1228

enssib

Ecole Nationale Supérieure
des Sciences de l'Information et des Bibliothèques

D E A

Sciences de l'Information et de la Communication

OPTION :
Systèmes d'Information Documentaire

MEMOIRE DE D.E.A

FILTRAGE DE L' INFORMATION SUR L'INTERNET

Auteur :
Mohammed BELKHEIR

Sous la Direction :
Jean-Pierre LARDY

BIBLIOTHEQUE DE L'ENSSIB
8024170

1996

Université Lumière
Lyon 2

Ecole Nationale Supérieure
des Sciences de l'Information et des Bibliothèques

Université Jean Moulin
Lyon 3

enssib

Ecole Nationale Supérieure
des Sciences de l'Information et des Bibliothèques

D E A

Sciences de l'Information et de la Communication

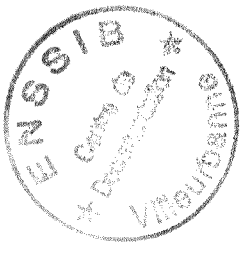
OPTION :
Systèmes d'Information Documentaire

MEMOIRE DE D.E.A

FILTRAGE DE L' INFORMATION SUR L'INTERNET

Auteur :
Mohammed BELKHEIR

Sous la Direction :
Jean-Pierre LARDY



1996

Université Lumière
Lyon 2

Ecole Nationale Supérieure
des Sciences de l'Information et des Bibliothèques

Université Jean Moulin
Lyon 3

TABLES DES MATIERES

CHAPITRE III LE FILTRAGE DE L'INFORMATION

CHAPITRE IV ALGORITHME DE FILTRAGE

CHAPITRE V Présentation de SIFT : Système de filtrage Online

CHAPITRE VI Présentation d'un filtre local : INFOSCAN

CHAPITRE VII Présentation d'un Agent : *Newt*

CHAPITRE VIII FILTRAGE PAR COLLABORATION

CHAPITRE IX EXPERIMENTATIONS

CONCLUSION

BIBIOGRAPHIE

INTRODUTION

Abstract

Information filters are essential mediators between information sources and their users. In most cases, both the information sources and the information users possess no mutual knowledge that can guide them in finding the information most relevant for the users momentary and long-term needs. Filters, wich are positioned logically as « *third parties* » to the communication between users and sources, should possess both the knowledge and the functionality to examine the information in the sources and to forward the information « *they judge* » as relevant to individual users.

Keywords

AGENTS, ARTIFICIAL INTELLIGENCE, ARTIFICIAL LIFE, COLLABORATIVE FILTERING, E-MAIL, EMPIRICAL STUDIES, EVOLUTION, FILTERS, GENETIC ALGORITHMS, INFORMATION NEEDS, INFORMATION RETREVIAL, INTERFACE AGENTS, INFORMATION OVERLOAD, INTERNET, NETNEWS, INFORMATION SCIENCE, SICIAL FILTERING.

Introduction

The number of networked users has increased rapidly with the proliferation of computers and networks. If the Internet is any indication, the number of people who have started using online services has increased dramaticaly in recent years. The number of machines wich have direct conectivity is over a million **[LOTTOR ; 92]** and is growing exponentialy. There are many more which connect to the Internet indirectly through intermediary services such as **America Online** and **Compuserve**. This explosive growth has fed the growth in the amount of information resources available over the networks. As more information becomes available, it becomes increasingly difficult to search for information. However, as the number of users increases, never users are likely to be less network savy. Getting information should become easier , not harder, if newer users are to be able to meet their information needs. It is, therefore, critically important to build tools that help users serve their information needs better.

Information Filtering deals with the delivery of the information that is relevant to the user in timely manner. An information filtering system assists users by filtering the data stream and delivering the relevant information to the users. Information preferences vary greatly across users, therefore, filtering system must be highly personalized to serve the individual interests of the user. A personilazed filtering system must satisfy three requirements :

- **Specialisation** : A personalized filtering system must serve the specific interests of the users. The system selects the articles deemed to be interesting to the user and eliminates the rest. However, a filtering system might not be able to perfectly differenciate the articles that are actually relevant to the users from the ones that are not. The propotion of irrelevant articles delivered to the user should be as low as possible. The proportion of relevant articles eliminated should also be low. Since filtering involves repeated interactions with the user, the system should be able to identify patterns in the users behavior. The filtering system must infer the habits of the user and specialize to them.
- **Adaptation** : Since filtering typically involves interactions over long periods of time, user interests cannot be assumed to stay constant. When they change, the system must first be able to notice that the users interests have changed. Secondly, the system must adapt its behavior in response to the change. Anticipating and adapting to user needs helps make the system more user friendly. This is essential if more and more people are to use it.
- **Exploration** : A filtering system should also be capable exploring newer information domains to find something of potential intererest to the user. There are two motivations for exploration. One is that exploration helps match a presently unknown but real user interest. The other motivation is that it helps improve the adaptation process. This is the case because newer kinds of information need to be explored to serve the changing user interests.

Résumé

Les systèmes de filtrage d'information sont des médiateurs indispensables entre les sources d'information et leurs utilisateurs. Dans la plus part des situations, Les outils de recherche d'information et les utilisateurs de cette information ne possèdent pas les connaissances qui les guideront, par la suite, à trouver des informations pertinentes pour l'usager à court ou à long terme. Les filtres, qui se positionne donc, comme le troisième partenaire dans un schéma de communication usager-source permettent de sélectionner les informations qu'il jugent pertinentes pour l'usager.

Mots clés

Agents, algorithme, algorithme génétique, besoins informationnels, courrier électronique, évolution, études empiriques, filtrage par collaboration, filtrage d'information, filtrage social, Intelligence artificielle, Interface, Internet, *Netnews*, *news*, sciences de l'information, vie artificielle.

Introduction

Le nombre d'utilisateurs des autoroutes de l'information a considérablement augmenté depuis la prolifération des ordinateurs personnels et la facilité de l'accès au réseau. Si on considère le nombre de machines connectées à l'Internet significatif, ces dernières années ont connu une explosion du nombre de personnes recherchant de l'information *online*. Le nombre de machines directement connectées dépasserai le million **[LOTTOR ; 92]** en plus d'autres personnes connectées indirectement grâce à des services intermédiaires tel que **America Online** ou **Compuserve**. Cette croissance explosive des utilisateurs de L'Internet a engendré beaucoup d'informations disponibles et en libre accès sur le réseau. Autant ce flot d'informations est disponible, si le nombre de diffuseurs et de chercheurs d'information continue à croître à ce rythme, un usager non expérimenté n'aura aucune chance de trouver ce qu'il cherche sans être submergé par des informations n'ayant aucun rapport avec ses besoins; c'est pourquoi il est urgent aujourd'hui de développer de nouveaux outils qui aideront l'usager à filtrer cette information.

Le filtrage d'information consiste à filtrer un flot d'informations et de ne présenter à l'usager que celles qui ont une chance de l'intéresser (informations pertinentes). Les besoins de l'usager peuvent varier, c'est pourquoi un système de filtrage doit être très personnalisé pour servir les intérêts individuels de l'usager. Un système de filtrage doit satisfaire trois conditions :

- **Spécialisation** : Un système de filtrage personnel doit servir les intérêts spécifiques des usagers. Le système sélectionne les articles qui ont une chance d'être intéressants pour l'usager et éliminer le reste. Il n'est pas obligé de différencier parfaitement les articles qui sont pertinents de ceux qui ne le sont pas mais la proportion des articles non pertinents présentés à l'usager doit être aussi faible que possible, de même pour la proportion d'articles pertinents éliminés. Comme le système a des interactions avec l'usager, il doit être capable de prédire son comportement .
- **Adaptation** : Très souvent, les centres d'intérêts de l'usager ne sont pas constants ; quand ils changent, le système doit, en premier lieu constater ce changement et, en plus, il doit être capable d'adapter son comportement à ces changements. Anticiper et s'adapter aux besoins de l'usager rend le système plus facile à utiliser, ceci est très important surtout s'il doit être utilisé par un grand nombre d'usagers.
- **Exploration** : Un système de filtrage doit être capable d'explorer de nouveaux domaines d'information. Ceci permettra d'une part, à l'usager de découvrir des domaines dont il n'avait pas connaissance auparavant, d'autre part, cela facilitera la tâche d'adaptation en cas d'éventuels changements dans le comportement de l'usager.

CHAPITRE I

PROBLEMES LIES A L'INFORMATION SCIENTIFIQUE & TECHNIQUE

Introduction

Avant d'entamer notre étude sur le filtrage de l'information sur l'Internet que nous avons consacré au filtrage de des articles de *news* qui concernent plus les domaines scientifiques et techniques, il nous a semblé important de situer les problèmes liés à l'information scientifique et technique. L' **IST** les connaissait bien avant l'arrivée de l'Internet qui n'a fait que les amplifier en « *court circuitant* » certaines étapes de sa diffusion.

Problèmes liés à l'information technique et scientifique

L'information dite grand public est caractérisée par son taux rapide d'obsolescence. Elle est faite d'événements, elle est substitutive, au sens que tout événement nouveau chasse par son caractère récent ou important l'événement plus ancien.

L'information scientifique et technique **IST** est cumulative, le présent intègre ou additionne le passé. Elle est la longue mémoire du progrès scientifique et technique : connaissance des lois de la nature, des hommes et des choses. Toutefois, elle n'est pas un produit neutre de qualité toujours égale, toujours parfaitement disponible au sens de son accès ou de sa compréhension.**[BORNES ; 1982]**.

Différents facteurs influent sur la qualité de l'information scientifique et technique, que ce soit directement par leur action sur l'information elle-même, ou indirectement par le jeu des médias que l'**IST** utilise .

1 Effets directs

La masse de l'information : Lorsque l'on dit accumulation, on introduit nécessairement avec le facteur temps un phénomène de quantité ; selon les auteurs, il y aurait un accroissement exponentiel de l'**IST** [1].

Les raisons de cette explosion sont de plusieurs ordres. On peut relever que

- Les sciences se démultiplient et chaque démultiplication devient un pôle de développement. Ce phénomène conduit à un nombre de plus en plus conséquent de professions scientifiques. En 20 ans, le nombre de chercheurs scientifiques aux Etats Unis a pratiquement doublé.
- Le volume des recherches recherche croît depuis une quinzaine d'années, ainsi, en Allemagne et au Japan, les rations dépenses- recherches sur le **PIB** sont passé respectivement de 1.4 et 1.5 en 1964 à 2.2 et 2 en 1980.

- L'accès de nouvelles nations à l'âge scientifique et technique augmente le volume de l'**IST**.

- Les chercheurs sont de plus en plus productifs, sous l'impulsion de ce qu'il est convenu d'appeler le « *le publishing game* ». Cette productivité croîtrait d'environ 4% par an. Si l'on prend les Etats Unis comme modèle, on relève que la croissance d'articles publiés dans les seuls journaux scientifiques et techniques a été de 100% entre 1970 et 1980.

Le scientifique est condamné à assurer et maintenir sa notoriété et donc à publier. Ceci n'est pas sans effet sur la qualité de l'**IST**.

L'aspect concurrentiel : Il contribue à la parution d'articles, de rapports, d'études..etc. Publier ou périr est une dure loi dans le domaine scientifique. Cette concurrence a été exacerbée par la création de l'indice de citation ; cette idée de 1927 a trouvé son aboutissement dans le « *Science Citation Index* ». La nécessité d'être citée conduit à multiplier les articles du même auteur. Cette multiplication aboutit à des articles de plus en plus courts qui finissent à la limite par n'être bien souvent que des variations à partir d'un même papier publiées dans plusieurs journaux.

L'originalité de l'information : Reconnaître comme originale une contribution scientifique ou technique, c'est avoir réglé de façon satisfaisante le problème de son évaluation.

L'expertise devient délicate lorsque le domaine est étroit. L'auto-citation par un scientifique peut être la traduction de son excentricité ou de sa solitude dans un champ de recherche qu'il vient d'ouvrir.

La faiblesse de ce contrôle a différentes illustrations. On a pu relever par exemple qu'un article avait pu être réédité à quelque temps de distance dans la revue qui l'avait

[1] On estime que la masse de l'IST disponible doublerait environ tous les 8 ans.

publié la première fois. Certains auteurs ont pu reprendre, sous leur signature, l'intégralité d'articles écrits par d'autres.

Une autre constatation plus anecdotique si elle ne met pas en cause l'originalité entraîne une certaine redondance de l'**IST**. Par exemple on relève l'émergence de façon quasi simultanée de résultats identiques par des chercheurs fort éloignés dans l'espace. Il arrive ainsi que les mêmes choses soient découvertes un nombre considérable de fois **[KERDERANT. 1981]**.

La fiabilité : La fiabilité d'un résultat voudrait que soient contrôlées les conditions d'obtention de ce résultat. La contre expertise conduit à refaire l'expérience en suivant les mêmes opérations contrôlées. Or, la nature de certaines découvertes l'interdit. Ainsi, qu'en est-il d'expérience qui ont duré plusieurs années mobilisant des moyens considérables et qui ne peuvent donc être renouvelées à l'identique.

La fiabilité peut être atteinte par les effets de course à la priorité. Devant les délais nécessaires pour l'édition d'articles, la puissante revue américaine, *Physical Review*, fonda *Physical Review Letters* avec un délai de parution de 3 à 4 semaines. Ceci a entraîné une telle accélération que certains papiers ont pu être acceptés sans contrôle suffisant. Pour aller plus vite même, des auteurs se sont mis à plusieurs ; ainsi a pu paraître dans *Physical Review* un article signé par 24 co-auteurs.

L'actualité : Malgré son caractère cumulatif, l'**IST** est, elle aussi, périssable. Son utilisation dans la pointe avancée da la recherche est directement liée à son caractère récent et à la rapidité de sa diffusion. Ceci est particulièrement sensible pour les domaines qui sont en développement rapide. L'**IST** de très haut niveau vieillit au point d'être non utilisable pour les scientifiques avancés. Mais si l'**IST** est influencée par toute une série de facteurs, il est bien évident que les médias qu'elle utilise pour son transfert ont aussi une action sur sa qualité.

2. Effets indirects

Il ne s'agit ici que d'un balayage extrêmement rapide et général sur :

L'écrit : La lenteur de l'écrit est un des élément qui vicie l'information en la rendant pour son lecteur non actuelle.

L'écrit permet la réflexion qui est un facteur de qualité mais nuit à l'actualité. Un livre met pour naître en matière scientifique et technique, 5 à 10 ans . L'article de revue peut facilement totaliser 18 à 24 mois en sciences physiques **[GARVEY et NAN LIN ; 1971]**. Il est clair que les nouvelles technologies appliquées à l'édition et la diffusion modifieront de façon considérable ces perspectives.

L'image : Elle apparaît de plus en plus comme un moyen de diffusion de l'**IST**[2]. Le problème rencontré dans ce genre d'information est sa difficulté de lecture. En effet l'analyse de contenu suppose une transposition d'un mode non linguistique à un mode linguistique avec tous les silences ou les bruits que peut créer une observation insuffisante ou trop subjective.

Le son : La transmission orale de l'**IST** permet d'accélérer la vitesse de diffusion vers des populations plus au mois réduites en nombre, en fonction de l'espace et le temps.

La qualité de cette transmission directe est au niveau de son contenu fortement influencée par les attitudes de l'émetteur, son comportement, en fonction de l'auditoire.

3. L'accès libre ou médiatisé

L'accès libre est une revendication de l'utilisateur mais qui se heurte à l'abondance même de l'**IST**. Cette masse complexe, considérable dans son état actuel, impose un

ensemble de procédures et de moyens, de supports et de traitements qui éloignent l'**IST** de ses utilisateurs potentiels.

Il faudrait préciser le contenu du terme « accès ». Il couvre aussi bien l'accès au document initial, primaire ou à un fac similé, que le document complet, un résumé ou simplement un signalement.

La masse d'information disponible impose son tri, sa préparation, son identification et enfin le repérage du document et les conditions de sa restitution[3].

4 La langue

Elle constitue un obstacle évident ; il n'y a pas à l'heure actuelle une langue scientifique unanimement reconnue et acceptée pour assurer le transfert de l'**IST**.

L'importance de cette difficulté est directement liée soit à la connaissance de la langue porteuse de l'information, soit à la qualité de sa traduction. Il est à noter que la langue prédominante est l'anglais ; toutefois certains pays considèrent leur langue comme un mode de protection supplémentaire du secret scientifique.

Les *jargons* scientifiques font obstacles à la compréhension. Ces systèmes de défense perçus au départ comme des facilités de communication sont devenus progressivement des barrières imperméables pour l'échange des connaissance **[BORNES ; 1982]**.

5 Le secret

Le secret est indispensable chaque fois que l'**IST** rencontre des préoccupations liées :

- à la concurrence scientifique.
- à la concurrence industrielle et commerciale ; innovations, monopoles, pénétration de marchés, stratégies industrielles.

---

[2] Application en imagerie médicale.

[3] Dans le cas de l'Internet, l' accès à l'information est directe, ce qui élimine bon nombre de ces traitements.

En se qui concerne la prise en considération des aspects politiques ou stratégiques dans le cadre des relations internationales, il apparaît que la circulation de l'**IST** se heurte aussi à toute une série d'obstacles qui peuvent concerner :

- La diffusion du « savoir faire ».
- La diffusion de licences et Brevets.

Depuis la généralisation de l'utilisation de l'Internet par les scientifiques, le monde de l'information n'a plus de frontières, mais le problème du flux trans-frontière demeure d'actualité.

CHAPITRE II

EVALUATION DES OUTILS DE RECHERCHE D'INFORMATION dans L' INTERNET

Introduction

1. Typologie des outils de recherche

Un outil de recherche se compose d'une partie technique et d'une partie fonctionnelle. La partie technique touche spécifiquement :

- *Le type de serveur* (**PC**, **Mac**,**Sun**).
- *Le système d'exploitation* (**UNIX**, **Microsoft Windows**).
- *Le nombre de serveurs* (sites miroirs).
- *Le type de bases de données* (fichiers plats, bases de données relationnelles ou orientés objets).
- *Le langage de programmation* (**C++**, **Pascal**).
- *Le langage d'interprétation* (**Perl**, **Tcl**).
- *Le moteur de recherche* (**WAIS**, **GLIMPSE**, **PURSUIT**, etc.).

Bien que très intéressante, cette partie ne fera pas l'objet de notre étude.

La partie fonctionnelle se compose essentiellement de trois parties :

- *Les services de mise à jour des données* : ont pour but de collecter les données du **Web** afin de constituer une base de données **W3**.
- *Les services de repérage* : ont pour mission de fournir une interface utilisateur doublée d'un moteur de recherche pour l'exploitation des données.
- *L'interface de présentation des résultats* : correspond aux fonctions offertes pour l'exploitation des données repêchées.

Les outils de recherche peuvent êtres classés selon plusieurs types. Il y a ceux qui produisent des index (**Lycos**) ou des répertoires (**Yahoo**), certains sont hybrides (**Infoseek**, **Alta Vista**).

1.1 Type index

Dans ce type d'outils, la fonction de repérage repose essentiellement sur l'utilisation d'index. Parmi ces outils on peut citer : **ALIWEB**, **CUI W3**, **Infoseek**, **Lycos**, **WWW Worm**, **Alta Vista** et bien d'autres services.

Un des avantages de ce type d'outils réside dans le fait que l'utilisateur n'a pas à connaître la catégorie et la structure hiérarchique dans laquelle pourrait se trouver l'information recherchée. La recherche s'opère principalement par concordance avec un modèle (*pattern matching*). Cette approche peut entraîner de bons taux de rappel mais aussi beaucoup de bruit[4] (**cf. résultats expérimentaux**).

1.2 Type répertoire

Ce type est associé à tous les outils de recherche dont les fonctions de repérage reposent principalement sur une classification afin d'organiser l'information selon une priorité thématique, chronologique ..etc. En général, c'est la classification de type thématique qui est privilégiée[5].

Lorsque l'utilisateur connaît le domaine de sa recherche et la structure hiérarchique liée à un outil de recherche, le type répertoire peut s'avérer très intéressant En catégorisant l'information, il est facile de butiner d'un document à l'autre portant sur un même sujet. Cette approche permet de réduire le taux de bruit mais aussi d'augmenter le taux de silence. Fondamentalement ce type est analogue au plan de classification que l'on retrouve dans les bibliothèques traditionnelles, mais en beaucoup moins élaboré. Notons également que, généralement le type répertoire est basé sur des techniques manuelles pour répertorier l'information produite dans le **Web**. En général il y a une valeur ajoutée. Bien que certains de ces outils ont la

[4] Ce type d'outils se rapproche de ceux utilisés pour la recherche par sujet dans les OPACs traditionnels, cependant contrairement à ces derniers , il n'y a pas de vocabulaire contrôlé.
[5] Dans ce type nous retrouvons, entre autre, les sites Einet Galaxy, WWW virtual Labrary, Yahoo..etc.

possibilité de recherche à l'aide d'index, celle-ci est beaucoup plus limitée que dans les outils de types index.

2. Les services de mise à jour

Les méthodes utilisées pour la mise à jour des bases de données **W3** sont principalement les robots, l'inscription, l'éditeur et le diffiseur-éditeur. Un outil de recherche peut utiliser l'une ou l'autre de ces méthodes, ou toutes à la fois.

2.1 Les robots

Les robots[6] sont sans doute les outils les plus répondus pour l'exploration du **Web**. Les sites tels que **Infoseek**, **Lycos**, **Open Text**, **WebCrawler**, **Alta Vista** les utilisent. Un robot est un programme qui s'exécute sur un ordinateur relié au ***Net*** et qui explore systématiquement celui-ci de manière à collecter l'information présente.

Avec le mécanisme de robot, le diffuseur de l'information joue un rôle complètement passif car le robot se charge de rechercher l'information diffusée sans demander l'autorisation à son propriétaire et visite lien par lien la toile du ***Web*** selon un algorithme de recherche. Cet algorithme suit généralement un de ces deux principes **[KOSTER ; 95]** :

- *Recherche en profondeur* : l'algorithme repose sur l'exploration récursive du premier lien hypertextuel rencontré.
- *Recherche en largeur* : l'algorithme repose sur l'exploration de tous les noeuds d'une page avant de descendre dans un des liens hypertextuels.

[6] Dans la littérature on trouve parfois plus évocateurs tels *spiders* (araignée), *worms* (ver de terre, se faufiler) ou *Web Wanderer* (vagabond du Web).

2.2 L'inscription

Plusieurs outils de recherche offrent aux utilisateurs la possibilité d'inscrire leurs publications. Parmi ceux-ci, nous retrouvons **Infoseek**, **Lycos**, **Open Text**, **WebCrawler**, **WWWWorm**, **Yahoo**...etc. Notons toutefois que le rôle du diffuseur de l'information primaire se limite en général à donner uniquement une adresse URL. Par la suite, l'outil de recherche visitera l'URL indiquée pour en extraire l'information.

2.3 L'éditeur

La méthode de l'éditeur est utilisée uniquement dans les outils de recherche de type répertoire. Des sites profitent de personnes qui se spécialisent dans un domaine précis et qui s'occupent d'en maintenir la cohésion.

Les site **Galaxy** et **Whole Internet Catalog** font partie de cette catégorie. Les diffuseurs sont également encouragés à proposer des sujets. Ceux-ci seront inclus dans la base de données uniquement sous l'approbation d'un responsable du domaine (*guest editors*). Evidement, contrairement aux techniques automatisées, cette technique est plus coûteuse en ressources humaines mais permet généralement une valeur ajoutée à l'information répertoriée.

2.4 Le diffuseur-éditeur

Cette méthode est très peu utilisée. En opposé à celle des robots, le diffuseur d'information joue un rôle plus actif. En effet l'outil de recherche est tributaire du bon vouloir des diffuseurs d'information primaire. Ce sont eux qui décident des objets à diffuser. Néanmoins, le diffuseur doit se conformer à une forme pour décrire l'objet de sa diffusion. Le seul site qui utilise cette méthode, à notre connaissance, est celui d'**Aliweb**. Pour tout objet de diffusion, le diffuseur prépare un fichier, selon des normes prescrites, puis le signal à l'outil de recherche qui se chargera de récupérer la description contenu dans le fichier pour l'inclure dans sa base. De plus, l'un des

services de l'outil de recherche visitera régulièrement le site **Web** pour prendre en compte toute modification éventuelle à la description.

Cette méthode comporte plusieurs avantages, le plus important concerne la description de l'objet à diffuser. Contrairement aux robots qui ne font qu'extraire les données du **Web**, cette méthode permet une valeur ajoutée par le diffuseur. Celui-ci peut, en effet, décrire l'objet diffusé, lui associer des mots clés et le classifier selon la norme émise. En contre partie, actuellement, **Aliweb** comporte peu d'éléments. (environ 6 000 documents[7]).**[PLOURDE ; 95]**

3. Les types d'objets répertoriés

Les types d'information retenue varie d'un outil de recherche à l'autre. Par exemple, la base de données **Lycos** est constituée uniquement de documents de type **Web**, **GOPHER** et **FTP**. Les fichiers de types **Usenet** (*groupe de discussion*) ou **Telnet** sont tout simplement ignorés par ses robots. Pour obtenir de l'information d'une page **Web**, l'outil est certes intéressant. Par contre, si vous chercher de l'information produite dans un groupe de discussion, à la fine pointe de l'actualité, il est recommandé d'utiliser **Infoseek** ou encore mieux **Alta Vista**.(**cf. résultats expérimentaux**).

Un autre exemple est la base de données **Infoseek**. C'est une base à but lucratif mais qui permet l'accès gratuit à un sous-ensemble des produits offerts. Cette base est en effet, constituée de plusieurs bases de données. D'une part, elle offre des références à des pages **Web** et à des messages récents de groupes de discussion. D'autre part, **Infoseek** offre, contre rémunération, un accès à des données autres que celles du **Net**.

[7] Résultat d'une étude menée en 1995.

Plus de 80 périodiques informatiques[8], certaines agences de presse, la base de donnée **Medline** et plusieurs autres.

4. L'information retenue des objets répertoriés

A chaque objet répertorié correspond un enregistrement dans la base de donnée. Selon l'outil de recherche, la nature de l'enregistrement peut varier énormément. Au minimum, celui-ci contient l'adresse URL et le titre de l'objet répertorié. Il peut également, par exemple contenir les en-têtes, les mots clés et mieux encore, l'intégralité des pages **Web**[9]. Evidemment la qualité des services de repérages sera directement proportionnelle à la richesse de cet enregistrement.

La base de données **Open Text**, qui contient l'intégralité de chaque page **Web** répertoriée, associe à chaque document un extrait, un titre, le premier en-tête, l'URL et les liens hypertextuels. C'est de loin le site qui exploite le mieux la structure logique des documents **HTML [PINON ; 95]**.

WebCrawler indexe tout le contenu des documents répertoriés, mais n'exploite pas la structure logique de ceux-ci. Pire encore, **CUI WWW Catalog** indexe uniquement les titres des documents répertoriés.

Alta Vista, dernier né des moteurs de recherche, fonctionne sur le même principe que **Infoseek**, et offre en plus une interrogation du **Web** ou de **Usenet** séparément.

5. Fréquence de mise à jour de la base de données

Les données du **Net** sont très volatiles : chaque jour s'ajoutent de nouveaux documents, d'autres sont modifiés ou détruits (*dead link*). Particulièrement pour les

[8] Données 1995

outils de recherche qui utilisent les robots, il est important que ces bases soient constamment remises à jour. Par exemple la base de données **Lycos** est entièrement reconstruite chaque semaine. Environ 10 000 documents s'y ajoutent hebdomadairement. **Open Text** suit la même fréquence. **WebCrawler**, quand à lui remet à jour sa base toutes les six semaines.

Les bases de données **Yahoo** et **Harvest Broker** opèrent une reconstruction journalière de leur base respective.

6. Les fonctions de repérages

Nous avons vu dans le paragraphe précèdent que les types d'objets et l'information retenue varient d'un outil de recherche à l'autre. L'organisation structurelle des données a un impact direct sur les outils de repérage et sur la qualité des fonctions offertes. Restreindre l'indexation à certains champs améliora la précision des recherches. De même, la notion de pertinence est très importante en recherche d'information. C'est pourquoi plusieurs outils de recherche utilisent le principe de pondération sur les mots recherchés.

De plus, une panoplie d'opérateurs est utilisée pour affiner le repérage. Citons entre autres, les opérateurs booléens, les opérateurs de voisinage et les opérateurs de chaînes de caractères. Il existe des bases de données qui offrent des langages de recherche qui permettent l'utilisation d'expressions régulières.

Contrairement aux recherches automatisées dans les bases de données traditionnelles, plusieurs fonctions ne sont pas toujours offertes. Pour n'en citer que quelques unes

[9] Par exemple à chaque document W3 répertorié par Lycos correspond un enregistrement contenant le titre, une partie de l'en-tête, les 100 mots les plus signifiants, les vingt premières lignes du texte, la taille e octets, la taille en nombre de mots et le nombre de liens.

prenons les opérateurs numériques et l'historique[10] de recherche. Actuellement il est possible d'effectuer une requête comportant une comparaison de dates.

6.1 Les champs indexés

L'indexation du texte intégral constitue un moyen, critiquable certes pour le bruit résultant **[FISCHER ; 91]**, qui peut s'avérer intéressant. Par ailleurs, l'indexation selon la structure logique d'un document est un moyen efficace pour cibler de l'information. Par exemple, le titre, les en-têtes, le résumé ou les premières lignes d'un texte forment à coup sûr des entités pour lesquelles les mots qui s'y rattachent ont une plus grande pertinence. Mais ce ne sont pas les seules, en plus de la structure logique du document, il y a les noms d'URL et les noms de fichiers qui forment également des entités très pertinentes pour certains types de repérage.

Prenons l'exemple de **Open Text** qui offre quatre outils de repérage. Parmi ceux-ci, nous retrouvons le « *Power Search* » qui, selon plusieurs analystes **[LIU ; 95]**, est l'interface la plus flexible du Web. Cette base contient des documents intégraux et la structure logique de ceux-ci. Il est possible de faire des recherche sur le titre, le premier en-tête, l'extrait, les liens hypertextuels ou tout simplement le texte complet.

L'outil de recherche **WWWWorm** offre, quand à lui, quatre champs indexés pour la recherche : le titre, le nom du fichier **HTML**, l'URL, et les références à un URL (références croisées). Ce dernier type de recherche permet de déterminer, par exemple, quels sont les documents de la base de données **WWWWorm** qui contiennent un lien hypertextuel à votre page d'accueil. La recherche dans une URL offre la possibilité, par exemple, de rechercher des films en formats **MPEG** (l'URL se termine par « .mpg »).

[10] Les utilisateurs avertis utilisent généralement le book mark.

6.2 La pondération des mots

Une autre technique utilisée pour améliorer le repérage est d'associer un « *poids* » à chacun des documents répertoriés. Cette technique a plusieurs variantes. Un mot reçoit un poids plus au moins grand selon le nombre de fois et/ou les positions ou il apparaît dans le document. Par exemple un mot apparaissant dans le titre d'un document recevra un poids plus grand qu'un mot n'apparaissant que dans le corps du document. De même, un mot apparaissant deux fois recevra un poids plus grand qu'un mot n'apparaissant qu'une seule fois. Ces poids associés aux mots présents dans un document sont utilisés lors de la recherche pour calculer la *pertinence estimée* du document par rapport à la requête formulée. D'autres facteurs peuvent participer également au calcul, tels la proximité des termes recherchés dans le document et le fait qu'ils apparaissent dans le même ordre que la requête ou non. Une des méthodes les plus populaires pour calculer la *pertinence estimée* d'un document par rapport à la requête est celle de type **WAIS.[COURTOIS ; 95]**

Quelle que soit la méthode utilisée, la *pertinence estimée* est représentée par une *pondération numérique*, qui s'exprime habituellement sous la forme d'un nombre compris entre 0 et 100. 100 représentant la plus grande *pertinence estimée* possible. Ainsi, la pondération d'un document en réponse à une requête de recherche donne une certaine indication de sa pertinence probable par rapport à cette requête.

Prenons l'exemple de l'outils de recherche **Alta Vista**, celui-ci classe les résultats, c'est à dire les références aux documents repêchés, par ordre décroissant des pondérations calculées pour chaque document en fonction des termes recherchés (*relevancy ranking of terms*). En d'autres mots, à chaque document de la base est calculé un poids (*weight terms*). Cette pondération est calculée à partir du nombre d'occurrences des termes recherchés, de leur position dans le document (le titre, les premières lignes du texte..) et de leur proximité s'il y a lieu.

L'outil de recherche **WebCrawler** utilise aussi l'algorithme de pondération. Néanmoins cet algorithme est moins développé que le précédant car malgré l'indexation intégrale du texte, cette base n'utilise pas la structure logique du document (en-tête, citation..). La pondération repose uniquement sur le nombre d'occurrence du terme recherché sans tenir compte de sa position dans le texte.

6.3 Les opérateurs booléens

Chacun connaît l'importance, dans la recherche automatisée, des opérateurs booléens (**AND, OR, NOT**). Plusieurs outils de recherche offrent ce type de fonction. Cependant peu d'entre eux permettent une utilisation rigoureuse de ces opérateurs qualifiés de pseudo-booléens [**NOËL, 95**]. Les outils de recherche **Open Text Web Index** et **Einet Galaxy** , quand à eux, permettent une utilisation des opérateurs **AND**, **OR** et **NOT** de manière précise. L'outil de recherche **Lycos** permet de spécifier le nombre minimum de termes à être présents dans le document sans pouvoir déterminer lesquels parmi la liste de termes fournis (c'est un pseudo **ET/OU** booléen). Il est également possible d'utiliser la négation d'un terme mais cette négation n'est pas exclusive. En effet, les documents contenant le terme recevront tout simplement une pondération moins significative et apparaîtront plus loin dans la liste des résultats. En revanche la pluspart des outils de recherche considère uniquement un **Ou** implicite entre les termes de la recherche.

6.4 Les opérateurs de voisinage

En recherche automatisée, on utilise souvent des opérateurs de voisinage telles l'*adjacence* et la *proximité*. Peu d'outils de recherche offrent ce type de service.

La base données **Open Text Web Index** permet *l'adjacence*. En effet, la pondération calculée repose, en partie, sur la distance entre les termes.

6.5 Les opérateurs de chaînes de caractères

Les opérateurs de chaînes de caractères correspondent le plus souvent à ceux que l'on connaît dans le traitement de texte. On retrouve principalement les caractères génériques pour la troncature ou la concordance d'un masque (*wild card*), les sous-chaines (*searching substring*) et les phrases (*searching phrase*). De plus, certains outils de recherche permettent la recherche sur des mots apparentés (*approximative matches*). La distinction entre les majuscules et les minuscules est également supportée par quelques outils.

6.6 Les langages de recherche

Certains outils de recherche offrent des langages pour raffiner les recherches ou l'utilisation d'expressions régulières. Citons[11] **WWWWorm** qui utilise le programme « *egrep* » d'**Unix** et la base de données **CUI WWW Catalog** qui demande une connaissance du langage **PERL**[12].

7. Etudes expérimentales

Plusieurs études sur les performances des outils de recherches ont été publiées. Notre étude diffère de celles-ci car elle n'a pas pour objectif d'évaluer les performances de tel ou tel outil, mais elle nous permettra de mettre l'accent sur certains problèmes qui pourraient nous intéresser afin d'introduire notre étude sur le filtrage de l'information.

Cette étude n'aura aucune évaluation scientifique vue la taille de notre échantillon et la durée de son déroulement. Elle porte sur les trois outils les plus en vogues sur le **Web** : **Infoseek**, **Lycos** et **Alta Vista**. Il est à noter que notre objectif n'est pas de comparer leurs performances. Elle permettra à l'usager d'avoir une idée simple sur certains aspects des problèmes liés à la recherche d'information sur l'Internet.

---

[11] Ces outils sont très peu utilisés

[12] L'investissement en temps pour l'apprentissage pourrait en décourager les utilisateurs.

7.1 Description des outils de recherche

7.1.1 Infoseek

Infoseek est à but lucratif. Heureusement un accès gratuit est disponible à une partie des données. C'est l'un des outils de recherche le plus apprécié dans plusieurs études comparatives.

Bien qu'étant de type index, il offre également une classification générée automatiquement. On peut considérer cette base comme un hybride index-répertoire. La recherche s'opère sur les noms de fichiers et leur contenu. Il est également possible de consulter une douzaine de sujets (*topics*). L'utilisateur peut spécifier le type d'objet désiré soit des pages **Web**, des groupes de discussion ou des **FAQ** (***Frequensy Asked Question***) dans l'Internet.

Plusieurs fonctions sont disponibles pour le repérage. L'utilisation de guillemets permet de spécifier des phrases pour la recherche. De plus, **Infoseek** porte une attention aux noms propres (les mots en majuscule) de manière à les distinguer des noms communs. Des opérateurs de voisinage, tels que l'*adjacence* et la proximité, sont également offerts. Par contre, il n'y a pas d'opérateurs booléens mais néanmoins l'utilisateur peut spécifier des mots obligatoire (+) ou en proscrire (-).

En sortie, on obtient, en plus du titre, la pondération, la taille et le type de document, une description, des références croisées et le tout par ordre décroissant de pondération. Un particularité intéressante d'**Infoseek** est la fonction *Find similar pages* qui, telle que son nom l'indique, permet de rechercher d'autre ressources similaires à un résultat obtenu.

Caractéristiques :

- *Mode opératoire de mise à jour* : Robot et inscription par courrier électronique.
- *Objets répertoriés* : pages **W3**, pages Newsgroupes, FAQ Reviewed pages (ces pages sont répertoriées dans des *topics*) et les *topics* (sujets).
- *Opérations* : L'adjacence, la proximité (à l'intérieur de 100 mots), mots obligatoires, mots non-désirés, phrase, utilisation la virgule et des majuscules pour distinguer les propres, recherche à l'intérieur d'un sujet.
- *Fréquence de mise à jour* : mensuelle.
- *Taille* : 400 000 documents W3.
- *Divers* : - L'utilisation de majuscules indique que l'on recherche des noms propres.
  - On ne peut pas utiliser les opérateurs booléens.

7.1.2 LYCOS

C'est l'outil de recherche le plus cité dans les études comparatives. Tout comme **Infoseek**, cet outil est à but lucratif. Par contre son utilisation est gratuite et ses revenus proviennent de sources publicitaires.

Lycos est constitué de deux bases[13] : une petite d'environ 500 000 URL, où chaque URL possède une description, et une grande d'environ 10.75 millions d'URL qui contient également les éléments de la petite base. Les éléments répertoriés sont de types **Gopher**, **FTP** et pages **Web**.

Les fonctions de recherche permettent des opérateurs pseudo-booléens (**ALL/ANY** en spécifiant un nombre de termes) ainsi que la négation. Le moteur de recherche s'appelle **PURSUIT**. Tout élément de la base est pondéré selon la position de celui-ci dans les textes. Par exemple, un mot situé dans le titre ou dans le premier paragraphe d'une page aura un plus grand poids. il est possible aussi de spécifier le degré de précision des termes recherchés (*loose, fair, good, close, stong match*).

En sortie, on peut préciser le nombre maximum de résultats et trois formats d'affichage. A chaque paramètre de recherche est associé le nombre de documents trouvés. De plus, pour chacun des résultats avec le format complet, il y a l'URL , le titre, les 200 premiers caractères dans les zones de types en-tête, les 100 mots les plus signifiants, les 20 premières lignes ou 20 % du document (le plus petit des deux) et d'autres informations tels que la pondération, le nombre de termes trouvés et le degré d'adjacence.

Caractéristiques :

- *Mode opératoire de mise à jour* : Robot et inscription.
- *Objets répertoriés* : pages **W3**, **Gopher** et **FTP**.

- *Information retenue* : Titre, en-tête, URL, 100 mots les plu signifiants du documents, liens hypertextuels (citation) et les 20 premières lignes.
- *Taille* : Il y a deux bases. La première, la petite, est constituée d'environ 500 000 références, alors que la seconde, la grande, est constituée d'environ 10 millions d'entrées.
- *Fréquence de mise à jour* : hebdomadaire
- *Moteur de recherche* : **PURSUIT**
- *Opérations* : pseudo-et et pseudo-ou (**ALL/ANY**), pseudo négation.

7.1.3 Alta Vista

Lancé par la société Digital Equipment en décembre 1995, c'est le service le plus récent. Il a ainsi bénéficié de l'expérience des autres et de la puissance de l'entreprise. Nous ne sommes plus en présence d'un prototype développé dans un laboratoire universitaire mais d'un véritable produit industriel. Il se veut le plus complet et effectue une indexation du texte intégral ce qui donnerai 11 milliards de mots trouvés

[13] Les articles récents ne mentionnent pas ces deux bases, nous n'avons pas pu vérifier si elles existent toujours.

dans 22 millions de pages **W3**. D'autre part il indexe en temps réel le contenu complet des messages de plus de 13 000 groupes de *news*.

- **Alta Vista** offre deux modes de recherche :
  - *Recherche simple* : il suffit de rentrer les mots les uns à la suite des autres.
  - *Recherche avancée* : on a le choix entre les serveurs **W3** et les groupes **USENET**.

- Il faut obligatoirement utiliser les opérateurs **AND**, **OR**, **NOT** ou **NEAR** pour combiner plusieurs termes.
- Des guillemets permettent d'encadrer un mot composé. Un signe + accolé à gauche d'un terme le déclare obligatoire, alors que le signe - indique le terme à refuser.
- Le logiciel fait la différence entre majuscules et minuscules.
- Il est possible de rechercher tous les documents ayant un lien vers son serveur **W3**.
- On peut limiter le recherche aux mots du titre, aux URL, aux liens contenus dans un document.
- Enfin, la limite par date est présente ce qui est assez rare.

Remarque : Pour les résultats expérimentaux, cf. expérimentation

CHAPITRE III

LE FILTRAGE D'INFORMATION

Introduction
1.Etude préliminaire
2.Recherche et filtrage d'information
- **2.1 Recherche d'information**
- **2.2 Filtrage d'information**

3. Approche pour une adaptation entre Recherche et Filtrage d'information
4. Les Agents (Software Agent)
- **4.1 Définitions**
- **4.2 Agents pour le filtrage d'information**

Introduction

Ces dernières années ont connu un accroissement très important de l'information électronique, depuis l'arrivée de l'Internet ce flot ne cesse de croître ce qui rend son contrôle et sa gestion une tâche très difficile. De plus en plus de personnes cherchent ou diffusent de l'information sur le réseau. Même s'il existe des outils de recherche performants, le problème qui se pose aujourd'hui n'est plus comment trouver de l'information mais comment éviter d'être submergé par ce grand flot qui rend la tâche de l'usager de plus en plus difficile **[LARDY ; 96]**. Il serait donc urgent de développer des systèmes capables de filtrer cette information et de ne présenter à l'usager que l'information dont il a réellement besoin.

Un tel **SFI** (Système de **F**iltrage d'**I**nformation) doit accomplir deux fonctions essentielles : d'une part, éliminer les informations non pertinentes pour l'usager d'autre part, ne présenter que les informations qui ont la plus grande chance de l'intéresser. Le filtre est le médiateur entre les sources d'information et leurs usagers. **BELKIN** et **CROFT [7]** font une comparaison entre un système de filtrage et un système de recherche d'information, en mettant l'accent sur les points communs et les différences entre les deux systèmes, ils concluent leur article en considérant le filtrage d'information (*information filtering*) comme une spécialisation de la fonction de recherche d'information (*information retrevial*).

Le contexte de filtrage d'information concerne un flot dynamique d'information tel que les informations que l'on trouve dans l'Internet, par opposition à un système de recherche classique qui concerne des informations statiques tel que celles que l'on trouve dans les bases de données traditionnelles. Du fait de la nature dynamique de l'information, un système de filtrage concerne souvent des informations semi ou non structurées, et un volume beaucoup plus important.

Le filtrage d'information implique des interactions répétitives avec l'usager qui a besoin d'être tenu à jour dans son domaine d'intérêt. Par opposition à un système de recherche d'information où, dans la plus part des cas, le besoin de l'usager peut être satisfait en une seule étape de consultation. Ceci implique que le système de filtrage doit suivre l'évolution des besoins de l'usager.

Contrairement à un **SRI** (Système de **R**echerche d'**I**nformation) qui gère des requêtes, un **SFI** gère des profils qui sont des représentations de l'usager et de ses besoins.

Un **SFI** est généralement utilisé par une communauté contrairement à un **SRI** qui est spécifique aux besoins d'un usager.

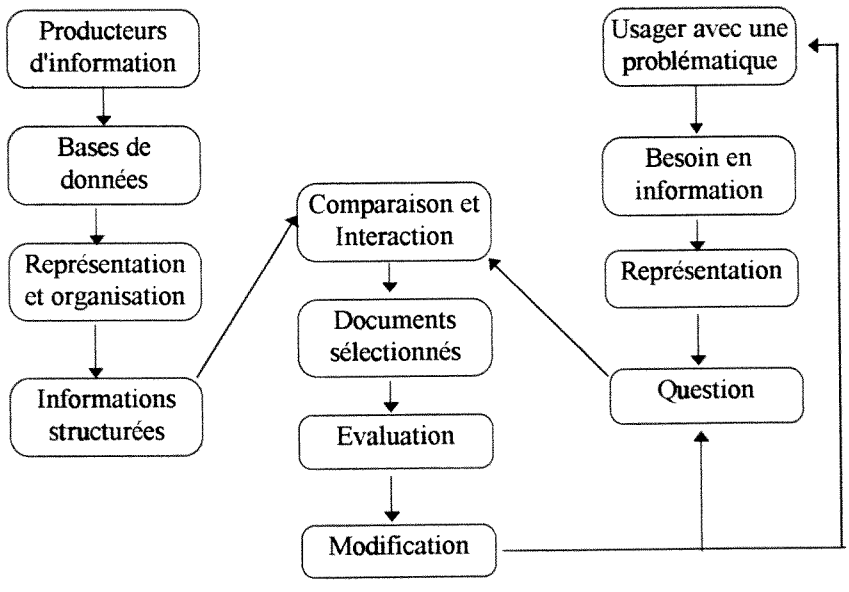


Fig.1 Modèle de recherche d'information

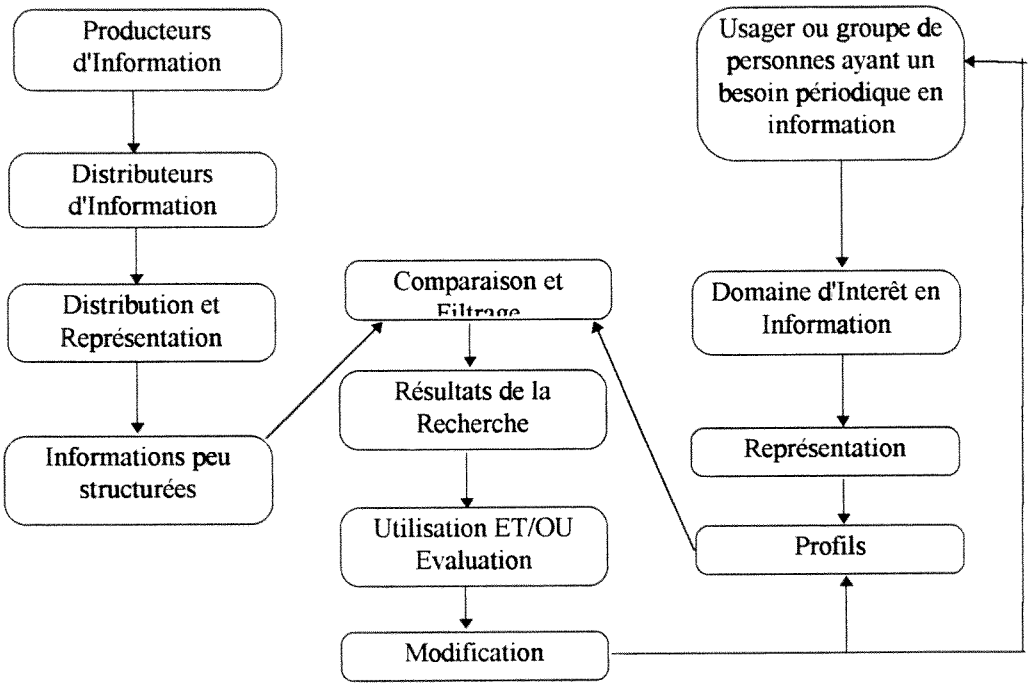


Fig.2 Modèle de filtrage d'information

1. Préliminaire

Le travail présenté dans ce mémoire se situe à l'intersection de deux domaines de recherche : le recherche d'information (*information retrevial*) et les Agents (*Intelligent Agent*).

La recherche d'information fait partie du domaine des sciences de l'information. Son but est d'extraire à partir d'une large collection de documents ceux qui ont la plus grande chance de répondre aux besoins de l'usager, La littérature sur la recherche d'information essaye aujourd'hui d'intégrer des facteurs tel que la rétroaction (*feedback*), le profil de l'usager et les caractéristiques du document **[LAINE ; 96]** ce qui la rapproche du filtrage d'information **[1,2,3]**. En revanche, le travail sur les Agents relève du domaine de l'**I**ntelligence **A**rtificielle (**AI**). Les travaux sur les

Agents concernent la conception d'automates intelligents capables d'effectuer un ensemble de tâches répétitives.

2. Recherche et filtrage d'information

2.1 Recherche d'information

On peut distinguer trois modèles de recherche d'information **[MARCUS, 91]** :

- Modèle statistique
- Modèle sémantique
- Modèle structurel

Le premier modèle tient compte de la fréquence des mots et de leurs localisations dans les documents. **Salton [4,5]** décrit l'utilisation des schémas statistiques tels que les probabilités et le modèle vectoriel pour la représentation des documents et de la requête. **Smart [6]** est un exemple de système de recherche basé sur le modèle vectoriel. **LSI** (**L**atent **S**emantic **I**ndexing) est autre exemple de système statistique qui tient compte de l'association des termes dans le document. Le second modèle, le modèle sémantique caractérise les documents et les requêtes pour présenter leur sens **[7,8]**, il est basé sur le traitement du langage naturel. Quand au troisième modèle, il prend les avantages structurels et contextuels disponible dans un système de recherche d'information ; par exemple, on peut utiliser un thesaurus dans lequel un grand nombre de mots ont été codés **[9]** ou bien prendre les avantages du contexte et de la structure généralement disponible dans les termes du document **[10]**. **CONIT [11]** en est un exemple qui utilise le modèle booléen généré par **Smart**.

Internet est considéré aujourd'hui comme la plus grande « base de données » où l'on peut trouver l'information la plus diversifiée. C'est donc le meilleur moyen pour tester les différentes techniques de recherche d'information, pour permettre à l'usager d'accéder à l'information diffusée par les différents serveurs dont le nombre ne cesse

de croître depuis quelques années. **WAIS**, **GOPHER** et **W3** les plus connus. **W**ide Area **I**nformation **S**ervers **(WAIS) [12]** est un service qui indexe les documents trouvés par les moteurs de recherche. Le service maintient à jour un index de mots clés qui seront utilisés pour une recherche efficace. **WAIS** permet à l'usager d'avoir une rétroaction afin de modifier sa requête et affiner sa recherche. **GOPHER [13]** est avant tout un outil qui permet d'organiser les documents en listes hiérarchiques mais permet aussi de trouver des informations en utilisant des indexes. Dans **WWW [14]**, l'information est organisée en hypertexte où l'usager peut explorer différents documents en utilisant les liens hypertextuels. Des moteurs de recherches **[BELKHEIR, 96]** permet l'accès direct à l'information par combinaison de mots clés : *requête*.

2.2 Filtrage d'information

Il existe aujourd'hui trois approches **[MALONE, 87]** dépendant de la manière dont les documents sont sélectionnés pour l'usager. Les systèmes de filtrage peuvent être classés en :

- système cognitif
- système social (collaboration)
- système économique

Dans le système cognitif, le choix des documents est basé sur les caractéristiques de leur contenu. Le système social sélectionne les documents suivant des annotations et des recommandations de l'usager (*filtrage par collaboration*) ; le système économique quand à lui, sélectionne les documents par calcul de critères au profit de l'usager en respectant des règles qu'i a défini auparavant.

Une grande variété d'approches a été établie pour décrire le contenu sémantique des documents. **Oval [15]** est un exemple de système qui utilise le principe des mots clés pour appliquer des règles définies par l'usager pour décrire les documents. **Foltz**

illustre l'utilisation du **LSI** pour le filtrage d'information et l'évalue pour des articles techniques de **Netnews [FOLTZ ; 90]**. **INFOSCOPE [16]** est un agent qui observe le comportement de l'usager et lui propose des suggestions. Il surveille le contenu des messages, détecte ce qui intéresse ou non l'usager, et effectuent des calculs statistiques qui lui permettront de prévoir dans tout changement dans le comportement de l'usager.

Le système social sélectionne les documents qui ont déjà été consultés et annotés par d'autres usagers. Les usagers collaborent avec le système et s'entraident pour filtrer les documents. Un comité de lecteurs conditionnels lit les nouveaux articles et les annotent, ces articles seront lus par la suite par d'autres lecteurs qui peuvent proposer d'autres annotations. Dans le service **GoodNews [17]**, la sélection des documents peut se faire aussi bien sur des critères personnels (lecture par collègue par exemple) ou collectifs (lecture par au moins la moitié du groupe) ; **Tapestry[18]** est un exemple d'environnement collaboratif qui accède à des articles en provenance de n'importe quelle source de *news*.

Les modèles cognitif et social sont tous les deux utilisés pour sélectionner les documents. La différence entre les deux dépend du domaine d'application où l'un peut être plus efficace que l'autre. Si, par exemple, l'information est cherchée par une certaine communauté de façon continue, le filtrage social est le mieux adapté ; tandis que si l'information est organisée en thèmes indépendamment de ses utilisateurs finaux, le système cognitif est le mieux adapté. Dans certains cas un modèle hybride est nécessaire.

Des services commerciaux de filtrage sont arrivés récemment sur le marché ; **First** est un service filtrage individualisé ; le profile de l'usager est défini par des personnes qualifiées après un entretien, les articles de *news* sont envoyés périodiquement. Le filtrage est géré par le service **Smart [19]**.

3. Approche pour une adaptation entre recherche et filtrage d'information

La différence entre filtrage et recherche d'information est relativement faible, tous les deux ont pour objectif de faciliter la tâche de l'usager dans sa quête d'information, **[BELKIN, CROFT ; 92]** cependant il y a certains aspects du problème qui ont été ignorés par la littérature de recherche d'information et qui prennent une importance considérable dans le domaine du filtrage. La recherche d'information s'intéresse aujourd'hui au phénomène de rétroaction (*feedback*), basée sur la reformulation de la requête initiale. La rétroaction a été introduite depuis longtemps **[4]**, elle a été utilisée dans le modèle vectoriel **[20]** où elle a donné des résultats assez satisfaisants **[21]**. Une autre approche adoptée dans la recherche d'information est l'utilisation d'algorithme génétique **GA** (Genetic Algorithm). **Yong Korphage [22]** a développé un algorithme de formulation des requêtes pour optimiser la recherche. **Gordon [23]** utilise la méthode des représentations compétitives (*competitive representation*) associées aux documents ; les représentations sont ensuite modifiées au cours du temps en utilisant le **GA**.

Les systèmes de filtrage sont souvent utilisés par une large communauté d'usagers, les intérêts de l'usager ne sont pas toujours bien décrits au système, de plus ils ne sont pas stables dans le temps **[BELKIN ; 92]**, le système doit donc respecter ses contraintes afin de répondre à ce besoin dynamique en information.

Baclace [24] propose un algorithme hybride pour la mise un point d'Agents qui peuvent filtrer les informations **IIF** (**I**nformation **I**ntake **F**iltering). Les Agents **IIF** utilisent un algorithme hybride, combinaison d'un algorithme génétique et économique.

Un grand nombre de techniques a été développé pour modéliser les intérêts de l'usager. **Doppelganger [25]** est un système de modélisation qui reçoit des

informations à partir d'un certain nombre de capteurs « *sensors* » par exemple le « *badge sensor* » transmet la localisation physique de l'usager tandis que le « *login sensor* » surveille les entrées et les sorties de l'usager à son poste de travail. Le système utilisera ces données pour prédire le comportement de l'usager.

4. Agent (*Software Agent*)

4.1 Définitions

Un Agent est système dont le but est d'accomplir un certain nombre de tâches dans un environnement complexe et dynamique. Il se place dans l'environnement et interagit à travers des capteurs « sensors» avec ses différents acteurs. Un agent opère automatiquement et améliore ses performances afin d'atteindre ses objectifs. Les Agents sont connus sous le nom de « *Software Agent* » ou « *Interface Agent* » **[26]**. Ces Agents sont des programmes qui automatisent des tâches répétitives pour fournir une assistance à l'usager pour une application bien définie.

L'idée d'utiliser les Agents pour leur déléguer un certains nombre de tâches de calcul remonte aux travaux de **Negroponte [23]** et **Kay [24]**. Les recherches dans ce domaine s'orientent aujourd'hui vers des systèmes avec de grandes capacités d'adaptation au comportement humain en acceptant ses différentes variantes afin de le substituer dans certaines de ses tâches.

Deux approches ont été traditionnellement utilisées pour désigner ces Agents. La première est que l'usager programme l'Agent en lui indiquant un ensemble de règles. **Oval system [25]** en est un exemple. Il permet à l'usager de programmer un ensemble de règles qui dictent le comportement de l'Agent. L'avantage de cette approche est qu'elle donne à l'usager un contrôle total sur le travail de l'Agent, ce qui lui permettra de lui léguer son travail en toute confiance si c'est lui même qui régit les règles ; cependant on lui reproche la lourdeur de la tâche de programmation. En effet, l'usager doit maintenir à jour l'Agent en lui communiquant tout changement. La

seconde approche est de programmer l'Agent avec un ensemble de règles régissant l'application et l'usager. **UCEgo [26]** est un assistant qui aide l'usager à utiliser le système **Unix** en lui proposant des suggestions afin de résoudre d'éventuels problèmes. L'avantage de cette approche est que l'usager n'est plus obligé de programmer l'Agent, on peut lui reprocher cependant sa rigidité. En effet, les programmes sont figés et ne peuvent pas avoir une utilisation individuelle. Il serait difficile pour l'usager de lui déléguer complètement son travail car il ne connaît ni son mode de fonctionnement ni ses limites.

Une troisième approche proposée par **Maes et Koziorok [27]** utilise les techniques d'apprentissage automatique (*machine learing techniques*). C'est une combinaison des deux précédentes et a pour but de construire des Agents qui acquièrent leurs compétences et s'adaptent aux besoins de l'usager dans un environnent où les tâches sont répétitives et où les différences individuelles sont plus marquées que les similarités.

L'agent peut apprendre en observant le comportement de l'usager. L'algorithme d'un tel Agent est décrit dans **[28, 29]**. Dans **[27, 30]** on peut trouver la description d'un Agent pour filtrer le courrier électronique.

Il existe aujourd'hui beaucoup de produits commercialisés : **Magnet** peut automatiser le classement selon les critères de l'usager, « **Open Sesame** » peut contrôler le clavier et la souris pour substituer l'usager dans certaines taches répétitives tandis que **Beynond Mail** permet de répondre à certains messages électroniques.

4.2 Agents pour le filtrage d'information sur l'Internet

L'Internet est un environnement idéal pour l'utilisation d'un Agent. Les informations y sont complexes et très dynamiques, les nouvelles informations arrivent de façon imprévisible ce qui augmente encore la complexité de leur gestion. De plus, les besoins

de l'usager peuvent être très variables, l'Agent pourra donc l'aider à profiter de cette richesse en information.

Un système de filtrage d'information nécessite pour l'application d'un Agent d'une part des interactions répétitives avec l'usager, d'autre part des différences dans les préférences individuelles.

L'idée proposée est de construire un ensemble d'Agents qui remplaceraient l'usager dans sa tâche da filtrage. Ces Agents doivent être sensibles à la rétroaction de l'usager et à toute modification de l'environnement de l'information. Ils doivent être automatiques et doivent agir en fonction du comportement de l'usager. Ils doivent s'adapter aux changements des préférences de l'usager. Le mécanisme d'apprentissage pour l'Agent est la rétroaction et l'algorithme génétique[14]. Les profils utilisés par le système sont constitués de termes qui sont en rapport avec le contenu du document. L'Agent utilise un filtrage cognitif. L'algorithme utilisé pour cet Agent et décrit dans le **chapitre III**.

Le mécanisme de régénération du filtrage est inspiré des recherches en algorithme génétique et intelligence artificielle **[31,32,33,34,35]**. Le filtrage d'information est effectivement un domaine en pleine évolution. Chercher dans un domaine très large et très dynamique implique un échange entre deux objectifs :

(*i*) exploiter les solutions déjà existantes
(ii) explorer de nouveaux domaines afin d'améliorer ces solutions.

L'algorithme génétique assure la combinaison entre exploitation et exploration et permet l'évolution artificielle du système grâce aux opérations de crossing-over et de mutation **[34]**.

[14] Voir chapitre III

CHAPITRE IV

ALGORITHME DE FILTRAGE

Introduction

1. Représentations

- **1.1 Documents**
- **1.2 Profils**

2. Filtrage des documents

- **2.1 Extraction des documents**
- **2.2Annotation des documents**
- **2.3 Sélection des documents**

3. La rétroaction

- **3.1 Rétroaction pour les documents**
- **3.2 Programmation par démonstration**

4. Algorithme génétique

- **4.1 Le crossing over (Crossover)**
- **4.2 La mutation**
- **4.3 La nouvelle génération**

1. Représentation

La représentation utilisée pour décrire les documents et les profils est basée sur le modèle vectoriel utilisé dans la littérature concernant les problèmes liés à la recherche d'information (*information retrevial*) **[SALTON; 83]**. Dans cette représentation, les documents et les profils sont représentés par des vecteurs dans un hyper-espace. La distance métrique qui mesure la proximité entre deux vecteurs est définie dans l'espace lui même. Quand un profil est défini, on lui associe une représentation vectorielle; les vecteurs documents qui ont la plus grande proximité avec ce profil lui seront associés. L'avantage d'utiliser un espace vectoriel commun aux documents et aux requêtes est d'une part, que la représentation d'un document peut être utilisée autant que profil et permettra de trouver tous les documents similaires à ce document; d'autre part, on peut utiliser la même notion de distance métrique entre deux documents, deux profils ou un couple document-profile. Cette propriété des espaces vectoriels pourrait être très intéressante dans le cas du filtrage d'information du moment que l'on peut remplacer le profil par un « *document modèle* ». Le système pourra construire des requêtes intelligentes (*intelligent queries*). Le système traitera donc de façon analogue les documents et les profils. La représentation de ces deux entités est décrite dans ce qui suit.

1.1 Le document

.Le texte est indexé de façon classique : on élimine les terme vides, on extrait les descripteurs ou mots clés, certaines phrases sont compactées et traitées comme un seul terme. Les termes doivent êtres ramenés à leur racine...etc. Au terme de ces opérations, le texte est représenté par un vecteur de termes qui peuvent être pondérés selon des critères pré -définis par le système.(fréquence d'apparition, occurrence, position dans le texte..)**[36]**.

$$T_i = < \omega_{ij} >$$

où ω_{ij} représente le poids du terme $\mathbf{t_j}$ dans le texte $\mathbf{T_i}$.

Les articles des *news*, par exemple, ne contiennent pas que du texte, d'autres informations tel que l'auteur, la localisation, la date, le nombre de pages...peuvent être utiliser pour le filtrage, il est donc nécessaire de leur associer une représentation vectorielle plus complète que celle décrite précédemment.

Dans le cas du filtrage, la représentation vectorielle est décrite comme suit : Le document est constitué de plusieurs champs (**F***ields*), le texte n'est entre autre que l'un de ces champs. Les autres champs peuvent inclure l'auteur, la localisation géographique, la date d'expédition ou de réception, le nombre de lignes.. ; il n' y a aucune restriction ni de contrainte au niveau du nombre de champs ; on peut définir autant de champs que l'on désire pour représenter les attributs du document.

A chaque champ on associe un ensemble de termes qui décrivent son contenu. Si ces descripteurs n'ont pas la même importance on peut leur attribuer un poids. Chaque champ sera donc présenté par un vecteur de termes :

$$F_i^d = < \omega_{ij}^d >$$

où ω_{ij}^d est le poids du terme $\mathbf{t_j}$ dans le champs (**F***ield*) F_i^d. L'indice **i** peut prendre les valeurs **a** (*auteur*), **k** (*keyword*), **l** (*localisation*) ou d'autres. L'indice **d** indique que ce champ appartient au document. (par opposition à l'indice **p** qui indique, comme nous le verrons ultérieurement l'appartenance au champ **P**rofil).

Un document classique tel qu'un article de news ou un message électronique *e-mail*, est constitué de plusieurs champs. Il sera donc présenté par une combinaison de vecteurs décrivant chacun des champs :

$$\mathbf{D} = \{ F_i^d \}$$

où F_i^d représente un champ du document **D.**

1.2 Le profil

La représentation de profil est similaire à celle du document. Le profil est constitué d'un ensemble de champs, tel que le *newsgroupe* à explorer, l'auteur, la localisation, la date...etc. Chacun de ces champs est représenté comme précédemment par un vecteur de termes, cependant il existe une différence entre les deux représentations. Dans le cas du profil, les différents champs décrivent les intérêts de l'usager, ils n'auront donc pas forcement la même importance, le champ date par exemple aura beaucoup d'importance pour un usager voulant assister un colloque se déroulant à une date précise. Il serait donc nécessaire d'attribuer à chaque champ un poids. La représentation du profil sera donc :

$$\mathbf{P} = \{ (F_i^p , \mathbf{W}(F_i^p))\}$$

où $\mathbf{W}(F_i^p)$ est le poids du champs F_i^p dans le profile **P**. L'indice **p** indique que **le** champ F_i^p appartient au profil et non pas du document. Chaque champ du profil est représenté par un vecteur de termes :

$$F_i^p = < \omega_{ij}^p >$$

newsgroups:	0.2	
	clari.news.cast	1
	clari.news.gov.international	1
	clari.news.hot.east_europe	1
	clari.news.hot.ussr	1
	clari.news.top.world	1
locations:	0.1	
learningrate:	0.05	
numarts:	5	
keywords:	0.7	
	ukraine	0.475
	nuclear	0.294
	weapons	0.290
	soviet	0.239
	tactical	0.224
	somewhat	0.224
	inherited	0.215
	encouraged	0.207
	missiles	0.201
	kiev	0.201
	funds	0.164
	additional	0.158
	returned	0.149
	washington	0.148
	...	

```
————ArtScores————
clari.news.gov.international:<boutrosghali-norkorURecb_3DP@clarinet.com>  0.0902
clari.news.gov.international:<year-diplomacyUR4c4_3DN@clarinet.com>       0.0458
clari.news.gov.international:<boutrosghali-japanUM8f9-22b@clarinet.com>   0.0457
clari.news.gov.international:<boutrosghali-koreaUR819_3DO@clarinet.com>   0.0452
clari.news.hot.east_europe:<year-republicsURaa1_3DN@clarinet.com>         0.0450
————ArtFeedback————
clari.news.gov.international:<boutrosghali-japanUM8f9-22b@clarinet.com>    +1
clari.news.gov.international:<boutrosghali-koreaUR819_3DO@clarinet.com>    -1
clari.news.hot.east_europe:<year-republicsURaa1_3DN@clarinet.com>          +1
————ArtRead————
clari.news.gov.international:<boutrosghali-japanUM147-22c@clarinet.com>
clari.news.hot.east_europe:<russia-polandUR979_3DD@clarinet.com>
clari.news.gov.international:<france-natoUR349_3DE@clarinet.com>
clari.news.hot.east_europe:<ukraine-russiaUR55b_3DE@clarinet.com>
clari.news.hot.east_europe:<russia-usUR90c_3DG@clarinet.com>
clari.news.top.world:<poland-walesaUM902-227@clarinet.com>
clari.news.hot.east_europe:<russia-germanyUM94b-22a@clarinet.com>
...
```

Fig.3 Exemple de profil

2. Filtrage des documents

Le processus de filtrage consiste à mettre les profils et les documents sous leurs formes vectorielles, trouver les documents ayant la même *similarité* que le profil et sélectionner les documents ayant le plus de chance de répondre aux besoins de l'usager pour les lui présenter.

2.1 Extraction des représentations des documents

Comme nous venons de le voir, les documents sont représentés par un ensemble de champs où chaque champ est un vecteur de termes (**cf. 3.1.2**). Tous les champs excepté le champ mots clés sont extraits de l'en-tête du document[15]. Le champ mot-clé quand à lui est extrait du texte.

On peut attribuer à chaque terme de n'importe quel champ un poids ; par exemple dans le champ localisation on peut trouver « China », « Taiwan », la représentation du champ localisation sera donc :

$$F_i^p = \langle y,y \rangle$$

où F_i^p est le champ localisation du document, y est le poids affecté par défaut[16] à t1= « China » et t2= « Taiwan ». Ces vecteurs seront normalisés. Le poids indique seulement l'importance relative des termes ; dans notre exemple, les deux termes ont le même poids.

[15] Dans notre cas les documents étudiés sont des articles de news .
[16] L'usager a la possibilité de modifier cette pondération en donnant plus d'importance à l'un des termes.

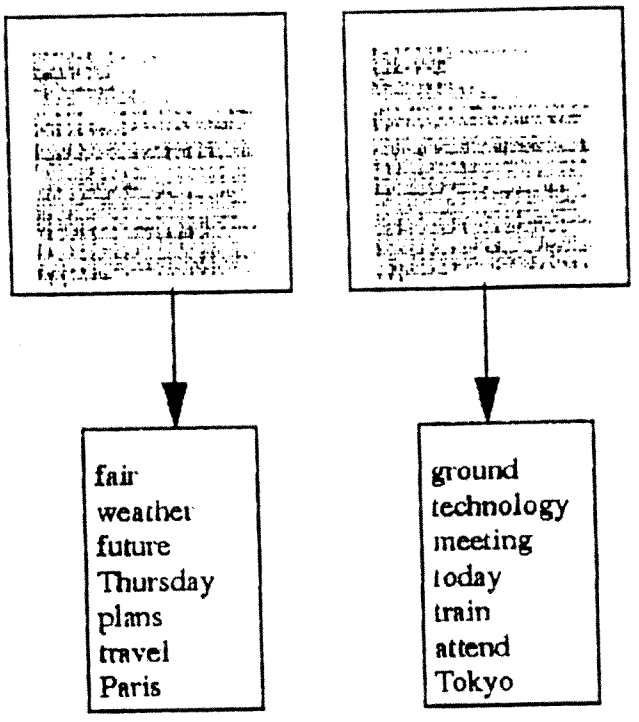


Fig.4 Extraction des mots-clés

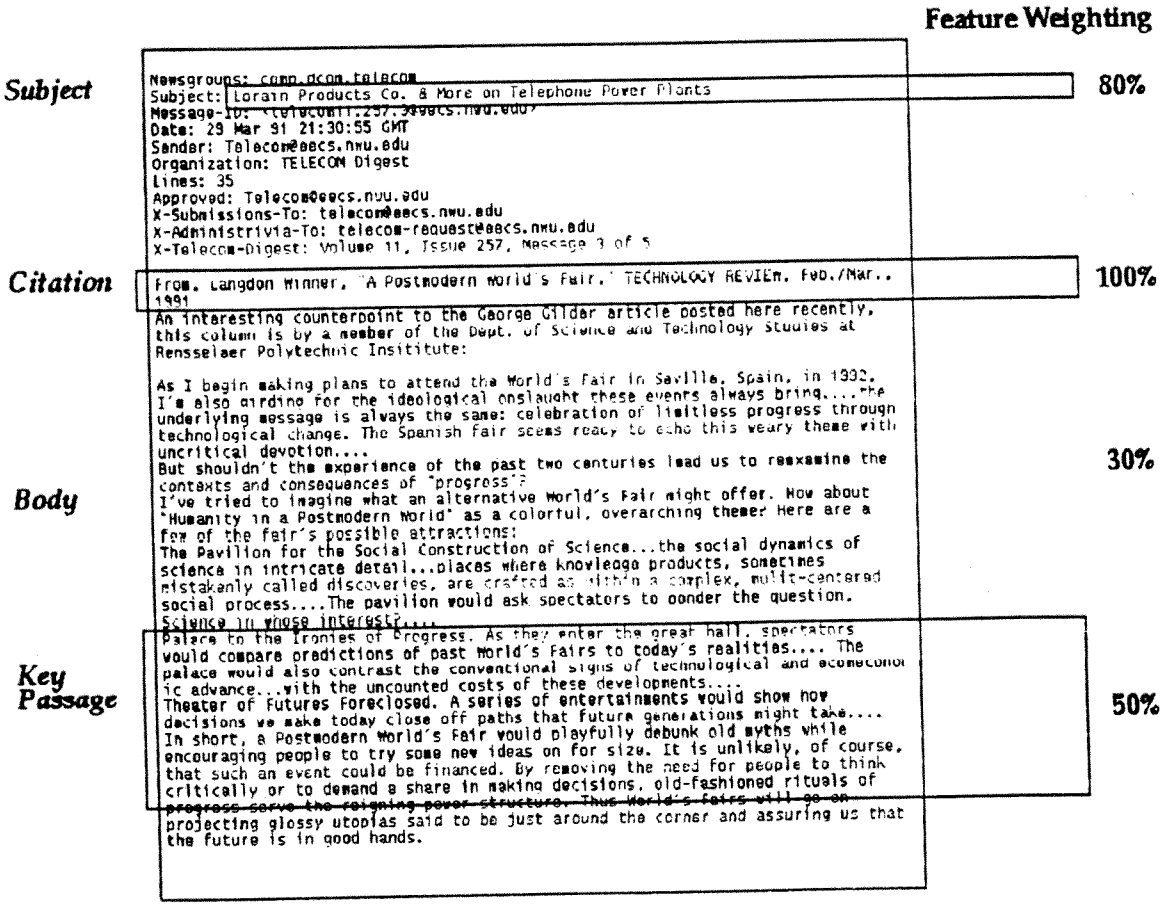


Fig.5 Pondération des champs

Le vecteur du champs mots-clés est obtenu par l'analyse du texte du document. Le poids d'un terme dépend de son occurrence et de sa fréquence d'apparition dans différents documents. C'est la méthode la plus utilisée dans les méthode d'indexation automatique **[SALTON ; 83]**.

Le poids d'un mot clé est le produit de sa fréquence par la fréquence de son document inverse. La fréquence d'un terme t_j est la fréquence d'occurrence du terme dans le texte, elle est généralement effective de l'importance du terme dans le texte. La fréquence du document inverse (*inverse document frequency*) **id*f*** est un facteur qui tient compte des termes qui n'apparaissent que dans un petit nombre de documents. Ce facteur permet de déclasser les termes qui n'apparaissent que dans un certains nombre de documents sans indication de l'usager. Le poids d'un terme se calculera comme suit :

$$\omega_{ik} = tf_{ik} \times idf_k$$

où tf_{ik} est le nombre d'occurrence du terme t_k dans le document **i**, et idf_k est la fréquence de documents inverse dans la collection considérée. On démontre que :

$$idf_k = \log(N/n_k)$$

où **N** est le nombre total de documents dans la collection dans laquelle n_k documents contiennent le terme t_k.

Le système évalue tous les articles dans lesquels certains *newsgroupes* sont mentionnés dans le champs *newsgroups* du profil. Il peut aussi chercher dans d'autres newsgroupes.

Ce sont les *newsgroupes* qui ont été mentionnés dans le *feedback* à travers la programmation par démonstration (**cf. 4.4.1**).

2.2 Annotation des documents

Dans la représentation vectorielle classique, les documents répondant à une requête sont ceux dont la distance métrique par rapport à la requête est minimale. La grandeur la plus utilisée pour représenter cette proximité est le cosinus de l'angle entre les deux vecteurs. Ceci peut être obtenu en calculant le produit scalaire entre les deux vecteurs :

$$\mathbf{S}(V_i,V_j) = \sum_k \omega_{ik}^{d} \times \omega_{ik}^{p} \qquad \textbf{(3.1)}$$

La même distance métrique est utilisée dans le cas du filtrage que l'on notera *Similarité* (*Similarity*). La *Similarité* entre le document et le profil dépendra de la similarité entre leurs champs correspondants. La *Similarité* entre deux champs est calculée comme indiqué en (**3.2**) :

$$\mathbf{S}(F_i^d, F_i^p) = \sum_k \omega_{ik}^{d} \times \omega_{ik}^{p} \qquad \textbf{(3.2)}$$

La *Similarité* entre le document et le profil se calcule comme suit :

$$\mathbf{S}(D,P) = \sum_i \mathbf{S}(F_i^d, F_i^p) \times \mathbf{W}(F_i^p) \qquad \textbf{(3.3)}$$

où les *Similarités* des différents champs, pondérés de leur poids dans le profil, sont ajoutées. Cette méthode de calcul pourrait favoriser les documents ayant des champs contenant des termes avec des poids très élevés, alors que le poids du champ est faible. Les annotations des champs doivent être d'abord comparées puis additionnés à moins qu'elles aient toutes la même valeur, ceci peut être obtenu en normalisant les vecteurs champs. Le produit scalaire de deux vecteurs normalisés appartiendra à l'intervalle **[-1,1]**, ce qui contraindra les scores à appartenir au même intervalle (avoir le même ordre de grandeur).

$$| F_i^d | = | F_i^p | = 1 \Rightarrow -1 \leq S(F_i^d, F_i^p) \leq 1 \quad \forall i \qquad (3.4)$$

l'avantage de cette normalisation est qu'il ne serait plus possible de comparer les poids des termes à ceux des champs. Ainsi si le poids d'un mot-clé t_i est très élevé par rapport à celui de l'auteur t_j ceci n'entraînera pas automatiquement que t_i contribuera plus que t_j dans l'annotation du document. Le poids de chaque terme exprime son importance relative par rapport aux autres termes du même champ. Le profil de l'usager peut inverser cette importance.

Le même problème peut être rencontré dans le cas où des documents auront le même score. Le système scrute différentes bases de données et annote les différents articles. Il collecte les articles ayant les scores les plus élevés et les compare au profile. Il ne serait pas possible de comparer les scores à moins qu'ils appartiennent à la même base. L'usager ne peut pas trouver de sens aux similarités des scores s'ils ne sont pas connus. C'est pourquoi, les scores des documents doivent appartenir à l'intervalle fermé **[-1,1]**.

Le plus grand score de 1 est assigné seulement quand les représentations vectorielles du document et du profil sont identiques.

$$S(P, D) = 1$$

Ceci entraîne si tous les champs sont identiques que le score de chaque champ est égal à 1 car par définition de la *Similarité* des champs et de la contrainte de l'équation **(3.4)**:

$$S(F_i^d, F_i^p) = 1$$

Ces deux équations entraîneraient :

$$\sum_i W(F_i^p) = 1 \qquad (3.5)$$

Les contraintes des équations **(3.4)** et **(3.5)** assurent que les scores des termes, champs et documents soient dans des échelles convenables.

2.3 Sélection des documents

Le système scrute une partie de la base de données et annote tous les documents qui y sont présents, les documents qui seront finalement présentés à l'usager seront sélectionnés parmi ceux ayant les meilleurs scores par rapport aux profils. Le paramètre utilisé pour sélectionner les documents, en plus de leurs scores, est leur ***finesse*** (*fitness*). La ***finesse*** est la mesure de l'aptitude qu'a eu le document dans le passé à répondre au besoin d'autres usagers. Il y a donc deux variables pour décider du document final. La première concerne le nombre de documents et la seconde leur classement. Il existe plusieurs approches pour sélectionner les documents finaux à présenter à l'usager :

- Le nombre de documents associés à chaque profil est proportionnel à sa ***finesse***. L'inconvénient de cette approche est que les documents ayant de faible score, mais une ***finesse*** élevée ont plus de chance d'être sélectionnés que des documents ayant des scores élevés mais des ***finesses*** faibles. (le cas d'un document qui vient d'être introduit dans la base de données).
- Le nombre de documents associés à chaque profil est le même, cependant le score des documents est corrigé par un facteur qui exprime leur ***finesse***. Ceci aura un impact sur l'ordre dans lequel les documents seront présentés à l'usager. L'avantage de cette approche est que l'usager pourra consulter les documents qui s'approchent le plus du profil avant les autres. L'inconvénient de cette approche est le fait que le nombre de documents associés à chaque profil est le même, certains documents avec un faible score peuvent être présentés à l'usager dans le cas où il n'y aurait pas assez de documents pertinents, ou bien des documents avec des scores élevés peuvent être éliminés dans le cas ou il y aurait un grand nombre de documents associés au profil.
- La troisième approche est l'utilisation d'un seuil. Tout document ayant un score supérieur à ce seuil sera présenté à l'usager indépendamment de sa distance du profil. L'avantage de cette méthode est que la qualité du document est assurée par le score. L'une des difficultés de cette approche est de fixer la valeur à attribuer au seuil.

Dans la pratique, la qualité d'un document dépend aussi bien de la définition du profil que du score du document. Chacune des approches paraît convenir à un cas particulier mais aucune des trois n'est fiable pour n'importe quel usager dans n'importe quelle situation. On pourrait envisager de combiner ces trois approches, mais ceci dépendra beaucoup des préférences de l'usager et du type de document à filtrer **[DILIP ; 94]**.

En pratique, il s'avère plus efficace de combiner les deux premières dans lesquelles le nombre de documents, leurs scores et leur classement dépend de la ***finesse*** du profil.

La troisième approche est abandonnée car le score d'un document n'a un sens que pour une comparaison des documents entre eux :*valeur relative*. Mais un document ne peut pas avoir un score absolu **[DILIP ; 94]**.

3.3 La rétroaction (*feedback*)

L'usager peut communiquer avec l'agent en émettant des réactions pour un article de deux façons. La première est d'émettre une rétroaction positive ou négative pour l'article proposé, la seconde est que l'usager peut fournir à l'agent un exemple d'article qu'il n'a pas trouvé. Cet article servira de modèle pour la programmation par démonstration (*programming by demonstration*). Dans les deux cas, la rétroaction de l'usager a deux effets : le premier est l'ajustement du profile au besoin de l'usager, le second est l'exploration de nouvelles sources.

3.1 Rétroaction pour les documents

La rétroaction a été utilisée pour mesurer les performances d'une fonction de recherche **[SALTON, 90]**. Dans le cas de la représentation en espace vectoriel, la méthode de reformulation permet l'ajustement de la requête. Dès que la requête et les documents sont dans leur représentation vectorielle, le vecteur requête est comparé aux vecteurs documents qui ont reçu une rétroaction positive ou négative afin d'en extraire ceux qui ont la plus grande chance d'intéresser l'usager.

Le mécanisme de rétroaction utilisé dans le filtrage est une généralisation de cette méthode. Chaque vecteur-champ du profil est modifié en fonction de la réaction de l'usager. Le processus de modification de chacun de ces vecteurs est identique à celui utilisé en recherche d'information (*information retrevial*). Considérant un profil **P** qui a permis l'extraction du document **D**. La rétroaction f est un entier positif ou négatif qui indique le niveau de rétroaction. Chaque vecteur-champ dans le profil est modifié comme suit :

$$\mathbf{P = P + \alpha \times f \times D} \qquad \mathbf{(3.6)}$$

où le poids de chaque terme dans chaque champ est modifié proportionnellement à un coefficient α et de la rétroaction f. α indique la sensibilité du profile à la rétroaction. L'équation **(3.6)** donne :

$$\forall \mathbf{i, k} \quad \omega_{ik}^{p} = \omega_{ik}^{p} + \alpha \times \mathbf{f} \times \omega_{ik}^{p}$$

où ω_{ik}^{p} est le poid du terme $\mathbf{t_k}$ dans le champs i du profile **P**. ω_{ik}^{d} est le poid du même terme dans le champ i du document **D**.

L'effet résultant de cette rétroaction sur les termes est la modification de leur poids. Les termes qui n'existaient pas déjà dans le profil seront rajoutés. La ***finesse*** du profil subit elle aussi des modification :

$$\mathbf{f(P) = f(P) + \beta \times f} \qquad \mathbf{(3.8)}$$

où β est un coefficient indiquant la sensibilité de la finesse à la rétroaction.

Dans un document sur lequel on a émis une rétroaction, on considère que cette rétroaction touche toutes les caractéristiques de ce document comme on peut le voir dans l'équation **(3.7)**. Si la rétroaction ne concerne que certains termes du document, seuls les champs correspondant seront modifiés ; par exemple si l'usager émet une préférence pour un auteur, seul le champ auteur sera modifié dans l'équation **(3.7)**. De la même façon l'usager peut avoir une rétroaction pour une partie du texte.

3.2 Programmation par démonstration (*Programming by demonstration*)

L'usager émet une rétroaction pour un document qui n'a pas été trouvé par aucun des profils. Une fois cette rétroaction enregistre, le système a deux possibilités pour agir : soit affecter ce document à un profil déjà existant en effectuant l'ajustement nécessaire. Dans le cas où il n'y aurait aucun profil proche de ce document, le système pourrait créer un nouveau profil.

S'il existe un profil **P** qui a déjà trouvé au moins un article du même *newsgroupe* que **D**, ce profil servira de modèle pour la recherche ultérieure. La justification est que si **D** appartient à un domaine déjà exploré par **P**, ce profil **P** est le plus apte à recevoir la rétroaction. La seul modification dans l'équation **(3.7)** est que le coefficient α aura un valeur beaucoup plus élevée. Seul inconvénient de cette approche, c'est que l'usager doit faire l'effort personnel de chercher des modèles de documents qui l'intéressent. Dans le cas où il n'y aurait aucun profil ne correspond à ce document, un nouveau profil est construit à partir de ce document. On obtient ainsi un profil décrivant ce document. La modification de la ***finesse*** est celle indiquée dans l'équation **(3.8)**.

4. Algorithme génétique

Comme nous venons de le voir, un Agent est constitué d'un ensemble de profils. Dans ce paragraphe nous nous intéresserons à l'étude des caractéristiques des profils pris ensemble, et leur comportement dans le cas de changement dans les intérêts de l'usager.

La représentation formelle de cette population est donnée dans l'équation **(3.9)**. Elle est définie comme un corps dans lequel chaque élément est un couple profil-*finesse*.

$$G = \{ (P, f(P)) \} \qquad (3.9)$$

La représentation du profil est la même que celle décrite dans le paragraphe **(3.1.2)**.

$$\mathbf{P} = \{(F_i^p, \mathbf{W}(F_i^p))\}$$

où **i** = a(**a**uteur), k(**k**eywords), l(**l**ocalisation), n(**n**ewsgroups), p(**p**riorité),s(source)..etc.

Les deux opérations possibles dans ce corps sont le croisement (*crossover*) et la mutation qui scrutent la population pour entretenir de nouvelles générations en introduisant ou en excluant de nouveaux membres **[Goldberg ; 94]**.

4.1 Le crossingover (*crossover*)

La condition nécessaire pour pouvoir appliquer cette fonction est d'assurer que les champs dans chaque profil sont ordonnés de la même façon. L'ordre exact n'est pas important du moment qu'il soit le même pour tous les profils. Le champs *newsgroup* est généralement le premier, suivi du champ mot-clés et ainsi de suite.

Soient $\mathbf{P_1}$ et $\mathbf{P_2}$ deux profils parents :

$$\mathbf{P_1} = \{(f_i^p, \mathbf{W}(f_i^p))\}$$

$$\mathbf{P_2} = \{(F_i^p, \mathbf{W}(F_i^p))\}$$

Le croisement de ces deux profils donnent naissance à deux profils fils. Chacun des profils fils possède des caractéristiques provenant de l'un ou l'autre des parents. Le croisement à deux points (*two-point crossover*) est utilisé par l'opérateur de croisement. Deux points sont choisis au hasard dans la liste des champs. Tous les champs désignés par ces deux points sont échangés entre les deux parents pour créer deux nouveaux fils. Si les champs dans les profils des parents sont dans le même ordre,

chaque profil fils aura un champ de chaque type. La ***finesse*** des profils fils est remise à la valeur initiale par défaut. On aura :

$$\mathbf{P_1 \otimes P_2 \Rightarrow P_3, P_4} \qquad \mathbf{(3.10)}$$

$$\mathbf{P_3} = \{ (g_i^p, \mathbf{W}(g_i^p)) \}$$

$$\mathbf{P_4} = \{ (G_i^p, \mathbf{W}(G_i^p)) \} \qquad \mathbf{(3.11)}$$

L'équation **(3.10)** définit l'opération de croisement (*crossover*), quand les profils parents $\mathbf{P_1}$ et $\mathbf{P_2}$ produisent les deux fils $\mathbf{P_3}$ et $\mathbf{P_4}$, les deux points de croisement sont générés comme indiqués dans l'équation **(3.12)**, ou le nombre de degré de liberté est un entier défini par deux paramètres inclusifs $\mathbf{k_1}$ et $\mathbf{k_2}$ (l'indexe du premier champs est 0). Si les profiles fils sont définis comme indiquée dans l'équation **(3.11)**, chaque champ est défini comme indiqué dans les équations **(3.13)** et **(3.14)**. La ***finesse*** des profils fils est affectée à une valeur par défaut comme indiqué dans l'équation **(3.15)**.

$$\mathbf{k_1 = random(1,max(fields)-2)}^{17}$$
$$\mathbf{k_2 = random(k_1+1,max(fields)-1)} \qquad \mathbf{(3.12)}$$

$$g_i^p \begin{cases} f_i^p & \mathbf{0 \le i < k_1 \text{ et } k_2 < i \le max(fields)\text{-}1} \\ F_i^p & \mathbf{k_1 \le i < k_2} \end{cases} \qquad \mathbf{(3.13)}$$

[17] Prend une valeur entre 1 et le nombre maximum de champs diminué de 2.

$$G_i \begin{cases} f_i^p & 0 \le i < k_1 \text{ et } k_2 < i \le \max(\text{fields})-1 \\ F_i^p & k_1 \le i \le k_2 \end{cases} \quad (3.14)$$

$$f(P_3) = 0.5$$
$$f(P_4) = 0.5 \quad (3.15)$$

4.2 La mutation

L'opération de mutation est définie pour contribuer à la découverte du comportement de la population de profils. Un profil fils provenant d'une mutation orientera la recherche dans un nouveau domaine non exploré par les parents ou les autres membres de la population. Ceci aide la population de profils à s'adapter à d'éventuels changements dans le comportement de l'usager.

Les profils fils sont légèrement différents des parents, tout en gardant un certain nombre d'attributs afin d'exploiter leurs connaissances. Ceci est obtenu en modifiant le champ *newsgroup* du profil. Un *newsgroup* est choisi au hasard et remplacera le *newsgroup* initial.

La fonction de *Similarité* est définie en dehors du newsgroup. Cette *Similarité* est utilisée pour calculer la similarité entre chaque paire de *newsgroups*. Quand le *newsgroup* original doit être remplacer, le *newsgroup* le plus proche sera le premier candidat pour le remplacer. Un seuil « **n** » peut être défini pour limiter le nombre de candidats à considérer. Les profils mutants doivent être différents des parents, aucun des candidats ne doit êtres présent parmi les parents. Une fois les **n** voisins les proches choisis, un *newsgroup* est choisi de façon aléatoire et sera utilisé dans les profils fils.

Pour calculer la *Similarité* entre deux *newsgroupes*, ils sont d'abord mis dans leurs représentations vectorielles. La fréquence des *newsgroupes* est utilisée comme poids des termes. La fréquence des documents pour un terme **t** dans le newsgroupe **M** est la fraction du nombre total de documents dans laquelle **M** contient le terme **t**. Elle est calculée comme indiqué dans l'équation **(3.16)**. Ceci donne un vecteur de termes pondérés par la fréquence des documents. Le même calcul est fait pour un autre *newsgroupe* **N**. La similarité entre les deux newsgroupes et donnée dans l'équation **(3.17)**.

$$df_i^M = n_i^M / N^M \qquad \textbf{(3.16)}$$

$$\mathbf{S(\,M \backslash N\,)} = \sum_i df_i^M \times df_i^N \qquad \textbf{(3.17)}$$

où df_i^M est la fréquence des documents contenant le terme **t** dans le *newsgroupe* **M**, n_i^M est le nombre de documents dans **M** contenant le terme **t** et N^M est le nombre total de documents dans **M**.

Comme dans le cas du croisement, la *finesse* des fils prend par défaut la valeur 0.5.

4.3 La nouvelle génération

Comme nous venons de le voir précédemment, une population est un ensemble de {(profil, *finesse*)}, on suppose que les profils sont ordonnés en d'ordre décroissant de leur *finesse*. Soit :

$$\mathbf{G = \{\,(p_i,\ f(p_i)\,\}}$$
$$\mathbf{i < j \ \Rightarrow \ f(p_i) > f(p_j)}$$

Quand une nouvelle génération est crée, les meilleurs éléments de la population seront retenus alors que les autres seront éliminés. Soit r le taux de rétention (*retention rate*), qui exprime la proportion des membres de la population retenus dans la nouvelle génération.la partie vacante sera proportionnelle à (1-r). Soit c le taux de croisement (*crossover rate*) qui représente la proportion de la partie restante concernée par le croisement, le restant de la population subira l'opération de mutation. La nouvelle génération pourra être décrite comme suit :

$$G_{new} = \{ (P_i, f(P_i) \}$$

où

$$p_i = \begin{cases} p_i & 0 < i < rN_G \\ p_j \otimes p_k & rN_G \leq i \leq rN_G + (1-r)cN_G \\ p_j' & rN_G + (1-r)cN_G < i \leq N_G \end{cases}$$

$$f(P_i) = 0.5$$

où N_G est la taille de la population **G**, j et k sont des nombres aléatoires ente 0 et rN_G. La probabilité pour que j (ou k) prennent une valeur x est proportionnelle à $f(P_x)$ c'est à dire la ***finesse*** de x.

Comme les membres rajoutés par les opération de mutation ou de croisement ont pour but de trouver de nouvelles informations, l'efficacité du système sera proportionnelle

au nombre de nouveaux membres rajoutés à chaque nouvelle génération (1-***r***). La stabilité du système sera proportionnelle à ***r***, qui indique la proportion de la population qui stable à travers les générations. Cette valeur peut être modifiée pour mieux suivre le comportement de l'usager.

CHAPITRE V

Présentation de SIFT Système de filtrage Online

1. Introduction

L'avancée des technologies de communication a induit un large volume d'information digitale dans le réseau. Une série d'outils ont émergé pour la recherche et la diffusion de l'information. **WAIS** (**W**ide-**A**rea **I**nformation **S**ervers), **Archie**, **WWW** (**W**orld-**W**ide **W**ord) et **Gopher [LARDY; 96]** sont les outils les plus connus dans l'Inernet. L'inconvénient de ces outils est que tous permettent d'accéder à des informations déjà existantes mais souffrent de l'absence de mécanisme permettant de tenir l'usager au courant des nouvelles informations. L'explosion de l'utilisation de l'Internet ces dernières années rend cette tâche de plus en plus difficile pour l'usager ne disposant que d'outils de recherche ordinaires. Pour faire face à cette révolution dans le domaine de l'information, une nouvelle vision dans l'accès à l'information commence à concurrencer les modèles de recherche classiques : au lieu que l'usager soit obligé de suivre cette évolution, il serait plus intéressant de lui présenter les nouvelles informations après sélection de celles qui peuvent l'intéresser. L'usager exprime ses intérêts par un ensemble de termes mis à jour quotidiennement, qui expriment son profil. De tels services deviennent très importants et constituent un outil indispensable dans l'univers très dynamique qu'est l'Internet.

Il existe déjà un genre simple de service de diffusion de l'information dans l'Internet. Les listes de diffusion **[KROL, 92]**. Il en existe des milliers couvrant des domaines très larges. L'usager s'inscrit aux listes de son choix et reçoit des messages dans son domaine *via* courrier électronique. Il peut aussi envoyer des messages à la liste pour atteindre d'autres usagers inscrits. **LISTSERV** est le système qui assure la gestion de ce service. Le problème avec ce genre de diffusion et qu'il répond grossièrement aux attentes de l'usager dont les besoins informationnels peuvent varier dans le temps, d'autre part, l'usager peut être inscrit à une liste qui ne couvre pas exactement son domaine et il sera très vite envahit par des messages non pertinents.. **USENET news**

(ou **Netnews**) est un autre service similaire aux listes de diffusion cependant, il a un plus grand succès sur l'Internet. Il est consulté par des milliers d'usagers avec des Mégabytes de trafic quotidien. L'information est y organisée en *newsgroupes* ; l'usager s'inscrit aux groupes qui l'intéressent mais ceci n'empêche pas la surabondance d'information non pertinente. De plus il arrive souvent que l'usager passe à côté d'articles pertinents qui se trouvaient dans des *newsgroupes* aux quels il n'était pas inscrit.

Les recherches actuelles dans le domaine du filtrage d'information converge vers un filtrage avec plus d'efficacité afin d'affmer les informations présentées à l'usager. Les techniques de recherches d'information classique (*information retrevial*) sont associées à des approches utilisées en intelligence artificielle.

SIFT (**S**tanford **I**nformation **F**iltering **T**ool) est l'un des produits de ces recherches, il assure une large diffusion de l'information. Il peut être utilisé comme service de défrichement qui rassemble un grand nombre d'information et les diffusent sélectivement à une large population. Ce service traite les documents en texte intégral utilisant les modèles et les techniques de recherches classiques, combinés à de nouvelles techniques d'indexation capables d'indexer un grand nombre de documents et de profils. Il fonctionne sous **Unix**, et est accessible gratuitement au public par **FTP** *anonymous* à l'adresse :

***f*tp ://db.stanford.edu/pub/sift/sift~1.0.tar.Z**

Dans ce chapitre nous allons décrire **SIFT**. En premier lieu, nous présenterons l'approche utilisée pour la modélisation des intérêts de l'usager et la communication usager-serveur. Puis nous utiliserons les résultats d'une étude pour mettre en évidence sa capacité de traitement, en comparaison à un système de recherche classique.

2. SIFT

Nous commencerons notre présentation de **SIFT** par l'étude d'un exemple simple. Imaginons un usager en quête d'articles sur la « *fission nucléaire* ». (**cf. fig.1**). Il adresse un message d'inscription au serveur de **SIFT** en spécifiant le profil (*fission nucléaire* par exemple) et d'autres paramètres optionnels qui le contrôlent tel que la durée de souscription, la fréquence d'envoi, le nombre de lignes... L'usager peut accéder au service par **WWW** en utilisant l'interface graphique (**cf.fig.2**). Il remplit une fiche d'inscription qui permettra au système de l'identifier et de définir son profil. Avant de s'inscrire définitivement, l'usager doit tester son profil à travers une collection de documents qu'il a préalablement sélectionnée. Cette inscription est stockée dans une base de données qui gère tous les profils. Dès que le service reçoit un nouveau document, il le compare aux différents profils et le distribue à ceux ayant la plus grande *Similarité* (**cf . chapitre III**).

2.1 Modélisation des intérêts de l'usager

L'usager s'inscrit au serveur **SIFT** avec une ou plusieurs inscriptions chacune dans un domaine d'intérêt. Une inscription inclue un profil de même type qu'en **IR** (**I**nformation **R**etrevial), plus d'autres paramètres pour contrôler la fréquence de mise à jour, la quantité d'information à présenter et la durée de l'inscription. L'inscription est identifiée par l'adresse électronique plus un mot de passe attribué lors de la première connexion.

2.1.1 Modèle de filtrage

Le profil peut être dans l'un des deux modèles de recherche : booléen ou vectoriel. **[SALTON, 89]**. Nous allons nous intéressé en premier lieu au modèle vectoriel.

Dans ce modèle, les requêtes et les documents sont identifiés par des termes $\mathbf{t_i}$. S'il existe **m** termes pour d'écrire un document **D**, ce document sera représenté par un vecteur de dimension **m**. $\mathbf{D} = (\omega_1,\omega_2,\omega_3,....\omega_i,...\omega_m)$ où le poids ω_i du terme $\mathbf{t_i}$ exprime son importance statistique dans le document. On peut aussi écrire $\mathbf{D} = \{(\mathbf{t_1},\omega_1)\ ;(\mathbf{t_2},\omega_2)\ ;.....(\mathbf{t_m},\omega_m)\}$ où $\omega_i \neq 0$. Dans notre exemple {(fission,50) ; (nucléaire,50)}[18]. La requête a la même représentation que le document. Pour une paire document-requête. La mesure de la *Similarité*[19] permet de déterminer les documents à présenter à l'usager.

Dans le cas du filtrage, la première préoccupation du système est le nombre de documents à envoyer à l'usager. Une solution est de laisser à l'usager le choix du nombre de document à lui présenter par période **[FOLTZ, 92]**. Cette approche présente deux inconvénients :

- Si pendant une période le nombre de documents pertinents dépasse le seuil fixé par l'usager, il peut passer à côté de certains documents même pertinents (*low recall*) .
- Si, au contraire, pendant une période il n'y a pas assez d'information pertinente, le filtre n'accomplira pas sa tâche car il présentera quand même des documents (*low precision*).

Pour remédier à ces problèmes, l'usager peut fixer un seuil (*relevance threshold*) qui représente le minimum de *Similarité* que le document doit avoir pour être sélectionné[20].

L'usager a la possibilité d'utiliser en plus le modèle booléen pour spécifier les termes qu'il veut inclure ou exclure des documents[21].

---

[18] Le système attribue par défaut la valeur de 50 au score des termes dans le corps du document.
[19] La similarité peut être mesurer en calculant la distance métrique entre les représentations respectives document-requête. En IR la grandeur la plus utilisée est le cosinus de l'angle entre les deux vecteurs.
[20] Dans le système actuel, une valeur est attribuée par défaut et en cas d'anomalie, l'usager peut la modifier pour affiner la recherche.
[21] Le modèle booléen de **SIFT** ne permet que la conjonction ou la négation des termes.

2.1.2 Construction et modélisation du profile

Pour assister l'usager dans la construction de son profil, le serveur **SIFT** permet de faire un test . L'usager peut tester son profil initial à travers une représentation de documents préalablement sélectionnés. L'usager a la possibilité de changer et d'ajuster le seuil à la précision désirée. Quand il satisfait des performances du filtre il peut envoyer son inscription définitive. Il recevra par la suite, à intervalle de temps réguliers les articles sélectionnées . Il a la possibilité de modifier a tout moment son profile ou n'importe quel paramètre. Il est également possibilité d'utiliser la rétroaction telle qu'elle a été décrite en **IR [G.SALTON ; 89]**. L'usager peut indiquer au système un ensemble de documents pertinents qui lui permettront de construire le profil le mieux adapté a ses besoins.

2.2 Protocole de communication

Il existe deux modes de communication entre l'usager et le serveur de **SIFT**. Dans le premier : le mode interactif, l'usager s'inscrit, teste son profil, effectue les mises à jour, visualise ses documents...Dans le second, le mode passif, l'usager reçoit périodiquement les nouvelles informations. Au lieu de développer un nouveau protocole de communication, **SIFT** utilise les possibilités qu'offre le courrier électronique **[CROCKER, 82]** et World-Wide Web http **[BERNERS ; 92]**.

En premier lieu, nous allons traiter le mode de communication interactif. La communication par *e-mail* est le plus simple des moyens de connexion en réseau. Grâce à son interface e-*mail*, **SIFT** est accessible par les usagers ne disposant pas de machines très puissantes, tout en évitant les risques de saturation du réseau.

Avec le développement de la navigation hypermédia, et d'interfaces sophistiquées en **WWW**, un accès **WWW** a été développé avec une interface plus conviviale où l'usager peut interagir directement avec le serveur.

Dans le mode passif, le serveur envoie périodiquement des messages contenant des extraits[22] d'articles. Après lecture de ces extraits, l'usager peut accéder au serveur pour récupérer l'intégralité de ceux qui l'intéressent. Souvent cet extrait est écrit dans un format reconnaissable par la plupart des logiciels de lecture du courrier électronique. Les travaux actuels essayent d'utiliser un format html afin que l'usager puisse récupérer directement ses articles , et émettre sa rétroaction *via* http.

2.3 Filtrage des news (SIFTing Netnews)

SIFT peut être utilisé comme un service de diffusion sélective des *news*. Comme tout service sur l'Internet, l'usager peut accéder à **Netnews SIFT** *via e*-mail ou l'interface **WWW**. Vue les problèmes de saturation du réseau, il est beaucoup plus facile d'accéder par voix courrier électronique à l'adresse : **netnews@db.stanford.edu** avec le terme « **help** » dans le corps du message. L'accès **WWW** à l'URL **http ://sift.stanford.edu**. demeure très difficile du fait du grand nombre d'adhérents. Le service a vu le jour en février 1994, il a été annoncé dans deux *newsgroups*, plus de 13 000 profils ont enregistrés pendant les dix premiers mois (Novembre 1994). Le tableau 1 montre les résultats d'une étude effectué le 10 Novembre 1994[23].

Tableau .1

nombre d'inscriptions	13 381
nombre d'usagers	5 146
nombre moyen d'articles distribués	45 127

le nombre moyen d'articles correspond à la période du 02/94 au 10/94.

[22] En général l'en-tête plus les dix premières lignes de l'article.

[23] Nous n'avons pas pu trouver d'autres études plus récentes, mais on estime que ce nombre a subit la même croissance exponentielle qu'a subit le nombre d'usager de l'Internet.

La première chose qui ressort de cette étude est que le système est très sollicité, en effet il est nécessaire de traiter plus de 45 000 articles et de les répartir sur environ 13 000 profils. Un tel système s'est, très vite trouvé saturé ce qui a conduit à son implantation sur des « *sites miroirs* » en Europe et en Asie[24].

Il est à noter que le **Netnews SIFT** serveur n'a pas pour objectif de remplacer un jour les services classiques de *news* qui ont encore des beaux jours devant eux, en effet ces derniers permettent la création et la distribution de discussion et en aucun cas, la fonction de filtrage ne pourrait les substituer. Les deux services auront plus de chance de collaborer que de se concurrencer.

Le filtrage des *news* n'est entre autre, que l'une des applications de **SIFT** ; il assure également la diffusion sélective d'articles techniques en informatique. (*e-mail* : **elib@db.stanford.edu** ; la version **WWW** est en cours de réalisation). Ce service lucratif, est destiné aux professionnels et compte aujourd'hui environ 10 000 clients.

4. Fonctionnement

SIFT fonctionne sous **C**. Il a été compilé dans plusieurs versions **Unix** : **DEC Ultrix 4.2**, **HPUX 8.07**, **SunOS 4.1** ....etc. Dans un premier temps nous nous contenterons de faire une description globale des composantes du système, puis nous focaliserons notre travail sur son fonctionnement.

[24] Malgré l'installation de ces nouveaux serveurs, la connecter *via* WWW est toujours difficile.

4.1 Les composantes du système

La **figure.7** montre les différents étages du système.

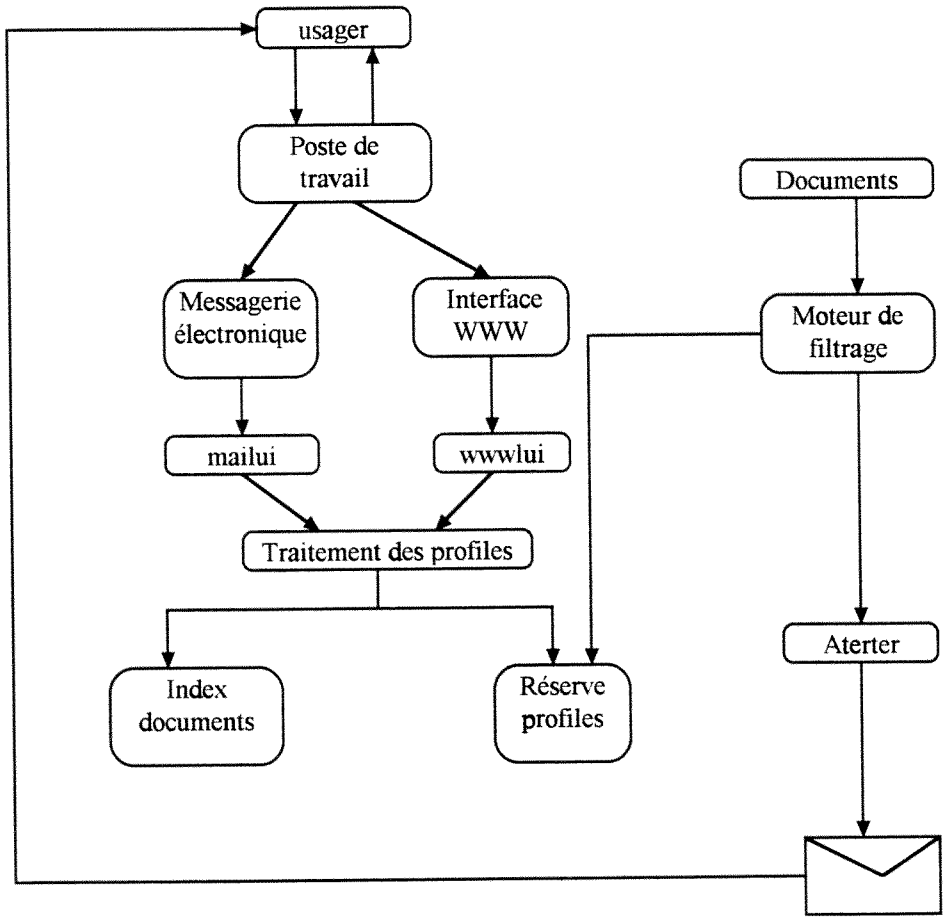


Fig.7 Composantes de SIFT

Le programme « ***mailui*** » est destiné au traitement des requêtes. Il analyse (grammaticalement) les messages électroniques supposés dans un format standard **[CROCKER ; 82]**. Le traitement des requêtes se fait en fonction du contenu de la base de souscription[25] (*suscription database*) et l'index des documents.

[25] subscription database : renferme les différents profiles.

Le programme « ***wwwui*** », appelé aussi « ***script-cgi***» permet une utilisation **HTTP**. Il a été développé par **NCSA** (**N**ational **C**enter for **S**upercomputing **A**pplication). Il permet l'accès à travers une interface **WWW** et a un fonctionnement identique au précédent

Le programme « ***Filter*** » est le cœur du moteur de filtrage (*filter engine*). Il reçoit une collection de documents qu'il traite en fonction des profils. Le résultat de ces opérations est un ensemble de couple profil-document. Ces couples seront ensuite triés par l'usager ce qui permettra au système d'ajuster les paramètres de souscription.

Le programme « ***alerter*** » se chargera de l'envoi des extraits[26] des documents sélectionnés en utilisant **Unix** *sendmail.*

4.2 Fonctionnement du moteur de filtrage

Dans le cas d'un système de recherche d'information classique, un index des documents est créé pour le traitement des requêtes **[SALTON ; 89]**, en associant à tous les termes leurs poids représentant leurs occurrences dans le document. Cette approche a été utilisée dans les premiers prototypes de **SIFT** mais a été très vite abandonnée du fait du grand volume de documents et de profils à traiter **[Tak W.Yan ; 95]**. Dans les versions plus récentes, on a adopté une autre approche. L'idée est de considérer le filtrage d'information comme un double problème de recherche d'information où l'on traiterait les profils comme des documents et les documents comme des requêtes. Au lieu de construire un index de documents, on construit un index des profils. Dans **[Yan, H.Garcia ; 37b,37c]** on trouvera une description des modèles vectoriel et booléen pour la modélisation des profils. Dans le cas de **SIFT**, une combinaison des deux techniques a été adoptée. Nous essayerons,

[26] En général l'en-tête de l'article plus les dix dernières lignes.

dans ce qui suit, de décrire brièvement, à travers un exemple simple, le principe de cette approche, le lecteur peut se référer à **[Yan, H .Garcia ; 37b,37c]** pour plus de détails.

Soit le profil **P1** (**cf. fig.8**) décrit par le vecteur **P1**= {(cold,50),(fision,650)}et P2 est un profil booléen « transition **and** Plutonium .**not** fision ». A chaque terme on associe les profils qui le contiennent. Par exemple au le terme « fision », on affecte les deux profile **P1** et **P2**. Ce terme est pondéré dans **P1** alors qu'on lui affecte la valeur -1 dans **P2**.

Fig.8 Index des profiles .

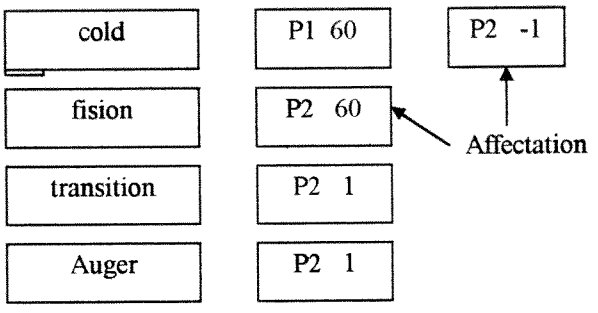


Tableau.2

Profiles	Seuil	Score
P1	8 400	9 600
P2	2	-1

5. Performances du système

Le principal objectif de **SIFT** est de traiter le maximum de documents dans un temps optimal inférieur à 24h, en effet si **SIFT** s'est spécialisé dans le filtrage des articles de *news*, le facteur temps prend une importance particulière si on veut maintenir l'usager à jour dans son domaine d'intérêts.

5.1 Influence du volume d'information

La première étude pour l'évaluation des performances de **SIFT** a été menée en 1994 **[Tak. Yan ; 95]**. L'expérience a été réalisée avec une collection de 38 000 articles de *news*, le nombre moyen de termes[27] par articles est de 269. 7 000 profils ont été choisis au hasard dans l'index profils. Nous avons pris quelques résultats de cette étude que nous essayerons d'interpréter.

L'étude a montré que le temps de filtrage est divisé en quatre partie :

- **t1** (*build time*) : le temps nécessaire pour la construction de l'index des profils.
- **t2** (*Filtering time*) : Les profils sont comparés aux documents puis les paires profils-document sont classées.
- **t3** (*Sorting time*) : Les paires documents-profils dont les scores ont dépassé un certain seuil sont affectées à l'adresse électronique correspondant au profil.
- **t4** (*Notify time*) : Le temps nécessaire à l'envoi des articles. **Unix sendmail** se charge de cette tâche.

La répartition des durées de chaque phase est indiqué sur la **figure.9**.

[27] Terme = plus de 2 caractères.

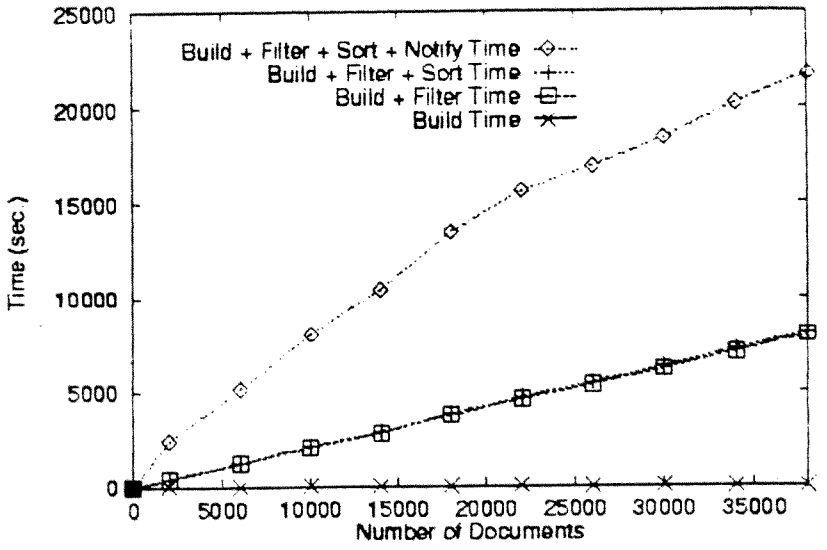


Fig.9 Evolution du temps de filtrage en fonction du nombre de documents

l'étude montre que le temps **t1** est négligeable. Le temps de filtrage **t2** est proportionnel au nombre de documents, soit environ 0.21 sec/document, t3 est négligeable. Le temps total nécessaire au traitement est de 0.63 sec/ document. Ceci en supposant que les documents soient dans un format reconnu par le système dans le contraire, il faudra compter environ 42.8 % de temps en plus.

5.2 Influence du nombre de souscription (profils)

Pour le même nombre d'articles, l'étude a été faite avec un nombre variable de profils (1, 1000, 4 000, 7 000, 14 000). Les résultats sont représentés dans la **figure.10**.

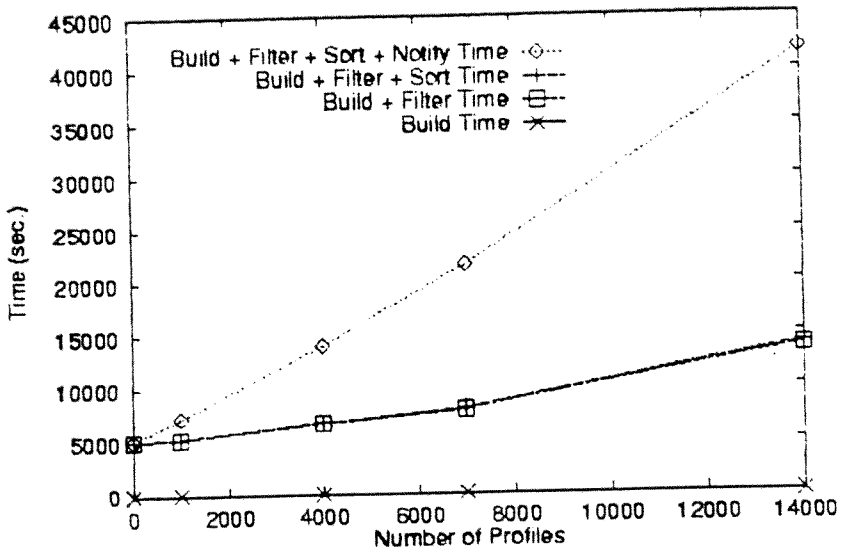


Fig.10 Evolution du temps de filtrage en fonction du nombre de profiles

Le temps **t1** demeure négligeable. Le temps de filtrage reste proportionnel au nombre de profil avec un coefficient de proportionnalité de 0.62 sec/profil. La durée totale de l'opération est de 2.63 sec/profil.

5.3 Capacité de mémoire

La capacité totale de mémoire nécessaire au moteur de filtrage peut être calculée comme suit :

$$\textbf{Capacité (octets)} = (\, n_p \otimes 132\,) + (n_t \otimes 32\,) + (\, f_i \otimes 12\,)$$

où n_p : nombre de profils

n_t : nombre de termes distincts par profil

f_i : occurrence des termes dans le profil

Le premier terme (132 octets par profil) permet d'enregistrer le seuil, le score en plus de l'adresse électronique, le second (12 octets par profil) permet d'identifier tous les termes présents dans l'index, le troisième terme permet d'enregistrer l'occurrence des termes présents dans le profil. Une étude a été effectuée pour déterminer la capacité sollicitée par le filtre pour différents nombres de souscriptions **[Talk ; 95]**. Le **tableau.3** montre les résultats de cette étude.

Tab.3

np	nt	fi	Mémoire (bytes)
1 000	1 595	2 334	211 048
2 000	2 806	4 756	410 864
3 000	3 746	6 893	598 588
4 000	4 657	9 244	787 952
5 000	5 535	11 735	977 940
6 000	6 278	14 102	1 162 120
7 000	7 001	16 489	1 345 900

Ces résultats montrent que le nombre moyen d'occurrence est de 2.34 termes/profil, d'autre part le nombre de termes distincts est très grand pour un petit nombre de profils, puis décroît rapidement pour se stabiliser à environ 1 terme/profil. On supposant que le nombre de termes distincts est proche du nombre de profils pour $\mathbf{n_p}$ très grand (comportement asymptotique) on peut démontrer (voir exemple de calcul) qu'il faut environ 195 octets /profil ce qui donnera une capacité de 5 128 profils par Mégaoctets. Avec les progrès faits en électronique , **SIFT** pourra sans problème traiter des centaines de milliers de profils.

Si $\mathbf{nf \geq 1000}$ alors $\mathbf{nf \approx nt}$

Exemple de calcul :

pour $\mathbf{n_p \approx n_t \approx 1000}$ avec $\mathbf{f_i \approx 2.34}$ on aura

capacité = (1000*132 + 1000* 32 + 2.34*000*12) / 1000 = 192 octets

5.4 Interprétation des résultats

Cette étude nous permet de faire un premier Costat : plus de 60 % du temps nécessaire au filtrage est dépensé par l'envoi des résultats du filtrage à l'usager et non pas par le filtrage proprement dit, d'autre part, même avec un nombre très important de profils ou de documents, le système fonctionne dans des conditions optimales raisonnables.

L'une des solutions envisagée dans les travaux actuels pour réduire le temps d'envoi des messages est de créer une diffusion par collaboration. En effet, on définit un regroupement géographique des usagers ; une seule copie du document est envoyée au groupe et un serveur local se chargera de sa diffusion après lecture par l'un des membres du groupe qui décidera de sa pertinence ou non. Ceci ne peut être intéressant et se faire que s'il existe un nombre suffisant d'usagers ayant les mêmes centres d'intérêts ce qui est le cas dans une équipe de recherche par exemple.

CHAPITRE VI

Présentation d'un filtre local : INFOSCAN

1. **Présentation Infoscan**
2. **Capacité**
3. **Création d'un filtre**
4. **Création d'une collection**
5. **Evaluation et visualisation des résultats**
6. **Vérification des performances du filtre**

1. Présentation INFOSCAN

Infoscan est un outil de filtrage et d'évaluation de l'information disponible dans le réseau. Il vise à rentabiliser la consultation des sources d'information connues de l'utilisateur : courrier électronique, *news*, bases de données locales ou éloignées...etc. Le texte complet des articles est évalué en fonction des mot-clés choisis par l'utilisateur. Les articles sont ensuite présentés de façon graphique intuitive, ce qui permet de repérer les articles les plus pertinents en un coup d'œil, et évite des explorations fastidieuse et des lectures inutiles.

Les principales fonctions d'**Infoscan** sont les filtres, La présentation visuelle des documents trouvés et l'extraction de mots-clés. Les filtres sont des descriptions des sujets d'intérêt de l'utilisateur. A partir des filtres, **Infoscan** évalue les documents et présente les meilleurs d'entre eux de façon visuelle et très intuitive . Un outil linguistique permet d'extraire les mots-clés des documents pour les ajouter aux filtres. Cet outil est disponible en français, en anglais et en espagnol.

L'interface d'**Infoscan** se divise en trois parties : la zone du profile de l'utilisateur permet de définir et de sélectionner les filtres. L'écran du radar sert à lancer les recherches et à visualiser les documents trouvés. Enfin, la zone de retour de l'information contient le titre du document courant, ainsi qu'un aperçu de son contenu. Une fenêtre permet de lire, d'enregistrer et d'imprimer les documents.

Bien que la recherche soit rapide sur un nombre raisonnable de documents, **Infoscan** offre un mécanisme automatique d'évaluation des documents. L'utilisateur peut faire travailler **Infoscan** pendant la nuit, de façon à ce que les résultats soient prêts à être

afficher le lendemain. Ce processus permet de consulter une grande quantité de documents[28] de manière efficace et agréable.

Le profil de l'utilisateur se compose de trois éléments de base. Les *filtres* contiennent les mots-clés qui décrivent les sujets intéressant l'utilisateur. Les *collections* sont des ensembles de sources d'information, chaque source étant un répertoire de documents. Les collections permettent de regrouper les sources traitant de sujets semblables. Les *requêtes* sont des couples de *collection-filtre* . L'évaluation d'une requête permet d'afficher les documents de la collection qui correspondent à son filtre.

Un filtre est composé de cinq sous-filtres. Chaque sous-filtre contient une liste de mots-clés, accompagnée d'une pondération et d'une portée. La pondération est la valeur qui sera ajoutée au score du document qui contient un mot de la liste ; elle permet de marquer les aspects prioritaires d'un sujet. La portée permet de chercher les mots-clées dans le document entier ou dans des en-têtes particulières. Les types de recherche disponibles pour chaque mot-clé sont : « mot entier », « respect des cases »(min./maj.) et « avec variation ». Cette dernière option permet de trouver les mots malgré les erreurs de frappe et d'orthographe.

Il existe plusieurs façons de s'assurer que les mots-clés choisis repèrent des documents intéressants. Tout d'abord, les mots-clés du filtre sont mis en évidence dans la fenêtre de lecture des documents. La liste des documents rejetés permet de vérifier que le filtre n'a pas oublier des documents intéressants (si c'était le cas, il suffirait d'ajouter les mots-clés de ce document dans le filtre pour l'améliorer).

2. Capacité

La demande de mémoire vive de la version commerciale est directement liée au nombre de document à traiter. Elle a besoin 1,5 Mo plus 2,2 Ko par document. Ceci

veut dire que pour traiter 1 000 documents dans une recherche, **Infoscan** aura besoin de 3,7 Mo de mémoire vive.

3. Création d'une collection

Une collection est un ensemble de sources d'information. Chaque source est un répertoire qui contient des documents en format texte (**ASCII**), ce répertoire doit être sur une machine ou sur le réseau local. Chaque collection peut contenir un nombre illimité de sources, ce qui permet de regrouper les sources qui traitent de sujets semblables. Trois paramètres accompagnent chaque source :

- le format du document
- le choix des documents à évaluer
- la représentation des documents de cette source.

Le format d'une source permet définit la structure que devrait avoir les documents qu'elle contient. Les documents issus du courrier électronique et des *news*, par exemple, possèdent des en-têtes qui donnent le titre du document, son auteur, sa date et d'autres informations. Pour le moment, **infoscan** ne reconnaît que les en-têtes « *subjets* », « *from* », et « *date* ».

4. Création d'un filtre

Un filtre est la représentation d'un des champs d'intérêts dans **Infoscan**. Chaque filtre est un ensemble de mots-clés. Ces mots-clés sont organisés en cinq listes, ou sous filtres, ce qui permet de regrouper les mots-clés reliés par le sens.

Tous les mots-clés d'un même filtre ont la même pondération et la même portée. La pondération est une valeur subjective qu'on associe au mot-clé. Par défaut, **Infoscan**

---

[28] Pour le moment, **Infoscan** est limité aux documents en format texte ASCII disponible sur un disque dur local. Il peut donc être utilisé pour des *news* ou du courrier électronique déjà téléchargés localement. La prochaine version offrira une meilleure intégration avec les news et le courrier électronique.

assigne des pondérations légèrement décroissantes aux sous filtres allant de (+5) a (+1) pour le cinquième sous-filtre.

La portée d'un mot-clé est la région du document dans laquelle ce mot-clé sera recherché. Par défaut, **Infoscan** cherche les mots-clés dans le document complet, c'est à dire dans le corps du texte mais aussi dans tous les en-têtes.

5. Evaluation et visualisation des résultats

Infoscan affiche tous les documents qui contiennent au moins un des mots-clés demandés.(**Fig. 11bis**). Plus un document est près du centre du radar et plus il correspond à la requête. En cliquant sur l'icône correspondant, on verra sur la zone de retour du document (en bas à gauche) le titre du document, ainsi que la proportion relative des mots-clés de chaque sous-filtres qu'il contient. Si l'aperçu du contenu du document est intéressant, on peut ouvrir le document en doublant le clique.

6. Vérification de la performance du filtre

En observant le rapport entre le nombre de documents affichés et le nombre total de documents (sous le radar). Si presque tous les documents sont sur l'écran, c'est que certains des mots clés sont très fréquents dans la collection. Il faudra donc les éliminer.

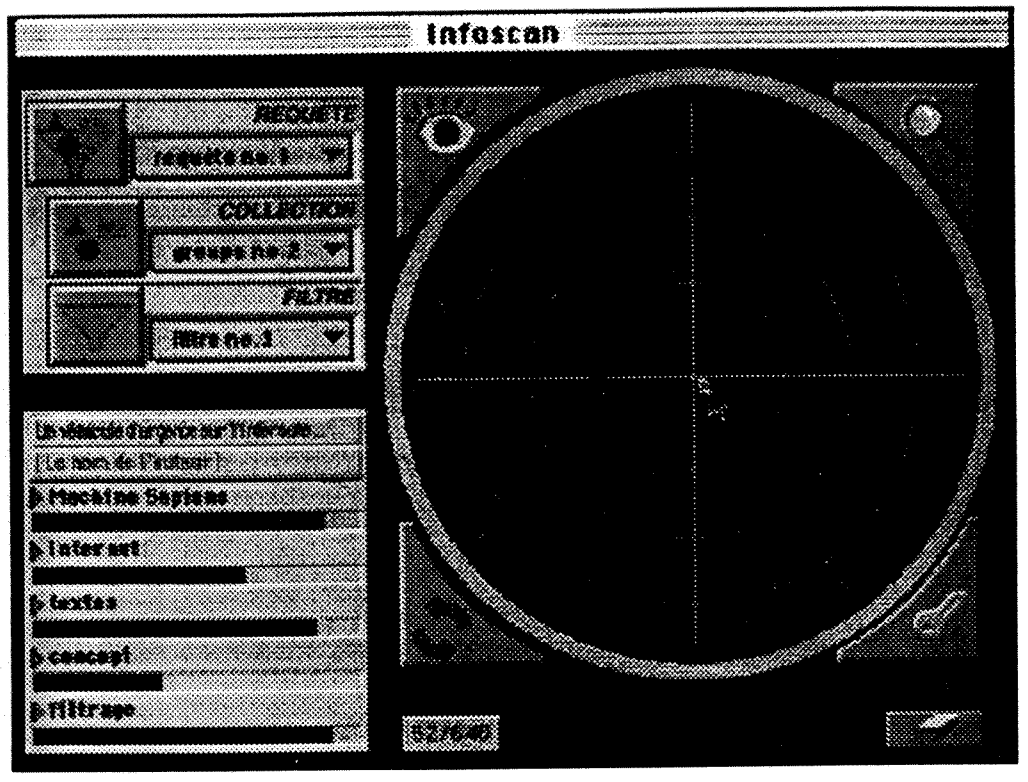


Fig. 11bis Fenêtre INFOSCAN

CHAPITRE VII

PRESENTATION D'UN AGENT : *Newt*

Introduction

1. Interface graphique

- **1.1 La fenêtre**
- **1.2 Lecture des articles trouvés par l'Agent**
- **1.3 rétroaction sur les articles trouver**
- **1.4 Recherche manuelle et programmation par démonstration**
- **1.5 Addition de nouveaux Agents**
- **1.6 Population des profils**
- **1.7 Visualisation des profils**

2. Le module d'Apprentissage

3. Le module de Filtrage

- **3.1 Extraction des documents**
- **3.2 Annotation et sélection des documents**

Introduction

Dans ce chapitre, nous allons présenter un Agent dont le fonctionnement est basé sur l'algorithme décrit dans le **chap III**. Cet Agent, **Newt**, est un prototype qui a été développé à l'Institut de Technologie de **Massachusetts [BEERUD ; 94]**.

L'agent est constitué de trois modules :

- L'interface graphique.
- Le module d'apprentissage.
- Le module de filtrage.

1. Interface graphique

L'interface graphique de **Newt GUI** (Graphical **U**ser **I**nterface) est constituée de plusieurs Agents.

1.1 La fenêtre

La fenêtre est constituée d'un ensemble d'Agents (**Fig.12**). Chaque Agent est représenté par une icône (*the Agent Icôn*). Il est responsable de la recherche d'articles dans un domaine précis. A chaque icône, sont associées trois autres icônes : (+), (-) et (**?**). Les deux premières assurent la rétroaction, quand à la troisième, elle permet à l'usager de connaître, à n'importe quel moment les paramètres qui ont permis à l'Agent de faire son choix.

Chaque Agent est associé à une couleur qui permettra par la suite d'identifier les articles qu'il a trouvé.

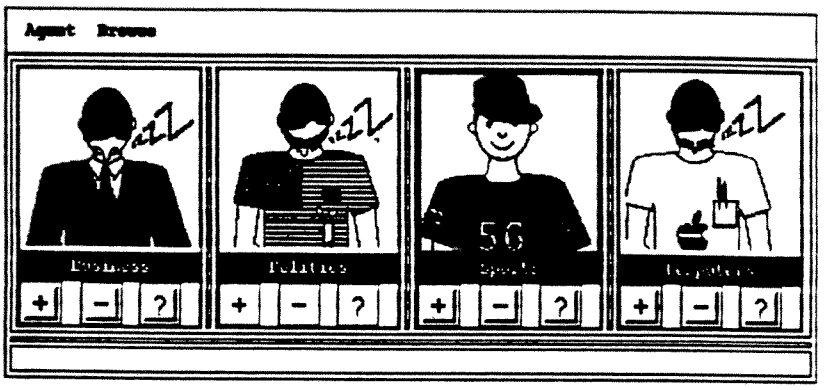
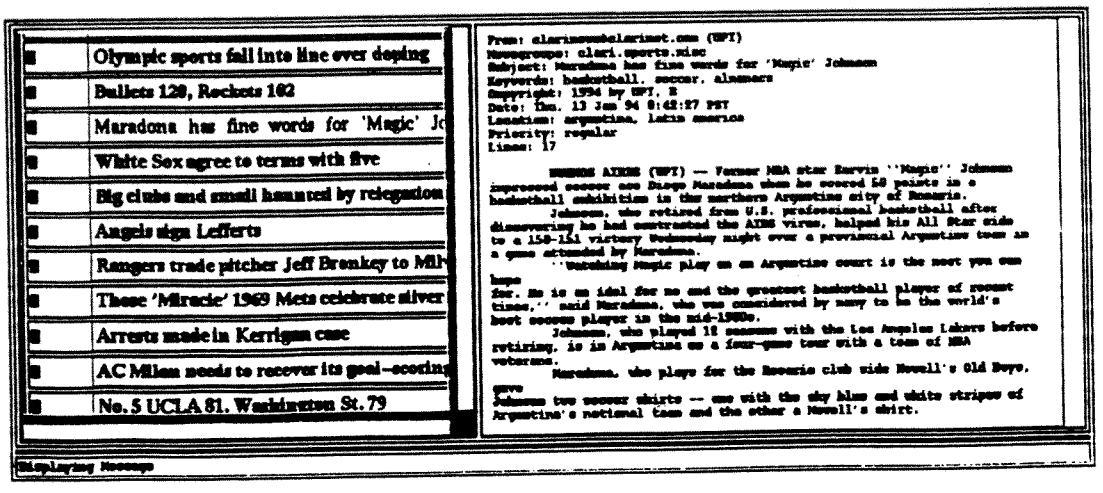


Fig.12 Fenêtre de l'Interface graphique

1.2 Lecture des articles trouvés par l'Agent

En cliquant sur l'Agent désiré, une autre fenêtre s'ouvre (*News Window*) qui permet de visualiser les articles trouvés par cet Agent. Sur la parte gauche de la fenêtre, on trouve les titres des articles sélectionnés classés dans l'ordre décroissant de leur score. Pour lire le contenu d'un article, l'usager clique sur le titre et le texte apparaîtra sur la fenêtre de droite (**Fig.13**). Les articles déjà lus seront en caractères gras.

Fig.13 Fenêtre des articles (*News Window*)



1.3 Rétroaction (feedback)

L 'usager peut émettre une rétroaction positive ou négative à n'importe quel moment en cliquant sur l'icône (+) ou (-).

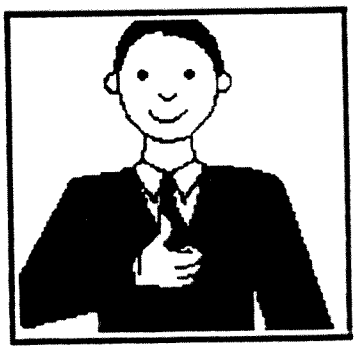


Fig.14 Agent recevant une rétroaction positive

Comme nous l'avant vu dans le **chap III**, la rétroaction modifie toutes les caractéristiques de l'article. Si l'usager ne veut émettre une rétroaction que sur une partie, il peut sélectionner le texte correspondant. Dans ce cas, seul les termes contenus dans cette partie seront ajoutés ou éliminés du profil.

1.4 Recherche manuelle et programmation par démonstration

L'usager a la possibilité de faire une recherche manuelle. Dans la barre des tâches, le menu « *Browse* » permet cette fonction. Le système invite l'usager à choisir un *newsgroupe* particulier, et lui présentera par la suite tous les articles trouvés dans ce *newsgroupe*. Ces articles n'ont pas été forcement trouvés par des Agents déjà existant, par conséquent, ils ne seront ni annotés ni classés. Dans le cas contraire, leurs titres seront dans la couleur de l'Agent qui les a trouvés, ce qui permettra à l'usager d'avoir une idée sur le contenu.

Si un article attire particulièrement l'attention de l'usager, il peut créer un Agent qui retrouvera des articles similaires. Pour initialiser ce nouvel Agent, il a le choix entre saisir les différents paramètres manuellement ou indiquer à l'Agent un ensemble d'articles qui lui permettront de construite un profil(*Programming by demonstration*).

1.6 Population des profils

Un Agent est constitué d'une population de profils qui permettent de définir les intérêts de l'usager. La population des profils peut êtres visualiser en cliquant sur « **SHIFT** ». Ceci ouvre une fenêtre (*Population Window*) (**Fig.14**). Chaque profil est accompagné d'une annotation indiquant sa *finesse* (**cf. chap.III**).

Un certains nombre d'opérateurs en bas de la fenêtre permettent de manipuler les profiles.

- **Add** : permet d'ajouter un nouveau Agent. L'Agent ainsi crée doit être initialiser immédiatement (**cf. 2.5**), car si aucun *newsgroupe* ne lui est indiqué, il ne pourra ni effectuer une recherche, ni recevoir de rétroaction.
- **Kill** : Permet d'éliminer un Agent.
- **Mutate** : une variante « *mutante* » d'un profil peut être rajoutée à la population.
- **Xover** : Permet de croiser deux profils en échangeant leurs champs.
- **NextGen** : Permet de modifier la totalité de la population. La nouvelle génération est générée en retenant certains membres (ceux ayant les plus grande *finesse*) et éliminer les autres en complétant le reste par des profils « *mutants* ».

1.7 Visualisation des profils

On peut visualiser le profil en cliquant sur l'icône correspondant . Ceci ouvre une nouvelle fenêtre (*Profile Window*) (**Fig.15**). On peut visualiser n'importe quel champs du profil.

Trois opérateurs : « **Add** », « **Del** » et « Edit » permettrons de modifier le profile.

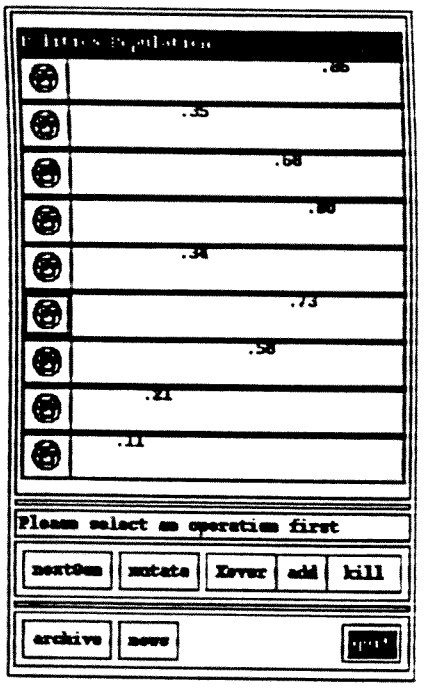


Fig.14 Fenêtre de la population des profils (*Populaion Window*)

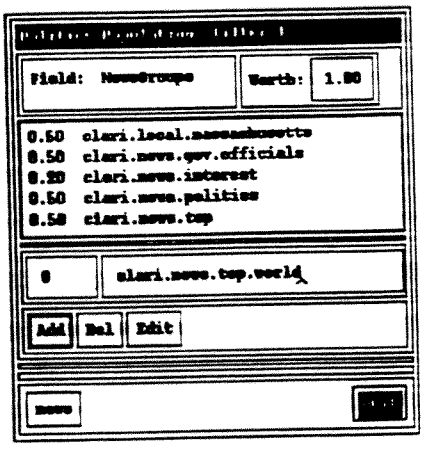


Fig.15 Fenêtre d'un profil (*Profile Window*)

2. Le module d'Apprentissage

Le module d'apprentissage assume la personnalisation des profils aux intérêts de l'usager et assure leur corrélation. L'ensemble des profils doit remplir deux conditions : la première et d'être spécifique aux besoins actuels de l'usager ; la seconde est d'être capable de s'adapter à tout changement dans ces intérêts.

Chaque profil doit respecter une certaine forme (**Fig.16**). Comme nous l'avons vu dans le **chapitre III**, l'apprentissage se fait par la combinaison de deux fonctions : la rétroaction et l'Algorithme génétique.

La rétroaction de l'usager a deux effets, le premier est au niveau de la population en modifiant la valeur de la ***finesse***. Le second, dans le profil même, dont les éléments peuvent être modifier soit directement soit en programmant par démonstration.

L'Algorithme génétique assure le renouvellement des générations de profils.(**cf. chap.III**).

3. Le module de filtrage

Le module de filtrage appelé **YAIF** (**Y**et **A**nother **I**nformation **F**ilter) a pour fonction de comparer les profils aux documents et de ne présenter à l'usager que ceux qui ont la plus grande chance de l'intéresser.

Chaque profil est traité à part. Deux types de newsgroupes sont recherchés pour chaque profil. Le premier correspond à ceux mentionnés dans son champs *newsgroup*. Le second correspond à ceux mentionnés dans les articles proposés par l'usager dans sa rétroaction. Tous les articles contenant ces newsgroupes seront associés au profil considéré.

```
newsgroups:   0.2
              clari.news.cast                  1
              clari.news.gov.international     1
              clari.news.hot.east_europe       1
              clari.news.hot.ussr              1
              clari.news.top.world             1
locations:    0.1
learningrate: 0.05
numarts:      5
keywords:     0.7
              ukraine                          0.475
              nuclear                          0.294
              weapons                          0.290
              soviet                           0.239
              tactical                         0.224
              somewhat                         0.224
              inherited                        0.215
              encouraged                       0.207
              missiles                         0.201
              kiev                             0.201
              funds                            0.164
              additional                       0.158
              returned                         0.149
              washington                       0.148
              ...
————————ArtScores————————
clari.news.gov.international:<boutrosghali-norkorURecb_3DP@clarinet.com>   0.0902
clari.news.gov.international:<year-diplomacyUR4c4_3DN@clarinet.com>        0.0458
clari.news.gov.international:<boutrosghali-japanUM8f9-22b@clarinet.com>    0.0457
clari.news.gov.international:<boutrosghali-koreaUR819_3DO@clarinet.com>    0.0452
clari.news.hot.east_europe:<year-republicsURaa1_3DN@clarinet.com>          0.0450
————————ArtFeedback————————
clari.news.gov.international:<boutrosghali-japanUM8f9-22b@clarinet.com>    +1
clari.news.gov.international:<boutrosghali-koreaUR819_3DO@clarinet.com>    -1
clari.news.hot.east_europe:<year-republicsURaa1_3DN@clarinet.com>          +1
————————ArtRead————————
clari.news.gov.international:<boutrosghali-japanUM147-22c@clarinet.com>
clari.news.hot.east_europe:<russia-polandUR979_3DD@clarinet.com>
clari.news.gov.international:<france-natoUR349_3DE@clarinet.com>
clari.news.hot.east_europe:<ukraine-russiaUR55b_3DE@clarinet.com>
clari.news.hot.east_europe:<russia-usUR90c_3DG@clarinet.com>
clari.news.top.world:<poland-walesaUM902-227@clarinet.com>
clari.news.hot.east_europe:<russia-germanyUM94b-22a@clarinet.com>
...
```

Fig.16 Exemple de profile

Un article typique (**Fig.17**) est constitué de deux parties : une partie structurée (*l'en-tête*), l'autre non structurée (*le texte*). L'*en-tête* varie d'une source à l'autre, mais il existe un certains nombre de lignes obligatoires pour que l'article respecte la forme standard **[HORTON ; 87]** (Date, From, Subject). Les autres (Index Keywords, Lines, Sender, Location) sont optionnelles.

Chaque article candidat est convertit dans sa représentation vectorielle avant d'être annoté.

3.1 Extraction de la représentation des documents

Une collection de documents contient tous les articles contenant les newsgroupes contenus dans le profils ou mentionnés dans la rétroaction. La partie texte de chaque article est indexée. L'indexation se fait en deux étapes car ***tf*** est calculé pour chaque article, tandis que ***idf*** est calculé par rapport à la collection entière (voir **chap.III**).

3.2 Annotation et sélection des documents

Les représentations vectorielles sont maintenant prêtes, le score de chaque article est déterminé en calculant le produit scalaire entre les deux représentations (profile-document). Les champs considérés par **YAIF** sont : *newsgroups*, *Location*, *Authors* et *Keywords*. Le score total du document est une combinaison linéaire des scores des différents champs.

Le nombre d'articles sélectionnés pour chaque profil est proportionnel à sa *finesse*, quand au nombre d'articles à présenter, il peut être fixé par l'usager lui même.

Xref: news.media.mit.edu clari.news.gov.officials:3216
clari.news.gov.international:46214
From: clarinews@clarinet.com (NICK DRIVER)
Newsgroups: clari.news.gov.officials,clari.news.gov.international
Subject: Economic takeoff masks pitfalls ahead for China
Keywords: international, non-usa economies, economy, domestic economy, government officials, government
Copyright: 1993 by UPI, R
Message-ID: <Xyear-chinaUR6f9_3DN@clarinet.com>
X-Supersedes: <year-chinaUR6f9_3DN@clarinet.com>
References: <yearURa3c_3DN@clarinet.com> <yearUR254_3DN@clarinet.com>
Date: Thu, 23 Dec 93 16:48:40 PST
Location: china
ACategory: international
Slugword: year-china
Priority: release-at-will
Format: annual, feature
ANPA: Wc: 880/828; Id: z7864; Src: upi; Sel: xxief; Adate: 12-23-N/A; V: atwill
Approved: clarinews@clarinet.com
Codes: yiefyxx., yiedfch., yigoyxx.
Note: (850) release at will
Lines: 96

BEIJING (UPI) -- China's own predictions of its revival as a world power moved toward reality in 1993 as its economic engine came to life, triggering an impressive array of achievements.
While acknowledging the social dislocations and economic dangers their capitalist-style market reform program has engendered, Chinese leaders have avoided the difficult political questions that will be posed by the death of frail 89-year-old senior leader Deng Xiaoping.
With a 1993 gross national product growth rate of 13.5 percent, China's economic takeoff has captured the hearts, minds and wallets of domestic and overseas investors.
Foreign investors have come flocking to the ''Middle Kingdom'' and to Hong Kong's stock market in search of the riches that only 1.2 billion potential consumers could produce.
Domestic consumers have reacted to the new economic realities by spending more, saving less and establishing thousands of new private enterprises. The private economy now accounts for more than half the country's production by some official estimates.
But the economic boom also has driven up prices and increased urban...

Fig.17 Article de USENET

CHAPITRE VIII

FILTRAGE PAR COLLABORATION

(Collaborative filtering)

1. Introduction
2. Principe
3. Exemple de système de filtrage par collaboration : *Tapestry*

1.Intoduction

Il existe actuellement des filtres commercialisés, d'autres sont à la phase de prototype, mais les résultats obtenus ne sont pas encore assez satisfaisants. A ce jour, le filtrage reste limité à des domaines bien déterminés et ne répond qu'à des profils bien définis. La grande difficulté étant de définir un bon profil. Beaucoup d'études sont menées dans ce sens et commencent à avoir des applications directes ou indirectes dans les processus de recherche d'information, mais l'étendue de ces applications au filtrage reste encore mal maîtrisé et présente plus de difficulté **[MALONE ; 92].**

Une nouvelle génération de filtres commence à voir le jour, basée sur un filtrage par collaboration (*collaborative filtering)*. Cette méthode a repris les mêmes principes que le filtrage classique, avec en plus, une collaboration entre les utilisateurs qui peuvent ainsi s'entraider et aider le filtre a répondre à leur besoins. (**Fig.18**)

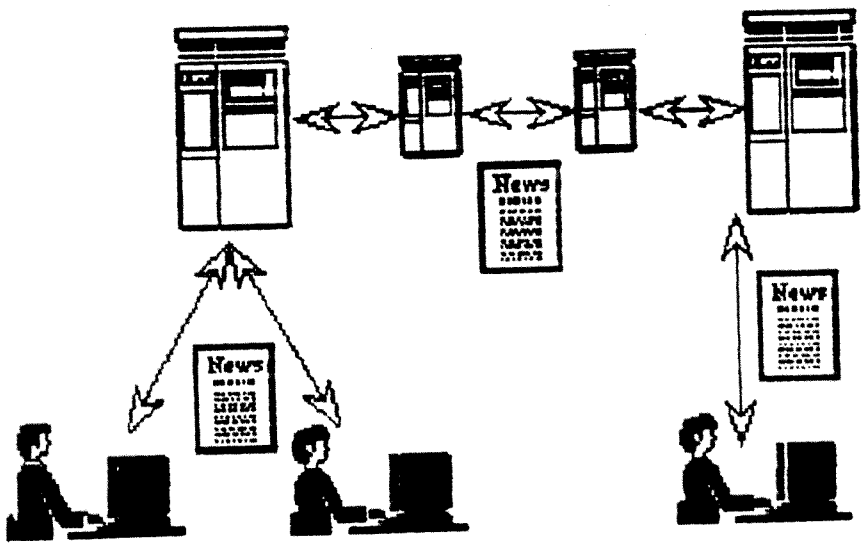


Fig.18 Principe de collaboration

2. Principe

Les utilisateurs participent activement à alimenter une base de données gérée par le filtre qui contient des informations concernant d'une part l'utilisateur, d'autre part, les documents qu'il a pu consulter et annoter. L'usager peut exprimer ses réactions sur un document, par exemple si ce document était particulièrement intéressant ou non (**Fig.19**). Ces réactions seront annotées et pourront êtres consulter par d'autres usagers[29]). Autre avantage de cette méthode, est qu'il existe des relations document-document et document-usager, ainsi si une personne recherche des informations sur un sujet donné mais qu'elle ne dispose pas de tous les éléments qui le caractérisent (ce qui arrive souvent lorsqu'on débute une recherche), en faisant une recherche classique par interrogation de bases de données ou les services de l'Internet. très souvent une telle démarche s'accompagne de beaucoup de bruit et c'est là qu'intervient le filtre. Il pourra lui proposer directement la liste des personnes travaillant sur le même sujet (**Fig.21**) , et lui sélectionner les documents qu'elles ont pu consulter. Si parmi ces documents, certains sont pertinents, le filtre pourra trouver l'ensemble des documents ayant les mêmes annotations que celui-ci **[LASHKARI ; 94]**.

[29] Il existe déjà des applications similaires dans les modérateurs de *newsgroupes.*

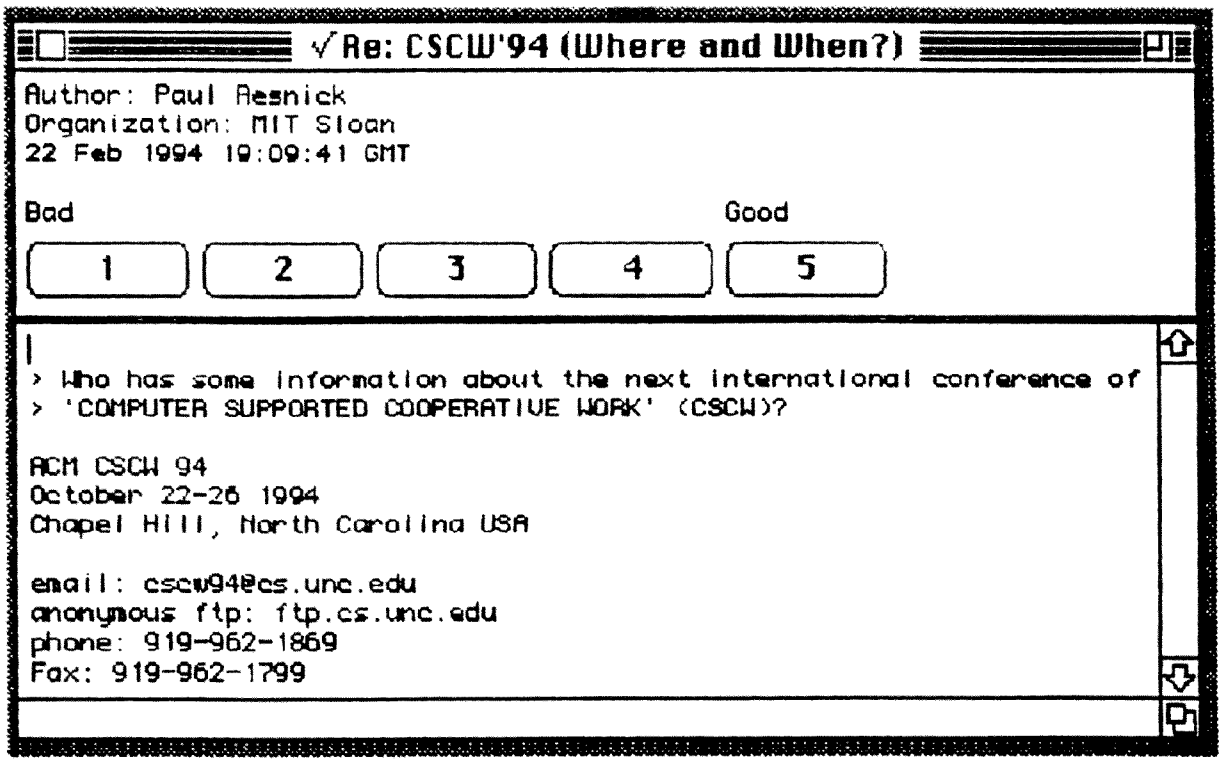


Fig.19 Annotation d'un article par l'usager

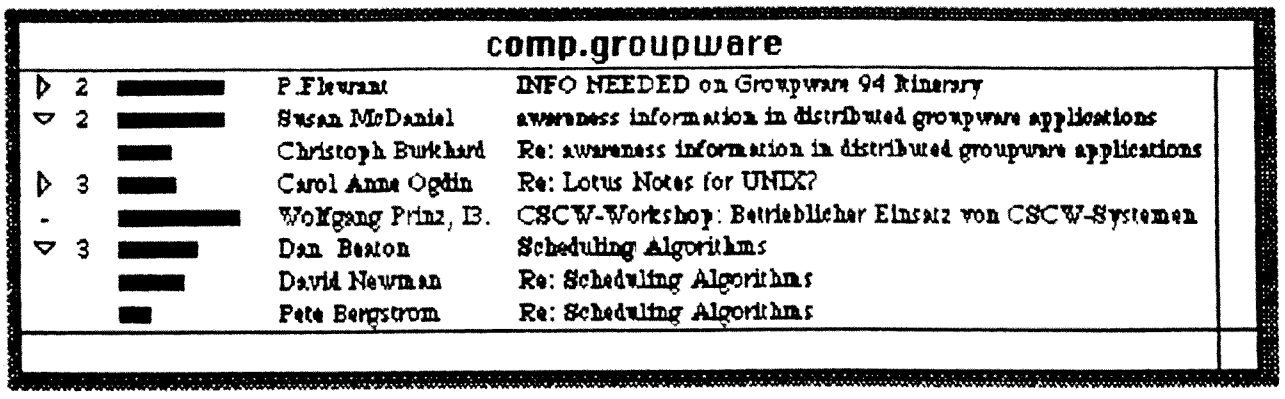


Fig.20 Annotation d'un article par les usagers

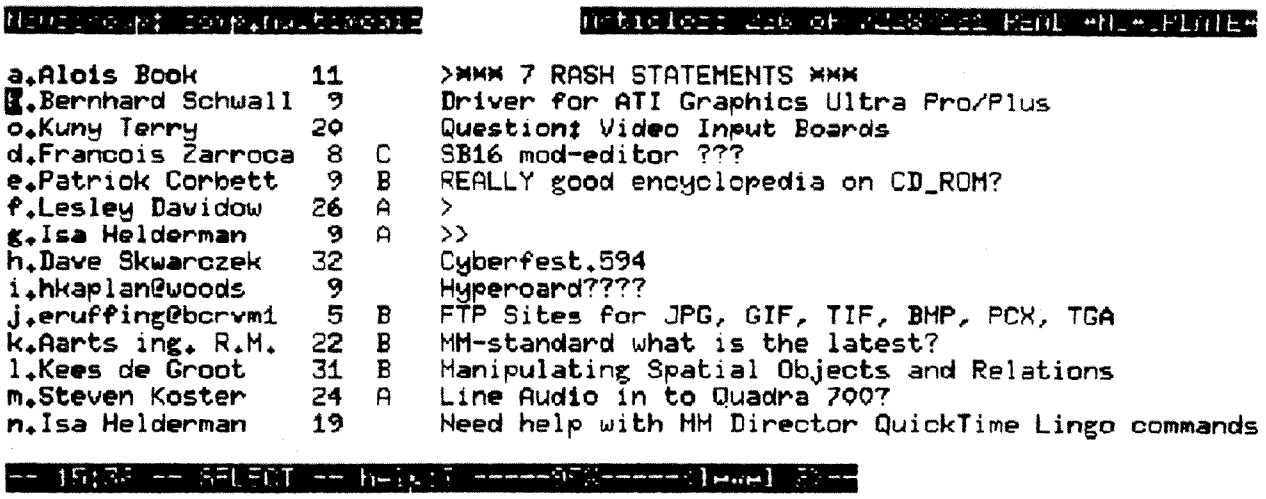


Fig.21 Listes des personnes travaillant dans un même domaine avec leurs annotations

3. Exemple d'un système de filtrage par collaboration : *Tapestry*

Le schéma d'un tel filtre est représenté sur la **Fig.22**.

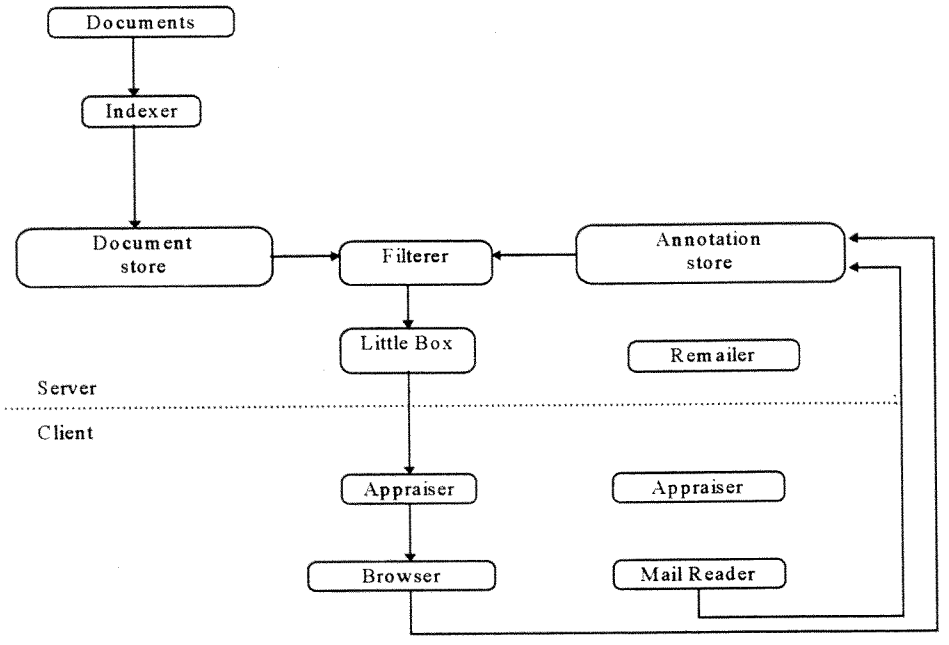


Fig.22 schéma de fonctionnent de *Tapestry*

- **l'Indexeur** *(Indexer)* : indexe les documents récupérer dans sources externes. L'indexation se fait soit sur le texte intégral *(fulltext)* soit sur des champs que l'utilisateur aura spécifié dans sa requête.
- **Réserve de documents** *(document store)* : tous les documents indexés y sont stockés. Le stockage peut être par texte intégral ou mots clés.
- **Réserve d'annotations** *(Annotaton store)* : les annotations associées aux documents sont gérés par cette partie.
- **Le filtre** *(filter)* : relie les nouveaux documents aux annotations.
- **la petite boîte** *(little box)* : chaque utilisateur dispose d'une boite où seront disposées les documents sélectionnes par le filtre.

- **le posteur** (*Remailer)* : Envoi périodiquement le contenu de la boîte personnelle à l'utilisateur *via* courrier électronique.
- **Evaluateur** *(Appraiser)* : Cette phase pourrait être automatisée, mais dans les prototypes actuels, c'est l'utilisateur qui évalue la pertinence des documents reçus et pourra ainsi modifier son équation de recherche.
- **stockage des documents et de leurs annotations** (*Document and annotation stores)* de nos jours, le stockage d'information ne pose pas de problème de mémoire, il existe de plus en des bases de données textuelles, il serait donc tout à fait envisageable de stocker tous les documents récupérés de l'extérieur, cependant la gestion d'un très grand nombre de documents pourrait évidement entraver le travail de l'utilisateur, il est nécessaire de rappeler que le but du filtrage est de réduire au maximum la quantité tout en conservant la qualité des documents, de ce fait les annotations sont stockées et gérées par un système indépendant avec des relations de renvois annotation-document. A première vue il parait plus normal de stocker les documents et leurs annotations dans un même système, en définissant par exemple un champs annotations qu'on peut consulter directement sans avoir à lire la totalité du document. un tel système a fait ses preuves dans les bases de données actuelles ou le champs mots clés permet d'avoir une idée sur le contenu du document. Les premiers travaux ont adopté ce système, mais très vite il s'est avéré lourd à gérer et ne répondais pas aux attente des utilisateurs. Plusieurs raisons ont donc pousser les chercheurs à adopter cette deuxième solution c'est à dire à séparer la gestion des documents et de leurs annotations. l'une des raisons essentielle et que les annotations sont parfois complexes et nécessitent donc un langage bien défini. (cf; Tapestry Query Langage). **[GOLDBERG; 92]**.

CHAPITRE IX

EXPERIMENTATIONS

Expérience n°1

Le but de cette première partie est de tester les performances de **SIFT** pour le filtrage d'articles scientifiques. Dans notre cas, des articles de physique nucléaire. Quatre profiles différents traitant des sujets indépendants ont été adressés au système. Dans un premier temps les profiles n'ont subit aucune modification au cours des quatre semaines de souscription. Pour quantifier les performances du système, nous avons définit trois grandeurs :

- *le taux de précision* : exprimé en pourcentage, correspond à la proportion d'articles pertinents par rapport aux nombre total d'articles.
- *le taux de bruit* : exprimé lui aussi en pourcentage, correspond à la proportion d'articles non pertinents. (Par non pertinent, nous désignons les articles n'ayant pas répondu à la requête de l'usager mais qui ont un rapport avec son domaine de recherche).
- *le taux de **HS*** : correspond à la proportion d'articles **H**ors **S**ujet. Ces articles correspondent le plus souvent à des annonces, des messages publicitaires ou des informations sur des jeux.

Nos quatre volontaires étaient des étudiants doctorants en physique, tous étaient intéressés par des articles qui les maintiendraient à jour dans leur domaine de recherche, et tous n'avaient aucune expérience préalable en recherche documentaire aussi bien sur l'Internet que sur les bases de données traditionnelles. La première étape de notre travail consistait à leurs présenter une approche simple des problèmes liés à la recherche automatique d'information afin qu'ils situent notre problématique. La seconde étape avait pour but de leurs faire connaître les différents services de recherche et de diffusion de l'information sur l'Internet. Elle s'est déroulé sous forme d'une séance de deux heures de travaux pratiques qui leurs a permis de se familiariser avec les différents outils de recherche et de se rendre compte par eux même des problèmes de surabondance de l'information dans le réseau. Après leurs avoir présenter le principe de fonctionnement de **SIFT**, et les différents paramètres pour modéliser leur profile, on leurs a laissé la liberté du choix du profile, tout en leurs précisant qu'ils avaient la possibilité de lui apporter des modifications en cas de réponse insuffisante du système.

Usager n°1 : Canalisation des particules chargées dans cristal.
(Canalisation d'ions Au (78+) à 53 MeV/u).

Usager n°2 : Caractérisation de matériaux par faisceaux d'ions.

Usager n°3 : Fission des noyaux légers. Etudes des bases énergies.

Usager n°4 : Analyse et modélisation spectrotemporelle de l'exceplexe KRYPTON-ARGON.

L'inscription au service se fait avec un profile par l'interface **WWW** (**Fig.23**). Un profile est constitué d'un ensemble de mots clés couvrant les intérêts de l'usager, plus d'autres paramètres qui permettrons à **SIFT** de le gérer. Le système envoi à intervalles de temps réguliers (tous les cinq jours) les en-têtes des articles qu'il a sélectionné (**Fig.24**).

Email : joe@cs.oceanview.edu
Profile : online information services
Type : boolean
Period : 5
Expire : 9999
Lines : 20

Fig.23 Interface WWW de SIFT

```
From netnews@hotpage.stanford.edu Mon Jan 24 10:40:35 1994
To: joe@cs.oceanview.edu
Subject: Netnews - online information services

Subscription 1: online information services

Article: misc.activism.progressive.11965
From: hn0003@handsnet.org
Subject: HandsNet WEEKLY DIGEST 1/15-21
Score: 100
First 20 lines:
 HANDSNET WEEKLY DIGEST  January 15 - 21, 1994
 News from HandsNet's Information Forums
 HandsNet is a national, nonprofit network connecting organizations working
 on social and economic justice issues.  Members use HandsNet to make new
 contacts, work collaboratively and to find and publish information, news
 ....

 Article: ca.politics.38420
 From: rlm@helen.surfcty.com (Robert L. McMillin)
 Subject: GOV-ACCESS #5:Cal.Emergency Svcs.online + Net-fax + MINN Pub Info Net
 Score: 100
 First 20 lines:
  Jan. 22, 1994
  CALIFORNIA OFFICE OF EMERGENCY SERVICES INFO AVAILABLE ONLINE

  The state Emergency Digitial Information Service is working fine
```

Fig.24 Exemple de messages envoyés par SIFT

Si un article intéresse l'usager, il envoi au serveur un *e-mail* avec les références de l'articles dans le corps du message (**Fig.25**). L'usager reçoit par la suite, l'article en entier.

```
From joe@cs.oceanview.edu Mon Jan 24 10:43:28 1994
To: netnews@hotpage.stanford.edu
Subject: you can leave this blank

get misc.activism.progressive.11965 ca.politics.30420
end
```

Fig.25 Message à retourner à SIFT

L'expérience s'est déroulée pendant quatre semaines (six pour l'usager n°3), Pour des raisons pratiques nous avons décidé de récupérer tous les articles proposés par le système. Les articles étaient imprimés et présentés aux intéressés qui les classaient en trois catégories :

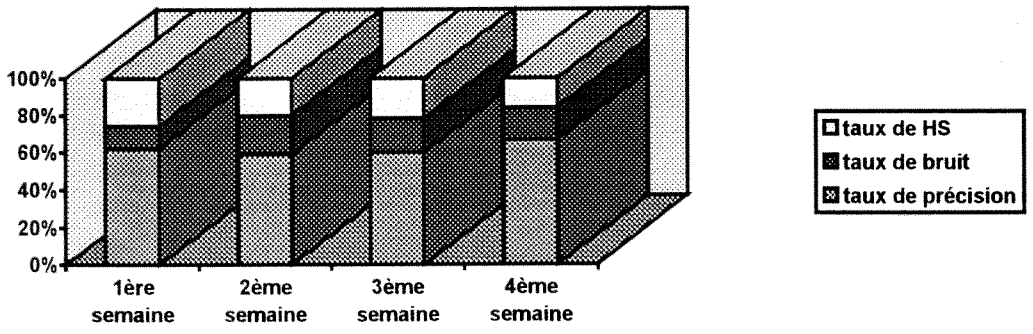
- Articles pertinents
- Articles non pertinents
- Articles Hors Sujet (HS)

Le nombre moyen d'articles envoyés était de 12,2 par semaines. Les résultats de cette première étude sont présentés ci dessous.

Résultats :

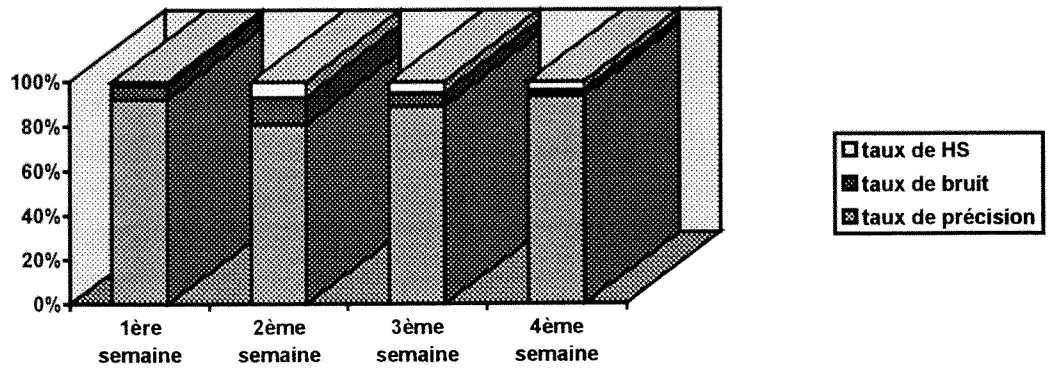
Tab.1 (Usager n°1) :

Période	1ère semaine	2ème semaine	3ème semaine	4ème semaine
taux de précision	62.3	59.2	60.1	66.9
taux de bruit	12.2	20.7	18.3	17.2
taux de HS	25.5	20.1	21.6	15.9



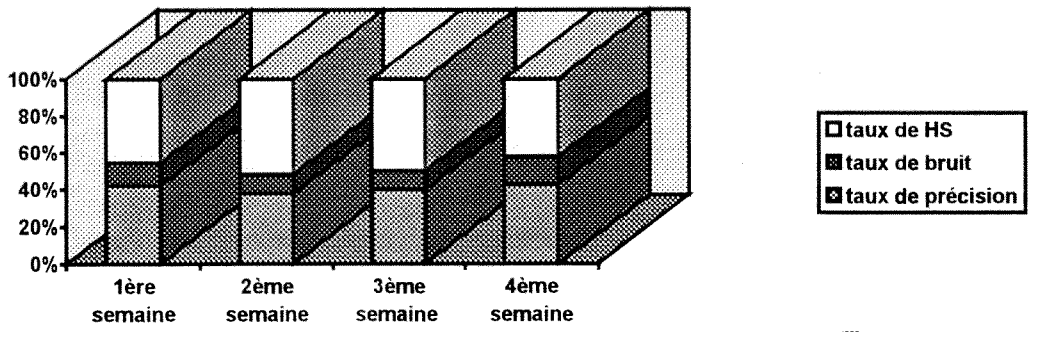
Tab.2 (Usager n°2):

Période	1ère semaine	2ème semaine	3ème semaine	4ème semaine
taux de précision	92.3	81.0	89.2	93.7
taux de bruit	6.1	12.3	5.9	2.6
taux de HS	1.6	6.7	4.9	3.7



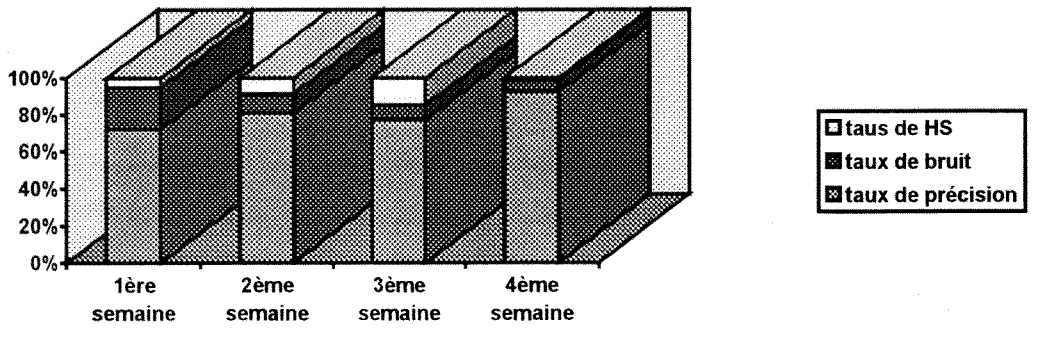
Tab.3 (Usager n°3) :

Période	1ère semaine	2ème semaine	3ème semaine	4ème semaine
taux de précision	42.3	37.8	40.0	42.8
taux de bruit	12.3	10.7	10.1	15.2
taux de HS	45.4	51.5	49.9	42.0



Tab.4 (Usager n°4) :

Période	1ère semaine	2ème semaine	3ème semaine	4ème semaine
taux de précision	72.3	80.8	77.0	92.3
taux de bruit	22.5	10.5	8.3	6.5
taus de HS	5.2	8.7	14.7	1.2

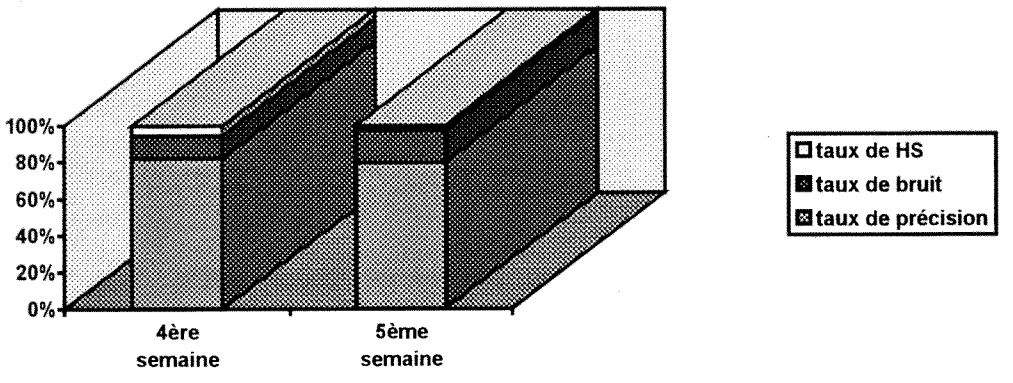


Interprétation des résultats :

Ces quatre histogrammes mettent en évidence le rapport précision/bruit. A l'exception du profile n°3, le taux de réponse du système est assez satisfaisant. Pour un nombre d'articles d'environ 12,2 par semaines, 80.3 % sont pertinents pour l'usager. Dans le cas du profile n°3, le rapport précision/bruit est d'environ 1. Dans ce cas, il nous a semblé nécessaire de modifier le profile, en effet, plus de 70% des articles **HS** concernaient un « *jeu de rôle* ». On a modifié le profile en utilisant le modèle booléen en éliminant le mot jeu du profile. (***not game***). et on poursuivi l'expérience pendant deux autres semaines. On a retrouvé le même taux de précision que les cas précédents soit environ 80%. (**tab. 5**)

tab.5 (Usager n°3 bis) :

Période	4ère semaine	5ème semaine
taux de précision	82.2	79.7
taux de bruit	12.3	17.9
taux de HS	5.5	2.4

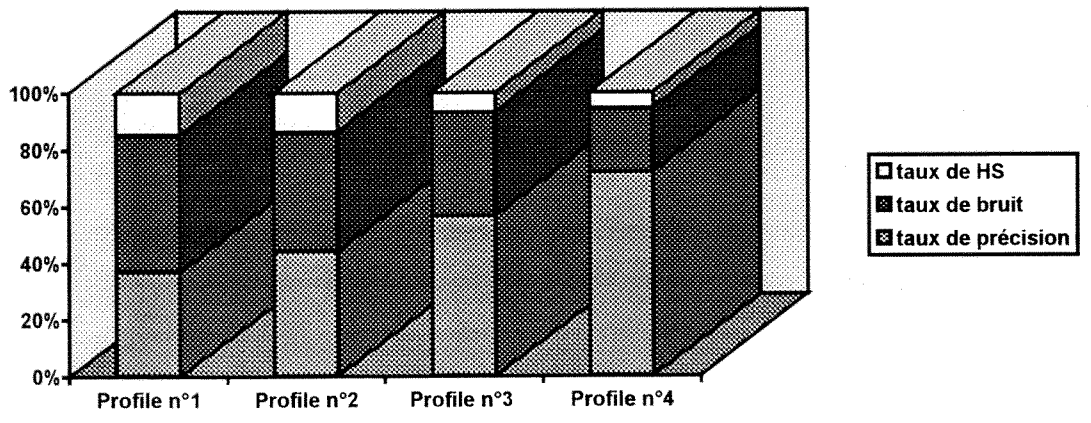


Expérience n°2

Dans cette partie, nous avons essayé de faire un filtrage en mode interactif. Le profile de l'usager n'a pas été défini par l'intéressé lui même, à partir de trois articles qu'il nous a présenté, nous avons construit un profile sans aucune connaissance dans son domaine de recherche (*Physique des agrégats*). En fonction des articles qu'il a accepté ou rejetés , nous avons affiner le profile jusqu'à l'obtention d'un taux de réponse convenable. L'expérience s'est déroulé pendant une période de quatre semaines. Les résultats de cette étude sont représentées dans le **tab.6**.

tab.6

	Profile n°1	Profile n°2	Profile n°3	Profile n°4
taux de précision	37.2	44.5	56.8	72.3
taux de bruit	48.2	42.0	36.4	22.2
taux de HS	14.6	13.5	6.8	5.5



La réponse au premier profile se caractérise par un taux de bruit supérieur au taux de précision. En ajustant le profile, on observe une inversion de la tendance pour arriver à des résultats qui se rapprochent à ceux de l'expérience.1. Cette expérience met en évidence deux choses, d'une part une application de la programmation par démonstration (Programming by demonstration), d'autre part, le phénomène de *reformulation*, à ne pas confondre avec le *feedback*. En effet, le *feedback*, tel qu'il a

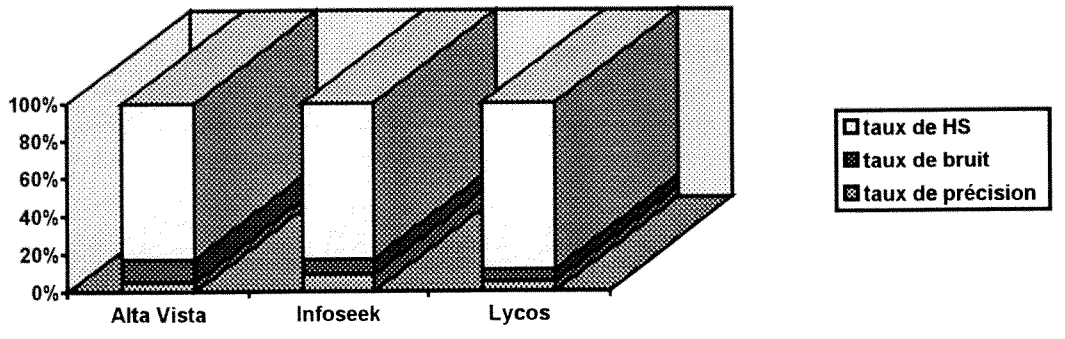
était définit dans le **chap. III**, consiste à augmenter ou diminuer le poids des mots qui appartiennent aux documents jugés pertinents ou non. La version actuelle de **SIFT** ne permet pas ce genre de rétroaction.

Expérience n°3

Dans cette partie, nous avons tester les trois outils de recherche les plus connus sur l'Internet, **Infoseek**, **Alta Vista** et **Lycos**. Contrairement un système de filtrage où les intérêts de l'usager sont exprimés sous forme d'un profile, dans un système de recherche, les intérêts de l'usager sont exprimés sous forme d'une requête (voir **chap.III**) qui est une combinaison de mots clés. Chaque service a été interrogé par une requête constituée de 1, 2, 3, 4 puis 5 mots clés traitant d'un même sujet. L'expérience a été reprise cinq fois. Et on n'a pris en considération que les douze premières réponses proposées par chaque système. Comme précédemment un calcul statistique[30] nous a permis d'obtenir les résultats suivants.

Tab.7 (1 mot clé)

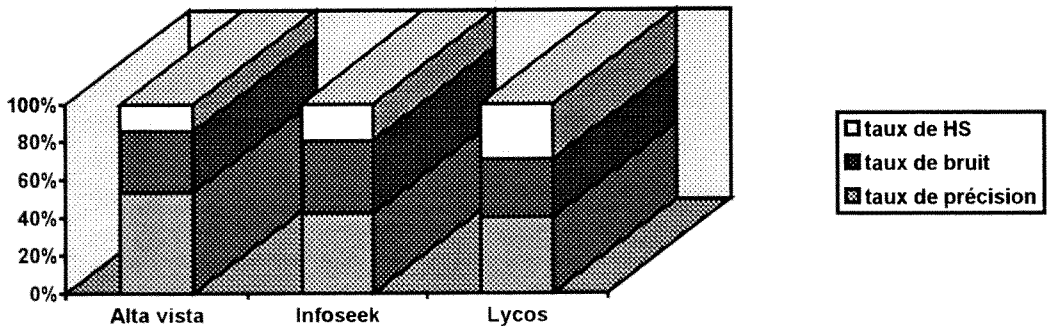
	Alta Vista	Infoseek	Lycos
taux de précision	5.2	10.1	5.6
taux de bruit	12.3	8.6	6.0
taux de HS	82.5	89.3	88.4



[30] Calcul de la moyenne des résultats de chaque expérience.

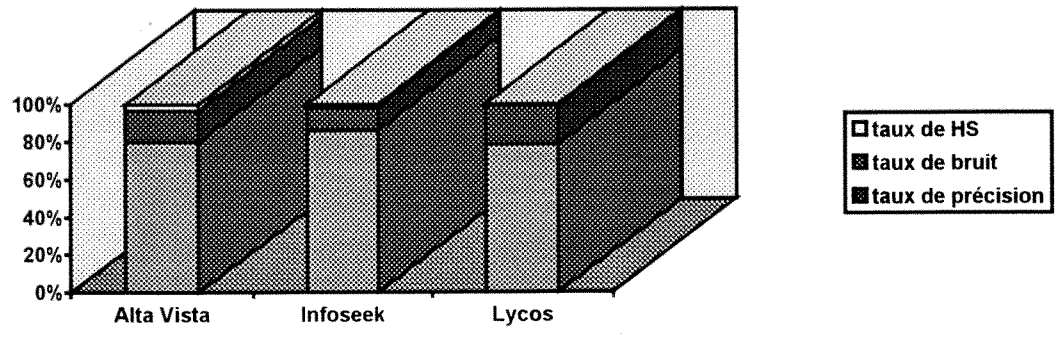
tab.8 (2 mots clés)

	Alta vista	Infoseek	Lycos
taux de précision	53.6	42.1	40.2
taux de bruit	32.1	38.2	30.3
taux de HS	14.3	19.7	29.5



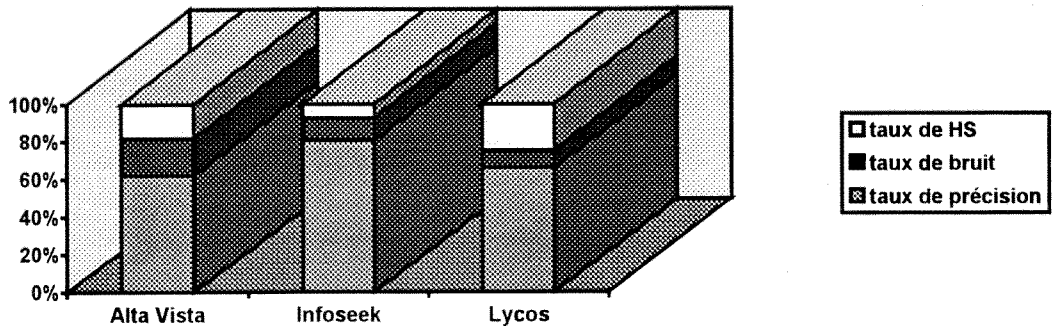
Tab.10 (3 mots clés)

	Alta Vista	Infoseek	Lycos
taux de précision	80.2	86.3	78.7
taux de bruit	16.1	12.2	20.5
taux de HS	3.7	1.5	0.8



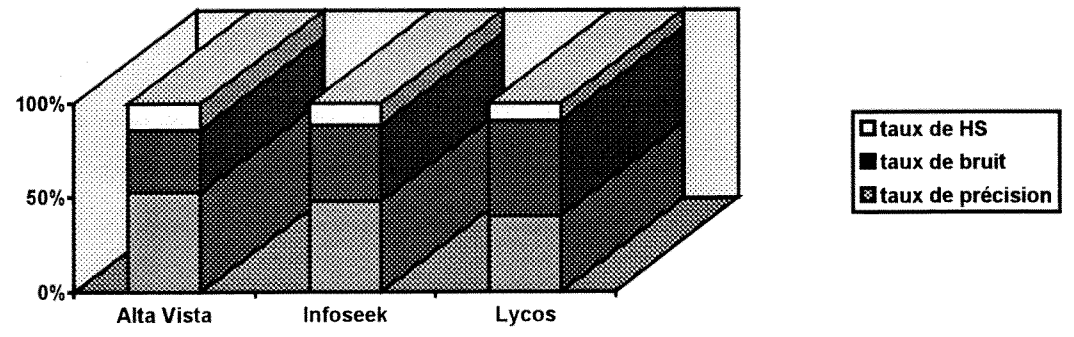
Tab.11 (4 mots clés)

	Alta Vista	Infoseek	Lycos
taux de précision	62.2	80.7	72.7
taux de bruit	19.5	12.0	10.3
taux de HS	18.3	7.3	27

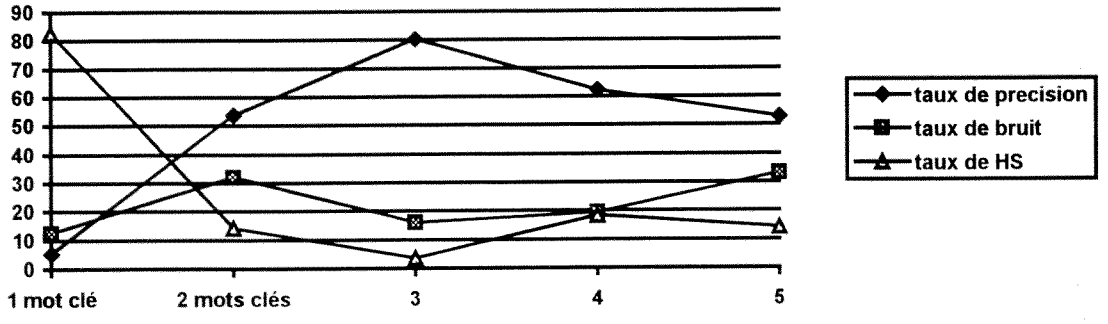


Tab.12 (5 mots clés)

	Alta Vista	Infoseek	Lycos
taux de précision	52.7	47.9	40.2
taux de bruit	33.2	40.3	50.5
taux de HS	14.1	11.8	9.3



Cette étude nous permet de faire deux observations. La première concernant l'évolution de la pertinence des moteurs de recherche en fonction du nombre de mots clés utilisés pour décrire les intérêts de l'usager. Dans le **tab.12**, on a représenté les valeurs moyennes[31] du taux de précision, de bruit et de HS, en fonction du nombre de mots clés pour l'ensemble des moteurs. Ces courbes montrent que pour l'utilisation d'un seul mot clé, le taux de **HS** est très élevé (environ 80%) tandis que la pertinence est très faible (inférieur à 7%). En augmentant le nombre de mots clés ; pour 2 et 3 mots, on observe une augmentation du taux de pertinence avec une valeur maximale de 80% pour trois mots clés tandis que la somme des taux de bruit et de HS se stabilise à 20%. Au delà de cinq mots clés, on observe une chute du taux de précision (< 2.5%) et une telle recherche devient sans aucun intérêt.



Tab.13 Evolution des résultats de recherche en fonction du nombre de mot clés

La seconde observation concerne une éventuelle substitution d'une requête par un profile. Comme nous l'avons vu dans le **chap. III**, un profile est constitué d'un grand nombre de descripteurs regroupés en champs. Souvent le champs mots clés a lui seul comporte une vingtaine de termes. Dans le cas du filtrage, plus le nombre de descripteurs est grand, plus on a de chance de s'approcher du profile de l'usager, alors que dans le cas de l'utilisation des moteurs de recherche, au delà de cinq mots clés, le système a une réponse quasi aléatoire.

[31] Ces valeurs moyennes ont été calculées par rapport aux résultats des trois moteurs de recherche.

Expérience n°4 :

Pour la suite de notre travail, nous avons constitué une base données contenant 1 200 articles de physique nucléaire couvrant quatre sujets différents (les mêmes que dans l'expérience n°1). Ces articles ont été obtenu par interrogation du **Web** en utilisant l'outil de recherche **Alta Vista** qui offre une recherche directe dans **Usenet** (voir **chap.I**). Notre objectif initial était d'utiliser les articles récupérés par **SIFT**. Ceci n 'a pas pu se faire pour des raisons techniques. (problèmes de compatibilité de logiciels). Ces articles ont été stockés sur disque dur.

Nous avons choisi **INFOSCAN** comme outil de filtrage. Pour plus de détail sur le fonctionnement de **INFOSCAN**, se conférer au **chap.V**. Nos quatre volontaires n'avaient aucune idée préalable sur le contenu de la base de données. A partir des articles qu'ils récupérés dans l'expérience n°1, ils ont construit eux même leurs profiles. Chaque profiles est constitué de cinq filtres avec une pondération décroissante de 5 à 1. Les filtres ont été organisés comme suit :

- Filtre n°1 (+5)[32] : termes dans le champs titre.
- Filtre n°2 (+4) : termes dans le champs newsgroupe.
- Filtre n°3 (+3) : termes dans le champs auteur.
- Les filtre n° 4 (+2) et 5 (+1) correspondent aux termes significatifs[33] dans le texte.

Pour quantifier nos résultats, nous avons utilisé quatre grandeurs :

- *le taux de rappel* : se définit comme étant le nombre de documents pertinents sélectionnés divisé par le nombre total de documents pertinents dans la base.
- *le taux de précision* : se définit comme étant le nombre de documents pertinents sélectionnés divisé par le nombre de documents sélectionnés.

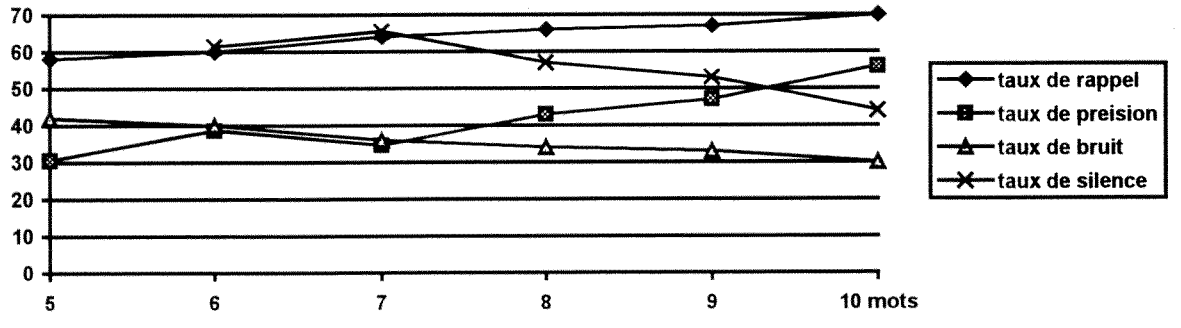
[32] Les valeurs entre parenthèses correspondent au poids attribué aux termes.
[33] Dans notre étude, on n'a considéré que les termes ayant un lien directe avec le sujet de recherche. ON a éliminer tous les articles, pronom, verbes..

- *le taux de bruit* : se définit comme étant a différence entre 1 (ou 100 si les résultats sont exprimés en pourcentage) et le taux de précision.
- *le taux de silence* : se définit comme étant la différence entre 1 (ou 100) et le taux de rappel.

Comme dans la plus part des expérimentations en recherche documentaire, la méthode d'évaluation des réponses la plus fiable est la méthode comparative. Cela consiste d'abord à effectuer un filtrage automatique, puis faire un dépouillement manuelle de la base de données et comparer les résultats. Pour éviter tout risque d'introduire une valeur ajoutée, le dépouillement n'a eu lieu, qu'une fois toutes les opérations automatiques effectuées.

Expérience n°4.1

Dans cette partie, nous avons essayer de tester l'efficacité du filtre pour mieux comprendre son fonctionnement. Nous avons décidé qu'un profile doit contenir cinq mots clés minimum (limite d'un outil de recherche) sans préciser à quel champs ils doivent appartenir (tous les termes ont la même pondération) puis on a augmenter le nombre de mots clés. Les résultats de cette étude sont regroupés dans **Tab.13**



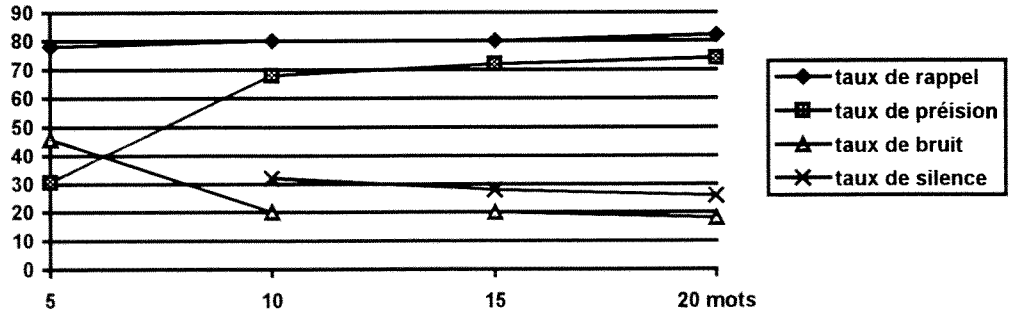
Tab.13 Evolution des résultats du filtrage en fonction du nombre de mots clés

Remarques :

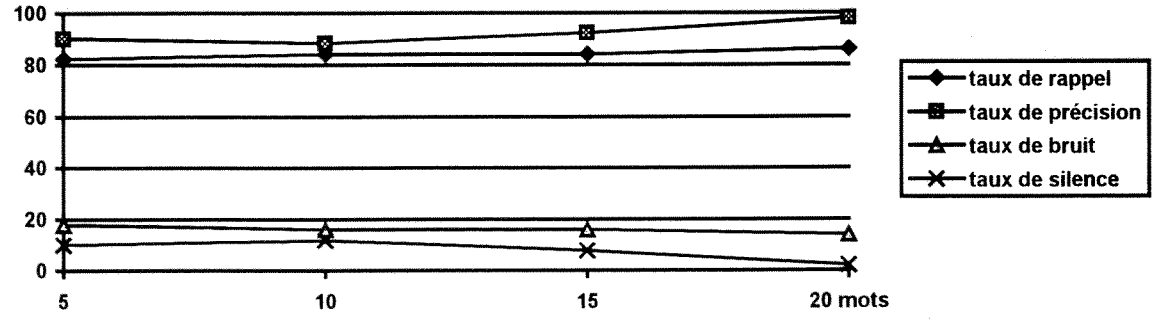
Les valeurs en ordonnées correspondent à des moyennes calculées par rapport aux quatre profiles.

Expérience n°4.2 :

Dans cette deuxième partie, on a utilisé la pondération des termes. Dans un premier temps, on ne s'est servi que du filtre n°1 (champs titre), puis on a combiner les cinq filtres. Les résultats dans cette étude sont représentés dans **Tab.14** et **Tab.15**.



Tab.14 Evolution des paramètres de filtrage en pondérant le champs titre



Tab.15 Evolution des paramètres de filtrage en pondérant tous les champs

Ces deux courbes montrent l'importance de la pondération des différents champs, ce qui distingue un outils de filtrage d'un outil de recherche d'information (voir **chap.III**). En effet, dans le cas de la pondération du champs mots clés seulement, on observe un faible taux de précision et un taux de bruit assez élevé pour un nombre de mots clés inférieur à 10, on utilisant toujours le même nombre de mots clés, mais en pondérant les différents champs, on arrive à inverser cette tendance.

CONCLUSION

Dans ce mémoire, nous avons essayé de présenter les différentes approches utilisées pour le filtrage de l'information sur l'Internet (dans notre cas, le filtrage d'articles de *news*). La première approche, l'utilisation de système de filtrage sur le réseau, connaît un grand succès sur l'Internet. Il existe aujourd'hui plusieurs serveurs qui offrent ce mode de filtrage. Notre étude expérimentale avec **SIFT**, dont le principe de fonctionnement est similaire à la plus part des services que nous avons recensés, nous permet de faire deux constats : d'une part, ce genre de service a plus une fonction de diffusion sélective d'information qu'un service de filtrage tel qu'il a été défini dans la littérature. D'autre part, le fait que l'usager reçoit passivement les articles sélectionnés sans avoir aucune idée, ni sur le contenu de la base de données, ni sur le fonctionnement du système pourrait mettre en cause sa confiance pour lui léguer sa tâche de filtrage même s'il a la possibilité d'ajuster son profil. La seconde approche, l'utilisation de filtres locaux, nous semble plus pratique et plus efficace. Elle répond mieux aux besoins de l'usager qui a le contrôle de tous les paramètres. En plus, un

grand travail a été fait pour rendre les interfaces graphiques très conviviales, **Infoscan** en est un bon exemple. La troisième approche utilisant des « Agents intelligents » nous semble bien complexe. Certes l'intelligence artificielle a fait ses preuves dans beaucoup de domaines, mais aura-t-elle une place dans l'Internet ? Un Agent pourra-t-il réellement remplacer l'intuition humaine ? Nous avons étudié le principe de fonctionnement de trois prototypes d'Agents qui ont fait l'objet de thèses, Les trois avaient des algorithmes très complexes, mais leur principe de fonctionnement était semblable à ceux des filtres classiques, avec quelques fonctions supplémentaires. Quant à la quatrième approche, le filtrage par collaboration, elle nous semble la plus appropriée pour le filtrage de l'information sur l'Internet. En effet, la notion de pertinence dans le cas de documents récupérés sur l'Internet prend une dimension différente par rapport à celle qu'on connaît dans la littérature concernant la recherche d'information dans les bases de données traditionnelles. Dans nos études expérimentales, un grand nombre de documents était pertinent, dans le sens, réponse à la requête, mais en réalité une grande proportion de ces documents n'était qu'une modification d'articles et parfois des copies intégrales d'articles se trouvant dans d'autres *newsgroups*. En dénombrant les articles qui apportaient des informations réellement nouvelles, le taux de pertinence ne dépassait pas les 20% (au lieu de 80%). Un autre problème que l'on pourrait se poser : comment pourrait-t-on vérifier la fiabilité de cette information ? Aucun contrôle n'existe actuellement. Le filtrage par collaboration pourrait s'avérer intéressant pour assurer, en plus du filtrage, ce contrôle.

BIBLIOGRAPHIE

CHAPITRE I & II

COURTOIS, MARTIN P. ; WILLIAM M. ; MARCELLA S. : Cool Tools for searching the Web : A Performance Evaluation. In : Online ; 1995 ; pp. 15-32.

IHADJADENE MADID : hypertexte : Modèles de navigation
In : Mémoire de DEA SIC ; 1993 ; enssib. Lyon.

LARDY, J.P. : Les moteurs de recherche
In : Les outils de recherche d'information dans l'Internet. URFIST Lyon.

LARDY, J.P. : Recherche d'information dans Internet : outils et méthodes.
In : ADBS EDITIONS ; 1996.

KOSTER, M. : Robots in the Web : threat or treat ? 1995.
URL : http ://web.nexor.co.uk/mak/doc/robots.html.

LEIGHTON, H. : Performance of Four World Wide Web (WWW) Index Services : Infoseek, Lycos, WebCrawer and WWWorm. 1995
URL : http ://www.winona.msus.edu/services-*f*/library-*f*/webing.html.

LIU, JIAN. : Understanding WWW Search Tools. Septembre 1995.
URL : http ://www.indiana.edu/~libcsd/search/.

Matrix of WWW Indices : A comparision of Internet indexing tools. 1995.
URL : http ://www.sils.umich.edu/~*f*prefect/matrix/.

PAUL, K. ; KATHLEEN,M. : Tools and techniques for Searching the Web : Subject Trees and Search Engines. 1995.
URL : http ://burns.library.uvic.cal/KWM_Pest_CLA.html.

PLOURDE, J. N.: Critères et évaluation d'outils de recherche des ressources dans Internet. 1996. URL : http ://mistral.ere...11no2/plourde.html.

STANLEY,T. : Searching the World Wide Web with Lycos and Infoseek. Oct 1995.
URL : http ://www.leeds.ac.uk/ucs/docs/fur14/fur14.html.

WINSHIP, I. R. : World Wide Web searching : An evaluation. 1995.
URL : http ://www.bulb.bath.ac.uk/BULB/Iwinship.html.

CHAPITRE III

ARENSBURGER, A. ; ROSENFELD, A. : To take Arms Agains a saa of Mail. In : Commun. ACM 38, 1995, 3 March, pp. 108-109.

BACLACE, P.E. : Personal Information Intake Filtering.
In : Bellcore Information Filtering Workshop, November 1991.

BELKHEIR, M. : Le filtrage de l'Information sur l'Internet.
In : note de synthèse DEA SIC, enssib, Lyon, 1995.

BELKIN, N.J. ; CROFT, W.B. : Information Filtering and Information Retrevial : Two Sides of the Same Coin ?. In : Communication of the ACM, Vol.35, No.12, pp.29-38, 92.

BELKIN, N.J. and Croft, W.B.: Retrevial techniques In Annual Review of Information Science and Technology. M.E Williams Ed. Chapt 4, pp 109-145. 1987.

BERNERS-LEE, T. ; CAILLIAU, R. ; GROFF, J. and POLLERMAN, B.: World-Wide Web : The Information Universe, Electronic Networking : Research, Application and Poilicy, Meckler Publications, 2(1), Spring 1992, pp.52-58.

DEERWESTER, S. ; Dumais, S.T. ; FURNAS, G.W. : Indexing by Latent Semantic Analysis. In : Journal of the American Society for Information Science, Vol.41, No.6, 1990, pp. 391-407.

FISCHER, G. ; STEVENS, C. : Information Access in Complex , Poorly Structured Information Spaces. In : Human Factors in Computing Systems CHI'91 Conference Proceedings, 1991, pp. 63-70.

FOLTZ, P.W. : Using Latent Semantic Indexing for information filtering. In : Proceegings of the ACM Conference on Office Information Systems, ACM/SIGOIS, New York, 1990, pp. 40-47.

FOLTZ, P.W, DUMAIS, T. : Personalized Information Delivery : An Analysis of Information Filtering Methods. In : Communication of the ACM, Vol.35, No.12, pp.51-60.

GOLDBERG, G. ; NICHOLS, D. ; OKY, B.M. ; TERRY, D. : Using Collaborative Filtering to Weave an Information Tapestry. In : Communication of the ACM, Vol. .35, No. 12, pp. 61-70.

LAI, K.; MALONE, T. : Objects Lens : A « SpeadSHEET » for Cooperative Work. In : ACM Transactions on Office Information Systems 5(4), 1988, pp. 297-326.

OARD, D. : Information Filtering Defined
URL : http :// meblab/filter/filter definition.html

POLLOCK, S. : A Rule-Based Message Filtering System.
In : ACM Trans. Off. Syst ; 1988 ; 3 july ; pp. 234-54.

RESNICK, P. ; LACOVOU, N. ; SUCHAK, M. ; BERGSTROM, P. ; RIELD, J. : An Open Architecture for Collaborative Filtering of Netnews. In : ACM Press, New York, 1994, pp. 175-86.

ROCCHIO, J.J. : Relevance Feedback in information Retrevial, the Smart System. In : Experiments in Automatic Document Proceessing, ed Salton, G., Prentice-Hall Inc., 1971, pp. 337-354.

SHETH, B. ; MAES, P.: Evolving Agents for Personalized Information Filtering. In Proceeding of the ninth IEEE Conference on AI for Applications, 1993.

STEVENS, C. : Automating the Creation of Information Filters.
In : Commun ACM 35 ; 1992 ; 12 Dec, pp .48.

SUCHACK, M.A. : GoodNews : A Collaborative Filter For Network News. SM Thesis, Department of Electronic Engeneering and Computer Science, MIT, Feb 1994.

CHAPITRE IV

ACHELEY, D. ; LITTMAN, M. : Interactions between Learing and Evolution.
In : Artificial life. Edited by C. Langton, C . Taylor ; 1991.

BACLASE, P. : Competetive Agent for Information Filtering.
In : commun. ACM 35, 1992, 12 Dec, pp. 50.

CHIN, D. : Intelligent Interfaces as Agents, Intelligent User Interfaces.
In : ACM Press, 1991, pp. 177-206.

CROFT, W. : NSF Center of Intelligent Information retrevial.
In : Commun. ACM 38, 1995, 4 April, pp. 42-43.

CYPHER, A. : Watch what I do : Programming by demonstration.
In : MIT press, Cambridge, MA, 1993.

DAVID, E. ; GOLDBERG, H. : Algorithme génétique.
In : Editions Addison-Wesley France, SA ; 1991.

DEJONG, K.A. : Adaptive system Design : A genetic approch. In : IEEE Transactions on systems, Man and cybernetics. Vol 10. No 19, 1980.

FUHR, N. and BUCKLEY, C. : Probabilistic document indexing from relevance feedback data. In : Proceeding of tht thirteen International Conference on Reseach and development in Information Retrieval, Jean-Luc VIDICK, Ed. ACM, New York, Sept ;1990 pp 45-61. 1990.

GANT, S. : A Portrait of Potential Adopters of Information Filter/
In : ISIS'95, Vol. 32. (Ed : Kinney, Tom) Information today, MedFord, Newjersey, 1995, pp. 167-171.

GAUSS, S. ; SMITH, J.B. : An Expert System for Seaching Full Text. In : Information Processing and Management. 25(3), 1989, pp 253-263.

GORDON, M. : Probabilisitc and Genetic Algorithms for document Retrevial and Their Application. In : Communication of the ACM, Vol. 31 No. 31, Oct 1988.

GRFENSTETTE, J.J. : Optimisation of control Parameters For Genetic Algorithm.
In : IEEE Transaction on System, Man and cybernetics, Vol. 16 No. 1, 1986.

HOLLAND, J.H. : Adaptation in Naturel And Artificial System, An Introductory Analysis with Application to biology, Control and Artificial Intelligence. In : University of Machigan Press, An Arbor, 1975.

HINTON, G.E. ; NOWLAN, S.G : How Learing Can Guide Evolution. In : Complex Systems 1 ; pp. 495-502 ; 1987.

KAY, A. : Computer Software. In : Scientific American ; 251(3) ; 1984 ; pp. 53-59.

KOZIEROCK, R. ; MAES, P. : A Learing Interface Agent for Schduling Meeting. In : ACM SIGCHI International Workshop on Intelligent User Interfaces. Foridia ; January 1993.

LASHKARY, Y. ; METRAL, M. ; MAES, P. : Collaborative Interface Agents. In : Proceeding of AAA'94 Conference Seattle, 31-Aug, 1994, pp. 1-6.

LOSEE, R. : Minimizing Information Overload : The ranking of electronic Messages. In : Journal of information, 15, 1989, pp. 179-89.

MAES, P. : Modeling Adaptive Autonomous Agents. In : Artificial Live journal ; Vol 1 ; Nos 1 & 2 ; MIT Press ; 1994.

MAES, P. ; KOZIEROCK, R. : Learing Interface Agent. In : Proceeding of AAAI ;1993.

MALONE, T.W. ; GRANT, S. : Intelligent Information-Sharing Systems. In : Communications of the ACM ; Vol. 30 ; No. 5 ; May 1987 ; pp. 390-402.

NEGPROPONTE, N. : The architecture Machine : Towards a more human Environment. MIT Press ; 1970.

PITRAT J. : Penser autrement l'informatique. In : collection perspectives ; Hermes, Paris, 1993.

SHETH, B. ; MAES, P. : Evolving Agent for Personalozed Information Filtering . In : Proceeding of the ninth IEEE Conference on AI for Application ; 1993.

CHAPITRE V & VII

CROCKER, D. : Standard for the format of ARPA Internet Text Messages (RFC 822). In : Network Information Center, SRI International. California ; 1982.

FOLTZ, P.W. and DUMAS, S.T. : Personilazed information delivery : an analysis of information filtering methods. In : Communication of the ACM, 35(12), pp.29-38 ; 1992.

GIFFORD, D. ; BALDWIN,S., BERLIN, A. : An architecture for large scale information systems. In : Symposium on Operating System Principles ; pp.161-170 ; 1985.

GOLDBERG, D. : Using collaborative filtering to weave an information tapestry. In : Communication of the ACM ; 35(12) ; pp. 61-67 ; 1992.

KAHLE, B. ; MEDLAR, A. : An information system for corporate users : Wide Area Information Servers. In : The Interoperability Report ; 5(11) ; pp.2-9 ; 1991.

MALONE, T. ; GRANT, K. : Intelligent Information sharing systems . In : The Communication of the ACM ; 30(5) ; pp. 390-402 ; 1987.

SALTON, G. : Automatic Text Processing. In : Addison Wesley, Reading, Masschusetts ; 1989.

YAN, T.W. ; GARCIA-MOLINA, H. : Distributed selective dissemination of information. In : Parallel and Distributed Information Systems ; pp. 89-98 ; 1994.

YAN, T.W. ; GARCIA-MOLINA, H. : Index structures for information filtering under the vector space model. In : Proc. International Conference on Data Engineering ; pp. 337-47 ; 1994.

YAN, T.W. ; GARCIA-MOLINA, H. : Index structure for selective dissemination of information under the boolean model. In : AC Transactions on Database Systems ; 19(2) ; pp.332-64 ; 1994.

CHAPITRE VI

MACHINA SAPIENS : Infoscan
URL : http ://www. MachinaSapiens.qc.ca/machina/

CHAPITRE VII

RUD, D. S. : A learing Approach to Personalized Information Fitering.
In : Master Thesis, Massacuetts Institue of technology, February 1994.

HUMPHREY, S. ;McELLIGOTT, M. : An Emergent Approach to Information Filtering. In : Computer Science Department. University College ; Cork. Ireland

CHAPITRE VIII

CROFT, W.B. ; DAS, R. : Experiments with query acquisition and use in document retrevial systemes. In Prceding of ACM SIGIR Conference on Reseach and Developement in Information Retrevial ; 1990 ;pp. 349-368.

KILANDER, F. : A Brief Comparison of News Filtering Software.
URL : http ://www.dsv.su.se/~fk

KILANDER, F. : Message Classification and Filtering.
URL : http ://www.dsv.su.se/~fk

HUMPHREY, K. ; McELLIGOTT, S. : An Online News Agent
In : Computer Science Department. University College ; Cork. Ireland.

MALONE, T. ; LAI K.Y. : Experiment with Oval : A radically Tailorable Tool for Cooperative Work. In : ACM Press, New-York, 1992, pp. 289-97.

MARKO, B. ; YAV, S.: Learing Information Retrevial Agents : Experiments with Automated Web Browsing. In : College of Computing. Department of Computer Science, Stanford Unoversity, Stanford, CA 94305.

RADASOA, H.P. : Méthodes d'amélioration de la pertinence des réponses dans un système de base de données textuelles. Thèse université de Paris-Sud ; 1993.

ROGER, D. : Système documentaire intelligent pour la consultation d'un ensemble hétérogène de bases de données documentaies. Mémoire de DEA SIC ; Université Standhal. Grenoble ; 1994.