**République Algérienne Démocratique et Populaire**

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**Université Kasdi Merbah de Ouargla (ALGERIE)**

**Faculté des Sciences et Sciences de l'ingénieur**

**Département de Mathématique et d'Informatique**

# Mémoire

*En vue de l'obtention du diplôme de*

## Magister

*Spécialité :*

**Informatique**

*Option :*

Informatique et Communication Electronique

# Construction et utilisation d'un thésaurus pour la recherche d'information sur le web

*Présenté le* ***27 Mai 2009***

*par*

*Slimane BELLAOUAR*

Devant le jury composé de :

| | | |
|---|---|---|
| *Présidente* | **Fatima-Zohra Laallam** | Université de Ouragla |
| *Examinateurs* | **Lamri Douidi** | Université de Sétif |
| | **Chabane Khentout** | Université de Sétif |
| *Rapporteur* | **Mahieddine Djoudi** | Université de Poitiers |
| *Co-Rapporteur* | **Samir Zidat** | Université de Batna |

# Remerciements

Après avoir remercié Dieu,

Je tiens, tout particulièrement et très sincèrement, à remercier Mr Mahieddine DJOUDI, Maître de Conférences à l'Université de Poitiers, de m'avoir proposé le sujet et de m'avoir encadré. Son suivi, ses encouragements et ses orientations ont été d'un grand réconfort et d'une aide précieuse. Qu'il trouve dans ces mots l'expression de ma profonde gratitude.

Je tien également à remercier Mr Samir Zidat, Maître de conférences à l'Université de Batna pour m'avoir co-encadré et soutenu durant toute la période de ce projet.

Je remercie très vivement Mr Abdelhakim HERROUZ, Chef de département informatique à l'Université de Ouargla, ses efforts et sa disponibilité ont assuré le bon déroulement de nos études de post graduation.

Ma gratitude s'adresse également à l'équipe des enseignants de notre première promotion PG informatique (ICE), à l'Université de Ouargla, pour tout le savoir qu'ils nous ont transmis.

Je remercie aussi très vivement Mr Djelloul ZIADI, Professeur à l'Université de Rouen, de l'intérêt qu'il a apporté à mon travail, de son précieux temps qu'il a investi dans la lecture et la discussion du contenu de ce mémoire.

Je suis très reconnaissant à Mr Karim SOUFFI qui a eu le courage de lire et de relire mon mémoire. Ses conseils sont inestimables.

J'adresse mes remerciements à Madame Fatima-Zohra Laallam, Maître de conférences à l'Université de Ouargla d'avoir accepté la charge de présidente du jury.

Je remercie chaleureusement Mrs Lamri Douidi et Chabane Khentout, Maîtres de conférences à l'Université de Sétif qui ont accepté d'être membres du jury.

Merci également à tous mes collègues de PG informatique (ICE) pour leur gentillesse, compréhension et ambiance.

Mes sincères remerciements vont à mes supérieures et collègues de la Direction de l'Hydraulique de la Wilaya de Ghardaia d'avoir donner toutes les commodités pour le déroulement de mon projet dans les meilleures conditions.

Mes remerciements vont également à tous mes amis pour leurs encouragements, compréhensions et aides.

Enfin, je remercie, de tout mon cœur, tous mes proches pour la confiance, le soutien, la patience et l'aide qu'ils m'ont apporté.

# Résumé

*Depuis son apparition, le web ne cesse de progresser en contenu et en nombre d'utilisateurs. Malgré les améliorations dans la technologie des moteurs de recherche sur le web, des millions d'internautes échouent à satisfaire leurs besoins informationnels. Le recours à l'expansion de requêtes à base de thésaurus semble une solution raisonnable.*

*Ce mémoire présente une méthode de construction automatique d'un thésaurus. Elle s'appuie sur l'analyse des hyperliens des pages web et non sur leurs contenus. Cette méthode permet d'extraire les nouveaux termes et les relations inter termes au fur et à mesure que le web progresse.*

*Le thésaurus construit, est utilisé comme outil d'expansion automatique des requêtes lors du processus d'interrogation de la RI sur le web. Le développement d'un méta moteur de recherche assure l'interaction entre le thésaurus, l'utilisateur et le moteur de recherche Google.*

*En plus, de la représentation sous forme d'une base de données relationnelle du thésaurus construit, le format RDF/XML est utilisé dans une perspective de partager et de réutiliser ses données.*

## Mots clés :

***Systèmes de recherche d'information, thésaurus, expansion de requêtes, hyperliens, web mining, RDF/XML.***

# ABSTRACT:

*Since its emergence, Web continues to grow in content and user number. Despite improvements in search engine technology on the Web, millions of Internet users fail to meet their information needs. Recourse to thesaurus based query expansion seems a reasonable solution.*

*This thesis presents a thesaurus automatic construction method. It is based on web page hyperlink analysis not on their content. This method can extract the new terms and relations between terms as the web progresses.*

*The built thesaurus is used as a tool for automatic query expansion during the interrogation process of web IR. The development of a meta-search engine ensures interaction between the thesaurus, the user and the Google search engine.*

*In addition of relational database representation of the built thesaurus, we have used the RDF/XML format in order to share and reuse its data.*

***KEY WORDS:***

***Information Retrieval systems, thesaurus, query expansion, hyperlink, web mining, RDF/XML.***

# ملخص

إن الشبكة العالمية العنكبوتية لا تزال في نمو مستمر من حيث المحتوى وكذا عدد المستخدمين و ذلك منذ ظهورها. و بالرغم من التحسينات في تكنولوجيا محركات البحث على شبكة الإنترنت، فإن الملايين من مستخدمي الإنترنت يفشلون في تلبية احتياجاتهم من المعلومات. إن اللجوء إلى توسيع الاستفسارات القائم على المكنز يبدو حلا معقولا .

هذه المذكرة تعرض طريقة للبناء التلقائي للمكنز. وهي ترتكز على تحليل الروابط الفائقة لصفحات الويب وليس على مضمونها. باستعمال هذه الطريقة يمكن استخلاص الكلمات الجديدة والعلاقات بينها و ذلك تزامنا مع نمو الويب .

بعد إنشاء المكنز، فإنه يستخدم كأداة للتوسيع التلقائي للاستفسارات و ذلك أثناء إجراء الاستجواب المتعلق بالبحث عن المعلومات في الويب. إن إنشاء ميتا-محرك بحث يكفل التفاعل بين المكنز،المستخدم و محرك البحث *Google*.

من جهة أخرى، و بالإضافة إلى تمثيل المكنز على شكل قاعدة بيانات ترابطية، فإنه تم استعمال الشكل *RDF/XML* وذلك بغية إتاحة تقاسم و إعادة استعمال معطياته.

**مفاتيح:**

نظم استرجاع المعلومات، مكنز، توسيع الاستفسارات، الروابط الفائقة، استكشاف بيانات الويب، ***RDF/XML***

# Table des matières

# Liste des figures

# Liste des tableaux

# Liste des algorithmes

# Introduction

« La recherche doit être ce que l'utilisateur souhaite, pas ce qu'il tape ». Ces mots résument l'objectif final de notre projet. Toutefois, la mise en pratique de ce slogan bute sur plusieurs obstacles pour les systèmes de recherche d'information classiques. Tandis que, pour les SRI sur le web, les défis se multiplient. Ceci est du, principalement, au fait que le web ne cesse de progresser. Les statistiques de janvier 2004 estimaient que le web contienne plus de 10 milliards de pages, dont la caractéristique principale est l'hétérogénéité.

Par ailleurs, le taux de croissance des internautes est très important. Pour l'Algérie, en mars 2007, le taux de croissance était de 3740% par rapport à décembre 2000.

Cette progression rapide du web a soulevé un sérieux problème quant à l'accès à l'information. Ainsi, des méthodes sophistiquées sont toujours requises pour permettre un accès facile à l'information souhaitée.

Notons que la façon la plus courante de trouver des informations sur le web est l'utilisation d'un moteur de recherche. L'utilisateur forme les mots de la requête pour exprimer son besoin informationnel et le moteur de recherche restitue les documents estimés pertinents.

Cependant, souvent, les requêtes n'expriment pas le besoin de l'utilisateur. Une classe plus importante comporte des requêtes ambiguës et/ou vastes.

Par ailleurs les requêtes courtes sont communément utilisées, la longueur moyenne d'une requête est environ 1.66 termes.

Pour pallier aux problèmes sus cités, il suffit d'utiliser la technique d'expansion de requêtes. Elle consiste à reformuler les requêtes en changeant ses mots clés ou en modifiant leur poids dans un but d'obtenir une meilleur correspondance avec les documents pertinents. Elle sert à réduire la distance entre la pertinence utilisateur et la pertinence système.

Un autre problème est la discordance de mots, en effet, l'auteur d'un document, notamment d'une page web, et le lecteur (utilisateur) n'utilisent pas le même vocabulaire. A cela, il faut rajouter la caractéristique d'hétérogénéité des pages web.

Une solution consiste à modéliser le domaine cible de la recherche. L'utilisation d'un thésaurus peut emmagasiner les connaissances d'un domaine donné.

La combinaison des deux solutions pré citées, à savoir, expansion de requêtes à base de thésaurus, donne naissance à l'approche choisie pour satisfaire les besoins des utilisateurs en matière de recherche d'information sur le web.

Ceci dit, la construction manuelle d'un thésaurus dédié à un domaine spécifique fait appel aux compétences d'ingénierie documentaire et de spécialistes du domaine. Il s'agit d'un processus fastidieux et coûteux. Une alternative consiste à se pencher vers l'approche de construction automatique des thésaurus.

Pour se faire, et tenant compte du contexte de notre projet qui est la recherche d'information sur le web, nous avons exploité les hyperliens des pages web qui constituent une caractéristique discriminante entre une page web et un document classique.

En outre, les techniques basées sur l'analyse des hyperliens sont plus fiables que celle basées sur le contenu. Ceci est du au fait que la qualité d'une page web dépend de la qualité des pages web liées à cette dernière, et elle est alors hors du contrôle du concepteur de la page.

Notons que la construction de notre thésaurus ne se base pas sur la totalité de la collection, il s'agit du web. Nous n'avons considéré que la partie pertinente des résultats de la recherche suite à une requête initiale transmise aux moteurs de recherche.

Les grands axes de notre approche pour la construction automatique d'un thésaurus sont :

- Collection des sites web de haute qualité : nous accordons une grande importance à cette étape car nous jugeons que la qualité d'un thésaurus d'un domaine est étroitement liée à la qualité des sites web de départ. Alors, un algorithme de tri local est appliqué.
- Construction de la structure de contenu de la collection : l'idée est de procéder à une ingénierie inverse qui consiste à prendre comme point de départ un site web et déduire, en fin, les intentions du concepteur de site sous forme d'un diagramme de concepts appelé structure de contenu. Cette structure de contenu est l'équivalent d'un document classique.
- Génération du thésaurus : éclairé par la méthode classique de construction automatique des thésaurus, nous appliquons un algorithme similaire aux structures de contenu pour extraire les relations sémantiques entre les concepts.

Une fois le thésaurus construit, son utilisation est traitée selon deux angles de vue. Le premier consiste à l'utiliser comme outil d'expansion de requêtes lors de la recherche d'information. Le second se préoccupe de son parcours et de sa visualisation.

Pour assurer l'interaction entre l'utilisateur, notre thésaurus et les moteurs de recherche (nous ne considérons que Google, dans un premier temps) nous développons un méta moteur de recherche qui présente une interface utilisateur (UI) qui joue le rôle d'un point d'entrée pour la construction du thésaurus puis son utilisation lors de l'expansion de requêtes pendant le processus d'interrogation.

Précisons qu'au stade implémentation de la solution proposée, nous utilisons la plate forme Eclipse RCP (Ritch Client Platform) version 3.3.2 pour le développement en Java des applications clientes.

La suite de ce mémoire s'organise de la manière suivante : le premier chapitre est consacré à la présentation des concepts préliminaires pour la compréhension des notions discutées tout au long de ce mémoire. A travers ce chapitre, nous définissons la recherche d'information, les différentes bases de connaissances utilisées dans la recherche d'information ainsi que le concept du web mining.

Dans le deuxième chapitre, nous nous focalisons sur la recherche d'information, en particulier sur le web. Ensuite, les fonctions d'un système de RI sont mises en évidence. Enfin, nous présentons deux SRI estimés représentatifs. Le premier est SMART, pour les SRI classiques et expérimentales, l'autre est google pour les moteurs de recherche sur le web.

Le troisième chapitre contient un état de l'art sur les approches d'expansion de requêtes ainsi que la construction et l'utilisation des thésaurus. Tout au long du chapitre nous présentons des tests d'évaluation ainsi que leurs résultats, concernant, essentiellement, l'expansion de requêtes par thésaurus sur le web. Ceci nous permet d'explorer les travaux existants dans le domaine et d'en positionner notre problématique.

Le quatrième chapitre est scindé en trois parties :

- La première partie expose l'approche proposée pour construire automatiquement un thésaurus pour la recherche d'information sur le web. Elle s'appuie sur les hyperliens des sites web et non pas sur leurs contenus. La sélection des sites web de haute qualité est réalisée par une combinaison du mécanisme du Pagerank de Google et celui d'un algorithme de tri local, HITS (Hyperlink Induced Topic Search). La structure de contenu d'un site web est générée en appliquant les techniques de web structure mining, dans ce contexte les textes de liens sont utilisés comme résumé sémantique des pages web destination. La génération du thésaurus consiste en une organisation des concepts des structures de contenu. L'organisation des concepts est réalisée par l'application des méthodes issues de la théorie de l'information. Soient l'information mutuelle et l'entropie.
- La deuxième partie est dédiée à l'utilisation du thésaurus construit. Pour être géré et utilisé lors du processus d'expansion de requêtes, nous représentons notre thésaurus sous forme

d’une base de données relationnelle. Par contre, pour assurer le partage et la réutilisabilité des données de notre thésaurus nous utilisons le format RDF/XML reposant sur le vocabulaire SKOS Core. Ceci donne la possibilité d’exporter notre thésaurus vers d’autres gestionnaires de thésaurus tel que Thmanager.

- La troisième partie clôture ce chapitre, elle présente un détail d’implémentation accompagné de valeurs expérimentales.

Finalement, nous concluons par une synthèse de notre travail. Nous citons également quelques perspectives possibles à ce projet.

# Chapitre 1 : Préliminaires

Ce chapitre est consacré aux définitions des concepts qui doivent être pré requis pour la compréhension du reste de ce mémoire.

## 1.1 Donnée, Information et Connaissance

Dans la littérature, il existe des définitions multiples et ambiguës des notions de donnée, information et connaissance. En effet, il n'existe pas de frontières, assez claires, entre ces notions. Une manière de palier ce problème consiste à situer une notion par rapport à l'autre.

### 1.1.1 Donnée

Dans un environnement numérique, « L'unité élémentaire d'information est la donnée qui n'est qu'une chaîne de caractères ou octets constitués de bits (0 ou 1) » (Dherent, 2002). Une définition plus générale est présentée dans (Benayache, 2005) comme suit : « Toute représentation à laquelle une signification peut être Attachée ». Une donnée peut être qualitative ou quantitative mais elle n'a pas de sens en elle-même. Les données peuvent être récupérées, représentées et réinterprétées.

### 1.1.2 Information

Une information est issue lorsqu'on donne un sens à une donnée. L'information est donc une collection de données organisées pour donner forme à un message le plus souvent sous forme visible, imagée, écrite ou orale, de telle sorte à réduire une incertitude et transmettre quelque chose qui déclenche une action (Benayache, 2005).

### 1.1.3 Connaissance

Debenham a défini la connaissance comme l'ensemble des associations fonctionnelles explicites entre les éléments de l'information et/ou des données (Kendal et Creen, 2006).

Le grand dictionnaire terminologique[1] définit le terme connaissance dans le domaine Informatique comme l' « ensemble de faits, événements, règles d'inférence et heuristiques permettant à un programme de fonctionner intelligemment ».

Pour mieux comprendre la relation entre donnée, information et connaissance, la figure 1. représente ces notions sous forme pyramidale illustrée par un exemple.

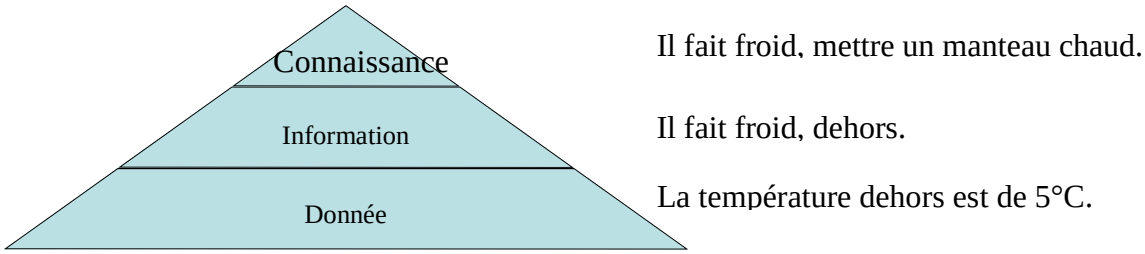



**Figure 1. : Donnée, Information et Connaissance**

[1]Ouvrage en ligne rassemblant un fonds terminologique de 3 millions de termes français, anglais et latin dans 200 domaines d'activité. www.granddictionnaire.com/

## 1.2 Le Document

Il existe plusieurs définitions de la notion de document, ceci est du aux différentes mutations des auteurs. Dans notre contexte, nous commençons par une définition très généraliste, puis nous introduisons une définition technique, en fin la notion de document numérique sera présentée.

Une définition généraliste et ancienne stipule qu'un document est l'expression d'une pensée humaine. Constatez que l'aspect connaissance est important dans cette vision.

Une définition technique est développée par l'institut international de la coopération intellectuelle[1] : « Un document représente toute base de connaissance fixée matériellement, susceptible d'être utilisée pour la consultation, l'étude ou la preuve, comme par exemple : manuscrit, imprimé, représentation graphique ou figurée, objet de collections etc. »

En ce qui concerne la définition du document numérique, nous avons repris celle de Gwendal AUFFRET (El-hachani, 2005) : un document, entité discrète provenant d'une activité éditoriale donnée. Le document est composé des éléments suivants :

- Le support d'enregistrement : papier, disque magnétique, cassette,…
- La forme d'enregistrement : papier, codage ascii, codage vidéo,…
- Le support de restitution : papier, écran d'ordinateur, écran de télévision, haut-parleurs,…
- La forme physique de restitution : encre sur papier, signal audio ou vidéo,…
- La forme sémiotique de restitution : une représentation qui respecte une certaine structure ou forme selon laquelle elle soit intelligible par le lecteur (utilisateur). Ecriture, sons, images animées,…

## 1.3 La Recherche d'Information

### 1.3.1 Définition

D'après l'AFNOR[2], la recherche d'information (RI) est définie comme « Actions, méthodes et procédures ayant pour objet d'extraire d'un ensemble de documents les informations voulues ». Dans un sens plus large, toute opération (ou ensemble d'opérations) ayant pour objet la recherche, la collecte et l'exploitation d'informations en réponse à une question sur un sujet précis.

### 1.3.2 Performances des systèmes de RI

La qualité d'un système de RI réside dans la pertinence des documents retournés pour l'utilisateur. Cette notion de pertinence est très complexe à évaluer car elle dépend fortement de l'utilisateur qui est le seul capable de juger si les documents retournés satisferaient son besoin informationnel.

Cependant, il est indispensable de disposer de mesures quantitatives pour évaluer les performances des SRI : (Figure 2.)

- $Rappel = \frac{C}{B+C}$, le taux des documents pertinents retrouvés par le système par rapport à l'ensemble de documents pertinents dans la collection.

[1] Une agence de la ligue de nations, travaillant en collaboration avec l'union française des organismes de documentation.

[2] AFNOR : association française de normalisation. www.afnor.org/

- $\text{Pr}écision = \frac{C}{A+C}$, le taux des documents pertinents retrouvés par le système par rapport à l'ensemble de tous les documents retrouvés par le système.
- $Silence = \frac{B}{B+C}$, représente l'ensemble des documents pertinents que l'interrogation n'a pas pu retrouver.
- $Bruit = \frac{A}{A+C}$, représente l'ensemble des documents non pertinents (selon l'utilisateur) que l'interrogation a restitué.

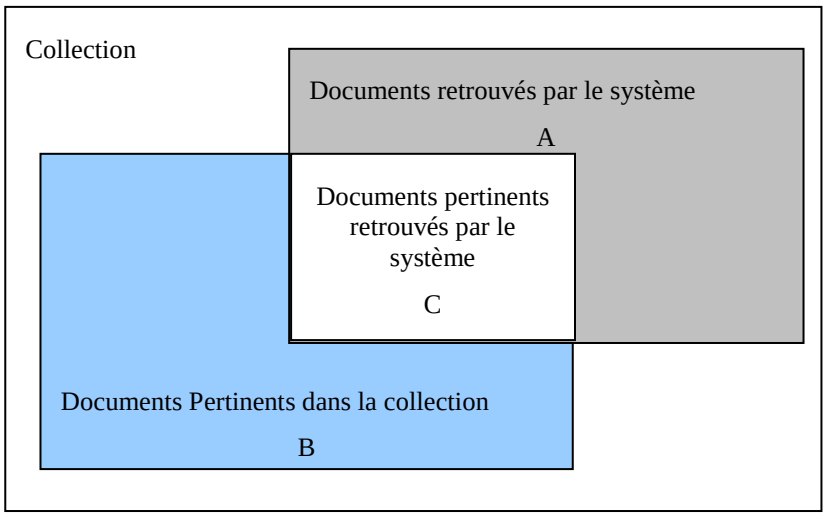



**Figure 2. : Mesures de performance dans la RI**

## 1.4 Les Bases de Connaissances dans la RI

Les bases de connaissance peuvent être utilisées comme tentative de combler le fossé sémantique entre le besoin informationnel de l'utilisateur et la requête de recherche, ceci en les insérant comme médiateur.

Dans le contexte de la RI, les bases de connaissance modélisent le domaine de recherche. En effet ils sont une abstraction de la réalité. Ils servent à améliorer les résultats de la recherche par la désambiguïsation des concepts du domaine et la fourniture d'une structure organisationnel (David et al., 2001).

Dans ce qui suit, nous nous intéressons à trois types de bases de connaissances, parmi les plus répandues, à savoir, taxonomie, thésaurus et ontologie (Schawarkz, 2005).

### 1.4.1 Taxonomie

Une taxonomie est une structure de domaine hiérarchique. Les relations relient les concepts généraux aux concepts spécifiques, elles sont transitives.

Notons qu'il n'existe pas une approche standardisée pour modéliser un domaine par une taxonomie.

A titre d'illustration, on peut citer deux grands exemples, Yahoo! Directory et DMOZ (Directory Mozilla).

L'application initiale des taxonomies était la structuration du monde naturel qui nous entoure (plantes, animaux,…). Dans un environnement d'entreprise, il existe deux applications répandues pour les taxonomies. La première consiste à les utiliser en tant que vocabulaire structuré pour

classifier les ressources et faciliter, ainsi, le processus de leur repérage. La seconde application est la structure de navigation visuelle dans une interface utilisateur.

Une taxonomie est un modèle pré coordonné, car les termes composés sont définis lors de la construction de la structure.

### 1.4.2 Thésaurus

La norme ANSI/NISO du thésaurus[1] monolingue définit un thésaurus comme « un vocabulaire contrôlé disposés dans un ordre connu et structuré de telle sorte que les relations d'équivalence, homographique, hiérarchique et associative entre les termes sont clairement montrées et identifiées par des indicateurs relationnels normalisés… Les principaux objectifs d'un thésaurus visent à faciliter la recherche de documents et à atteindre la cohérence dans l'indexation des écrits ou des documents enregistrés autrement et d'autres items. »

En général, trois types de relations inter termes sont utilisés, l'équivalence, la hiérarchie et l'association.

Les standards concernant les thésaurus fournissent des instructions et des recommandations détaillées en matière de modélisation, parmi lesquelles :

- La spécificité : étant donné que les termes les plus spécifiques auront plus de chances de correspondre aux termes de la requête de recherche, alors cette caractéristique de spécificité se montre importante pour améliorer la performance de la RI en utilisant un thésaurus.
- Choix des termes : la relation d'équivalence résout le problème de surcharge du vocabulaire d'un thésaurus. En effet, parmi plusieurs synonymes, on en choisit un (le plus représentatif possible) que l'on introduit comme descripteur.

Au contraire des taxonomies les thésaurus présentent une structure poly hiérarchique. Cela se produit quand un terme possède plus de termes génériques (broader terms).

Les thésaurus sont principalement des modèles post coordonnés car, dans un thésaurus toutes les relations d'équivalence, hiérarchique et d'association d'un terme donné sont spécifiés dans le terme lui-même. Quand un thésaurus est utilisé pour l'indexation ou pour la recherche, n'importe quelle combinaison de termes, avec le descripteur concerné, peut être appliquée.

### 1.4.3 Ontologie

La notion d'ontologie est à l'origine une branche de la métaphysique qui s'intéresse à l'étude de l'être en tant que être.

Studer et al. présentent une définition la plus couramment citée dans le domaine de représentation de connaissance, traduite en français comme suit : « une ontologie est une spécification formelle et explicite d'une conceptualisation partagée. Une conceptualisation fait référence à un modèle abstrait d'un certain phénomène dans le monde en identifiant les concepts pertinents de ce phénomène. Explicite signifie que les types de concepts utilisés et les contraintes pesant sur leur utilisation sont explicitement définis. Par exemple, dans les domaines médicaux, les concepts sont les maladies et les symptômes, les relations entre eux sont la causalité et une contrainte imposant qu'une maladie ne peut pas causer elle-même. Formelle fait référence au fait qu'une ontologie doit être lisible par la machine, ce qui exclue le langage naturel. Partagée renvoi à la notion selon laquelle une ontologie capture des connaissances consensuelles, c'est-à-dire, non privées à un certain individu, mais acceptées par un groupe » (Studer et al., 1998).

[1] Est un mot latin signifiant recueil ou répertoire, son pluriel devrait être « thésauri », terme utilisé lors des premières utilisations. Mais l'usage semble avoir finalement rendu ce terme invariable au singulier.

Une des principales différences avec les deux schémas suscités est le besoin d'une spécification formelle. Ceci rend les ontologies, particulièrement, mieux adaptées pour l'utilisation dans le web sémantique.

Les éléments principaux modélisés dans une ontologie sont les concepts, les relations entre concepts et les propriétés de ces concepts ainsi que les valeurs associées.

Le schéma RDF (Resource Description Framework) est le langage de représentation des ontologies le plus simple. Le langage OWL (Web Ontology Language) est développé pour donner plus d'expressivité pour la modélisation du domaine. Un autre langage de représentation des ontologies est le standard ISO Topic Maps.

Les ontologies permettent une structure poly hiérarchique ce qui implique plus de flexibilité mais aussi plus de complexité.

Un domaine d'application des ontologies est la RI intelligente car une des caractéristiques des ontologies est l'inférence automatique.

### 1.4.4 Comparaison de taxonomie, thésaurus et ontologie

Le tableau 1. résume les différences entre les trois types de base de connaissances étudiés dans les sections précédentes en matière de standardisation, modélisation, application et outils de création.

| | **Taxonomie** | **Thésaurus** | **Ontologie** |
|---|---|---|---|
| **Origine** | Sciences de la nature (biologie, chimie,…) | Sciences bibliothécaires | Métaphysique, Intelligence artificielle, ingénierie de connaissance |
| **Standard** | Aucun | ISO 5964, ISO 2788, BS 5723, BS 6723, ANSI NISO Z39-19, Z47-100, Z47-101, SKOS, أسمو 578، أسمو 795، | recommandations W3C (RDF, OWL) |
| **Construction du modele** | | | |
| **Relations** | Hiérarchiques | Hiérarchiques non typées, associatives et équivalences | Hiérarchiques typées, associatives |
| **Propriétés** | Aucune | Si nécessaire, elles peuvent être décrites sous forme de notes d'application (scope note) | Dans le schéma RDF, il existe des propriétés de relations ainsi que les propriétés de restriction (domaine, intervalle) |
| **Application** | Classification, navigation, recherche. | Classification, navigation, recherche. | Classification, navigation, recherche, visualisation, raisonnement automatique. |
| **Outils de création** | MindManager | MultiTES | Protégé |

**Tableau 1. : Différences entre taxonomie, thésaurus et ontologie**

## 1.5 Le Web mining

### 1.5.1 Définition

Deux approches différentes ont été considérées dans la définition initiale du web mining. La première approche est une vue centrée processus qui définit le web mining comme une séquence de taches. La seconde est une vue centrée données qui définit le web mining en s'appuyant sur les types de données du web utilisées dans le processus d'exploration.

La seconde approche est devenue plus acceptable. Dans ce contexte Zdravko et Daniel définissent le web mining comme suit : « Par web mining, nous nous référons à l'application des méthodes, techniques et modèles du data mining, à la variété de formes de données, structures et types d'usage contenus dans le World Wide Web » (Zdravko et Daniel, 2007).

Une définition du data mining est présentée dans (David et al., 2001) : « Data mining est l'analyse des ensembles de données d'observation (souvent volumineuses) pour trouver des relations implicites et pour résumer les données de différentes manières qui sont à la fois compréhensibles et utiles au propriétaire de données ».

### 1.5.2 Taxonomie du web mining

En général, le web mining peut être devisé en trois catégories distinctes selon le type de données à explorer. La figure 3. illustre une telle taxonomie (Srivastava et al., 2005).

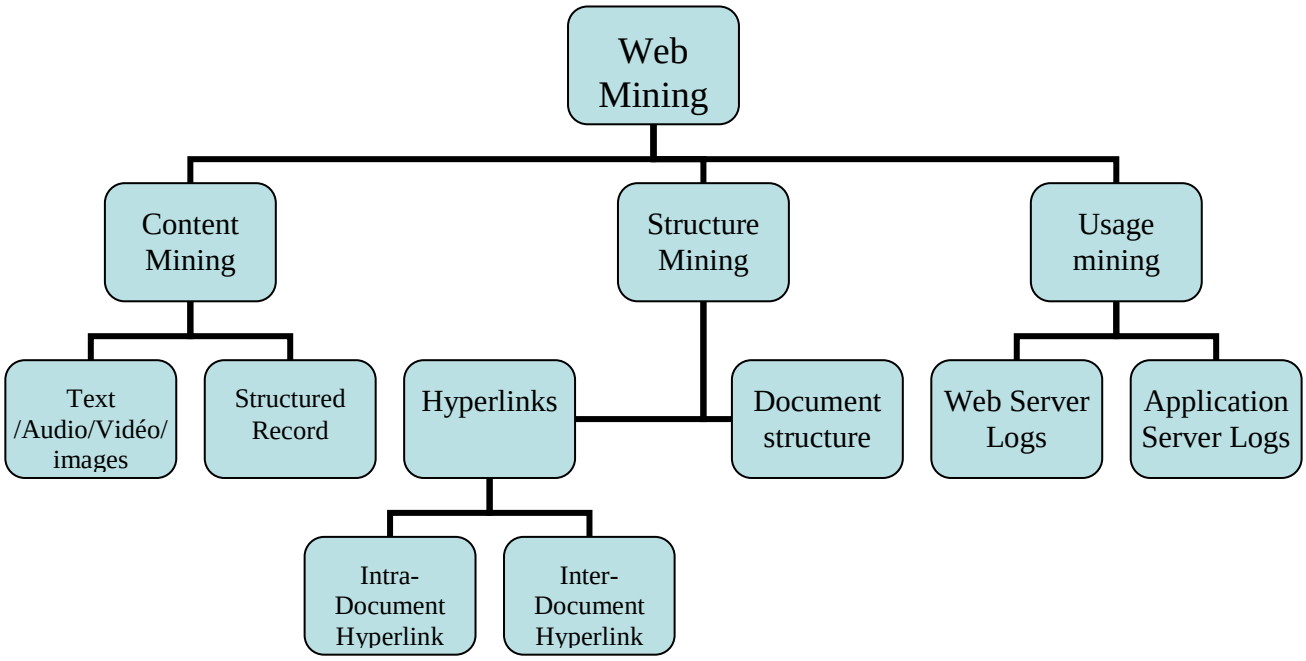



**Figure 3. : Taxonomie du Web mining**

- Web content mining : est le processus d'extraction de l'information utile à partir du contenu des documents du web. Les données de contenu peuvent être texte, images, audio, vidéo ou bien des données structurées telles que les listes et les tables. L'application du texte mining au contenu du web est la plus largement étudiée. Les axes étudiés sont la découverte thématique (topic discovery), extraction de modèles d'association, clusterisation et classification.

  Par ailleurs, il existe un travail significatif, pour l'extraction des connaissances à partir des images, dans le domaine de traitement d'images et de vision par ordinateur. L'application de ces techniques dans le web content mining est limité.

- Web structure mining : la structure d'un graphe du web consiste en pages web en tant que nœuds et les hyperliens en tant que arcs reliant les différentes pages. Le web structure mining est le processus de découverte de l'information de la structure à partir du web. Le web structure mining peut être scindé en deux classes selon la structure utilisée :
  - Hyperlien : une unité structurelle qui relie un emplacement dans une page web à d'autres emplacements dans la même page (hyperlien intra-document) ou dans une autre (hyperlien inter-document). Il existe plusieurs travaux sur l'analyse des hyperliens.
  - Structure de document : un page web peut être aussi structuré sous forme arborescente en se basant sur les balises HTML et XML. Les efforts d'exploration ont été focalisés sur l'extraction automatique d'un modèle objet de document (DOM, Document Object Model).
- Web usage mining : est l'application des techniques du data mining pour la découverte des modèles d'usage à partir des données de web, dans une perspective de comprendre et mieux servir les besoins des applications à base de web. Les données d'usage capturent l'identité des utilisateurs web ainsi que leurs comportements de navigation. Le web usage mining peut être classifié selon le type des données d'usage :
  - Les données d'un serveur web.
  - Les données d'un serveur d'application.
  - Les données au niveau des applications.

  Les données d'usage peuvent aussi être divisées en trois types selon la source de leur collecte :
  - Dans le coté serveur : la collection de données décrit l'usage d'un service par tous les utilisateurs.
  - Dans le coté client : la collection de données décrit l'usage complet de tous les services par un client particulier.
  - Dans le coté proxy : la collection de données décrit l'usage d'une situation intermédiaire.

# Chapitre 2 : La Recherche d'Information

## 2.1 - Recherche d'Information : Tour d'Horizon

### 2.1.1 - Naissance :

L'être humain est doté de divers types de besoins qui seraient à l'origine de nombreux comportements permettant leur satisfaction[1], parmi lesquels, le besoin informationnel qui portrait l'individu à s'engager dans une activité de recherche d'informations (Simonnot, 2006). Il en ressort que la recherche d'information est une activité constamment évolutive, c'est pourquoi, dans cette section, nous nous limitons à citer quelques événements estimés marquants dans l'historique de la RI[2].

Le premier événement concerne la naissance de l'écriture, qui remontera à 3000 av. J-C. (Singhal, 2001). Ce sont les Sumériens qui inventèrent l'écriture cunéiforme[3]. Ils ont réalisé la nécessité de l'organisation et l'accès à l'information. Ils utilisaient des tablettes d'argiles pour inciser des signes et des dessins.

L'invention du papier au III$^{ème}$ siècle av. J-C. par les Chinois[4] a participé fortement dans la représentation, le stockage, l'organisation et l'accès à l'information. Le papier est le grand média de masse.

L'apport des arabes est concrétisé au VIII$^{ème}$ siècle, d'une part par l'introduction du coton comme matière première fibreuse pour améliorer la blancheur, et d'autre part, plus tard, par l'exportation de l'art de l'industrie du papier vers l'occident[5] (Coste, 2004).

Un autre événement qui peut s'inscrire, dans notre contexte, c'est bien l'invention de la typographie, vers 1440, basée sur le principe des caractères mobiles par Gutenberg[6]. Cette invention a dopé l'utilisation du papier et donc la production informationnelle.

En 1945, Vannevar Bush a exposé dans son article "As We May Think" le principe de sa machine MEMEX, MEMory EXtension. MEMEX est conçue pour faciliter la gestion et l'indexation des quantités d'informations de toutes sortes (livres, notes personnelles, idées,…). Il reste à noter que MEMEX n'a jamais été réalisé puisqu'il butait sur des obstacles technologiques.

Dans le même ordre d'idées, en 1946, avec l'invention du premier ordinateur totalement programmable, les gens ont découvert la possibilité de son utilisation dans la sauvegarde et la recherche automatique dans une grande masse d'information.

C'est en 1948, que Calvin Mooers[7] a forgé le terme "Information Retrieval" (Manning et al., 2007) pour la première fois dans son mémoire de maîtrise au MIT. A partir de ce moment, le domaine de RI commença à se distinguer et plusieurs conférences et groupe ont eu lieu. C'est ce qu'on va traiter dans la section suivante.

---

[1] Le site http://www.psychobiology.org contient d'amples informations.

[2] Pour plus de detail, voir "History of Information Retrieval Systems and Increase of Information over Time" dans http://online.sfsu.edu/~fielden/hist.htm

[3] Le Sumérien est la plus ancienne langue écrite connue ente le Tigre et l'Euphrate.

[4] http://cerig.efpg.inpg.fr (dans la rubrique Dossiers).

[5] A Xavia (San Felipe, Espane) en 1506, à Sicile en 1102, à Fabriano (Italie) en 1276 et en France en début du XIV$^{ème}$ siècle.

[6] 1400-1468: un imprimeur Allemand, le premier à avoir utilisé les caractères mobiles.

[7] 1914-1994 : informaticien américain connu pour ses travaux dans le domaine de la RI et le développement du langage de programmation TRAC.

### 2.1.2 - Recherche d'Information Classique

#### *2.1.2.1 Premières manifestations*

Plusieurs travaux inhérents au domaine de la recherche d'information sont apparus au milieu des années 50, dont l'intérêt commun est la recherche du texte par le biais d'un ordinateur, notamment dans les bibliothèques.

L'un des premiers travaux notable est bien la proposition de H.P. LUHN[1] en 1957, qui stipule qu'un système RI se base sur une représentation des documents (et des requêtes) obtenue d'une façon automatique à partir des contenus de ces documents (Piwowarski, 2003). Cette proposition est basée sur une approche statistique qui utilise la fréquence des données pour l'extraction des mots et des phrases dans une perspective d'indexation automatique des documents (Luhn, 1957).

Dès lors, le domaine de RI a connu plusieurs développements :

Dans la période 1961-1965, G. SALTON[2], en collaboration avec ses étudiants, a développé le système SMART (une description de SMART sera exposée dans la section 2.3.1).

Dans la période 1957-1967, Cyril CLEVERDON a dirigé le projet Cranfield, dans le collège d'Aéronautiques (U.K.). L'évaluation Cranfield I consiste en une collection de test et est constituée d'un ensemble de 18000 articles et d'un ensemble de 1200 requêtes (Nie, 2007). Les requêtes étaient d'abord évaluées par des experts afin de déterminer les réponses pertinentes. Ensuite, les résultats d'une recherche automatique étaient comparés avec les réponses pertinentes pour mesurer la performance en termes de précision et de rappel.

Au constat du système SMART et de l'évaluation de Cranfield, il s'avère que la communauté de RI, a instauré, dès les premiers jours, une tradition d'expérimentation et d'évaluation pour tester n'importe quelle méthode d'indexation et de recherche de documents afin de connaître son effet en réalité.

Les années 70 et 80 ont été témoins de plusieurs développements en matière de modèles et techniques en RI. Les modèles booléen, vectoriel et probabiliste constituent les modèles de base pour la RI classique. Ces modèles ainsi que leurs dérivés seront exposés en détail dans la section 2.2.4.3.

Toujours dans une optique d'évaluation et d'expérimentation, des conférences de RI annuelles se sont organisées (Langville et Meyer, 2006). SIGIR (Special Interest Group on Information Retrieval), TREC (Text REtrieval Information), CIR (Context-based Information Retrieval), en sont des exemples. Elles se sont utilisées pour comparer différents modèles propriétaires des moteurs de recherche afin d'aider le domaine à progresser vers des moteurs de recherche meilleurs et efficaces, notamment pour des collections de données plus larges (par exemple la collection de test TREC primaire, en 1992, contient autour de 2 giga octets de textes, soit entre 500.000 et 1.000.000 de documents) (Voorhees, 2003).

Du point de vue contenu, ces conférences présentent des collections de tests comportant trois parties, un ensemble de documents, un ensemble de besoins informationnels (appelés topics dans TREC) et un ensemble de jugements de pertinence indiquant les documents répondant au mieux à un besoin informationnel donné. (Voorhees, 2003) donne plus de détail sur ces notions.

[1] 1896-1964 : il a enregistré 80 inventions et il a rejoint IBM en 1941.

[2] 1927-1995 : Allemand, professeur informaticien à l'université Cornelle, Leader du domaine de RI

### *2.1.2.2 Caractéristiques des SRI classiques*

Dans un but de mettre en évidence la RI sur le web qu'on va développer plus tard et pour en faire la distinction avec la RI classique nous avons jugé nécessaire de citer les caractéristiques principales de cette dernière.

La RI classique s'occupe de petites collections de documents, limitées à quelques centaines de millier de documents, collectés, contrôlés et maîtrisés par des spécialistes.

Ces collections de documents sont généralement statiques. Le mot statique peut porter deux sens : Le premier vise les documents, une fois un document rajouté à la collection, il demeure inchangé (un livre dans une étagère). Le second vise la collection, elle est relativement statique comparée aux pages web.

Les documents de ces collections sont conservés physiquement sous forme de livres, de journaux ou électroniquement sous forme de microfiches[1] et CDs (Langville et Meyer, 2006).

Il est à noter que le processus de recherche, dans un tel type de collections de documents, est, actuellement, pratiquement, informatisé. La recherche dans une collection de livres d'une bibliothèque universitaire, ou la recherche dans une réserve de slides d'un professeur pour un thème donné font des exemples de la RI classique. Le mécanisme d'automatisation fait référence aux moteurs de recherche.

Ceci dit, deux facteurs sont totalement ignorés par la RI classique, à savoir, le contexte et l'usager. Plusieurs conférences ont été organisées, récemment, autour de ces sujets. SIGIR 2004 IRix workshop et SIGIR IRix workshop ont débattu l'idée d'un processus de recherche qui dépend de plusieurs facteurs : le temps, l'espace, l'historique de l'interaction, etc. (Ingwersen et Jarvelin, 2005a).

### *2.1.2.3 La RI et la communauté francophone*

Pour la communauté francophone, le domaine de RI a commencé dans les années 80. RIAO (Recherche d'Information Assistée par Ordinateur) est la première conférence dédiée à la recherche d'information. Elle a eu lieu à Grenoble (France) en 1985 (Nie, 2007).

L'équipe MRIM[2] (Modélisation et Recherche d'Information Multimédia) a développé le prototype IOTA en 1980. IOTA est un SRI permettant de construire automatiquement un thésaurus à partir des textes pour ensuite rendre les résultats (Haddad, 2002).

Dans les années 90, les équipes de recherche en RI se sont répondues à travers toute l'Europe, témoignant l'importance d'un tel domaine.

Toujours dans une perspective d'évaluation et d'expérimentation, et en suivant la trace du TREC, l'INIST (Institut de l'Information Scientifique et Technique) de Nancy a lancé le projet Amaryllis I (1996-1997) puis Amaryllis II (1998-1999) dont l'objectif est de promouvoir le domaine de RI pour la langue française.

En 2002, Amaryllis a participé aux expérimentations CLEF[3] (Cross Language Evaluation Forum) (Haddade, 2002 ; Nie, 2007). Elle comportait approximativement 150.000 documents bibliographiques.

[1] Microfiche (microform en anglais) : support de stockage analogique contenant environ 100 à 130 pages, elle fut inventé en 1961 par Carl O. Carlson.

[2] Le site http://apmd.prism.uvsq.fr/partenaires.html comporte une description détaillée des travaux de l'équipe MRIM.

[3] Voir le site http://clef.iei.pi.cnr.it

### 2.1.3 - Recherche d'Information sur le Web

#### *2.1.3.1 Emergence de La RI sur le Web*

En 1989, l'informaticien du CERN (Conseil Européen pour la Recherche Nucléaire), Tim Berners-Lee[1] proposa un système de gestion décentralisé de l'information destiné à la communauté des physiciens des hautes énergies (Langville et Meyer, 2006). A la fin de 1990, et en collaboration avec plusieurs scientifiques du monde entier, l'idée de Berners-Lee était devenue le World Wide Web, ou tout simplement le Web, appelée la toile en français.

Cet événement est caractérisé par la publication décentralisée du contenu, avec essentiellement, l'absence d'un contrôle centralisé (Manning et al., 2007). Cette caractéristique a influencé considérablement le processus de recherche d'information donnant naissance à une nouvelle branche dans la RI, la recherche d'information sur le web qui est devenue parmi les activités les plus dominantes dans le web.

La RI sur le web utilise des techniques de la RI classique, mais avec de nouveaux défis. C'est ce qu'on va développer dans la section suivante.

#### *2.1.3.2 Les Défis de la RI sur le Web*

##### *2.1.3.2.1 Le déluge informationnel*

Le phénomène du déluge informationnel est exprimé dans (Serres, 2004) par le passage de la métaphore de "l'explosion documentaire" des années 60 à celle du "déluge informationnel" d'Internet. Les statistiques de janvier 2004 (Langville et Meyer, 2006) estimaient que le web contienne plus de 10 milliards de pages, avec une taille moyenne de page de 500 kilo octets. Or la population mondiale était de l'ordre de 6,4 milliards, soit presque deux pages pour chaque habitant.

Le web est la collection de documents la plus large dans le monde. A titre indicatif, le projet d'information de Berkeley "How Mutch Information"[2], en 2003, estimait que la quantité d'information du web est 20 fois plus la taille de la collection entière de la bibliothèque du congress.

De plus, l'étude netcraft[3] de Mai 2007, recense 118.023.263 sites, soit une augmentation de 4,4 millions sites par rapport au mois d'Avril 2007.

Par ailleurs, une compagnie nommée Bright Planet[4] estimait, en 2003, que le web invisible[5] est de 400 à 550 fois plus que le web normal, soit entre 66.000 Téra octets et 92.000 Téra octets. Ceci en se basant sur les statistiques qui évaluaient la taille du web normal à 167 Téra octets.

Le défis qui s'impose, vis-à-vis ce déluge informationnel, se résume principalement dans la couverture entière du web. Une étude réalisée par LAWRENCE et GILES (Lawrence et Giles, 1998) affèrant à la couverture des pages web et le recouvrement entre six moteurs de recherche a tiré plusieurs constations surprenantes, mais importantes, parmi lesquelles :

- La couverture d'un moteur de recherche individuel est sensiblement limitée, pas plus d'indexation d'un tiers du web.

[1] Né à Londres en 1955. En 1980, il était consultant à CERN. En 1994, il fonda le W3C au MIT qui le préside actuellement.

[2] http://www2.sims.berkeley.edu/research/projects/how-mutch-info-2003/

[3] http://news.netcraft.com/archives/web-server-survey.html

[4] http://www.brightplanet.com

[5] Web invisible = web profond = deep web = hidden web : fait référence au contenu du web qui ne fait pas partie du "surface web" indexé par des moteurs de recherche.

- La combinaison de résultats de plusieurs moteurs de recherche peut améliorer considérablement la couverture : les six moteurs, testés collectivement, ont couvert 60% du web.

#### *2.1.3.2.2 Le web est dynamique*

Dans la section 2.1.2.2, nous avons discuté l'aspect statique caractérisant la RI classique. La dynamicité du web peut être résumée dans les points suivants (Baeza et al., 2004 ; Yiping et al., 2006) :

- La création : de nouvelles pages sont crées d'une façon continue et sont alors ajoutées à des sites web. Le rôle des moteurs de recherche est de capturer les nouvelles informations induites de cette création et de les rendre visibles d'une manière opportune.
- Les mises à jour : les mises à jour font référence au changement du contenu de pages. Ces changements peuvent être sérieux. Le défi des moteurs de recherche est de décider sur la fréquence de rafraîchissement de leurs références de pages.
- La suppression : une page web devient inexistante si elle est supprimée, ou si tous les liens menant à cette page sont supprimés. Les suppressions non détectées sont plus préjudiciables à la réputation des moteurs de recherche que les mises à jour non détectées.

Une étude mentionnée dans (Langville et Meyer, 2004) signale que toutes les pages web dans leur collection de test changent en moins au bout d'une semaine et 23% des pages .com changent quotidiennement.

Une autre étude plus large et plus récente réalisée par FETTERLY et al. (Fetterly et al., 2003) met en jeu 151 millions pages web. Ces pages sont crawlées une fois par semaine pendant onze semaines. Un certain nombre de conclusions de cette étude est :

- 88% des pages demeurent disponibles pendant le crawl final, ça implique que la suppression des pages dans une semaine est faible.
- 65,20% de pages demeurent inchangées. Pour les autres pages, les modifications sont localisées habituellement au niveau des balises HTML. Les autres changements sont triviaux.
- Les pages web de grande taille changent plus fréquemment que celles de petite taille.

#### *2.1.3.2.3 Le web est auto organisé*

Le contenu de cette section est largement inspiré de (Langville et Meyer, 2006). Contrairement aux collections de documents classiques, qui sont collectées et classifiées par des spécialistes, sur le web, n'importe quelle personne peut poster une page web et faire des liens à volonté.

Souligner qu'il n'existe pas de standards ni de politique concernant le contenu, la structure et le format. C'est pourquoi les données sont hétérogènes, elles existent en plusieurs formats, langages et alphabets.

En outre, et à titre illustratif, cette caractéristique d'auto organisation encourage les spammeurs de tirer profit du potentiel commercial offert par le web. Bien que le nom spammer est donner originalement à celui qui envoi massivement des emails de publicité, avec la recherche sur le web et la distribution en ligne, ce nom est élargi pour inclurent ceux qui utilisent des techniques de création de pages trompeuses afin d'être bien classées parmi les résultats d'un moteur de recherche suite à une requête particulière (Gong et al., 2005).

D'un autre angle de vue, auto organisé signifie aussi que les pages web sont crées pour différentes raisons. Quelques pages web visent des surfeurs qui sont entrain de faire du shopping, d’autres sont

dédiées pour la recherche. En effet, un moteur de recherche doit être capable de satisfaire plusieurs types de requêtes, quelles soient transactionnelles, de navigation ou informationnelles.

Tous ces aspects précités se fusionnent pour fabriquer un défi complexe qui rend la tâche d'un moteur de recherche délicate.

#### *2.1.3.2.4 Les hyperliens*

Un hyperlien ou lien hypertexte ou simplement un lien est une référence ou un élément de navigation qui permet d'associer plusieurs nœuds entre eux dans un hypertexte.

Cette caractéristique du web découle des fondations de la machine MEMEX de Vannevar BUSH et constitue le pilier centre des moteurs de recherche sur le web.

Parmi les avantages principaux des hyperliens, on peut citer la possibilité d'exploration de la structure du web tentaculaire. Cependant plusieurs inconvénients en résultent. Le plus intéressant est celui des spammeurs qui abusent toujours de cette caractéristique et cherchent à augmenter le trafic sur leurs sites en biaisant les algorithmes d'estimation de pertinence des moteurs de recherche.

Ceci dit, un autre défi doit être relevé par les moteurs de recherche. Il consiste à détecter les spams dans un but d'améliorer la qualité de leurs résultats.

#### *2.1.3.2.5 La duplication du contenu*

La duplication du contenu est la reproduction d'un contenu d'une page sur une autre, les pages concernées sont celles présentant des URL différentes et un contenu similaire ou trop approchant[1]. En effet plus de 48% des pages ont des copies (Cho et al., 2000).

Les moteurs de recherche essaient d'éviter le crawl et l'indexation des pages web dupliquées à l'identique ou presque[2], car le phénomène de duplication engendre une surinformation qui encombre les résultats de tels moteurs. Google et Yahoo sont des précurseurs dans la détection de la duplication du contenu[3].

Ce phénomène est bien étudié, BRIN et al. ont proposé un système de détection de copies complètes ou partielles pour un système de bibliothèque numérique (Brin et al., 1995).

Dans le contexte du web, BRODER a étudié le même problème, et proposa une solution mathématique basée sur les concepts de ressemblance et du "contanement" (Broder, 1997).

Il est à noter que la détection d'une page déjà crawlée est un travail du hasard (Henzinger et al., 2003). Cependant, si on se limite à la détection des hôtes dupliqués, le problème deviendra abordable. C'est sur cette constatation que l'étude (Cho et al., 2000) était menée. Elle s'intéresse à l'identification des réplications des documents et des collections.

Actuellement, la syndication du contenu entre, également, dans la catégorie de duplication du contenu, obligeant les détenteurs de sites à surveiller et protéger leurs différents contenus.

### *2.1.3.3 La RI et la Communauté Arabe*

On assiste aujourd'hui à un développement rapide d'une société globale d'information. Les usagers de l'Internet veulent se débarrasser des frontières géographiques et spatiales et retrouver l'information pertinente où quelle soit et quel qu'en soit la langue. Le témoin de cette société globale

[1] www.journaldunet.com

[2] Duplicated or near duplicated pages

[3] Le brevet sur les calculs de similarité proposé par google à la fin de l'année 2001 vient d'être breveté (28-01-2007) par l'USPTO (United States Patent and Trandmark Office)

d'information est l'explosion dominant des ressources multilingues qui a conduit au développement des techniques de RI multilingues et de croisement linguistique (translinguistique).

L'arabe est l'une des six langues officielles des nations unis, c'est la langue de plus de 300 millions de personnes.

Les statistiques ont montré une croissance exponentielle des sites web arabes (Abdelali et al., 2004) ; depuis 1995, l'année de l'apparition du premier journal en ligne[1], le nombre de sites arabes a atteint 20.000 sites pour l'année 2000, soit 7% des sites publiés dans le web.

En ce qui concerne les internautes parlant l'arabe, le nombre est estimé à 4,4 millions, soit 1,5% de la population arabe (Abdelali et al., 2004).

Les chiffres actuels[2] concernant l'usage des dix premières langues dans l'Internet, mises à jour le 10 mars 2007, confirment la même idée de la croissance de l'usage arabe de l'Internet. Ils révèlent que le nombre des internautes arabes est de 28.540.700, soit 2,6% du nombre de tous les internautes avec un taux de croissance entre 2000 et 2007 de 931,8%.

Pour l'Algérie, avec une population de 33.506.567, le nombre des internautes en décembre 2000 était de 50.000, alors qu'en mars 2007 le nombre est devenu 1.920.000, soit un taux de croissance de 3.740,0%.

Importants ces taux de croissance, toutefois, ils ne doivent pas voiler l'autre face de l'image, le nombre insuffisant des moteurs de recherche pour cette catégorie d'utilisateurs, ce qui nécessite la multiplication des efforts.

L'étude réalisée par (Abdelali et al., 2004) a classifié les systèmes de RI arabes en deux principales catégories, les systèmes plein texte et les systèmes basé sur la morphologie :

- Les systèmes plein texte : cette catégorie comprend la plupart des moteurs commerciaux, par exemple le moteur de recherche al_Idrisi[3] de Sakhr ou encore le moteur ayna[4]. D'autres moteurs multilingues et unicodes appartiennent à la même catégorie, tels que www.alltheweb.com et www.google.com.
- Les systèmes basés sur la morphologie : chaque langue a ses particularités morphologiques et syntaxiques. Cela implique que les règles, les théories, les algorithmes et les méthodes de recherche conçus et développés pour une langue, telle que l'anglais, ne peuvent pas être calqués pour une autre langue, telle que l'arabe qui possède une morphologie très riche.

En plus des difficultés intrinsèques de la langue arabe, il faut rajouter les difficultés humaines et financières nécessaires pour construire des ressources linguistiques pour un but d'évaluation. Néanmoins, on peut citer deux corpus standards comme exemple :

- Une collection LDC (Linguistic Data Consortium) de 383.872 journaux rassemblés entre 1994 et 2000 depuis l'Agence France Presse (AFP) utilisée pour l'évaluation TREC. TREC-2001 a introduit une évaluation à grande échelle des systèmes de RI des documents arabes utilisant des requêtes en arabe, en anglais ou en français (Oard et Gey, 2001).
- Une collection Al-Hayat de plus de 42.000 journaux rassemblés depuis 1994, disponible depuis ELRDA (European Language Ressource Distribution Agency).

[1] www.asharqalawsat.com

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

[3] www.alidrisi.com

[4] www.ayna.com

C'est pour les raisons de difficultés précitées que la grande partie des efforts restent emprisonner aux environnements académiques qui vont éclairer toujours le chemin pour les nouvelles générations des moteurs de recherche arabes.

Ces efforts ont identifiés quatre approches de lemmatisation (Larkey et al., 2002) : construction manuelle de dictionnaires, lemmatisation assouplie (light stemming) qui fait la troncature d'un ensemble restreint d'affixes, analyse morphologique qui tente de trouver les racines et la lemmatisation statistique qui consiste à grouper des variantes de mots en utilisant des techniques de segmentation (clustering).

Les conclusions de ses efforts affirment que la lemmatisation améliore le rappel et la précision. En outre les expérimentations ont montré que les performances de la lemmatisation assouplie sont meilleurs que celles de la lemmatisation régulière (Abdelali et al., 2004).

Il est à noter qu'une bonne tradition s'est instaurée par ARADO (ARab Administrative Development Organisation)[1], c'est bien l'organisation d'une conférence annuelle des moteurs de recherche arabes. Les conférences organisées sont décrites brièvement comme suit :

- 6-10 février 2005 : première conférence à Sharm Elsheikh, Egypt. Elle s'est intéressée principalement au concept des moteurs de recherche et leurs évaluations ainsi que leurs intérêts économiques, sociales et culturels.
- 5-9 février 2006 : seconde conférence à Sharm Elsheikh, Egypt qui porte sur l'utilisation des techniques d'amélioration des moteurs de recherche pour l'assistance des sites arabes.
- 25-29 mars 2007 : troisième conférence à Sharm Elsheikh, Egypt. Elle se focalise sur la construction des moteurs de recherche arabes sur l'Internet, en traçant les stratégies de leur expansion dans le marché arabe.

Nous clôturons cette section par la présentation de deux exemples de moteurs de recherche :

- Morfix est un moteur de recherche unique pour deux langues sémitiques l'arabe et l'hébreu. C'est le produit de MELINGO[2]. Morfix comprend un analyseur morphologique puissant. Etendu en 2003 pour inclure des techniques de RI translinguistique arabe anglais. Il effectue la recherche d'information en arabe en offrant plusieurs options : recherche exacte, morphologique, utilisation d'un thesaurus et l'expansion de requêtes (Moukdad, 2004).
- Sawafi est un projet Germano-Saoudien de moteur de recherche arabe prévu pour la fin de l'année 2006[3]. Il se veut géant tels que les moteurs de recherche internationaux (Google, Yahoo,…). Il est planifié pour être le premier moteur (non-directory) pour chercher le contenu arabe sur le web. Toutefois, en 2007, on n'a pas trouvé sa trace dans la toile.

## 2.2 – Fonctions d'un SRI

### 2.2.1 Architecture générale d'un SRI

La littérature de la RI présente différentes architectures d'un SRI mais avec une grande ressemblance. Dans notre contexte, nous avons inspiré et adopté le schéma de la figure 4. depuis les schémas étudiés dans (Chebeir, 2001) (p. 34) ; (Diem Le, 2003) (p. 9) ; (Mechtri, 2003) (p.15) ; (Schawarkz, 2005) (p. 22).

[1] المنظمة العربية للتنمية الإدارية www.arado.org

[2] Un leader dans l'informatisation de l'hébreu, voir http://www/melingo.com

[3] www.abondance.com

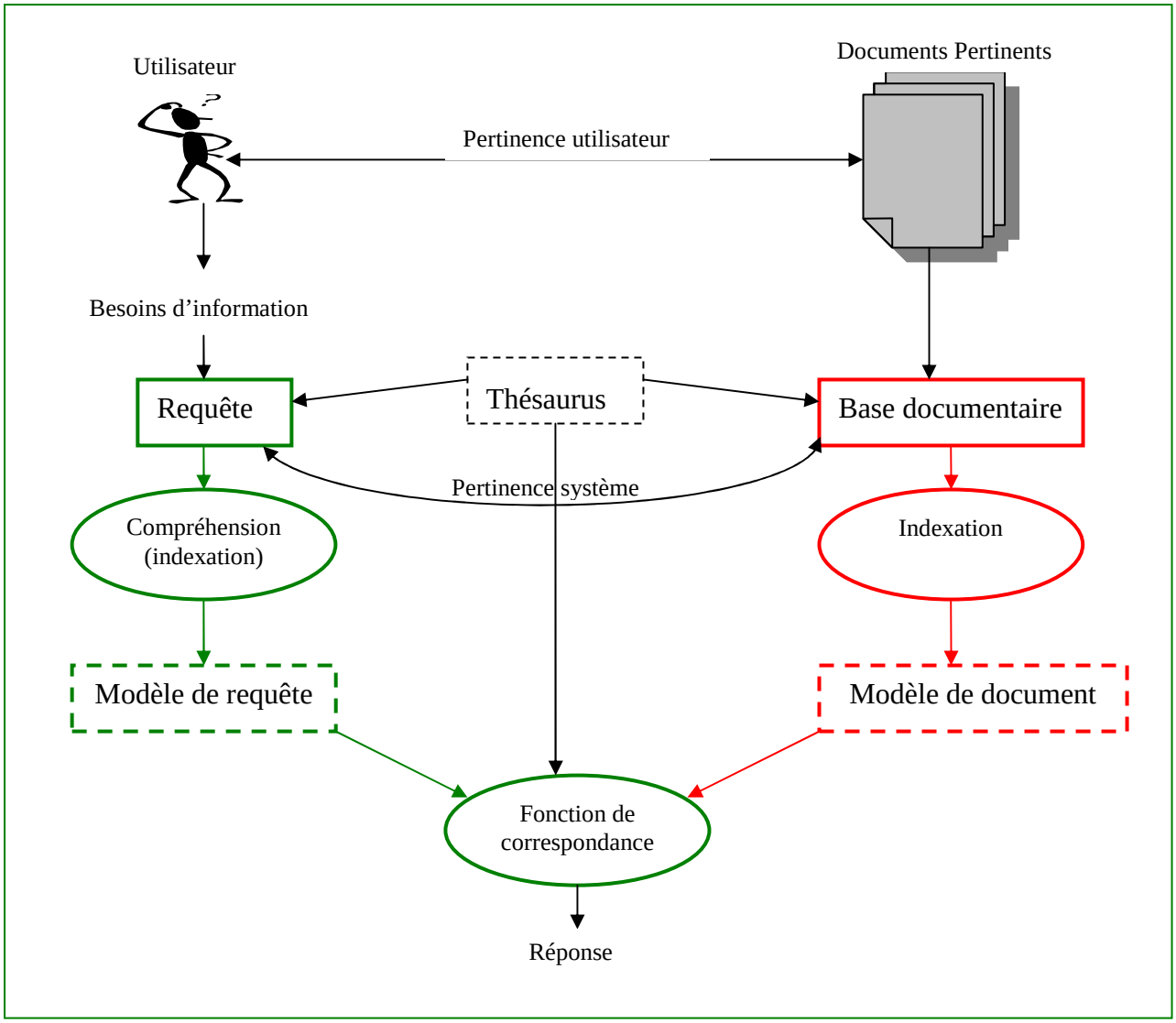



**Figure 4. : Architecture générale d'un SRI**

### 2.2.2 Construction de la base documentaire

Dans (Baeza et Ribeiro, 1999), cette fonction est qualifiée la première de toutes, elle consiste à spécifier les documents à utiliser. La construction de la base documentaire peut être manuelle (bibliothèque universitaire) ou automatique (web).

Dans le web, cette fonction est confiée à un module nommé Robot[1], il se charge de parcourir le web, de collecter les pages avec la structure de liens les reliant pour un objectif d'indexation et pour qu'elles soient supportées par les moteurs de recherche (Manning et al., 2007).

A cet effet, une légère modification de l'architecture générale d'un SRI est nécessaire. La figure 5. illustre les tâches d'un robot.

[1] Robot = wanderer (vagabond) = spider (Araignée) = crawler (qui rampe)

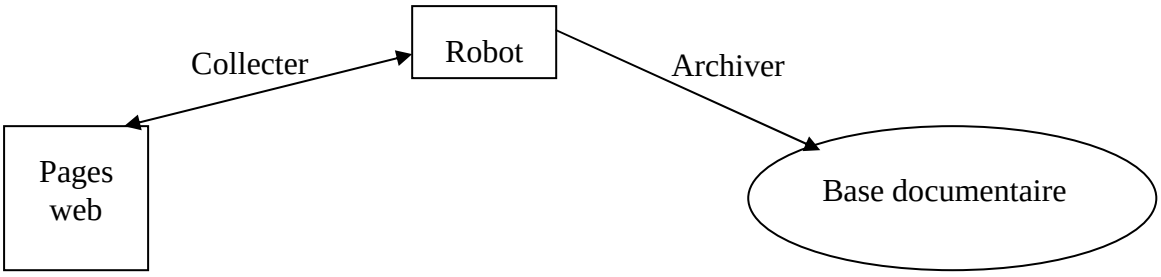



**Figure 5. : Les tâches d'un robot**

Noter que les modules de collecte d'information sont paramétrables. En effet, il est possible de modifier les formats de fichiers pris en compte, la profondeur et la largeur des sites parcourus, la langue du contenu, temps maximal du parcours, nombre de processus parallèles pour parcourir le web,…

Pour plus d'informations techniques sur le développement d'un robot, le chapitre 20 de (Manning et al., 2007) et le chapitre 2 de (Chakrabarti, 2003) méritent d'être étudiés.

### 2.2.3 Expression du besoin informationnel

Un besoin d'information est une sensation qui porterait l'individu à s'engager dans une activité de recherche d'information à travers laquelle le besoin d'information peut être observé (Simonnot, 2006).

Depuis les années 70, beaucoup d'études se sont centrées sur l'usager pour comprendre son comportement informationnel, donnant naissance à plusieurs approches, dont la plus grande partie est réservée pour l'approche cognitive. Une tentative de l'état de la littérature sur le sujet est développée dans (Simonnot, 2006).

L'activité de la recherche d'information peut être facilitée, si un intermédiaire humain (documentaliste) entre dans un dialogue avec l'usager pour mieux définir ses besoins et de les mettre en correspondance avec la collection interrogée.

Dans le cas contraire, l'usager s'adresse directement à un SRI et formule son besoin sous forme d'une requête. Cette requête représente plus ou moins approximativement le besoin d'information.

A titre d'illustration, examinons un extrait du 57$^{eme}$ du RESEAU[1] : "malgré leur expérience dans l'utilisation d'un ordinateur et du web, les étudiants du secondaire ont de réelles difficultés à formaliser leur besoin d'information, à cibler leur sujet de recherche, à formaliser des requêtes efficaces, à identifier la meilleur stratégie pour une recherche donnée et à évaluer la pertinence et la validité des ressources obtenues.".

De ce qui précède, on réalise que l'élaboration d'une continuité sémantique entre des besoins d'information d'un utilisateur et des systèmes informatiques est un processus complexe. Néanmoins nous pensons que des efforts pour alléger une telle complexité doivent être multipliés :

[1] RESEAU (Revue au Service de l'Enseignement et de l'Apprentissage à l'Université) n° 57, avril 2005 : Les étudiants et la recherche d'information.

- Compréhension du comportement informationnel de l'usager : c'est le rôle des chercheurs dans le domaine de sciences de l'information et éventuellement d'autres disciplines (médecine, droit, marketing,…).
- Développement d'outils et techniques d'assistance pour préparer une recherche, tels que la technique d'expansion de requêtes et l'utilisation des thésaurus et des ontologies. Cette tâche est encadrée par les spécialistes en conception et développement des SRIs.
- Développement de compétences informationnelles chez l'usager qui doit être pris en charge par l'usager lui-même, les écoles et les universités.

### 2.2.4 Indexation de la base documentaire et des requêtes

#### *2.2.4.1 Définition :*

Nous proposons la définition de la norme AFNOR NF Z47-102 1996b qui porte sur les principes généraux pour l'indexation des documents : "l'indexation est l'opération qui consiste à décrire et à caractériser un document à l'aide des représentations des concepts contenus dans ce document, c'est-à-dire à transcrire en langage documentaire les concepts après les avoir extraits du document par une analyse".

D'après cette définition, on constate que l'opération d'indexation se déroule en deux phases :

- L'analyse documentaire qui consiste à sélectionner et extraire les concepts les plus importants dans le document.
- La création d'une représentation concise (modèle) pour chaque document.

Noter que ce mécanisme d'indexation peut être appliqué également pour les requêtes.

L'objectif primordial de l'indexation est de faciliter la recherche d'information. Pour mener à bien cette opération, on est amené à utiliser des outils, parmi les quels, les thésaurus.

L'indexation peut se faire automatiquement telle que dans la RI sur le web. En effet c'est le type d'indexation le plus étudié dans la RI. Nous allons l'aborder selon les deux phases sous citées.

#### *2.2.4.2 Sélection des termes d'indexation*

Vu la complexité d'extraction des concepts, en pratique, on cherche des représentations à la place des concepts. Ces mots représentants sont souvent appelés descripteurs ou termes d'indexation.

L'idée de substituer les concepts par des unités linguistiques est assez naturelle, parce qu'il est plus facile de les reconstruire et qu'elles sont porteuses de sens.

Dans la littérature de la RI, il existe plusieurs approches pour extraire des termes d'indexation. C'est l'objet des sections qui suivent.

##### *2.2.4.2.1 Approche basée sur la fréquence d'occurrence des mots*

Cette approche, appelée aussi approche de représentation, se base sur l'assomption qu'un mot qui apparaît souvent dans le texte représente un concept important, ainsi la fréquence d'occurrence des mots est calculée et un seuil sur la fréquence doit être choisi à partir du quel un mot est considéré important. Cependant les statistiques montrent que les mots les plus fréquents sont les mots fonctionnels[1] ("un", "de", "les",…), d'où la nécessité de définir un autre seuil maximal à partit duquel, le mot n'est pas considéré comme index.

[1] Mot fonctionnel = mot outil = mot vide.

Dans le contexte des fréquences d'occurrence des mots, les travaux de Zipf (Elayari, 2005) ont constaté une certaine régularité dans le produit de la fréquence de l'occurrence d'un mot par son rang :

$$rang * fréquence \approx cons\tan te$$

Ce qui signifie que les mots dans les documents ne s'organisent pas d'une manière aléatoire. Cette loi est ensuite étendue à d'autres domaines (Baziz, 2005) tel que la répartition des pixels dans une image, des populations dans une ville et récemment les pages web dans l'Internet. Dans ce dernier cas la relation peut s'exposer comme suit :

Popularité d'une page * nombre d'accès à une page par mois ≈ constante

Dans cette approche, pour choisir les mots qui représentent au mieux le contenu d'un document, un autre concept doit être introduit, il s'agit de la conjecture de Luhn[1] (Luhn, 1958). Cette conjecture met en jeu le facteur d'informativité, autrement dit l'information contenue dans les termes. La juxtaposition des courbes de la fréquence et de l'informativité illustre cette conjecture (Figure 4).

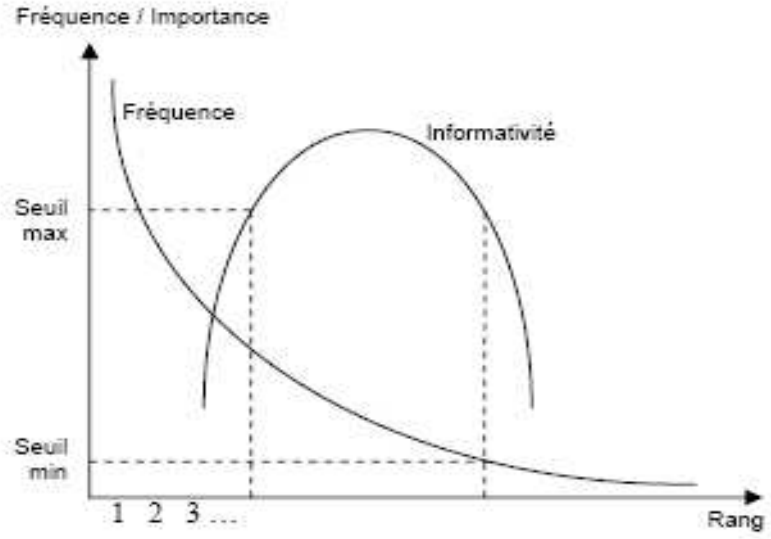



**Figure 6. : La conjecture de Luhn**

Elle stipule que les termes les moins informatifs sont ceux de rangs faibles-très fréquents, et de rangs élevés-très rares. Ainsi, en choisissant les mots ayant des fréquences entre les seuils (Figure 4), on espère obtenir les mots dont l'informativité est la plus élevée.

### *2.2.4.2.2 Approches basées sur la valeur de discrimination*

Cette approche consiste à distinguer complètement ou partiellement un document du reste. On s'intéresse à la discrimination d'un terme dans le corpus au lieu de sa fréquence d'occurrence dans le document.

L'idée est de savoir si un terme à un poids discriminatif dans le corpus. Pour en savoir, on calcule la valeur de discrimination d'un tel terme.

Le principe est de comparer l'uniformité au sein d'un corpus avec celle du corpus transformé dont le terme en question est uniformisé (mis au même poids).

[1] La conjecture de Luhn était utilisée initialement pour le résumé automatique de la littérature technique en se basant sur les méthodes statistiques.

Le calcul est développé dans le modèle vectoriel proposé par SALTON (le modèle vectoriel va être développé plus loin). Dans ce modèle chaque document est représenté par un vecteur de poids comme suit :

$$d_j = (w_{j,1},\ldots,w_{j,i},\ldots,w_{j,n})$$

Où $w_{j,i}$ représente le poids du $i^{eme}$ terme dans le document j.

Formellement, le calcul s'effectue comme suit :

1. Calcul du vecteur moyen (centroïde) du corpus V : le poids de chaque terme du vecteur V est le poids moyen de ses poids dans le corpus :
   $$w_j = \sum_i w_{i,j} / N$$
   Où N est le nombre des documents dans le corpus
2. Calcul de l'uniformité initiale du corpus comme la similarité moyenne des documents avec le centroïde.
   $$U_1 = C * \sum_j sim(d_i,V)$$
   C est une constante de normalisation, en général 1/N.
   $sim(d_i,V)$ est la similarité entre le document $d_i$ et le centroïde V.
3. Calcul de l'uniformité transformée $U_2$ (étape 1 et 2) après uniformisation du poids du terme en question à 0.
4. Calcul de la valeur de discrimination $U=U_2-U_1$

**Listing 1. : Calcul de la valeur de discrimination d'un terme**

Si on obtient une grande amélioration dans l'uniformité du corpus (grande valeur de discrimination) ça implique que ce terme était non uniformément distribué dans le corpus et donc peut être gardé comme terme descripteur. Par contre, si l'amélioration dans l'uniformité est minime ça implique que ce terme était uniformément distribué, et il sera écarté de l'index.

### *2.2.4.2.3 L'indexation dans la pratique*

Les deux approches citées précédemment sont contradictoires, l'une consiste à caractériser un document à travers son contenu, l'autre tend à caractériser un document en le distinguant des autres (Baziz, 2005).

Dans la pratique, on cherche un compromis entre la représentation et la discrimination. La méthode TF*IDF est un bon exemple de ce mélange d'approches.

TF signifie (Term Frequency), pour chaque terme $t_j$ on calcule sa fréquence $tf_{ij}$ dans le document $d_i$.

$$tf_{ij}=f(t_j,d)/Max[f(t,d_i)]$$

Où $f(t_j, d_i)$ représente la fréquence d'occurrence $t_j$ dans le document $d_i$ et $Max[f(t,d_i)]$ est la fréquence maximale des termes dans le document $d_i$.

IDF signifie (Inverted Document Frequency), SALTON a proposé le facteur de fréquence documentaire inverse comme suit :

$$IDF = Log\ N/n$$

Où N est le nombre de documents dans le corpus, n est le nombre des documents qui contiennent le terme en question.

La pondération se réalise par le produit :

$$W_{ij} = TF* IDF = tf_{ij}* Log\ N/n$$

Dans ce cas, on peut choisir à garder, seulement, les termes dont la valeur TF* IDF dépasse un certain seuil.

Ceci dit, dans la pratique et dans une perspective d'améliorer le processus d'indexation, on peut procéder au filtrage des mots fonctionnels à l'aide de l'utilisation de stoplist[1].

Un stoplist est une liste de termes qui apparaissent approximativement dans tous les documents. Ces termes ne servent pas à distinguer un document d'un autre et ils ne seront pas utilisés comme index par les SRIs. De cette manière, on réduit la taille de l'index et le processus de recherche sera accéléré (Schawarkz, 2005).

La lemmatisation coule dans cette même vague d'idées, l'amélioration du processus d'indexation. Dans (Moreau, 2006), les expériences de GAUSSIER et al., pour le français, ont montré que l'intégration d'un module de lemmatisation, dans les SRIs, présente une amélioration de la précision moyenne de 16%. Pour l'arabe, on a déjà introduit, dans la section 2.1.3.3 l'impact de la lemmatisation sur la précision et le rappel.

Le premier algorithme de lemmatisation[2] publié est celui de LOVINS J.B. en 1968, il comprend 35 règles de transformation.

L'algorithme le plus cité est bien celui de PORTER en 1980 (Porter, 1980). Il n'est pas assez conservateur que son précédent en ce qui concerne la production des lemmes. A titre d'exemple "general" devient "gener" et "iteration" devient "iter". PORTER applique 80 règles de transformation.

Les travaux de SAVOY en 1999 méritent d'être cités, leur objectif est d'adapter les approches disponibles en anglais pour le français, l'allemand, l'italien et l'espagnole.

Pour clore ce point, le chapitre 5 (Index Compression) de (Manning et al., 2007) présente l'effet des prétraitements, parmi les quels le stoplist et la lemmatisation, sur la taille des index.

#### *2.2.4.3 Les modèles de la base documentaire et les requêtes*

Après la sélection des termes d'indexation, c'est aux modèles de caractériser le contenu des documents ou des requêtes. Les modèles dans la RI doivent remplir deux rôles (Mechtri, 2003) :

- Créer une représentation interne pour un document ou une requête.
- Définir une méthode de correspondance entre les deux représentations des documents et des requêtes. Ce que qualifie (Moreau, 2006) par une caractéristique solide pour représenter la notion de pertinence.

Formellement, un modèle de RI peut être défini par le triplet {d, q, RSV(d, q)}:

- d : représentation de documents
- q : représentation d'une requête
- RSV(d, q) : (Relevant Status Value) la fonction de correspondance, elle est utilisée pour ordonner les documents retrouvés

Dans ce qui suit, nous nous intéressons aux modèles souvent utilisés dans la RI, en se basant sur le schéma de la figure 5, proposé dans (Moreau, 2006), qui met en évidence les trois familles essentielles des modèles de RI, à savoir :

- Les modèles ensemblistes.
- Les modèles algébriques.
- Les modèles probabilistes.

[1] Le site http://textalyser.net/stoplist.html propose une liste de mots sans intérêt pour le français et pour l'anglais.

[2] Le site http://snowbalt.tartus.org/algorithme./lovins/stemmer.html décrit l'algorithme de Lovins.

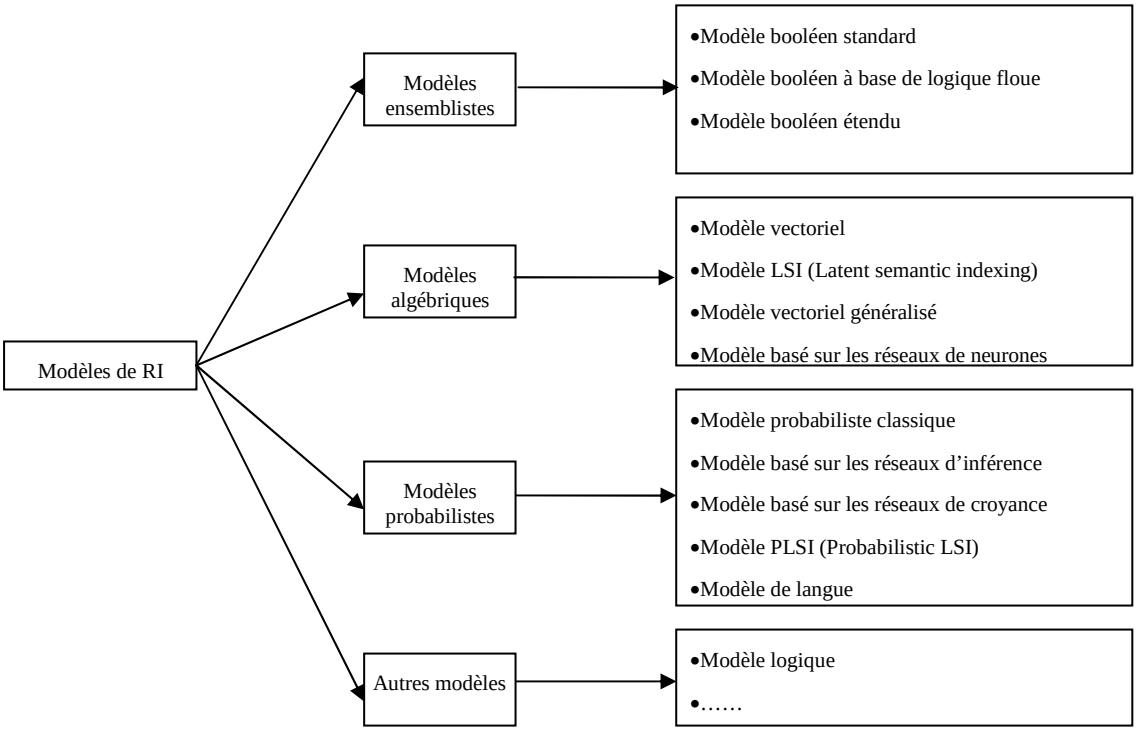



**Figure 7. : Les principaux modèles de RI**

### *2.2.4.3.1 Les modèles ensemblistes*

- **Modèles booléens :**

  C'est le modèle le plus ancien de tous les modèles de RI. Il propose la représentation d'une requête sous forme d'une expression logique quelconque de termes $(q = (t_1 \wedge t_2) \vee (t_3 \wedge \neg t_4))$ tandis qu'un document est représenté par une conjonction de termes $(d = t_1 \wedge t_2 ... \wedge t_n)$. La correspondance RSV(d, q) est déterminée de la façon suivante :

  RSV(d, $t_i$) = 1 si $t_i \in$ d ; sinon 0

  RSV(d, $q_1 \wedge q_2$) = 1 si RSV(d, $q_1$) = 1 et RSV(d, $q_2$) = 1 ; sinon 0

  RSV(d, $q_1 \vee q_2$) = 1 si RSV(d, $q_1$) = 1 ou RSV(d, $q_2$) = 1 ; sinon 0

  RSV(d, $\neg q_1$) = 1 si RSV(d, $q_1$) = 0 ; sinon 0

✓ Avantages :

1. Un formalisme précis, la logique des propositions.
2. Les formalismes de description des documents et des requêtes font partie du même langage.

✓ Inconvénients :

1. La sélection d'un document est basée sur décision binaire, un document est soit pertinent ou non.
2. Pas d'ordre pour les documents sélectionnés, notamment pour les collections volumineuses, le nombre de documents retournés peut être considérable.
3. La formalisation de la requête est difficile et n'est pas toujours évidente pour beaucoup d'utilisateurs. En particulier, les opérateurs booléens ne correspondent pas exactement aux connecteurs linguistiques.

4. La notion de pondération des termes n'est pas prise en compte, un terme a un poids égal à 1 s'il appartient au document, 0 sinon.

Pour remédier à ces inconvénients, une extension du modèle semble indispensable. Le modèle booléen à base de logique floue et le modèle booléen étendu constituent des solutions.

- **Modèles booléens à base de logique floue :**

  L'objectif de cette extension du modèle booléen est de remédier à l'inconvénient de l'absence de la notion de pondération pour les termes. Elle repose sur la théorie des ensembles flous, proposée par ZADEH[1] en 1965, où un élément possède un degré d'appartenance à un ensemble.

  Cette idée a motivé les chercheurs en RI pour modéliser la notion de vague et d'imprécision qui existait à différents niveaux du processus de RI (Baziz, 2005).

  Dans ce modèle, la représentation de la requête demeure invariée par rapport à celle du modèle booléen ordinaire. Toutefois, un document est représenté comme un ensemble de termes pondérés :

  $$d = \{(t_1, a_1),\ldots, (t_i, a_i),\ldots\}$$

  Où $a_i$ est le degré d'appartenance du terme $t_i$ au document d, en général, ce poids est principalement basé sur le nombre d'occurrences d'un terme dans le document.

  En s'inspirant des évaluations classiques de ZADEH, la fonction de correspondance entre une requête et un document peut être formalisée comme suit :

  $RSV(d, t_i) = a_i$

  $RSV(d, q_1 \wedge q_2) = \min[RSV(d, q_1), RSV(d, q_2)]$

  $RSV(d, q_1 \vee q_2) = \max[RSV(d, q_1), RSV(d, q_2)]$

  $RSV(d, \neg q_1) = 1 - RSV(d, q_1)$

  Cette évaluation ne convient pas parfaitement à un processus de RI. Par exemple, pour les requêtes $q_1$ et $q_2$ dont les composants ne jouant pas le même rôle, l'évaluation de la conjonction s'intéresse à la requête dont le degré d'appartenance aux documents est faible, tandis que, lors de l'évaluation de la disjonction, le composant ayant un degré d'appartenance élevé est pris en compte. En outre, les évaluations $RSV(d, q \wedge \neg q) \equiv 0$ et $RSV(d, q \vee \neg q) \equiv 1$ ne sont pas vérifiées. D'autres contre exemples sont présentés dans (Baziz, 2005).

- **Modèles p-norme :**

  Le modèle p-norme[2] est introduit par SALTON (Picarougne, 2004). Le but consiste à attribuer une pondération aux termes des documents et des requêtes ainsi qu'aux opérateurs booléens AND et OR.

  L'idée de base réside dans l'observation de la table de vérité de la conjonction et de la disjonction (Tableau 2.).

[1] Lotfi Askar ZADEH, né le 4 février 1921 à Bakou en Azerbaïdjan. Il a étudié à l'université de Téhéran. Il a introduit la théorie des ensembles flous. Il est professeur à l'université de Berkeley.

[2] Ce modèle tire son nom de la norme vectorielle étudiée dans le calcul matriciel.

| A | B | A∧B | A∨B |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 |

**Tableau 2. : Table de vérité de la conjonction et de la disjonction**

Pour la conjonction, la meilleure correspondance est atteinte dans le cas de la dernière ligne. Tandis que, pour la disjonction, le pire des cas est atteint dans le cas de la première ligne. Ainsi la fonction de correspondance consiste à calculer une sorte de distance entre le point à atteindre ou à éviter. La figure 8. illustre un tel concept.

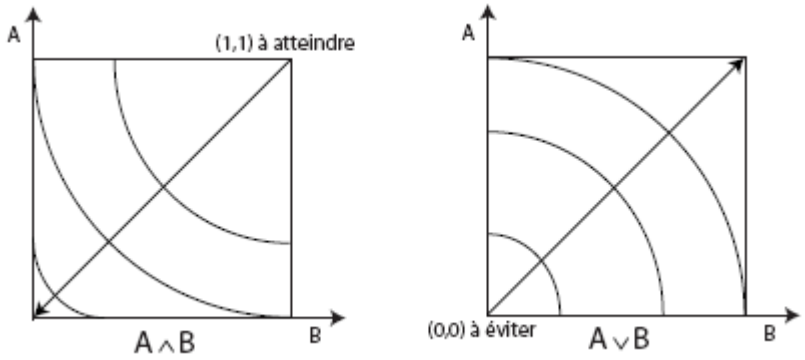



**Figure 8. : Evaluation d'une conjonction ou d'une disjonction**

Les axes des abscisses sont l'évaluation du document B et les axes des ordonnées sont l'évaluation du document A. Dans le cas de A∧B, on cherche à atteindre le point (1,1), par contre, dans le cas de A∨B, on cherche à éviter le point (0,0).

Dans cette optique, SALTON proposa les évaluations normalisées suivantes :

$RSV(d, t_i) = a_i$

$$RSV(d, q_1 \wedge q_2) = 1 - \sqrt{\frac{(1 - RSV(d, q_1))^2 + (1 - RSV(d, q_2))^2}{2}}$$

$$RSV(d, q_1 \vee q_2) = \sqrt{RSV(d, q_1)^2 + RSV(d, q_2)^2}$$

$RSV(d, \neg q) = 1 - RSV(d, q)$

Avant de clore le modèle p-norme, il est à dégager la conclusion suivante : le comportement de ce modèle varie entre le modèle booléen et le modèle vectoriel (Figure 9.) qu'on va développer dans la section suivante.

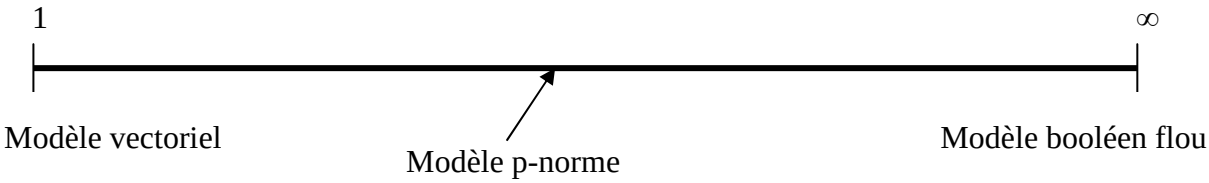



**Figure 9. : Comportement du modèle p-norme**

### *2.2.4.3.2 Les modèles algébriques*

Cette famille de modèles rassemble des modèles de RI basés sur une représentation vectorielle des documents et des requêtes. L'évaluation de la fonction de correspondance revient donc à calculer algébriquement la similarité entre vecteurs.

Dans ce qui suit, on s'intéresse, à titre illustratif, aux modèles suivants : le modèle vectoriel, le modèle LSI (Latent Semantic Indexing), le modèle vectoriel généralisé, et le modèle basé sur les réseaux de neurones.

- **Le modèle vectoriel :**

  Le modèle vectoriel ou aussi VSM (Vector Space Model) a été mis en œuvre, dès 1971, par SALTON dans le système SMART (Baziz, 2005). Les documents et les requêtes sont représentés par des vecteurs dans le même espace vectoriel. Un vecteur est composé de termes pondérés dans un espace à n dimensions, où n représente le nombre de termes d'indexation.

$$\vec{d} = w_{j,1},..., w_{j,i},...w_{j,n}$$

$$\vec{q} = w_{q,1},..., w_{q,i},...w_{q,n}$$

  Où $w_{j,i}$ (resp. $w_{q,i}$) représente le poids du $i^{ème}$ terme dans le document j (resp. dans la requête q).

  Une collection de d documents décrite par n termes peut être représentée par une matrice dite termes x documents de dimension n x d. chaque colonne de la matrice représente un document de la collection.

  Une des mesures de similarités les plus utilisées est celle du cosinus de l'angle entre le vecteur du document et celui de la requête (Figure 10.).

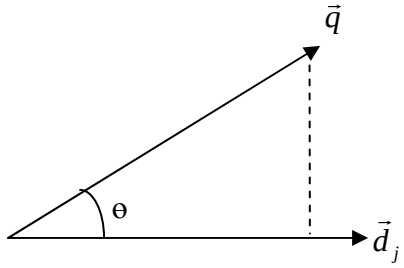



**Figure 10. : Mesure de similarité cosinus**

$$sim(d_j, q) = \cos\theta = \frac{\|d_g\| \cap \|q\|}{\sqrt{\|d_g\| * \sqrt{\|q\|}}} = \frac{\sum_{i=1}^{n} w_{g,i} * w_{q,i}}{\sqrt{\sum_{i=1}^{n} w_{j,i}^2} * \sqrt{\sum_{i=1}^{n} w_{q,i}^2}}$$

  Cette valeur de similarité correspond au score qui sera attribué à chaque document et traduit son degré de pertinence par rapport à la requête. En effet, une valeur de sim ($d_j$, q) égale à 1 traduit une inclusion du document $d_j$ de tous les termes de la requête q, par contre une valeur de sim ($d_j$, q) égale à 0 traduit une disjonction entre le document et la requête.

  Il est à souligner que Les calculs de similarités s'effectuent sur la matrice termes x documents. (Berry et al., 1999) contient plusieurs exemples illustratifs de calculs inhérents au modèle VSM.

✓ Avantages :

1. La pondération améliore les résultats de la recherche.

2. La mesure de similarité permet un classement ordonné des documents selon leur pertinence vis-à-vis la requête

✓ Inconvénients:

1. l'ordre de mots n'est pas pris en compte à cause de la représentation des documents et des requêtes sous forme de "sac de mots"[1].
2. L'hypothèse de l'orthogonalité (indépendance des mots) n'est pas valide dans la pratique, car la plupart des mots dans une langue entretiennent des relations les uns avec les autres (Moreau, 2006). A titre d'exemple, un document qui contient le mot véhicule et une requête représentée par le mot automobile ne peuvent pas être appariés.

Pour contourner ces inconvénients, des variantes du modèle vectoriel ont été proposées. C'est ce qu'on va développer dans les sections qui suivent.

- **Le modèle LSI (Latent Semantic Indexing) :**

Le modèle LSI est une variante du modèle vectoriel qui transforme la représentation traditionnelle, par mots clé en une représentation plus conceptuelle et plus sémantique. Il vise à rapprocher les documents et les requêtes sémantiquement similaires (Moreau, 2006).

Autrement dit, le modèle LSI transforme (map) un espace vectoriel de dimension élevé en un espace de dimension faible (Berry et al., 1999). Il remplace la matrice originale (termes x documents) par une matrice dont les colonnes constituent un sous-espace de l'espace des colonnes originales en considérant des combinaisons linéaires de termes au lieu de représenter chaque dimension par un terme unique.

Ces combinaisons linéaires de termes permettent de mieux ressortir le sens latent entre les mots, ainsi les concepts contenus dans les documents peuvent être mieux exprimés.

Cette fonction de transformation peut être réalisée, mathématiquement, par une décomposition en valeurs singulières[2] de la matrice termes x documents. Cette décomposition se matérialise par une multiplication de la matrice de vecteurs singuliers U, de la matrice diagonale de valeurs singulières $\sum$ et de la matrice de vecteurs singulier droite $V^T$.

$$X = U\sum V^T$$

Après la traduction de la requête utilisateur dans l'espace réduit, la correspondance entre documents et requête consiste à mesurer la similarité entre les vecteurs. En général, on utilise la mesure de cosinus. Les documents résultat peuvent être classés selon leur pertinence par rapport à la requête.

✓ Avantages :

1. La représentation par clustering de mots permet une représentation plus sémantique des documents.
2. La réduction de la dimension de l'espace vectoriel permet d'économiser en espace de stockage et en nombre de ressources nécessaires pour le calcul (Berry et al., 1999 ; Aswani et Srinivas, 2006).

✓ Inconvénients:

1. La phase de réduction de la matrice originale peut être coûteuse en terme de calcul pour les matrices d'envergure.

[1] En anglais "bag of words"

[2] En anglais Singular Value Decomposition (SVD), outil de factorisation de matrices rectangulaires réelles ou complexes.

- **Le modèle vectoriel généralisé :**

  Le modèle vectoriel classique repose sur le principe de l'orthogonalité des vecteurs de termes deux à deux $(\vec{d}_i \bullet \vec{d}_j = 0, \forall i \neq j)$ (Desjardins, 2006).

  Par contre, WONG et al. (Wong, 1985) ont constaté que les vecteurs de termes sont linéairement indépendants mais pas nécessairement orthogonaux. En se basant sur ce principe, ils ont développé le modèle vectoriel généralisé (General Vector Space Model, GVSM).

  Dans le modèle GVSM, le calcul se base sur la cooccurrence des termes dans les documents de la collection, c'est-à-dire qu'on s'intéresse aux documents deux à deux, trois à trois, etc…

  Avec cette nouvelle vision, un document sera représenté dans un espace à $2^n$ dimensions, où n est le nombre de termes du vocabulaire. En effet, chaque combinaison représente un cas de cooccurrence. Par exemple la combinaison (0,0,…,0) représente les documents ne contenant aucun terme, la combinaison (0,0,…,1) représente les documents contenant uniquement le premier terme et la combinaison (1,1,…,1) représente les documents contenant la totalité des termes du vocabulaire.

  Le fondement mathématique et la représentation matricielle d'un tel modèle sont détaillés dans (Wong, 1985 ; Desjardins, 2006).

  Du point de vue évaluation du modèle, les expérimentations (Wong, 1985) réalisées sur deux collections ont montré que le modèle GVSM est significativement meilleur (rappel et précision) par rapport au modèle classique ceci en compromis avec une complexité nettement supérieur.

- **Le modèle basé sur les réseaux de neurones :**

  Le paradigme de réseaux de neurones artificiels (RNA)[1] est un modèle d'apprentissage connu dans le domaine de l'apprentissage machine comme celui de l'IA.

  En RI, le formalisme des réseaux de neurones propose un modèle composé de plusieurs couches (Figure 11.) (Moreau, 2006) :

  - Une couche d'entrée qui représente la requête, chaque nœud correspond à un de ses termes.
  - Une couche qui représente les termes de la collection, chaque nœud correspond à un terme.
  - Une couche qui représente les documents, chaque nœud correspond à un document de la collection

  Le mécanisme d'appariement peut être décrit comme suit :

  - Initialement, les termes de la requête provoquent des activations qui se propagent vers les nœuds termes de la collection.
  - Les nœuds termes de la collection envoient des signaux aux nœuds documents à travers les différentes connexions pondérées du réseau.
  - Les documents qui ont reçu le plus de signaux sont considérés comme les plus pertinents.

[1] McCulloch et Pitts furent les premiers à développer la théorie fondamentale sur les neurones artificiels en 1943.

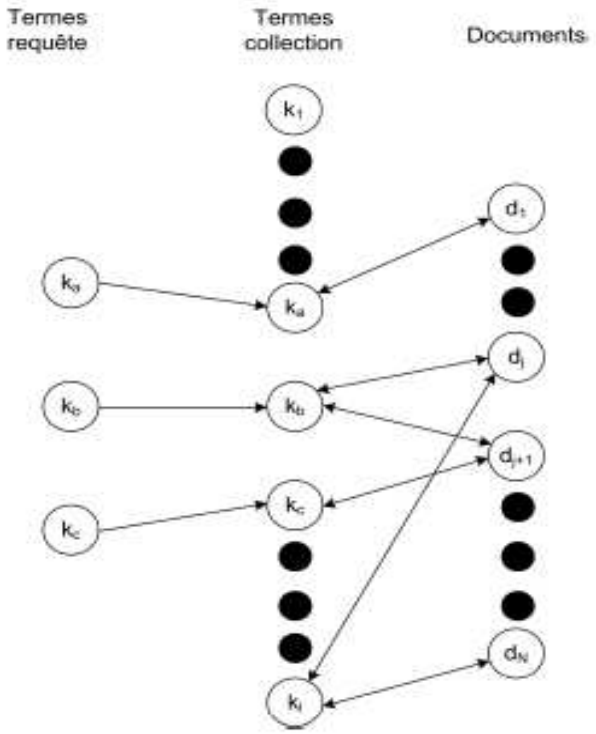



**Figure 11. : Modèle de réseau de neurones pour la RI**

Les trois points suscités s'appellent phase d'activation qui peut être complétée par une seconde phase qui correspond à une forme de rétroaction de pertinence (Moreau, 2006) :

- Les documents considérés pertinents génèrent de nouveaux signaux vers des termes de la collection.
- Les nœuds des termes de la collection envoient à leur tour de nouveaux signaux dirigés vers les nœuds des documents, ainsi le résultat peut contenir des documents dont les termes ne sont pas nécessairement contenus dans la requête.

✓ Avantages :

1. Possibilité de représenter les divers relations entre termes (synonymie, voisinage), entre documents (similitude, référence) ainsi que la fréquence et le poids des termes dans les documents.

✓ Inconvénients:

1. Manque de fondement mathématique.
2. Difficulté, voire incapacité à expliquer, par l'utilisateur, les résultats obtenus.

### *2.2.4.3.3 Les modèles probabilistes*

Les recherches sur les modèles probabilistes ont commencé depuis le milieu des années 70. C.J. Van Rijsbergen, S. Robertson et K. Spark Jones sont les précurseurs à proposer des modèles probabilistes (Nie, 2007).

A partir des années 90, les approches probabilistes ont connu une dynamique fluorescente et se sont montrées performantes dans TREC.

Le principe des modèles probabilistes est de représenter la similarité d'un document vis-à-vis une requête par une probabilité de pertinence.

Dans ce qui suit, on considère un document d, une requête q et on cherche à calculer la probabilité que d soit pertinent pour q.

Il est à signaler que les modèles probabilistes ne tiennent pas compte des autres documents lors du calcul de la correspondance entre documents et requête. Autrement dit, chaque document est considéré individuellement au besoin d'information de l'utilisateur (Moreau, 2006).

- **Le modèle probabiliste classique :**

  L'idée de base des modèles probabilistes est de définir l'ensemble de documents pertinents (noté R pour Relevance) et l'ensemble de documents non-pertinents (NR).

  Si le système est en mesure de définir une probabilité de pertinence P(R/d) (resp. de non-pertinence P(NR/d)) pour qu'un document d soit pertinent par rapport à une requête q, alors il est possible de procéder à un classement des documents résultat dont la probabilité de pertinence est supérieur à la probabilité de non pertinence (P(R/d) > P(NR/d)). Un tel classement est appelé "Probability Ranking Principle", pour plus de détail de ce principe, se référer à (Robertson, 1977).

  Pour le calcul de P(R/d) et P(NR/d), on doit passer par le théorème de Bayes :

$$P(R/d) = \frac{P(d/R) \bullet P(R)}{P(d)}$$

$$P(NR/d) = \frac{P(d/NR) \bullet P(NR)}{P(d)}$$

  P(d/R) : la probabilité que d fait partie de l'ensemble pertinent.

  P(R) : la probabilité de pertinence, c'est-à-dire, si on choisit un document au hasard dans le corpus, la chance de tomber sur un document pertinent.

  P(d) : la probabilité que d soit choisi.

  Le calcul du score peut être noté comme suit :

$$RSV(d,q) = \frac{P(d/R)}{P(d/NR)}$$

  Car P(R) et P(NR) sont des constantes pour une requête donnée.

  Pour calculer P(d/R) (resp. P(d/NR)) on s'appuie généralement sur la probabilité de pertinence des termes individuels du document. D'où

$$P(d/R) = P(t_1=x_1, t_2=x_2, \ldots, t_i=x_i, \ldots/R)$$

$$P(d/NR) = P(t_1=x_1, t_2=x_2, \ldots, t_i=x_i, \ldots/NR)$$

  Où $x_i = 1$ ou 0 représente la présence ou l'absence du terme.

  Pour simplifier les calculs, on suppose que les différents termes sont indépendants (hypothèse d'indépendance). Ainsi :

$$P(d/R) = \prod_{(ti=xi) \in d} P(t_i = x_i / R)$$

$$P(d/NR) = \prod_{(ti=xi) \in d} P(t_i = x_i / NR)$$

  Le problème qui réside maintenant, c'est bien l'estimation de $P(t_i = x_i / R)$ (resp. $P(t_i = x_i / NR)$). Une solution consiste à appliquer une loi de distribution sur un échantillon de documents où les hypothèses de pertinence sont déjà connues (Moreau, 2006).

Ultérieurement, plusieurs améliorations ont été apportées au modèle de base, parmi lesquelles, la formule BM25 qui constitue une approximation du modèle 2-Poisson (Robertson et Walker, 1994). Cette formule est à la base du SRI OKAPI[1]

- **Les modèles basés sur les réseaux d'inférence**

  Les modèles probabilistes classiques souffrent de la difficulté d'établir un compromis raisonnable entre le nombre de probabilités de base à estimer et les hypothèses d'indépendances nécessaires à la réduction de ce nombre (Desjardins, 2006).

  Dans cette optique, les modèles d'inférences tentent de combler cette faille en proposant une représentation qui permettrait de choisir plus précisément les hypothèses d'indépendances et d'obtenir un compromis raisonnable.

  Le facteur commun avec les modèles probabilistes classiques et qu'elles utilisent la théorie des probabilités et le théorème de Bayes.

  Un réseau d'inférence est un cas particulier des réseaux bayesiens. Il est représenté sous forme d'un graphe acyclique directionnel[2] (orienté), où les nœuds représentent des variables propositionnelles ou des constantes et les arcs représentent les relations de dépendances entre les propositions (Turtle et Croft 1989 ; Turtle 1991 ; Desjardins, 2006). Si une proposition, représentée par un nœud p, implique une proposition, représentée par un nœud q, la relation entre p et q sera représentée par un arc orienté de p vers q.

  Appliqué à la RI, le réseau d'inférence se divise en deux sous-réseaux, un pour les documents et l'autre pour la requête (Figure 12.).

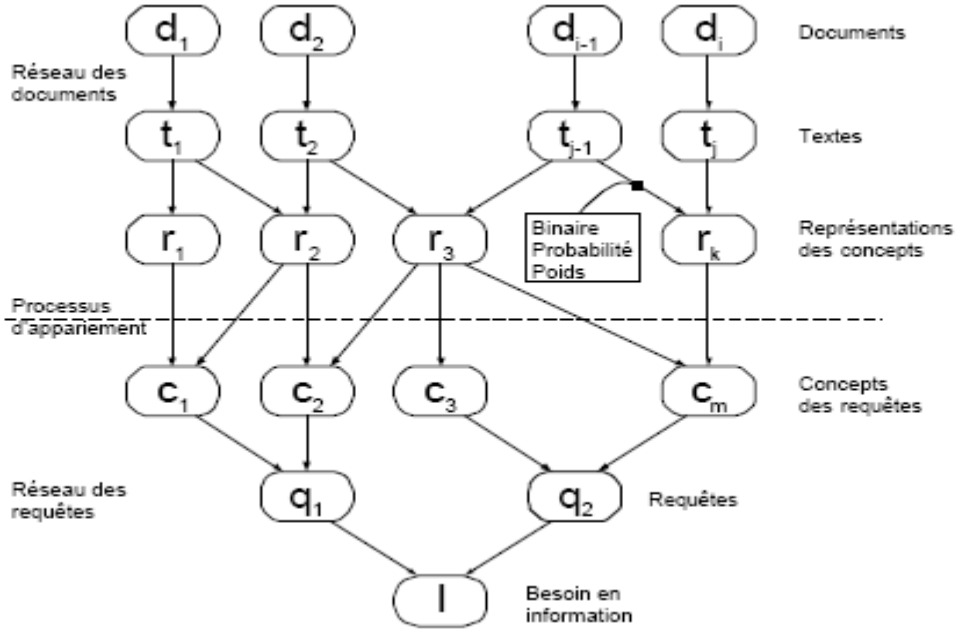



**Figure 12. : Modèle générique d'un réseau d'inférence**

Le sous-réseau document est composé de trois couches hiérarchiques :

[1] OKAPI est un SRI développé au "polytechnic of central London" entre 1982 et 1988. L'installation d'OKAPI est disponible sur le site http://www.soi.city.ac.uk/~andym/OKAPI-PACK/

[2] En anglais Directed Acyclic Graph (DAG)

- Les nœuds documents $d_i$ qui correspondent à la probabilité d'observer un document de la collection. Ces probabilités sont habituellement initialisées à (1/nombre de documents de la collection)
- Les nœuds représentant les termes $t_i$ qui correspondent à la probabilité d'observer un terme dans un document.
- Les nœuds représentant les concepts $r_i$ peuvent être générés par différentes techniques (Desjardins, 2006) : assignation manuelle des mots ou d'expression, extraction automatique, utilisation de thésaurus ou d'ontologies, etc… Ces nœuds correspondent à une probabilité conditionnelle $P(r_k/t_j)$ d'observer un concept étant donné l'ensemble de ses nœuds parents.

En ce qui concerne le sous-réseau requête qui comporte une feuille unique pour représenter le besoin d'information et des racines qui correspondent aux concepts qui le représentent. Telle que illustré dans la figure 12, une couche intermédiaire de nœuds de requête peut être utilisée quand multiples requêtes expriment le besoin d'information. Cette couche intermédiaire peut modéliser la rétroaction pour reformuler le besoin. Le sous-réseau requête est reconstruit à chaque nouvelle demande de besoin.

Ceci dit, les deux sous-réseaux sont reliés par le biais des relations de dépendance entre les concepts de requêtes ($c_i$) et des concepts représentant la collection ($r_k$). Cette connexion modélise le processus d'appariement entre de tels concepts.

Dans le cas le plus simple, les deux concepts sont identiques, ainsi chaque concept de requête possède exactement un parent. Dans des cas plus complexes, un concept de requête peut avoir plusieurs concepts de la collection.

Une fois un document activé, le réseau calcule la probabilité qu'un tel document rencontre le besoin d'information, ainsi les documents résultat, dépassant un seuil fixé de pertinence, peuvent être ordonnés.

Le SRI INQUERY est un exemple utilisant le modèle d'inférence (Callan et al, 1992).

✓ Avantages :

1. Possibilité de représenter les documents sous plusieurs formes.
2. Possibilité d'exprimer le besoin d'information par une combinaison de requêtes.
3. Adaptation du processus d'appariement en offrant la possibilité d'intégrer différentes stratégies de recherche en parallèle (Moreau, 2006).

✓ Inconvénients:

1. Le calcul des probabilités nécessite un temps exponentiel par rapport au nombre de termes de la requête.

- **Les modèles basés sur les réseaux de croyance**

Les réseaux de croyance, proposés par RIBEIRO-NETO B. et MUNTZ B. (Moreau, 2006), sont une généralisation des réseaux d'inférence. Ils s'en distinguent principalement par le sens des arcs. En effet, les valeurs se propagent de la requête vers les documents (Figure 13).

Formellement, les réseaux de croyance utilisent la probabilité P(d/q). Cette mesure peut être interprétée comme le rappel d'un document $d_j$ par rapport à une requête q.

En pratique, seuls les termes des requêtes sont considérés pour un appariement. Dans ces circonstances, les réseaux de croyances définissent un espace commun et unique pour les concepts des documents et ceux des requêtes (Desjardins, 2006).

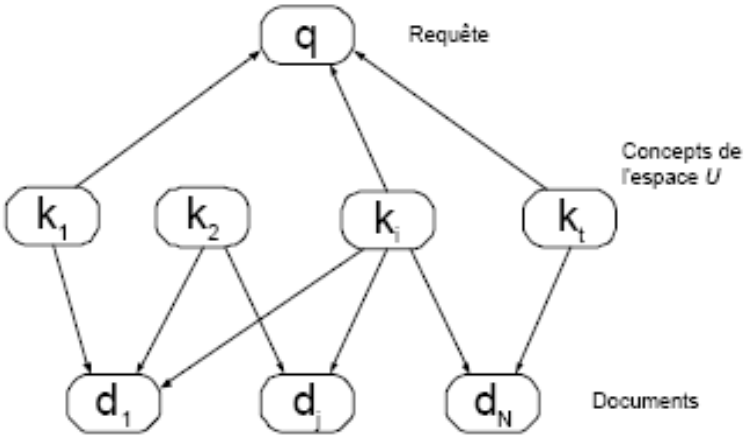



**Figure 13. : Modèle générique d'un réseau de croyance**

De plus, les réseaux de croyances utilisent les probabilités suivantes :

- $P(q) = \sum_u P(q/U)*P(U)$ comme le degré de couverture d'une requête q sur U.
- $P(dj) = \sum_u P(d_j/U)*P(U)$ comme le degré de couverture d'un document $d_j$ sur U.

En ce qui concerne le classement des documents résultat, les réseaux de croyances utilisent la mesure suivante (Desjardins, 2006) :

$$P(dj/q) = \eta \sum_u P(d_j/k_i)*P(q/k_i) * P(k_i)$$

Où les concepts $k_i \in U$ sont, initialement, équiprobables : $P(k_i) = \left(\frac{1}{2}\right)^t$, t est égale au nombre de termes dans U et $\eta$ est une constante de normalisation.

- **Le modèle d'indexation sémantique latente probabiliste (PLSI)**

Nous avons vu précédemment que l'idée directive du modèle LSI repose sur une réduction de la matrice *termes x documents* à ses dimensions principales en utilisant une décomposition en valeurs singulières.

La même idée a été reprise par HOFMANN T. (Hofmann, 1999), mais cette fois ci par une approche probabiliste nommée PLSI (Probabilistic Latent Semantic Indexing).

Le corpus est vu comme un ensemble de couples termes/document représenté par une matrice X de taille N x D. chaque élément de la matrice représente la probabilité de cooccurrence du mot i et du document j ($w_{ij} = X_i(t_i, d_j)$).

La méthode suppose, pour chaque couple, l'existence d'une variable de classe cachée (non observée) liée au thème. Si on considère K classes, on peut définir trois matrices U, $\sum$ et V :

U : N x K, $u_{ij} = P(t_{ij}/z_j)$

$\sum$ : K x K, $\sigma_{ij} = P(z_j)$

V : D x K, $v_{ij} = P(d_i/z_j)$

Sous l'hypothèse d'indépendance conditionnelle, pour la variable de termes et la variable de documents, conditionnellement à la variable de classe non observée, on aura la relation :

$$X = U\sum V^t$$

Ceci est analogue à une SVD de la matrice X.

PLSI a été évaluée en utilisant quatre standards en matière de collection de RI (Chakrabarti, 2003) : MED (1033 résumés de journaux médicaux), CRAN (1400 documents sur l'aéronautique), CACM (3204 résumés depuis des périodiques d'informatique) et CISI (1460 résumés liés aux sciences bibliothécaires).

L'évaluation s'est montrée favorable pour PLSI en terme de rappel et de précision avec le standard TFIDF avec un classement basé sur la mesure de cosinus[1].

- **Les modèles de langue**

Les modèles de langue initiés par PONTE et CROFT (Ponte et Croft, 1998) reposent sur un principe qui diffère à celui des approches probabilistes classiques. En effet, ils ne cherchent pas à modéliser la notion de pertinence (évaluation de la probabilité qu'un document soit pertinent étant donné une requête) ; ils tentent, plutôt, à estimer uniquement la probabilité qu'une requête puisse être générée par le document (Baziz, 2005).

Les modèles de langue sont généralement utilisés en reconnaissance de la parole et en traduction (Moreau, 2006).

En RI, plusieurs hypothèses sont considérées :

- Un document est considéré comme un échantillon d'un langage particulier, ainsi, un modèle de langue (Md) est suggéré pour chaque document de la collection.
- Une requête (q) est considérée comme une phrase générée par un document (modèle de langue)
- La fonction de correspondance entre une requête et un document consiste à estimer la probabilité qu'une requête q puisse être générée par un modèle de langue $M_d$.
- Le classement des documents résultat est réalisé selon un ordre décroissant de la probabilité de génération $P(q/M_d)$.

Un défi surgira lors du calcul de la probabilité $P(q/M_d)$. Il consiste à éviter d'assigner une probabilité nulle à un document qui lui manque un ou plusieurs termes de la requête (Ponte et Croft, 1998). Une solution consiste à utiliser des techniques de lissage[2].

Sur la façon de réaliser telles techniques, une série de méthodes est proposée dans la littérature, à savoir, lissage de Laplace, de Good-Turing, de Backoff, par interpolation, de Dirichlet, etc… Pour d'avantage informations, voir (Boughanem et al., 2004).

Les recherches actuelles, dans le domaine, tendent vers la conception des modèles de langue susceptibles de dépasser l'hypothèse d'indépendance entre les termes.

✓ Avantages :

1. Intégration dans un seul modèle la phase d'indexation et de recherche, ainsi une réduction de calcul.
2. Ils ne nécessitent aucun jugement de pertinence.

✓ Inconvénients:

1. Difficultés d'incorporer les stratégies de rétroaction de pertinence et les préférences de l'utilisateur.

[1] Se référer à (Chakrabarti, 2003) pour la représentation graphique des résultats de l'évaluation.

[2] En anglais smoothing, une technique qui consiste à assigner une probabilité aux termes (ou n-grammes) non observés dans le corpus.

## 2.3 - Exemples de SRI

Dans cette section, on va décrire deux systèmes de recherche d'information. Les systèmes sont choisis pour être représentatifs et non pas exhaustifs. L'un est le premier SRI expérimental, le système SMART, l'autre est l'un des moteurs de recherche les plus utilisés, actuellement, dans le monde, c'est bien Google.

### 2.3.1 SMART

Le contenu de cette section est largement inspiré[1] de (Buckley C., 1996). Le système SMART[2] est un système de RI expérimental. C'est une implémentation du modèle vectoriel proposé par SALTON dans les années 60.

SMART a été construit entre 1968 et 1970 dont l'objectif primaire était de fournir une plateforme pour conduire des recherches dans le domaine de RI. Il a été réécrit dans les années 80[3].

#### *2.3.1.1 Indexation*

La figure 14. illustre le processus d'indexation des documents et des requêtes qui sont très similaires.

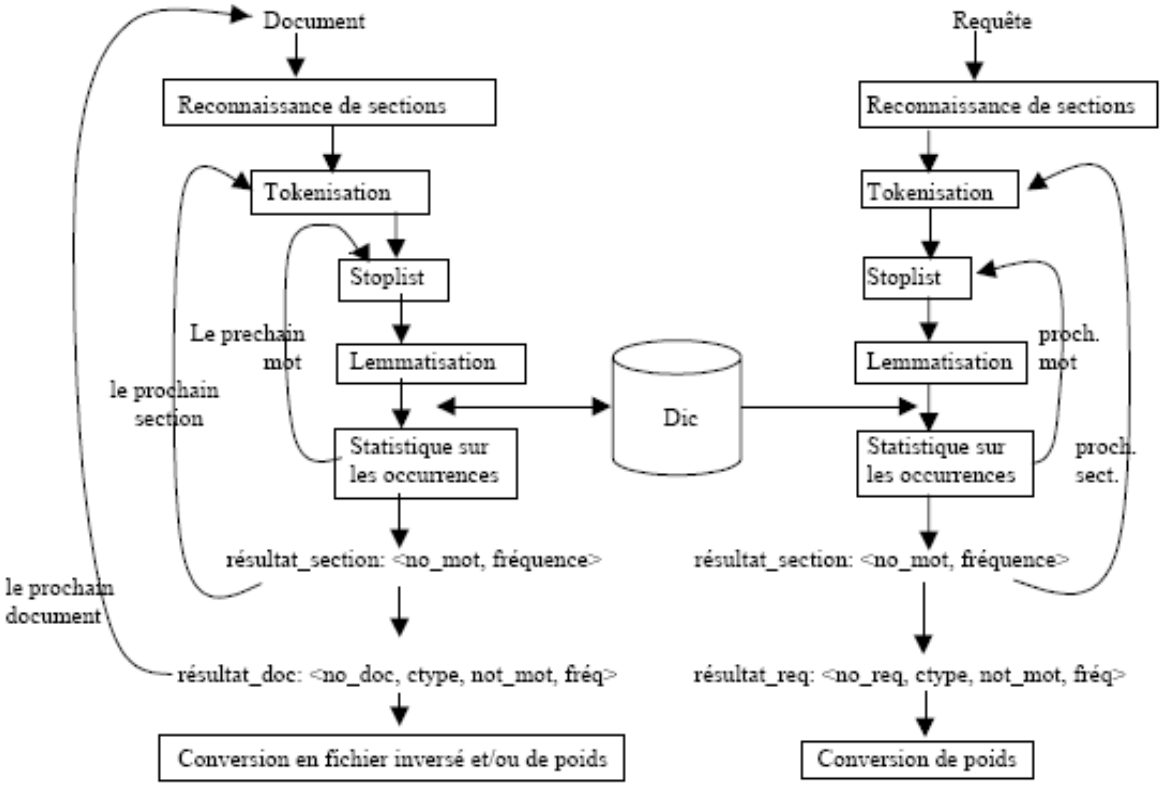



**Figure 14. : Indexation de document et de requête**

Dans cette section, le mot document désigne à la fois document et requête, dans le cas de différence, des précisions seront mentionnées.

Le processus d'indexation comprend les étapes suivantes :

[1] Les notes de cours IFT6255-Hiver 2007 inhérentes à la recherche d'informations incluent une bonne présentation du système SMART.

[2] SMART (System for the Mechanical Analysis and Retrieval of Text) aussi appelé (Salton's Magic Automatic Retrieval Technique).

[3] Version disponible gratuitement pour les recherches à l'adresse : ftp://ftp.cs.cornell.edu/pub/smart/

- Reconnaissance de sections[1] qui se fait par le biais des marqueurs tels que : <Title>, <Date>, <Body> etc.
- Tokenisation : chaque section doit être coupée en mots (tokens), et les mots séparateurs seront identifiés.
- Stoplist : les tokens reconnus seront filtrés en utilisant un stoplist.
- Lemmatisation : les tokens filtrés seront assujettis, par la suite, à une transformation en une certaine forme standard. Après l'obtention d'une forme standard, une comparaison de cette dernière aux entrés d'un dictionnaire sera effectuée. Chaque entrée du dictionnaire correspond à un index ayant la structure < no-token, nature, token>, où no-token est le numéro d'identification, token est le mot correspondant qui peut être de nature chiffre, nom propre, mot normal, etc.
- Statistique : calcul de la fréquence d'occurrence de chaque token dans le document.
- Itération : on distingue trois itérations pour l'indexation de documents : une itération au niveau de mot, une autre au niveau de section et une dernière au niveau de document, cette itération peut être appliquée à une requête si le corpus utilisé contient un ensemble de requêtes.
- Résultat d'indexation : le résultat, pour les documents, est présenté selon la structure suivante : <no-doc, ctype, no-token, freq>.
- Conversion de résultat : cette étape s'effectue suite à l'indexation de tous les documents. Elle peut prendre deux formes :
    - Convertir des vecteurs en fichier inversé : cette conversion est appliquée uniquement au résultat d'indexation de documents. Dans ce cas, le fichier inversé sera trié selon l'ordre croissant du no-token.
    - Convertir les poids de termes : cette conversion consiste à transformer le poids de chaque terme d'un document.

#### *2.3.1.2 L'Evaluation*

L'évaluation SMART ne peut être effectuée que sur une collection de test qui comprend un ensemble de requêtes et les jugements de pertinence portés par des experts pour chaque requête.

##### *2.3.1.2.1 Evaluation d'une requête*

L'évaluation d'une requête consiste en un appariement entre le fichier inversé des documents et le résultat d'indexation de la requête.

Pour se faire, SMART utilise le produit interne comme similarité :

$$Sim(d, q) = \sum_i (p_i * q_i)$$

Où $p_i$ et $q_i$ sont les poids d'un mot dans le document d et dans la requête q.

SMART adopte une méthode globale d'évaluation, il effectue le calcul de similarité pour tous les documents à la fois selon l'algorithme suivant :

[1] Une section dans SMART correspond à un champ de document.

```
Initialiser Sim(dj, q) à 0 pour tous dj ;
Pour chaque mot (no-tokeni) dans la requête q (avec poids qi) faire ;
     Trouver dans le fichier inversé tous les documents dj incluant ce mot
     avec le poids pi
     Pour chaque dj dans cet ensemble faire ;
       Sim (dj, q) = Sim (dj, q) + pi * qi
```

**Listing 2. : Evaluation d'une requête dans SMART**

A la fin de cet algorithme, on obtient la valeur de similarité pour chaque document.

Il est à noter que dans SMART, on peut procéder à un calcul de similarité séquentiel.

#### *2.3.1.2.2 Evaluation d'un système*

L'évaluation d'un système repose sur l'évaluation de la précision et le rappel. Ainsi, le système à tester doit procéder en premier lieu à une indexation de documents et de requêtes et de produire, ensuite, une liste de documents comme réponse pour chaque requête. Les réponses du système seront alors comparées avec celles des experts.

En général, on ne modifie qu'un seul composant du système pour mesurer l'effet d'une technique (lemmatisation, tokenisation, stoplist,…) qu'on peut implémenter de diverses façons.

SMART contient des outils qui permettent d'évaluer la courbe précision-rappel inhérente aux diverses implantations du composant testé.

#### *2.3.1.2.3 Rétroaction de pertinence (relevance feedback)*

La technique de rétroaction de pertinence a été inventée par ROCCIO. L'idée est de prendre en compte l'évaluation de l'utilisateur qui peut indiquer les réponses qui lui sont pertinentes et celles qui ne le sont pas. Sur l'optique de ces indications, le système peut reformuler la requête, ainsi la nouvelle requête peut devenir plus proche des documents pertinents et plus loin des documents non pertinents.

La formule générale de Roccio est la suivante :

$$q' = \alpha*q + \beta*P - \gamma*NP$$

où : q' : nouvelle requête

q : ancienne requête

P : vecteur centroïde des documents jugés pertinents

NP : vecteur centroïde des documents jugés non pertinents

$\alpha$, $\beta$, $\gamma$ : des paramètres à fixer de façon expérimentale

### 2.3.2 Google

Google a commencé comme un projet de recherche en janvier 1996 par Sergey BRIN et Lawerence PAGE en tant qu'étudiants ph. D. à l'université de Standford.

Actuellement, Google est un moteur de recherche à grande échelle[1] qui utilise fortement la structure hypertexte.

[1] Google tire ses origines du mot googol ($10^{100}$) introduit pat le mathématicien américain EDWARD Kasner. Ce terme est choisi pour symboliser sa mission : organiser l'immense volume d'information disponible sur le web.

Google est conçu pour crawler et indexer le web d'une façon efficace et de produire des résultats plus satisfaisants que ceux des systèmes existants.

Les fonctionnalités de Google vont être décrite dans les sections qui suivent.

#### *2.3.2.1 Les caractéristiques du système*

Google possède deux caractéristiques principales (Brin et Page, 1998) :

##### *2.3.2.1.1 Le Page Rank*

Google utilise la structure de liens du web pour calculer un indice de popularité (page rank) qui sert à sélectionner les pages qui répondent le plus pertinemment à une requête et de trier les réponses d'une recherche par mot clés selon l'ordre d'importance.

L'idée du page rank repose sur la considération qui stipule qu'une page et d'autant plus importante qu'elle a un grand nombre de pages populaires la référençant.

Formellement le page rank peut s'énoncer comme suit :

Si on suppose qu'une page web A a $T_1$, $T_2$,…,$T_n$ pages qui la référencent. Soit d[1] un facteur compris entre 0 et 1. Souvent d est mis à 0,85.

C(A) est le nombre de liens sortant de la page A.

$$PR(A) = (1-d) + d * (PR(T_1)/C(T_1) + \ldots + PR(T_n)/C(T_n)$$

Cette formule est récursive et peut être calculée en utilisant des algorithmes itératifs.

Pour plus d'informations sur le page rank, se référer à (Page et al., 1998 ; Langville et Meyer, 2006).

##### *2.3.2.1.2 Textes de liens (ancres)*

La plupart des moteurs de recherche associent le texte de lien[2] à la page sur laquelle il est mentionné. Google l'associe également à la page à laquelle il pointe. Plusieurs avantages découlent de cette astuce :

- L'ancre fournit des descriptions plus précises sur les pages web destination.
- L'ancre est utile pour indexer les documents non textuels (images, programmes, bases de données, …)

Il est à signaler que l'utilisation de l'ancre d'une façon efficace est techniquement difficile, ceci revient à la masse importante de données à traiter. A titre d'exemple, dans (Brin et Page, 1998), lors du crawl de 24 millions de pages, les auteurs ont indexé plus de 259 millions d'ancres.

#### *2.3.2.2 Architecture de Google*

La figure (Figure 15.) présente une vue globale du moteur de recherche Google à travers laquelle on va exposer ses fonctions.

[1] d est un facteur d'amortissement qui limite l'importance de la quantité des votes donnés à une page.

[2] Texte de lien = ancre = texte offshore = texte cliquable ; en anglais anchor text.

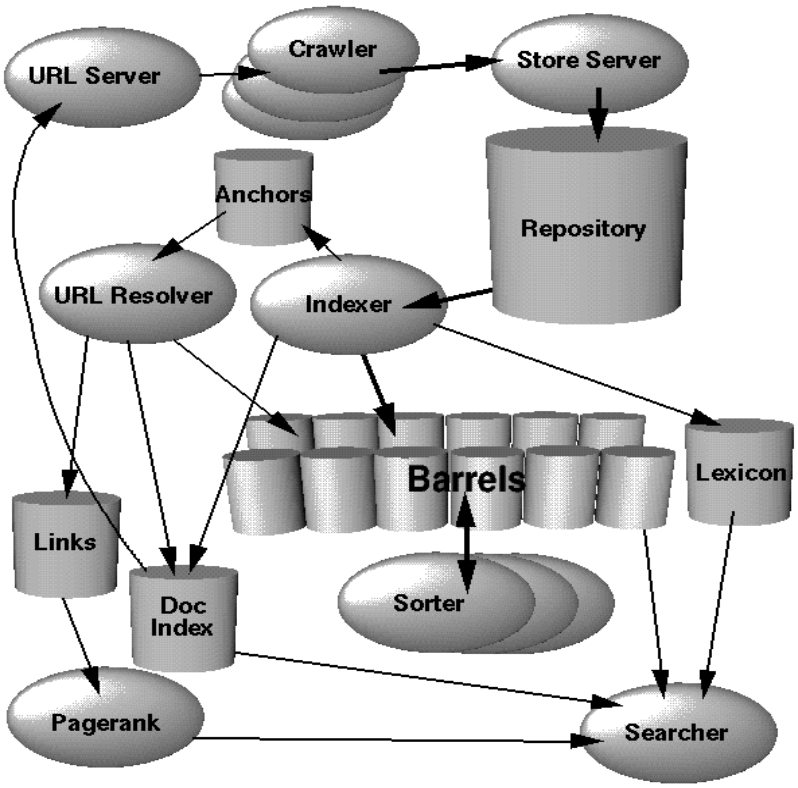



**Figure 15. : Architecture Google de haut niveau**

#### *2.3.2.2.1 Le crawling du Web*

Le crawling du web est réalisé par un système de crawling distribué. En effet, un serveur unique, nomme *URL Server* envoi une liste d'URLs à chercher par le *crawler*[1].

Chaque Crawler garde environ 300 connexions ouvertes à la fois, ceci pour une raison de rapidité : le système peut crawler 100 pages web par seconde en utilisant 4 Crawlers, un équivalent de 600 kilo octets par seconde de données (Brin et Page, 1998).

Les pages trouvées sont ensuite envoyées au *Store Server* qui procède à leur compression et leur stockage dans un entrepôt (repository).

Chaque page est compressé en utilisant Zlib[2] assurant un taux de compression de 3 à 1. A titre d'illustration 53,5 giga octets dans l'entrepôt représentent 147,8 giga octets incompressés.

#### *2.3.2.2.2 Indexation du web*

La fonction d'indexation est effectuée par *l'Indexer* et le *Sorter*. *L'Indexer* réalise trois tâches principales : la lecture de l'entrepôt, la décompression des documents ainsi que leur analyse.

Par la suite, chaque document est converti en un ensemble d'occurrences de mots appelées hits. Les hits enregistrent le mot, sa position dans le document ainsi qu'une approximation de la taille de la police et la majuscule.

*L'indexer* distribue les hits en un ensemble de conteneurs "barrels" créant ainsi un index futur "Forward Index" dont la structure de données ressemble à celle du tableau 3.

[1] Le Crawler et l'URL Server sont implémentés par Python.

[2] Voir RFC 1950 : Zlib Compressed Data Format Specification.

| Document | Mots |
|---|---|
| docID | $wordID_1$, $wordID_2$, … |

**Tableau 3. : Index futur**

Une autre fonction importante de *l'Indexer* est d'analyser tous les liens dans chaque page web et d'enregistrer les informations importantes les concernant dans un fichier ancre "Anchors". Ce fichier contient les informations nécessaires pour déterminer les sources et les destinations de liens ainsi que leurs textes.

Pour sa part, le *Sorter* génère un index inversé (Tableau 4.) pour les titres, les hits des ancres et les "barrels" plein texte.

| Mot | Documents |
|---|---|
| wordID | $docID_1$, $docID_2$, … |

**Tableau 4. : Index inversé**

Il agit sur les "barrels" qui sont triés par docID ; il les retrie par wordID. Noter que pour réaliser cette tache, un certain nombre de problèmes surgissent, notamment ceux liés à la gestion de la mémoire. Pour plus de détail, voir (Brin et Page, 1998).

#### *2.3.2.2.3 La recherche*

Le mécanisme de recherche est illustré à travers les étapes suivantes :

1. Analyser (parse) la requête.
2. Convertir les mots en wordID.
3. Rechercher chaque mot dans la docList dans un conteneur "barrel".
4. Scanner à travers les doclists jusqu'à ce qu'un document comprenne tous les termes de la recherche.
5. Calculer le rang de ce document par rapport à la requête.
6. Et ainsi de suite.
7. Finalement, trier les documents en fonction de leur rang, et retrouver les k premiers.

# Chapitre 3 : Expansion de Requêtes et Thésaurus

## 3.1 Expansion de requêtes

### 3.1.1 Introduction

L'expansion de requêtes, qui est souvent appelée modification de requêtes, signifie la reformulation de requêtes en changeant ses mots clés ou en modifiant leur poids dans un but d'obtenir une meilleure correspondance avec les documents pertinents (Ingwersen et Jarvelin, 2005b).

L'expansion de requêtes a été considérablement étudiée, car la sélection de bons mots clés est difficile mais cruciale pour de meilleurs résultats.

Noter que, souvent, les requêtes n'expriment pas le besoin de l'utilisateur. Une classe plus importante comporte des requêtes ambiguës et/ou vastes (Kraft et Zien, 2004).

Un autre problème est la discordance de mots, en effet, l'auteur d'un document, notamment d'une page web, et le lecteur (utilisateur) n'utilisent pas le même vocabulaire (Chen et al., 2003).

Par ailleurs, les requêtes courtes sont communément utilisées. L'étude Fireball[1] conclue que la longueur moyenne d'une requête était de 1,66 termes et plus de 54% (8.873.001) de requêtes comportaient seulement un terme. L'étude Excite[2] confirme les derniers résultats ; elle conclue que la longueur moyenne d'une requête était de 2,21 termes et 30,81% (15.854) de requêtes comportaient seulement un terme (Jansen et Pooch, 2000).

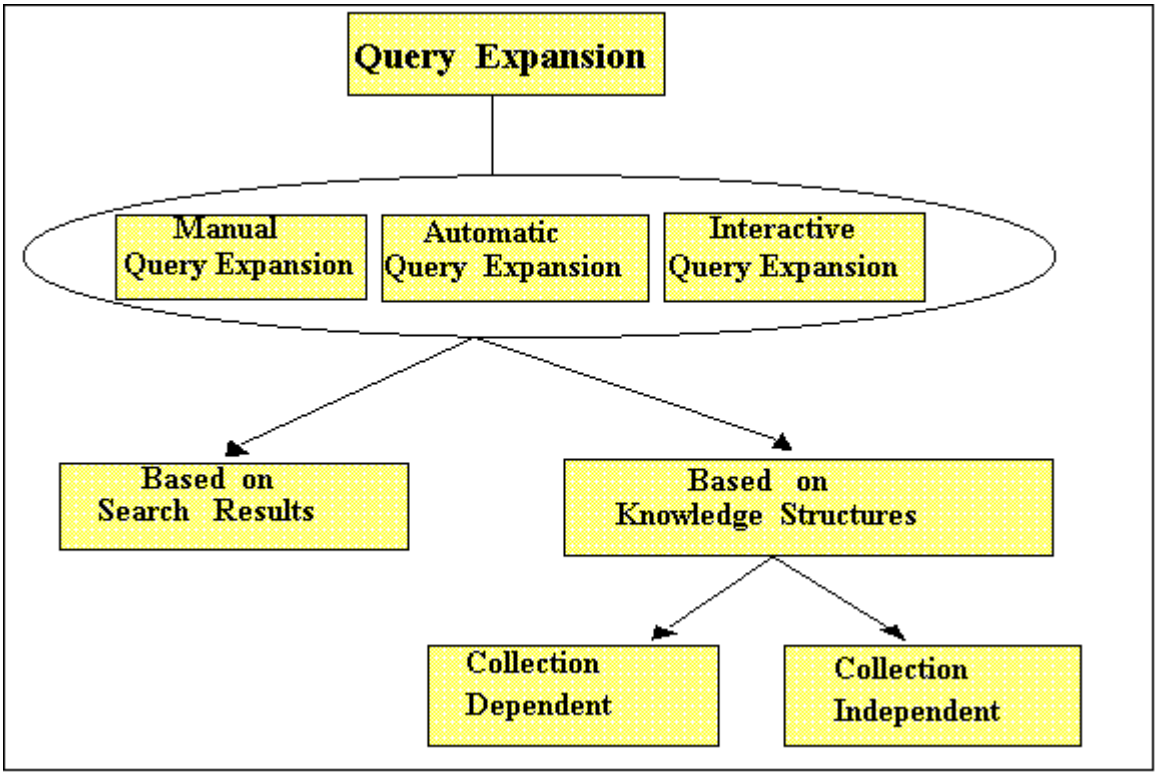



**Figure 16. : Les types d'expansion de requêtes**

[1] Analyse de données depuis le moteur de recherche Fireball (Allemagne) (http://www.Fireball.de/). Le nombre de requêtes dans l'ensemble de données est de 16.252.902.

[2] Analyse de données depuis le moteur de recherche Excite (http://www.excite.com). Le nombre de requêtes dans l'ensemble de données est de 54.573.

Dans les sections qui suivent, on va décrire les techniques d'expansion de requêtes à travers la taxonomie de EFTHIMIADIS (Efthimiadis, 1996) (voir Figure 16)

On va discuter la technique de rétroaction de pertinence en tant que technique d'expansion de requêtes ainsi que l'utilisation des bases de connaissances. Notre étude sera complétée par la mise en évidence des recherches récentes dans le domaine d'expansion de requêtes, en particulier celles inhérentes aux requêtes structurées et au web mining.

### 3.1.2 La rétroaction de pertinence

Nous avons déjà introduit la notion de rétroaction de pertinence lors de l'étude du système SMART (section 2.3.1.2.3). L'idée est qu'après une formulation initiale de la requête, l'utilisateur examine les résultats de la recherche pour identifier les documents pertinents et non pertinents. Les SRIs reformulent automatiquement la requête initiale pour être plus similaire aux documents pertinents et moins similaires aux documents non pertinents.

Dans la rétroaction de pertinence, le nombre optimal de termes à ajouter varie entre quelques termes et plusieurs centaines selon les études (Ingwersen et Jarvelin, 2005b).

L'utilisateur peut être demandé soit de juger la pertinence des résultats et/ou de choisir les termes d'expansion depuis une liste triée de termes.

EFTHIMIADIS enquêtait sur la sélection des termes d'expansion obtenus suite à une rétroaction de pertinence, ceci par des utilisateurs réels (non simulés) (Efthimiadis, 1992). Les utilisateurs sont demandés d'établir la relation des cinq bons termes d'expansion avec les termes originaux. Les résultats de cette étude indiquent que 34% des termes d'expansion n'ont aucune relation avec les termes originaux. Pour le reste (66%), 70% sont des hyponymes[1], 5% sont des hypéronymes et le 25% restant constitue des relations associatives.

Dans l'ensemble, les résultats de l'étude, soulignent une certaine évidence pour l'efficacité d'expansion interactive des requêtes en se basant sur la rétroaction de pertinence.

D'un autre angle de vue, plusieurs études (Ingwersen et Jarvelin, 2005b) confirment que les utilisateurs expérimentés sont en mesure d'exprimer leurs besoins, tandis que les non expérimentés trouvent beaucoup de difficultés de reconnaître le vocabulaire pertinent à cause de leur pauvreté de connaissance de domaine.

HAWKING et al. (Hawking et al., 1997) ont testé l'expansion de requêtes en se basant sur la rétroaction de pertinence et l'identification de concepts dans TREC-5. Dans leur étude, les concepts à rechercher sont manuellement sélectionnés depuis les requêtes et ensuite les termes de recherche sont générés pour chaque concept sans utiliser l'information de la collection. Les requêtes sont ensuite reformulées par les termes obtenus à partir des documents classés résultant de la requête initiale.

Pour effectuer leur test, HAWKING et al. ont utilisé trois méthodes de test de pertinence et cinq types de requêtes ont été construits. Les résultats obtenus affirment que la performance de toutes les requêtes structurées à base de concept est meilleure à celle des requêtes non reformulées ou automatiquement reformulées.

L'expansion de requêtes peut engendrer un effet indésirable connu sous le nom de « query drift » dans la rétroaction de pertinence où les requêtes mal reformulées conduisent à un changement de direction du sujet de la requête initiale.

[1] Hyponymie est la relation d'inclusion entre deux mots dont l'un, l'hyponyme, est plus spécifique que l'autre, l'hypéronyme. Ainsi la fleur est un hypéronyme de tuple.

Dans ce contexte, MITRA et al. (Mitra et al., 1998) ont discuté le problème de « query drift » dans la rétroaction de pertinence sans jugement de pertinence manuel de la part de l'utilisateur[1] et sous l'assomption qui présume que les premiers documents classés sont pertinents. Les auteurs effectuent, d'abord, un reclassement des documents pertinents en appliquant une approche ROCCHIO modifiée. Deux types de reclassement sont étudiés. Les expérimentations ont montré que les deux approches améliorent l'efficacité de la rétroaction de pertinence et confirment qu'une approche automatique efficace de rétroaction de pertinence est faisable.

### 3.1.3 Expansion de requêtes à base de structures de connaissances dépendantes de la collection

Un autre type d'expansion de requêtes peut mettre en jeux des structures de connaissances. Dans cette section, on s'intéresse aux structures de connaissances dépendantes du corpus.

EFTHIMIADIS (Efthimiadis, 1996) cite plusieurs exemples d'un tel type d'expansion :

- Clusterisation de termes.
- Cooccurrence de termes.
- Thésaurus d'association.

Dans notre contexte, on s'intéresse aux thésaurus comme outil d'expansion de requêtes.

JING et CROFT (Jing et Croft, 1994) ont construit automatiquement PhraseFinder[2], un thésaurus d'association dépendant de la collection. Les termes de la requête sont recherchés dans PhraseFinder et les syntagmes retrouvés sont ensuite utilisés pour reformuler la requête initiale. Les résultats de l'étude montrent qu'un thésaurus à base de syntagmes fournit plus de performances qu'un thésaurus à base de mots mais les deux améliorent la performance de la recherche comparés à des requêtes non reformulées, ceci dans une petite collection.

De plus, les requêtes courtes bénéficient beaucoup de l'expansion. Cependant, dans l'ensemble, elles restent moins performantes par rapport aux requêtes longues.

XU et CROFT (Xu et Croft, 1996) essayent de combiner la technique d'expansion utilisant la totalité de la collection (technique globale) et la technique utilisant les résultats de la recherche de la requête initiale (feedback) donnant naissance à une nouvelle technique nommée analyse de contexte local (Local context Analysis, LCA). Les syntagmes sont sélectionnés sur la base de leur cooccurrence avec les termes de recherche dans les n premiers passages de 300 mots. Les syntagmes sont ensuite classés et les 70 premiers seront rajoutés à la requête. La requête reformulée inclut la requête originale en tant que partie à part et les termes d'expansion comme une autre partie.

Les auteurs ont montré que LCA est beaucoup plus efficace que la rétroaction de pertinence et l'expansion de requêtes à base de PhraseFinder uniquement. En effet, la précision moyenne pour les requêtes non reformulées était de 25,2% ; PharseFinder 26,0% ; Rétroaction de pertinence 27,9% et LCA 31,1%.

### 3.1.4 Expansion de requêtes à base de structures de connaissances indépendantes de la collection

EFTHIMIADIS (Efthimiadis, 1996) liste les exemples suivants concernant les structures de connaissances indépendantes de la collection :

- Thésaurus spécifique à un domaine, construit manuellement (MeSH, INSPEC).

[1] Appelé pseudo-relevance feedback = blind relevance feedback (Manning et al., 2007) p. 137.

[2] Accessible à partir des requêtes en langage naturel dans le SRI INQUERY.

- Thésaurus généraliste (Roger's ou WordNet).
- Dictionnaires et lexiques (dictionnaire collin).

Les algorithmes d'expansion de requêtes appliqués à cette catégorie de structures de connaissances sont qualifiés de techniques externes car ils n'utilisent pas les statistiques de la collection pour trouver les termes candidats. Lors de la recherche, les requêtes sont reformulées en consultant simplement les termes connexes dans la structure de connaissance.

VOORHEES (Voorhees, 1994) reformule les requêtes par le biais de WordNet, elle examine l'utilité de l'expansion de requêtes dans les collections TREC. VOORHEES présente une méthode pour utiliser les structures de connaissances pour l'expansion de requêtes :

- Les termes de la requête doivent être désambiguïsés de telle façon qu'ils correspondent à un concept unique de la structure de connaissance.
- Les termes connexes dans la structure de connaissance sont rajoutés à la requête en tant que termes d'expansion.

La collection de test utilisée est une combinaison de collections TREC et d'atelier Tipster. Le modèle VSM est utilisé comme modèle de recherche. Trois types de requêtes et quatre stratégies d'expansion sont mis en œuvre. Les résultats de l'étude insistent sur l'impact de la longueur de requête non reformulée sur l'efficacité de l'expansion.

L'étude faite par KRISTENSEN et JARVELIN (Kristensen et Jarvelin, 1990) est conduite sur une base de données opérationnelle d'archive de journaux et sur un SRI booléen pour tester l'expansion de requêtes par le biais d'un thésaurus lié à un domaine spécifique. Pour se faire, un petit thésaurus est construit.

Les besoins des journalistes sont initialement formulés sous forme de requêtes (originales) qui seront par la suite reformulées par des expressions fournies par le thésaurus. Chaque requête est recherchée dans trois modes différents :

1. Recherche de base : la requête contient uniquement tout les termes fournis par le journaliste.
2. Recherche par synonymie : les termes de la recherche de base sont étendus par disjonctions des synonymes fournis par le thésaurus.
3. Recherche par termes connexes : les synonymes sont étendus par disjonction des termes connexes (quasi-synonymes) fournis par le thésaurus.

Les résultats de recherche de chaque mode sont analysés par rapport à un rappel et une précision relatifs (le rappel moyen de la meilleur expansion de requêtes est mis à 100%).

Le rappel relatif moyen des requêtes non reformulées (type 1) était de 45% alors que pour les requêtes synonymes (type 2) était de 82%. La précision relative moyenne des trois types de requêtes était respectivement 51%, 41% et 33%.

On peut constater que la substitution des synonymes montre une augmentation considérable dans le rappel avec un déclin négligeable dans la précision.

KEKALAINEN (Kristensen, 1993) confirme les derniers résultats, dans une autre étude d'expansion de requêtes. La méthodologie est similaire mais le thésaurus est plus large et des expansions hiérarchiques (hyponymie) sont aussi testées.

Les résultats soulignent un doublement dans le rappel avec seulement 11% de déclin dans la précision pour le meilleur type d'expansion comparé aux requêtes non reformulées.

Dans un autre stade, JARVELIN et al. (Jarvelin et al., 1996) ont développé l'outil *ExpansionTool* (Jarvelin et al., 2001) pour une expansion basée sur un thésaurus et ils ont testé son effet sur des SRIs booléen et probabiliste.

Les résultats de test pour dix requêtes ont montré que la performance de requête peut être contrôlée et améliorée par *ExpansionTool*. L'outil *ExpansionTool* peut être utilisé dans l'amélioration de la paramétrésation et de la qualité de la fonction de correspondance.

### 3.1.5 Expansion de requêtes et les requêtes structurées

Dans la littérature, l'expression requêtes structurées fait référence à des requêtes formulées par des opérateurs booléens à la différence des requêtes à base de langage naturel (sac de mots) (Kekalainen, 1999). Ainsi, la structure peut être comprise comme des relations entre les mots clés de recherche dans une requête. Cette structure est exprimée dans un langage de requête par des opérateurs ou bien des pondérations affectées aux mots clés pour guider le processus de correspondance entre la requête et les documents.

KEKALAINEN a étudié dans sa thèse de PHD (Kekalainen, 1999) l'effet de la complexité, la structure et l'expansion de requêtes sur la performance du processus de recherche en terme de rappel et de précision. Dans notre contexte, on s'intéresse plus aux structures de requêtes.

La figure 17. montre la typologie des structures de requêtes développée dans (Kekalainen, 1999).

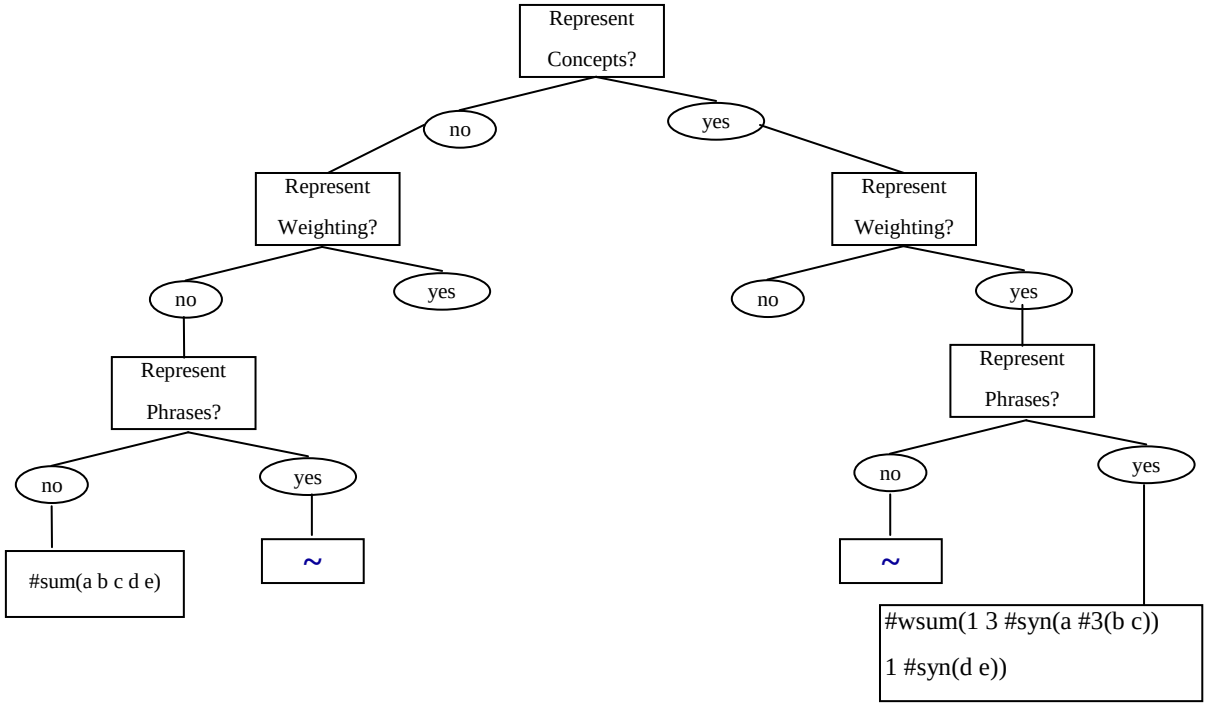



**Figure 17. : Typologie des structures d'une requête**

La figure 17. représente trois décisions faites lors de la formulation d'une requête :

- Représentation explicite des besoins de l'utilisateur à travers les opérateurs du langage de requête.
- La pondération des mots clés : si des mots clés sont considérés avoir plus de valeur dans la représentation des besoins de l'utilisateur, ils doivent être pondérés plus que les autres.
- Le marquage des syntagmes par le biais des opérateurs de proximité.

Les feuilles de la figure 17. présentent, à titre d'illustration, deux requêtes qui ont les même mots clés (a, b, c, d, e) utilisant le langage de requêtes du système InQuery.

Dans le contexte de l'étude, la structure d'une requête fait référence à la structure syntaxique de la requête marquée par les opérateurs du langage de requêtes et les parenthèses. En effet, les structures de requêtes utilisées sont soit faibles (requêtes sans relations discriminatives entre les clés de

recherche, sauf la pondération) ou fortes (différentes relations entre les mot clés de recherche). Plus précisément, les structures de requêtes fortes sont basées sur les facettes[1] ou par l'intersection de concepts.

Pour l'expérimentation, cinq types de structure faible et huit types de structure forte sont testés. Le test implique 300 demandes formulées en 110 requêtes représentant les différentes combinaisons de structure, d'expansion et de complexité. La collection de test est une base de données de 53.893 articles de journaux et le test est effectué dans le SRI InQuery.

Le test a révélé que :

- Quand les requêtes ne sont pas reformulées, il n'y aura pas de grandes différences entre les différents types de structure, ceci sans tenir compte du niveau de complexité.
- Quand les requêtes sont reformulées (l'expansion est faite via un thésaurus) :
    - La performance des requêtes à structure faible chute.
    - La performance des requêtes ayant de meilleures structures fortes s'est améliorée.
    - La meilleure performance est atteinte avec la combinaison des structures à base de facettes, de la haute complexité et de l'expansion la plus large.
- Les clés de recherche combinées par des opérateurs dans une facette sont plus décisives que les facettes combinées par des opérateurs.

Dans l'ensemble, l'étude a enregistré l'impact de la structure de requêtes sur l'expansion de requêtes, ainsi l'utilisation des opérateurs est critique.

Selon une autre étude, (Kekalainen et Jarvelin, 2000) la division sémantique, notamment pour les thesaurus, n'est pas utile pour l'expansion de requêtes. Dans la plupart des cas, les meilleures performances sont obtenues avec les expansions les plus larges incluant toutes les relations sémantiques.

### 3.1.6 Expansion de requêtes sur le web

#### *3.1.6.1 Expansion de requêtes et les moteurs de recherche*

Dans cette section, on va prendre comme exemple trois moteurs de recherche sur le web, à savoir, Yahoo, Google et Ask. Pour chaque moteur, deux techniques d'expansion de requêtes sont présentées :

- La technique « Post-recherche », elle suggère des termes d'expansion soit après l'affichage des résultats de la recherche soit simultanément.
- La technique « Pré-recherche », elle consiste à proposer des termes d'expansion au fur et à mesure que l'utilisateur exprime son besoin informationnel.

Ceci dit, l'expansion de requêtes a trouvé son chemin pour les moteurs de recherche populaires sur le web et est même devenue une des caractéristiques dont les moteurs de recherche reposent pour créer leur propre identité.

Simultanément avec l'affichage des résultats de la recherche, Yahoo[2] affiche une liste cliquable de requêtes reformulées dans une option « Essayez Aussi » (« Also Try ») sous la boite de recherche. Ces requêtes reformulées dérivent depuis des fichiers logs contenant des requêtes déjà effectuées par d'autres utilisateurs.

[1] Une facette est un aspect d'un document ou d'une requête qui se compose d'un ou plusieurs concepts reliés, d'une manière ou d'une autre, par le sens.

[2] http://search.yahoo.com

Yahoo a aussi commencé à tester « Search assist »[1], un nouveau outil qui fournit des thèmes connexes aux termes de la requête, pendant la saisie, pour assister les utilisateurs à travers les termes proposés de trouver la plupart des informations pertinentes. « Search assist » utilise une technologie Yahoo acquise avec AltaVista, appelée Prisma. Prisma combine classification et segmentation pour fournir aux utilisateurs des termes ayant une forte association avec la requête originale.

Google offre l'option « Pages similaires » qui peut être vue comme une forme particulière de rétroaction de pertinence. Par sélection d'une page similaire, on indique que cette page est pertinente pour le besoin de l'utilisateur. Google, alors, sélectionne les termes de cette page pour alimenter la requête originale. Ensuite, Google retrouve des résultats pertinents pour cette nouvelle requête reformulée.

Google est aussi un précurseur dans la technique de « pré-recherche » de l'expansion de requêtes en proposant la technique « Google Suggest »[2] en décembre 2004. la spécificité de cette technique est que les termes suggérés surgissent (pop up) dans la boite de recherche au moment de la saisie de la requête originale. Ces termes ne se basent pas sur l'historique de recherche de l'utilisateur. « Google Suggest » utilise les données d'un ensemble d'utilisateurs afin de classer les raffinements qu'il offre.

Ask Jeeves fournit une option zoom, permettant aux utilisateurs de rétrécir ou d'élargir le champ des résultats de recherche aussi bien comme l'affichage des résultats pour des concepts connexes (Related concepts). Récemment, en Juin 2007, Ask a ajouté une technique (Pré-recherche » d'expansion de requêtes avec le lancement de l'outil Ask3D.

#### *3.1.6.2 la Rétroaction de pertinence sur le web*

Nous avons déjà introduit, dans la section précédente, différentes options offertes par quelques moteurs de recherche telles que : Page similaire, Related pages. Ceci peut être vu comme une forme particulière de rétroaction de pertinence.

Toutefois, la rétroaction de pertinence est moins utilisée dans la RI sur le web. En effet, le moteur de recherche Excite fait l'exception, il fournissait, initialement, complètement la technique de rétroaction de pertinence. Cependant cette option est abandonnée dans le temps, par manque d'utilisation.

Ce manque d'utilisation de la rétroaction de pertinence sur le web peut être du à plusieurs raisons (Manning et al., 2007) :

- sur le web, presque, personne n'utilise des interfaces de recherche avancées, et préfère compléter sa recherche dans une seule itération.
- Il est difficile d'expliquer la rétroaction de pertinence à un utilisateur de niveau moyen.
- La rétroaction de pertinence est, principalement, une stratégie qui tente d'améliorer le rappel, alors que les utilisateurs du web ne s'intéresse pas au rappel.

Ces affirmations sont appuyées par l'étude[3] faite par SPINK et al. (Spink et al., 2000) concernant l'utilisation de la rétroaction de pertinence dans le moteur Excite. Les résultats annoncent, essentiellement, qu'environ 4% des utilisateurs utilisent l'option rétroaction de pertinence, 70% des utilisateurs consultent seulement la première page des résultats sans aucune suite et que les recherches qui utilisent la rétroaction de pertinence sont améliorées par deux tiers (2/3).

[1] Voir l'article intitulé « Yahoo Testing New Query Refinement Tools » écrit par Kevin Newcomb, le 26 juillet 2007 ; http://searchenginewatch.com

[2] http://labs.google.com/suggest/

[3] Le projet d'étude comporte 18.113 utilisateurs (sessions), 51.473 requêtes et 113.793 termes.

### *3.1.6.3 L'utilisation des structures de connaissances*

Comme dans la RI traditionnelle, les structures de connaissances ont été largement utilisée dans l'expansion de requêtes sur le web. Deux exemples vont être exposés, le premier utilise WordNet, tandis que le second utilise un thésaurus d'associations.

GONG et al. (Gong et al., 2005) utilisent une méthode originale dans l'expansion de requêtes sur le web. Ils utilisent WordNet et TSN. WordNet décrit des relations entre les mots dans des dimensions arborescentes d'hyperonymie, d'hyponymie et de synonymie qui ont des impacts différents sur l'expansion de requêtes.

Cependant, WordNet apporte beaucoup de bruit, ceci est du à ces caractéristiques intrinsèques. Par ailleurs, WordNet ne peut pas suivre l'état courant des mots et leurs relations à cause de la propagation explosive du web. Pour surmonter ces problèmes, les auteurs ont crée une collection à base de TSN (Term Semantic Network) selon la cooccurrence des mots de la collection.

Pour mener leur expérimentation, 150.000 pages web sont rassemblées. Après filtrage des icônes, des bannières et des logos, etc., 120.000 images embarquées dans les pages web ont été gardées. Cinq experts humains sont désignés pour définir, d'une façon intellectuelle, les thèmes des images web. Le système utilisé pour exécuter les expérimentations est bien le moteur de recherche d'images sur le web de la faculté des sciences et de technologies, université de Macau, Chine.

Les expérimentations ont révélé que l'expansion combinée, WordNet et TSN, peut fournir un résultat satisfaisant pour la performance des requêtes sur le web.

LEE et al. (Lee et al., 2007) présentent une méthode d'expansion interactive par le biais d'un thésaurus d'associations qui est extrait à partir les pages web référenciées dans les logs de requêtes utilisateurs. La technique d'exploitation des règles d'associations (Association Rule Mining), en particulier, l'exploitation de la corrélation entre termes est appliquée.

Les pages web sélectionnées sont transformées en des ensembles de termes de requêtes qui vont être utilisés pour l'extraction des corrélations entre termes. En conséquence, différents thésaurus d'association concernant différents termes de requêtes sont construits. La figure 18. montre

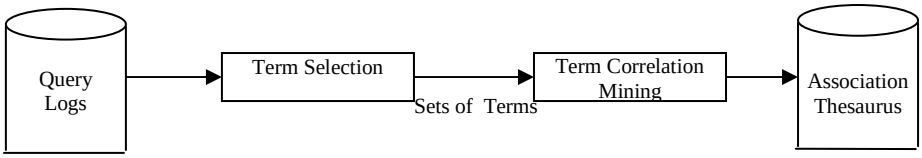



**Figure 18. : Architecture de génération d'un thésaurus d'associations**

La méthode d'expansion proposée combine les termes originaux de la requête spécifiée par l'utilisateur avec les thésaurus correspondants pour offrir des résultats plus précis.

Les résultats expérimentaux ont montré que les taux de précision et de rappel sont améliorés quand la méthode proposée est appliquée.

### *3.1.6.4 L'expansion de requêtes et le web mining*

Dans la section 2.1.3.2, nous avons introduit les défis de la RI sur le web, à savoir, le déluge informationnel, les caractéristiques de la dynamicité et de l'auto-organisation, les hyperliens et la duplication du contenu.

Ces défis rendent le processus d'expansion plus onéreux et demande, ainsi, une mise à niveau des techniques précédemment utilisées dans les petites collections. Une telle mise à niveau émerge sous le vocabulaire web mining (voir section 1.5).

#### *3.1.6.4.1 L'expansion de requêtes et le web structure mining*

Nous rappelons que la plupart des moteurs de recherche se basent uniquement sur le contenu textuel des documents pour identifier les documents pertinents répondant au besoin informationnel des utilisateurs.

Or, les hyperliens jouent un rôle très important pour les pages web, ils fournissent une évaluation indépendante du contenu pour leur popularité.

Le rôle des liens des pages web est similaire au rôle des citations dans la littérature scientifique. En effet, tous les algorithmes basés sur l'étude des liens sont les héritiers, plus ou moins directes, au domaine bibliométrique dont les citations constituent l'objet de l'étude.

Ceci étant dit, dans ce qui suit, on va discuter l'effet de l'intégration des hyperliens dans le processus de RI, en particulier l'expansion de requêtes.

CARRIERE et KAZMAN (Carriere et Kazman, 1997) ont proposé le système WebQuery offrant une méthode de recherche sur le web à base de liens et de contenu dans le but de réordonner les pages web restituées suite à une requête initiale.

La méthode est basée essentiellement sur le dénombrement des liens entrants et sortants d'une page web, ainsi une valeur de popularité peut être calculée pour chaque page exprimant son rang. Il est à noter que l'expansion, ici, concerne les résultats de la requête (hit set) pour avoir un ensemble complet de voisins.

Pour le PageRank proposé par BRIN et PAGE (Brin et Page, 1998), nous l'avons déjà introduit dans la section 2.3.2.1. Néanmoins, il faut noter qu'à l'opposé de la méthode précédente, les liens entrant d'une page ne possèdent pas la même pondération.

Un autre algorithme exploitant les liens des pages web est celui de KLEINBERG[1] (Kleinberg, 1999). L'algorithme HITS (Hyperlink Induced Topic Search) proposé s'appuie sur le principe que toutes les pages web n'ont pas la même importance. Certaines pages comportent un contenu pertinent, elles sont souvent citées par d'autres pages. Ces pages sont appelées autorités (authories), d'autres pointent des pages autorités, elles sont appelées centrales (hubs) ; elles jouent un rôle important, bien qu'elles ne contiennent pas un contenu informationnel.

Contrairement au PageRank, l'algorithme HITS est restreint à l'ensemble constitué par les pages résultat en réponse à une requête initiale. Le processus entier est appelé distillation de thèmes (topic distillation).

Jusqu'à ce point, constatez, que nous avons discuté l'effet des hyperliens sur la performance du processus de RI par reclassement des pages web résultats et nous n'avons pas parler du rôle de l'expansion de requêtes combinée à une telle technique.

BHARAT et HENZINGER (Bharat et Henzinger, 1998) ont mené une étude pour avoir des solutions alternatives à l'algorithme de KLEINBERG pour combiner l'analyse de liens avec le contenu. Leur travail est caractérisé par deux spécificités :

1. ils ont utilisé les mots clés pour déterminer la pertinence des liens, en utilisant les textes intégraux des documents.
2. ils ont procédé à l'expansion de la requête originale avec les mots clés des premiers documents restitués par le système pour produire des meilleurs ensembles de départ et par conséquence les pondérations des liens calculées par rapport à cette requête étendue seront optimales.

[1] Jon KLEINBERG est titulaire d'un PhD en informatique obtenu au MIT en 1996. Il a développé l'algorithme HITS au laboratoire Almaden d'IBM.

Lors de l'expérimentation, huit algorithmes ont été évalués. 28 requêtes précédemment utilisées pour l'évaluation de l'algorithme de KLEINBERG sont prises comme base d'évaluation. Dans chaque cas, le meilleur algorithme améliore la précision par rapport à celle de KLEINBERG par au moins 45%.

KRAFT et ZIEN (Kraft et Zien, 2004) ont proposé une nouvelle méthode pour l'expansion de requêtes. Ils utilisent l'exploration des textes de liens (Anchor text mining) qui possèdent les propriétés suivantes :

- les textes de liens fournissent une description plus exacte que les pages web elles mêmes.
- Les textes de liens existent pour les pages qui ne peuvent pas être indexées par les moteurs à base de texte, par exemple les pages qui contiennent des images, des logos, etc.

L'algorithme d'expansion de requêtes examine, d'abord, la requête originale de l'utilisateur pour trouver, ensuite, tous les textes de liens similaires qui jouent le rôle de termes d'expansion. En effet, ils utilisent une méthode d'agrégation de classement médiane (median rank aggregation) qui s'est montré comme un algorithme de classement efficace et flexible (Fagin et al., 2003).

Les avantages principaux des textes de liens vis-à-vis l'expansion de requêtes peuvent être résumés comme suit :

- L'ordre de grandeur des données des textes de liens est nettement inférieur à celui de toute la collection. Ainsi, on peut économiser le temps de traitement des données.
- Les pages web pointées par un grand nombre de textes de liens tendent à avoir un bon classement, en se basant sur l'analyse des liens (Link analysis). Ainsi, l'utilisation de ces textes de liens pour l'expansion de requêtes conduit à des résultats qui sont pertinents pour la collection.

La collection de données utilisée découle du crawl de l'intranet IBM, elle comporte plus de 33 millions de textes de liens (2,8 giga octets) et plus de 4,0 millions de documents (60 giga octets de données de documents analysées).

Pour l'évaluation, les auteurs ont construit 29 requêtes ; et approximativement 5.000 requêtes reformulées, par l'algorithme d'exploration des textes de liens, sont manuellement classées.

Les résultats expérimentaux montrent que l'utilisation des liens de textes comme base pour l'expansion de requêtes produit des suggestions d'expansion de haute qualité qui sont significativement meilleures comparées aux expansions dérivées de l'utilisation du contenu de documents.

En outre, l'étude révèle que l'expansion avec les textes de liens peut être utilisée pour améliorer l'expansion à base de logs de requêtes. Cette dernière technique fait l'objet des discussions de la section suivante.

#### *3.1.6.4.2 L'expansion de requêtes et le web usage mining*

Les techniques d'expansion de requêtes précédemment étudiées y compris celles qui reposent sur les hyperliens des documents sur le web ignorent une caractéristique spécifique de la RI sur le web. C'est bien la disponibilité d'une grande quantité d'information inhérente aux interactions des utilisateurs enregistrées dans des fichiers logs de requêtes (query logs).

En effet, la technique d'expansion de requêtes analysant les fichiers logs est une application du web usage mining appelée aussi web log mining.

Dans la littérature (Hoi, 2006), on discute le sujet d'exploration des logs de requêtes sur le web selon deux aspects : le premier aspect concerne l'expansion de requêtes avec les données par-cliques (click-through data), tandis que le second aspect s'intéresse à l'analyse de la nature temporelle des données par-cliques.

- **Expansion de requêtes avec les données par-cliques**

Cette technique est motivée par les techniques de rétroaction de pertinence qui modifient les requêtes en se basant sur les jugements de pertinence des utilisateurs. Cependant, il est difficile d'obtenir des rétroactions suffisantes car les utilisateurs sont peu enthousiastes pour fournir de telles informations de rétroaction (Hoi, 2006).

L'analyse des données par-cliques peut constituer une alternative pour surmonter ce problème. Pour l'expansion de requêtes, elle peut être étudiée selon deux approches :

- **Première approche** : expansion de requêtes avec les requêtes similaires en se basant sur l'idée qui stipule que la similarité entre requêtes peut être déduite depuis les documents communs dont les utilisateurs ont visités après l'émission de leurs requêtes.

  A titre d'illustration, on va discuter le travail de BEEFERMAN et BERGER (Beeferman et Berger, 2000). Les auteurs ont introduit une technique pour explorer une collection de transactions des utilisateurs avec un moteur de recherche sur Internet pour découvrir des clusters de requêtes similaires et des URLs similaires.

  Chaque enregistrement des données par-cliques est composé d'une requête utilisateur accompagnée de l'URL que l'utilisateur a sélectionné parmi les URLs candidates offertes par le moteur de recherche.

  L'ensemble des données est représenté sous forme d'un graphe bipartite. Les sommets d'un côté correspondent aux requêtes et de l'autre côté aux URLs. La relation de cooccurrence entre requêtes et pages (URLs) correspond aux arcs entre les sommets.

  Une caractéristique de l'algorithme de clusterisation itératif agglomératif (Listing 3) est l'indépendance du contenu (content-independant), l'algorithme n'utilise pas le contenu actuel des requêtes et des URLs.

```
Entrée : un graphe bipartite G (les sommets de requêtes en blanc et les
     sommets URLs en noir)
Sortie ; un nouveau graphe G' (chaque sommet blan (noir) de G' correspond à un
     ou plusieurs sommets blancs (noirs) de G)
1- marquer tout les paires de sommets blans dans G selon la fonction de
     similarité σ.
2- fusionner les deux sommets w_i, w_j dans laquelles σ(w_i, w_j) est importante.
3- marquer tout les paires de sommets noirs dans G selon la fonction de
     similarité σ.
4- fusionner les deux sommets b_i, b_j dans laquelles σ(b_i, b_j) est importante.
5- Aller à l'étape 1, sauf si une condition d'arrêt est vérifiée.
```

**Listing 3. : Algorithme de clusterisation itératif agglomératif**

La fonction de similarité σ est donnée comme suit

$$\sigma(x,y) = \begin{cases} \dfrac{N(x) \bigcap N(y)}{N(x) \bigcup N(y)} & si \quad \left| N(x) \bigcup N(y \right| \succ 0 \\ 0 & \text{sinon} \end{cases}$$

Où

- N(x) est l'ensemble des sommets voisins de x.
- y est similaire à x si N(x) et N(y) possèdent un recouvrement important.

- **Seconde approche :** expansion de requêtes avec les termes similaires dans les documents déjà visités.

CUI et al. (Cui et al., 2002) se sont basés sur l'idée d'extraire des corrélations probabilistes entre les termes de la requête et les termes des documents par analyse des fichiers logs des requêtes. Ces corrélations sont ensuite utilisées pour choisir les termes d'expansion de haute qualité pour la nouvelle requête (Figure 19.).

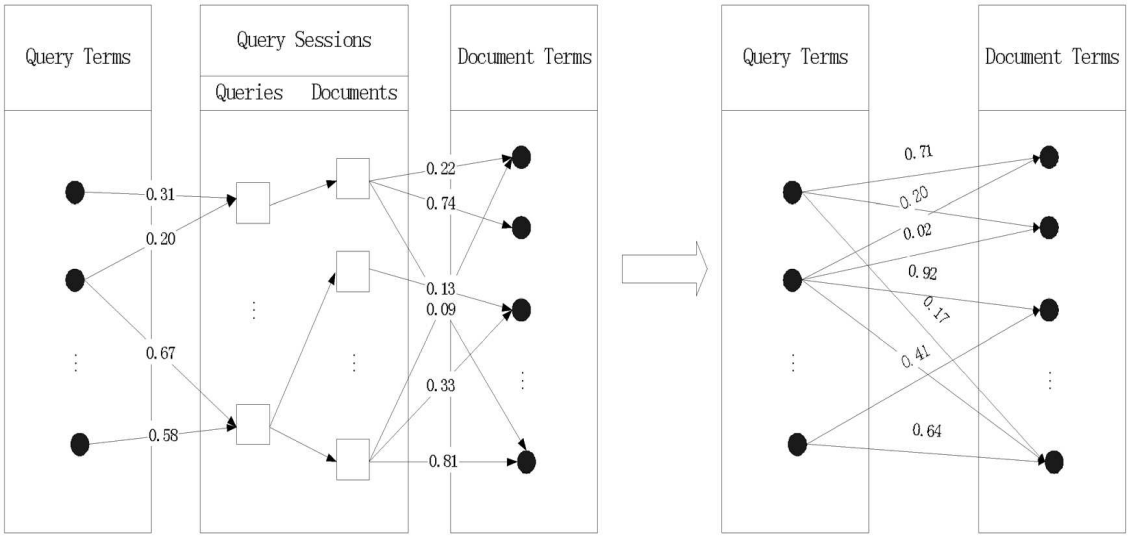



**Figure 19. : Etablissement des corrélations entre les termes d'une requête et les termes des documents via les sessions de requêtes**

Plusieurs sessions de requêtes peuvent être extraites à partir des logs de requêtes. Elles sont définit comme suit :

Session := <query text>[click document]*

Chaque session contient une requête et un ensemble de données similaires à l'environnement réel du web. Des séries d'expérimentations ont montré que la méthode à base de logs de requêtes peut atteindre des améliorations de performances substantielles, même par rapport à la méthode d'analyse de contexte local (LCA) qui est une des méthodes d'expansion de requêtes les plus efficaces.

- **Expansion de requêtes avec analyse de la nature temporelle des données par-cliques**

De la même manière qu'un texte n'est pas un sac de mots, une session d'interactions n'est pas seulement une collection de pages.

Récemment, et dans cette optique, plusieurs études ont commencé d'analyser la nature temporelle et dynamique des données par-cliques.
SHEN et al (Shen et al., 2005) ont mené une étude sur la dynamique des thèmes pour les pages visitées par un groupe d'utilisateurs. Ils ont construit des modèles probabilistes pour prévoir des transitions en matière de thèmes dans le futur, en se basant sur les probabilités de transitions de Markov et marginales.
Pour modéliser le comportement de recherche des utilisateurs, les auteurs ont analysé les fichiers logs des requêtes, les URLs visitées ainsi que les catégories thématiques associées à chaque URL. En effet, ils ont analysé la nature et l'uniformité des thèmes des URL que l'utilisateur a visité au fil du temps.
L'ensemble de données utilisées pour tester les méthodes proposées se compose d'un échantillon du trafic du moteur de recherche MSN. Les résultats d'expérimentations ont formulé l'idée qui prétend que la compréhension de la nature dynamique des recherches thématiques au fil du temps permettra une meilleure personnalisation de recherche et qui peut être considérée comme une forme particulière d'expansion de requêtes dite adaptée.

VLACHOS et al. (Vlachos et al., 2004) ont proposé l'identification des requêtes similaires en se basant sur les modèles de demandes historiques qui sont représentés sous forme de séries chronologiques utilisant les meilleurs coefficients de Fourier et l'énergie des composants omis.
Ils ont présenté plusieurs méthodes pour l'extraction de connaissances à partir des logs de requêtes du moteur de recherche MSN.
Les expérimentations ont montré que les méthodes proposées peuvent surmonter les problèmes des limites inférieurs et supérieurs dans la distance euclidienne. Ainsi, l'information sera soigneusement sélectionnée et permettra, alors, l'utilisation d'une requête complète.
CHIEN et IMMORLICA (Chien et Immorlica, 2005) ont examiné l'idée de trouver les requêtes sémantiquement connexes en se basant sur les corrélations temporelles.
L'ensemble de données utilisé comme base de test est les logs de requêtes du marché U.S. du moteur de recherche MSN.
Les expérimentations ont révélé que la technique proposée peut découvrir un éventail de requêtes sémantiquement similaires. Ainsi, une méthode est développée pour filtrer les requêtes les plus fortement corrélées pour avoir des résultats plus pertinents.

## 3.2 Les Thésaurus

Nous avons déjà introduit la notion de thésaurus dans la section 1.4.2, dans laquelle on a essayé de clarifier la notion de thésaurus et le distinguer des autres bases de connaissances.

Dans ce qui suit on s'intéresse à la présentation de l'évolution des thésaurus depuis le format papier passant par le format électronique et jusqu'au format sur l'Internet. Ensuite une description des approches de construction automatique des thésaurus sera exposée incluant celles sur le web.

### 3.2.1 Les thésaurus : une vue générale

On peut commencer par 1852, la date de conception du fameux thésaurus par Peter Mark Roget, intitulé « The thesaurus of English Words and Phrases ». Ce thésaurus n'est pas classé par ordre alphabétique mais, systématiquement, suivant les concepts exprimés par les mots. L'objet étant de trouver le(s) mot(s) qui peuvent correspondre à une idée.

C'est encore tôt, un siècle plus tard, pour que la notion de terme soit appliquée pour les listes de vocabulaire utilisées dans la RI (Aitchison et Clarke, 2004).

Dans le contexte de la RI, il y a un accord que le mot thésaurus est utilisé pour la première fois, en 1957, par Luhn H.P. d'IBM (Luhn, 1957).

Le premier thésaurus, actuellement utilisé, pour contrôler le vocabulaire d'un SRI est développé par l'organisation Dupont de Nemours, aux Etats-Unis. Il est utilisé seulement pour un but d'indexation.

Du point de vue disponibilité, les thésaurus les plus largement disponibles sont le « Thesaurus of ASTIA Descriptors » réalisé par le département de la défense des Etats-Unis, en 1960 et le « Chemical Engineering Thesaurus » publié par « the American Institue of Chemical Engineers ».

Les thésaurus élaborés, dans les années 50, utilisent les systèmes unitermes. Ces termes singuliers, non contrôlés, sont extraits à partir des textes des documents. Dans ces circonstances, des difficultés surviennent, car les termes composés d'un seul mot possèdent différentes significations selon le contexte, l'application, etc. Par manque de qualificateurs, ces termes sont incapables d'exprimer ces différences spécifiques. Ainsi, les unitermes sont remplacés par un vocabulaire contenant un nombre suffisant de termes composés (Aitchison et Clarke, 2004).

Dans cette optique, ces premiers thésaurus commencent à contrôler les synonymes, les homographes[1] et de montrer les relation hiérarchiques et associations entre les termes.

Il est à signaler que tout au long de cette période, les thésaurus listent les termes descripteurs et non descripteurs dans un ordre alphabétique montrant sous chaque descripteur les synonymes, les termes génériques, spécifiques et associés. En ce qui concerne la présentation des sujets, si elle existe, elle constitue une partie subordonnée du thésaurus.

En 1967, un format standard s'est émergé, quand TEST (Thesaurus of Engineering and Scientific Terms) est publié, remplaçant ainsi le premier thésaurus « Thesaurus of Engineering Terms" de l'EJC (Engineers Joint Council).

Le thesaurus alphabétique TEST possédait déjà la plupart des caractéristiques présentes dans le thésaurus standard des 30 années à venir, incluant les relations d'équivalence, hiérarchique et association encore utilisées de nos jours.

#### *3.2.1.1 Une approche systématique de classification*

L'approche systématique a été largement influencée par les travaux du mathématicien bibliothécaire, Indien, Ranganathan S.R. portant sur l'analyse des facettes.

Le groupe CRG (Classification Research Group) a travaillé sur les schémas de classification à facettes à la fin des années 50. Ces travaux ont donné naissance à un système de classification à facette pour la bibliothèque de « English Electric Company ». Quelques années plus tard, le système a été assisté par un thésaurus alphabétique dont les termes et les relations dérivent de ce système de classification.

Ensemble, en 1964, les deux parties, le système de classification et le thésaurus ont formé le Thesaurofacet, le premier thésaurus utilisant l'analyse des facettes.

Thesaurofacet est suivi par une série de thésaurus à base de classification à facette. A titre d'exemple, la première édition de UNESCO Thesaurus[2], BSI Root Thesaurus[3] et International Thesaurus of Refuge Terms. Dans ce style, le thésaurus est subdivisé par thèmes, les concepts sont regroupés par discipline.

Dans un autre style de thésaurus à base de facette, le domaine de connaissance couvert par le thésaurus est divisé d'abord en des catégories fondamentales (entités, actions, espaces, temps, etc.) et non par champ sujet ou discipline. Le « Construction Industry Thesaurus » est un des premiers exemples de ce style.

Art & Architecture Thesaurus[4] est un autre exemple. L'AAT se compose de 125000 termes organisé en sept facettes (concepts associés, attributs physiques, styles et périodes, agents, activités, matériaux, objet).

#### *3.2.1.2 Thésaurus et Informatique*

Dans les premiers temps, les thésaurus sont compilés manuellement, c'est une opération lente et frustrante. Ce n'est qu'à la fin des années 70 que la compilation des thésaurus assistée par ordinateur était devenu plus courante.

Passant par l'ère à base de papier, des cartes perforées, les bases de données des années 70 et 80, les changements environnementaux caractérisés par la prolifération de l'Internet imposent beaucoup de défis.

---

[1] Caractère de terme ayant la même forme graphique et des sens distincts. De tels termes sont dits homographes.

[2] http://www2.ulcc.ac.uk/unesco/

[3] http://www.mdocassn.demon_co.uk/descbib.htm

[4] Crée par J. Paul GETTY TRUST, http://www.getty.edu/research/conducting_research/vocabularies/aat/

En effet les utilisateurs, de différents niveaux, viennent avec l'espérance de pouvoir satisfaire n'importe quel besoin informationnel. A cela, il faut rajouter le manque d'un médiateur entre l'utilisateur et les ressources disponibles sur le web.

La confrontation de ces défis a mené à deux tendances principales de remèdes (Aitchison et Clarke, 2004) :

- Des adaptations qui peuvent rendre un vocabulaire contrôlé plus rapide, plus simple et plus intuitif à utiliser.
- L'interopérabilité des systèmes implique une conception de vocabulaire facile à intégrer dans les applications telles que les systèmes de gestion de contenu, moteurs de recherche et portails

#### *3.2.1.3 Les normes de construction des thésaurus*

Nous avons déjà mentionné que les gens avaient développé les thésaurus depuis longtemps. En effet, la majorité des problèmes inhérents à la construction des thésaurus avaient été déjà identifiés et résolus en 1967 (Rosenfeld et Morville, 2002).

Les normes et directives de construction des thésaurus sont influencées par deux éléments :

- Les règles et convention du thésaurus TEST.
- Les contributions de Derek Austin avec Dale dans la rédaction des directives de l'UNESCO pour l'établissement et le développement des thésaurus monolingues.

La première édition de la norme internationale pour les thésaurus monolingues était publiée en 1974.

En 1985, la norme ISO 5964 (principes directeurs pour l'établissement et le développement des thésaurus multilingues) a vu le jour. C'est l'extension de la norme liée aux thésaurus monolingues. Elle contient des lignes directrices concernant les degrés d'équivalence et de non équivalences de termes, les équivalences de un à plusieurs, etc.

La norme ISO 2788 (principes directeurs pour l'établissement et le développement des thésaurus monolingues) a été publiée en 1986. Elle comprend les lignes directrices concernant les références, les définitions, les abréviations, le contrôle du vocabulaire, les termes d'indexation, les termes composés, les relation fondamentales, la présentation et la gestion.

Soulignez que les normes internationales des thésaurus ont été adoptées par plusieurs pays y compris le Royaume-Uni, la France et l'Allemagne, en tant que leurs propres normes nationales.

Ainsi, dans le Royaume-Uni, par exemple, il existe les normes nationales BS 5723 et BS 6723 respectivement identique à l'ISO 2788 et ISO 5964.

La norme des Etats-Unis pour les thésaurus monolingues est ANSI/NISO Z39.19 : 1993. Elle est largement comparable avec l'ISO 2788 :1986. Il est à noter qu'il n'existe pas une norme américaine concernant les thésaurus multilingues.

Pour la France, la norme Z47-100 : 1981 est dédiée aux principes directeurs pour l'établissement des thésaurus monolingues. Tandis que la norme Z47-101 :1990 s'intéresse aux thésaurus multilingues.

Pour les pays Arabes, dans les années 80, ont connu une activité considérable pour la construction des thésaurus mais ils reposent principalement sur des thésaurus étrangers. Ces efforts sont

accompagnés par la publication de deux normes, [1]أسمو578 et [2]أسمو795 compatibles respectivement avec les normes ISO 2788 et ISO 5964.

Pour plus de détail sur les thésaurus arabes, le rapport (Tamim, 2001) présente les efforts de construction des thésaurus dans les pays arabes et comprend une étude critique de quelques thésaurus.

Par ailleurs, l'émergence du web a conduit à l'étude de l'opportunité et de la faisabilité du développement d'un standard pour les thésaurus électroniques.

Un atelier sur le sujet s'est tenu en 1999 a Washington et sponsorisé par NISO (National Information Standards Organisation), APA (American Psychological Association), ASI (American Society of Indexers) et ALCTS (Association for Library and Technical Service).

L'accent a été mis sur l'interopérabilité des applications au niveau de contenu et non pas sur méthodes de construction et d'affichage des systèmes d'organisation de connaissances. Dans cette optique, XML et RDF ont été avancés comme formats qui peuvent être très utiles pour différents outils de navigation sur le web.

En parallèle des travaux de cet atelier, le NKOS (Networked Knowledge Organisation Systems/Services) a clairement démontré le besoin et l'importance des thésaurus comme outil d'organisation de connaissance.

Comme fruits de ces travaux, et sous l'égide du W3C, SKOS (Simple Knowledge Organisation System) a été développé. En effet SKOS est un vocabulaire RDF permettant de définir (selon le formalisme RDF) des systèmes d'organisation de connaissance tels que les thésaurus, les classifications, les taxonomies, etc. Pour une description plus détaillée du langage SKOS voir (Miles et al., 2005).

### 3.2.2 Les thésaurus sur le web

#### *3.2.2.1 Pourquoi les thésaurus sur le web?*

Avec l'apparition du web, une activité intense est sensée chez les développeurs des thésaurus pour la mise en disponibilité de leur thésaurus sur le web avec la conviction que les structures sémantiques fournées par les thésaurus peuvent jouer un rôle dans l'organisation et le repérage de l'information sur ce nouveau environnement. Cette conviction est liée, essentiellement, aux points suivants (Shiri et Revie, 2000) :

- La croissance colossale des ressources d'information exigeant une identification meilleure de leurs sujets.
- La migration des ressources d'information traditionnelles vers le web nécessite des approches consistantes.
- Un besoin urgent pour la description et la découverte à travers la réutilisation des outils de gestion de l'information existants tel que le vocabulaire contrôlé.
- Les problèmes associés à la qualité de repérage des informations non structurées sur le web.
- Le besoin de fournir aux utilisateurs des structures de connaissances telles que les thésaurus pour un accès rapide et simple à une information structurée.

[1] إرشادات لإعداد و تطوير المكانز أحادية اللغة

[2] إرشادات لإعداد و تطوير المكانز متعددة اللغة

#### *3.2.2.2 Types des thésaurus sur le web*

Les producteurs de logiciels thésaurus ont utilisé les langages HTML, Java et XML pour publier leurs thésaurus sur le web. Dans (Shiri et Revie, 2000), Davies suggère que les thésaurus peuvent être publiés sur le web sous forme statique ou dynamique. Le choix entre ces deux formes influence le format et l'organisation des thésaurus.

Les thésaurus à base de web peuvent être classés, selon leurs formats et leurs structures, comme suit :

- Des thésaurus dans un format texte statique (ASFA[1], Aquatic Sciences and Fisheries Abstracts).
- Des thésaurus dans un format HTML, ils demeurent toujours statiques (infoterm).
- Des thésaurus dans un format HTML dynamique, format avec des hyperliens permettant une navigation libre (MeSH).
- Des thésaurus avec des interfaces graphiques visuelles (Plumb Design Visual Thesaurus).
- Des thésaurus au format XML (Virtual HyperGlossary, MeSH en format XML).
- Des thésaurus en format RDF (AGROVOC[2]).

Par ailleurs, les thésaurus à base de web peuvent aussi être scindés en deux types généraux selon leur fonctionnalité et leur utilisation :

- Les thésaurus autonomes qui ne sont pas une partie d'un système d'information (le thésaurus ASIS de Librarianship and Information Science).
- Les thésaurus qui sont complètement intégrés dans des bases de données ou des SRIs (le thésaurus qui est totalement adopté et intégré dans la base de données ERIC).

### 3.2.3 Utilisation et construction des thésaurus

#### *3.2.3.1 Utilisation des thésaurus*

On s'intéresse, dans cette section, à l'utilisation des thésaurus dans le domaine de RI. En effet, au cours des années, l'utilisation des thésaurus a connu plusieurs changements. D'un outil d'aide à la recherche, dans les premier temps, à un outil d'indexation. Dans les années 80, les thésaurus ont tendu à abandonner leur rôle comme une liste d'autorité d'indexation, devenant encore un outil de recherche (Nielsen, 2004).

Aitchison et Bawden (Nielsen, 2004) ont divisé les thésaurus de RI en quatre possibilités, ainsi couvrant les deux approches d'indexation et de recherche :

- Thésaurus utilisé à l'indexation et à la recherche.
- Thésaurus utilisé à l'indexation mais pas à la recherche.
- Thésaurus utilisé à la recherche mais pas à l'indexation.
- Thésaurus utilisé ni dans l'un, ni dans l'autre cas.

Les thésaurus utilisés à l'indexation servent de normes pour les indexeurs, ils permettent, donc, une pré coordination des termes d'indexation (Bruandet et Chevallet, 2003). Par contre, à la recherche, l'usage le plus courant est celui de l'expansion de requêtes de l'utilisateur lors du processus

[1] Thésauus des résumés des sciences aquatiques et halieutiques édité par la FAO (Food and Agriculture Organisation), disponible sur le site http://www4.org/asfa/asfa.htm.

[2] AGROVOC est le thésaurus agricole multilingue de la FAO, http://www.fao.org/agrovoc.

d'interrogation. On peut dire qu'il s'agit d'une post coordination des termes d'indexation (Bruandet et Chevallet, 2003). .

Oakes (Oakes, 2007) a présenté les principales utilisations des thésaurus dans la RI, proposées par Foskett :

1. Les thésaurus fournissent une carte d'un domaine de connaissance, montrant des concepts et des relations.
2. Les thésaurus fournissent un vocabulaire standard pour une indexation consistante.
3. Les thésaurus assistent les utilisateurs dans la localisation des termes pour une formulation appropriée des requêtes.
4. Les thésaurus aident à s'assurer que seulement un terme d'un ensemble de synonymes est employé pour l'indexation et la recherche.
5. Les thésaurus fournissent des hiérarchies classifiées pour élargir et rétrécir une recherche.

#### *3.2.3.2 Le processus de construction des thésaurus*

Le processus de construction des thésaurus est traditionnellement divisé en trois sous processus :

- Collection des concepts et des termes
- Formation et définition des concepts et des termes
- Organisation des concepts et des termes.

##### *3.2.3.2.1 Collection des concepts et des termes*

La construction d'un thésaurus passe par le rassemblement du matériel terminologique nécessaire. En effet, un thésaurus peut être compilé en employant deux méthodes :

- La méthode déductive (synthétique) : on essaye de collecter tous les termes qui couvrent un domaine depuis des index, des fichiers, des dictionnaires, etc.
- La méthode inductive (analytique) : au fur et à mesure que les documents sont traités, on note les termes d'indexation et leur fréquence d'utilisation. On établit alors la liste sémantique et la liste alphabétique.

##### *3.2.3.2.2 Formation et définition des concepts*

Les termes et les concepts sont contrôlés de diverses façons, dans les thésaurus. La forme des termes peut être contrôlée selon plusieurs facteurs : grammaticale, orthographique, forme singulière et plurielle, abréviation ou forme composée des termes.

Le sens des concepts est contrôlé et un choix est effectué entre plusieurs synonymes disponibles pour exprimer le même concept. Lorsque les termes peuvent prêter à confusion, l'ajout de notices descriptives[1] ou notes d'application permettant de définir le sens dans lequel il faut utiliser les termes. Par contre, dans le cas des homographes, les définitions doivent être mentionnées.

Dans ce contexte, il faut retenir que les normes et directives distinguent entre note d'application et définition. La première fournit les informations à l'indexeur de la façon à utiliser les termes pour l'indexation, tandis que, la seconde, peut être considérée comme des descripteurs supplémentaires précisant le sens d'un terme.

[1] En anglais Scope Note = SN

#### *3.2.3.2.3 Organisation des concepts et des termes*

La capacité de distinguer et de montrer la relation entre termes et concepts est une fonctionnalité intrinsèque d'un thésaurus (Nielsen, 2004). La structuration et la classification des termes signalent le sens et l'usage du vocabulaire.

En général, trois types de relations inter termes sont utilisés :

- La relation d'équivalence : La relation d'équivalence est une relation entre les termes préférés et les termes non préférés lorsque deux ou plusieurs termes se réfèrent au même concept (Nielsen, 2004). La relation d'équivalence couvre les types de relations suivants :
  - Synonymes
  - Variantes lexicales
  - Quasi synonymes

  Dans un thésaurus de recherche, les termes de recherche alternatifs représentant le même concept sont affichés pour aider l'utilisateur à choisir plus de termes spécifiques et précis.

  Dans un système d'expansion automatique, l'ensemble de synonymes est utilisé automatiquement pour étendre la recherche.

- La relation hiérarchique : La relation hiérarchique montre la relation de supériorité et de subordination où le terme supérieur représente la classe et le terme subordonné représente son membre ou sa partie.

  On utilise les capitales TG (Terme Générique), en anglais BT (Broader Term), pour définir un descripteur de sens immédiatement plus large.

  Pour les termes subordonnés, on les précise par TS (Terme Spécifique), en anglais NT (Narrow Term).

  Les notations TGn et TSn (n=1, 2, …) représente le niveau de hiérarchisation, respectivement, générique et spécifique.

  Il est à noter que les premiers thésaurus n'admettaient qu'un seul rattachement de niveau générique ou spécifique.

- La relation associative : Une relation associative indique une relation de proximité sémantique entre les termes. A titre d'exemple, on trouve les types de relations suivantes : cause et effet, action et agent, action et produit.

  Ces relations s'expriment sous la forme VA (Voir Aussi) ou TA (Terme Associé). La notation anglaise est RT (Related Term).

  Soulignez que la relation associative est la plus difficile à définir et à maintenir (Nielsen, 2004). Cependant, les relations associatives fournissent à l'utilisateur du thésaurus de précieuses informations montrant le sens, et en particulier, le contexte de l'utilisation du terme analysé.

#### *3.2.3.3 Construction automatique des thésaurus*

Les méthodes de construction automatique des thésaurus peuvent être classifiés en deux catégories (Bruandet et Chevallet, 2003). :

- Méthodes statistiques
- Méthodes linguistiques

#### *3.2.3.3.1 Les méthodes statistiques*

Les méthodes statistiques se basent sur le principe de contexte documentaire qui se fonde sur l'hypothèse qui stipule que les termes qui partagent le même contexte sont sémantiquement proches. C'est dans cette optique que Van Rijsbergen a proposé son hypothèse d'association (Bruandet et Chevallet, 2003). : « Si un terme d'indexation est bon pour distinguer les documents pertinents des documents non pertinents, alors tout terme proche est aussi possiblement un bon choix »

Le contexte documentaire fait référence aux termes partageant un ensemble de documents voisins. Il peut également être réduit à un paragraphe ou à une phrase.

Les mesures les plus rencontrées pour le calcul de la similarité entre deux termes sont les suivants (Van Rijsbergen, 1979) :

$$\cos(X,Y)=\frac{|X\cap Y|}{\sqrt{|X|\cap|Y|}}\ ; \qquad dice(X,Y)=2\frac{|X\cap Y|}{|X|+|Y|}\ ; \qquad jaccard(X,Y)=\frac{|X\cap Y|}{|X\cup Y|}\ ;$$

$$overlap(X,Y)=\frac{|X\cap Y|}{\min(|X|,|Y|)}$$

Où : X désigne l'ensemble des documents où apparaît le terme x.

Y désigne l'ensemble des documents où apparaît le terme y.

|X| le cardinal de l'ensemble X.

Dans (Bruandet et Chevallet, 2003), plusieurs variantes de contexte documentaire ont été discutées, et se sont présentées brièvement comme suit :

- **Contexte documentaire non symétrique :**

  Par opposé aux mesures de similarité présentées précédemment, les formules utilisées dans un contexte documentaire non symétrique définissent des règles de comparaison telle que les probabilités conditionnelles de X sachant Y.

  Cette formule est utilisée pour construire un thésaurus d'association (Haddad, 2002). L'approche statistique utilisée se base sur la fouille de données textuelles, en particulier, la technique des règles d'association.

  En effet, le modèle proposé démontre l'impact de combinaison d'une méthode statistique et d'une méthode linguistique sur les performances d'un SRI.

- **Contexte documentaire de cluster :**

  Crouch et al. (Bruandet et Chevallet, 2003), ont utilisé le paradigme de la valeur de discrimination d'un terme (voir section 2.2.4.2) pour construire automatiquement un thésaurus. L'approche consiste à classer les documents dans des clusters et de réaliser un thésaurus à partir des termes de faibles fréquences contenus dans les clusters.

  En effet, une arborescence de clusters de documents est construite en utilisant l'algorithme de classification complet.

  Une classe de thésaurus est définie comme l'ensemble des termes discriminants communs à tous les documents d'une classe.

- **Contexte lexical :**

  Le contexte lexical fait référence aux termes en relation grammaticale. Grefenstette (Grefenstette, 1993) exploite cette notion de contexte lexical pour extraire des termes sémantiquement proches. Son système SEXTANT permet d'extraire un contexte local pour chaque terme sous forme de structures syntaxiques : adjectifs-noms, noms-noms et verbes-noms.

La distance entre deux termes peut être calculée avec la formule de jaccard, ce qui engendre, pour chaque terme, une liste pondérée, ordonnée, des termes proches. Il est évident que ces termes n'apparaissent pas forcément dans les mêmes documents. Cette constatation implique que le contexte lexical peut être utilisé comme méthode complémentaire aux contextes documentaires.

Une partie du thésaurus construit a été testé sur un corpus de résumés dans le domaine médical et a donné des résultats positifs (Bruandet et Chevallet, 2003).

#### *3.2.3.3.2 Les méthodes linguistiques*

Les méthodes linguistiques visent à extraire les relations entre les termes grâce aux phénomènes langagiers (Haddade, 2002 ; Diem Le, 2003). Elles exploitent, en général, des patrons syntaxiques et/ou marqueurs linguistiques.

- **Exploitation des patrons syntaxiques :**

  Un patron syntaxique s'appuie sur un segment de texte, souvent des syntagmes nominaux simples ou complexes qui font référence à des candidats termes du domaine, pour mettre en évidence des relations entre ces constituants selon leur enchâssement (Haddad, 2002).

  HEARST (Hearst, 1992) propose d'exploiter des patrons syntaxiques de la forme suivante :

  $NP_1$ such as $NP_2$

  Les patrons sont efficaces, notamment, dans le cas de détection des groupements de termes aux sens proches inclus dans des énumérations (Bruandet et Chevallet, 2003) telles que :

  « … most European countries, specially France, England, and Spain… »

  Ces regroupements sont en fait des synonymies contextuelles qui peuvent être utilisées, en combinaison avec des outils de calcul de variation terminologique sur les multitermes, dans l'extension des thésaurus structurés de termes à des multitermes.

  Ceci dit, la définition des patrons ne peut être fait automatiquement et requiert l'intervention manuelle des experts.

- **Exploitation des marqueurs linguistique :**

  Un marqueur linguistique peut être défini comme étant une forme linguistique faisant partie des catégories prédéfinies (grammaticales, lexicales, syntaxiques ou sémantiques) dont l'interprétation définit régulièrement le même rapport de sens entre des termes (Haddad, 2002).

  Le travail de Nazarenko A. (Haddad, 2002)., dans sa thèse, s'intéresse au problème de causalité dans le cadre d'un système de question/réponse, nommé KALIPSO (Système de compréhension automatique de texte et de question réponse en langage naturel).

  A l'aide des marqueurs linguistiques, le système procède à la construction d'une représentation d'un texte sous la forme de graphes conceptuels. Les marqueurs traités par ce système appartiennent à l'un des trois groupes suivants (Diem Le, 2003) :

  - Trois subordonnants causaux fondamentaux : parce que, puis que, comme.
  - Propositions causales : à cause de, pour.
  - Adverbe et conjonctions : car, en effet, ainsi, de fait, donc, etc.

  Les expérimentations ont montré l'augmentation des performances d'un système de question/réponse par l'utilisation d'un tel type d'analyse.

#### *3.2.3.4 Construction automatique des thésaurus sur le web*

Lors de notre étude de l'expansion de requêtes sur le web, nous avons mentionné l'émergence de deux particularités : les hyperliens et les logs de requêtes. Par ces mêmes particularités que la construction des thésaurus sur le web est influencée.

Dans ce qui suit, nous allons discuter des cas de construction des thésaurus dans l'optique de ces deux caractéristiques.

##### *3.2.3.4.1 Construction automatique des thésaurus et les hyperliens*

Dans cette section, nous allons présenter deux travaux faisant référence aux hyperliens pour construire un thésaurus.

Le premier cas s'intéresse à la structure de liens du web d'un domaine spécifique. Le second cas se focalise sur la construction d'un thésaurus d'association par l'exploitation de Wikipédia en tant que corpus d'extraction de connaissance.

- **Le premier cas : Construction d'un thésaurus à partir de la structure de lien du web (Chen et al., 2003)**

  Une caractéristique discriminante entre une page web et un texte pur est bien les hyperliens. En plus du texte, les pages web contiennent aussi des hyperliens. Un hyperlien contient une information abondante incluant la localité thématique et les textes de liens.

  La localité thématique signifie que les pages web reliées par des hyperliens sont plus susceptibles d'être du même sujet que ceux non reliées. L'étude (Davison, 2000) confirme cette assomption.

  Le texte de liens qui est le contenu textuel de l'hyperlien décrit toujours la page cible.

  Par ailleurs, la structure de liens du web constitue un réseau sémantique, dans lequel, les nœuds représentent les mots ou les phrases des textes de liens et les arcs représentent les relations sémantiques. D'où la possibilité de construire un thésaurus en utilisant l'information du réseau sémantique.

  Cette idée est exploitée par (Chen et al., 2003) pour à construire un thésaurus dédié à un domaine à partir de la structure de liens sur le web.

  Le thésaurus est expérimenté en utilisant le système OKAPI. Les résultats concernant l'expansion de requêtes révèlent que :

  - L'expansion de requêtes à base de relations frères (sibling) est mauvaise.
  - L'expansion de requêtes à base de relations enfants améliore les performances du système.

  Pour plus de détail, en chiffre, voir (Chen et al., 2003).

- **Le seconde cas : Construction d'un thésaurus d'association sur le web par l'exploration de Wikipédia (Nakayama et al., 2007)**

  Nakayama et al. proposent une méthode d'exploration à base de structure de liens pour construire un thésaurus d'association à partir de Wikipédia. Dans ce contexte, Wikipédia est considérée comme un corpus web à structure de liens dense qui définit plusieurs liens internes entre une immense quantité d'articles (concepts).

  - **La stratégie de base (pfibf)**

    Wikipédia peut être exprimée par un graphe G = {V, E} (V : l'ensemble d'articles, E : l'ensemble des liens).

La question est comment mesurer la relation entre une paire d'articles ($v_i$, $v_j$)? L'étude suppose qu'une telle relation est fortement affectée par trois facteurs :

1. le nombre de chemins depuis l'article $v_i$ à l'article $v_j$.
2. la longueur de chaque chemin entre l'article $v_i$ et l'article $v_j$.
3. le nombre de liens vers l'arrière (backward link) depuis un article $v_i$.

Par analogie à la méthode TF-IDF, la méthode pfibf (Path frequency – backward link frequency) est définie sous les considérations qui stipulent qu'une page web (wikipédia) correspond à un concept et les liens sont des associations sémantiques entre les concepts.

L'utilisation d'une matrice d'adjacence pour représenter le nombre de chemins entre les paires d'articles semble inadéquate à cause de la loi de scalabilité. Wikipédia comprend 1.3 millions d'articles, soit plusieurs téraoctets, juste pour sauvegarder les données. Pour surmonter ce problème, les auteurs ont proposé une structure de données, DBT (Dual Binary Tree) comme une compression des données de la matrice d'adjacence, ainsi qu'un algorithme de multiplication opérationnel sur cette structure de données.

Suite à une constatation qui stipule que l'analyse pfibf donne de mauvais résultats pour les termes généraux par rapport aux termes d'un domaine spécifique, les auteurs ont enrichi leur approche par une méthode de pondération « FB weighting ». Cette méthode modifie d'une façon flexible le poids des analyses de liens en avant et les analyses de poids en arrière.

- **Les expérimentations**

  Pour évaluer les avantages de l'approche proposée, les auteurs ont construit quatre thésaurus à partir de Wikipédia en utilisant quatre méthodes, à savoir, TF-IDF, analyse de cooccurrence de liens, pfibf et pfibf avec « FB weighting ».

  La collection de test utilisée est « word similarity 353 » qui est largement utilisée dans les recherches sur Wikipédia.

  Les expérimentations ont révélé un avantage notable pour le thésaurus construit automatiquement pour pfibf avec « FB weighting »[1].

#### *3.2.3.4.2 Construction automatique d'un thésaurus à partir des logs (Chuang et al., 2000)*

L'objectif de Chuang S-L. et al. était de construire un thésaurus en direct (live thesaurus) à partir des logs de termes de recherche.

Sur la base de ce thésaurus, la recherche tente de construire un moteur de recherche frontal avec l'aptitude de suggérer des termes automatiquement, ainsi que la réalisation des métas-recherches (Figure 20).

[1] Le thésaurus est accessible sur http://wikipedia-lab.org:8080/wikipediathesaurusv2

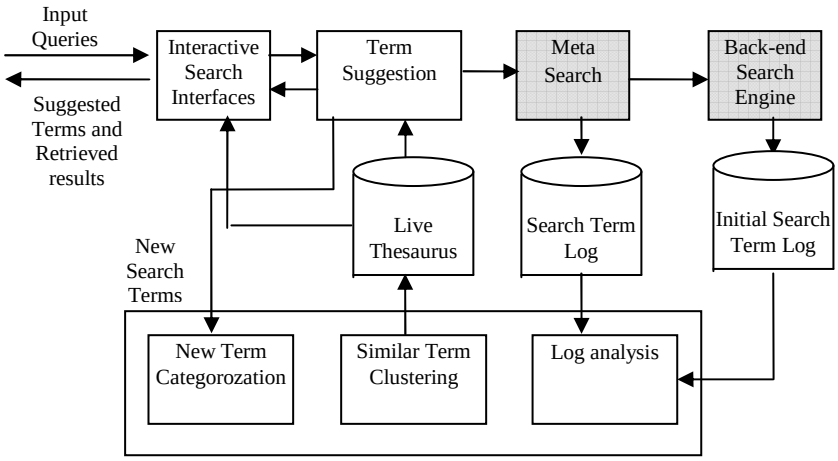



**Figure 20. : Schéma fonctionnel de la recherche sur le web avec le thésaurus en direct**

L'approche proposée est une intégration entre l'effort humain et automatique. Elle se déroule en trois étapes :

1. **analyse de logs des termes de recherche et extraction des termes noyaux**

   Les logs de requêtes utilisées parviennent des deux moteurs de recherche, Dreamer et GAIS dans Taiwan.

   Une log de requête consiste en une série de questions, et chaque question inclut le terme de recherche et le temps d'envoi de la question.

   Le travail commence par l'utilisation d'une taxonomie de sujets en vue de construire un schéma de classification et la classification de termes de recherche dans certaines catégories de sujets. La plupart de ce travail est réalisé manuellement.

2. **Classification de nouveaux termes.**

   Dans un objectif de garder le thésaurus incrémental et adaptatif, la méthode procède à la classification automatique de chaque nouveau terme n'apparaissant pas dans le thésaurus.

3. **Clusterisation des termes similaires.**

   L'extraction de termes similaires s'effectue à partir des termes de recherche dans le même domaine de sujets en se basant sur les techniques de classification des nouveaux termes et l'analyse des logs de termes de recherche. Ceci peut réduire considérablement le coût de calcul et l'ambiguïté liée au sens des mots.

   Les mots similaires à clustériser peuvent être classifier selon deus types, les termes similaires par contenu (exemple : les abréviations) et les termes différents dans le contenu mais similaire dans le concept.

   Il reste à noter que plusieurs applications sont possibles pour le thésaurus en direct. Parmi les quelles, l'extraction des termes pertinents, la classification des sites/pages web et la recherche interactive sur le web.

# Chapitre 4 : Construction et Utilisation d'un Thésaurus pour la RI en Elearning

## 4.1 Elearning

Le phénomène du déluge informationnel du web est déjà discuté (voir section 2.1.3.2), notez qu'une bonne partie de cette information concerne la communauté d'éducation. Ce potentiel d'information demeure inexploitable sans l'utilisation d'un SRI efficace qui assiste l'utilisateur à repérer l'information pertinente dont il a besoin.

De son coté, le domaine elearning s'occupe de la présentation de l'information pour cette communauté dans un but d'apprentissage (Kumar et al., 2005).

Ce chapitre présente un rapide historique sue l'évolution des systèmes de formation à distance. Le rôle des objets pédagogiques est ensuite clarifié. Enfin, le besoin en normalisation est mis en valeur.

### 4.1.1 De la formation par correspondance au elearning

La notion de formation à distance est apparue à partir du XIXe siècle à Londres en 1840 sous forme de « cours par correspondances » acheminés par poste ensuite par fax (Payement, 2005). A la seconde moitié du XIXe siècle et du début du XXe siècle, d'autres pays européens ont développé cette forme de formation.

En Algérie, c'est le CNEPC (Centre National de l'Enseignement Professionnel par Correspondance) crée en 1984[1]qui assure les différents types de formation professionnelle par correspondance. Le CNEPC est érigé, en 1990, en Centre National de l'Enseignement Professionnel à Distance (CNEPD)[2].

La seconde moitié du XXe siècle est marquée par l'apparition de nouveaux besoins en matière de formation et l'émergence de nouveaux outils technologiques tels que les cassettes audio et vidéo, puis la diffusion hertzienne via la radio et la télévision pour arriver à l'enseignement assistée par ordinateur (Benayache, 2005).

Un peu plus tard, un autre concept a vu le jour, la FOAD[3] pour Formation Ouverte et A Distance définie comme suit[4] :

« Une formation ouverte et/ou à distance est un dispositif souple de formation organisé en fonction de besoins individuels ou collectifs (individus, entreprises, territoires). Elle comporte des apprentissages individualisés et l'accès à des ressources et compétences locales ou à distance. Elle n'est pas exécutée nécessairement sous le contrôle permanent d'un formateur ».

L'apprenant, de sa part, peut utiliser multiple médias complémentaires comme support (Payement, 2005) :

- Les cours par correspondance,
- les systèmes de formation en ligne,
- les centres de ressources,

[1] Décret exécutif n° 84-271 du 15 septembre 1984.

[2] Décret exécutif n° 90-298 du 6 octobre 1990.

[3] En anglais ODL pour Open and Distance Learning.

[4] Circulaire n° 2001/22 du 2 juillet 2001 relative aux formations ouvertes et à distance de la délégation générale à l'emploi et à la formation professionnelle, placée sous l'autorité du ministère de l'emploi, de la cohésion et du logement- France.

- les cours télédiffusés par radio ou télévision,
- le téléprésentiel collectif ou individuel (télécours, télétutorat),
- les campus virtuels ou classes virtuelles, etc.

L'expansion des réseaux et d'Internet, depuis peu, à fait émerger un nouveau concept, le elearning. La commission européenne le définit comme étant :

« L'utilisation des nouvelles technologies multimédias et de l'Internet dans le but d'améliorer la qualité de l'apprentissage en facilitant l'accès à des ressources et des services, ainsi que les échanges et la collaboration à distance ».

Le elearning peut être défini comme un sous ensemble de la FOAD, s'appuyant sur les réseaux électroniques et offrant des échanges qui se font aux travers de forums, classes virtuelles, chats et projets à mener en commun et à distance permettant de développer les compétences des apprenants d'une façon autonome et rendant le processus d'apprentissage indépendant du temps et du lieu (Payement, 2005).

Notons que les systèmes elearning s'appuient généralement sur les objets pédagogiques et les plates-formes de formation en ligne.

### 4.1.2 Documents pédagogiques et objets pédagogiques

Dans le contexte elearning, les documents pédagogiques constituent un sous ensemble des documents disponibles sur le web. Ils sont qualifiés de pédagogiques parce qu'ils ont été crées ou qu'ils peuvent être utilisés pour l'enseignement (Bourda, 2002).

Ces documents pédagogiques se transforment en objets pédagogiques, entités numériques utilisées dans un contexte d'enseignement ou d'apprentissage ayant les propriétés suivantes (Bourda, 2002) :

- Autonomie : chaque objet pédagogique peut être utilisé indépendamment des autres, ce qui nécessite un choix rigoureux de sa granularité. L'unité d'un objet pédagogique peut être un programme, un module, une leçon,…
- Reutilisabilité : un objet pédagogique élémentaire peut être utilisé dans de nouveaux contextes d'apprentissage sans beaucoup d'efforts supplémentaires.
- Agrégation : les objets pédagogiques peuvent être regroupés pour constituer d'autres objets pédagogiques.
- Indexation : chaque objet pédagogique doit être décrit d'une façon qui simplifie son accessibilité. Un objet pédagogique non indexé est considéré comme perdu.

Ceci dit, plusieurs types de documents pédagogiques peuvent être distingués (Rouissi, 2007) :

- Les documents méthodologiques.
- Les supports de cours (résumés, plans détaillés, contenu complet).
- Les documents annexes (illustrations, bibliographie, glossaires, exemples, articles).
- Les documents d'évaluation des connaissances (devoir, tests).
- Les travaux des étudiants (production web, réalisation de présentations assistées, exposés, dossier, rapport, thèses,…).

Ces documents pédagogiques peuvent apparaître sous plusieurs formes :

- Textes : format PDF (Portable Document Format), RTF (Rich Text Format), TXT (format texte basé sur ASCII),…
- Images : photos, cartes, schémas,…

- Présentations assistées par ordinateur.
- Emission sonores, vidéo.
- Multimédia : combinaison de diverses formes (texte, son, image, vidéo).
- Hypertexte.
- Animations : les documents hypertextes peuvent être enrichi par des animations (animation des flash, applet Java, code javascript, SMIL, HTML + TIME).

Ces différents documents peuvent être accédés dans un environnement de travail numérique[1] ou tout simplement en libre accès sur un site web. Ils peuvent aussi être organisés dans des plates-formes de formation en ligne qui font l'objet de la section suivante.

### 4.1.3 Les plates-formes pédagogiques

Une plate-forme pédagogique est un logiciel qui assiste la conduite des formations ouvertes et à distance (Benayache, 2005). Elle se repose sur les techniques de travail collaboratif. Elle permet un accès à distance à des contenus pédagogiques ainsi que des échanges entre les principaux acteurs de la formation : apprenant, tuteur et administrateur.

Une plate-forme pédagogique regroupe les outils nécessaires permettant aux acteurs d'incorporer des contenus pédagogiques, de participer à des activités et de réaliser un suivi pédagogique et administratif des apprenants.

L'usage des plates-formes est relativement standard (Benayache, 2005 ; Oubahssi, 2005) :

- Le tuteur crée des parcours type, les individualise, incorpore des ressources pédagogiques multimédias et effectue un suivi des activités des apprenants.
- L'apprenant peut consulter en ligne ou télécharger les contenus pédagogiques qui lui sont recommandés, effectuer des exercices, s'auto évaluer et transmettre des travaux à son tuteur pour les corriger. La communication entre tuteur et apprenant peut être individuelle ou en groupe. Ils peuvent créer des thèmes de discussions et collaborent à des travaux communs.
- L'administrateur assure la maintenance du système, gère les comptes et les droits des utilisateurs, crée des liens vers d'autres systèmes et ressources externes.

Cependant, les plates-formes ne sont pas toutes identiques, elles se différencient par les fonctionnalités relatives à la communication, à l'ingénierie pédagogique et à la gestion (Even, 2003).

Comme exemples de plates-formes pédagogiques, on peut citer les suivantes :

- ARIDNE : Alliance of Remote Instructional and Distribution Networks for Europe.
- INES : INteractive Elearning System.
- SERPOLET : Système d'Enseignement et de Recyclage par Ordinateur Liant Expertise et nouvelles Technologies.
- WebCT : Web Course Tools.
- ECSAIWeb : Environnement de Conception de Système d'Apprentissage Intelligent sur le Web.

[1] Un dispositif global fournissant à un usager un point d'accès à travers les réseaux à l'ensemble des ressources et des services en rapport avec son activité (Behaz et Djoudi, 2005).

Soulignons que le nombre de plates-formes ne cesse de croître[1]. A chaque contexte de formation peut correspondre des plates-formes potentielles qui utilisent à leur tour des formats de ressources pédagogiques propriétaires.

Pour des raisons, essentiellement, d'interopérabilité et de reutilisabilité, le recours à la standardisation s'impose.

### 4.1.4 Normes en elearning

Dans les premiers temps, les différentes institutions éducatives se sont engagées d'une façon individuelle et autonome dans la conception et le développement des produits destinés au elearning. On parlait d'une approche propriétaire. Toute tentative d'exploitation d'un tel produit par un autre utilisateur, par un autre logiciel ou son intégration dans un autre environnement requiert d'énormes efforts.

De ce qui précède, la normalisation se montre comme solution qui vise, à la fois, l'efficacité économique et pédagogique.

En effet les enjeux de la normalisation sont nombreux, les anglophones les regroupent souvent sous le vocable d' « -ilities » (Uyttebrouck, 2002) :

- Interoperability : la possibilité de deux ou plusieurs systèmes ou composants d'interchanger les informations et l'utilisation de cette information interchangée.
- Portability : opérationnel sur différents supports matériel et logiciel.
- Rusability : utilisation des mêmes objets pédagogiques pour différentes fins, dans différentes applications, dans différents contextes et par différents modes d'accès.
- Flexibility : adaptabilité aux changements.
- Accessibility : recherche, identification, accès et livraison des contenus et composantes de la formation.
- Durability : affronter les changements technologiques (support logiciel ou matériel) sans la nécessité de développer à nouveau.

#### *4.1.4.1 Les domaines de normalisation en elearning*

Le périmètre de la normalisation en elearning peut inclure les domaines suivants (Oubahssi, 2005) :

- Vocabulaire : une terminologie à la fois lisible par l'utilisateur et interprétable par la machine.
- Architectures : garantir la durabilité des dispositifs tels que les plates-formes pédagogiques ainsi que les procédures d'échange des données entre elles.
- Informations sur l'apprenant : définition des structures de données décrivant un apprenant, ses compétences, son profil d'apprentissage et son plan de formation.
- Contenus de formation : définition de métadonnées permettant l'accessibilité des contenus éducatifs.
- Systèmes de gestion : description des modèles et de scénarios pédagogiques.
- Qualité : définition d'un référentiel de qualité permettant l'évaluation des services elearning, ceci en considérant le contenu éducatif ainsi que le degré de satisfaction des apprenants.

[1] L'organisme THOT (http://thot.cursus.edu/) recense plus de 240 plates-formes de formation en octobre 2003.

### *4.1.4.2 Les acteurs et les travaux de la normalisation en elearning*

Pour une représentation synthétique des acteurs et les travaux de normalisation, nous avons estimé utile de présenter un extrait du rapport élaboré par le groupe de travail sur les normes et standards de la formation en ligne du sous-comité sur les technologies de l'information et de la communication (SCTIC, 2002).

#### *4.1.4.2.1 Les types d'acteurs*

Un nombre important d'acteurs engagé dans le domaine de développement des normes et standards pour la formation en ligne. Ces acteurs peuvent être classés selon trois types (Figure 21.) :

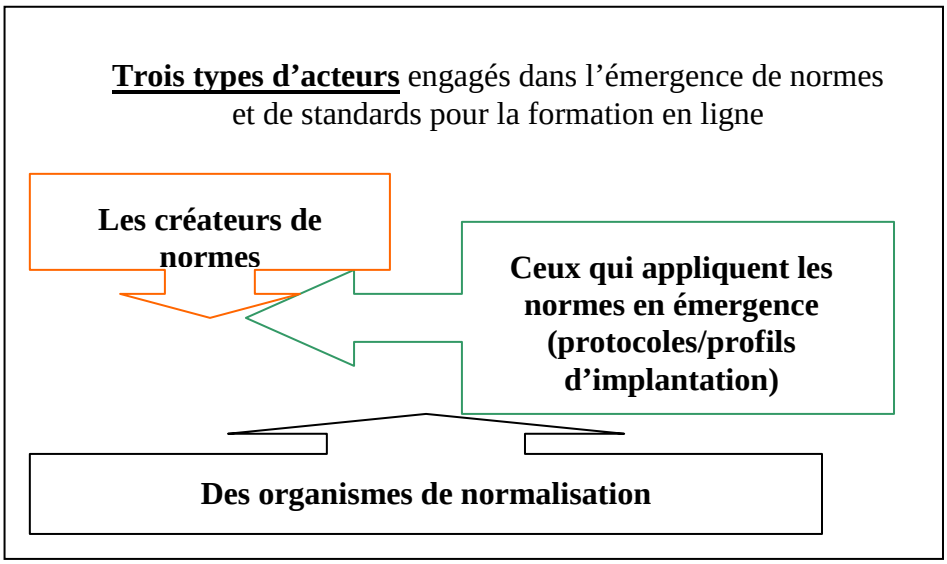



**Figure 21. : Les types d'acteurs du domaine elearning**

- Les créateurs : développement des spécifications susceptibles de devenir de nouvelles normes (Tableau 5.).
- Les groupes qui appliquent les normes en définissant des protocoles décrivant leur implantation (Tableau 6.).
- Les organismes de normalisation qui sont concernés par la normalisation des pratiques et outils de formation en ligne (Tableau 7.).

| |
|---|
| **IMS** - IMS Global Learning Consortium (IMS) - *États-Unis*<br>http://www.imsproject.org/aboutims.html |
| **DCMI - Dublin Core** - Dublin Core Metadata Initiative - *Ohio, États-Uni*<br>http://dublincore.org |
| **ECTS** -European Community Course Credit Transfer System - *Communauté européenne*<br>http://europa.eu.int/comm/education/socrates/ects.html |
| **AICC** - Aviation Industry CBT Committee (EAO)<br>http://www.aicc.org/index.html |
| **EML** - Educational Modelling Language - Open University of the Netherlands<br>http://eml.ou.nl/introduction/ |
| **ALIC** - Advanced learning infrastructure Consortium – *Japon*<br>http://www.alic.gr.jp/eng/index.htm |

**Tableau 5. : Les principaux créateurs de normes et de standards**

| | | |
|---|---|---|
| **CanCore**<br>http://www.cancore.ca | Canadian Core Learning Resource Metadata Application Profile –<br>*Canada* | Sous ensemble de IMS |
| **ADL – SCORM**<br>http://www.adlnet.org | Projet Sharable Content Object Reference Model du Advanced distributed learning -<br>*Défense américaine et enseignement universitaire* | IMS |
| **MERLOT**<br>http://www.merlot.org<br>http://taste.merlot.org/ | The Multimedia Educational Resource for Learning and Online Teaching<br>*Californie* | IMS |
| **ARIADNE**<br>http://ariadne.unil.ch/Metadata/ | Alliance of Remote Instructional Authoring and Distribution Networks for *Europe* | IMS |
| **GESTALT**<br>http://www.fdgroup.co.uk/gestalt/ | Getting Educational Systems Talking Across Leading-Edge Technologies<br>*Royaume-Uni* | IMS<br>IEEE |
| **EdNA**<br>http://standards.edna.edu.au/metadata/elements.html | Metadata Standard de l'Education Network Australia - *Australie* | Dublin Core<br>IMS |
| **MEG**<br>http://www.ukoln.ac.uk/metadata/education/ | Metadata for Education Group<br>*Royaume-Uni* | IMS |
| **OKI - MIT**<br>http://Web.mit.edu/oki | Open Knowledge Initiative du Massachuset Institute of Technology (MIT) | IMS<br>ADL |
| **LRN**<br>http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/itsolutions/education/deploy/lrntoolkit/lrndeflt.asp | Learning Resource interchange - initiative de Microsoft Technet semblable à SCORM et fondée sur la norme Content Packaging format de IMS | IMS<br>AICC<br>(ADLSCORM)<br>IEEE |
| **ULF**<br>www.saba.com/standards/ulf/ | Universal Learning Format - initiative d'un fournisseur (SABA) de solutions pour la formation en ligne fondée sur les travaux de IMS, ADL et IEEE | IMS<br>ADL<br>IEEE |

**Tableau 6. : Des groupes qui appliquent les normes et standards**

| | | |
|---|---|---|
| **ISO – JTC 1- SC36**<br>http://jtc1sc36.org/<br>http://jtc1sc36.org/related_activities.html | International Standards Organisation -<br>Joint Technical Committee no 1 -<br>Sous-comité 36 -<br>Chantier de normalisation des systèmes d'information destinés à l'enseignement et la formation.<br>International | IEEE/LTSC<br>CEN/ISSS<br>AICCC<br>ARIADNE<br>IMS<br>ALIC<br>ADL<br>DCMI |
| **IEEE – LTSC**<br>http://ltsc.ieee.org/wg12/index.html | Institute of Electrical and Electronics Engineers, Inc. - Learning Objects Metadata working group -<br>International | ISO-JTC1-SC36 |
| **CEN/ISSS**<br>http://www.cenorm.be/isss/Workshop/lt/ | Comité européen de normalisation - Information Society Standardization System<br>Prometeus initiative PROmoting Multimedia access to Education and Training in EUropean Society), URL :<br>http://prometeus.org | IEEE-LTSC |
| **W3C**<br>http://www.w3.org/Metadata/Activity.html | The World Wide Web Consortium | |

**Tableau 7. : Les organismes de normalisation**

## 4.2 La construction du thésaurus

Cette section est consacrée à la présentation de l'approche choisie pour la construction d'un thésaurus qui sera utilisé pour l'expansion de requêtes lors de la recherche d'informations sur le web.

Les grands axes de l'approche sont inspirés des travaux exposés dans (Chen et al., 2003). Toutefois des critiques, des perfectionnements et des ajouts constituent un apport personnel estimé nécessaire pour atteindre les objectifs tracés pour notre projet.

Le dit thésaurus est spécifique à un domaine. Le domaine « elearning » nous sert comme une étude de cas.

Comme les documents cibles sont des pages web, nous nous sommes appuyés sur les hyperliens qui constituent une caractéristique discriminante entre une page web et un document classique, pour construire le thésaurus.

Par ailleurs les techniques basées sur l'analyse des hyperliens sont plus fiables que celles basées sur le contenu. Ceci est du au fait que la qualité d'une page web dépend de la qualité des pages web liées à cette dernière, et elle est alors hors du contrôle du concepteur de la page.

La construction du thésaurus ne se base pas sur la totalité de la collection, il s'agit du web. Alors, la collection considérée est la partie pertinente des résultats de la recherche suite à une requête initiale transmise aux moteurs de recherche.

Pour assurer l'interaction entre les utilisateurs, notre thésaurus et les moteurs de recherche nous développerons un méta moteur de recherche dont la description des fonctionnalités illustre notre approche de construction automatique d'un thésaurus.

### 4.2.1 Architecture du méta moteur proposé

Cette section se focalise sur les différents composants du méta moteur proposé. La figure 22. illustre leurs différentes interactions pour la construction automatique du thésaurus sur le web.

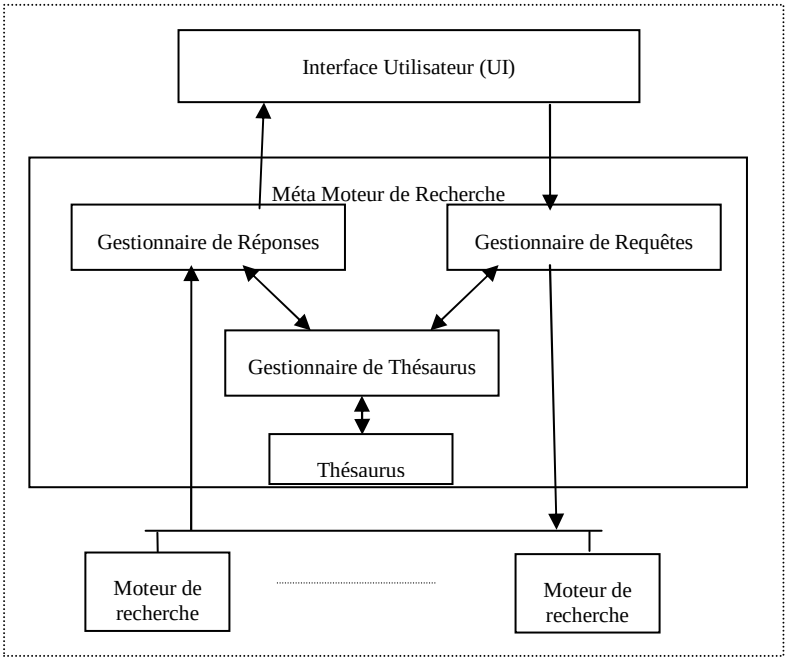



**Figure 22. : Architecture générale du méta moteur de recherche**

A travers l'interface utilisateur (*UI*), l'utilisateur spécifie un nom de domaine, par exemple, « elearning » sous forme d'une requête qui sera transmise aux moteurs de recherche (dans un premier temps nous nous limiterons à un seul moteur de recherche, Google) par le biais du

*Gestionnaire des Requêtes.* Les résultats retournés par les moteurs de recherche sont traités par le *Gestionnaire des Réponses* dont le rôle principal est d'appliquer un tri local aux sites web retournés.

Les sites web ordonnés, considérés comme sites de haute qualité vis à vis la requête de l'utilisateur sont confiés au *Gestionnaire du Thésaurus* qui procède à en appliquer l'analyse de liens. Les structures de liens ainsi obtenues sont, ensuite, translatées en structures de contenu en utilisant les techniques d'analyse d'URLs et le texte de lien (Anchor text) comme résumé sémantique des pages web destination.

En fin la génération de thésaurus s'effectue en appliquant des mesures de similarités sur les structures de contenu.

Dans ce qui suit, nous expliquons notre approche de construction du thésaurus étape par étape selon les points suivants :

- Collection des sites web de haute qualité.
- Construction des structures de contenu de la collection.
- Génération du thésaurus.

### 4.2.2 Collection des sites web pour un domaine

Pour construire un thésaurus sur le web inhérent à un domaine spécifique, nous aurons besoin d'abord d'un certain nombre de sites web de haute qualité et représentatif du domaine.

L'utilisateur soumet le nom du domaine, dans notre cas « elearning », sous forme d'une requête au moteur de recherche google pour recevoir une liste de sites web estimés pertinents grâce au mécanisme de Pagerank.

Cependant, le Pagerank est indépendant de la requête, autrement dit, le Pagerank d'une page web est le même pour différentes requêtes.

Notons, que nous accordons une grande importance à cette étape de sélection de sites web de haute qualité dans le processus de construction du thésaurus car nous jugeons que la qualité du thésaurus d'un domaine est étroitement liée à la qualité des sites web de départ.

Dans notre contexte, le recours à un autre mécanisme de sélection des sites web est indispensable. Le tri local est adopté, dans une perspective de prendre en compte essentiellement la requête utilisateur lors du processus d'ordonnancement des sites web résultats.

Notre choix s'est dirigé vers l'algorithme HITS (Hyperlink Induced Topic Search) qui mesure les scores d'une page web d'une façon dynamique pour chaque requête, plutôt que d'attribuer un score global indépendamment de la requête.

Par ailleurs, les scores HITS sont calculés à partir d'un sous ensemble relativement petit, par rapport à la totalité du web.

#### *4.2.2.1 L'algorithme HITS (Tri local)*

Le principe de l'algorithme HITS consiste à construire un sous graphe du web appelé graphe de voisinage et ordonner les pages web de ce graphe en affectant un score à chacune d'elle.

##### *4.2.2.1.1 Construction du graphe de voisinage*

Après la récupération d'un ensemble de pages web (Rq) suite à une requête utilisateur (q), nous procédons à l'extension de cet ensemble en rajoutant le voisinage direct des pages web de cet ensemble. Nous entendons par voisinage direct l'ensemble des pages pointées (ou qui pointent) par (ou vers) les pages de Rq.

Lors des expérimentations, Rq comprend typiquement 200 pages et les liens entrants pour chaque page de Rq seront limitées à 50. D'où, la taille de l'ensemble étendu (Sq) varie entre 1000 et 5000.

Afin de garantir que l'ensemble étendu Sq contienne des pages pertinentes, nous ne considérons que les liens externes, c'est-à-dire les liens entre pages web de sites différents.

#### *4.2.2.1.2 Calcul des scores autorités et hubs*

Kleinberg (Kleinberg, 1999) considère une page web comme autorité si elle est pointée par beaucoup de pages web et comme hub (central) si elle pointe vers beaucoup de pages web.

De manière récursive, il considère une page web comme une bonne autorité si elle est pointée par de bons hubs, et comme de bon hub si elle pointe vers de bonnes autorités. C'est ce qu'on appelle le renforcement mutuel.

Le listing 4. illustre les grandes étapes de calcul des scores autorité et hub (Markov et Larose, 2007).

1. En utilisant un SRI standard, un petit ensemble de pages web pertinentes, appelé ensemble racine $R_q$ est trouvé ( q est une requête)
2. $R_q$ est étendu par l'ajout de pages pointées (ou qui pointent) par (ou vers) les pages de $R_q$. Cet ensmble étendu est appelé ensemble $S_q$
3. la structure hyperliens de $S_q$ analysée pour trouver les pages autorités et centrales, comme suit :
4. Soit E la matrice d'adjacence du graphe web de $S_q$, où E(u,v)=1 si la page u pointe la page v, et E(u,v)=0 sinon. (u et v $\in$ $S_q$)
5. Soit X=($x_1$ $x_2$ … $x_n$)le vecteur autorité et Y=($y_1$ $y_2$ … $y_n$) le vecteur central. X et Y peuvent être calculés par itération comme suit (k est un parmètre de réglage (tuned parameter) :
   - X ← ( 1 1 … 1)
   - Y ← ( 1 1 … 1)
   - Répeter k fois
     - $x_u \leftarrow \sum_{\{v,\ E(v,u)=1\}} y_v, \quad pour\ \ u = 1,2,...,n$
     - $y_u \leftarrow \sum_{\{v,\ E(u,v)=1\}} x_v, \quad pour\ \ u = 1,2,...,n$
     - normaliser X et Y
   - Fin répéter

**Listing 4. : L'algorithme HITS**

### 4.2.3 Construction des structures de contenu de la collection

L'étape précédente nous a permis de disposer d'une collection de sites web de haute qualité et représentative pour le domaine.

Le but de cette étape est d'extraire les concepts et les relations entre les concepts véhiculés par cette collection.

Constatons que le concepteur d'un site web conçoit la structure informationnelle de son site web dans son esprit qui sera éventuellement transposé en un diagramme en utilisant un outil de conception.

Ensuite, le concepteur procède à la compilation de cette structure informationnelle pour générer un ensemble de pages web (écrit en HTML) inter-liées, avec des outils d'édition.

L'idée est de procéder à une ingénierie inverse qui consiste à prendre le point de départ un site web et déduire, en fin, les intentions du concepteur de site sous forme d'une structure de contenu.

Pour atteindre cet objectif, les étapes entreprises sont :

- Analyse de liens des sites web.
- Construction de la structure de liens.
- Construction de la structure de contenu.

#### *4.2.3.1 Analyse de liens des sites web*

L'analyse de liens d'un site web est réalisée par le module *Spider* inclue dans le *Gestionnaire de Réponses*. Le *Spider* voyage d'une page web à une autre pour accomplir cette tâche. Il doit être capable de localiser les liens contenus dans chaque page.

Pour se faire, le *Spider* examine toutes les balises du code HTML de la page. La plupart des balises utilisent l'attribut HREF (Hypertext REFerence) pour pointer une autre page. A titre d'exemple, la balise <a href = « /referencement/ e-learning.html »> indique qu'une autre page e-learning.html peut être visitée.

Il est à noter qu'il existe trois types de liens que le *Spider* peut rencontrer. Les liens internes pointent des pages du même serveur web que la page contenant le lien. Les liens externes référencent des pages appartenant à des sites web différents. Alors que, le troisième type, autres liens, référencent des ressources autres que des pages web.

Dans notre contexte, on se limite aux liens internes, car notre objectif, pour chaque site web, est de construire sa structure de liens.

- **Méthode de construction du Spider**

Dans la littérature, il existe deux manières pour construire un spider. La première consiste à utiliser la technique de programmation récursive. La seconde utilise la technique itérative, mais souvent, elle fait appel au multi-threading qui n'est pas compatible avec la récursivité (chaque thread possède sa propre pile).

Du moment que le nombre de sites web de la collection est limité, notre choix est orienté vers la solution récursive (listing 5.).

Toutefois, il est à noter que pour un crawling ouvert du web, la solution itérative multi-threaded est recommandée pour éviter, d'une part, le problème de débordement de la pile, et d'autre part, d'accélérer le processus d'analyse de liens.

```
Spider_recursive (lien_url)
Debut
  Telechargement de lien_url
  Analyse lien_url
  Pour chaque sous_url trouvée faire
  Debut
        Appeler Spider_recursive (sous_url)
  Fin
Fin
```

**Listing 5. : L'algorithme d'un spider récursif**

- **Liens absolus et lien relatifs**

Les liens peuvent être absolus ou relatifs. Un lien absolu est une URL complète de la forme :

Protocol://host/domain/directory/file.

Un lien relatif concerne les liens internes, il dépend de son contexte, c'est-à-dire de son emplacement dans le site web. Il existe plusieurs types de liens relatifs. Ils peuvent être relatifs vers une page de même niveau, vers une page de niveau inférieur ou supérieur, vers des répertoires inférieurs ou supérieurs,…

Les liens relatifs sont fréquemment utilisés dans Internet. C'est pour cette raison que le *Spider* développé doit les translater en liens absolus, car lors de l'analyse, ils seront dissociés de leurs emplacements.

- **L'identificateur de fragment « # »**

Pour les URLs qui pointent vers des portions de la même page en utilisant l'identificateur de fragment « # » sont considérés comme identique à l'URL de base, car elles n'affectent pas la structure de liens des pages web.

- **Les chaines de requêtes**

Les sites web dynamiques permettent aux utilisateurs d'ajuster certains paramètres tels que les propriétés d'affichage (http://abc.de/?cs=bleu). La question posée ici est, est-ce que les différentes URLs qui se diffèrent uniquement au niveau de la chaîne de requête sont co-extensives ? Autrement dit, est ce qu'elles renferment le même contenu et la même structure ? La tache d'examination de cette propriété semble infaisable. Pour cette raison, la plupart des spiders suppriment les chaînes de requêtes et considèrent, ainsi, les URLs comme identiques.

- **Les pages par défaut**

En ce qui concerne les pages par défaut, les serveurs web peuvent spécifier un fichier index.html, default.html ou bien n'importe quel autre nom de fichier qui peut être paramétré dans la configuration du serveur web. Ceci peut induire une duplication des URLs. Pour remédier à ce problème, le *Spider* procède à une comparaison des ressources susceptibles d'être identiques.

#### *4.2.3.2 Construction de la structure de liens*

Les hyperliens dans une page HTML sont unidirectionnels, ce qui implique que la structure de liens reliant les différents documents engendre un graphe orienté.

Les nœuds d'un tel graphe représentent les pages web, tandis que les arcs représentent les liens entre les pages. Il est évident que ce graphe est cyclique, chaque page devrait au moins comporter un lien vers la page d'accueil.

Ceci dit, beaucoup de recherches ont montré que les diagrammes arborescents sont les représentations les plus souhaitées pour les développeurs, les testeurs d'utilisabilité et les fournisseurs de contenu.

Par conséquent, il s'avère utile de générer une hiérarchie qui représente la structure d'un site, appelée structure de liens.

- **Méthode de parcours**

Nous avons choisi d'utiliser l'algorithme d'exploration en largeur d'abord (Breadth First Search, BFS) comme stratégie de parcours des différents liens des pages web. L'algorithme est décrit dans le listing 6.

```
Algorithme BFS (Graphe G, Sommet S)
// G est un graphe , S est le sommet du graphe
Debut
  Créer une file d'attente Q
Enfiler(Q,S)
Tant que Q est non vide Faire
Debut
  X=Defiler(Q)
  Pour tout voisin Z de X Faire
  Debut
        Si non Marque(Z) alors
        Debut
              Marquer(Z)
```

```
                Enfiler(Q,Z)
            Fin
  Fin
Fin

Fin
```

**Listing 6. : Algorithme BFS**

D'une part, l'algorithme BFS est la méthode standard du spidering, et d'autre part, il s'est montré complet (si une solution existe elle sera trouvée) et optimal (il fournira la meilleur solution).

L'algorithme BFS marque une complexité accrue en temps et en espace $O(b^d)$, (b: facteur de branchement, d : profondeur de la solution).

Toutefois, dans notre cas, le nombre de sites web à visiter est limité, d'où un effet négligeable de la complexité en temps et en espace sur le processus de construction de la structure de liens.

En outre, l'algorithme BFS est un algorithme de plus court chemin. Par contre, l'autre alternatif de parcours de graphe, l'algorithme d'exploration en profondeur d'abord (Depth First Search, DFS) n'est ni complet ni optimal.

La figure 23. montre le principe général des deux algorithmes.

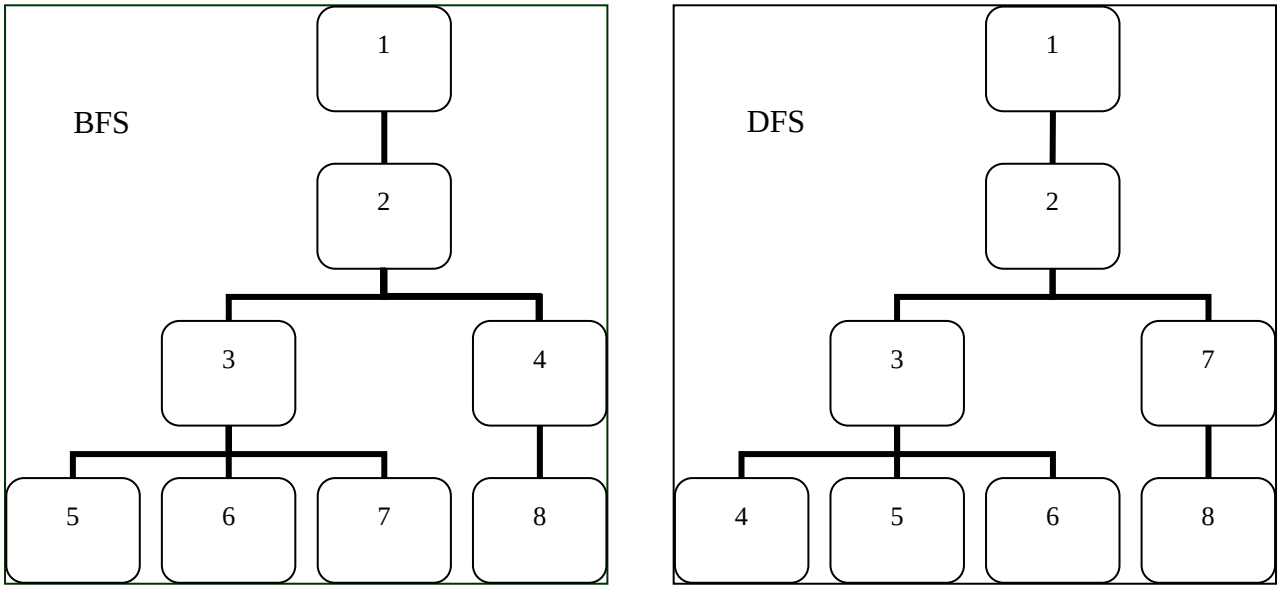



**Figure 23. : BFS vs. DFS**

### *4.2.3.3 Construction de la structure de contenu*

L'étape précédente nous a permis d'obtenir des structures de liens qui représentent les sites web d'un domaine. Cependant, ces structures de liens ne sont pas purement sémantique et ne reflètent guère les intentions des concepteurs des sites web.

La présente section se focalise sur l'extraction de la structure de contenu d'un site web à partir de sa structure de liens.

L'idée est de substituer les pages web (nœuds) de la structure de liens par des concepts qui peuvent véhiculer le sens générique des pages web et d'assurer le passage des liens de la structure de liens en des relations sémantiques entre les concepts.

Pour se faire, les étapes suivantes sont suivies :

- Résumé d'une page web à un concept.
- Elimination des liens de navigation.
- Découverte des relations sémantiques entre concepts.

- **Résumé d'une page web à un concept**

Le résumé des pages web dérive des techniques de résumer du texte qui utilisent des approches statistiques et/ou linguistiques.

Noter qu'il est très difficile de résumer automatiquement une page web d'une manière efficace, car les pages web diffèrent des documents de texte traditionnels, à la fois en structure et en contenu. L'existence des textes de liens et des textes spéciaux contribue à cette différence.

Par ailleurs, il a été constaté que le texte de liens fournit des descriptions plus précises que les pages web elles même (Brin et Page, 1998). Par conséquent, la prise en compte de cette information descriptive peut améliorer les valeurs de rappel des moteurs de recherche.

Sur la base de cette constatation, nous avons opté pour l'utilisation du texte de lien pour résumer chaque page web, visitée par le *Spider*, à un concept.

Pendant l'analyse HTML des pages web, le *Spider* collecte les textes de liens des pages destination. En terme de balise HTML, un texte de lien apparaît enfermer entre <a> et </a>. A titre d'exemple, le texte « exemple de texte de lien » est un texte de lien associé au document index.html :

<a href= « index.html » > exemple de texte de lien </a>.

Notons que les textes de lien nulles ou appartenant à une liste de stopwords prédéfinie sont écartés et seront remplacés par les titres des pages destinations.

- **Elimination des liens de navigation**

D'une façon générale, les hyperliens possèdent deux fonctions, une pour des convenances de navigation et l'autre pour la connexion des pages web ayant une relation sémantique.

Pour la structure de contenu d'un site web, nous n'aurons besoin que des liens sémantiques, d'où la nécessité d'éliminer des liens de navigation.

En effet, l'information encodée dans une URL peut être exploitée. Dans les URLs, les répertoires sont toujours séparés par des slashs. En se basant sur la structure des répertoires, les liens peuvent être classifiés tel que illustré dans la figure 24.

- Lien upward : la page destination est dans un répertoire parent.
- Lien downward : la page destination est dans un sous répertoire.
- Lien crosswise : la page destination est dans un autre répertoire autre que parent ou sous répertoire.
- Lien outward : la page destination est dans un autre site.

Sur l'optique de cette classification, un lien est dit de navigation si sa direction est vers le haut (upward). Ceci est justifié par le fait que les liens upwards fonctionnent en tant que retour à la page précédente.

Notons que notre *Spider* assure le passage des URLs relatives aux URLs absolues avant de procéder à leur analyse structurelle.

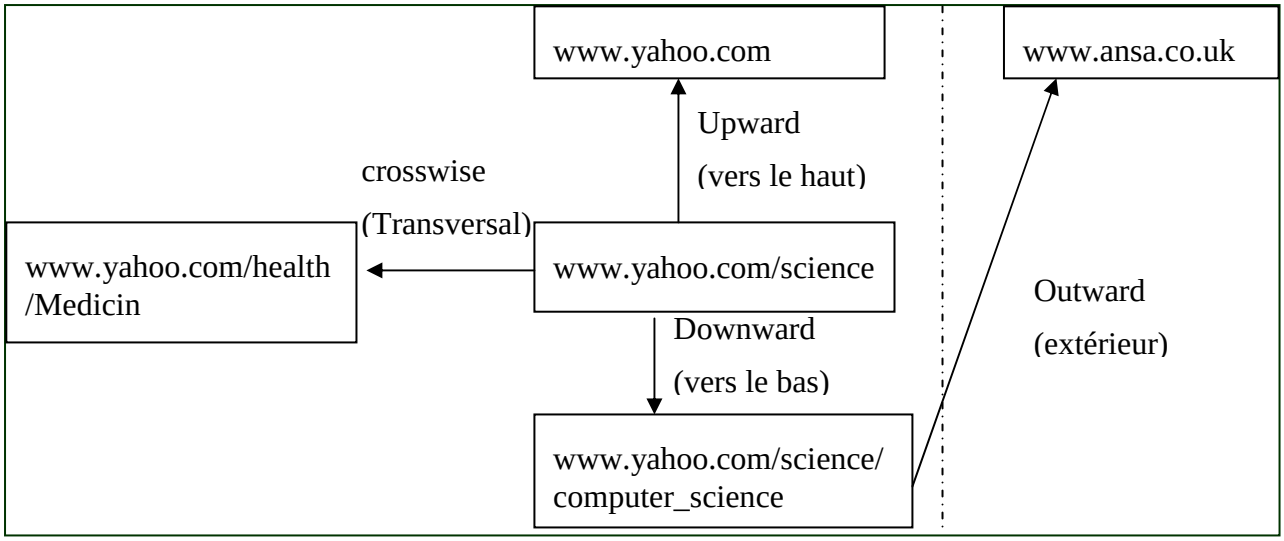



**Figure 24. : Les directions des hyperliens**

- **Découvertes des relations sémantiques entre concepts**

Dans l'étape précédente, nous avons éliminé les liens de navigation de la structure de liens. Le reste des liens est considéré comme des relations sémantiques entre concepts.

Pour faire la discrimination entre ces relations sémantiques, nous avons utilisé les règles énumérées dans (Chen et al., 2003) :

- Un lien dans une page de contenu traduit une relation d'association, car une page de contenu représente un concept concret et il est supposé être l'unité d'information minimale qui ne comporte pas des relations d'agrégation avec d'autres concepts.
- Un lien dans une page d'index, souvent, traduit une relation d'agrégation, car les pages d'index fonctionnent comme pages hubs (centrales) pour accéder aux pages de type contenu. Ainsi, elles représentent des concepts plus génériques.
- Si deux pages possèdent des relations d'agrégation dans les deux directions, le type de relation est modifié en association.

De ce qui précède, il est claire que la découverte de type de relation dépend totalement du type de la page web.

Pour réaliser cette tache, nous avons utilisé l'analyse FOM (Function-based Object Model) (Chen et al., 2001) qui s'occupe principalement de la découverte des pages web index et pages web contenu.

Cette analyse FOM stipule que si l'URL d'une page web comporte le texte « index » ou « default » ou dans le cas où l'URL est un répertoire, alors la page web est considérée comme index.

Autrement, on doit calculer le rapport entre la longueur de tous les textes de lien downward et la taille de texte de la page web. Si ce rapport est supérieur à un certain seuil (dans les expérimentations on prend typiquement 0.4) alors la page web est considérée comme page index. Dans le cas contraire la page est de type contenu.

### 4.2.4 Génération du Thésaurus

Dans la méthode traditionnelle de construction automatique de thésaurus, un certain nombre de documents pertinents est sélectionné comme corpus d'apprentissage. Puis une méthode statistique est utilisée pour extraire les relations entre les termes du thésaurus projeté.

Eclairée par cette méthode, un algorithme similaire peut être appliqué aux structures de contenu des sites web qui jouent le rôle des documents d'un corpus d'apprentissage.

Ainsi, le problème de construction automatique d'un thésaurus dédié à un domaine spécifique deviendra principalement le problème d'organisation des concepts des structures de contenu des sites web.

Le processus de génération automatique du thésaurus est décrit selon les points suivants :

- Pré-traitement des textes de liens.
- Organisation des concepts.

#### *4.2.4.1 Pré-traitement des textes de liens*

Les textes de liens qui sont choisis comme résumé des pages web et qui constituent les noeuds du graphe représentant les structures de contenu des sites web diffèrent en matière de format (mots, groupes nominaux, phrase,…).

Dans une perspective de simplifier les calculs statistiques requis dans l'étape suivante, un pré-traitement de ces textes de liens est jugé nécessaire.

Le dit pré-traitement consiste à appliquer des techniques de traitement automatique de la langue (TAL), à savoir, segmentation, filtrage des stop-words et lemmatisation.

D'abord, chaque texte de lien est segmenté en un ensemble d'unités linguistiques. Les délimiteurs, tels que espace et signes de ponctuation sont utilisés.

En outre, un filtre est appliqué pour omettre les stop-words. Dans notre cas, et en plus des stop-words communément utilisés, nous avons rajouté des stop-words utilisés dans le contexte des textes de liens, tels que link, page, click, here, etc…

En fin, dans un but de construire des unités linguistiques normalisées, un algorithme de lemmatisation est appliqué pour chaque unité linguistique des textes de liens. Cette opération consiste à chercher le « lemme » des mots. Autrement dit, débarrasser les mots de leur nombre (singulier, pluriel), leur personne et leur mode (impératif, indicatif,…).

Dans notre contexte, nous nous sommes limités à la langue anglaise et nous avons appliqué l'algorithme de lemmatisation de M. Porter.

Comme résultat de cette opération de pré-traitement, chaque texte de lien lié à un nœud est formalisé comme suit : $t_{li} = [t_{i1}, t_{i2}, \ldots, t_{im}]$.

Où $t_{li}$ est le $i^{ème}$ texte de lien dans la structure de contenu et le $t_{ij}$ (j=1, …, m) est le $j^{ème}$ terme de $t_{li}$.

#### *4.2.4.2 Organisation des concepts*

A ce stade, chaque site web est représenté par une structure de contenu. L'extraction de relations sémantiques, reliant les différents termes, à partir des ces structures de contenu peut être plus complexe que celle à partir des documents classiques ; ceci est du à la structure non linéaire des structures de contenu. Autrement dit, la séquence de mots doit être prise en compte lors du processus d'extraction des relations sémantiques.

En ce qui concerne les types de relations à extraire, nous n'avons considéré que le type hiérarchique, car dans notre contexte, où le thésaurus va être utilisé pour l'expansion de requêtes, les relations d'associations n'apportent aucune valeur ajoutée au facteur de précision (Chen et al., 2003).

Rappelons que chaque structure de contenu est analogue à un document classique. Le mécanisme de la fenêtre glissante peut être projeté comme suit : au niveau de chaque nœud de la structure de contenu, nous définissons un sous-arbre avec une certaine profondeur qui ne doit pas dépasser la profondeur de la structure de contenu elle-même.

Pour la relation hiérarchique enfant, la fenêtre glissante est donnée par la formule suivante :

$SA_i = (n_i, enfant_1(n_i), \ldots, enfant_d(n_i))$.

Où $SA_i$ est le sous-arbre pour calculer la relation enfant pour le nœud $n_i$, $enfant_d$ représente le nœud enfant de niveau d dans la structure de contenu.

Selon la formule précédente, il est évident de déduire que $SA_i$ n'est que l'ensemble des termes des textes de liens associés à ces nœuds. Ces termes seront soumis aux calculs statistiques dans un but de découvrir la puissance des associations qui les relient.

- **Calcul de l'information mutuelle**

Nous avons choisi d'utiliser l'information mutuelle, un concept de la théorie de l'information, largement utilisé pour mesurer le taux d'association entre les termes d'un corpus. L'information mutuelle de deux points (termes) $t_i$ et $t_j$ est définie comme suit (Thomas et Joy, 2006) :

$$MI(t_i, t_j) = P(t_i, t_j) * \log \frac{P(t_i, t_j)}{P(t_i) * P(t_j)}$$

Où $P(t_i, t_j)$ représente la probabilité que les termes $t_i$ et $t_j$ apparaissent ensemble dans un sous-arbre. $P(t_i, t_j)$ est définie comme suit :

$P(t_i, t_j) = \frac{D(t_i, t_j)}{\sum_k \sum_l D(t_k, t_l)}$, $D(t_i, t_j)$ est le décompte du nombre de fois que les termes $t_i$ et $t_j$ apparaissent ensemble dans un sous-arbre.

P(t) représente la probabilité que le terme t apparaît dans un sous-arbre, elle est définie par la formule suivante :

$P(t) = \frac{D(t)}{\sum_k D(t_k)}$ , D(t) est le décompte du nombre de fois que le terme t apparaît dans un sous-arbre.

D'une façon informelle, l'information mutuelle compare la probabilité d'observer, à la fois, $t_i$ et $t_j$ avec les probabilités d'observer indépendamment $t_i$ et $t_j$. S'il existe une véritable relation entre $t_i$ et $t_j$, alors la probabilité conjointe $P(t_i, t_j)$ sera nettement supérieur au produit $P(t_i)* P(t_j)$, et par conséquence $MI(t_i, t_j)>>0$.

Dans le cas où il n'existe pas de relation intéressante entre $t_i$ et $t_j$, alors $P(t_i, t_j) \approx P(t_i)* P(t_j)$, ainsi $MI(t_i, t_j) \approx 0$.

Par contre, si $t_i$ et $t_j$ existent dans des distributions complémentaires, alors $P(t_i, t_j)$ sera nettement inférieur au produit $P(t_i)* P(t_j)$, ce qui donne $MI(t_i, t_j)<<0$.

- **Calcul de l'entropie**

Nous avons appliquer la mesure de l'information mutuelle au niveau de chaque sous-arbre, mais notre corpus est composé de plusieurs sous-arbres, d'où la nécessité d'une autre mesure qui s'intéresse à évaluer la relation entre une paire de terme tenant compte des différentes distributions.

Dans notre projet, nous avons utilisé l'entropie, un concept de la théorie de l'information qui trouve ses origines dans le domaine de la thermodynamique. L'entropie est utilisée pour mesurer la distribution des paires de termes dans les différents sous-arbres. Elle est définie comme suit :

$$entropie(t_i, t_j) = -\sum_{k=1}^{N} P_k(t_i, t_j) * \log\left(P_k(t_i, t_j)\right)$$

Où N représente le nombre total des sous-arbres, $P_k(t_i, t_j)$ représente la probabilité que $t_i$ et $t_j$ co-occurrent dans un sous arbre. Elle est donnée par la formule suivante :

$$P_k(t_i, t_j) = \frac{D(t_i, t_j / SA_k)}{\sum_{l=1}^{N} D(t_i, t_j / SA_l)}$$

Où $D(t_i, t_j / SA_k)$ décompte le nombre de fois où $t_i$ et $t_j$ co-occurrent dans un sous arbre $SA_k$.

- **Calcul de la similarité.**

L'information mutuelle a permis de mesurer l'association entre une paire de terme au sein d'un sous-arbre. L'entropie, de sa part, a mesuré la distribution des paires de termes au niveau de la totalité des sous-arbres. Une combinaison de ces deux mesures (Chen et al., 2003) engendre une mesure de similarité pour une paire de terme :

$$Sim(t_i, t_j) = MI(t_i, t_j) * \frac{entropie(t_i, t_j) + 1}{\alpha * \log(N)}$$

Log(N) est choisi pour normaliser l'entropie, sachant que $entropie(t_i, t_j)$ varie entre 0 et Log(N). $\alpha$ est un paramètre de réglage, il sert à ajuster l'importance accordée à l'information mutuelle par rapport à l'entropie.

Une fois que la similarité soit calculée pour chaque paire de termes, celle excédant un certain seuil prédéfini est sélectionnée pour la construction du thésaurus pour la relation hiérarchique enfant.

- **Représentation du thésaurus**

Pour être géré, un thésaurus doit être emmagasiné dans une base de données relationnelle, en format XML ou en format fichier texte structuré.

Nous avons choisi le modèle relationnel pour représenter notre thésaurus dont le modèle entité-association est donné comme suit (figure 25.).

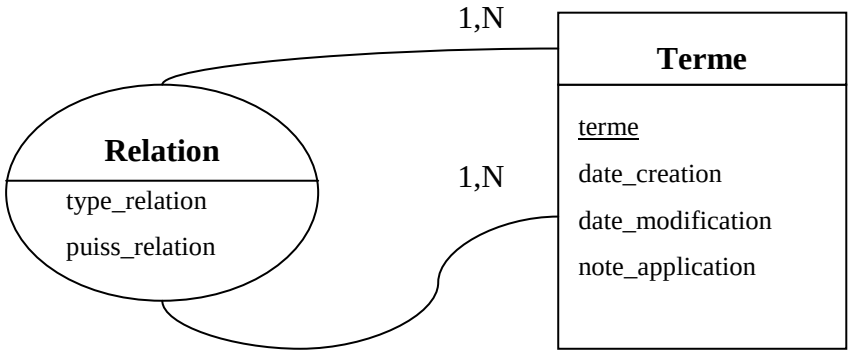



**Figure 25. : Modèle entité-association du thésaurus**

Comme tous les termes du thésaurus sont soumis au processus de lemmatisation, l'attribut *terme* peut être choisi comme identificateur de l'entité *Terme*. L'attribut *type_relation* fait référence aux relations hiérarchiques, équivalences et associations. L'attribut *puiss_relation* est, en effet, la valeur de la similarité calculée pour la paire de terme, il sert à sélectionner les termes candidats lors du processus d'expansion des requêtes.

Il est à signaler que l'attribut note_application est à éditer par l'utilisateur. Le reste des attributs est généré automatiquement.

Par ailleurs, pour assurer le partage et la reutilisabilité des données de notre thésaurus, nous avons pensé au langage RDF (Resource Description Framework). RDF est un langage du web sémantique qui fournit un formalisme de données simple pour modéliser les objets, leurs propriétés ainsi que les relations inter-objets.

La figure 26. montre un extrait de notre thésaurus, pour le terme « e-learn » sous forme d'un graphe RDF utilisant le vocabulaire SKOS Core.

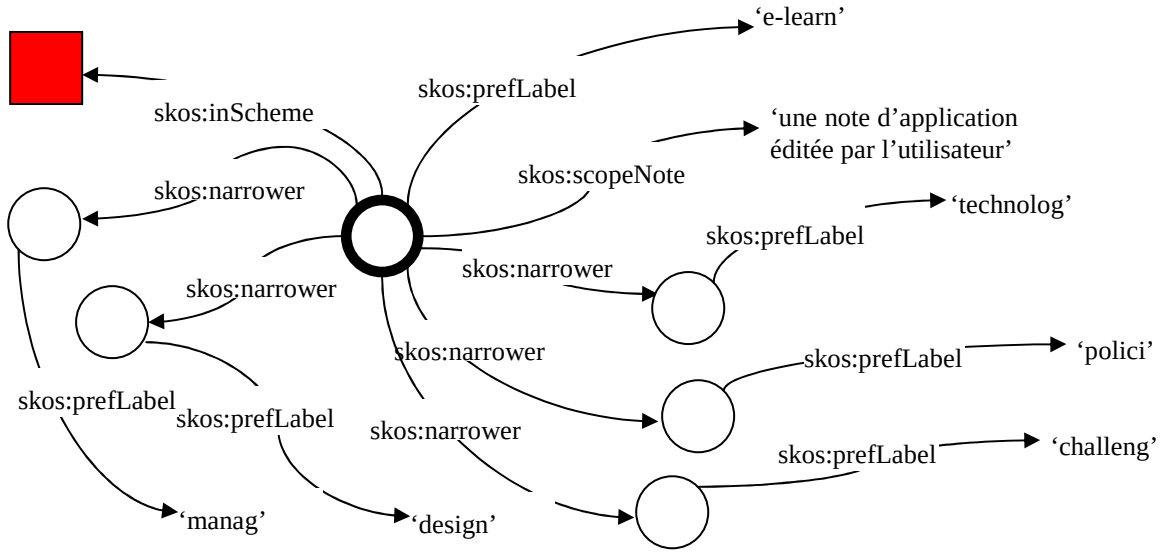



**Figure 26. : Graphe RDF d'un extrait du thésaurus**

Chaque cercle du graphe représente un concept du thésaurus, le carré représente le thésaurus lui-même. Chaque concept du thésaurus possède un URI (Universal Resource Identifier). L'allocation des URIs pour les concepts leur permet d'être référenciés sans ambiguïté.

Une sérialisation RDF/XML de la description RDF du terme « e-learn » de notre thésaurus est illustrée par le listing 7.

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:skos="http://www.w3.org/2004/02/skos/core#">
    <skos:Concept
rdf:about="http://bellaouars.googlepages.com/thesaurus/concept/1440">
        <skos:prefLabel>e-learn</skos:prefLabel>
        <skos:scopeNote>une note d'application éditée par
l'utilisateur</skos:scopeNote>
        <skos:narrower
rdf:resource="http://bellaouars.googlepages.com/thesaurus/concept/1708"/>
        <skos:narrower
rdf:resource="http://bellaouars.googlepages.com/thesaurus/concept/3144"/>
        <skos:narrower
rdf:resource="http://bellaouars.googlepages.com/thesaurus/concept/2020"/>
        <skos:narrower
rdf:resource="http://bellaouars.googlepages.com/thesaurus/concept/2108"/>
        <skos:narrower
rdf:resource="http://bellaouars.googlepages.com/thesaurus/concept/3155"/>
        <skos:inScheme
rdf:resource="http://bellaouars.googlepages.com/thesaurus/"/>
    </skos:Concept>
</rdf:RDF>
```

**Listing 7. : Sérialisation RDF/XML de la description RDF du concept « e-learn »**

## 4.3 Utilisation du Thésaurus

L'utilisation de notre thésaurus est traitée selon deux angles de vue. Le premier est l'objectif de notre projet, utiliser le thésaurus comme outil d'expansion de requêtes dans le processus d'interrogation de la RI. Tandis que le second se préoccupe du parcours et de la visualisation du thésaurus.

### 4.3.1 Utilisation du thésaurus pour l'expansion de requêtes

Le point d'entrée de notre application est bien la spécification d'une requête via l'*Interface Utilisateur (UI)* qui est de type faire-faire (figure 22.). Nous avons choisi la formule textuelle (figure 27.) qui est sans doute le point d'entrée le plus courant. La plupart des moteurs de recherche comme Google et AltaVista invitent l'utilisateur à saisir une requête.

Dans notre contexte, l'*Interface Utilisateur (UI)* transmet la requête au *Gestionnaire de Requêtes* qui procède à sa segmentation, filtrage des stopwords et à sa lemmatisation. Ensuite, il demande au *Gestionnaire de Thésaurus* de proposer les termes candidats pour l'expansion de la requête. Une fois la requête étendue, elle sera transmise aux moteurs de recherche. Les résultats sont interceptés par le *Gestionnaire de réponses* qui s'occupe de leur présentation.

Nous avons choisi la présentation des résultats sous forme d'une liste qui est la forme la plus simple et la plus courante (figure 27.). Cette forme s'appelle aussi présentation en une dimension. Cette dimension correspond, généralement, à l'ordre de pertinence calculé par le système. Cette présentation, souvent textuelle, est celle employée par tous les moteurs de recherche.

A titre d'illustration, le tableau 8. montre des exemples d'expansion de requêtes proposés par notre thésaurus.

| Requête initiale | Requête étendue |
|---|---|
| Computer-based training | Computer-based training collabor develop map privacy confer site online |
| Learning management system | Learning management system circuit confer evalu elearnspac start blog email |
| m-learning | m-learning perform astd circuit learn |

**Tableau 8. : Exemples d'expansion de requêtes**

Sachant que la lemmatisation peut améliorer le rappel, mais au prix de la précision. Pour limiter cet effet indésirable, nous avons décidé que les termes de la requête initiale gardent leurs formes dans la requête étendue.

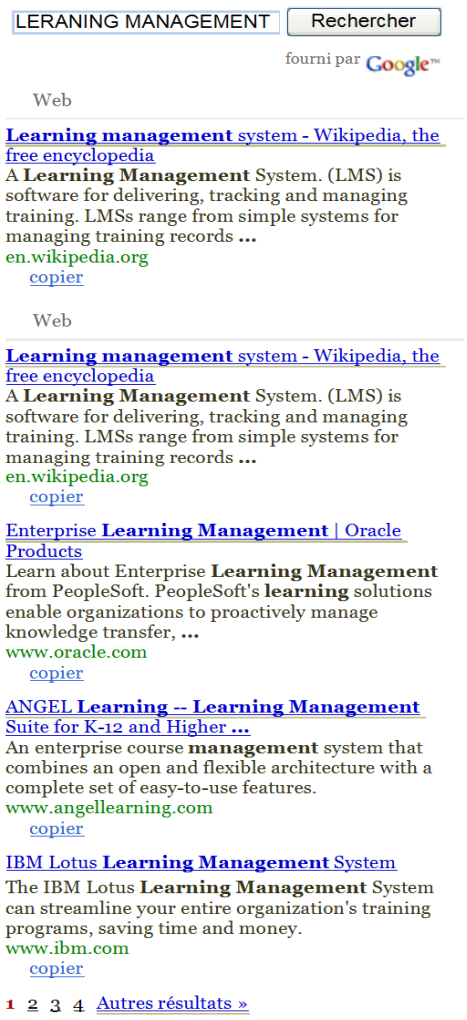



**Figure 27. : Point d'entrée et représentation des résultats**

### 4.3.2 Parcours et visualisation du thésaurus

Le parcours et la visualisation du thésaurus sont assurés par l'open source Thmanager[1]. Notre système procède à l'exportation du thésaurus construit sous format RDF à Thmanager qui permet à son tour la visualisation du thésaurus à l'aide du module *Thesaurus Viewer* (Figure 28.).

La vue principale montre complètement les informations concernant le concept sélectionné et permet à l'utilisateur de parcourir le thésaurus par le biais des hyperliens. En plus, l'utilisateur peut choisir entre trois onglets :

- Tree : visualisation hiérarchique du thésaurus.
- List : visualisation du thésaurus dans un ordre alphabétique.
- Search : permet une recherche dans les libelles préférés des termes du thésaurus selon le type de recherche 'exact match', 'start with', ou 'contains'.

[1] http://thmanager.sourceforge.net/

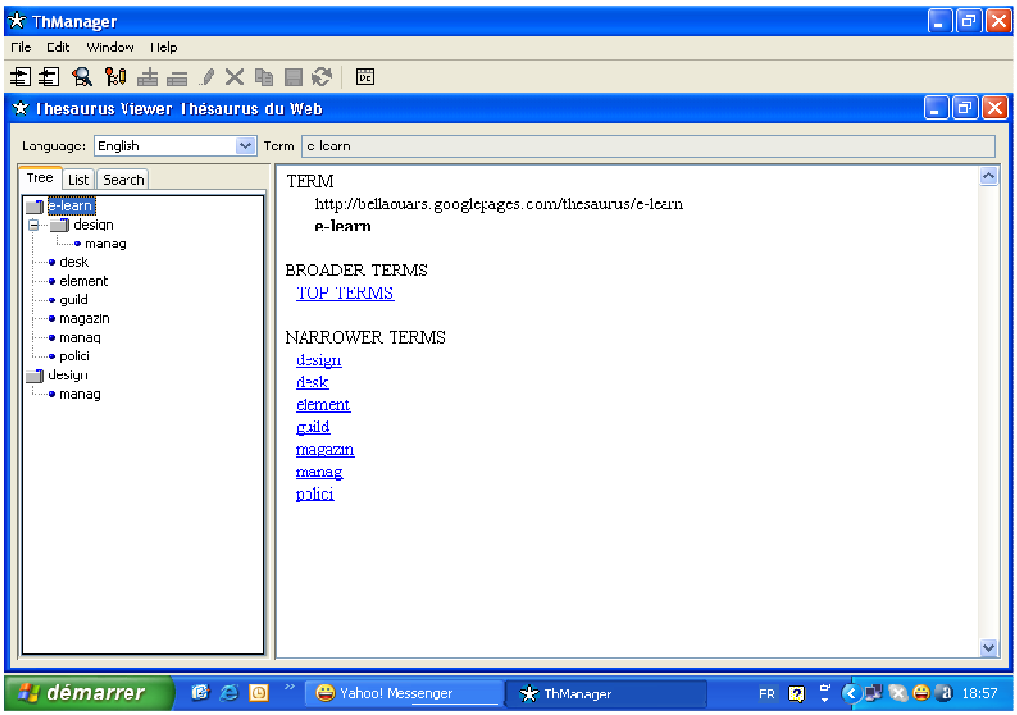



**Figure 28. : Interface de visualisation du thésaurus**

## 4.4 Détail d'implémentation

Cette section se focalise sur la partie implémentation de la solution proposée. Nous avons utilisé la plate forme Eclipse RCP[1] (Ritch Client Platform) version 3.3.2 pour le développement en Java des applications clientes.

La figure 29. montre un diagramme de paquets de haut niveau, à travers le quel nous allons illustrer un certain détail d'implémentation.

### 4.4.1 Interface graphique utilisateur

Ce paquet contient essentiellement deux classes. La première joue le rôle d'un point d'entrée pour la construction du thésaurus et l'autre pour l'utilisation du thésaurus lors de l'expansion de requêtes pendant le processus d'interrogation.

### 4.4.2 Méta moteur de recherche

Ce paquet joue le rôle d'un dispatcher qui contrôle le passage d'un paquet à un autre.

Lors du processus de construction du thésaurus, il prend en charge la requête en entrée (nom du domaine) , interroge le moteur de recherche, passe les sites web résultats à *l'analyseur HTML*, présente les structures de liens des sites web au constructeur de structures de contenu, demande au paquet de *traitement langage naturel* de procéder à la segmentation, filtrage des stop-words et lemmatisation des textes de lien, pour enfin solliciter le paquet *constructeur thésaurus* de procéder à l'organisation des concepts.

Pour le processus d'expansion de requêtes, le paquet *méta moteur de recherche* intercepte la requête en entrée, demande au paquet de *traitement langage naturel* de procéder à la segmentation, filtrage des stop-words et lemmatisation des termes de la requête. Ensuite, il demande au paquet *gestionnaire thésaurus* de proposer les termes similaires dans le thésaurus pour construire une

[1] http ;//www.eclipse.org/platform

requête étendue qui sera transmise à l'*interface moteur de recherche.* Les résultats de la recherche sont envoyés au paquet *interface graphique utilisateur* pour leur présentation.

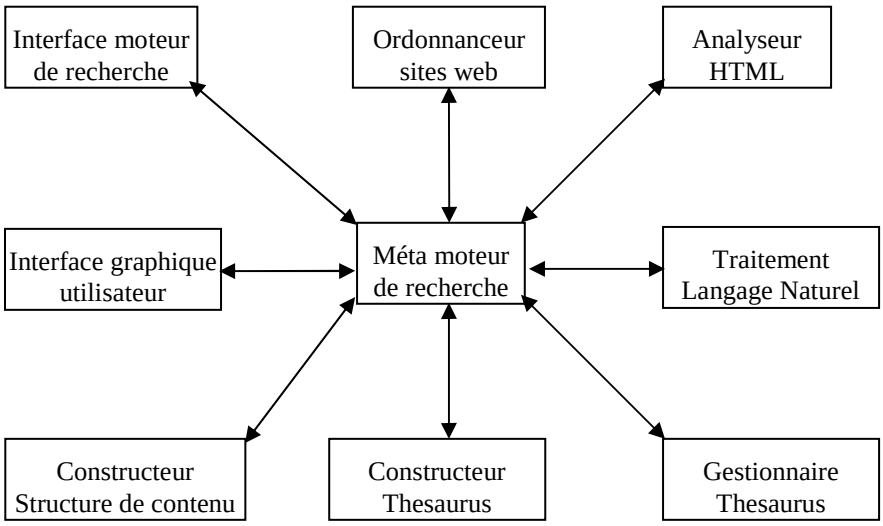



**Figure 29. : Diagramme de paquet de haut niveau**

### 4.4.3 Interface méta moteur de recherche

Les classes de ce paquet interagissent avec le moteur de recherche google. Ces classes utilisent l'API que google fournit. Cette API est basée sur le protocole SOAP (Simple Object Acces Protocol) qui est un protocole d'appel d'objets à distance (Remote Procedure Call, RPC) orienté objet bâti sur XML.

Notons qu'avant de développer l'interface méta moteur de recherche, il faut s'inscrire auprès de google et obtenir une clé. Ensuite, il faut récupérer les classes java constituant l'API de google.

Dans notre cas, nous avons utilisé les classes *GoogleSearch* et *GoogleSearchResult* qui donnent accès aux fonctions de recherche des pages web sur google.

Les méthodes utilisées sont présentées dans le listing 8.

```
GoogleSearch googleServer= new GoogleSearch();
googleServer.setKey("H/ufNfpQFHIXOhEAuGHVIRl59K66koWo");
googleServer.setQueryString("elearning");
googleServer.setLanguageRestricts("lang_en");
GoogleSearchResult r = googleServer.doSearch();
```

**Listing 8. : Les méthodes de recherche des pages web sur google**

Pour outrepasser la contrainte qui limite le nombre des résultats à dix, nous avons procédé à l'appel de la méthode de recherche *doSearch()* autant de fois que nécessaire mais avec un changement d'indice par le biais de la méthode *SetStartResult(int debut).*

### 4.4.4 Ordonnanceur des sites web

Le rôle de ce paquet est d'obtenir des sites web de haute qualité dédiés à un domaine. Les classes de ce paquet appliquent l'algorithme HITS aux sites web renvoyés par google.

Dans notre implémentation, l'ensemble racine de départ est constitué de 200 premières URLs proposées par google. Le graphe de voisinage est construit par l'extension de cet ensemble racine. Sa taille a atteint 3569 nœuds (URLs). Pour un seuil de $10^{-3}$ des scores autorités et hubs, 29 sites web sont élus pour former un corpus à partir du quel le thésaurus sera construit.

Notons que dans la pratique, le calcul des scores autorités et hubs converge au bout d'un maximum de 25 itérations.

### 4.4.5 Analyseur HTML

Pour réaliser l'analyse HTML des pages web, nous avons utilisé l'open source HyperSpider[1]. Une modification au niveau de certaines classes est opérée pour la prise en compte des textes de lien comme résumés sémantiques des pages web.

Les classes de ce paquet reçoivent le contenu des pages web d'un domaine de la part de *l'Ordonnanceur des sites web* (par le biais du paquet *Méta moteur de recherche*). Pour être analysé, le contenu d'une page web doit être passé à un objet Java *HTML Parser*. Dans notre implémentation, nous avons utilisé le *HTML Parser* Swing. Ceci nous permet de parcourir toutes les balises HTML d'une page web.

La classe *LinksExtractor* du paquet *Analyseur HTML* est une extension de la classe Java ParserCallback. Elle surcharge plusieurs méthodes Java qui sont appelées à chaque fois qu'un type de balise HTML est trouvé. Notre implémentation est concernée par deux méthodes, *handleStartTag* et *handleText*.

*handleStartTag* est responsable de l'extraction des hyperliens à partir des documents HTML. Elle doit détecter la présence de l'attribut 'href' (Hypertext REFerence) dans la balise en cours d'analyse. La balise HTML.Tag.A (classe Java HTML.Tag) et l'attribut HTML.Attribute.href (classe Java HTML.Attribute) sont utilisés pour identifier la présence d'un hyperlien.

Dans le cas où un hyperlien est découvert, la méthode *handleText* est utilisée pour extraire le texte de lien correspondant.

Le résultat de l'*Analyseur HTML* est de produire une structure de liens pour chaque site web en entrée.

### 4.4.6 Constructeur structure de contenu

Ce paquet se préoccupe du passage des structures de liens générées par l'*Analyseur HTML* aux structures de contenu. Ce passage apporte une sémantique à la structure de liens qui exprime l'intention de l'auteur d'un site web.

Dans le cas étudié, notre C*onstructeur structure de contenu*, pour une profondeur de trois a traité 45108 nœuds. Chaque nœud représente une page web dont le texte de lien présente un résumé sémantique.

L'ensemble des textes des liens d'un site web forme une structure de contenu qui joue le rôle d'un document dans les collections traditionnelles.

### 4.4.7 Traitement langage naturel

Les textes des liens diffèrent d'un nœud à un autre du point de vue structure de phrase et nombre de mots d'où un traitement spécifique est appliqué au niveau de la segmentation, filtrage des stop-words et lemmatisation.

Un autre rôle de ce paquet est de procéder à la lemmatisation de la requête de l'utilisateur.

L'open source porter-java[2] est utilisé pour la lemmatisation. Il implémente l'algorithme de M. Porter.

A ce stade, la construction d'un sous arbre de type enfant pour chaque nœud est réalisée.

[1] http://sourceforge.net/projects/hyperspider/

[2] http://www.dcs.gla.ac.in/idom/ir_resources/linguistic.util/porter.java

### 4.4.8 Constructeur thésaurus

Ce paquet s'intéresse au calcul de la similarité entre les paires de termes des sous arbres. 277268 paires de termes sont considérées. Pour un seuil de similarité de 0.002, 3330 paires de termes sont retenues pour construire le thésaurus.

Par ailleurs, les classes de ce paquet assurent la présentation du thésaurus sous forme d'une base de données relationnelle. Elles utilisent la technologie JDBC (Java DataBase Connectivity) permettant de se connecter, dans notre cas, à une base de données MYSQL.

Les étapes techniques de la construction de la base de données représentant le thésaurus sont décrites dans l'annexe A.

### 4.4.9 Gestionnaire thésaurus

Le rôle principal de ce paquet est de réaliser l'expansion des requêtes lors du processus d'interrogation.

Le *Gestionnaire de thésaurus* ne doit pas dépasser 10 termes par requête. Contrainte imposée par l'API de recherche google.

En plus de l'expansion des requêtes, les classes de ce paquet assurent l'exportation du thésaurus sous format RDF/XML, ceci pour un éventuel parcours et visualisation par l'open source Thmanager.

# Conclusion

La RI sur le web présente plusieurs défis par rapport à la RI classique. Parmi ces défis, la croissance dramatique du contenu du web et des internautes, l'aspect dynamique du web, la duplication du contenu, etc.

De plus, les problèmes de discordances de mots entre auteur et lecteur des pages web ainsi que la caractéristique de non expressivité des requêtes participent sérieusement à la divergence entre la pertinence calculée par les SRIs et celle que l'utilisateur donne aux documents.

Pour améliorer la qualité des moteurs de recherche d'information sur le web, dans ce travail, nous avons projeté à réduire, le plus possible, la distance entre la pertinence système et la pertinence utilisateur.

La stratégie adoptée est celle de l'expansion des requêtes à base de thésaurus dédié. Elle est appliquée en recherche d'information sur le web dans le domaine e-learning.

A travers ce travail,

- Pour construire automatiquement le thésaurus, nous avons évoqué l'aspect hyperlien qui est une caractéristique discriminante entre une page web et un document textuel classique.
- Nous avons construit automatiquement le thésaurus. Nous avons utilisé les techniques de web mining et de traitement automatique de la langue naturel (TALN) : L'algorithme HITS (Hyperlink Induced Topic Search) est appliqué pour la collection des sites web de haute qualité. Les techniques d'analyse de liens sont utilisées pour générer les structures des liens des sites web collectés. Les textes de liens sont choisis comme résumés sémantiques des pages web destination, un algorithme de lemmatisation est appliqué pour générer des termes normalisés. Avec la découverte des liens sémantiques entre les termes normalisés des textes de liens, les structures de contenu sont construites. Les mesures de l'information mutuelle et de l'entropie sont appliquées sur les structures de contenu pour extraire les termes candidats du thésaurus projeté.

  Telle que réalisée, l'approche proposée est capable d'extraire les nouveaux termes d'un domaine spécifique ainsi que les relations les reliant au fur et à mesure que le web progresse.
- Nous avons développé un méta moteur de recherche pour assurer l'interaction entre l'internaute, notre thésaurus et les moteurs de recherche. La plate forme Eclipse RCP pour le développement en Java est utilisée.
- Nous avons utilisé le thésaurus construit comme outil d'expansion automatique de requêtes lors du processus d'interrogation de la RI sur le web.
- Nous avons assuré l'exportation des données de notre thésaurus vers d'autres gestionnaires de thésaurus par une représentation au format RDF/XML utilisant le vocabulaire SKOS Core. Ceci les rend à la fois partageables et réutilisables.

Comme perspectives à ce travail,

- Nous envisageons d'effectuer des évaluations expérimentales de toutes les phases de construction du thésaurus, à savoir, la collection des sites web de haute qualité, la construction des structures de contenu ainsi que la génération proprement dite du thésaurus. Ceci nous permet de mesurer la qualité du thésaurus construit.
- Notons que l'approche est basée sur l'analyse des liens, donc elle peut être appliquée sur les fichiers logs de requêtes historisant la navigation utilisateur pour construire des thésaurus personnalisés, qui peuvent participer à l'adaptabilité des SRIs.

- Le format RDF/XML permet à notre thésaurus d'être parcouru et visualisé par les utilisateurs. Ainsi, il pourra être utilisé dans le processus de la formulation des requêtes initiales.
- Comme le web est multilingue, nous proposons d'étendre notre thésaurus pour être un thésaurus dédié multilingue.
- Notre méta moteur, actuellement, utilise uniquement les résultats de google. D'autres moteurs de recherche sont disponibles tels que Yahoo, MSN, Askjeeves, Altavista etc. Ils peuvent être ajoutés au méta moteur dans une perspective d'augmenter la couverture du web. La conception de notre méta moteur permet de simples modifications.
- Le passage de notre thésaurus à une ontologie peut ouvrir des portes sur le web sémantique. Nous pensons que l'utilisation des ontologies dans la RI peut améliorer, considérablement, la qualité des moteurs de recherche sur le web.

# Bibliographie


Abdelali A., Cowie J., Soliman H.S., (2004). Arabic Information Retrieval perspectives. JEP-TALN 2004, Arabic Language Processing - Text & Speech, Avril 2004. Disponible sur http://aune.lpl.univ-aix.fr/jep-taln04/proceed/actes/arabe2004/TAAA13.pdf

Aitchison J., Clarke S. D., (2004). The thesaurus: A Historical Viewpoint, with a Look to the Future. published in: The Thesaurus : Review, Renaissance, and Revision, Ed: ROE S. K., THOMAS A. R., The Haworth Information Press, an imprint of The Haworth, pp. 5-21.

Aswani K.C., Srinivas S., (2006). LATENT SEMANTIC INDEXING USING EIGENVALUE ANALYSIS FOR EFFICIENT INFORMATION RETRIEVAL. Int. J. Appl. Math. Comput. Sci., Vol. 16, No. 4, 551–558. Disponible sur http://matwbn.icm.edu.pl/ksiazki/amc/amc16/amc16411.pdf

Baeza-Yates R., Castilo C., Saint-Jean F., (2004). Web dynamics, structure, and page quality. Web Dynamics: Adapting to Change in Content, Size, Topology and Use, chapter 5. Disponible sur http://citeseer.ist.psu.edu/726557.html

Baeza-Yates R., Ribeiro-Neto B., (1999). Modern Information Retrieval. ACM Press, New York. chapter1 et chapter10 disponibles sur http://www.dcc.ufmg.br/irbook/

Baziz M., (2005). Indexation Conceptuelle Guidée Par Ontoogie Pour La Recherche D'Information. Thèse de doctorat, Institut de Recherche en Informatique de Toulouse.

Befferman D., Berger A., (2000). Agglomerative clustering of a search engine query log. In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 407–416. Disponible sur http://www.dougb.com/papers/kdd.pdf

Behaz A., Djoudi M., (2005). Génération dynamique de documents hypermedias adaptatifs dans un environnement numérique de travail. Revue ARIMA, numéro special CARI'04, 25-53. Disponible sur http://www-direction.inria.fr/international/arima/CARI04/CARI04_02.htm

Benayache A., (2005). Construction d'une mémoire organisationnelle de formation et évaluation dans un contexte e-learning: Le projet MEMORAe. thèse PHD, Université de technologie de Compiègne . Disponible sur http://www.hds.utc.fr/%7Eabenayac/PhD/PhD-Ahcene.pdf

Berry M.W., Drmac Z., Jessup E.R., (1999). Matrices, Vector Spaces, and Information Retrieval. SIAM REVIEW, Vol. 41, No. 2, pp. 335–362. Disponible sur http://www.ryanstephens.com/ir.pdf

Bharat K., Henzinger M.R., (1998). Improved algorithms for topic distillation in a hyperlinked environment. Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Distributed Retrieval, pp. 104-111. Disponible sur http://citeseer.ist.psu.edu/bharat98improved.html

Boughanem M., Kraaij W., Nie J.Y., (2004). Modèles de langue pour la recherche d'information. Dans : Les systèmes de recherche d'informations, majid Ihadjadene (Eds.), Hermes-Lavoisier. Pp.163-182. Disponible sur http://www.iro.umontreal.ca/~nie/IFT6255/modele_langue.pdf

Bourda Y., (2002). Des objets pédagogiques aux dossiers pédagogiques (via l'indexation). Lavoisier, Document numérique, vol. 6, pp. 115-128. Disponible sur www.cairn.info/load_pdf.php?ID_ARTICLE=DN_061_0115.

Brin S., Davis J., Garcia-Molina H., (1995). Copy detection mechanisms for digital documents. In Pro-ceedings of the ACM SIGMOD International Conference on Management of Data, pp. 398-409. Disponible sur http://infolab.stanford.edu/~sergey/copy.ps

Brin S., Page L., (1998). The Anatomy of a Large-Scale Hypertextual Web Search Engine. WWW conference. Disponible sur http://infolab.stanford.edu/oub/papers/google.pdf

Broder A.Z., (1997). On the resemblance and containment of documents. In Proceedings of Compression and Complexity of Sequences, IEEE Computer Society, pp 21-29. Disponible sur ftp://ftp.digital.com/pub/DEC/SRC/publications/broder/positano-final-wpnums.pdf

Bruandet M-F., Chevallet J-P., (2003). Utilisation et construction de bases de connaissances pour la Recherche d'Informations. In Assistance Intelligente à la Recherche d'Information, M.-H. Stefanini, E. Gaussier, Hermes, chapter 3, pp85-118. Disponible sur www-mrim.imag.fr/publications/2003/Chapitre3.pdf

Buckley C., (1996). SMART System Overview. Cornell University Computer System Department.

Callan J.P., Croft W.B., Harding S.M., (1992). The inquery Retrieval System. In Proceedings of the International Conference on Database and Expert Systems Applications, Valence, Espagne. Disponible sur http://citeseer.ist.psu.edu/26307.html

Carriere J., Kazman R., (1997). WebQuery : Searching and visualizing the web through connectivity. In Proceedings of the 6th International World Wide Web Conference. Disponible sur http://www.cgl.uwaterloo.ca/Projects/Vanish/webquery-1.html

Chakrabarti S., (2003). Mining the Web : Discovering Knowledge from Hypertext Data. Morgan Kaufmann Publishers, an imprint for Elsevier Science (USA).

Chebeir R., (2001). Modélisation de la description d'images : application au domaine médical, chapitre I – Généralités. Thèse de doctorat, laboratoire INSA, pp. 22-40. Disponible sur http://docinsa.insa-lyon.fr/these/2001/chbeir/

Chen J.L., Zhou B.Y., Shi J., Zhang H.J., Wu Q.F, (2001) Function-based Object Model Towards Website Adaptation, Proc. of the 10th International World Wide Web Conference, Hong Kong, China, pp. 587-596. Disponible sur http://citeseer.ist.psu.edu/chen01functionbased.html

Chen Z., Liu S., Wenyin L., Pu G., Ma W., (2003). Building a web thesaurus from web link Structure. Proceedings of the 26th annual international ACM SIGIR conference on Research

and development in information retrieval. Toronto, pp. 48– 55. Disponible sur http://research.microsoft.com/~zhengc/papers/p14325-chen.pdf

Chien S., Immorlica N., (2005). Semantic similarity between search engine queries using temporal correlation. In Proceedings of the 14th international conference on World Wide Web, pp. 2–11. Disponible sur http://www2005.org/cdrom/docs/p2.pdf

Cho J., Shivkumar N., Garcia-Molina H., (2000). Finding replicated web collections. In Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 355-366. Disponible sur www.cs.brandeis.edu/~mfc/cs120/papers/sigmod-cho-mirror.ps

Chuang S-L., Pu H-T., Lu W-H., Chien L-F., (2000). Auto-Construction of a live Thesaurus From search Term Logs For interactive web search. In: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 334-336. Disponible sur https://netfiles.uiuc.edu/schuang2/www/papers/slchuang-sigir00-livethesau.pdf

Coste G., (2004). Le papier, un matériau complexe. EFPG/IRFIP, dossier technique. 05 avril 2004, disponible sur http://cerig.efpg.inpg.fr/dossier/papier-materiau/page01.htm.

Cui H., Wen J-R., Nie J-Y., Ma W-Y., (2002). Probabilistic expansion using query logs. In Proceedings of the 11th international conference on World Wide Web, pp. 325–332. Disponible sur http://research.microsoft.com/users/jrwen/jrwen_files/publications/QE-WWW2002.pdf

David H., Heikki M., Padhraic S., (2001). Principles of Data Mining, MIT Press, Cambridge, MA (2001).

Davison. B.D., (2000). Topical locality in the Web. In Proc. Of SIGIR'00, pp. 272-279. Disponible sur http://www.cse.lehigh.edu/~brian/pubs/2000/sigir/sigir2k.pdf

Desjardins G., (2006). MODÉLISATION CONNEXIONNISTE DU REPÉRAGE DE L'INFORMATION. Thèse présentée comme exigence partielle du doctorat en informatique cognitive, université du Québec à Montréal, Août 2006. disponible sur http://www.dic.dinfo.uqam.ca/etudiants/diplomes/Desjardins_these

Dherent C., (2002). Les Archives électroniques. Manuel pratique. Paris, Direction des Archives de France (2002) 104 p.

Diem Le T.H., (2003). Extraction et structuration de connaissances pour la recherche d'information. Rapport de DEA, Groupe MRIM - CLIPS-IMAG. Disponible sur www-mrim.imag.fr/publications/2003/DEA03/Memoire-DEA.pdf

Efthimiadis E.N., (1992). Interactive Query Expansion and Relevance Feedback for Document Retrieval Systems. Ph.D. dissertation. City University, London. Disponible sur http://faculty.washington.edu/efthimis/pubs/Dissertation/Dissertation_ENE.pdf

Efthimiadis E.N., (1996). Query Expansion. In M. E. Williams (Ed.), Annual Review of Information Science and Technology Vol. 31, pp. 121-187. disponible sur http://faculty.washington.edu/efthimis/pubs/Pubs/qe-arist/QE--arist.html

Elayari S., (2005). Moteurs d'indexation et de recherche : vers une recherche intelligente de l'information sur l'Internet. Mémoire de maîtrise en sciences du langage, ILPGA, Université de Paris III - Sorbonne Nouvelle, juillet 2005, disponible sur www.cavi.univ-paris3.fr/Ilpga/ilpga/tal/sitespp/maitrise-2005/memoire-sarra-20005.pdf

Even N., (2003). Qu'est-ce qu'une plate-forme pour la formation ouverte et à distance ?. Disponible sur http://ressourcesv2.e-motive.com/virtual/30/Documents/pdf/plate-forme.pdf

Fagin R., Kumar R., Siyakumar D., (2003). Efficient similarity search and classification via rank aggregation. In Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 301–312. Disponible sur http://www.almaden.ibm.com/cs/people/fagin/sigmod03.pdf

Fetterly D., Manasse M., Najork M., Wiener J.L., (2003). A large-scale study of the evolution of Web pages. In Proceedings of the 12th International WWW Conference. Disponible sur http://research.microsoft.com/research/sv/sv-pubs/pageturner-spe2004.pdf

Grefenstette G., (1993). Automatic thesaurus generation from raw text using knowledge-poor techniques. In Making Sense of Words. Ninth Annual Conference of the UW Centre for the New OED and text Research. Disponible sur http://citeseer.ist.psu.edu/grefenstette93automatic.html

Gong Z., Cheang C-W., Hou U-L., (2005). Web Query Expansion by WordNet. In Proceedings of the 16th international conference on Database and Expert Systems Applications (DEXA 2005), Copenhagen, Denmark, pp. 166-175, August 22-26. Disponible sur http://www.sftw.umac.mo/~fstzgg/dexa2005.pdf

Haddad M.H., (2002). Extraction et Impact des connaissances dur les performances des Systèmes de Recherche d'Information. Thèse de Doctorat, Université Joseph Fourier – Grenoble1, 24 Septembre 2002. Disponible sur http://tel.archives-ouvertes.fr/docs/00/04/60/54/PDF/tel-00004459.pdf

Hawking D., Thistlewaite P., Carswell P., (1997). ANU/ACSys TREC-6 experiments», In: Voorhees, E.M. & Harman, D.K. (Eds.) [TREC 6]: pp. 275-290. Disponible sur http://trec.nist.gov/pubs/trec6/papers/anu.ps.gz

Hearst M.A., (1992). Automatic acquisition of hyponyms from large text corpora. In Proceedings of the Fourteenth International Conference on Computational Linguistics, pages 539--545, Nantes, France, July 1992. Disponible sur http://citeseer.ist.psu.edu/hearst92automatic.html

Henzinger M.R., Motwani R., Silverstein C., (2003). Challenges in Web Search Engines. In Proceedings of the 18th International Joint Conference on Artificial Intelligence, pp. 1573-1579. Disponible sur http://citeseer.ist.psu.edu/henzinger02challenges.html

Hofmann T., (1999). Probabilistic Latent Semantic Indexing, Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval.

Hoi C.H., (2006). Statistical Machine Learning for Data Mining and Collaborative Multimedia Retrieval. The Chinese University of Hong Kong, Ph.D. Thesis, September 2006. Disponible sur http://www.cse.cuhk.edu.hk/~lyu/student/phd/steven/thesis_hoi.pdf

Ingwersen P., Jarvelin K., (2005a). Information Retrieval in context - IRiX. SIGIR workshop report, ACM SIGIR forum, Vol. 39 No. 2, December 2005. Disponible sur http://www.acm.org/sigs/sigir/forum/2005D/2005d_sigirforum_ingwersen.pdf.

Ingwersen P., Jarvelin K., (2005b). THE TURN: Integration of Information Seeking and Retrieval in Context. THE KLUWER INTERNATIONAL SERIES ON INFORMATION RETRIEVAL, Series Editor: W. Bruce Croft, University of Massachusetts, Amherst, Published by Springer.

Jansen B.C., Pooch U., (2000). Web user studies: A review and framework for future work. Journal of the American Society of Information Science and Technology 52(3), pp. 235-246. Disponible sur citeseer.ist.psu.edu/417587.html

Jarvelin K., Kekalainen J., Niemi T., (2001). ExpansionTool: Concept-based query expansion and construction», Information Retrieval, 4(3/4), pp. 231-255. Disponible sur http://www.info.uta.fi/tutkimus/fire/archive/ET3-IR01.pdf

Jarvelin K., Kristensen J., Niemi T., Sormunen E., Keskustalo H., (1996). A Deductive Data Model for Query Expansion. In: H.-P. Frei, D. Harman, P. Schäuble, and R. Wilkinson (eds.) Proceedings of the 19th Annual International ACM‹SIGIR Conference on Research and Development in Information Retrieval, pp. 235-249, Zürich, August 18.-22, 1996. Disponible sur http://www.uta.fi/~likaja/abstracts/JKNSKsigir96.html

Jing Y., Croft W.B., (1994). An association thesaurus for information retrieval, Amhesrt, MA: University of Massachusetts, Dept. of Computer Science, Technical Report TR-1994-17. disponible sur http://citeseer.ist.psu.edu/jing94association.html

Kekalainen J.,(1999). The effects of query complexity, expansion and structure on retrieval performance in probabilistic text retrieval. Tampere, Finland: University of Tampere, Department of Information Studies, Ph.D. Thesis, 1999, disponible sur http://www.info.uta.fi/tutkimus/fire/archive/QCES.pdf

Kekalainen J., Jarvelin K., (2000). The co-effects of query structure and expansion on retrieval performance in probabilistic text retrieval. Information Retrieval, 1(4), pp. 329–344. Disponible sur http://www.info.uta.fi/tutkimus/fire/archive/JK&KJ-IR'00.pdf

Kendal S.L., Creen M., (2006). An Introduction to Knowledge Engineering, Springer, 1 edition, ISBN: 1846284759, 290 pages.

Kleinberg J.M., (1999). Authoritative sources in a hyperlinked environment. Journal of the ACM, Vol. 46, No. 5, pp. 604 –632, September 1999. Disponible sur http://citeseer.ist.psu.edu/87928.html

Kraft R., Zien J.Y., (2004). Mining anchor text for query refinement. In 13th international conference on World Wide Web (WWW), New York, NY, USA. Disponible sur www.soe.ucsc.edu/~rekraft/papers/p462-kraft.pdf

Kristensen J., (1993). Expanding end-users' query statements for Free text searching with a search-aid Thesaurus. Information Processing & Management, 29(6): pp. 733-744, résumé disponible sur http://www.uta.fi/~lijakr/ipm.html

Kristensen J., Jarvelin K., (1990). The Effectiveness of a Searching Thesaurus in Free-Text Searching of a Full-Text Database. International Classification, 17(2):1990, résumé Disponible sur http://www.uta.fi/~lijakr/ic.html

Kumar P., Kashyap S., Mittal A., Gupta S. (2005). A Fully Automatic Question-Answering System for Intelligent Search in E-Learning Documents, International Journal of ELearning (IJEL), AACE Publishers, Volume 4, No. 1, pp. 149-166.

Langville A.N., Meyer C.D., (2004). Deeper Inside PageRank. Internet Mathematics Vol. 1, No. 3: pp. 335-380. Disponible sur http://www.internetmathematics.org/volumes/1/3/Langville.pdf

Langville A.N., Meyer C.D., (2006). Google's PageRank and Beyond: The Science of Search Engine Rankings. Princeton University Press, chapter 1. Disponible sur http://press.princeton.edu/chapters/s8216.pdf.

Larkey L. S., Ballesteros L. and Connel, (2002). Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval.

Lawrence S., Giles C.L., (1998). Searching the World Wide Web. Science, Vol. 280(5360): pp. 98–100, 3 Avril 1998. Disponible sur http://citeseer.ist.psu.edu/lawrence98searching.html.

Lee H-M., Huang C-C., Chao C-Y., (2007). Association Thesaurus Construction for Interactive Query Expansion Based on Association Rule Mining. JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 23, pp. 617-627. Disponible sur http://www.iis.sinica.edu.tw/JISE/2007/200703_16.pdf

Luhn H.P., (1957). A Statistical Approach to Mechanized Encoding and Searching of Literary Information. IBM Journal of Research and Development, 1, pp. 309-317. Disponible sur www.research.ibm.com/journal/rd/014/ibmrd0104D.pdf

Luhn H.P., (1958). Automatic Creation of Literature Abstract. IBM Journal of Research and Development, 2(2): pp. 159-165. Disponible sur http://www.research.ibm.com/journal/rd/022/luhn.pdf

Manning C., Raghavan P., Schütze H., (2007). An Introduction to Information Retrieval. Preliminary draft (c) of January 2, 2007 Cambridge University Press, England, disponible sur http://www-csli.stanford.edu/~schuetze/information-retrieval-book.html.

Markov Z., Larose D.T., (2007). DATA MINING THE WEB Uncovering Patterns in Web Content, Structure, and Usage. WILEY-INTERSCIENCE, A JOHN WILEY & SONS, INC., PUBLICATION.

Mechtri R., (2003). Prise en compte des syntagmes dans le calcul de la fonction de correspondance en recherche d'information. Rapport de DEA, Groupe MRIM - CLIPS-IMAG. Disponible sur http://www-mrim.imag.fr/publications/2003/DEA03/Rapport.pdf

Miles A., Matthews B., Beckett D., Brickley D., Wilson K., Rogers N., (2005). SKOS A language to describe simple knowledge structures for the web, W3C. Disponible sur http://idealliance.org/proceedings/xtech05/papers/03-04-01/

Mitra M., Singhal A., Buckley C., (1998). Improving Automatic Query Expansion experiments, In Proceedings of the 21 st Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Disponible sur http://citeseer.ist.psu.edu/121460.html

Moreau F., (2006). Revisiter le couplage traitement automatique des langues et recherche d'information. Thèse de doctorat, l'université de Rennes 1, décembre 2006. Disponible sur http://www.irisa.fr/texmex/people/moreau/publications/these.pdf

Moukdad H. (2004). Cross-language information on the Web: An exploratory study of an Arabic-English search engine. In Proceedings of the 67th Annual Meeting of the American Society for Information Science and Technology, November 12-17, 2004, Providence, Rhode Island. Disponible sur http://www.asis.org/Conferences/AM04/posters/260.doc

Nakayama K., Hara T., Nishio S., (2007). Wikipedia Mining for An Association Web Thesaurus Construction. International Conference on Web Information Systems Engineering (WISE2007).

Nie J.Y., (2007). Le domaine de recherche d'information – Un survol d'une longue histoire. support de cours Recherche d'Information, Département d'informatique et recherche opérationnelle, Université de Montréal, Hiver 2007. Disponible sur http://www.iro.umontreal.ca/%7Enie/IFT6255/historique-RI.pdf.

Nielsen M. L., (2004). Thesaurus Construction : Key Issues and Selected Readings. Published in: The Thesaurus : Review, Renaissance, and Revision, Ed: ROE S. K., THOMAS A. R., The Haworth Information Press, an imprint of The Haworth, pp. 57-74.

Oakes M., (2007). Thesauri: Ontologies for Information Retrieval. University of Sunderland. Visité en octobre 2007. Disponible sur http://osiris.sunderland.ac.uk/~cs0moa/iistut2.doc

Oard D.W., Gey F.C., (2001). The TREC-2001 Arabic Information Retrieval Evaluation. Disponible sur http://citeseer.ist.psu.edu/456335.html

Oubahssi L., (2005). Conception de plates-formes logicielles pour la formation à distance, présentant des propriétés d'adaptabilité à différentes catégories d'usagers et d'interopérabilité avec d'autres environnements logiciels. Thèse de doctorat de l'Université René Descartes – Paris V.

Page L., Brin S., Motwani R., Winograd T., (1998). The PageRank Citation Ranking: Bringing Order to the Web. Disponible sur http://infolab.stanford.edu/~backrub/pageranksub.ps

Payement F., (2005). Le e-learning . Revue des médias. Université de Valenciennes et du Hainaut -Cambrésis. Disponible sur www.univ-valenciennes.fr/rdm/article.php3?id_article=292.

Picarougne F., (2004). Recherche d'information sur Internet par algorithmes évolutionnaires. Thèse de doctorat, Université François Rabelais Tours, novembre 2004. Disponible sur www.antsearch.univ-tours.fr/publi/picarougne04these.pdf

Piwowarski B., (2003). Techniques d'apprentissage pour le traitement d'informations structurées : application à la recherche d'information. Thèse de doctorat, Université Paris 6, 17 Juillet 2003. Disponible sur www-connex.lip6.fr/download_article/695.pdf.

Ponte J.M., Croft W.B.,(1998). A Language Modeling Approach to Information Retrieval. In Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR), Melbourne, Australie. Disponible sur http://www.iro.umontreal.ca/~nie/IFT6255/ponte-croft.pdf

Porter M. F., (1980). An Algorithm for Suffix Stripping. Program, v. 14(3), pp. 130-137. Disponible sur http://www.tartarus.org/~martin/PorterStemmer/def.txt

Robertson S.E, (1977). The probability ranking principle in IR. Journal of Documentation 33, pp. 294-307. disponible sur http://www.soi.city.ac.uk/~ser/papers/ProbabilityRankingPrinciple.pdf

Robertson S.E, Walker S., (1994). Some simple effective approximations to the 2- Poisson model for probabilistic weighted retrieval. Presented at SIGIR 94, Dublin, In: W.B. Croft and C.J. van Rijsbergen (eds.), SIGIR '94. Springer-Verlag, pp. 232-241. Disponible sur http://www.computing.dcu.ie/~gjones/Teaching/CA437/p232.pdf

Rosenfeld L., Morville P., (2002). Information Architecture for the World Wide Web. 2nd Edition, Designing Large-Scale Web Sites. O'Reilly & Associates, chapter9 : Thesauri, Controlled Vocabularies, and Metadata. Disponible sur http://www.fucinaweb.com/materiale/iaftwww/ch09.pdf

Rouissi S., (2007). Production de document numérique pédagogique dans un contexte normalisé. Actes du colloque Initiatives 2005 [en ligne], Débat thématique 4, 2 mars 2007. Disponible sur http://www.initiatives.refer.org/Initiatives-2005/document.php?id=258.

Schawarkz K., (2005). Domain model enhanced search–A comparaison of taxonomy, thesaurus and ontology. Master of Content and Knowledge Engineering, University of Utrecht. Disponible sur www.cwi.nl/~media/publications/masterthesis_kat_domainmodel_2005.pdf.

SCTIC. (2002). Les normes et standards de la formation en ligne : état des lieux. Rapport réalisé par le Groupe de travail sur les normes et standards de la formation en ligne du sous-comité sur les technologies de l'information et de la communication. Québec : CREPUQ, septembre 39 p. Disponible sur http://profetic.org/file/norm-0210-d-RAPPORT.pdf

Serres A., (2004). Recherche d'information sur Internet : où en sommes-nous, où allons-nous ? Paris : CNDP, SavoirsCDI, Juin 2004. disponible sur http://savoirscdi.cndp.fr/culturepro/actualisation/Serres/Serres.htm

Shen X., Dumais S., Horvitz E., (2005). Analysis of topic dynamics in web search. In Proceedings of the 14th international conference on World Wide Web, pp. 1102–1103. Disponible sur http://www2005.org/cdrom/docs/p1102.pdf

Shiri A-A., Revie C., (2000). Thesauri on the Web: current developments and trends. Online Information Review. Volume 24, No. 4, pp. 273-279.

Simonnot B., (2006). Le besoin d'information : principe et compétences. Actes de la journée Thémat'IC : Information - besoins et usages. Illkirch 17 mars 2006. Disponible sur http://infocom.u-strasbg.fr/~thematic/thematic_site_06/documents/actes/actes_simmonotbrigitte_txt.pdf.

Singhal A., (2001). Modern Information Retrieval: A Brief Overview. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 24(4): pp. 35-43. Disponible sur www.cs.wisc.edu/~cs784-1/ir_overview.pdf.

Spink A., Jansen B.J., Ozmultu H.C., (2000). Use of query reformulation and relevance feedback by Excite users. Internet Research: Electronic Networking Applications and Policy 10(4), pp. 317-328. Disponible sur http://citeseer.ist.psu.edu/spink00use.html

Srivastava J., Desikan P., Kumar V., (2005). Web Mining – Concepts, Applications and Research Directions, Studies in Fuzziness and Soft Computing, Volume 180, pp. 275–307.

Studer R., Benjamins V., Fensel D., (1998). Knowledge engineering: Principles and methods. IEEE Transactions on Data and Knowledge Engineering, 25:161 -- 197. Disponible sur http://citeseer.ist.psu.edu/article/studer98knowledge.html

Tamim M-M., (2001). محمود أحمد تميم « المكانز في الوطن العربي », العربية 3000, 2001, موجود على http://www.arabcin.net/arabiaall/4-2001/8.html

Thomas M.C., Joy A.T., (2006). Elements of Information Theory, second edition, WileyInterscience, John Wiley & Sons, Inc., Publication.

Turtle H.R., (1991). Inference Networks for Document Retrieval. Thèse de doctorat, université de Massachussets, Février 1991, Disponible sur http://citeseer.ist.psu.edu/turtle91inference.html

Turtle H.R., Croft W.B., (1989). Inference Networks for Document Retrieval. Proceedings of the 13th annual international ACM SIGIR conference on Research and development in information retrieval, pp.1-24. Disponible sur http://www.doc.ic.ac.uk/~jmag/classic/1991.Inference%20networks%20for%20document%20retrieval.pdf

Uyttebrouck, E. (2002). WebCT et la normalisation. Rapport de veille, Université Libre de Bruxelles, Centre des Technologies pour l'Enseignement. Disponible sur http://www.profetic.org/file/webct_ims.pdf

Van Rijsbergen C.J., (1979). Information Retrieval. Londres. Disponible sur http://www.dcs.gla.ac.uk/Keith/Preface.html

Vlachos M., Meek C., Vagena Z., Gunopulos D., (2004). Identifying similarities, periodicities and bursts for online search queries. In Proceedings of the ACM SIGMOD international conference on 14th international conference on Management of data, pp. 131–142. Disponible sur http://www.cs.ucr.edu/~mvlachos/pubs/sigmod04.pdf

Voorhees E.M., (1994). Query expansion using lexical-semantic relations. In:Croft, W. B. & van Rijsbergen, C. J. ed. SIGIR 94: Proceedings of 17th International Conference on Research and Development in Information Retrieval, pp. 61-69, 1994, Dublin, Ireland. Berlin: Springer-Verlag. Disponible sur http://www.iro.umontreal.ca/~nie/IFT6255/voorhees-94.pdf

Voorhees E.M., (2003). Overview of TREC 2002. NIST Special Publication: SP 500-251, The Eleventh Text Retrieval Conference (TREC 2002), February 2003. Disponible sur http://trec.nist.gov/pubs/trec11/t11_proceedings.html.

Wong S.K.M., Ziarko W., Wong P.C.N., (1985). Generalized Vector Space Model in Information Retrieval. Proceedings of the 8th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 18-25. Disponible sur http://www.iro.umontreal.ca/~nie/IFT6255/general-vsm.pdf

Xu J., Croft W.B., (1996). Query Expansion Using Local and Global Document Analysis. In Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 4—11. Disponible sur http://citeseer.ist.psu.edu/xu96query.html

Yiping K., Lin D., Wilfred N., Dik-Lun L., (2006). Web Dynamics and their Ramifications for the Development of Web Search Engines. Computer Networks 50(10): pp. 1430-1447. Disponible sur http://www.cs.ust.hk/~wilfred/paper/cnj05.pdf

Zdravko M., Daniel T.L., (2007). Data mining the Web : Uncovering patterns in Web content, structure & usage, WILEY-INTERSCIENCE A JOHN WILEY & SONS, INC.

# Annexe A : Etapes de la construction de la base de données

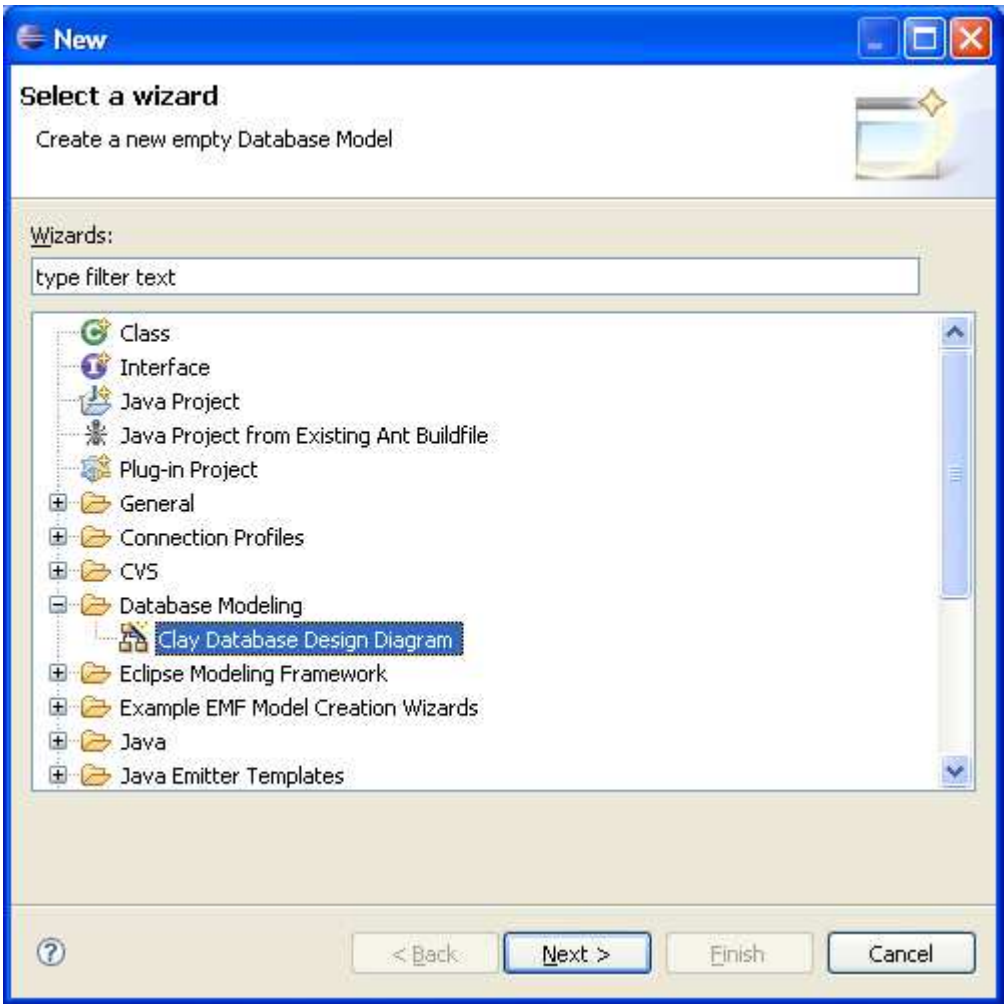
Les étapes techniques de la construction de la base de données représentant notre thésaurus du web peuvent être décrites comme suit :

## A.1 Installation des différents plugins et logiciels

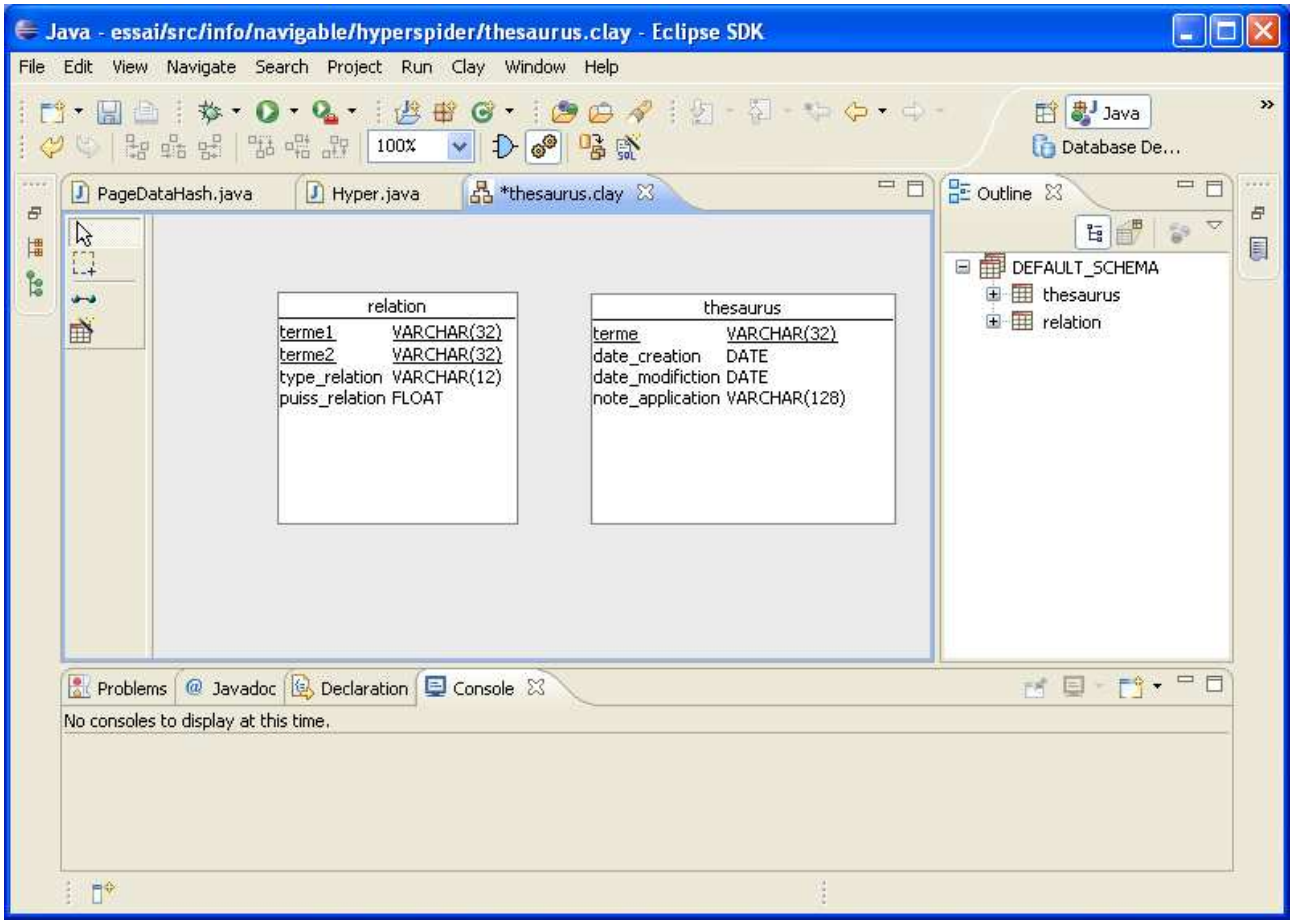
- Télécharger et installer *MySQL*.
- Télécharger et décompresser un driver *MySQL*, qui est nécessaire pour se connecter à MySQL depuis Java. Nous avons utilisé la version 5.1 (mysql-connector-java-5.1.7-bin.jar).
- Télécharger et installer l'outil de modélisation des bases de données *Clay*, nous avons utilisé la version 1.4.2.
- Prise en charge du connecteur *MySQL* et de l'outil de modélisation *Clay* dans l'environnement *Eclipse*.

## A.2 Conception de la base de données

- Création d'un sous répertoire *bd* dans le répertoire du projet.
- Création d'un diagramme de la base de données en utilisant la commande *New/Other*, ensuite la sélection de *Clay Database Design Diagram*.



- En utilisant *Clay*, créer les tables *Thesaurus* et *Relation*.




## A.3 Création des tables par programme

- Connexion à la base de données MySQL

```
StringTokenizer token;
String userName = "root";
String password = "root";
String url ="jdbc:mysql://localhost:3306/thesaurus";
try {
       /* chargement le pilote JDBC pour MySQL, et création d'une instance
       de cette classe */
       Class.forName("com.mysql.jdbc.Driver").newInstance();
} catch (Exception e) {
       System.err.println("Unable to load driver.");
       e.printStackTrace();
       }
       Connection connexion = null;
       try {

      /* Connexion à la base de données thesaurus, création d'une instance
      de la classe Connection grâce à la méthode getConnection de l'objet
      DriverManager en indiquant la base de données à charger à l'aide de son
      URL */
       connexion = DriverManager.getConnection(url, userName, password);
                    System.out.println("Connected to the database");
       } catch (Exception e) {
              e.printStackTrace();
              }
```

- Création des tables *thesaurus* et *relation*

```
/* Création d'une instruction (Statement) JDBC simple */
stmt = connexion.createStatement();
String sqlquery ="CREATE TABLE thesaurus ("
                      +"terme VARCHAR(32) NOT NULL"
                      +", date_creation DATE"
                      +", date_modifiction DATE"
                      +", note_application VARCHAR(128)"
                      +", PRIMARY KEY (terme)"
                      +");";
/* Exécution de l'instruction CREATE TABLE. La méthode executeUpdate est
utilisée */
               stmt.executeUpdate(sqlquery);

sqlquery ="CREATE TABLE relation ("
                      +"terme1 VARCHAR(32) NOT NULL"
                      +", terme2 VARCHAR(32) NOT NULL"
                      +", type_relation VARCHAR(12) DEFAULT 'hierarchique'"
                      +", puiss_relation FLOAT DEFAULT 0"
                      +", PRIMARY KEY (terme1, terme2)"
                      +");";
               stmt.executeUpdate(sqlquery);
```

- Edition des tables

```
/* Création et ouverture d'un lecteur de tompan vers le fichier texte
resultats_pertinents.txt contenant les paires de termes pertinents*/

lecteurAvecBuffer = new BufferedReader(new FileReader(project-
folder+"\\resultats_pertinents.txt"));

/* Lecture de la ligne courante */
termes = lecteurAvecBuffer.readLine();

while ((termes = lecteurAvecBuffer.readLine()) != null) {
/* Traitement de la ligne courante */
 token = new StringTokenizer(termes, "   ");
 terme1 = token.nextToken();
 terme2 = token.nextToken();
/*parcours des élements (token) de cette ligne avec un objet StringTokenizer*/
 while (token.hasMoreTokens ()){
        s = token.nextToken();
        }
 sim = Double.parseDouble(s);
 try {
 results = stmt.executeQuery("SELECT terme, COUNT(*) FROM thesaurus WHERE
terme = '"+terme1.toString()+ "'");
 results.next();
 int rowcount=results.getInt(2);
 if (rowcount==0){
/* Création d'une instruction (Statement) JDBC paramétrée et pré-compilée pour
un ordre donné. Elle est associée à la table thesaurus */
 pstmtthesaurus = connexion.prepareStatement("insert into thesaurus(terme,
date_creation,date_modifiction) values (?, ?, ?)");
 pstmtthesaurus.setString(1, terme1);
 pstmtthesaurus.setDate(2,  sqlDate);
 pstmtthesaurus.setDate(3, sqlDate);
```

```
/* exécution de l'insertion */
 pstmtthesaurus.executeUpdate();
 }
/* Création d'une instruction (Statement) JDBC paramétrée et pré-compilée pour
un ordre donné. Elle est associée à la table relation */
 pstmtrelation =connexion.prepareStatement("insert into
relation(terme1,terme2,puiss_relation ) values (?, ?, ?)");
 pstmtrelation.setString(1, terme1);
 pstmtrelation.setString(2, terme2);
 pstmtrelation.setDouble(3, sim);
/* exécution de l'insertion */
 pstmtrelation.executeUpdate();

 } catch (java.sql.SQLException e) {
       System.out.println("SQLException: " + e.getMessage());
       System.out.println("SQLState:     " + e.getSQLState());
       System.out.println("VendorError:  " + e.getErrorCode());
 }
 }
 lecteurAvecBuffer.close();
 pstmtrelation.close();
 pstmtthesaurus.close();
 } catch (Exception e) {
       e.printStackTrace();
 }
```

# Annexe B : Exportation du web thésaurus au standard SKOS

La commande *Données/Exporter Thésaurus* permet d'exporter le web thésaurus au format RDF/XML selon le standard SKOS. Dans ce qui suit, nous allons présenter tout d'abord le standard SKOS, puis nous décrivons le processus de conversion de notre web thésaurus au standard SKOS.


## B.1 Présentation du SKOS

SKOS ou Simple Knowledge Organisation System est une famille de langages formels permettant une représentation standard des thésaurus, classifications ou tout autre type de vocabulaire contrôlé et structuré. SKOS est construit sur la base du langage RDF, et son principal objectif est de permettre la publication facile de vocabulaire structuré pour leur utilisation dans le cadre du web sémantique.

Nous pouvons citer trois stations historiques dans le développement du SKOS :

- SWAD Europe (2002-2004) : SKOS a été un produit du projet SWAD Europe, un projet financé par la communauté européenne, dans le cadre du programme Technologie de la Société de l'Information.
- Activité Web Sémantique (2004-2005) : Suite au projet SWAD Europe, le travail sur SKOS a été pris en charge par l'activité web sémantique du W3C dans le cadre de travail sur les bonnes pratiques et le déploiement du standard RDF.
- Etat actuel (2006-2008) : SKOS est en cours de développement. Les principaux documents publiés sont :
  - Le guide SKOS Core.
  - La spécification du vocabulaire SKOS Core.
  - Guide pratique pour la publication d'un thésaurus par le web sémantique.

Ces documents ont le statut de « W3C Working Draft ».

Wa, Le nouveau groupe de travail du W3C, a pour mission l'avancement de SKOS au statut de recommandation W3C.

## B.2 Conversion du web thésaurus au format SKOS

Notre thésaurus peut être considéré comme un ensemble de concepts avec des libellés préférés. Les concepts peuvent être reliés par la propriété spécifique (narrower).

Le processus de conversion peut être décrit à travers les points suivants :

- Définition d'un schéma de concepts : la première étape consiste à décrire le schéma des concepts. Cette étape permet de référencer d'une manière non ambiguë le thésaurus. La propriété utilisée est *skos:ConceptScheme*. Dans notre cas, la description d'un tel schéma se fait par le biais des métas données, dans un fichier séparé (exemple : *webthesaurus.MD.DC*), en utilisant le Dublin Core.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <thesaurusdc xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:iaaaterms="http://purl.org/dc/IAAAterms/"
xmlns:iemsr="http://www.ukoln.ac.uk/projects/iemsr/terms/" xmlns:jml="jml"
xmlns:msxsl="urn:schemas-microsoft-com:xslt"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" editable="true" exclude-
result-prefixes="jml" fileID="webthesaurus" xml:lang="en">
  <dc:title>webthesaurus</dc:title>
  <dcterms:alternative>webthesaurus</dcterms:alternative>
  <dc:format>SKOS</dc:format>

<dc:identifier>http://iaaa.cps.unizar.es/thesaurus/webthesaurus</dc:identi
fier>
  <dc:language>en</dc:language>
  </thesaurusdc>
```

- Définition des concepts : On doit définir chaque concept d'une manière qu'il soit compréhensible par l'être humain et la machine. La façon la plus simple pour assurer la non ambiguïté des références est d'assigner à chaque concept une URI.

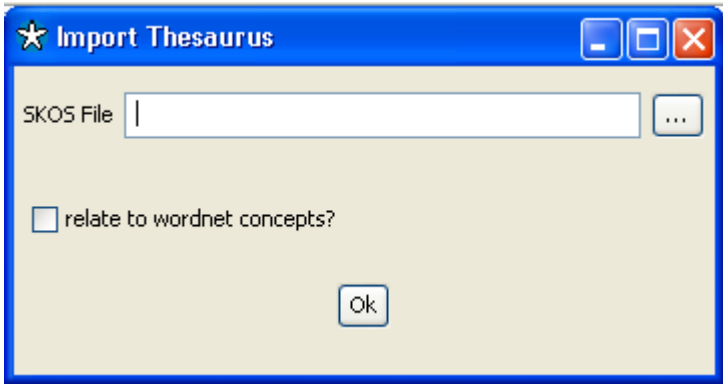
  Les concepts sont décrits par la propriété *rdf:description.* Pour exprimer qu'un concept fait partie d'un schéma de concepts, on utilise la propriété *rdf:inScheme*. Un concept doit être libellé par un et un seul libellé préféré. La propriété *skos:prefLabel* est utilisée.

- Relation entre les concepts : SKOS Core contient une famille de propriétés pour exprimer des relations entre les concepts dans le schéma conceptuel. Les propriétés disponibles sont *skos:related*, *skos:broader* et *skos:narrower*.

  Dans notre thésaurus, nous nous somme limité aux relations hiérarchiques *skos:narrower*. L'expression des relations hiérarchiques entre les différents concepts génère un graphe. Pour assurer la validité de l'exportation du thésaurus, il faut veiller à éliminer les cycles dans le graphe. Dans notre cas, cette contrainte est vérifiée lors de la génération du thésaurus.

## B.3 Importation du web thésaurus par Thmanager

La commande *File/Import Thesaurus* de Thmanager permet à l'utilisateur d'importer les fichiers thésaurus dans le format RDF/XML du standard SKOS. Par défaut, il cherche les fichiers avec une extension « skos.xml », mais l'utilisateur a le choix d'introduire le fichier qu'il désire.




Si le fichier metadata est fourni, il sera également importé. Dans l'autre cas, il sera crée automatiquement.

Le thésaurus doit être rajouté dans un entrepôt interne de Thmanager. Pour charger le thésaurus importé, on doit cliquer sur le bouton « Refresh »  .