}$$

Il se peut qu'un document soit doublé (cela peut arriver à une requête aussi) et que chaque duplication obtienne une valeur $c\text{-}soutien(Doc)/c\text{-}soutien(q_k)$ pour la réalisation d'un critère de soutien. Pour résoudre le problème de plusieurs valeurs d'un document (ou d'une requête) pour le même critère nous avons recours à la moyenne de ces valeurs : prenons $c\text{-}soutien_1(Doc)$, $c\text{-}soutien_2(Doc)$, ..., $c\text{-}soutien_K(Doc)$ les différentes valeurs d'un document *Doc* pour le critère *c-soutien.* La valeur finale accordée à ce document *c-soutien* (*Doc*) est alors la moyenne de ces valeurs :

$$c\text{-}soutien\ (Doc) = \frac{1}{K}\ (\ c\text{-}soutien_1\ (Doc) + c\text{-}soutien_2\ (Doc) + ... + c\text{-}soutien_K\ (Doc)\ )\quad [3.3]$$

Par exemple, le document *ref* retrouvé et jugé par deux utilisateurs, et on a les valeurs de jugement $JugeDoc_1(ref) = 1$, $JugeDoc_2(ref) = 0.5$. Pour obtenir une valeur de jugement unique *JugeDoc*(*ref*) pour *ref*, on effectue la moyenne de ces valeurs :

$$JugeDoc(ref) = ½\ [JugeDoc_1(ref) + JugeDoc_2(ref)] = ½\ [1+0.5] = 0.75$$

5.3.1.4. Sélection (Nombre de Seuil-Limite)

Après avoir assigné une valeur à chaque élément mis en commun. Nous nous intéressons à l'ensemble QEL_{tc} de *QNbLimite* requêtes ayant obtenues les plus grandes valeurs d'importance ou à l'ensemble REL_{tc} des documents ayant obtenus les valeurs d'importance les plus élevées où *RNbLimite* est le nombre de documents sélectionnés.

$|QEL_{tc}| \leq QNbLimite$
$|REL_{tc}| \leq RnbLimite$

5.4. Présentation Collaborative de l'Aide selon le Type de Soutien/ Réalisation des Différents Types de Soutien

Après avoir sélectionné un ensemble de requêtes ou de documents, on doit les utiliser afin d'établir le soutien personnalisé selon le type demandé. Nous avons noté que l'utilisateur peut choisir dans sa demande un des types de soutien. Dans la suite, nous étudions comment obtenir chacun de ces types : la présentation des requêtes/résultats (paragraphe 5.4.1), la fusion de requêtes (paragraphe 5.4.2), et le bouclage de pertinence collaborative (paragraphe 5.4.3).

5.4.1. *Présentation de Requêtes ou de Résultats*

La présentation collaborative de requêtes ou de résultats se fait en ordonnant les requêtes ou les documents extraits précédemment dans l'ensemble QEL_{tc} ou REL_{tc}. Pour cela, nous associons une valeur de rangement à chacune des requêtes ou à chacun des documents de façon à établir un ordonnancement.

Dans l'étape *calcul* de la méthode d'extraction (voir le paragraphe 5.3.1.3), nous avons calculé une valeur pour chacune des requêtes ou des documents mis en commun. Nous utilisons à nouveau cette valeur comme valeur de rangement des requêtes ou des documents.

Alors, la présentation collaborative de requêtes ou de résultats, relative aux critères de soutien précisés par l'utilisateur, est une liste de requêtes ou de documents dans laquelle la valeur de rangement *Range* d'une requête ou d'un document est égale à la valeur de la combinaison linaire des valeurs associées à cette requête ou à ce document relativement aux critères de soutien.

$Range\ (Etape\text{-}R.q_k) = L(Etape\text{-}R.q_k) \wedge Etape\text{-}R.q_k \in QEL_{tc}$
$Range\ (Etape\text{-}R.Doc) = L(Etape\text{-}R.Doc) \wedge Etape\text{-}R.Doc \in REL_{tc}$

Ainsi, une requête ou un document qui a une valeur élevée apparaît en tête de la liste, et une requête ou un document qui a une valeur basse apparaît en queue de la liste.

La présentation des résultats (ou la fusion de résultats), que nous proposons ici, prend en compte les souhaits de l'utilisateur. Elle peut prendre en compte plusieurs aspects à la fois selon le(s) critère(s) de personnalisation précisé(s) par l'utilisateur. De plus sont pris en compte non seulement les outils et les collections de recherche (critère contexte matériel de

recherche) mais aussi d'autres critères comme la similarité de requêtes, les autres utilisateurs à travers leurs jugements et leurs degrés de préférences, …etc. Le Tableau 5.3 suivant compare de façon synthétique la fusion des résultats décrite dans la littérature (voir le CHAPITRE.3) et celle que nous proposons.

Points de Comparaison	Littérature de Fusion	Fusion de Résultats Proposés
nombre d'utilisateurs considérés	individuelle, un seul utilisateur	collaborative, plusieurs utilisateurs
personnalisation	non-personnalisée	personnalisée
nombre de critères de fusion	un seul critère	un ou plusieurs critères
critères de fusion	• outils différents. • collections différentes. • formulations différentes de requête.	• outils différents/identiques. • collections différentes/identiques. • similarité de requêtes. • fréquence. • jugement. • préférence.

Tableau 5.3. *Synthèse de fusion de résultats dans la littérature et proposée.*

5.4.2. *Fusion de Requêtes*

A partir de l'ensemble QEL_{tc} de requêtes sélectionnées, nous construisons une seule requête en appliquant une opération de fusion sur les requêtes de QEL_{tc}. Dans cette étude, Nous voulons que la fusion soit la plus exhaustive possible. Afin d'établir cette fusion selon la structure de la requête nous envisagions la fusion de requêtes vectorielles (paragraphe 5.4.2.1) la fusion de requêtes booléennes (paragraphe 5.4.2.2) et la fusion mixte des requêtes vectorielles et booléennes (paragraphe 5.4.2.3). Une discussion sur l'intérêt de la fusion des requêtes est présentée dans le paragraphe 5.4.2.4.

5.4.2.1. Fusion des Requêtes Vectorielles

Nous définissons l'opération de fusion comme la somme de tous les vecteurs des requêtes extraites dans QEL_{tc} :

$$q_{fusion} = \frac{1}{|QEL_{tc}|} \sum_{qi \in QELtc} q_i \qquad [5.3]$$

où $\sum$ représente la somme des vecteurs. Ainsi la requête q_{fusion} contient tous les termes Tq_j qui apparaissent dans chacune des requêtes $q_j \in QEL_{tc}$.

5.4.2.2. Fusion des Requêtes Booléennes

La fusion des requêtes booléennes est plus difficile parce qu'il faut tenir compte de leurs structures logiques. Supposons que les termes qui apparaissent dans toutes les requête q_u

appartiennent à l'ensemble ($\cup_{qj \in QELtc}\ Tq_j$), cet ensemble de disjonction contient deux sous-ensembles de termes :

- le premier $(\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD}}$: est un sous-ensemble de termes à propos duquel les requêtes sont en accord. A son tour ce sous ensemble est devisé en deux sous-ensembles :
  - $(\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD-ET/OU}}$: un sous-ensemble des termes désirés (liés par ET/OU) dans le résultat de toutes les requêtes. Les termes t_d de ce sous-ensemble sont inclus et liés par OU dans la requête fusionnée.
  - $(\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD-NON}}$: un sous-ensemble des termes non désirés (qui sont lié par NON dans les requêtes) dans le résultat de toutes les requêtes. Les termes t_{nd} de ce sous-ensemble sont inclus et liés par NON dans la requête fusionnée.
- le deuxième $(\cup_{qj \in QELtc}\ Tq_j)^{\text{DESACCORD}}$: est un sous-ensemble de termes désirés (liés par ET/OU) dans certaine(s) requête(s) et non désirés (liés par NON) dans d'autre(s), c'est-à-dire les requêtes sont en conflit ou en désaccord à propos de ce sous-ensemble :

  $$(\cup_{qj \in QELtc}\ Tq_j)^{\text{DESACCORD}} = [\cup_{qj \in QELtc}\ (Tq_j)^{\text{ET/OU}}] \cap [\cup_{qj \in QELtc}\ (Tq_j)^{\text{NON}}]$$

  Où $[\cup_{qj \in QELtc}\ (Tq_j)^{\text{ET/OU}}]$ contient les termes liés par ET/OU par n'importe quelle requête de l'ensemble QEL_{tc}. $[\cup_{qj \in QELtc}\ (Tq_j)^{\text{NON}}]$ contient tous les termes liés par NON par n'importe quelle requête de l'ensemble QEL_{tc}.

  Les termes de ce sous-ensemble $(\cup_{qj \in QELtc}\ Tq_j)^{\text{DESACCORD}}$ sont omis dans la requête fusionnée.

Alors, la requête fusionnée q_{fusion} et son ensemble de termes Tq_{fusion} seront :

$$q_{fusion} = (\text{NON } t_{nd1} \text{ NON } t_{nd2} \text{ NON } \ldots) \text{ OU } (t_{d1} \text{ OU } t_{d2} \text{ OU } \ldots) \qquad [5.4]$$

où : $t_{nd} \in (\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD-NON}}$, $t_d \in (\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD-ET/OU}}$

$Tq_{fusion} = (\cup_{qj \in QELtc}\ Tq_j)^{\text{ACCORD}}$

5.4.2.3. Fusion des Requêtes Vectorielles et Booléennes

Dans le cas des requêtes vectorielles booléennes à fusionner, nous pouvons procéder de manière semblable au paragraphe 5.2.3.1.3 en transformant les requêtes afin d'obtenir des représentations homogènes de toutes les requêtes que ce soient vectorielles ou booléennes, puis nous pouvons appliquer la formule [5.3] ou [5.4] correspondant à la représentation uniforme des requêtes à fusionner.

5.4.2.4. Discussion

Nous avons choisi l'opération de fusion des requêtes comme un soutien collaboratif pour les raisons suivantes :

- La fusion des requêtes améliore la performance de recherche : l'étude présentée dans [Belkin 1993] (voir paragraphe 3.1) prouve que les combinaisons de différentes expressions du même besoin d'information ont, en général un effet positif sur la

performance de recherche[4]. Nous pensons que la fusion des requêtes de plusieurs utilisateurs peut être considérée comme une fusion de différentes expressions du même besoin d'information. Les données supplémentaires dont nous disposons (l'évaluation de la requête, la préférence, et la similarité …etc) sont utilisées comme des poids « optimaux » pour fusionner les différentes requêtes, et permettent de repérer les requêtes susceptibles d'optimiser la fusion, (par exemple les requêtes ayant obtenu le meilleur résultat, ou les requêtes d'un utilisateur compétent).

- La fusion des requêtes tire profit de la variété des termes utilisés par les utilisateurs : les auteurs dans [Xu 2000] insistent sur le problème de la « non-correspondance » de termes (word mismatch) comme un problème fondamental en recherche d'information, tel qu'il a été observé par [Furnas 1987] où dans moins de 20% des cas, deux personnes utilisent le même terme pour décrire un même but.

  Une autre étude de [Vakkari 2001] démontre que l'exécution de la tâche de recherche est liée systématiquement aux tactiques de recherche et aux choix des termes utilisés pendant cette exécution, l'auteur pense que l'utilisateur a besoin d'aide[5]. Les résultats de son expérimentation suggèrent qu'une interface qui recommande aux utilisateurs d'introduire dans les requêtes des termes parallèles, reliés, synonymes, et spécifiques, peut les aider à trouver plus de références pertinentes.

  En se basant sur ces études, nous pensons que la fusion des requêtes introduit des termes utilisés par les différents utilisateurs (termes synonymes, variété morphologique, …).

- La fusion élargie la requête en utilisant les connaissances des utilisateurs : en fait la fusion peut être vue comme une sorte d'expansion de la requête. L'expansion de requête est une approche qui a déjà été étudiée via l'utilisation d'un thesaurus où l'on élargit une requête à partir de la connaissance contenue dans un thesaurus. [Bruandet 2003] s'est intéressé à l'utilisation et la construction de bases de connaissances pour la recherche d'information :

  « Les techniques d'expansion automatique de requêtes ont pour objectif de décharger l'utilisateur de l'effort cognitif qu'il doit fournir, que ce soit pour le choix des documents pertinents nécessaires pour le bouclage de pertinence ou le choix des termes de sa requête dans un thésaurus manuel en ligne. Les techniques d'expansion se veulent entièrement automatiques. Le problème est alors le choix des termes à ajouter à la requête. Les méthodes diffèrent selon le choix de ces termes, de la méthode de construction des thésaurus et/ou de leur provenance. »

  Dans notre cas, au lieu d'utiliser un thesaurus, la fusion des requêtes a à sa disposition la mémoire collaborative et nous pouvons élargir une requête à partir des différentes connaissances des utilisateurs exprimées au travers de leurs requêtes. Ainsi, une telle expansion utilise les connaissances des spécialistes du domaine éventuellement non encore répertoriées dans un thesaurus.

---

[4] « *Results show that progressive combination of query formulations leads to progressively improving retrieval performance*. »

[5] « *In additional to support in finding appropriate vocabulary, user also need help in formulating the query.* »

5.4.3. *Bouclage de Pertinence Collaborative*

Le mécanisme de bouclage de pertinence suppose que la requête q_u de l'utilisateur u fournit un ensemble de documents que l'utilisateur évalue R_{qu}. De nouvelles requêtes sont ensuite générées à partir de ces jugements en ajoutant aux termes de la requête initiale des termes extraits de documents sélectionnés *Rjugé-per*$_u$, on appelle cela l'expansion de requête en fonction du contexte, dans laquelle on prend en compte le jugement de l'utilisateur.

Ce mécanisme est une approche « individuelle » de la fonction de bouclage de pertinence où la requête q_u et l'ensemble *Rjugé-per*$_u$ des documents jugés pertinents par un seul utilisateur u sont pris en compte. On peut également remarquer que cette fonction ne prend en compte les documents que selon un seul aspect, celui de leur jugement de pertinence. Cependant, nous pensons que le fait de prendre en compte les documents sélectionnés selon d'autres aspects que la pertinence peut aussi être intéressante.

Il y a trois manières d'introduire notre approche *collaborative* dans le bouclage de pertinence :

1. Introduire l'ensemble de requêtes QEL_{tc} : la fusion de requêtes que nous avons proposée dans la section précédente prend en compte la collaboration et la personnalisation. Au lieu de prendre en compte la requête de l'utilisateur, on prend en compte et on fusionne les requêtes de l'ensemble QEL_{tc} pour obtenir une nouvelle requête. On peut utiliser la formule [5.3] ou [5.4] (fusion de requêtes) pour proposer une nouvelle requête en tenant compte des requêtes ayant obtenu une meilleure évaluation *EvalEtape*$_j$, donc plus efficaces.
2. Introduire l'ensemble de documents REL_{tc} : qui remplace l'ensemble *Rjugé-per*$_u$ du mécanisme de bouclage de pertinence « individuelle » et le personnalise.
3. Introduire les ensembles de requêtes REL_{tc} et des documents REL_{tc} : la reformulation de la requête pour l'utilisateur u prend en compte à la fois des requêtes et des documents des autres utilisateurs selon plusieurs aspects. Où les ensembles QEL_{tc} et REL_{tc} sont pris en compte dans l'ensemble des requêtes et des documents que la fonction de bouclage de pertinence utilise.

La mise en oeuvre du mécanisme de bouclage de pertinence dans les modèles vectoriel et booléen ne présente pas le même degré de difficulté. Puisque ce mécanisme prend essentiellement en compte les documents et leurs jugements. Alors, notre considération au début de ce chapitre que les documents sont des ensembles de termes de leur titre, interviennent dans la mise en ouvre du bouclage de pertinence pour chacun des modèles. Ainsi, notre approche collaborative du bouclage de pertinence est présentée pour le modèle vectoriel (paragraphe 5.4.3.1) pour le modèle booléen (paragraphe 5.4.3.2). En ce qui concerne le bouclage de pertinence collaborative pour des requêtes vectorielles et booléennes, nous précédons de la même manière que pour la fusion des requêtes de modèles différents en transformant le tout à une seule représentation puis nous appliquons le bouclage de pertinence collaborative selon la représentation unique.

5.4.3.1. Bouclage de Pertinence Collaborative pour le Modèle Vectoriel

Nous présentons tout d'abord une méthode de bouclage de pertinence habituelle (paragraphe 5.4.3.1.1), puis l'introduction de notre approche collaborative à cette méthode

(paragraphe 5.4.3.1.2), ensuite, nous présentons et discutons d'une autre approche collaborative déjà étudiée au sein de ce modèle (paragraphe 5.4.3.1.3).

5.4.3.1.1. Bouclage de Pertinence dans le modèle Vectoriel

Plusieurs méthodes de reformulation ont été proposées dans le cadre de ce modèle. La formule suivante de Roccchio [Rocchio 1971] est la définition d'une fonction de reformulation d'un vecteur de requête q_u par un vecteur de requête $q_{reformulée}$:

$$q_{reformulée} = \alpha\, q_u + \beta \frac{1}{|D_p|} \sum_{Doci \in Dp} Doc_i - \gamma \frac{1}{|D_{np}|} \sum_{Doci \in Dnp} Doc_i$$

où D_p est l'ensemble des vecteurs-documents proposés pour q_u et jugés pertinents par l'utilisateur. D_{np} est l'ensemble des vecteurs-documents proposés pour q_u et jugés non pertinents. α, β, $\gamma \in$ [0-1] des facteurs qui permettent d'exprimer les différentes probabilités de fonction de reformulation. $\sum$ représente la somme des vecteurs.

Dans le cas où l'ensemble de document non pertinents D_{np} n'est pas pris en compte ($\gamma = 0$), la requête q_u et l'ensemble de documents pertinents D_p (l'ensemble D_p est l'ensemble $Rjugé\text{-}per_u$ selon notre convention) ont la même importance ($\alpha = \beta = 1$). La formule précédente de fonction de bouclage de pertinence peut être exprimée ainsi :

$$q_{reformulée} = q_u + \frac{1}{|Rjuge\text{-}per_u|} \sum_{Doci \in Rjugé\text{-}peru} Doc_i$$

Nous expliquons dans la suite comment l'approche collaborative intervint dans l'ensemble $Rjugé\text{-}per_u$ de documents Doc_i , et puis dans la requête q_u.

5.4.3.1.2. Approche Collaborative Proposée

La première approche reformule la requête q_u en utilisant l'ensemble des termes $TDoc_i$ qui apparaisse dans les titres de tous les documents Doc_i extraits dans REL_{tc}. Puisque la requête q_u est un vecteur, alors nous présentons cet ensemble de termes des documents sous la forme d'un vecteur dans l'espace des termes où ont un poids non nul les mots qui apparaissent dans les titres des documents :

$(w_{11}, w_{12}, \ldots, w_{ij}, \ldots)$ $\quad w_{ij} = 1$ si $t_{ij} \in \cup_{Doci \in RELtc} TDoc_i$

$\quad w_{ij} = 0$ si $t_{ij} \notin \cup_{Doci \in RELtc} TDoc_i$

La requête $q_{reformulée}$ contient tous les termes qui apparaissent dans la requête Tq_u et dans les titres de document extraits $TDoc_i$:

$$q_{reformulée} = q_u + (w_{11}, w_{12}, \ldots w_{ij}, \ldots) \quad \text{où} \quad w_{ij} \neq 0 \text{ si } t_{ij} \in \cup_{Doci \in RELtc} TDoc_i \qquad [5.5]$$

$Tq_{reformulée} = Tq_u \cup (\cup_{Doci \in RELtc} TDoc_i)$ est l'ensemble des termes de la requête reformulée.

La seconde approche reformule la requête en utilisant non seulement l'ensemble des documents Doc_i extraits dans REL_{tc}. Mais l'ensemble de requêtes extraites dans QEL_{tc} Où la

requête q_{fusion} calculée par la formule [5.3] remplace la requête q_u de la formule [5.5] précédente ainsi :

$$q_{reformulée} = q_{fusion} + (w_{11} , w_{12} , \ldots w_{ij}, \ldots) \text{ où } \quad w_{ij} \neq 0 \text{ si } t_{ij} \in \cup_{Doci \in RELtc} TDoc_i \qquad [5.6]$$

Les termes de la requête sont choisis dans l'ensemble suivant :

$$Tq_{reformulée} = (\cup_{qi \in QELtc} Tq_i) \cup (\cup_{Doci \in RELtc} TDoc_i)$$

5.4.3.1.3. Autre Approche

Une approche semblable s'est intéressée à la collaboration pour l'expansion de requête [Hust 2002a] et [Hust 2002b], dans le cadre du modèle vectoriel, cette approche est un cas particulier de la nôtre dans laquelle les auteurs utilisent dans bouclage de pertinence les documents dont les requêtes sont similaires du point de vue de système c'est-à-dire $REL_{tc} = RSimReq\ (q_u)$ où :

$$QSimReq\ (q_u) = \{q_j : SimReq\ (q_u, q_j) \geq \alpha_{simreq}\ \}$$
$$RSimReq\ (q_u) = \{\ R_{qj} : q_j \in QSimReq\ (q_u)\ \}$$

Nous utilisons nos conventions pour présenter cette méthode, comme la suite :

$$q_{reformulée} = q_u + \sum_{qj \in QSimReq} \lambda_{qj} \frac{\sum_{Doci \in R_{qj}} \overrightarrow{Doc_i}}{\left\| \sum_{Doci \in R_{qj}} \overrightarrow{Doc_i} \right\|} \qquad [5.7]$$

où || . || est la norme Euclidienne d'un vecteur. $QSimReq(q_u)$ l'ensemble des requêtes similaires à la requête q_u, et R_{qj} le résultat de q_j. λ_{qj} est un paramètre permettant de faire varier le poids donné aux différents termes d'expansion. Pour déterminer les coefficients λ_{qj}, les auteurs proposent différentes méthodes :

- QSD (Query Similarity and relevant Documents) :

  $\lambda_{qj} = SimQ(q_u, q_j)$

  où leur fonction de similarité de deux vecteurs de requête est : $SimQ(q_u, q_j) = cos(q_u, q_j)$

- QLD (Query Linear combination and relevant Documents) : cette méthode consiste à :

  a. utiliser les requêtes q_j similaires à la requête q_u pour construire la requête q_u, comme une combinaison linaire des requêtes q_j avec les coefficients σ_{qj}, c'est-à-dire :

  $$q_u = \sum_{qi \in QSimReq(qu)} \sigma_{qj} * q_j \qquad \Leftrightarrow \qquad q_u = Q * \sigma$$

  où Q est une matrice de M lignes (dimension de requête) et de $|QSimReq(q_u)|$ (le nombre de requêtes similaires) colonnes. σ est un vecteur d'une colonne de $|QSimReq(q_u)|$ éléments.

  b. trouver le vecteur $\hat{\sigma}$ qui ajuste au mieux l'équation précédente, leur approche étant de minimiser la norme euclidienne du vecteur $Q * \sigma - q_u$:

  $$\hat{\sigma} = argmin_{\sigma} \| Q * \sigma - q_u \|$$

c. prendre le coefficient λ_{qj} égale à $\hat{\sigma}_{qj}$ si ce dernier est plus grand ou égal à un seuil donné β :

$\lambda_{qj} = \hat{\sigma}_{qj}$ si $|\hat{\sigma}_{qj}| \geq \beta$
$\lambda_{qj} = 0$ si $|\hat{\sigma}_{qj}| < \beta$

Nous remarquons que dans cette approche le problème est de trouver la valeur des coefficients λ_{qj}. Alors dans notre approche, nous n'avons pas ce problème car les valeurs de ces coefficients sont les valeurs des autres critères de soutien. Par exemples λ_{qj} peut être le degré de préférence de l'utilisateur u_j qui a posé la requête q_j ($\lambda_{qj} = Pref(u_j)$), où l'efficacité de requête q_j ($\lambda_{qj} = Eval\text{-}Etape_j$), …et la formule [5.7] devient :

$$q_{reformulée} = q_u + \sum_{qj \in QSimReq} Pref(u_j) \frac{\sum_{Doci \in R_{qj}} Doc_i}{\left\| \sum_{Doci \in R_{qj}} Doc_i \right\|} \quad \text{ou}$$

$$q_{reformulée} = q_u + \sum_{qj \in QSimReq} Eval\text{-}Etape_j \frac{\sum_{Doci \in R_{qj}} Doc_i}{\left\| \sum_{Doci \in R_{qj}} Doc_i \right\|}$$

5.4.3.2. Bouclage de Pertinence Collaborative pour le Modèle Booléen

La première approche introduit la collaboration et les critères de personnalisation dans l'ensemble de documents. La requête reformulée contient en plus de l'expression de la requête de l'utilisateur q_u, tous les termes qui apparaissent dans les titres des documents de REL_{tc} liés par OU. Pour les termes qui apparaissent à la fois dans les titres des documents et dans la requête q_u, ils sont liés comme dans la requête q_u. Alors la nouvelle requête $q_{refourmulée}$:

$$q_{reformulée} = q_u \text{ OU } t_{11} \text{ OU } \ldots \text{ OU } t_{ij} \text{ OU } \ldots \qquad t_{ij} \in [(\cup_{Doci \in RELtc} TDoc_i) | Tq_u] \qquad [5.7]$$

$$Tq_{reformulée} = Tq_u \cup (\cup_{Doci \in RELtc} TDoc_i)$$

La seconde approche, ressemble à la première, mais l'expression de la requête fusionnée q_{fusion} calculée dans la formule [5.4] (fusion des requêtes booléennes) remplace celle de la requête q_u ainsi :

$$q_{reformulée} = q_{fusion} \text{ OU } t_{11} \text{ OU } \ldots \text{ OU } t_{ij} \text{ OU } \ldots \qquad t_{ij} \in [(\cup_{Doci \in RELtc} TDoc_i) | Tq_{fusion}] \qquad [5.8]$$

$$Tq_{reformulée} = Tq_{fusion} \cup (\cup_{Doci \in RELtc} TDoc_i)$$

Donc, la reformulation de la requête est personnalisée selon plusieurs aspects et elle prend en compte la recherche collaborative.

5.5. Conclusion

Nous avons construit un soutien adapté à trois niveaux : selon le degré de collaboration du groupe, selon le contexte de l'utilisateur ces deux niveaux ont été expliqués à travers le CHAPITRE.4 et dans ce chapitre nous avons proposé une procédure afin de construire un soutien personnalisé et adapté aux désirs et souhaits de l'utilisateurs qu'il les exprime à travers des critères de soutien.

Le chapitre suivant est consacré à la description du prototype réalisé sur la base des idées exposées dans ce chapitre et le chapitre précédent.

CHAPITRE.6. Réalisation du Système

Ce chapitre présente la réalisation de notre proposition, où dans le paragraphe 6.1 nous proposons l'architecture de notre système, dans le paragraphe 6.2 l'implantation de cette architecture. Un exemple clarifiant notre approche est présenté dans le paragraphe 6.3, et dans le paragraphe 6.4 nous présentons un résumé de l'évaluation des systèmes de groupware en général et l'évaluation de notre prototype en particulier, nous conclurons ce chapitre par une discussion sur cette évaluation du prototype et quelques pistes d'amélioration et de développement (paragraphe 6.5).

6.1. Architecture Système

Nous avons réalisé un prototype dont l'architecture est présentée dans la Figure 6.1. Elle consiste en trois couches : *interface d'accès*, *noyau de mise en œuvre des fonctions* et *mémoire collaborative*.

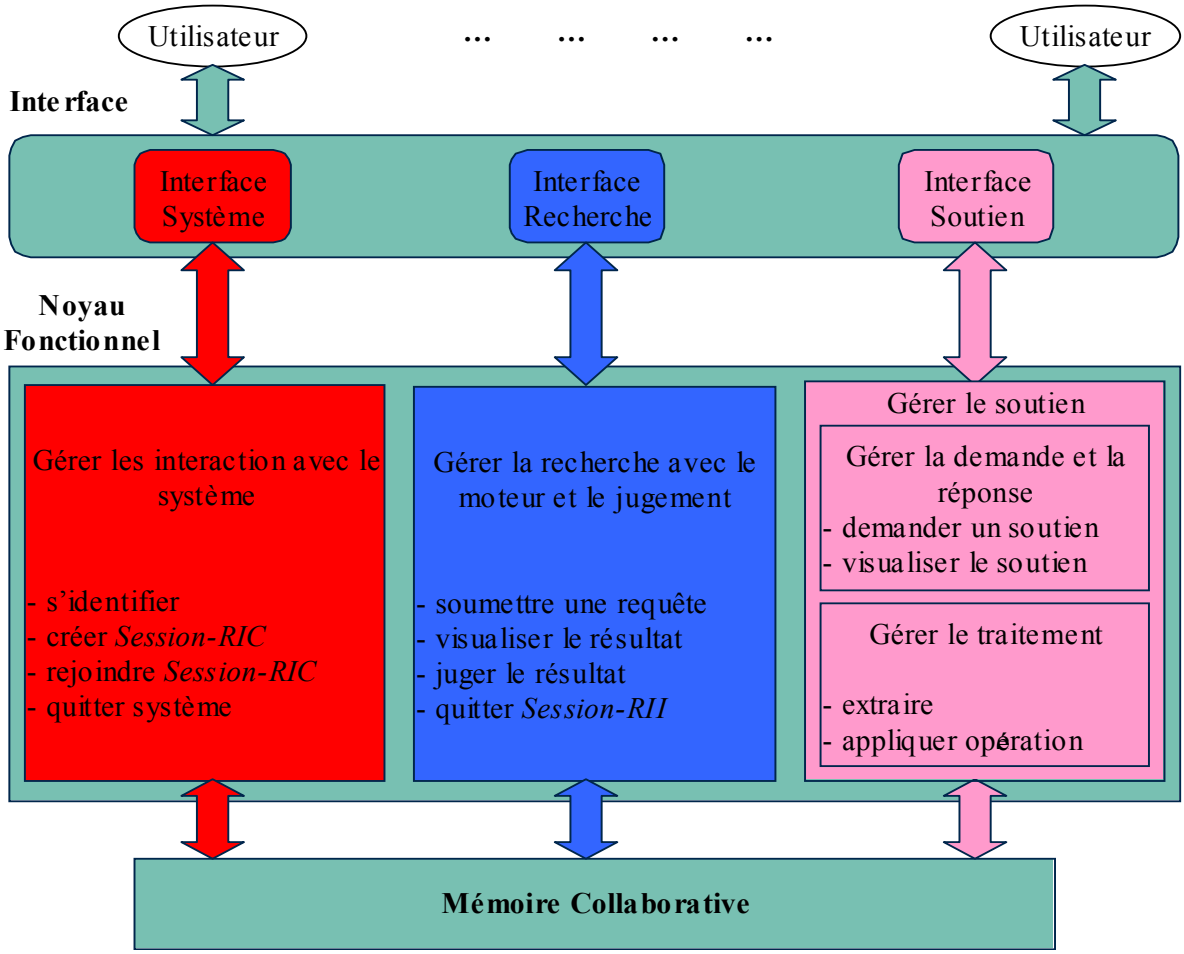

Figure 6.1. *Architecture système.*

Les utilisateurs accèdent aux services via l'*interface* (paragraphe 6.1.1) du *noyau fonctionnel* du système de soutien (paragraphe 6.1.2), qui enregistre toutes les informations dans le *mémoire collaborative* dont le modèle a été présenté dans les paragraphes 4.5.3.3 et 4.1, dans le paragraphe 6.1.3 nous présentons l'architecture système du point de vue collaboratif ou individuel et nous situons notre approche dans les classifications groupware.

6.1.1. *Interface Collaborative*

L'interface consiste en trois groupes : l'*interface système*, l'*interface recherche* et l'*interface de soutien* qui permettent respectivement l'accès aux fonctions des trois unités du noyau fonctionnel qui gèrent les interactions avec le système, le moteur de recherche, et le soutien.

6.1.2. *Noyau Fonctionnel*

Le noyau fonctionnel se situe entre l'interface collaborative et la mémoire collaborative. Il est composé de trois unités qui gèrent respectivement les interactions entre le système et l'interface fournie à l'utilisateur (paragraphe 6.1.2.1), la recherche (paragraphe 6.1.2.2), et, enfin, le soutien (paragraphe 6.1.2.3).

6.1.2.1. Unité Système

Cette unité fournit toutes les fonctions nécessaires à l'organisation du groupe d'utilisateurs et des sessions collaboratives. Les fonctions proposées sont les suivantes :

- *Identification*. Pour utiliser le système l'utilisateur doit s'identifier en précisant son nom, son prénom, et son adresse e-mail.
- *Création d'une session collaborative.* Le démarrage d'une session collaborative *Session-RIC* se fait en fournissant : le titre de la session, le type de recherche (relâchée, coordonnée ou jointe) et, éventuellement, une description de sujet de recherche. Dans le cas d'une recherche jointe, d'autres paramètres sont à définir à savoir les critères de jugement, leur nombre, leurs poids, l'échelle de jugement et l'échelle de préférence.
- *Accès à une session collaborative.* Cette fonction permet à l'utilisateur de joindre une session déjà commencée, il lui suffit de désigner la session qu'il désire rejoindre.
- *Abandon du système*. Cette fonction permet à l'utilisateur de quitter le système et par conséquent toutes les sessions collaboratives auxquelles il participe. Une fois qu'on a quitté le système, on perd tout droit y compris son identification et l'on doit s'identifier à nouveau pour y accéder.

Les deux fonctions qui permettant de créer et de rejoindre une session collaborative créent implicitement une session individuelle *Session-RII* de l'utilisateur dans cette session collaborative. De même, la fonction permettant de quitter le système ou de quitter une session collaborative conduit implicitement à quitter aussi la session individuelle de l'utilisateur dans cette session collborative.

6.1.2.2. Unité Recherche

L'unité de recherche interagit avec des moteurs de recherche sur l'Internet, elle n'intervient pas dans les fonctions internes de ces moteurs de recherche. Cette unité fournit à l'utilisateur le moyen de réaliser les fonctions suivantes :

- *Soumettre une requête*,
- *Visualiser le résultat* rendu par l'outil de recherche en réponse à sa requête,
- *Juger le résultat* retourné par le système selon des critères de jugement déjà précisés par le système ou par le groupe (via l'*administrateur* qui a démarré la session collaborative) pour une recherche jointe,
- *Quitter la session individuelle* et par conséquent quitter la session collaborative.

Si un utilisateur quitte une session individuelle appartenant à une session collaborative, il peut participer à nouveau à cette session collaborative au moyen d'une des deux fonctions *rejoindre une session collaborative* (au cas où la session quittée serait encore en cours de déroulement) et *créer une session collaborative* (au cas où la session quittée serait déjà terminée).

6.1.2.3. Unité Soutien

Cette unité comporte deux parties. D'une part, celle qui gère l'extraction des requêtes et/ou des documents de la mémoire collaborative et leur applique une opération pour réaliser le soutien demandé (voir les paragraphes 5.3 et 5.4). D'autre part, celle qui gère la demande de soutien effectuée par l'utilisateur et la visualisation du résultat en réponse à cette demande ainsi :

- *Demander un soutien* : l'utilisateur demande un soutien au cours de sa recherche en précisant : le type de soutien, la dimension temporelle, les critères de soutien et éventuellement les valeurs de préférence pour les autres utilisateurs.
- *Visualiser le soutien* : fournit à l'utilisateur le moyen de visualiser les résultats du soutien personnalisé demandé. L'utilisateur a ensuite le choix entre se servir du soutien obtenu pour sa recherche si le résultat du soutien demandé lui semble intéressant ou ne pas en tenir compte.

6.1.3. *Architecture du Système du Point de Vue Collaboratif ou Individuel*

L'architecture du système a été analysée, dans ce qui précède, du point de vue de son fonctionnement par couche. Mais elle peut l'être selon que l'on s'intéresse à l'aspect individuel (mono-utilisateur) ou collaboratif (multi-utilisateur). C'est à ce second point de vue que nous allons nous intéresser maintenant.

Donc, le système peut être décomposé en deux parties, la première concerne la recherche d'information individuelle classique. Elle comporte l'*interface recherche* et l'*unité recherche*. La deuxième concerne notre système comme outil spécialisé de collaboration. Elle comporte l'*interface système*, l'*unité système*, l'*interface soutien* et l'*unité soutien*. La mémoire collaborative peut être vue comme un élément intermédiaire entre les deux parties. Ainsi, la partie mono-utilisateur stocke toutes les informations concernant chaque recherche

individuelle dans la mémoire, et la partie multi-utilisateurs utilise cette mémoire pour fournir le soutien demandé.

En nous basant sur les descriptions des situations de travail et les fonctions de groupware dans le paragraphe 1.1 nous déterminons la position de notre approche selon ces situations et ces fonctions :

- Situations de Travail : le Tableau 6.1 résume les différentes classifications de notre approche selon :
  1. *Lieu et Temps* : cette classification nous permet de situer le contexte de notre travail dans les deux dimensions spatiale et temporelle : la dimension spatiale est « à distance », où nous considérons que les utilisateurs travaillent séparément dans des endroits différents, et la dimension temporelle est « synchrone », « asynchrone », nous considérons que les utilisateurs travaillent dans un contexte « synchrone », car ils travaillent en même temps, chacun sur son outil de recherche, et ils consultent pendant le temps de leur travail le système qui les coordonne.

     On s'appuie ici sur la transition synchrone-asynchrone. Un aspect important de notre système à long terme est que la base collective pourra être utilisée non seulement de façon synchrone, mais aussi pour des applications asynchrones comme une mémoire de groupe et/ou comme une base de recommandation.

  2. *Durée de vie et Taille de groupe* : la taille du groupe est restreinte et les membres travaillent ensemble pour une durée restreinte le temps de la session de recherche collaborative.
  3. *Types de résultat du travail effectué* : le travail du groupe débouche sur l'exécution d'une mission, qui est la tâche de recherche.

Classifications	Approche Proposée
Lieu et Temps	• local synchrone. • local asynchrone. • à distance synchrone. • à distance asynchrone.
Durée de vie et Taille de groupe	• temporaire restreint
Types de résultat du travail effectué	• exécution d'une recherche sur un sujet commun

Tableau 6.1. *Résumé des classifications des situations de travail de l'approche proposée.*

- Fonctionnalités du Groupware : qui nous permettent de situer les objectifs de notre système. Le Tableau 6.2 résume les différentes fonctionnalités de notre approche suivantes :
  1. communication interpersonnelle : est indirecte, car elle se passe par l'intermédiaire de la demande d'un soutien particulier à partir des recherches des autres.
  2. coordination : qui est différente selon le type de groupe. Dans les trois types il y a un rôle d'utilisateur qui démarre la session *administrateur* et les autres qui le joignent

membres. Pour les deux types de recherche relâchée et cordonnée, il n'y a pas de distribution et de coordination des tâches tandis que pour la recherche jointe les utilisateurs se cordonnent en étapes parallèles ou complémentaires et peuvent préciser de temps d'exécution pour chaque tâche. Cette coordination n'est pas gérée, elle peut être faite par e-mail, ou par d'autres moyens.

3. collaboration : les rôles des utilisateurs ne sont pas hiérarchiquement définis.
4. mémoire du groupe : la recherche effectuée en commun alimentent une mémoire collaborative du groupe.
5. circulation de l'information : le système permet la circulation des informations individuelles (les requêtes, les références et les jugements, etc.…) entre les utilisateurs. La circulation de l'information n'est pas systématique, et elle intervient quand l'utilisateur la demande.

Fonctionnalités	Détailles des Fonctionnalités pour l'Approche Proposée
Communication interpersonnelle	• indirecte (plusieurs à plusieurs).
Coordination des activités	• pour tous les types de groupe : définir des rôles (administrateur, membre) pour tout les type de groupe. • pour le groupe de recherche jointe : prévoir des dates d'achèvement des tâches, assigner les tâches à des personnes, définir un séquencement des tâches.
Collaboration	• les utilisateurs combinent leurs efforts.
Mémoire du groupe	• les informations organisationnelles (paramètres de session collaborative, comme le sujet de recherche, ...), • les informations sociales (préférences), • l'expression de recherche collective effectuée (requêtes, résultats, jugements…)
Circulation de l'information	• les requêtes, • les documents, • notifications (jugements, préférences)

Tableau 6.2. *Résumé des fonctionnalités de l'approche proposée.*

6.2. Prototype

Nous décrirons tout d'abord la restriction par rapport au modèle du soutien théorique que nous avons proposé dans les deux chapitres précédents (CHAPITRE.4 et CHAPITRE.5) (paragraphe 6.2.2), puis nous présentons une description technique du prototype avec les modules que nous avons développés et ceux que nous avons réutilisés (paragraphe 6.2.2).

6.2.1. *Parties Implémentées /Restriction du Modèle Théorique*

L'hypothèse de la diversité des outils et des collections de recherche a des implications importantes sur l'implantation. En effet, il faut traiter deux aspects :

D'une part, en ce qui concerne le document, plus précisément :

- La *disponibilité* (ou l'*accessibilité*) : il est nécessaire que tous les utilisateurs puissent visualiser et consulter les documents retrouvés par les autres.
- l'*unité d'identification* : il est indispensable de s'apercevoir que deux ou plusieurs documents sont identiques même s'ils ont été retrouvés par des outils différents ou à partir de deux collections différentes. Donc, on impose à des documents identiques d'avoir le même identifiant.

D'autre part, en ce qui concerne la *normalisation* des requêtes et des documents. Les requêtes et les documents de différents outils et collections de recherche doivent être rendus conformes à une norme unique pour tous, afin de garantir la persistance et l'intégrité de la mise en commun dans la mémoire collaborative et du traitement de ces requêtes et de ces documents.

Deux solutions sont possibles pour résoudre ces aspects :

- On maintient l'hypothèse de diversité des outils et des collections de recherche, et l'on suppose qu'on dispose de l'outil nécessaire qui garanti d'un coté, la *disponibilité* et l'*unité d'identification* des documents obtenus, et de l'autre coté, la représentation des requêtes et des documents à l'aide d'une norme unique.
- On suppose que les utilisateurs travaillent sur le même type de système de recherche d'informations et sur les mêmes collections, alors on n'a aucun problème de normalisation.

Nous avons choisi la dernière solution plus simple à réaliser. Donc un seul moteur de recherche (« Google ») [Brin 1998] a été mis à la disposition des utilisateurs. Nous indiquons aux utilisateurs que le prototype interagit avec ce moteur de recherche, alors ils n'ont pas besoin d'apprendre une nouvelle syntaxe pour formuler la requête puisqu'ils sont déjà habitués à l'utiliser.

Nous avons réalisé un prototype, qui permet à l'utilisateur de participer simultanément à plusieurs sessions collaboratives et d'être soutenu de façon *collaborative synchrone* pour les trois types de recherche du groupe (relâchée, coordonnée et jointe), nous avons implanté aussi la *présentation* et la *fusion de requêtes* la *présentation de résultats* par rapport à la dimension de temps dans la session *courant* et *historique*, et selon les critères de personnalisation suivants : le *jugement* pour la valeur *jugé-per*, *préférence*, *similarité de requêtes* où nous avons préféré décliner ce critère de *similarité de requêtes* en deux critères : celui de la *similarité de requêtes* (similarité de requêtes du point de vue du système), et celui de la *similarité de résultats* (similarité de requêtes du point de vue sémantique) et ceci pour les deux valeurs *similaire* et *dissimilaire*. Cependant, dans l'état actuel du prototype, l'utilisateur peut choisir le soutien selon les requêtes similaires dont les résultats sont différents ou les requêtes différentes dont les résultats sont similaires. Ces deux cas envisagés ne présentent pas souvent d'intérêt dans le contexte de l'expérimentation.

Le prototype permet de soumettre des requêtes en langue anglaise. Les requêtes sont considérées comme des ensembles de termes, nous ne prenons en compte ni la structure de la requête, ni sa représentation. Par exemple la requête q_j sera considérée comme l'ensemble de termes Tq_j, un traitement linguistique ou une sorte de lemmatisation a été appliquée sur les termes de requête, ainsi les termes ont été traités en supprimant les préfixes, suffixes selon les

règles dans le Tableau 1 de l'annexe B. Puisque nous considérons la requête comme un ensemble de termes, nous avons appliqué le modèle ensembliste et les opérations ensemblistes (intersection, disjonction, …) pour :

- calculer la similarité $SimReq(q_u, q_j)$ du point de vue du système entre la requête q_u de l'utilisateur u et une autre requête q_j ainsi :

$$SimReq\ (q_u, q_j) = \frac{|T_{q_u} \cap T_{q_j}|}{|T_{q_u} \cup T_{q_j}|}$$

- fusionner les requêtes q_j extraites dans QEL_{tc} en utilisant une disjonction entre leurs ensembles de termes Tq_j :

$$Tq_{fusion} = \cup_{qj \in QELtc}\ Tq_j$$

L'évaluation de la requête par rapport au sujet de recherche est implantée (*Eval-Etape* = *EvalPrec-Etape*) tandis que son évaluation par rapport à la pertinence système *EvalOrd-Etape* n'a pas été implémentée.

Nous n'avons pas implanté la stratégie de soutien selon la typologie de l'utilisateur ainsi l'évaluation de l'efficacité de l'utilisateur est prise en terme de sa compétence dans le domaine c'est-à-dire la formule qui permet d'évaluer la session individuelle *Eval-RII* est celle de l'évaluation par rapport au sujet de recherche (*Eval-RII* = *EvalPrec-RII*) et donc l'évaluation par rapport à la pertinence système *EvalOrd-RII* n'a pas été implantée.

6.2.2. *Implémentation Technique*

Le système est maintenu sur un serveur de la machine mrim4. L'accès au système se fait au moyen d'une connexion Internet via un navigateur Web standard à l'adresse http://mrim4:8080/CollaborativeResearch/html/Interface.html.

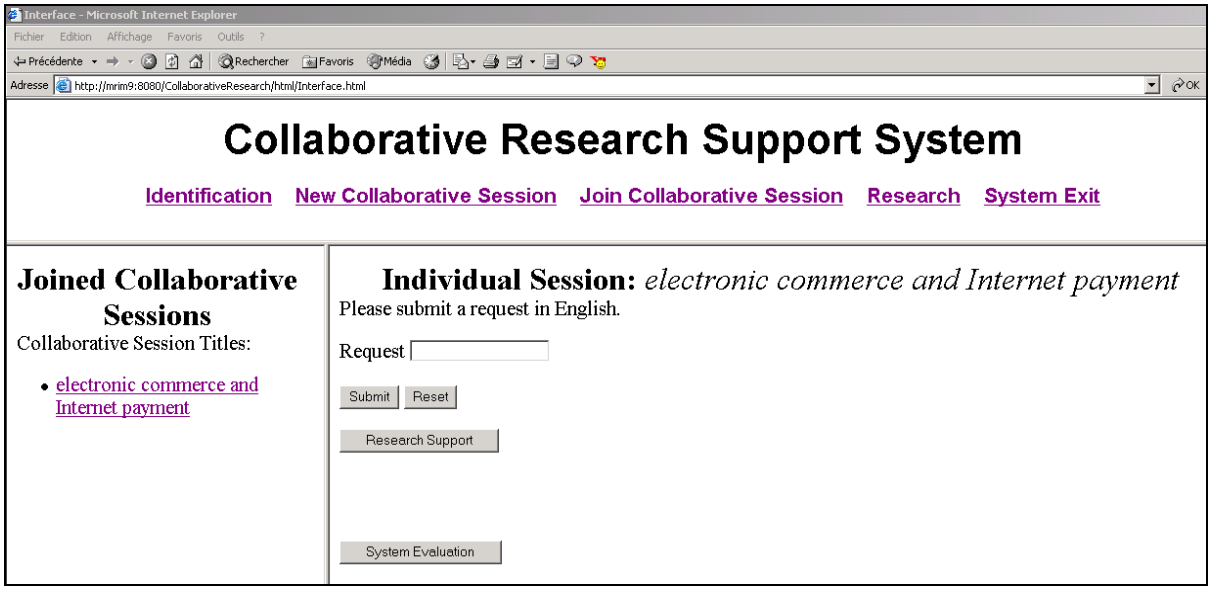

Figure 6.2. *Un exemple de l'interface de notre système.*

Nous avons implanté l'interface en utilisant (HTML), et le JSP (*JavaServer Pages*) [Bergsten 2001] pour son contenu dynamique, le noyau en utilisant le langage orienté objet Java. Nous avons utilisé Eclipse (www.eclipse.org), un environnement de développement Java *open-source* et extensible, et la mémoire collaborative[1] en utilisant la mémoire.

Un serveur Tomcat est utilisé pour gérer l'exécution de logiciel pour le compte de plusieurs utilisateurs. Le serveur Apache Tomcat, développé par le projet Apache Jakarta (http://jakarta.apache.org), Tomcat est un serveur web totalement écrit en Java supportant la spécification des servlets 2.2 et des JSP 1.1. pour l'utiliser, un environnement d'exécution Java doit être installé sur la machine. JSDK (*Java Software Development Kit*) pour windows (http://java.sun.com/j2se).

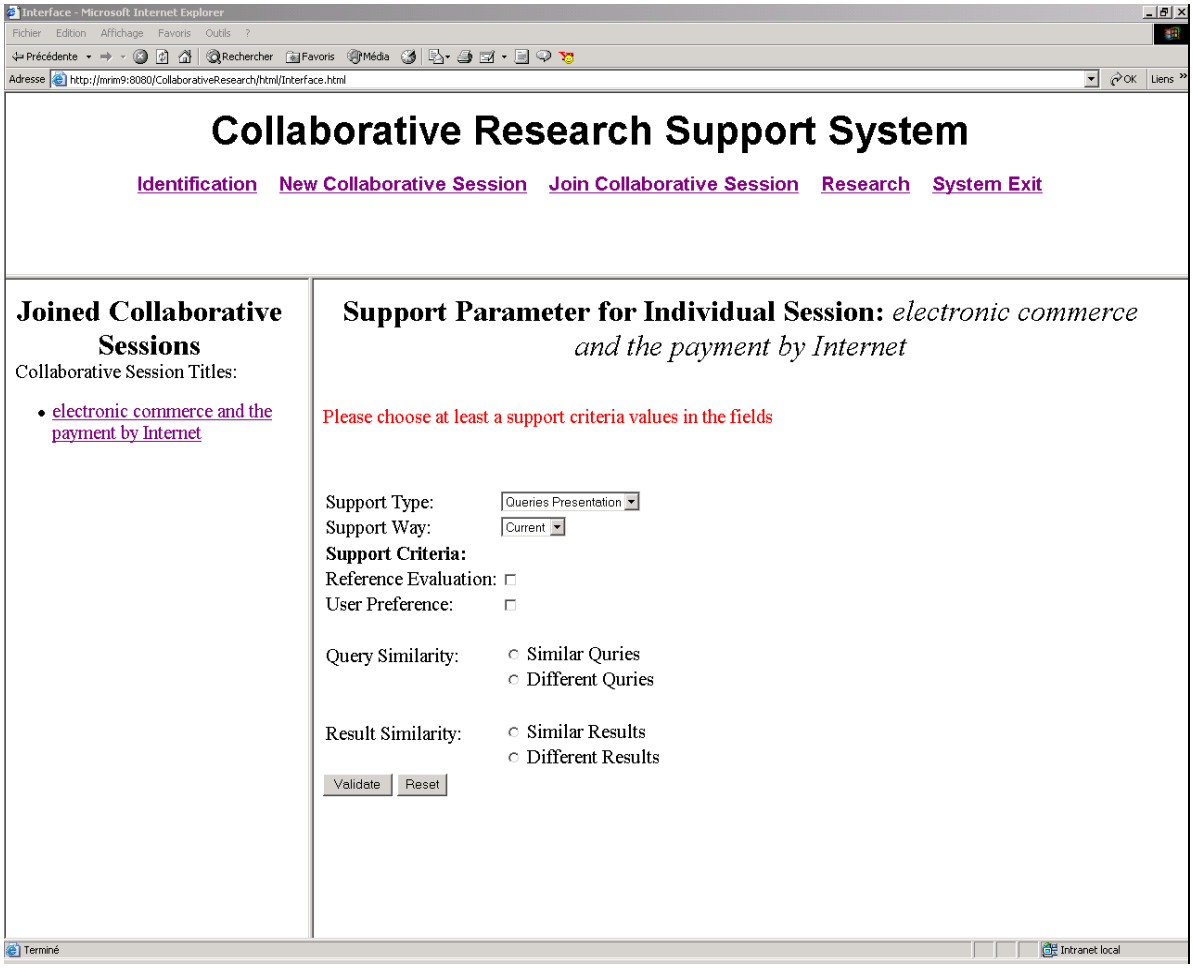

Figure 6.3. *Demande de soutien dans l'interface soutien.*

Si nous prenons l'architecture système de la Figure 6.1. Nous avons réalisé la couche *interface d'accès* avec ces trois groupes d'*interface système*, d'*interface recherche* et d'*interface soutien*, la Figure 6.2 et la Figure 6.3 montrent des exemples de l'interface. En ce qui concerne le noyau fonctionnel, nous avons développé l'unité système, et unité soutien, pour le traitement linguistique des requêtes ainsi que l'unité recherche qui interagit avec le

[1] La mémoire collaborative pourrait être réalisée en utilisant une base de données.

moteur de recherche google, nous avons réutilisé les composants techniques, développé par Bottraud [Bottraud 2003] dans le system AIRA (Adaptive Information Research Assistant).

Après cette explication sur l'architecture système et son implémentation, nous clarifions notre système avec l'exemple suivant.

6.3. Exemple

L'objectif de cet exemple est d'illustrer l'application de l'approche de soutien personnalisé sur la tâche de recherche.

Considérons le scénario suivant, où trois personnes u_1, u_2 et u_3, de la même équipe de travail, effectuent une recherche d'information jointe pour écrire un article sur « le commerce électronique et le payement par Internet ». Le Tableau 6.3 résume la dernière étape de la recherche de u_1, u_2 et u_3 dans la session collaborative, et la prise en compte des cinq premiers documents.

L'utilisateur u_1 est un débutant dans le domaine de recherche, après avoir soumis deux requêtes, qui contiennent des mots clés pertinents au thème du besoin d'information, il n'a pas obtenu des informations vraiment pertinentes, en fait, il a besoin d'apprendre dans le domaine de recherche. Maintenant, si cet utilisateur demande un conseil ou une aide au système de soutien, il peut la demander ainsi : « je voudrais avoir une présentation des résultats obtenus dernièrement par mes utilisateurs préférés et qui ont obtenu les meilleurs jugements ».

L'utilisateur u_1 formule alors sa demande ainsi :

- type de soutien : *présentation des résultats*.
- dimension temporelle : *courant*.
- critères de personnalisation :
  - jugement pour la valeur jugé pertinent, *c-juge = jugé-per*.
  - préférence *c-pref*, il donne ses préférences par rapport aux autres utilisateurs[2] $Pref(u_i)$, ces valeurs de préférence sont : $Pref(u_1) = 0$, $Pref(u_2) = 1$, $Pref(u_3) = 0.5$.

Afin d'illustrer plusieurs types de soutien selon les mêmes critères de personnalisation, nous allons expliquer non seulement le soutien demandé (la présentation des résultats) mais aussi d'autres types à savoir la présentation des requêtes, et la fusion des requêtes. Alors dès la procédure d'extraction nous sélectionnons deux ensembles des requêtes QEL_{tc} et des documents REL_{tc}. Le soutien fonctionne comme indiqué dans la suite :

[2] Puisque la recherche est jointe et les utilisateurs se connaissent bien, alors la préférence est donnée totalement par l'utilisateur u_1 : $Pref(u_i) = Pref_{u1}(u_i)$.

Usager u_i	Requête q_i	Résultat Rq_i Document $Doc_{i,j}$ (le titre de document et son URL)	Jugement de document *JugeDoc*	Evaluation de requête $Eval\text{-}Etape_i$
u_1	q_1 : Electronic Payment	$Doc_{1,1}$: Electronic Commerce http://www.semper.org/sirene/outsideworld/ecommerce.html	0.5	0.1
		$Doc_{1,2}$: www.mastercard.com/Sepp/sepptoc.htm http://www.mastercard.com/Sepp/sepptoc.htm	0	
		$Doc_{1,3}$: CheckFree - The #1 Way to Pay Online http://www.checkfree.com/	0	
		$Doc_{1,4}$: Electronic payment processing from CyberSource http://www.cybersource.com/	0	
		$Doc_{1,5}$: Welcome to EFTPS http://www.eftps.gov/	0	
u_2	q_2 : e-business solutions online catalogue	$Doc_{2,1}$: Suppliers Directory - E-Commerce Focus - Internet One-Stop http://www.ecomfocus.com/cgi-bin/directory/list.asp?sector=28	1	0.8
		$Doc_{2,2}$: in2info limited - ebusiness solutions http://www.in2info.co.uk/starter/	1	
		$Doc_{2,3}$: Clear IT - eBusiness Solutions - Web Development http://www.clearit.com.au/eBusiness/WebDevelopment.htm	1	
		$Doc_{2,4}$: Seagrass - eBusiness strategy - eBusiness software solutions http://www.seagrasssoftware.com/77.htm	1	
		$Doc_{2,5}$: First Internet Marketing is an e-business solutions company http://www.firstinternet.co.uk/showitem.asp?i=19	0	
u_3	q_3 : Internet payment e-trade	$Doc_{3,1}$: ePSO Newsletter http://epso.jrc.es/newsletter/vol06/5.html	0.5	0.4
		$Doc_{3,2}$: E*Trade Rides Billowing Wave of E-Payments http://www.internetnews.com/ec-news/article.php/2232791	0	
		$Doc_{3,3}$: Internet Shopping Value Chain http://iis.kaist.ac.kr/KAISTEC/2001lecture/KAIST2001F-eBank.ppt	non-jugé	
		$Doc_{3,4}$: Payment Schemes for Internet http://www.magnet-i.com/magnet/techzone/guide/ecomm/207	1	
		$Doc_{3,5}$: Suppliers Directory - E-Commerce Focus - Internet One-Stop http://www.ecomfocus.com/cgi-bin/directory/list.asp?sector=28	0.5	

Tableau 6.3. *Les étapes de recherche courantes de la session collaborative*

Premièrement, l'extraction de la mémoire collaborative. Un ensemble des étapes de recherche courantes est repéré : $E_t = E_{courant}$ {*Etape-R*$_1$, *Etape-R*$_2$, *Etape-R*$_3$}. Les ensembles des requêtes QE_{tc} et des documents RE_{tc} sont construits où seulement les requêtes q_i et les documents $Doc_{i,j}$ qui réalisent les critères de personnalisation précisés sont mis en commun :

$$QE_{tc}\ \{q_i : Eval\text{-}Etape_i > 0\ \&\ Pref(u_i) > 0\} \Rightarrow QE_{tc}\ \{\ q_2, q_3\ \}$$

$$RE_{tc}\ \{Doc_{i,j} : JugeDoc(Doc_{i,j}) > 0\ \&\ Pref(u_i) > 0\} \Rightarrow RE_{tc}\ \{Doc_{2,1}, Doc_{2,2}, Doc_{2,3}, Doc_{2,4}, Doc_{3,1}, Doc_{3,4}, Doc_{3,5}\}$$

Une fois les requêtes et les documents mis en commun, le système de soutien calcule la valeur d'importance d'une requête (voir le Tableau 6.4), ou d'un document (voir le Tableau 6.5) dans les ensembles, cela se passe en deux temps : dans un premier temps, le système enlève les duplications, ici les documents $Doc_{2,1}$, $Doc_{3,5}$ sont identiques, $Doc_{2,1}$ est gardé, sa valeur *JugeDoc* combinée unique de jugements de plusieurs utilisateurs est :

$JugeDoc(Doc_{2,1}) = ½ [JugeDoc_{u2}(Doc_{2,1}) + JugeDoc_{u3}(Doc_{3,5})] = ½ [1+ 0.5] = 0.75$

sa valeur combinée et unique de préférence est :

$½ [Pref(u_2) + Pref(u_3)] = ½ [1+ 0.5] = 0.75$

Dans un deuxième temps, le système calcule la valeur d'importance d'une requête ou d'un document par rapport aux deux critères de soutien :

$L(Doc_{i,j}) = ½ (Pref(u_i) + JugeDoc(Doc_{i,j}))$

$L(q_i) = ½ (Pref(u_i) + Eval\text{-}Etape_i)$

Requête q_i	**Evaluation de requête *Eval-Etape*$_i$**	**Préférence *Pref*(u_i)**	***L*(q_i)**
q_2 : e-business solutions online catalogue	0.8	1	0.9
q_3 : Internet payment e-trade	0.4	0.5	0.45

Tableau 6.4. *La mise en commun des requêtes et le calcul de leur valeur d'importance.*

***R*$_{qi}$ Document *Doc*$_{i,j}$**	***JugeDoc***	**Préférence *Pref*(u_i)**	***L*(*Doc*)**
$Doc_{2,1}$: Suppliers Directory - E-Commerce Focus - Internet One-Stop http://www.ecomfocus.com/cgi-bin/directory/list.asp?sector=28	0.75	0.75	0.75
$Doc_{2,2}$: in2info limited - ebusiness solutions http://www.in2info.co.uk/starter/	1	1	1
$Doc_{2,3}$: Clear IT - eBusiness Solutions - Web Development http://www.clearit.com.au/eBusiness/WebDevelopment.htm	1	1	1
$Doc_{2,4}$: Seagrass - eBusiness strategy - eBusiness software solutions http://www.seagrasssoftware.com/77.htm	1	1	1
$Doc_{3,1}$: ePSO Newsletter http://epso.jrc.es/newsletter/vol06/5.html	0.5	0.5	0.5
$Doc_{3,4}$: Payment Schemes for Internet http://www.magnet-i.com/magnet/techzone/guide/ecomm/207	1	0.5	0.75

Tableau 6.5. *La mise en commun des documents et le calcul de leur valeur d'importance.*

Le nombre de documents ou de requêtes à garder est limité dans ces ensembles à cinq donc les cinq premières requêtes et les cinq premiers documents qui ont obtenu la plus grande valeur *L* sont sélectionnés.

QEL_{tc} { q_2, q_3 }

REL_{tc} { $Doc_{2,2}$, $Doc_{2,3}$, $Doc_{2,4}$, $Doc_{2,1}$, $Doc_{3,4}$ }

Deuxièmement, la réalisation des différents types de soutien collaborative. Si le type de soutien est :

1. Présentation de résultats :

   $Range\ (Doc_{i,j}) = L(Doc_{i,j})$

   Le soutien personnalisé est celui du Tableau 6.6 :

Range **($Doc_{i,j}$)**	***$Doc_{i,j}$***	**Documents**
1	$Doc_{2,2}$	in2info limited - ebusiness solutions http://www.in2info.co.uk/starter/
	$Doc_{2,3}$	Clear IT - eBusiness Solutions - Web Development http://www.clearit.com.au/eBusiness/WebDevelopment.htm
	$Doc_{2,4}$	Seagrass - eBusiness strategy - eBusiness software solutions http://www.seagrasssoftware.com/77.htm
0.75	$Doc_{2,1}$	Suppliers Directory - E-Commerce Focus - Internet One-Stop http://www.ecomfocus.com/cgi-bin/directory/list.asp?sector=28
	$Doc_{3,4}$	Payment Schemes for Internet http://www.magnet-i.com/magnet/techzone/guide/ecomm/207

Tableau 6.6. *Présentation de meilleurs résultats des utilisateurs préférés de u_1.*

2. Présentation de requêtes :

   $Range\ (q_j) = L(q_j)$

   Le soutien personnalisé est celui du Tableau 6.7 :

Range **(q_j)**	***q_i***	**Requêtes**
0.9	q_2	e-business solutions online catalogue
0.45	q_3	Internet payment e-trade

Tableau 6.7. *Présentation de requêtes efficaces des utilisateurs préférés de u_1.*

3. Fusion de requêtes :

$Tq_{fusion} = \cup_{qj \in QELtc} Tq_j$

Le soutien personnalisé est :

Tq_{fusion} : {e-business, solutions, online, catalogue, Internet, payment, e-trade}

6.4. Evaluation

Les auteurs dans [Twidale 1994] ont posé la question : **qu'est ce qu' une évaluation ?**

Ils trouvent à travers les revues des réponses comme l'évaluation est : une estimation si le logiciel accomplit le but pour lequel on l'a prévu ; Ou une estimation du degré auquel le logiciel accomplit ses spécifications en termes de fonctionnalité, vitesse, taille ou d'autres mesures qui ont été pré-spécifiées ; Ou encore une estimation de quelle partie du logiciel qui n'exécute pas comme désiré...

Selon [Pinelle 2000], les systèmes de groupware deviennent plus communs, plus d'attention est prêtée à la façon d'évaluer les systèmes multi-utilisateurs. Les chercheurs et les développeurs ont employé une gamme des techniques comprenant des méthodologies scientifiques d'engineering, et des sciences sociales, mais il n'y a aucun consensus clair sur quelles méthodes sont appropriées, et dans quelles circonstances. En outre, le groupware est traditionnellement considéré comme difficile à évaluer à cause des effets des personnes, et du contexte social et organisationnel.

Une meilleure compréhension de la façon dont des systèmes de groupware ont été évalués dans le passé (paragraphe 6.4.1) peut nous aider à encadrer la discussion de quelles méthodes et techniques devraient être considérées pour l'évaluation de notre système de soutien (paragraphe 6.4.2).

6.4.1. *Evaluation de Groupware*

Les auteurs dans [Pinelle 2000] ont passé en revue tous les papiers de la conférence ACM CSCW (1990-1998) qui ont présenté ou ont évalué un système de groupware. Ils décrivent un ensemble de catégories et de critères afin d'analyser les papiers, et ils font une classification des évaluations de groupware selon : les types d'évaluation (paragraphe 6.4.1.1), les caractéristiques de l'évaluation (paragraphe 6.4.1.2), les techniques utilisée pour collectionner les données (paragraphe 6.4.1.3), le placement de l'évaluation dans le cycle de développement de logiciel (paragraphe 6.4.1.4), et le type de conclusions tirées de l'évaluation (paragraphe 6.4.1.5).

Dans le paragraphe 6.4.1.6 nous résumons les évaluations adoptées par certaines approches collaboratives de la recherche d'information présentées précédemment dans le CHAPITRE.1.

6.4.1.1. Type d'Evaluation

Les types d'évaluation sont distingués par rapport à l'encadrement de l'évaluation et à la « formalité » du processus d'évaluation, le Figure 6.4 illustre cela. Ainsi, il y quatre types :

- encadrement contrôlé et manipulation rigoureuse : comme l'expérimentation de laboratoire.
- encadrement naturel et manipulation rigoureuse : comme l'expérimentation sur le terrain.
- encadrement naturel et manipulation minimale ou pas de manipulation : comme l'étude sur le terrain, et l'étude de cas.
- encadrement contrôlé et manipulation minimale ou pas de manipulation : cela est appelé l'exploration puisque le chercheur recueille des informations générales de l'étude et ne poursuit pas une méthodologie fixe pour contrôler l'expérimentation.

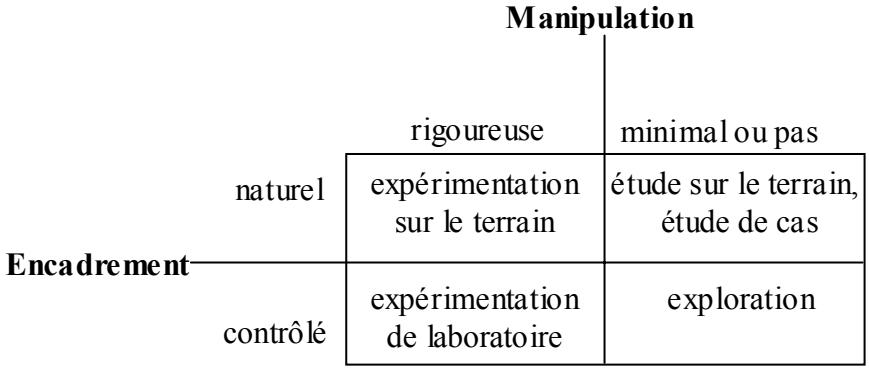

Figure 6.4. *Classification de type d'évaluation.*

6.4.1.2. Caractéristiques de l'Evaluation

Les évaluations ont été encore classifiées selon la rigueur de la manipulation expérimentale et la rigueur des mesures effectuées, ce qui produit les caractéristiques suivantes de l'évaluation :

- *formative* c'est une évaluation progressive qui permet l'apprentissage (l'évaluation d'un prototype) contre *summative* qui est une évaluation d'une implémentation finale et complète.
- quantitatif contre qualitative.
- manipulation formelle/rigoureuse, manipulation minimale, pas de manipulation.
- mesure formelle, mesure informelle.

6.4.1.3. Techniques pour Collectionner les Données

Il y a sept techniques principales :

1. l'observation de l'utilisateur.

2. l'entretien.
3. la discussion.
4. le questionnaire.
5. la mesure qualitative du travail, c'est-à-dire essayer d'évaluer certains aspects qualitatifs de travail de l'utilisateur. Cela implique habituellement un juge qui critique la performance de la tâche de l'utilisateur ou le produit final qu'il a obtenu.
6. la mesure quantitative du travail, cela est basé sur le temps et l'efficacité de l'utilisateur à accomplir une tâche donnée, en faisant enregistrer des données quantitatives et objectives à propos du travail de l'utilisateur.
7. la collection des matières archivistiques, comme des bulletins (newsletters), e-mail, …etc.

6.4.1.4. Placement de l'Evaluation dans le Cycle de Développement de Logiciel

On considère six placements potentiels de l'évaluation :

1. évaluations périodiques à travers la procédure de développement,
2. évaluation continue tout au long le développement,
3. évaluation d'un prototype,
4. évaluation d'une partie spécifique du logiciel,
5. évaluations périodiques après l'implémentation de logiciel,
6. évaluation continue après l'implémentation de logiciel.

6.4.1.5. Type de Conclusions Tirées de l'Evaluation

L'évaluation peut inclure :

- l'impact organisationnel/l'impact sur les pratiques du travail,
- le produit final obtenu en employant le logiciel,
- l'efficacité de l'exécution de tâche en utilisant le logiciel,
- la satisfaction d'utilisateur avec le logiciel,
- le soutien de tâche fournie par le logiciel,
- les caractéristiques spécifiques de l'interface de groupware,
- les modèles/les patrons de l'utilisation du système,
- l'interaction de l'utilisateur tout en utilisant le logiciel.

Plusieurs de ces classifications mentionnées ci-dessus ne sont pas mutuellement exclusives. Par exemple, une approche peut avoir une évaluation *formative* suivie par une évaluation *summative* de son implantation finale ; Ou une évaluation peut rassembler des données qualitatives et quantitatives.

6.4.1.6. Evaluation Adoptée dans les Approches Collaboratives de Recherche d'Information

Nous avons parcouru les expérimentions effectuées par certains approches collaboratives présentées dans le CHAPITRE.1 afin d'évaluer ces approches, nous résumons la démarche effectuée ainsi :

- Comparaison entre une utilisation individuelle et collaborative : l'expérimentation est d'effectuer la même tâche avec des utilisateurs qui réalisent la tâche individuellement et avec des utilisateurs qui réalisent la tâche en collaboration puis de comparer ces deux réalisations [Jaczynski 1998]. Dans [Chau 2003] la même tâche a été réalisée individuellement (avec un seul utilisateur) puis en collaboration (avec 2 utilisateurs, puis trois utilisateurs…) puis ces différentes réalisations ont été comparées afin d'examiner si une amélioration progressive est obtenue avec l'augmentation du nombre d'utilisateurs qui réalisent la tâche.
- Observation : l'expérimentation est d'effectuer la tâche avec des utilisateurs en collaboration et de les observer, c'est-à-dire observer leurs comportements et leurs façons d'utiliser le prototype, … [Romano 1999], [Twidale 1996] et [Zhang 2002].

6.4.2. *Evaluation de Prototype*

L'objectif est de réaliser des expériences pour mieux comprendre l'impact de l'outil utilisé sur la tâche à réaliser. L'étude d'évaluation est conçue plus précisément pour répondre aux questions suivantes :

- La recherche d'information collaborative amène-t-elle une amélioration de l'efficacité et de la qualité de recherche ?
- Est-ce que l'utilisateur a tiré profit de l'aide que le système lui a fourni ?

L'évaluation de la contribution du système de soutien nous a conduit à mettre en place deux expérimentations :

1. Effectuer deux situations de recherche une individuelle et l'autre collaborative et comparer ces deux situations afin d'identifier l'impact et les apports de la collaboration par rapport à la recherche individuelle (paragraphe 6.4.2.1).
2. Effectuer des recherches collaboratives, et étudier la qualité et l'efficacité de la recherche et de son soutien (paragraphe 6.4.2.2).

Dans les situations de recherche collaborative, nous avons voulu recueillir la perception des participants vis-à-vis de l'outil. Un questionnaire a été élaboré et est soumis aux participants à la fin de la recherche collaborative. Le questionnaire contient des questions fermées et des questions ouvertes (voir annexe B).

6.4.2.1. Impact de la Collaboration sur la Recherche

Pour mettre en place l'expérimentation, nous avons demandé à trois personnes de chercher individuellement des informations sur le sujet « *le commerce électronique et le paiement par Internet* » pendant une demi-heure, dont le but déclaré est d'écrire un article sur ce sujet.

Dans cette première expérimentation, le prototype implante seulement la *présentation* et la *fusion de requêtes*.

Puis, nous avons formé un groupe, que nous baptisons le $groupe_1$, des mêmes trois participants situés dans des lieux distants, nous leur avons demandé de chercher collectivement des informations sur le même sujet et pendant le même laps de temps avec la possibilité d'utiliser le soutien.

Pendant la recherche individuelle les utilisateurs jugent les références par rapport à un seul critère qui est la pertinence du contenu et de façon binaire, pour la recherche collaborative le type de recherche du groupe est la recherche *jointe*, donc au démarrage de la session collaborative, le groupe a déterminé les même paramètres de jugement afin de pouvoir faire la comparaison entre les deux situations.

Les trois utilisateurs se connaissent bien, ils sont de la même équipe de recherche, deux sont des thésards en deuxième année, la troisième personne est un étudiant en DEA.

Nous avons collecté d'une part, les sessions de recherche individuelle des trois utilisateurs u_1, u_2, u_3 dont les sessions sont S_1, S_2, S_3, et d'autre part la session de recherche collaborative *Session-RIC* du groupe qui contient les trois sessions individuelles $Sesssion\text{-}RII_1$, $Sesssion\text{-}RII_2$, $Sesssion\text{-}RII_3$. Les résultats obtenus sont détaillés dans le paragraphe 6.4.2.1.1 et discutés dans le paragraphe 6.4.2.1.2.

6.4.2.1.1. Recherches Individuelle et Collaborative

Nous avons effectué les comparaisons suivantes entre la recherche individuelle et collaborative :

- au niveau des requêtes : comparer le nombre des requêtes soumises par les utilisateurs, et la distribution des termes de ces requêtes, le Tableau 6.8 illustre cela :

	Recherche Individuelle	**Recherche Collaborative**	**En Commun**
nombre de requêtes soumises q_i	11	12	-
nombre de requêtes différentes soumises q_i	11	8	-
Nombre total de termes des requêtes $\|T_{qi}\|$	14	10	7

Tableau 6.8. *Résumé des résultat de la 1ère experimentation au niveau des requêtes.*

  - le nombre des requêtes : le nombre des requêtes par utilisateur est à peu près le même (11 dans la recherche individuelle et 12 dans la recherche collaborative). Dans la recherche individuelle, les 11 requêtes sont différentes. Dans la recherche collaborative, parmi les 12 requêtes il n'y a que 8 requêtes différentes.
  - la distribution des termes des requêtes : les termes des requêtes dans la recherche individuelle sont plus nombreux et plus dispersés ; il y a 14 termes différents utilisés dans la recherche individuelle, et 10 termes différents dans la recherche collaborative.

7 termes sont en commun et utilisés dans les deux recherches collaborative et individuelle.

- au niveau des résultats : nous faisons les deux comparaisons suivantes comme l'illustre le Tableau 6.9 :

  - Comparer le résultat collectif avec le résultat de recherche individuelle c'est-à-dire : RC_1 avec RI_1, RC_2 avec RI_2, et RC_3 avec RI_3. Le but de savoir qu'est-ce que la collaboration a apporté à chacun des utilisateurs par rapport à leur recherche individuelle.

    Nous remarquons que le nombre de documents obtenus par la recherche individuelle de chaque utilisateur RI_{ui} est proche de celui obtenu par l'utilisateur dans une recherche collaborative RC_{ui}. Les documents jugés pertinents retrouvés $Rjugé\text{-}per_{ui}$ dans la recherche individuelle sont plus nombreux que ceux qui sont retrouvés et jugés pertinents dans la recherche collaborative.

  - Comparer le résultat collectif *RC* (qui est la disjonction des résultats RC_1, RC_2 et RC_3) avec le résultat de recherche individuelle *RI* (qui est la disjonction des résultats RI_1, RI_2, RI_3). Le but de savoir qu'est-ce que la collaboration tout au long de la session a apporté aux utilisateurs par rapport à une collaboration à la fin de leurs recherches individuelles (c'est-à-dire par rapport à une mise en commun des trois résultats RI_1, RI_2, RI_3 issus de la recherche individuelle de chacun une fois est finie).

    Nous remarquons également que le nombre de documents retrouvés dans le résultat mis en commun des recherches individuelles *RI* est plus grand que celui du résultat collectif *RC*. Parmi ces documents retrouvés le nombre des documents pertinents dans *RI* est significativement plus grand que celui de *RC*.

R_{ui}		$Rjugé\text{-}per_{ui}$	$Rjugé\text{-}non\text{-}per_{ui}$	$Rnon\text{-}jugé_{ui}$	**Jugé différemment** ***per/ non-per***
RC_1	43	13	26	2	2
RI_1	46	23	21	2	-
RC_2	59	35	19	4	1
RI_2	64	38	24	2	-
RC_3	77	14	57	5	1
RI_3	62	20	38	4	-
RI	163	76	76	7	4
RC	116	30	57	8	21

Tableau 6.9. *Résultats collectifs RC_1, RC_2, RC_3, RC et individuelles RI_1, RI_2, RI_3, RI.*

Les résultats de ces deux comparaisons vont dans le sens envers de notre motivation à savoir que la recherche collaborative est meilleure que la recherche individuelle. Nous pensons que l'infériorité de l'efficacité de la recherche collaborative est normale parce que les utilisateurs sont arrivés au résultat collaboratif *RC* avec un nombre de requêtes

moindre du fait que les utilisateurs ont réutilisé les requêtes d'où la diminution du nombre des documents différents.

Le Tableau 6.10 présente l'intersection entre les résultats collaboratif et individuel. Nous remarquons qu'il y a très peu de documents communs et aussi très peu de documents pertinents communs entre le résultat individuel de chaque utilisateur RI_{ui} et son résultat collaboratif RC_{ui} et cela s'applique aussi sur le résultat individuel mis en commun *RI* et le résultat collaboratif *RC* global. Cela montre qu'avec la recherche collaborative les utilisateurs trouvent des documents (en général, et des documents pertinents, plus précisément) qu'ils n'ont pas pu retrouver dans leur recherche individuelle, et même s'il y a une mise en commun à la fin.

	∩ commun	**∩ commun** ***Rjugé-per***	**∩ commun** ***Rjugé-non-per***	**∩ commun jugé différemment** ***jugé-per/ jugé-non-per***
RI1, RC1	7	2	3	2
RI2, RC2	9	7	1	1
RI3, RC3	3	-	1	2
RI, RC	20	3	4	13

Tableau 6.10. *Comparaisons entre les résultats collectifs et individuels.*

Alors la collaboration tout au long de la session a permis aux utilisateurs de trouver des documents pertinents qu'ils n'auraient pas pu les retrouver autrement.

6.4.2.1.2. Discussion

Pour la recherche collaborative et individuelle le nombre des requêtes est presque égal tandis que la distribution des termes des requêtes est différente. Par contre, on remarque une focalisation sur un ensemble des termes communs en recherche collaborative, ainsi les utilisateurs quand ils ont accès aux requêtes et aux termes utilisés par les autres ont préféré les réutiliser.

Nous avons mentionné dans l'introduction que notre approche est censée réduire la redondance de requêtes mais cela n'est pas le cas ; Nous avons pensé que cela est dû au fait que la première version du prototype ne permettait que l'accès aux requêtes des autres utilisateurs. En effet, une fois une présentation des requêtes faite, l'utilisateur n'a pas la possibilité de visualiser et de parcourir les résultats de ces requêtes. D'ailleurs on (l'utilisateur) a mentionné cette frustration dans la réponse au questionnaire.

Ce constat nous a conduit pour la deuxième expérimentation à développer une deuxième version du prototype qui donne la possibilité de visualiser le résultat d'une requête avec son jugement.

6.4.2.2. Impact du Soutien sur la Recherche

Le contexte de cette deuxième expérimentation ressemble à celui de $group_1$ de la première expérimentation où le but est d'écrire un article sur le même sujet de recherche « *le commerce électronique et le paiement par Internet* » mais nous avons prolongé le temps de recherche jusqu'à une heure parce que nous nous sommes aperçus qu'une demie heure était trop juste. Le type de recherche des groupes est la recherche *jointe*, ils jugent les références par rapport à un seul critère qui est la pertinence du contenu et de façon binaire. Nous avons formé deux groupes : le $groupe_2$ qui consiste en trois personnes qui sont des thésards dans le domaine de la recherche d'information ; et le $groupe_3$ qui comprend quatre participants, qui sont des thésards aussi mais dans différents domaines en informatique. Les participants de deux groupes ne sont pas spécialistes dans le domaine de recherche mais plutôt débutants, et ils se situent dans des lieux différents.

Dans cette expérimentation, le prototype implante non seulement la *présentation* et la *fusion de requêtes*, mais aussi la *présentation des résultats*. Une évaluation du système est aussi implantée et demandée par le jugement des meilleures références collaboratives retrouvées par le groupe, afin de mesurer la satisfaction des utilisateurs par rapport au résultat de recherche collaborative. Cette présentation exclue pour chaque utilisateur son propre résultat. Ainsi, nous avons demandé aux utilisateurs d'évaluer les 20 premières références retrouvées et jugées pertinentes par les autres.

Nous avons collecté d'une part, la session de recherche collaborative de deux groupes et d'autre part les évaluations du système c'est-à-dire les évaluations des résultats collaboratifs. Nous présentons dans la suite les mesures selon lesquelles nous étudions ces informations collectées (paragraphe 6.4.2.2.1), puis nous détaillons les résultats obtenus (paragraphe 6.4.2.2.2). Dans le paragraphe 6.4.2.2.3 nous analysons les réponses au questionnaire où les réponses des trois groupes sont prises en compte.

6.4.2.2.1. Mesures

Dans l'étude de l'impact du soutien sur la recherche, nous prenons en compte les données issues des situations de recherches collaboratives, c'est-à-dire la recherche collaborative des groupes $groupe_2$ et $groupe_3$ mais aussi celle du $groupe_1$ de la première expérimentation. Les variables suivantes sont à mesurer : la *performance* et l'*utilité* de système.

- la *performance* est mesurée à travers les indicateurs suivants :
  - au niveau des fonctionnalités : la facilité à comprendre, la qualité de soutien, les types de soutien utilisés réellement c'est-à-dire pendant la recherche et l'importance de chacun de ces types du point de vue utilisateur, les critères de soutien utilisés réellement ainsi que l'importance de ces différents critères selon l'utilisateur.
  - au niveau du fonctionnement : les contraintes, les points forts et les points faibles.
- l'*utilité* : étudie l'étendu du soutien fourni à la réalisation de la tâche de recherche et elle est mesurée à travers les indicateurs suivants :
  - au niveau des requêtes : l'impact et l'influence de soutien sur les requêtes, et leur redondance.

- au niveau des résultats : l'impact de l'outil sur les résultats finaux, et si l'outil aide à obtenir un résultat collaboratif global plus satisfaisant.

6.4.2.2.2. Résultats

Dans la deuxième expérimentation, nous avons enregistré la succussion/relation soutien(s) personnalisé(s)-requête(s) soumise(s), afin d'étudier l'impact du soutien fourni sur la ou les requête(s) soumise(s). Nous avons imposé aux utilisateurs de demander le soutien mais ils avaient le choix de tenir compte ou non de l'aide fournie dans leur recherche. Nous avons remarqué que le soutien a influencé l'utilisateur dans sa formulation de nouvelles requêtes ; les utilisateurs ont réutilisé un ou plusieurs mot(s) clé(s) et ils ont complété la formulation par leurs mots clés. A ce propos plusieurs utilisateurs se sont exprimés et ont déclaré « *cela me donne des idées* ». Nous avons remarqué aussi l'absence de la redondance des requêtes, aperçu dans la première expérimentation où les utilisateurs disposent ici des résultats des requêtes des autres et n'ont pas alors besoin de les re-exécuter. Les participants ont utilisé l'aide proposée par le système.

Nous étudions maintenant deux points à propos des types et des critères de soutien : le premier point est l'importance des types et des critères de soutien selon la fréquence de leurs utilisations, le deuxième est leur importance exprimée par les utilisateurs dans le questionnaire c'est-à-dire l'idée que l'utilisateur se fait de leur importance, après l'avoir expérimenté.

Nous étudions l'importance des types et des critères de soutien selon leurs utilisations et nous vérifions cela par rapport au questionnaire[3], puis nous discutons de l'importance de ces types et critères.

- Types de soutien : nous avons pris en compte les données issues de la recherche de chacun des $groupe_2$ et $groupe_3$, puisque le $groupe_1$, de la première expérimentation n'avait pas le type de soutien « présentation des résultats » et il n'a pas été interrogé à propos de l'importance des types. L'importance des différents types est résumée dans le Tableau 6.11 et analysé ainsi :

moyen	Présentation des requêtes	Fusion des requêtes	Présentation des résultats
utilisation	57.29%	19.86%	22.86%
questionnaire	37.86%	23.71%	38.86%
total	47.57%	21.78%	30.86%

Tableau 6.11. *Importancemoyenne des types de soutien.*

[3] Le calcul en pourcentage (%) de l'importance des types et des critères de soutien se déroule ainsi :
- en ce que concerne l'utilisation :
importance% (type ou critère) = Nb-fois-utilisé(type ou critère) / $\sum_{\text{type ou critère}}$ Nb-fois-utilisé(type ou critère)
donc l'importance de tous les types ou les critères est égale à cent :
100% = $\sum_{\text{type ou critère}}$ [Nb-fois-utilisé (type ou critère) / $\sum_{\text{type ou critère}}$ Nb-fois-utilisé (type ou critère)]
- en ce que concerne le questionnaire :
importance% (type ou critère) = importance-donnée(type ou critère) / $\sum_{\text{type ou critère}}$ importance-donnée(type ou critère)
donc l'importance de tous les types ou les critères est égale à cent :
100% = $\sum_{\text{type ou critère}}$ [importance-donnée (type ou critère) / $\sum_{\text{type ou critère}}$ importance-donnée (type ou critère)]

1. Importance selon l'utilisation : la présentation des requêtes est le type de soutien très fréquemment utilisé tandis que la présentation des résultats et la fusion des requêtes le sont beaucoup moins.
2. Importance selon le questionnaire : les utilisateurs apprécient presque également la présentation des résultats et des requêtes alors que la fusion des requêtes vient loin après.
3. Importance totale : la présentation des requêtes est visiblement la plus importante, puis la présentation des résultats et la fusion de requête arrive la dernière.

Nous remarquons que la présentation des requêtes se place selon l'importance de l'utilisation et du questionnaire en tête et la fusion des requêtes se place au bout selon l'importance des deux, donc il y un accord et une homogénéité entre les deux points malgré la grande différence entre les deux valeurs d'importance.

Par contre, il y a un grand décalage entre l'importance de la présentation des résultats dans l'utilisation et celle déclarée dans le questionnaire. Nous pensons que ce décalage est arrivé parce que les utilisateurs en situation de recherche réelle apprécient beaucoup d'avoir un résultat collaboratif puisque les références d'un tel résultat concernent un sujet de recherche qui les intéresse, donc dans le questionnaire ils placent la présentation des résultats en tête tandis que dans l'expérimentation les utilisateurs simulent une recherche et alors ils sont moins intéressés par le produit de la recherche.

En effet, les utilisateurs apprécient surtout la présentation des requêtes puis la présentation des résultats et la rubrique « points forts et intéressants » montre en fait cela. Nous pensons que le peu d'intérêt porté à la fusion des requêtes peut être dû à deux raisons : d'une part parce que les utilisateurs préfèrent déduire eux même une nouvelle requête à partir des présentations des requêtes, d'autre part parce que les mots clés de la requête fusionnée sont présentés sous une forme « lemmatisée » c'est-à-dire sans les préfixes, les suffixes,…

- Critères de soutien : le Tableau 6.12 résume les statistiques sur l'importance des critères de soutien, nous avons pris en compte les données issues de la recherche des trois groupes, et nous les avons analysées comme suit :
  1. Importance selon l'utilisation : le jugement est le critère le plus importent suivie par la préférence, viennent beaucoup plus loin les requêtes similaires et différentes et ainsi que plus loin les résultats similaires et différents.
  2. Importance selon le questionnaire : la préférence est perçue par les utilisateurs comme le critère le plus important suivi tout de suite par le jugement, plus loin les requêtes différentes ensuite les requêtes similaires et les résultats similaires et différents ont tous les trois à peu près la même importance.
  3. Importance totale : selon les critères utilisés et leurs importances annoncés le jugement et la préférence sont significativement les critères les plus importants ; plus loin derrière viennent les requêtes différentes ensuite les requêtes similaires et les résultats similaires et différents.

moyen	jugement	préférence	similarité des requêtes		similarité des résultats	
			similaires	différentes	similaires	différents
utilisation	33.5%	28.5%	11.7%	13.1%	6.3%	6.6%
questionnaire	22.8%	24.9%	12%	15.8%	12.9%	11.9%
total	28.15%	26.7%	11.85%	14.45%	9.6%	9.25%

Tableau 6.12. *Importance moyenne des critères de soutien.*

Nous remarquons que globalement il n'y pas de décalage mais plutôt une corrélation et continuité entre l'importance de chaque critère dans l'utilisation et celle déclarée dans le questionnaire.

Nous venons d'analyser et comparer selon les valeurs de certains des critères c'est-à-dire nous avons considéré séparément les valeurs similaire et différente des critères « similarité de requêtes » et « similarité de résultats ». En effet, nous pensons que l'analyse de cette façon nuance plus l'importance des critères par rapport à une analyse qui considère la somme des importances des deux valeurs « similaire » et « différente » pour le critère « similarité de requête » et pour le critère « similarité de résultat ». Ainsi même dans le questionnaire nous avons demandé aux utilisateurs d'indiquer l'importance de ces valeurs séparément c'est-à-dire indiquer l'importance : des requêtes similaires, des requêtes différentes, des résultats similaires et des résultats différents.

6.4.2.2.3. Questionnaire

Nous présentons l'analyse des réponses au questionnaire posé aux participants des trois groupes sachant que certaines questions ont été ajoutées dans la deuxième expérimentation (pour $groupe_2$ et $groupe_3$) qui prend en compte la discussion sur le résultat de la première expérimentation et le nouveau développement que nous avons fait en y basant, dans ce qui suit nous mentionnons quand il s'agit des questions ajoutées.

Les utilisateurs sont moyennement satisfaits (80%), voire satisfaits (20%)[4] de l'aide apportée par le système pour la recherche : certains ont justifié la satisfaction moyenne que nous résumons ainsi : d'une part, parce que la quantité des informations présentées grossit très vite et on se retrouve alors submergé, ce constat est légitime et compréhensible, car nous n'avons pas implanté l'étape qui limite le nombre de requêtes, documents, ou termes présentés à l'utilisateur. D'autre part, parce qu'il y a peu d'utilisateurs (groupe de 3 ou 4 personnes) et qu'avec une mémoire collaborative plus grande, on bénéfice davantage de l'aide sur la recherche. En effet, cette remarque ressemble au problème du « démarrage à froid » (voir le paragraphe 1.2.2.2) perçu dans le filtrage collaboratif actif et le groupware en général, mais notre outil peut fournir de l'aide malgré le peu de données retrouvées dans la mémoire, bien sûr la qualité de l'aide s'améliore avec un plus grand nombre d'utilisateurs ou avec une utilisation asynchrone.

[4] Les utilisateurs ont le choix pour décrire l'aide entre : satisfaisante, moyenne, peu satisfaisante, insuffisante.

Le système et la procédure de recherche sont jugés compréhensibles (90%), néanmoins les utilisateurs ont déclaré certaines difficultés de compréhension, quand on leur a posé la question, à propos des types de soutien et des critères de personnalisation (surtout la similarité des requêtes, la similarité de résultats et la préférence) où il a fallu les essayer une fois au début pour les comprendre, ainsi les utilisateurs voudraient qu'ils soient expliqués et que le système anticipe sur le soutien qu'on obtient avec tel ou tel critère.

L'obligation de juger les documents a été ressentie par la plupart (70%) lourde et prend du temps, pourtant certains des participants disent aimer avoir plusieurs critères de jugement et plusieurs valeurs de jugement. En effet, nous sommes d'accord, d'ailleurs dans notre modèle théorique ainsi dans le prototype, nous avons étudié et implanté le jugement selon plusieurs critères de qualité et avec plusieurs valeurs de jugement, mais dans les différentes expérimentations nous avons demandé un jugement selon la pertinence du contenu et de façon binaire pour alléger cette tâche qui est apparu malgré cela un peu lourde. Nous ajoutons que les utilisateurs simulent une recherche c'est-à-dire ils ne sont pas réellement ou spécialement intéressés par le sujet. Le jugement de référence n'aurait pas été perçu comme fastidieux, si les utilisateurs avaient été motivés par l'obtention de résultats de qualité.

Deux autres tâches sont ressenties comme lourdes : la détermination des préférences des autres utilisateurs (40%) et la demande du soutien en sélectionnant les critères (30%), où l'utilisateur doit fournir cela à chaque demande. La rubrique « améliorations suggérées » clarifie cela, car 60% des utilisateurs préfèrent l'automatisation de demande de soutien (même si 40% des utilisateurs préfèrent ne pas l'automatiser) c'est-à-dire les utilisateurs préfèrent plutôt que le système leur suggère des nouvelles requêtes, des présentations de résultats,… selon sa sélection des critères, certains ont conditionné une telle automatisation par une explication à l'utilisateur de ce qui se passe, ou par une adaptation à l'utilisateur, comme le système affiche les requêtes qui sont similaires à la requête de l'utilisateur, ce qui facilite le travail, dans notre étude théorique nous sommes allées dans cette direction d'adaptation de soutien où nous avons défini une stratégie de soutien selon la typologie de l'utilisateur. Nous pouvons alléger ces deux tâches en mémorisant les valeurs de leurs paramètres au sein de la session avec possibilités de changement. Par contre, presque tous les participants (90%) sont contre un soutien dit indépendant c'est-à-dire un système qui déduit et soumet des requêtes et donne le résultat à l'utilisateur sans que ce dernier intervienne sur la formulation des requêtes.

	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	***Moyen***
Précision Rjugé-per / RC	0.75	0.53	0.3	0.75	1	0.33	0.75	0.63

Tableau 6.13. *Evaluation du système par les utilisateurs.*

Pour la deuxième expérimentation (les groupes 2 et 3, donc les utilisateurs u_4, u_5,…, u_{10}) une évaluation du système est demandée par le jugement des meilleures références collaboratives retrouvées par le groupe (voir le paragraphe 6.4.2.2), la satisfaction des utilisateurs mesurée par la précision de ce résultat collaboratif, et elle est présentée dans le Tableau 6.13. Le résultat collaboratif est précis à (0.63%) en moyenne sur tous les utilisateurs, les utilisateurs sont satisfait du système. En réponse à la question « Est-ce que le résultat collaboratif vous satisfait plus que vos résultats individuels ? » les utilisateurs ont répondu à (30%) par oui et à (70%) par non ou par oui hésitant et conditionné. Ils ont justifié cela par la

généralité du sujet de recherche ce qui a conduit à une diversité de sa compréhension et une diversité des objectifs de recherche pris par chacun. Certains pensent qu'un tel résultat collaboratif peut être plus satisfaisant que les résultats individuels si les jugements des références est à peu près homogène à travers le groupe ou si la mémoire collaborative est assez grande pour contenir des utilisateurs dont l'objectif est proche. En revanche, tous les utilisateurs (100%) sont d'accord que si cette recherche a pour but de rédiger une synthèse sur le sujet en question, le résultat collaboratif apporte un plus par rapport à ce qu'ils avaient obtenu.

Ce que les utilisateurs ont exprimé précédemment renforce notre étude à propos de la définition de plusieurs types de groupe ainsi selon les utilisateurs et la similarité de leurs centres d'intérêt et leurs objectifs, ils peuvent collaborer et même se mettre d'accord à propos des critères de jugement afin que leur jugement soit homogène dans le cas de recherche jointe. Le sujet de recherche a été sciemment choisi général, parce que dans notre étude nous visons une collaboration plus générale à propos d'un sujet de recherche et pas seulement à propos d'une question de recherche, (cela peut être le cas pour une recherche jointe), alors nous voulons que les utilisateurs collaborent et s'entraident malgré la diversité de directions de recherche, la rubrique « points forts et intéressants » l'affirme.

La rubrique « améliorations suggérées » a incité les utilisateurs à donner les propositions suivantes :

- Au niveau d'interface les utilisateurs ont proposé d'avoir une interface plus synthétique (par exemple, la demande du soutien et le soutien personnalisé s'affiche sur la même page) et intuitif, avec plus de texte explicatif de façon à guider l'utilisateur ; ou d'avoir une interface graphique comme celui proposé dans CoVitesse. Le système CoVitesse permet la création et l'organisation des différents types de groupe pour la navigation collaborative synchrone (voir le paragraphe 4.5.2) ainsi que la visualisation de groupe et des documents pertinents retrouvés.
- En ce qui concerne plus précisément l'*interface recherche*, les utilisateurs proposent de fournir certains avantages de l'interface de moteur de recherche (en occurrence google ici) comme : afficher le texte pertinent et les mots clés en gras ce qui facilite le jugement, et augmenter le langage de requête où le système actuel ne supporte pas encore le recherche par un ensemble de mots exacts cernés par des guillemets « ».
- prendre en compte le type de documents ou de site (commercial .com, gouvernement, faculté...) comme un critère de personnalisation de soutien. Nous pensons que ce critère peu être intéressant dans certains cas seulement, comme celui du sujet de recherche de l'expérimentation où le sujet est assez général et il est abordé par différentes sortes de site, tandis qu'un tel critère peut s'avérer moins intéressant pour un sujet qui ne contient pas autant d'aspects par exemple un sujet de recherche à propos de l'environnement, ou de la santé…
- présenter des informations statistiques comme le nombre d'utilisateurs, de documents jugés,…
- automatiser le jugement selon le temps passé sur la page.
- indiquer les pages visitées et jugées par l'utilisateur lui-même, et celles qui sont visitées et jugées par les autres, et ne présenter à l'utilisateur que les pages nouvelles.

- utiliser le système de façon asynchrone, nous avons étudié cette possibilité mais elle n'a pas été implantée.
- garder une trace de l'utilisateur afin de ne pas s'identifier à chaque fois qu'il utilise le système.

Sous la rubrique « points forts et intéressants » du système les utilisateurs ont exprimé leur satisfaction générale du système ainsi :

- le système est simple, compréhensif et rapide (sauf l'interrogation de google) malgré son aspect synchrone (son utilisation par plusieurs utilisateurs en même temps), agréable, facile à utiliser et bien conçu.
- il permet un gain de temps.
- il enrichit la recherche de l'utilisateur parce que les autres utilisateurs pensent à des choses auxquelles il n'a pas pensé.
- la recherche collaborative est un plus par rapport à un simple moteur de recherche.
- le système permet de bénéficier du jugement d'autres personnes sur des documents que l'utilisateur lui-même n'avait pas retrouvés et jugés, de plus la présentation des résultats donne des idées pour la formulation des requêtes.
- la présentation des requêtes et leur fusion donnent des idées pour trouver les mots les plus efficaces surtout pour les utilisateurs qui ne sont pas spécialistes dans le domaine et elle permet à l'utilisateur de modifier ses propres requêtes.
- la dimension temporelle *historique* a été appréciée.

6.5. Conclusion

Nous situons dans le Tableau 6.14 l'évaluation réalisée du prototype selon l'ensemble des catégories et des critères définis par [Pinelle 2000] pour classifier les évaluations de groupware et que nous avons présenté précédemment dans le paragraphe 6.4.1.

Nous pensons qu'il faut expérimenter avec un grand nombre d'utilisateurs et dans des situations de recherche réelles c'est-à-dire avec des sujets de recherche qui intéressent vraiment les utilisateurs, nos expérimentations n'ont nous permis qu'une première évaluation du prototype. Néanmoins les résultats de cette évaluation sont encourageants car ils ont montré les intérêts et les motivations des utilisateurs pour la collaboration dans la recherche, en terme de gain de temps, d'enrichissement de la recherche et de diminution de la redondance de requêtes, ce qui renforce notre approche. Le soutien fourni a aidé les utilisateurs, puisque ces derniers en ont tiré profit et réutilisé dans leurs requêtes certains mots clés, et puisque le résultat collaboratif les a satisfaits.

Les résultats de l'évaluation nous suggèrent aussi d'améliorer l'interface afin de faciliter la compréhension et la réalisation des différentes tâches comme le jugement, la demande de soutien, … Par exemple, dans l'état actuel du prototype, quand l'utilisateur « clique » sur un lien dans la liste de résultat une nouvelle fenêtre s'ouvre et affiche la page correspondante, une fois que l'utilisateur a examiné la page, il doit fermer cette fenêtre et retourner à la liste de résultat pour juger la page. Donc une amélioration de l'interface est nécessaire pour la rendre plus synthétique et conforme aux critères ergonomiques de l'interface. Nous avons tiré

d'autres suggestions à partir de nos expérimentations que nous exposons dans la conclusion de cette thèse qui suive ce chapitre.

type d'évaluation	• encadrement contrôlé et manipulation rigoureuse (expérimentation de laboratoire).
caractéristiques de l'évaluation	• formative. • quantitatif et qualitatif. • manipulation formelle. • mesure formelle.
techniques de collectionner les données	• questionnaire. • mesures quantitatives de travail.
placement de l'évaluation dans le cycle de développement de logiciel	• évaluation d'un prototype.
type de conclusions tirées de l'évaluation	• l'impact sur les pratiques du travail. • le produit final. • l'efficacité de l'exécution de tâche en utilisant le logiciel. • la satisfaction d'utilisateur avec le logiciel. • le soutien de tâche fournie par le logiciel.

Tableau 6.14. *Résumé de la classification de l'évaluation de prototype réalisée.*

Conclusion

Nous avons présenté dans ce travail une approche collaborative pour la recherche d'information. Notre problématique était de soutenir un groupe d'utilisateurs dans leur processus de recherche à propos d'un sujet de recherche commun. Nous avons cerné les différents aspects concernant la problématique.

Nous nous sommes intéressées aux collecticiels en général, leurs classifications et leurs fonctionnalités, puis nous avons examiné plusieurs approches de soutien collaboratif pour la recherche d'information ce qui nous a permis de déterminer les manques de ces approches. En général, on ne prend en compte dans ces approches collaboratives que le profil de l'utilisateur en terme de mots clés et ou de références sélectionnées (comme dans les systèmes de filtrage collaboratif ou de recommandation, …).

Nous pensons qu'un environnement collaboratif doit aider l'utilisateur à obtenir une meilleure qualité de l'information, et à bénéficier efficacement des expériences des autres membres du groupe. Pour cela, nous avons été amené, tout d'abord, à étudier la nature des soutiens possibles et préciser les informations intéressantes à fournir à l'utilisateur pour l'aider dans sa tâche de recherche d'information. Puis, à faire une étude détaillée sur l'utilisateur afin de définir les mesures nécessaires pour évaluer ses compétences techniques et ses compétences dans le domaine de recherche. Ensuite, à considérer la nature du groupe effectuant une recherche collaborative et le degré de collaboration entre ses membres. Nous avons pu ainsi définir une typologie de l'utilisateur et une typologie du groupe. L'aide fournie dépend de ces deux typologies. Enfin, nous avons défini plusieurs critères de personnalisation et une procédure pour exploiter la mémoire collaborative afin de réaliser le soutien personnalisé.

L'environnement proposé dans ce travail, élimine des contraintes « même outil, même endroit, même temps, et même question de recherche » et ainsi permet à des membres du groupe de rechercher ensemble, même s'ils sont distribués physiquement, temporellement, et techniquement, et même si leurs objectifs de recherche sont différents. L'environnement utilise la notion de *préférence* qui prend en compte les relations sociales entre les utilisateurs s'ils se connaissent. Il utilise, également, la notion de *complémentarité* de compétences et expertises des utilisateurs sur la technique et sur le domaine. Il permet d'offrir un soutien pour un utilisateur donné à partir des recherches des utilisateurs du groupe qui ont des qualités complémentaires et qui peuvent compenser ses manques de compétence.

Avec le prototype qui n'implante qu'une partie de cet environnement, la qualité finale de recherche s'est améliorée, les expérimentations l'ont prouvé ; de plus, les utilisateurs ont été satisfaits de l'aide apportée. Du point de vue théorique, la typologie de l'utilisateur permet d'adapter le soutien à l'utilisateur, mais elle n'était pas intégrée dans le prototype qui a servi aux expérimentations ; cette adaptation a manqué aux utilisateurs. Le jugement des documents est utilisé comme une information clé pour établir le soutien, cependant, dans l'expérimentation les utilisateurs trouvent contraignant d'avoir à juger les documents. Alors, certains aspects sont à compléter et à améliorer à court terme.

1. Perspectives à Court Terme

Nous résumons les aspects à développer à court terme ainsi :

- Compléter l'implémentation et mettre en place toute la proposition étudiée théoriquement, cela comprend le développement du prototype, notamment, l'implémentation de stratégie de soutien selon la typologie de l'utilisateur. La prise en compte des différents outils de recherche et des différentes collections.
- Faciliter le jugement des documents, ainsi le jugement peut être fourni explicitement par l'utilisateur ou déduit implicitement selon les actions de l'utilisateur. Par exemple, le temps d'ouverture du document, l'ajout du lien du document dans les favoris de l'utilisateur, l'enregistrement du document, l'impression du document, …etc.
- L'interface doit être améliorée en différents points. Premièrement, intégrer une interface collaborative qui offre toutes les fonctionnalités nécessaires pour créer et organiser différents types de groupe d'utilisateurs. Deuxièmement, tenir compte de la familiarité de l'utilisateur avec le moteur de recherche, en employant et intégrant le moteur de recherche et son interface à l'interface de recherche de notre système. Cela veut dire que, d'une part, les utilisateurs n'ont pas besoin d'apprendre avec une nouvelle interface de recherche parce qu'ils sont déjà familiarisés et habitués à utiliser le moteur de recherche, d'autre part les utilisateurs peuvent bénéficier des avantages du moteur de recherche (comme la présentation en gras d'une ou des plusieurs occurrence(s) du ou des terme(s) de la requête dans le page Web de la référence). Troisièmement, étudier une présentation graphique qui illustrent la visualisation selon les différents critères de personnalisation de façon à rendre l'effet de ces critères plus explicite ; par exemple, un graphe termes-utilisateurs qui montre les utilisateurs qui ont utilisé le même terme dans leurs requêtes ; un graphe requête-requête qui montre les requêtes qui sont similaires, …
- Réaliser des expérimentations plus exhaustives : en plus d'expérimenter les points précédents non implantés, il est intéressant d'expérimenter le prototype avec un grand nombre d'utilisateurs, ceci, en situation de recherche réelle, pour mieux étudier l'efficacité du système. Il faudrait de plus, expérimenter les différentes stratégies de recherche selon la typologie de groupe que nous avons implantées, et étudier l'adaptation des stratégies aux différents types de groupe.
- Etudier la situation, que nous avons mentionné dans l'introduction et que nous n'avons délibérément pas étudié, dont l'objectif est de construire un document selon un plan. Ce plan peut être prédéfini individuellement par un utilisateur ou collaborativement par le groupe.

Notre proposition est une première étude dans le vaste domaine qu'est la recherche collaborative. Dans la suite, nous présentons des idées et des directions de travail à long terme.

2. Perspectives à Long Terme

De plus en plus, l'Internet et les différentes outils de communications proposés (courrier électronique, forum de discussion, le *chat*, formulaires administratifs électroniques …) jouent un rôle important. Ainsi, à la fin du siècle dernier et au début de ce siècle, les modes de communication entre les gens ont changé que ce soit dans les communications officielles,

administratives, ou personnelles, et l'on parle même de la transformation du monde en un village grâce à ces nouvelles techniques. Vu que l'être humain est par nature sociable et éprouve le besoin de communiquer, il est naturel que de nouveaux modes de communication émergent. En particulier, les personnes éprouvent le besoin de partager (et/ou de réutiliser) des connaissances et surtout de collaborer sur l'Internet. Par exemple, les gens conseillent par é-mail ou par liste de diffusion à leurs proches (collègues ou amis, …) une nouveauté (un livre, une cause, une nouvelle, …) ou éventuellement à des étrangers via les forums de discussion. En général, lorsque l'on a besoin d'aide, on le demande à une personne que l'on connaît ou que l'on pense capable de nous aider. On observe que, de plus en plus, de personnes adhèrent et participent à des forums de discussion pour chercher quelqu'un qui a eu un problème semblable ou identique au leur, afin de bénéficier d'une solution déjà trouvée. Si tel n'est pas le cas, on peut « poster » une demande d'aide et l'on mise alors sur la générosité des autres pour obtenir une réponse. Bien que la démarche de rechercher dans le forum de discussion ou de poster un é-mail … puisse être un peu longue, ces forums rencontrent un succès.

Nous pensons que la recherche d'information ne peut plus être un phénomène essentiellement individuel et doit prendre en compte l'aspect collaboratif de l'activité de recherche que ce soit de manière synchrone ou asynchrone.

Les progrès technologiques et la mise à disposition de grands volumes d'informations complexes et hétérogènes obligent à définir des moyens d'accès intelligents et collaboratifs. Beaucoup d'études sont faites dans ce sens. Nous pensons que la problématique de recherche d'information doit évoluer vers la définition de nouveaux modèles et des systèmes *complets* intégrant les différentes dimensions de la collaboration et où l'utilisateur serait de plus en plus pris en compte. A l'heure actuelle, la collaboration n'a été étudiée que selon des points de vue particuliers : l'interface collaborative, le filtrage collaboratif, recommandation, collecticiel, …

Un tel système doit fournir une interface collaborative qui permet les interactions entre les gens pour s'entraider, se conseiller ou s'organiser dans des groupes…, il doit permettre aussi bien le partage du produit de recherche par les services de recommandation ou de filtrage collaboratif, que le partage du processus de recherche lui-même et l'aide en ligne comme c'est le cas de notre approche.

L'on peut souligner que le problème de « démarrage à froid » dont souffrent la plupart des approches collaboratives peut être compensé par l'intégration d'autres approches qui prennent en compte les caractéristiques de l'utilisateur collectées dans son contexte de travail. On peut citer le système AIRA (Adaptive Information Research Assistant) [Bottraud 2003] qui emploie, d'une part, les références collectées par l'utilisateur (ses favoris par exemple) pour établir un modèle hiérarchique (profil de l'utilisateur) des intérêts de cet utilisateur, d'autre part, les informations recueillies tout en observant les activités de l'utilisateur (documents édités ou consultés, segments de texte copiés, …etc.) pour choisir les parties les plus appropriées de ce profil et les utiliser alors afin d'interpréter les requêtes et de filtrer les résultats retournés par les moteurs de recherche sur le Web. Une telle approche devrait être intégrée dans un tel système dit *complet* pour prendre en compte davantage l'utilisateur et sa situation ou son contexte de recherche.

Un tel système pourrait faire collaborer les utilisateurs et atteindre un niveau de perfection tel qu'il puisse remplacer la collaboration face à face !

Bibliographie

[Aroyo 1999] Aroyo, L., Dicheva, D. « *Information Retrieval and Visualisation within the Context of an Agent-based Information Management System* ». Proceedings of Educational Multimedia, Hypermedia and Telecommunications (Ed-Media). pp. 195-200, 1999.

[Bartell 1994] Bartell, B. T., Cottrell, G. W., Belew, R. K. « *Automatic combination of multiple ranked retrieval systems* ». Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. pp. 173-81, 1994.

[Belkin 1993] Belkin, N.J., Cool, C., Croft, W.B., Callan, J.P. « *The effect of multiple query representations on information retrieval performance* ». Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. pp. 339-346, 1993.

[Bergsten 2001] Bergsten, H. « *JavaServer Pages* ». Editions O'Reilly, Edition française : Traduction Soulard, H. 2001.

[Berrut 1997] Berrut, C. « *Indexation des données multimédia, utilisation dans le cadre d'un système de recherche d'informations* ». Habilitation à Diriger des Recherches, Université Joseph Fourier, Grenoble,1997.

[Bottraud 2003] Bottraud, J.C., Bisson, G., Bruandet, M.F. « *An Adaptive Information Research Personal Assistant* ». Proceedings of Workshop Artificial Intelligence, Information Access and Mobile Computing (AI2IA), of International Joint Conference on Artificial Intelligence (IJCAI). 8 pages. 2003. http://www.dimi.uniud.it/workshop/ai2ia.

[Brin 1998] Brin, S., Page, L. « *The Anatomy of a Large-Scale Hypertextual Web Search Engine* ». Proceedings of the 7th international conference on World Wide Web 7. pp. 107-117, 1998.

[Bruandet 2003] Bruandet, M. F., Chevallet, J. P. « *Assistance Intelligente à la Recherche d'Informations* ». Edition Hermes, Auteurs : Gaussier, E., Stefanini, M. H. Chapitre 3, pp. 85-118, 2003.

[Chau 2003] Chau, M., Zeng, D., Chen, H., Huang, M., Hendriawan, D. « *Design and Evaluation of a Multi-agent Collaborative Web Mining System* ». Decision Support Systems (DSS). V.35, N°1, pp. 167-183, 2003.

[Chignell 1999] Chignell, M. H., Gwizdka, J., Bodner, R. C. « *Discriminating meta-search: a framework for evaluation* ». Information Processing and Management : an International Journal.V.35, N°3, pp. 337-362, 1999.

[Courbon 1997] Courbon, J. C., Tajan, S. « *Groupware et Internet : application avec Notes et Domino* ». Edition Masson/InterEditions. 1997.

[Croft 1995] Croft, W.B. « *What do people want from information retrieval*? ». D-Lib Magazine, November 1995. http://www.dlib.org/dlib/november95/11croft.html.

[Defude 1986] Defude, B. « *Etude et réalisation d'un système intelligent de recherche d'informations : Le prototype IOTA* ». Thèse, Institut National Polytechnique de Grenoble, 1986.

[Diamond 1996] Diamond, T. « *Information retrieval using dynamic evidence combination* ». Thèse Dissertation Proposal, 1996. http://www.nwlink.com/_tgdiamon. (Mentionné par [Vogt 1999]).

[Dunlop 1997] Dunlop, M. D. « *Time Relevance and Interaction Modelling for Information Retrieval* ». Proceedings of the 20th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. pp. 206-213, 1997.

[Favier 1998] Favier, M., Coat, F., Couron, J.C., Trahand, J. « *Le travail en Groupe à l'Age des Réseaux* ». Edition economica. Chapitre V. Courbon J,C. 1998.

[Fox 1994] Fox, E. A., Shaw, J. A. « *Combination of Multiple Searches* ». Proceedings of the 2nd Text Retrieval Conference (TREC-2), National Institute of Standards and Technology (NIST) Special Publication 500-215, pp. 243-252, 1994.

[Furnas 1987] Furnas, G. W., Landauer, T. K., Gomez, L. M., Dumais, S. T. « *The vocabulary problem in human-system communication* ». Communications of the ACM. V.30, N°11, pp. 964-971, 1987.

[Glance 1999] Glance, N., Grasso, A., Borghoff, U. M., Snowdon, D., Willamowski, J. « *Supporting Collaborative Information Activities in Networked Communities* ». Proceedings of the 8th International Conference on Human-Computer Interaction (HCI), V.2, pp. 422-426, 1999.

[Harman 1993] Harman, D. « *Overview of the first TREC conference* ». Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. pp. 36-47, 1993.

[Hoppe 1994] Hoppe, H. U., Zhao, J. « *C-TORI: an interface for cooperative database retrieval* ». Proceedings of the 5th International Conference on Database and Expert Systems Applications (DEXA). pp. 103-113, 1994.

[Hust 2002a] Hust, A., Klink, S., Junker, M., Dengel, A. « *Query Reformulation in Collaborative Information Retriev*al ». Proceedings of International Conference on Information and Knowledge Sharing (IKS). ACTA Press, pp. 95-100, 2002.

[Hust 2002b] Hust, A., Klink, S., Junker, M., Dengel, A. « *Towards Collaborative Information Retrieval: Three Approaches* ». Text Mining, Theoretical Aspects and Applications. pp. 97-112. 2002.

[Jaczynski 1998] Jaczynski, M. « *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, Université de Nice-Sophia Antipolis, France, 1998.

[Kanawati 1999] Kanawati, R., Jaczynski, M., Trousse, B., Andreoli, J. M. « *Applying the Broadway Recommandation Computation Approach for Implementing a Query Refinement Service in the CBKB Meta-search Engine* ». Conférence Nationale de Raisonnement à partir de cas (RàPC), 1999.

[Lardy 2002] Lardy, J. P. « *RIsI : Recherche d'Information Sur l'Internet-Outils et Méthodes* ». Livre électronique, 7ème édition papier-Mai 2001 ADBS. Unité Régionale de Formation et de promotion pour l'Information Scientifique et Technique (URFIST de Lyon). dernière mise à jour : septembre 2002. http://urfist.univ-lyon1.fr/risi/risi.htm.

[Largouet 2000] Largouet, A. « *La recherche d'information sur Internet* ». Support de cours. Unité Régionale de Formation et de promotion pour l'Information Scientifique et Technique (URFIST de Bordeaux). Octobre 2000. http://www.montesquieu.u-bordeaux.fr/urfist/.

[Laurillau 1999] Laurillau, Y., Nigay, L. « *Modèle de Navigation Collaborative Synchrone pour l'Exploration des Grands Espaces d'Information* ». Actes ERGO-IHM2000, CRT ILS&ESTIA, pp. 121-128, 2000.

[Lee 1997] Lee, J. H. « *Analyses of Multiple Evidence Combination* ». Proceedings of the 20th annual international ACM-SIGIR Conference on Research and development in information retrieval. pp. 267-276, 1997.

[Lee 1998] Lee, J. H. « *combining the evidence of different relevance feedback methods for information retrieval* ». Information Processing & Management. V.34, N°6, pp. 681-691, 1998.

[Leloup 1998] Leloup, C. « *Moteurs d'indexation et de recherche* ». Edition Eyroles. 1998.

[Lueg 1998] Lueg, C. « *Considering Collaborative Filtering as Groupware: Experiences and Lessons Learned* ». Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM). pp. 161-166, 1998.
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-13/

[Maltz 1994] Maltz, D. A. « *Distributing Information for Collaborative Filtering on Usenet Net News* ». Master's thesis, MIT Department of Electrical Engineering and Computer Science EECS, Massachusetts Institute of Technology, Cambridge, 1994.

[Masinter 1993] Masinter, L., Ostrom, E. « *Collaborative Information Retrieval: Gopher from MOO* ». Proceeding INET'93. 1993.
ftp://parcftp.xerox.com/pub/MOO/papers/moogopher.

[Mizzaro 1998a] Mizzaro, S. « *How many relevances in information retrieval?* ». Interacting With Computers. V.10, N°3, pp. 303-320, 1998.

[Mizzaro 1998b] Mizzaro, S. « *Relevance: The whole history* ». Historical Studies in Information Science, Journal of the American Society for Information Science (JASIS), Special Issue, pp. 221-243, 1998.

[Mizzaro 1999] Mizzaro, S. « *Measuring the agreement among relevance judges* ». Proceedings of MIRA Conference: Evaluating interactive information retrieval, electronic Workshops in Computing (eWiC). pp. 105-115, 1999.

[Mizzaro 2001] Mizzaro, S. « *A new measure of retrieval effectiveness* », (Or: What's wrong with precision and recall). International Workshop on Information Retrieval (IR). pp. 43-52, 2001.

[Mizzaro 2002] Mizzaro, S., Tasso, C. « *Ephemeral and Persistent Personalization in Adaptive Information Access to Scholarly Publications on the Web* ». Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH). pp. 306-316, 2002.

[Muller 2000] Muller, P. A., Gaertner, N. « *Modélisation objet avec UML* ». Edition Eyrolles, 2000.

[Neches 1998] Neches, R., Abhinkar, S., Hu, F., Eleish, R., Ko, I.,Yao, K., Zhu, Q., Will, P. « *Collaborative information space analysis tools* ». D-Lib Magazine, Octobre 1998. http://www.dlib.org/dlib/october98/dasher/10dasher.html.

[Pinelle 2000] Pinelle D., Gutwin C. « *A Review of Groupware Evaluations* ». Proceedings of the 9th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises. pp.86-91, 2000.

[Redman 1998] Redman, T. « *La qualité des données à l'âge de l'information* ». Edition Masson/InterEditions. 1998.

[Reid 2000] Reid, J. « *A task-oriented non-interactive evaluation methodology for information retrieval systems* ». Information Retrieval. V.2, N°1, pp. 115-129. 2000.

[Rocchio 1971] Rocchio, J.J. « *Relevance feddback in Information Retrieval* ». The SMART Retrieval System: Experiments in Automatic Document Processing, Editeur Gerard Salton, Prentice Hall , Inc., Englewood Cliffs NJ, 1971.

[Romano 1999] Romano, N., Roussinov, D., Nunamaker, J.F., Chen, H. « *Collaborative information retrieval environment: integration of information retrieval with group support systems* ». Proceedings of the 32nd Annual Hawaii International Conference on System Sciences (HICSS). V.1, pp. 1053, 1999.

[Rumbaugh 1995] Rumbaugh, J., Blaha, M., Eddy, F., Premerlani, W., Lorensen, W. « *OMT Modélisation et conception orientées objet* ». Edition Masson/InterEditions, Edition française : Traduction Fontaine, A. 1995.

[Salton 1971] Salton, G. « *The SMART Retrieval System: Experiments in Automatic Document Processing* ». Prentice Hall Inc., Englewood Cliffs NJ, 1971.

[Salton 1990] Salton, G., Buckley, C. « *Improving retrieval performance by relevance feedback* ». Journal of the American Society for Information Science (JASIS). V.41, N°4, pp. 288-297, 1990.

[Savoy 1994a] Savoy, J. « *Principe de la recherche d'informations* ». Séminaire du 3ème cycle romand d'informatique. Université de Neuchâteul, juin 1994.

[Savoy 1994b] Savoy, J. « *Méthodologie pour l'évaluation comparative de systèmes de recherche d'informations* ». Cahier de recherche en informatique, N° CR-I-94-03, Université de Neuchâteul, 31 pages. 1994.

[Sire 1999] Sire, S., Chatty, S. « *Les collecticiels peuvent-ils préserver nos compétences sociales de collaboration ?* ». Revue Informations In Cognito99. N°13, pp. 23-36, 1999.

[Smail 1994] Smail, M. « *Raisonnement a base de cas pour une recherche évolutive d'informations ; prototype Cabri-n. Vers la définition d'un cadre d'acquisition des connaissances* ». Thèse, Université Henri Poincar'e, Nancy, 1994.

[Smeaton 1998] Smeaton, A. F. « *Independence of Contributing Retrieval Strategies in Data Fusion for Effective Information Retrieval* ». Proceedings of the 20th BCS-IRSG Colloquium, Springer-Verlag, Workshops in Computing, 1998.

[Taher 2000] Taher, R. « *Outil pour la recherché d'information collaborative synchrone* », Rapport de DEA, Université Joseph Fourier, Grenoble, 2000.

[Taher 2002] Taher, R. « *Fusion de Requête pour la Recherche d'Information Collaborative et Synchrone* ». 20ème Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID). pp. 431-432, 2002.

[Taher 2003] Taher, R. « *Recherche d'Information Collaborative* ». 1er Congrès Francophone MAnifestation des JEunes Chercheurs STIC (MAJECSTIC). pp. 105, 2003.

[Towell 1995] Towell, G., Voorhees, E. M., Gupta, N. K., Johnson-Laird, B. « *Learning Collection Fusion Strategies for Information Retrieval* ». Proceedings of the 20th Annual Machine Learning Conference. pp. 540-548, 1995.

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

[Turtle 1991] Turtle, H., Croft, W. B. « *Evaluation of an inference network-based retrieval model* ». ACM Transactions on Information Systems. V.9, N°3,pp. 187-222, 1991.

[Twidale 1994] Twidale, M. B., Randall, D., Bentley, R. « *Situated evaluation for cooperative systems* ». Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW). pp. 441-452, 1994.

[Twidale 1995] Twidale, M. B., Nichols, D. M., Smith, G., Trevor, J. « *Supporting Collaborative Learning during Information Searching* ». The 1st international conference on Computer Support for Collaborative Learning (CSCL). pp. 367-370, 1995.

[Twidale 1996] Twidale, M. B., Nichols, D. M. « *Interfaces to support collaboration in information retrieval* ». Proceedings of the BCS (British Computer Society) Information Retrieval and Human Computer Interaction Workshop (BCS IR & HCI Workshop). pp. 25-28, 1996.

[Twidale 1997] Twidale, M. B., Nichols, D. M., Paice, C.D. « *Browsing is a Collaborative process* ». Information Processing and Management: an International Journal. V.33, N°6. pp. 761-783, 1997.

[Vakkari 2001] Vakkari, P. « *Changes in Search Tactics and Relevance Judgements when Preparing a Research Proposal A Summary of the Findings of a Longitudinal Study* ». Information Retrieval. V.4, N°3-4, pp. 295-310, 2001.

[Vogt 1997] Vogt, C C., Cottrell, G. W., Belew, R. K., Bartell, B. T. « *Using Relevance to Train Linear Mixture of Experts* ». Proceedings of the 5th Text Retrieval Conference (TREC-5), National Institute of Standards and Technology (NIST) Special Publication 500-238. pp. 503-516, 1997.

[Vogt 1999] Vogt, C. C., Cottrell, G. W. « *Fusion Via a Linear Combination of Scores* ». Computer Science and Engineering. V.1, N°3, pp. 151-173, 1999.

[Voorhees 1994] Voorhees, E. M., Gupta, N. K., Johnson-Laird, B. « *The collection fusion problem* ». Proceedings of the 3rd Text REtrieval Conference (TREC-3). pp. 95-104, 1994.

[Xiong 1998] Xiong, R., Smith, M., Drucker, S. « *Visualizations of Collaborative Information for End-Users* ». Technical Report MSR-TR-98-52, Microsoft Research, 1998.

[Xu 2000] Xu, J., Croft, W. B. « *Improving the Effectiveness of Information Retrieval with Local Context Analysis* ». ACM Transaction on Information System. V.18, N°1, pp. 79-112, 2000.

[Yao 1995] Yao, Y. Y. « *Measuring Retrieval Effectiveness Based on User Preference of Documents* ». Journal of the American Society for Information Science (JASIS). V.46, N°2, pp. 133-145,1995.

[Zhang 2002] Zhang, X. « *Collaborative Relevance Judgment: A Group Consensus Method for Evaluating User Search Performance* ». Journal of the American Society for Information Science (JASIS). V.53, N°3, pp. 210-219, 2002.

Références

[Aroyo 1999] 3, 164
[Bartell 1994] 164
[Belkin 1993] 164
[Bergsten 2001] 139, 164
[Berrut 1997] 38, 39, 164
[Bottraud 2003] 140, 164
[Brin 1998] 137, 164
[Bruandet 2003] 126, 165
[Chau 2003] 165
[Chignell 1999] 40, 88, 165
[Courbon 1997] 11, 165
[Croft 1995] 3, 165
[Defude 1986] 84, 165
[Diamond 1996] 55, 165
[Dunlop 1997] 4, 42, 165
[Favier 1998] 11, 13, 165
[Fox 1994] 165
[Furnas 1987] 126, 166
[Glance *1999*] 9, 17, 166
[Harman 1993] 50, 166
[Hoppe 1994] 9, 17, 76, 166
[Hust 2002a] 129, 166
[Hust 2002b] 129, 166
[Jaczynski 1998] 147, 166
[Kanawati 1999] 166
[Lardy 2002] 65, 166
[Largouet 2000] 65, 167
[Laurillau 1999] 93, 95, 167
[Lee 1997] 167

[Lee 1998] 53, 54, 55, 167
[Leloup 1998] 39, 167
[Lueg 1998] 20, 167
[Maltz 1994] 82, 167
[Masinter 1993] 9, 16, 17, 167
[Mizzaro 1998a] 4, 31, 33, 167
[Mizzaro 1998b] 167
[Mizzaro 1999] 36, 46, 168
[Mizzaro 2001] 4, 42, 168
[Mizzaro 2002] 19, 168
[Muller 2000] 71, 168
[Neches 1998] 28, 168
[Pinelle 2000] 144, 157, 168
[Redman 1998] 36, 168
[Reid 2000] 39, 168
[Rocchio 1971] 128, 168
[Romano 1999] 9, 26, 147, 169
[Rumbaugh 1995] 71, 169
[Salton 1971] 105, 169
[Salton 1990] 2, 169
[Savoy 1994a] 1, 169
[Savoy 1994b] 115, 169
[Sire 1999] 78, 169
[Smail 1994] 3, 169
[Smeaton 1998] 169
[Towell 1995] 4, 62, 63, 170
[Trousse 1999] 9, 21, 170
[Turtle 1991] 51, 170
[Twidale 1994] 144, 170
[Twidale 1995] 7, 170
[Twidale 1996] 9, 25, 147, 170
[Vakkari 2001] 126, 170
[Vogt 1997] 171

[Vogt 1999] 171
[Voorhees 1994] 4, 62, 63, 171
[Xiong 1998] 4, 23, 171
[Xu 2000] 126, 171
[Yao 1995] 4, 43, 171
[Zhang 2002] 86, 147, 171

Table d'Illustration

Figures

Tableaux

Annexe A. Fusion

Le processus général de la combinaison des résultats des requêtes issues de plusieurs méthodes de bouclage de pertinence est illustré dans la Figure 1.

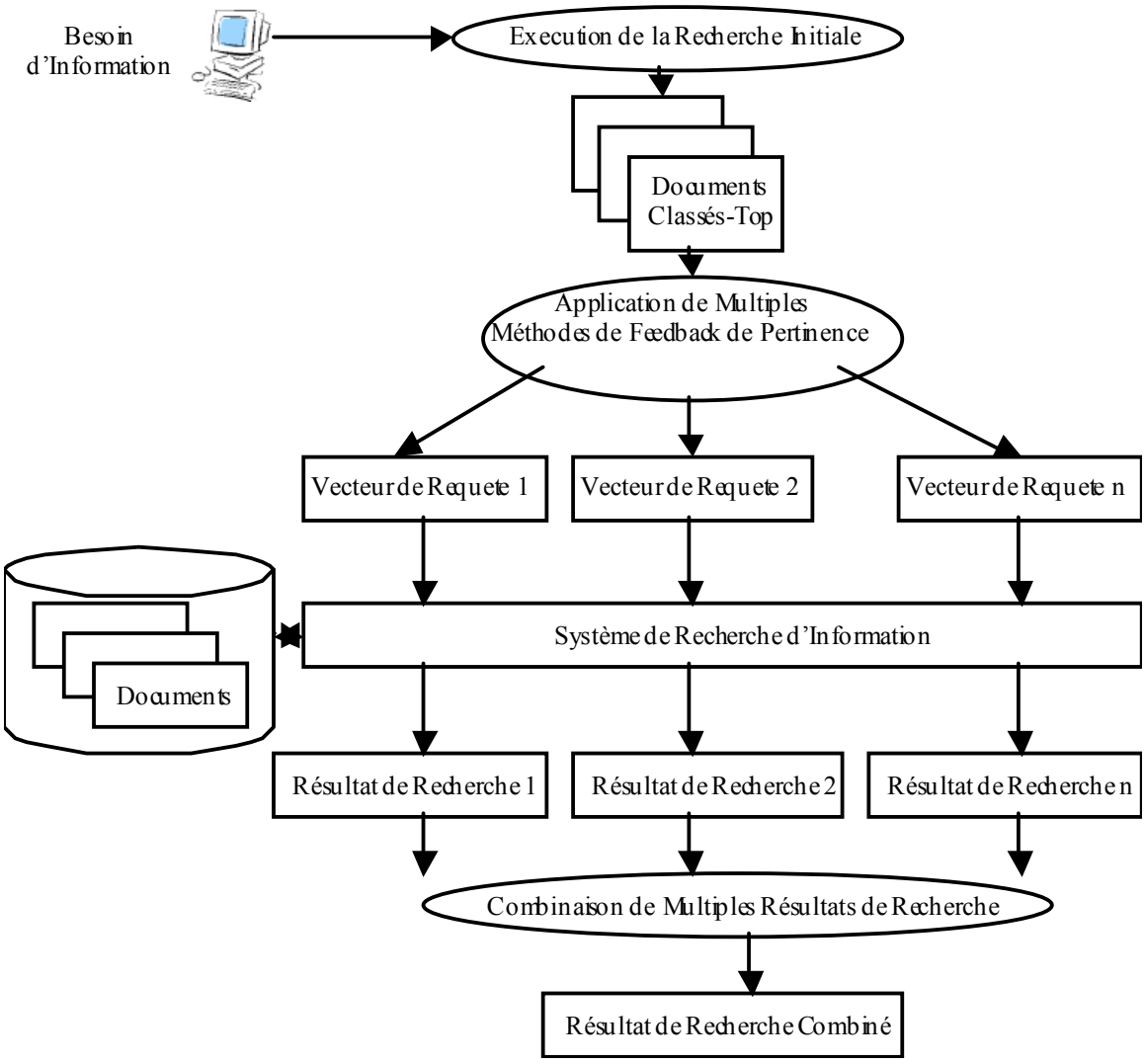

Figure 1. La combinaison de résultats de différentes méthodes de bouclage de pertinence.

La Figure 2 illustre la stratégie de *modélisation de distribution des documents pertinents* (MRDD Modeling Relevant Document Distributions) et la Figure 3 illustre la stratégie de *groupement de requêtes* (QC Query Clustering). Ces deux stratégies sont des stratégies d'apprentissage pour fusionner les résultats issus de plusieurs collections de recherche.

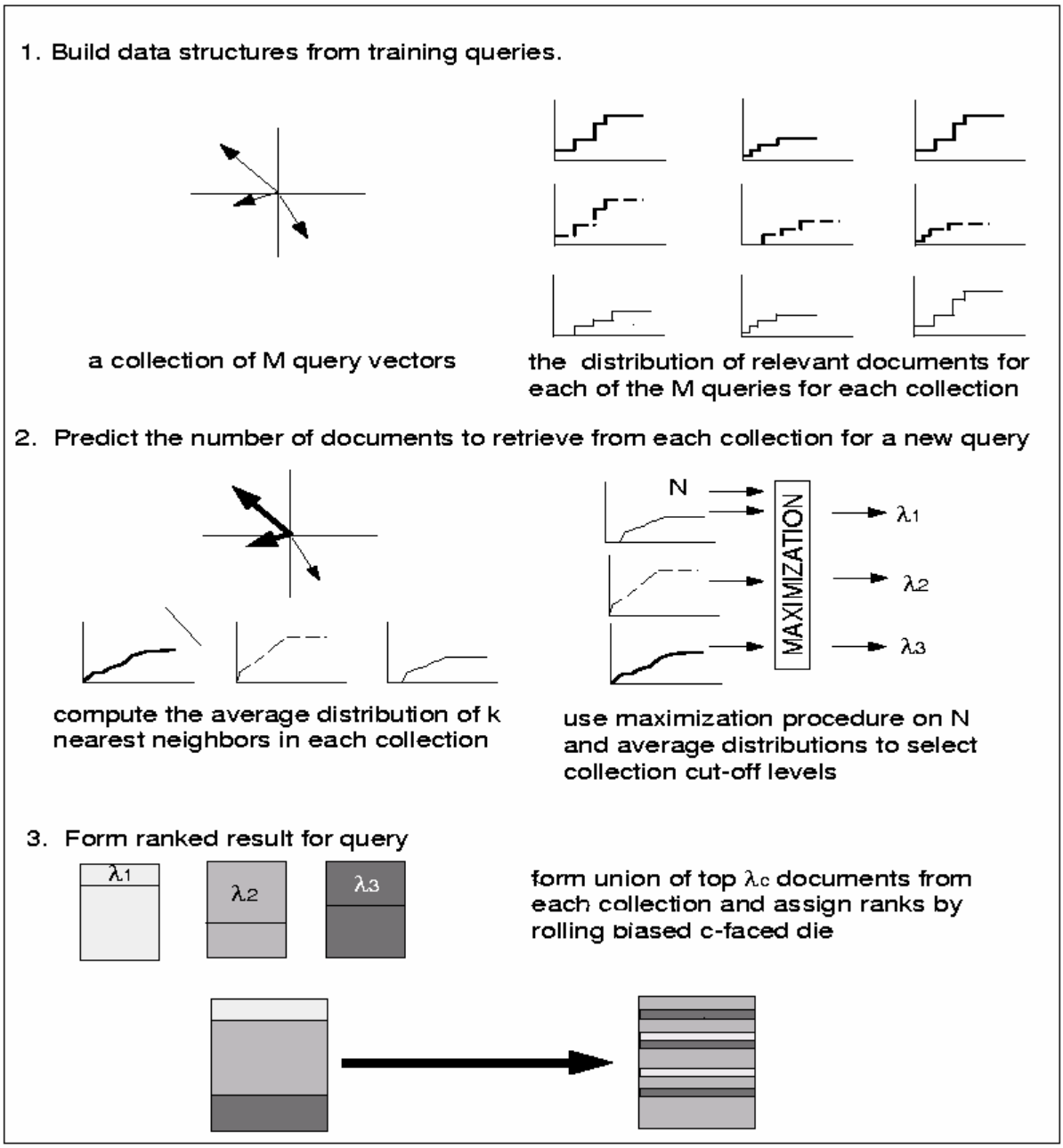

Figure 2. *Stratégie de modélisation de distribution des documents pertinents (MRDD).*

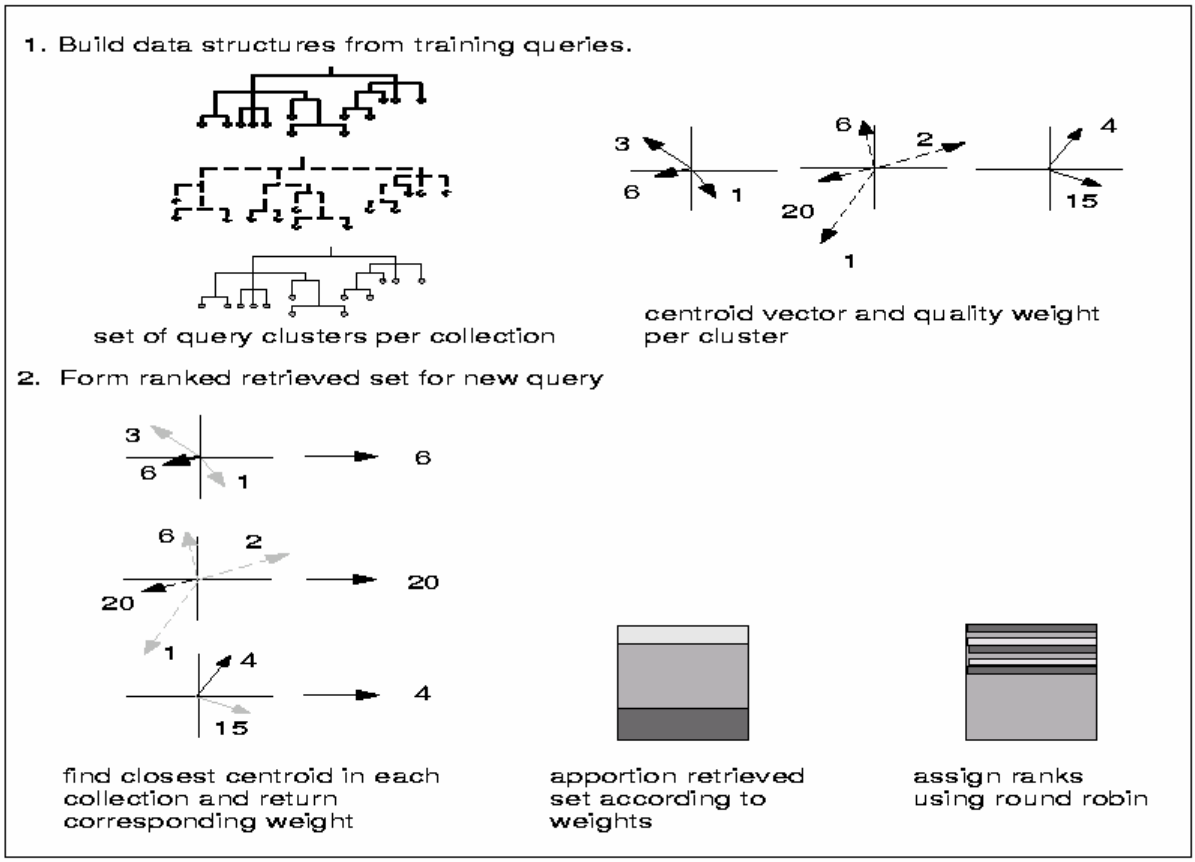

Figure 3. *Stratégie de groupement de requêtes (QC).*

Le Tableau 1 compare plusieurs méta-moteurs.

Nom	Choix des outils	Choix des opérateurs	Réglage temps	Résultats intégrés	Note
All4One	Non 4	Non	Non	Non	-
DigiSearch	Oui 18	Non	O	Non	-
DogPile	Non 14	Non	Oui	Non	+
Highway 61	Non	Oui	Oui	Oui	+
Ixquick	Oui	Oui	Non	Dédoublonnés et triés	++
MetaCrawler	Non	Oui	Non	Oui et triés	++

Tableau 1. *Tableau comparatif des méta-moteurs en ligne.*

Annexe B. Expérimentations

Nous proposons dans la suite :

1. la procédure de l'expérimentation qui consiste de la situation, la tâche, et le scénario donnés aux utilisateurs pour effectuer les différentes expérimentations, cependant l'étape 4 (l'évaluation du système) a été ajoutée pour les groupes *groupe*$_2$ et *groupe*$_3$.
2. le questionnaire élaboré et soumis aux participants à la fin de la recherche collaborative. Où les questions 2, 7 et 8 ont été ajoutées pour les groupes *groupe*$_2$ et *groupe*$_3$.
3. le traitement linguistique appliqué sur les termes de requête dans le prototype.
4. les résultat détaillées des différentes expérimentations.

B.1. Procédure

B.1.1. *Situation et Tâche*

Vous faites partie d'un groupe de 3 personnes, votre tâche de groupe est de rechercher l'information sur le sujet « electronic commerce and the payment by Internet ».

Pour accomplir cette tâche de recherche, vous avez à votre disposition un système de soutien collaboratif dans lequel est intégré le moteur de recherche d'information Google.

Le système, est maintenu sur un serveur de la machine mrim4. Vous avez accès au système à travers une connexion Internet (via un navigateur Web standard) à l'adresse http://mrim4:8080/CollaborativeResearch/html/Interface.html. Ce prototype vous permet de : soumettre une requête, de récupérer le résultat, de l'évaluer, et de demander éventuellement de l'aide. Les requêtes soumises ainsi que les résultas et leur évaluation sont destinés à être communiqués aux autres membres du groupe.

Le type de recherche du groupe dont vous faites partie correspond à une recherche jointe où vous avez tous le même objectif, vous connaissez bien chacune des autres personnes.

Les exigences imposées sont :

- les requêtes que vous soumettrez doivent être en anglais.
- Vous devez évaluer les résultats obtenus.
- Vous devez utiliser le système pour obtenir un soutien.

B.1.2. *Scénario*

Etape 1. Une présentation du système et de ses fonctionnalités, afin de vous faciliter l'utilisation, suivi d'un test pour s'assurer de bonne compréhension de l'outil et de son utilisation.

Etape 2. Le démarrage de la session collaborative par une personne parmi vous : vous vous installez sur un poste, vous démarrez une session collaborative.

1. vous vous identifiez en précisant {nom, prénom, adresse émail}.
2. vous démarrez une session collaborative en précisant (le titre, le type de recherche jointe et éventuellement une description de sujet de recherche), définir les autres paramètres (le nombre de critères de jugement, les critères de jugement, leurs poids, l'échelle de jugement, et l'échelle de préférence).
3. vous commencez une session de recherche individuelle, pendant laquelle vous soumettez des requêtes, et vous évaluez leurs résultats, puis vous procédez selon 4, 5 de l'étape 3 suivante.

Etape 3. une personne a joué dans la deuxième étape un rôle de créateur de session collaborative. Les autres vont joindre la session collaborative, démarrée précédemment :

1. vous vous identifiez.
2. vous joignez la session collaborative.
3. vous commencez une session de recherche individuelle.
4. vous demandez un soutien au cours de votre recherche en précisant : le type de soutien (présentation ou fusion de requête ou présentation/fusion des résultats), la dimension temporaire (courant ou historique), les critères de soutien (jugement, préférence, similarité de requête « similaires ou différentes », similarité de résultats « similaires ou différents »). Il est possible de choisir plusieurs critères à la fois.
5. vous obtenez le soutien, vous pouvez vous servir de ce soutien pour formuler une requête si la proposition du système vous semble intéressante.

Etape 4. la session de recherche dure une heure, après ce temps une évaluation du résultat collaboratif (20 documents) est demandée via le bouton « system evaluation ».

Etape 5. vous fermez la session individuelle via exit.

Etape 6. répondre au questionnaire.

B.2. Questionnaire

Question 1. Lors d'une demande de soutien pour la recherche, indiquez l'importance, pour vous, des critères de soutien suivants (l'égalité est possible)

	Pas important ⟶ important						
- Préférence	0	1	2	3	4	5	6
- Jugement	0	1	2	3	4	5	6
- Requête similaire	0	1	2	3	4	5	6
- Requête différente	0	1	2	3	4	5	6
- Résultat similaire	0	1	2	3	4	5	6
- Résultat différent	0	1	2	3	4	5	6
- Autres critères de soutien souhaités :							

Question 2. Indiquez l'importance, pour vous, des types de soutien suivants (l'égalité est possible)

	Pas important ⟶ important			
- Présentation des requêtes	0	1	2	3
- Fusion des requêtes	0	1	2	3
- Présentation des résultats	0	1	2	3
- Autres types de soutien souhaités :				

Question 3. L'aide que le système vous a apporté pour la recherche est :

- Satisfaisante
- Moyenne
- Peu Satisfaisante
- Insuffisante

Question 4. Selon vous est-ce que le système et la procédure de recherche sont compréhensibles

- Oui
- Non

Question 5. Quelle tâche ou quels termes vous posent des problèmes de compréhension ?

Question 6. Quelles contraintes, vous apparaissent lourdes ?

- Aucune
- Juger les documents
- Fournir des paramètres de la session collaborative (pour une recherche jointe)
- Donner des préférences aux autres utilisateurs
- Demander le soutien et sélectionner les critères de soutien
- Autres, lesquels ?

Question 7. Est-ce que le résultat collaboratif vous satisfait plus que vos résultats individuels ?

Question 8. Si cette recherche a pour but de rédiger une synthèse sur ce sujet est-ce que les résultats collaboratif apporte un plus par rapport à ce que vous avez obtenu ?

Question 9. Etes-vous satisfait du système ? Justifiez votre réponse.

- Oui
- Non

Question 10. En quoi le système pourrait-il être amélioré, suggestion

- Interface :

- Automatiser le soutien : plutôt que le système vous suggère des nouvelles requêtes, de présentation de résultats,… selon son sélection des critères :

- Indépendance : le système déduit et soumet des requêtes et vous donne le résultat sans que vous puissiez intervenir sur la formulation de requête :

- Autres, précisez :

Question 11. Quels sont les points que vous jugez intéressants, et qui vous ont aidé dans votre session de recherche.

B.3. Traitement Linguistique

Une sorte de lemmatisation a été appliquée sur les termes de requête dans le prototype, ainsi les termes ont été traités selon les règles dans le Tableau 1.

Traitement Linguistique	Exemples
Supprimer la marque du pluriel	Ponies → poni -ties → -ti caresses → caress caress → caress cats → cat
Supprimer la marque de « -ed »	feed → feed agreed → agree disabled → disable
Supprimer la marque de « -ing »	matting → mat mating → mate meeting → meet milling → mill messing → mess meetings → meet
Remplacer « -Y » par « -i »	
Supprimer les doubles suffixes : par exemple « -ization » = « -ize » plus « -ation » on supprime « -ation » pour obtenir « -ize »	-ational → -ate -tional → -tion -enci → -ence -anci → -ance -izer → -ize -bli → -ble -alli → -al -entli → -ent -eli → -e -ousli → -ous -ization → -ize -ation → -ate -ator → -ate -alism → -al -iveness → -ive -fulness → -ful -ousness → -ous -aliti → -al -iviti → -ive -biliti → -ble -logi → -log
Traiter les suffixe « -ic- » , « -full », « -ness » …etc. de façon similaire aux suffixes précédents.	-icate → -ic -ative → - -alize → -al -iciti → -ic -ical → -ic -ful → - -ness → -
Supprimer « -ant », « -ence » …etc.	al, ance, ence, er, ic, able, ible, ant,ement, ment, ent (si elle n'est pas précédé par m), ion, ism, ate, iti, ous, ive, ize, e,l

Tableau 1. *Synthèse des règles de traitement linguistique pour les requêtes en anglais.*

B.4. Résultats Détaillés des Expérimentations

Les tableaux (de 2 à 10) suivants présentent les différents résultats obtenus à partir des expérimentations effectuées.

		R_{qi}	$Rjugé\text{-}per_{qi}$	$Rjugé\text{-}non\text{-}per_{qi}$	$Rnon\text{-}jugé_{qi}$
u_1	q_1	16	10	6	-
	q_2	14	5	9	-
	q_3	16	8	6	2
	RI_1	46	23	21	2
u_2	q_1	18	10	7	1
	q_2	18	12	6	-
	q_3	14	7	7	-
	q_4	16	11	4	1
	RI_2	66/64 différents	40/ 38 différents	24	2
u_3	q_1	12	5	6	1
	q_2	16	7	9	-
	q_3	16	4	10	2
	q_4	18	4	13	1
	RI_3	62	20	38	4

Tableau 2. *Résultat de recherche individuelle de u_1, u_2, u_3.*

	Doc	*Rjugé-per*	*Rjugé-non-per*	*Rnon-jugé*	**Jugé différemment** *jugé per/non-per*
***RI* :** $(RI_1 \cup RI_2 \cup RI_3)$	163	76	76	7	4
$(RI_1 \cup RI_2 \cup RI_3) \mid (RI_1 \cap RI_2 \cap RI_3)$	154	75	75	4	-
$RI_1 \cap RI_2 \cap RI_3$	9	1	1	3	4

Tableau 3. *Résultats des recherches individuelles des trois utilisateurs mis en commun.*

	R_{qi}	$Rjugé\text{-}per_{qi}$	$Rjugé\text{-}non\text{-}per_{qi}$	$Rnon\text{-}jugé_{qi}$	**Jugé différemment** *jugé per/non-per*
q_1	16	3	11	2	-
q_2	18	6	11	1	-
q_3	18	8	10	-	-
RC_1	52/43 différents	17/13 différents	32/26 différents	3/2 différents	2
q_1	12	10	2	-	-
q_2	15	7	5	3	-
q_3	18	10	7	1	-
q_4	18	11	7	-	-
RC_2	63/59 différents	38/35 différents	21/19 différents	4	1
q_1	18	3	13	2	-
q_2	16	7	9	-	-
q_3	18	3	15	-	-
q_4	16	1	12	3	-
q_5	12	3	9	-	-
RC_3	80/77 différents	17/14 différents	58/57 différents	5	1

Tableau 4. *Résultat de recherche collaborative de u_1, u_2, u_3 dans le $groupe_1$.*

	Doc	*Rjugé-per*	*Rjugé-non-per*	*Rnon-jugé*	**Jugé différemment** *jugé per/non-per*
RC : $(RC_1 \cup RC_2 \cup RC_3)$	116	30	57	8	21
$(RC_1 \cup RC_2 \cup RC_3) \setminus (RC_1 \cap RC_2 \cap RC_3)$	112	30	56	8	18
$RI_1 \cap RI_2 \cap RI_3$	4	-	1	-	3

Tableau 5. *Résultat de la recherche collaborative du $groupe_1$.*

Les requêtes soumises par les utilisateurs pendant la recherche individuelle	**Les requêtes soumises par les utilisateurs pendant la recherche collaborative**
e-business solutions online catalogue	ecommerce net business
e-commerce net business	business ecommerce net
electronic commerce online payment	online payment accepted cards
eBusiness	ecommerce internet payment
ecommerce	problems e-commerce payment
Electronic Payment	business ecommerce net
Internet Payment +ecommerce	internet payment
Internet payment e-trade	internet problems payment
electronic payment problems	problems e-commerce payment
e-commerce payment	report internet payment
electonic trade payment	ecommerce net business
-	ecommerce internet payment

Tableau 6. *Synthèse des requêtes des recherches individuelles et collaborative des u_1,u_2, u_3.*

Termes utilisés dans les requêtes	Fréquence dans les recherches individuelles	Fréquence dans la recherche collaborative
business	1	4
online	2	1
e-commerce, ecommerce	4	8
net	1	4
Payment	7	8
Internet	2	5
problems	1	3
accepted	-	1
Cards	-	1
Report	-	1
solutions	1	-
catalogue	1	-
Electronic	4	-
trade	1	-
commerce	1	-
e-business, eBusiness	2	-
e-trade	1	-

Tableau 7. *Synthèse de fréquence des termes dans les requêtes des recherches individuelles et collaborative de u_1, u_2, u_3.*

	Précision Rjugé-per$_{ui}$ / RI$_i$	***Précision Rjugé-per$_{ui}$ / RC$_i$***
u_1	0.5	0.07
u_2	0.59	0.59
u_3	0.32	0.18
Précision moyenne	0.47	0.26

Tableau 8. *La précision des u_1, u_2, u_3 dans les recherches individuelles et collaborative.*

		Jugement	Préférence	Requêtes		Résultats	
				Similaires	Différentes	Similaires	Différentes
u_1	utilisation	25%	25%	24%	3%	19%	3%
	questionnaire	13%	13%	20%	17%	20%	17%
u_2	utilisation	33%	33%	-	33%	-	-
	questionnaire	50%	33%	0	17%	0	0
u_3	utilisation	30%	6%	24%	10%	25%	5%
	questionnaire	22%	56%	-	11%	-	11%
u_4	utilisation	35%	20%	20%	25%	-	-
	questionnaire	-	27%	33%	13%	20%	7%
u_5	utilisation	67%	33%	-	-	-	-
	questionnaire	22%	26%	13%	9%	22%	9%
u_6	utilisation	53%	33%	13%	-	-	-
	questionnaire	19%	23%	12%	19%	19%	8%
u_7	utilisation	53%	41%	12%	6%	-	6%
	questionnaire	21%	26%	11%	16%	11%	16%
u_8	utilisation	10%	24%	7%	28%	-	31%
	questionnaire	20%	16%	20%	12%	20%	12%
u_9	utilisation	22%	39%	11%	11%	11%	6%
	questionnaire	29%	24%	6%	12%	12%	18%
u_{10}	utilisation	25%	31%	6%	15%	8%	15%
	questionnaire	32%	5%	5%	32%	5%	21%
moyen total	utilisation	33.5%	28.5%	11.7%	13.1%	6.3%	6.6%
	questionnaire	22.8%	24.9%	12%	15.8%	12.9%	11.9%

Tableau 9. *Importance des critères de soutien pour chaque utilisateur u_i.*

		Présentation des requêtes	Fusion des requêtes	Présentation des résultats
u_4	utilisation	75%	25%	-
	questionnaire	50%	17%	33%
u_5	utilisation	83%	-	17%
	questionnaire	25%	38%	38%
u_6	utilisation	60%	13%	27%
	questionnaire	33%	17%	50%
u_7	utilisation	41%	29%	29%
	questionnaire	40%	40%	20%
u_8	utilisation	59%	14%	28%
	questionnaire	38%	25%	38%
u_9	utilisation	50%	39%	11%
	questionnaire	50%	0%	50%
u_{10}	utilisation	33%	19%	48%
	questionnaire	29%	29%	43%
moyen total	utilisation	57.29%	19.86%	22.86%
	questionnaire	37.86%	23.71%	38.86%

Tableau 10. *Importance des types de soutien pour chaque utilisateur u_i.*

Résumé :

L'arrivée « World Wide Web » et la grande utilisation d'Internet ont ouvert de nouvelles perspectives pour la diffusion et le partage d'information. Nous présentons dans cette thèse une approche collaborative pour la recherche d'information.

Nous définissons un environnement collaboratif dédié à un groupe d'utilisateurs désirant effectuer ensemble des recherches d'information sur un sujet commun. Cet environnement aide l'utilisateur à obtenir une meilleure qualité de l'information, et à « bénéficier » efficacement des expériences des autres membres du groupe. Différents contextes dans lesquels le soutien s'avère nécessaire sont analysés : l'aide à la formulation de requête, l'évaluation de sa recherche par rapport aux autres membres du groupe et une vision globale des résultats obtenus.

Nous avons conçu et validé expérimentalement un modèle de soutien souple et adapté à la typologie de l'utilisateur pris individuellement et à la typologie du groupe d'utilisateur pris dans son ensemble. Le soutien est aussi personnalisé selon ses désirs et souhaits exprimés au moyen, entre autres, de critères de personnalisation de soutien.

Mot clés :

Recherche d'Information Collaborative, collecticiel, environnement de soutien personnalisé. Typologie des utilisateurs et du groupe, fusion de données.

Abstract:

The existence of "World Wide Web" and the great Internet use gave new prospects for the diffusion and the sharing/distribution of information. We present in this thesis a collaborative approach for the information retrieval.

We define a collaborative environment dedicated to the users group who wish/would to carry out together an information researches about a common topic. This environment helps the user to obtain a better quality of the information, and to "benefit" efficiently from the experiments of other group's members. It is proved that the support is necessary in several contexts, these contexts are analyzed like following: the support for the query formulation, the evaluation of user's research compared to the other group members and the global view of the results obtained.

We designed and validated experimentally a support model which is flexible and adapted to the user typology (taken individually) and to the user's group typology (taken as a whole). A set of criteria is defined to personalize the support according to user's wishes and contexts.

Keywords:

Collaborative Information Retrieval, Groupware/CSCW, personalised user support, user and group typology, data (query and result) merging.