Indexation et interrogation de chemins de lecture en contexte pour la recherche d'information structurée sur le web

Mathias Géry

UNIVERSITÉ JOSEPH FOURIER - GRENOBLE I
U.F.R. EN INFORMATIQUE ET MATHÉMATIQUES APPLIQUÉES

THÈSE

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ JOSEPH FOURIER - GRENOBLE I
Discipline : Informatique Systèmes et Communication

présentée et soutenue publiquement
par

Mathias GÉRY

le 24 octobre 2002

TITRE

Indexation et interrogation de chemins de lecture en contexte pour la Recherche d'Information Structurée sur le Web

Directeur de thèse : M. Yves CHIARAMELLA

Composition du jury :

Présidente :	Mme Catherine GARBAY
Rapporteurs :	Mme Florence SÈDES
	M. Alan F. SMEATON
Examinateurs :	Mme Cécile ROISIN
	M. Michel BEIGBEDER
	M. Jean-Pierre CHEVALLET

Thèse préparée au sein du laboratoire CLIPS-IMAG
(Communication Langagière et Interaction Personne-Système)
Université Joseph Fourier - Grenoble I

Remerciements

L'aboutissement de ma thèse, fruit de 4 années de dur labeur, est une grande satisfaction. C'est donc avec un immense plaisir que j'exprime ma profonde gratitude aux quelques milliers de personnes qui ont contribué, directement ou indirectement, à la réussite de ma thèse : je n'aurais jamais pu achever ce travail sans le soutien dont j'ai bénéficié.

Je remercie Mme Catherine GARBAY, directrice de recherche au CNRS, pour m'avoir fait l'honneur de présider le jury de cette thèse.

Je remercie Mme Florence SÈDES, Professeur à l'Université Paul Sabatier de Toulouse, pour avoir accepté de juger ce travail et pour les "chemins de lecture" qui ont contribué à l'amélioration de ce manuscrit, mais également pour sa gentillesse et notamment les encouragements prodigués à une certaine session d'INFORSID'99.

It is also a great pleasure for me to thank Mr Alan SMEATON, Professeur à Dublin City University, pour avoir accepté de juger ce travail, mais aussi pour tous le temps qu'il m'a consacré.

Je remercie Mme Cécile ROISIN, Professeur à l'Université Pierre Mendès-France de Grenoble, et M. Michel BEIGBEDER, Maître-Assistant à l'École des Mines de Saint-Étienne, pour leur lecture (minutieuse !) du manuscrit, et pour leurs commentaires qui ont grandement contribué à sa qualité finale.

Je remercie M. Yves "Big Boss" CHIARAMELLA, Professeur à l'Université Joseph Fourier de Grenoble, pour m'avoir initié à l'art subtil de la RI en DEA, pour avoir dirigé ce travail et l'avoir éclairé de ses connaissances phénoménales dans le domaine de la Recherche d'Information.

Je remercie M. Jean-Pierre CHEVALLET, Maître de Conférence à l'Université Pierre Mendès-France de Grenoble, qui a co-encadré ce travail, pour le temps qu'il m'a consacré durant toutes ces années, pour ses jeux de mots pleins d'humour drôle, et pour avoir partagé ce lourd fardeau et souffert avec moi ! Mais au fait, Jean-Pierre, « Qu'est-ce que tu entend par "information", exactement ? As-tu une définition précise du terme "concept" ? »

Je re-remercie Yves CHIARAMELLA pour m'avoir accueilli au laboratoire CLIPS en DEA, et Jean CAELEN pour ne pas avoir dégraissé l'effectif quand il a pris la direction du labo ! Je remercie également Marie-France BRUANDET, pour m'avoir accueilli au sein de l'équipe MRIM, pour sa gentillesse, et pour avoir materné "les deux zouzous au fond du couloir" ! Merci enfin à l'ensemble du personnel administratif du CLIPS, qui tiennent notre destinée entre leurs mains, et particulièrement à Bernard qui m'a maintes fois sauvé la vie et sans qui le labo ne pourrait plus tourner.

Un grand merci à Philippe MULHEM pour m'avoir co-encadré au début de mes travaux de recherche, pour sa disponibilité de tous les instants (une deadline le 31 décembre au soir ? Vous pouvez compter sur lui à mon avis !), pour sa bonne humeur, et pour ses blagues inégalées même par JPC : « Hello ! Elle bout... » Reviens nous vite Philippe !

Mener à bien un tel travail est une tâche très particulière, qui ne réclame pas uniquement un investissement scientifique de tous les instants associé à un travail acharné. Ce serait trop facile. Non, la thèse est un sacerdoce, un engagement total, qui nécessite le sacrifice de 4 des plus belles années de notre jeunesse... Euh, n'exagérons rien. La cloche ne sonne pas à 17h, et les jours où on réussit

à “décrocher” ne sont pas monnaie courante. Un soutien extérieur est indispensable, pour pouvoir continuer à avancer.

Je remercie tous les membres du CLIPS, croisés à la Kfét au détour d’un café, en particulier Brigitte, Solange, Jean-Claude, Jean-Philippe, Yannick, Jean-François, Laurent, Richard et les “séniors” de l’équipe MRIM : Catherine, Georges, Nathalie, Anne et le petit dernier Christophe, qui contribuent à faire du labo un lieu de travail convivial et agréable. Mention spéciale à Jean, Lizbeth, Anne, et M. C++, directement câblé à son Windows : nul n’est parfait Doms, on se le fait quand ce billard ?

Je remercie de tout cœur Chiraz pour sa gentillesse et ses Baklawas, et bien sûr mon “pot” (sic) Hatem, compagnon de tant de virées nocturnes, de plans foireux en conf’, d’Arc de Triomphe, d’hôtels pas chers, et d’errances éthyliques dans les bars et les rues un peu partout autour du monde. Ben tu vois, c’était pas si dur que ça Docteur !

Un grand coucou remerciesque aux Mautrans de Grenoble : Magali, Olivier la mignonne petite Julie nouvelle venue dans le groupe, Sébastien et Camille, Yann et Anne-Lise, Ninie et Fanf’, Nils, Bertrand, avec une mention particulière à Christèle pour être elle-même et pour tout ce qu’elle m’a apporté, et aussi pour avoir corrigé moultes fautes : elle qui est une fan de RI. Je n’oublierai pas euhmaiszob (qui fait de bien beaux tee-shirts, merci !) et les pseudo-mautrans : Sophie et les deux Fred qui s’incrustent.

Je remercie mes copains de Grenoble : Lionel, Thierry, Cécile, un poutou pour Xavier, Papy, Benoïte, J’enfile, Laurent, Eric. Merci à Ivan pour m’avoir appris que décidemment, le travail c’est la santé, mais pas trop quand même. Spéciale dédicace à Sandrine qui m’a apporté énormément.

Merci à tous mes copains de Valence, en souvenir de nos tendres années, et pour leur amitié indefectible. Par ordre (approximatif) d’apparition à l’écran : Gaël “7 trèfle surcontré” et Sophie, Nicolas “Gros Bill”, Julien “grosse loque”, le type là-haut il a dit que tu étais une fiotte, Alex “rouflaquettes”, Thierry “Jackson Five”, Séb, Paxs, Lætitia, Isa, Fred et Jérémy, et Julie quand je serais grande je veux être pompière. Merci à vous pour m’avoir permis de m’évader pendant mes (rares) vacances, j’espère qu’il y aura encore beaucoup de GR20, Corrençon, jeux, bouffes, discussions au coin du feu, vin chaud, raids & Co.

Je terminerai ces remerciements par ma famille, qui compte beaucoup pour moi : mes grands-parents, mon p’tit (sic) frangin Thibaud “tranquille la vie” et Émilie, mon tout p’tit frérot voyageur Mayeul, un jour moi aussi je partirai, et enfin mes parents (sans qui je ne serais pas là aujourd’hui, si si c’est prouvé), non pas pour le patrimoine génétique qu’ils m’ont légué ;-) mais pour m’avoir ouvert sur le monde, pour être là, toujours, quand on a besoin d’eux, et pour ... tout !

Enfin, un gros câlin à ma chérie-poupoune-adorée Blandine, qui a réussi à me supporter en fin de thèse, alors là vraiment chapeau ! « Dans tes yeux, y’a tant d’soleil, que quand tu me r’gardes, je bronze... » (c).

Résumé :

L'explosion du Web représente un nouveau défi pour la Recherche d'Information (RI). La plupart des systèmes actuels d'accès à l'information sont basés sur des modèles classiques, qui ont été développés pour des documents textuels, atomiques et indépendants et qui ne sont pas adaptés au Web. La structure du Web est un aspect essentiel de la description de l'information. Les travaux qui utilisent cette structure pour la RI simplifient le modèle du Web en un graphe orienté, dont les nœuds sont des pages HTML et les arcs sont des liens hypertextes, sans tenir compte du type des liens.
L'objectif de ce travail est de prendre en compte l'impact des liens lors de la phase d'indexation et à la phase d'interrogation d'un Système de Recherche d'Information Structurée (SRIS). Le modèle de RI proposé est fondé sur un modèle d'hyperdocuments en contexte considérant quatre facettes de la description d'information sur le Web : le contenu, la structure hiérarchique, la lecture linéaire/déambulatoire et le contexte. Un hyperdocument est modélisé par un contenu au sens des documents structurés, un ensemble de chemins de lecture et un contexte (espace d'information accessible et espace d'information référençant). Un processus d'indexation spécifique est proposé pour chaque facette.
L'évaluation de notre système SmartWeb montre l'intérêt de l'information accessible combinée avec le contenu. Puis, à l'aide de collections de test structurées construites automatiquement, nous montrons l'intérêt d'une indexation au niveau des documents structurés et des chemins de lecture. Le modèle est également implanté dans un SRIS complet, montrant ainsi la faisabilité de notre approche dans sa globalité et sur le Web. En particulier, le typage des liens est à la fois un des aspects les plus importants du modèle et une difficulté majeure de sa mise en œuvre : nous montrons qu'il est possible d'extraire une structure hiérarchique du Web et d'identifier différentes granularités d'information.

Mots-clés : Recherche d'Information, World Wide Web, hypertexte, structure, chemin de lecture, contexte, zone de pertinence

Abstract :

The growth of the Web gives new challenges in Information Retrieval (IR). Most of current systems are based on a re-use of traditional models, which have been developed for textual, atomic and independent documents and are not adapted to the Web. The Web structure is an essential aspect of the information description. Some approaches use this structure for IR, but most of them consider the whole set of links as a "bag-of-links", modelling the Web as a directed graph with HTML pages as nodes and hypertext links as edges, without taking into account the type of the links.
The aim of our work is to take into account the links at both indexing and query time of a Structured Information Retrieval System (SIRS). The proposed IR model is based on a model of hyperdocuments in context, considering four facets of information description on the Web : the content, the hierarchical structure, the linear or non-linear reading paths and the context. A hyperdocument is modelled by a content (like for the structured documents), a set of reading paths and a context (accessible information space and referencing information space). A specific indexation process is proposed for each facet.
The evaluation of our SmartWeb system shows the interest of the accessible information combined with the content. Then, we show the interest of an indexation of both "structured documents" and "reading paths", using several structured test collections automatically constructed. The model is also implemented in a full SIRS, showing the feasibility of our overall approach on the real Web. In particular, the links typing is one of the most important aspects of our model and is also the main difficulty of its implementation : we show that it is possible to extract a hierarchical structure from the Web and to identify different granularities of information.

Keywords : Information Retrieval, World Wide Web, hypertext, structure, reading paths, context, relevance area

Table des matières

Table des figures

Chapitre 1

Introduction

1.1 Les révolutions de l'information

La première révolution de l'information se fait avec l'invention de l'écriture (-4 000 avant J.C.), et donc celle du document. Estival en donne la définition : « *Toute connaissance mémorisée, stockée sur un support, fixée par l'écriture ou inscrite par un moyen mécanique, physique, chimique, électronique, constitue un document* » [Estival et al.81]. Nous considérons le lecteur comme étant un élément essentiel du processus de transmission (d'un auteur vers un lecteur, par le biais d'un document), comme l'exprime la définition de l'Organisation Internationale de Normalisation (OIN) :

Un document est un « *ensemble formé par un support et une information, généralement enregistré de façon permanente et tel qu'il puisse être lu par l'homme ou la machine* ».

Il est nécessaire d'organiser les collections de documents, afin de pouvoir retrouver une information pertinente : dès la création de la bibliothèque du Musée d'Alexandrie (-290 à -280 avant JC), les hommes ont tenté de mettre de l'ordre dans ces documents, en développant des techniques permettant de retrouver une information ou une référence à une information : catalogues, encyclopédies, annuaires, chronologies, bibliographies, index, etc. Ainsi, en -270 avant JC, le poète Callimaque dresse l'inventaire de la bibliothèque d'Alexandrie en 120 rouleaux [FS97], qui peuvent être considérés comme les prémices d'index. Le Moyen-Âge voit le triomphe de l'index pour organiser la quantité toujours croissante de documents. L'*Encyclopædia Universalis* donne la définition suivante de l'indexation de l'époque :

L'indexation « *L'indexation consiste à identifier dans un document certains éléments significatifs qui serviront de clé pour retrouver ce document au sein d'une collection. Ces éléments comprennent le nom de l'auteur, le titre de l'ouvrage, le nom de l'éditeur, la date de publication et l'intitulé du sujet traité* » [Encyclopaedia].

La deuxième révolution de l'information a lieu avec l'invention de l'imprimerie par Gutenberg (milieu du XV^{ème} siècle [Gutenberg54]), qui démocratise le livre et facilite la diffusion de la connaissance, accroissant encore le besoin d'organiser les documents. Les bibliothécaires sont amenés à affiner leurs méthodes (utilisation de bibliographies, de catalogues et

généralisation de l'index) et à en trouver de nouvelles pour organiser toujours plus de livres. La révolution de l'imprimé se situe plus au niveau du stockage et de sa diffusion qu'au niveau de son accès, et il faut attendre la troisième révolution de l'information, c'est-à-dire l'apparition de l'information numérique, pour voir une réelle avancée des techniques d'accès à l'information. Dans ce contexte, l'*Encyclopædia Universalis* propose une définition adaptée de l'indexation : « *il s'agit d'automatiser la classification et l'indexation de documents par la recherche de mots clés préétablis, ou en calculant les mots importants du texte en indexation libre ; on peut aller jusqu'au résumé automatique, qui peut soit extraire les phrases jugées les plus importantes (selon des métriques linguistiques ou statistiques) soit regénérer un texte résumé, à l'instar de l'humain* » [Encyclopaedia].

1.2 Modélisation d'un SRI

1.2.1 Trouver une aiguille dans une botte de foin

Avec la naissance de la Recherche d'Information (RI) et des Systèmes de Recherche d'Information (SRI), Salton [Salton71], [Salton et al.83b] et van Rijsbergen [vR79] développent des modèles de RI sur lesquels sont basés les moteurs de recherche actuels du Web, autour du triplet : < document, besoin, correspondance >.

1.2.2 Définition

On donne la définition suivante d'un SRI :

Définition 1 *Un **Système de Recherche d'Information (SRI)** est un système informatique qui facilite l'accès à un ensemble de **documents** (corpus), pour permettre de retrouver ceux dont le contenu **correspond** le mieux à un **besoin** d'information d'un utilisateur.*

Les SRI et les modèles sous-jacents se basent donc sur trois concepts essentiels : le document, le besoin et la correspondance. Les documents, atomiques et indépendants, doivent correspondre avec la représentation du besoin de l'utilisateur : la requête.

Pour cela, on distingue les deux tâches principales d'un SRI :

L'indexation automatique, c'est-à-dire l'extraction et le stockage du contenu sémantique des documents du corpus. Cette phase nécessite un modèle de représentation de ce contenu sémantique, appelé *modèle de documents*.

L'interrogation, c'est-à-dire l'expression du besoin d'information de l'utilisateur sous la forme d'une requête, la recherche dans le corpus, et la présentation des résultats. Cette phase nécessite un modèle de représentation du besoin de l'utilisateur, appelé *modèle de requête*, ainsi qu'une *fonction de correspondance* qui doit évaluer la pertinence des documents par rapport à la requête.

La réponse du système est un ensemble de références à des documents qui obtiennent une valeur de correspondance élevée. Cet ensemble est généralement présenté sous la forme d'une liste ordonnée suivant la valeur de correspondance. D'autres paramètres peuvent être considérés : le nombre de documents à présenter, la quantité d'information à fournir pour chaque document, le format de présentation utilisé, etc. Éventuellement, le système propose un mécanisme de retour de pertinence ("relevance feedback" [Rocchio71], [Salton et al.90]) : quand le résultat de la recherche n'est pas satisfaisant, le système reformule automatiquement la requête, en fonction du jugement de pertinence de l'utilisateur sur les documents déjà proposés. Il y a alors un apprentissage par étapes du besoin de l'utilisateur. Cette méthode permet à l'utilisateur de s'abstraire en partie des problèmes de formulation : syntaxe et complexité de la requête. De plus, certains concepts difficiles à exprimer le seront plus facilement "par l'exemple".

1.2.3 Composants principaux d'un modèle de RI

La problématique d'un SRI est de modéliser ce processus de recherche d'information. Pour cela, on distingue quatre composants principaux (cf. figure 1.1), qui utilisent le même formalisme de représentation des connaissances. Ce formalisme peut être très simple, comme par exemple des mots-clés, ou plus complexe, comme par exemple des graphes conceptuels.

Modèle de documents : correspond à la modélisation du contenu sémantique des documents, dans le formalisme de représentation de connaissances. Le choix du formalisme utilisé est crucial, mais il est toujours difficile, sinon impossible, d'obtenir une modélisation exprimant parfaitement l'idée initiale de l'auteur.

Modèle de requête : correspond à la modélisation du besoin d'information de l'utilisateur, dans le formalisme de représentation de connaissances. Ce formalisme limite souvent la précision de définition du besoin. De plus, la "qualité" de la requête exprimée par l'utilisateur varie considérablement avec sa connaissance du domaine et avec son aptitude à définir son besoin. Il y a donc souvent une importante perte d'information entre le besoin et son expression.

Fonction de correspondance : le système évalue la pertinence (la valeur de correspondance) des documents par rapport à la requête. La fonction de correspondance est un élément clé d'un SRI, car la qualité des résultats dépend de l'aptitude du système à calculer une pertinence des documents la plus proche possible du jugement de l'utilisateur.

Base de connaissances : un thésaurus, composé de concepts apparaissant dans le corpus, reliés entre eux par diverses relations (spécificité/généricité, synonymie, *voir-aussi*, etc.). En considérant par exemple les relations de synonymie entre les concepts, il est ainsi possible de retrouver, pour une requête composée du terme "voiture", des documents traitant de voiture ou d'automobile.

Ces quatre éléments permettent de modéliser un processus de recherche d'information : ils forment ce qu'on appelle un **modèle de RI**, tel que représenté dans la figure 1.1.

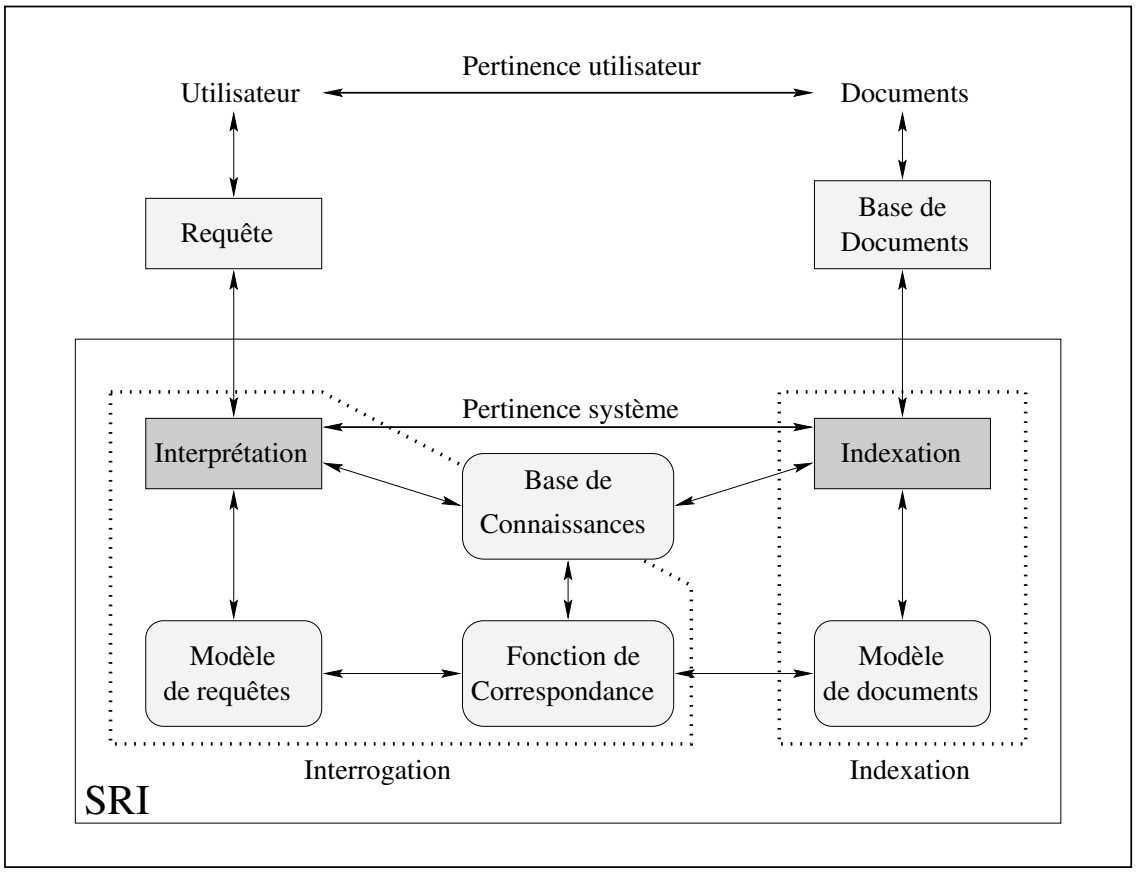

FIG. 1.1 – Schéma général d'un modèle de Recherche d'Information.

1.3 Exemples de modèles de RI

La majorité des modèles de RI se basent sur un formalisme simple : des mots-clés, c'est-à-dire un sous-ensemble des termes du corpus : T = $\{t_i\}$. Ce formalisme de représentation de connaissances est très réducteur et entraîne l'existence d'ambiguïtés, comme celles provenant de la synonymie et de la polysémie. De plus, il ne considère pas les fréquentes liaisons sémantiques entre les termes. Un terme sorti du contexte dans lequel il est employé peut en effet changer de sens : par exemple, dans l'expression "rage de dents", le terme "rage" pris indépendamment possède plusieurs synonymes. De même, le sens du terme "dents" perd en précision : s'agit-il d'une dent de scie ? L'utilisation de mots-clés est cependant très répandue, car ce formalisme est très simple et facile à mettre en œuvre. Il est utilisé par tous les modèles présentés ici, les différences se situant dans le modèle de documents et la fonction de correspondance.

On trouve dans la littérature plusieurs modèles "classiques" de RI, comme le modèle booléen, le modèle booléen étendu [Salton et al.83a], le modèle vectoriel [Salton et al.83b],

le modèle logique [vR86] [Nie90], et le modèle probabiliste [vR79]. A titre d'exemple, nous présentons brièvement le modèle booléen (et son extension pondérée) et le modèle vectoriel.

Le modèle booléen doit son nom à l'utilisation des opérateurs "OU", "ET" et "NON" pour la représentation des documents et des requêtes. Un document (ou une requête) est représenté par une conjonction de termes (propositions atomiques). La fonction de correspondance peut donc se résumer en une implication logique, les documents retrouvés étant ceux qui "impliquent" la requête. Un inconvénient du modèle booléen est son indexation binaire. Pour y remédier, des travaux étendent ce modèle en indexant les documents d'une manière plus souple : à la place d'une restriction à 0 ou à 1, un poids est associé à chaque terme d'indexation. Ce modèle permet un ordonnancement des documents par rapport à leur valeur de correspondance pour une requête, tout en conservant la possibilité d'exprimer une requête structurée à l'aide d'opérateurs booléens.

Le modèle vectoriel, très utilisé en Recherche d'Information, représente un document (ou une requête) par un vecteur dans un espace à n dimensions, n étant le nombre de termes du langage d'indexation. Une composante $w_{ij} \in [0, 1]$ d'un vecteur de document représente le *poids* du terme t_j dans le document d_i. La fonction de correspondance évalue la *similarité* de d_i par rapport à Q. Pour cela, l'utilisation du cosinus entre le vecteur du document et celui de la requête est courante.

1.4 Toujours plus d'aiguilles, toujours plus de foin

Les SRI ont été utilisés sur des collections de données purement textuelles aussi bien que sur des collections multimédia, comme des corpus de données médicales, bibliographiques, de documents techniques, etc. En ce début de XXI$^{\text{ème}}$ siècle, la civilisation de l'information est en train de supplanter la civilisation de l'automobile. Avec l'avènement du tout-numérique, la production d'information augmente considérablement chaque année : livres, documents multimédia, vidéos, photos, données audio, etc. En 1999, Lyman a estimé cette augmentation à 50% [Lyman et al.00], avec une production mondiale de "contenus" (documents papier, films, CD-ROMs, DVDs, etc.) de l'ordre du milliard de Go. La croissance concerne principalement l'information numérisée (de 50 à 70%), et beaucoup moins l'information diffusée sur papier (environ 2%).

1.4.1 La révolution des hypertextes

Parmi les technologies modernes de diffusion de l'information, la nouvelle révolution de l'information est celle de l'hypertexte et du Web, qui permettent l'interconnexion des données, des ordinateurs et, finalement, des personnes. S'inspirant du fonctionnement du cerveau humain par association d'idées, Vannevar Bush a introduit le concept d'hypertexte dans "*As we may think*" [Bush45], en décrivant le premier système hypertexte dénommé *Memex*. Il s'agissait à l'époque d'élaborer des liens associatifs entre des informations archivées sur d'importantes collections de microfilms pour permettre une consultation par navigation.

Memex est resté un système conceptuel, faute de moyens technologiques pour le mettre en œuvre. Le terme *hypertexte* lui-même a été utilisé pour la première fois par Ted Nelson en 1965 [Nelson65] dans le contexte du projet *Xanadu* [Nelson80] [Nelson93], et mis en œuvre par Douglas Engelbart (l'inventeur de la souris) dans les années 60 avec le système NLS (*oN-Line-System*, [Engelbart63]). NLS permet de consulter des documents textuels et de cliquer sur certains mots pour faire apparaître de nouveaux documents. Mais les premiers systèmes hypertextes ne sont commercialisés que dans les années 80, comme HyperCard en 1987, et la révolution de l'hypertexte ne prend toute son ampleur qu'avec l'invention du Web par Tim Berners-Lee en 1989 [BL89] [BL et al.92], et son explosion au milieu des années 90.

1.4.2 La révolution du World Wide Web

Avec l'émergence des réseaux, de l'Internet et du Web, le domaine de la RI se trouve face à de nouveaux défis pour l'accès à l'information. Internet est l'interconnexion mondiale de réseaux, avec tous les services proposés : e-mail, Web, forums de discussion, discussions en ligne, WAP, FTP, ou les plus anciens telnet, wais, gopher, etc. Internet est basé sur un ensemble de standards, qui décrivent chaque protocole utilisé. Le Web concerne uniquement les "documents", accessibles par le protocole HTTP [Fielding et al.99] à partir de n'importe quel terminal connecté à Internet.

Le nombre d'utilisateurs d'Internet dans le monde, demandeurs d'information, a été estimé à 119 millions en 1998, 333 millions en 2000, et à plus de 500 millions en 2001 [Nua01]. D'un autre côté, Le Web est un gigantesque espace d'information hétérogène, distribué à l'échelle planétaire, qui connaît une croissance exponentielle : on estimait la taille du Web à 320 millions de documents disponibles en 1998 [Lawrence et al.98], à 800 millions de documents en 1999 [Lawrence et al.99], et à plus de 2 milliards en 2000 [Murray et al.00]. De plus ces chiffres sont largement sous-estimés en raison de la taille considérable du "Web invisible[1]" (*deep Web*), qui est 500 fois plus importante que la taille du "Web de surface" [Bergman00]. Dans ce contexte, rechercher une information revient souvent à chercher une aiguille dans une botte de foin. L'espace d'information est gigantesque et les documents sont de plus en plus diversifiés : ils sont hétérogènes dans leur contenu, ils sont hétérogènes dans leur présentation (structure, mise en forme, etc.) malgré l'utilisation du standard HTML, ils sont écrits dans un très grand nombre de langues[2], et les utilisateurs et leurs besoins sont très variés.

Pour assister l'utilisateur dans sa recherche, les SRI actuels du Web (les moteurs de recherche, par exemple : Altavista, Excite, HotBot ou Lycos [Schwartz98], ou plus récemment Google [Brin et al.98]) permettent de retrouver des pages suivant différents critères, qui portent principalement sur le contenu textuel des documents. Ces systèmes traitent d'importants volumes de documents avec plusieurs centaines de millions de pages indexées. Ils

[1] Les sites Web dont la consultation nécessite une intervention humaine, par exemple une interrogation de Bases de Données.

[2] En septembre 2002, les moteurs AllTheWeb, Google et Altavista proposent de retrouver des pages Web écrites en respectivement 49, 35 et 25 langues.

sont néanmoins très rapides et sont capables de résoudre plusieurs milliers de requêtes par seconde. Malgré les moyens mis en œuvre, les réponses fournies par ces systèmes s'avèrent généralement peu satisfaisantes, car trop nombreuses, bruitées, et peu précises. Ces moteurs privilégient la puissance (nombre de documents indexés, nombre de requêtes par jour) souvent au détriment de la qualité des résultats.

Des résultats obtenus avec une collection de test de la "piste Web" de TREC ont montré la qualité inférieure des résultats de 5 moteurs du Web bien connus, par rapport à ceux de 6 systèmes participant à TREC [Hawking et al.99]. Plus récemment, 20 moteurs du Web ont donné des résultats s'approchant un peu plus de la qualité des systèmes de TREC, le meilleur d'entre eux dépassant à peine la précision[3] moyenne médiane des systèmes de TREC [Hawking et al.01b].

1.4.3 Limites des méthodes de RI actuelles

Les moteurs actuels du Web sont basés sur des modèles de RI qui ont été développés pour des documents textuels classiques depuis déjà plus de 30 ans [Salton71] [vR79] [Salton et al.83b]. Ces modèles ont été très étudiés dans le contexte de documents classiques : atomiques, "plats" et indépendants. De ce fait, la plupart des moteurs considèrent le Web comme un ensemble de documents atomiques et indépendants, dont la granularité est celle d'une page HTML. Ce choix a été fait pour des raisons pratiques : on fait alors l'hypothèse que l'auteur d'une page Web cherche à communiquer des informations de la granularité d'une page HTML, comme on le fait avec des documents classiques et des documents papier. Mais ce n'est pas toujours le cas, et cette hypothèse est souvent prise en défaut. De plus, beaucoup de moteurs ignorent purement et simplement les liens au cours de leur processus de RI. D'autres approches considèrent le Web comme un graphe orienté : les nœuds sont des pages HTML et les arcs sont des liens hypertextes, mais peu d'entre eux utilisent la structure du Web avec plus de finesse, comme nous le verrons dans la partie I. Les moteurs ne tiennent donc pas compte de la structure intra-page, et si la structure inter-page est parfois utilisée, elle n'est pas intégrée dans le modèle de documents. Les pages HTML étant indexées indépendamment les unes des autres, elles perdent leur contexte. Nous présentons dans la partie I les principaux travaux qui, bien que basés sur des modèles classiques, tentent d'intégrer la structure des documents et de l'hypertexte dans le processus de RI.

Les moteurs de recherche actuels ne sont pas adaptés aux caractéristiques des documents du Web. Un axe de recherche prometteur consiste donc à étudier l'impact de la structure du Web sur l'indexation et l'interrogation. Nous pensons que pour profiter de la richesse de cette structure il est nécessaire de l'intégrer directement au sein du modèle de documents, plutôt que de répercuter la structure sur un modèle classique de documents ou de rajouter une opération à la correspondance.

L'hypertexte apporte une nouvelle dimension à la diffusion de l'information, en particulier sur le Web : non seulement dans la présentation de l'information ou dans la structure

[3] Précision : proportion du nombre de documents pertinents dans les documents retrouvés (cf. section 9.2.2).

logique des documents, mais aussi dans la structure même de l'information, à un niveau sémantique. Par exemple, la lecture d'un document structuré est linéaire, alors qu'un hypertexte permet une lecture non-linéaire. Cette structure particulière de l'information doit donc être prise en compte par le modèle de Recherche d'Information.

1.5 Problématique de la thèse

1.5.1 Le Web : dualité documents structurés/hypertextes

Les documents du Web ont des caractéristiques de documents structurés. Divers langages sont utilisés pour la description des documents de cette immense collection, parmi lesquels HTML (Hypertext Markup Language) [Raggett et al.99], langage dérivé de SGML, occupe une place prépondérante. SGML (Standard Generalized Markup Language) est un langage structuré qui a été normalisé par ISO en 1986 (ISO 8879-1986) et qui permet de décrire la structure logique d'un document. HTML hérite de certaines possibilités de SGML et permet de décrire la structure logique et la présentation d'un document. La figure 2.2 du chapitre 2 montre une exemple simple de structure hiérarchique d'une page HTML.

D'un autre côté, le Web est aussi un hypertexte distribué à l'échelle planétaire : des liens hypertextes entre les pages sont définis grâce à la norme URL [BL et al.94] [Raggett et al.97] (ou plus généralement grâce à la norme URI [BL et al.98]). Ces liens décrivent une structure hypertexte des sites Web et une structure macroscopique entre les sites. La figure 2.4 du chapitre 2 montre un exemple simple de graphe hypertexte sur le Web avec les deux types de structure hypertexte.

La dualité documents structurés/hypertextes implique non seulement l'existence d'une structure du Web, mais l'existence de plusieurs structures : structure hiérarchique, structure hypertexte et structure macroscopique. Chacune des structures du Web est une composante essentielle de la description de l'information.

1.5.2 La structure du Web

Selon que l'on utilise HTML ou URL pour la décrire, nous distinguons donc plusieurs niveaux de structure : les pages Web possèdent une structure interne (grâce au langage HTML) et sont connectées par un réseau de liens hypertextes (grâce à la norme URL). Ce réseau de liens décrit une structure externe, composée de la structure des sites Web (interne à un site) et de la structure macroscopique du Web (externe aux sites). Nous faisons donc la distinction entre la structure de type "document structuré" (structure arborescente, sens de lecture linéaire) et la structure de type "hypertexte" (structure de graphe, lecture non-linéaire). De nombreux travaux ont porté sur l'extraction de structure sur le Web, nous en reparlerons dans les sections 2.3 (documents structurés) et 2.5 (hypertextes).

1.5.3 Intégration de la structure du Web dans le modèle de RI

Notre problématique consiste à intégrer la structure du Web (ou *les* structures du Web) au sein d'un modèle de RI. Nous devons donc répondre aux questions suivantes : quelle structure existe-t-il sur le Web, comment l'identifier et l'extraire, comment la modéliser au sein d'un modèle de documents, et comment l'utiliser à la phase d'interrogation ? En effet, un index est « *une représentation synthétique de l'information relative à un document, qui met en évidence sa sémantique en vue d'une requête* » [Paradis96]. L'objectif d'un modèle de RI Structurée pour le Web est de prendre en compte sa structure, ce qui nécessite de s'interroger sur la sémantique de la structure, et donc des relations, pour pouvoir comprendre son impact sur la description de l'information. En d'autres termes, il faut se demander comment l'auteur d'un site Web utilise les relations pour décrire le message qu'il veut faire passer. Par exemple, est-ce que le fait de référencer une page Web indique une appréciation de la part de l'auteur, une similarité entre les documents, un conseil de lecture, ou bien une composition des contenus ?

Nous considérons les sites Web à la fois du point de vue des documents structurés et du point de vue des hypertextes. Un document possède rarement les caractéristiques d'un document structuré "pur" ou d'un hypertexte "pur" (*a fortiori* sur le Web), mais plutôt une combinaison des deux. Un document structuré "pur" possède une structure hiérarchique (arborescente) basée sur la relation de composition. Un hypertexte "pur" possède une structure de graphe, basée sur les relations de cheminement et de référence entre les nœuds. Les relations de cheminement sont des références internes au site : l'auteur propose au lecteur de poursuivre sa lecture dans un autre nœud du graphe. Les relations de référence sont externes au site : l'auteur propose au lecteur d'aller consulter d'autres sites.

Ces trois types de relations jouent un rôle majeur dans la construction de l'information, en raison de leur impact sur la lecture des "documents". Un modèle de RI adapté au Web doit prendre en compte ces trois types de relations et la structure associée, et les répercuter sur le modèle de documents. Pour cela, nous proposons un "modèle de Recherche d'Information Structurée" basé sur la modélisation d'hyperdocuments en contexte, et plus précisément de **chemins de lecture en contexte**. Un chemin de lecture est *un enchaînement possible de tout ou partie des composants d'un document, qu'un auteur propose comme solution de lecture du document*. Un chemin de lecture donné permet au lecteur de se fabriquer une interprétation parmi d'autres de l'information présentée dans le document.

1.6 Vers un modèle de RI structuré adapté au Web

La présentation de notre travail est divisée en 3 parties. Nous commençons par présenter l'état de l'art de la problématique qui nous intéresse dans la partie I : l'**Utilisation de la structure en Recherche d'Information**. Puis, nous détaillons notre proposition de modèle de RI adapté au Web dans la partie II : **Un modèle de Recherche d'Information Structuré en Contexte**. Enfin, nous présentons les expérimentations que nous avons menées pour valider certains aspects de ce modèle dans la partie III : **Mise en œuvre : un Système de RI**

Structuré sur le Web. Ces trois parties sont organisées comme suit :

Chapitre 1 : Introduction.

Chapitre 2 : Structure du Web : dans le deuxième chapitre, nous étudions la structure et la dualité documents structurés/hypertexte du Web, puis nous présentons des travaux visant à identifier et à extraire cette structure.

Chapitre 3 : Intégrer la structure à l'indexation : dans le troisième chapitre, nous présentons un état de l'art sur l'utilisation de la structure à la phase d'indexation d'un SRI, par le biais d'un modèle de documents structurés ou d'un processus d'indexation adapté.

Chapitre 4 : Intégrer la structure à l'interrogation : le quatrième chapitre complète l'état de l'art, en récapitulant les approches qui utilisent la structure à la phase d'interrogation et qui proposent un modèle de requête structuré ou une fonction de correspondance adaptée.

Chapitre 5 : Structure du Web et RI : nous discutons dans le cinquième chapitre des avantages et des inconvénients des approches de l'état de l'art, et nous en tirons les orientations de notre modèle de RI.

Chapitre 6 : L'information structurée sur le Web : le sixième chapitre est une présentation informelle des principes sur lesquels se fonde notre approche, basée sur les éléments qui nous semblent essentiels pour la description et la compréhension de l'information : le contenu, la composition, la lecture linéaire ou non-linéaire avec les chemins de lecture, et le contexte. Ce chapitre introduit les éléments de notre modèle de Recherche d'Information Structurée.

Chapitre 7 : Modèle d'hyperdocuments en contexte : nous présentons dans le septième chapitre un modèle d'*hyperdocuments* pour le Web, qui résume et formalise les principes développés dans le chapitre 6.

Chapitre 8 : Indexation et interrogation structurée : le huitième chapitre présente le processus d'indexation et la fonction de correspondance de notre modèle de Recherche d'Information Structurée, qui exploitent l'aspect structuré du modèle d'hyperdocuments.

Chapitre 9 : Expérimentations et évaluation : nous présentons dans le neuvième chapitre la mise en œuvre du modèle d'hyperdocuments, avec les expérimentations visant à valider notre approche.

Chapitre 10 : Un SRI Structurée sur le Web : le dixième chapitre décrit un prototype de SRIS pour le Web, et explore la problématique de l'extraction automatique de structure.

Chapitre 11 : Conclusion et perspectives.

Première partie

Utilisation de la structure en Recherche d'Information

Chapitre 2

Structure du Web

La problématique de l'utilisation de la structure pour la RI a été étudiée dans de très nombreux travaux, que ce soit dans un contexte de documents structurés, d'hypertextes, ou plus généralement dans le contexte du Web. La problématique sous-jacente qui fait l'objet de ce chapitre est l'étude de cette structure, particulièrement dans le contexte hétérogène du Web, qui nous permettra d'aborder dans les chapitres 3 et 4 la problématique de l'intégration de la structure au processus de Recherche d'Information.

2.1 Le World Wide Web

Le Web peut être considéré comme un ensemble de documents structurés mais aussi comme un gigantesque hypertexte. Cette dualité implique l'existence de plusieurs structures logiques du Web, selon l'utilisation de HTML et/ou de URL pour la décrire. Nous distinguons donc les niveaux suivants :

1) La structure interne aux pages, qui est décrite grâce au langage HTML.

2) La structure externe aux pages, qui est décrite par le réseau de liens hypertextes (norme URL). La structure externe se décompose de la manière suivante :

- **2.1) La structure hiérarchique** des sites Web, c'est-à-dire la structure arborescente interne à un site.
- **2.2) La structure hypertexte** des sites Web, c'est-à-dire la structure de graphe interne à un site.
- **2.3) La structure macroscopique** du Web, c'est-à-dire la structure de graphe externe aux sites.

Parmi ces niveaux, nous faisons la distinction entre la structure de type "document structuré" (structure arborescente, sens de lecture linéaire) et la structure de type "hypertexte" (structure de graphe, lecture non-linéaire).

Nous présentons dans ce chapitre les principes des documents structurés dans la section 2.2 et des hypertextes dans la section 2.4, afin de déterminer dans les sections 2.3 et 2.5 comment considérer le Web selon ces deux points de vue. Nous présentons également différentes

approches d'extraction de structure implicite du "sac de nœuds" et du "sac de liens" du Web, afin de comprendre quelle est la structure cachée du Web, et quelle est la part d'héritage provenant des documents structurés et des hypertextes. Enfin, nous terminons ce chapitre par la présentation d'un exemple concret de site Web structuré, afin de mettre en évidence notre propre vision de la structure du Web.

2.2 Documents structurés

Un document se présente rarement sous la forme d'un "bloc" textuel car il est utile de le structurer pour établir une décomposition des thèmes abordés et une hiérarchie entre les différentes sous-parties. L'auteur structure le document pour faciliter la compréhension de l'information qu'il cherche à communiquer.

Nous commençons par présenter les principales caractéristiques des documents structurés, avec en particulier la description de la structure hiérarchique des documents du Web. Puis nous présentons dans la section suivante plusieurs travaux proposant d'identifier ou d'extraire une structure hiérarchique du Web.

2.2.1 Définitions

Classiquement, un document structuré est composé d'un *ensemble de parties* (le contenu), organisées de manière *hiérarchique* (la structure logique). De plus, on peut définir des *attributs externes* associés à chacune des parties. Un auteur rédige un document dans le but de communiquer une information aux lecteurs, avec une cohésion entre les différentes parties.

Nous allons maintenant présenter la notion de document structuré, à travers les définitions de ses principaux composants : contenu, structure, attributs externes et sens de lecture.

a) Contenu

Le contenu d'un document structuré désigne le contenu textuel ou multimédia, représenté sous la forme d'un ensemble de fragments insécables et non structurés, comme par exemple des paragraphes, des figures ou des images. Le contenu est alors l'atome de description des documents.

b) Structure

La création et l'échange de documents sur des plateformes hétérogènes a conduit à la définition de normes de représentation de documents structurés, telles ODA (Office Document Architecture), SGML ou HyTime. Ces normes distinguent deux types de structures, la structure physique et la structure logique, qui sont définies de la manière suivante :

Structure physique : correspond à l'organisation d'affichage des données qui composent le document c'est-à-dire la présentation. Généralement, un document est composé d'un ensemble de pages, elles-mêmes composées d'un en-tête, de lignes, de notes de bas de

pages, de figures, etc. (cf. figure 2.1). La structure physique dépend de l'environnement de présentation du document, comme le format du papier ou l'écran d'un ordinateur. Par exemple, un document électronique n'aura pas la même structure physique, selon les fonctionnalités du système utilisé pour sa présentation, comme la résolution, le mode d'utilisation (portrait ou paysage) ou la surface d'affichage disponible.

Structure logique : correspond à l'organisation hiérarchique des données du document. La structure logique est spécifiée par l'auteur, elle lui permet de décomposer et d'organiser le document pour mieux exprimer ses idées, généralement à l'aide d'abstractions représentant des parties du document. Par exemple, un document se compose d'un titre, d'une ou plusieurs sections, elles-mêmes composées d'un titre, d'une ou plusieurs sous-sections, etc. (cf. figure 2.1).

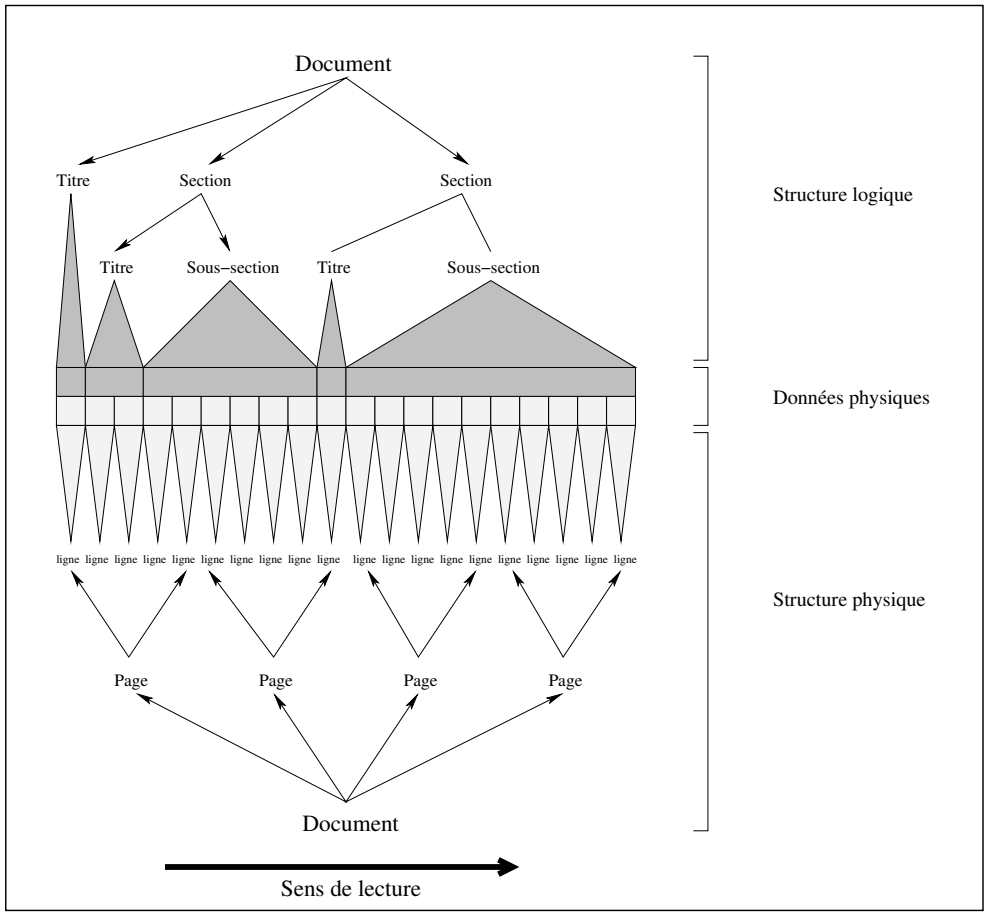

FIG. 2.1 – Structure logique et structure physique d'un document.

La structure logique propose un "**sens de lecture**" implicite, qui correspond à l'enchaînement des idées de l'auteur. La décomposition d'un document structuré reste la même quel

que soit l'environnement de présentation : la structure logique est donc indépendante de celui-ci.

Par contre, la structure physique est dépendante de cet environnement, qui n'est généralement pas connu du SRI. On ne peut donc pas en tirer d'information relative au document : par exemple, les pages d'un document au format A4 ne présenteront pas les mêmes informations que celles du même document au format A3. La structure physique n'est d'aucune utilité pour la RI, sauf si l'utilisateur précise le mode de présentation dans la requête.

La structure logique est définie à la création du document. Elle est porteuse d'une partie du message que l'auteur cherche à exprimer. Par exemple, nous avons choisi de séparer en deux sections intitulées "Documents structurés" et "Hypertextes" la question des documents structurés et celle des systèmes hypertextes, car il s'agit de deux domaines distincts.

La structure logique est donc susceptible d'être utilisée au cours d'un processus de recherche d'information, comme information supplémentaire pour améliorer les résultats de la recherche, mais aussi comme critère de recherche mis à la disposition de l'utilisateur.

c) Attributs externes

Les attributs externes sont des éléments de description de l'information attachés aux éléments d'un document structuré. Il s'agit de méta-information, c'est-à-dire d'*information à propos de l'information*, comme par exemple le titre, l'auteur ou la date de création d'un document. Ils apportent une information supplémentaire qui ne concerne pas l'information décrite dans le document, mais le document lui-même. Comme la structure logique, les attributs externes représentent une information supplémentaire qui peut être utilisée lors d'une recherche d'information.

d) Sens de lecture

Comme avec un livre, la lecture d'un document structuré consiste à consulter l'introduction et à enchaîner la lecture des chapitres successifs, jusqu'à la conclusion. Dans le cas particulier d'un document structuré "pur", il n'est pas possible de suivre un autre sens de lecture sans nuire à la compréhension de l'ensemble.

Définition 2 *La lecture d'un document structuré consiste à suivre un chemin de lecture linéaire et unique qui parcourt la totalité du document. Il est imposé par l'auteur, et le lecteur consulte les informations de manière séquentielle.*

2.3 Web et documents structurés

Le Web a les caractéristiques d'un document structuré. Nous les présentons dans cette section, en particulier la structure hiérarchique.

2.3.1 Structure hiérarchique interne des pages Web

Les pages HTML possèdent une structure interne que nous appelons *structure hiérarchique intra-page*, grâce à l'utilisation de balises HTML qui permettent de définir des éléments de différentes granularités. Par exemple, la balise <*P*> définit un paragraphe, les balises <*H1*>, <*H2*>, <*H3*>, <*H4*>, <*H5*>, et <*H6*> définissent des sections.

La figure 2.2 montre un exemple simple de structure hiérarchique d'une page HTML :

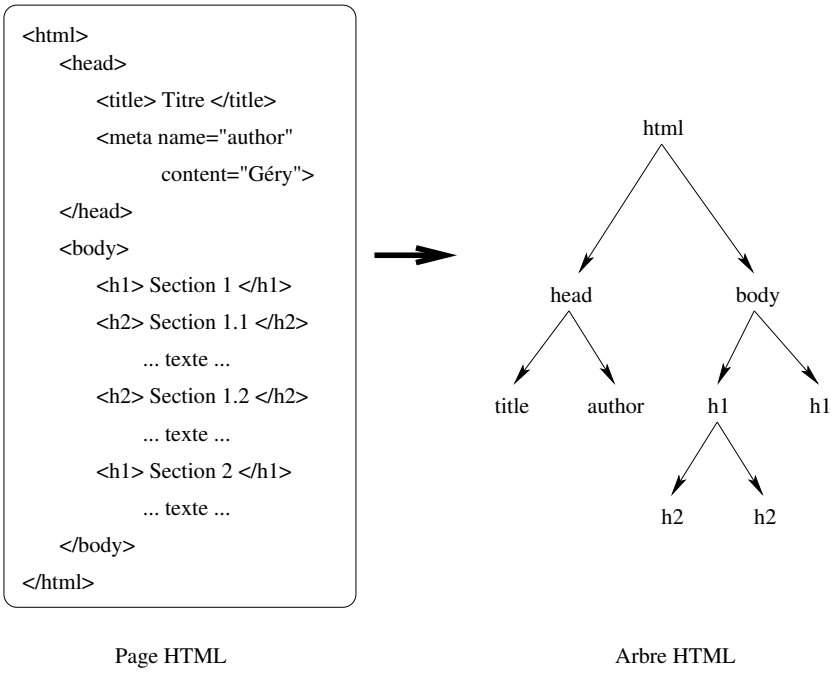

FIG. 2.2 – Structure hiérarchique d'une page HTML.

2.3.2 Structure hiérarchique interne des sites Web

Les liens hypertextes peuvent être utilisés pour décrire la structure interne d'un document, que nous appelons *structure hiérarchique intra-site*, auquel cas ses différentes parties sont fragmentées en plusieurs pages HTML. Ce type de structure se rencontre fréquemment sur le Web, en particulier en raison de l'utilisation de logiciels comme LaTeX2HTML[1] qui permet de transformer des documents structurés en un ensemble de pages HTML reliées. De nombreux documents, même décrits en HTML et utilisant des liens hypertextes, sont quand même toujours construits à la manière des documents structurés. Les liens ne sont alors utilisés que pour faciliter la lecture et la maintenance.

La figure 2.3 montre un exemple typique de document structuré fragmenté en plusieurs pages HTML : le livre de Keith van Rijsbergen "*Information Retrieval*" [vR79].

[1] LaTeX2HTML : http ://www.latex2html.org

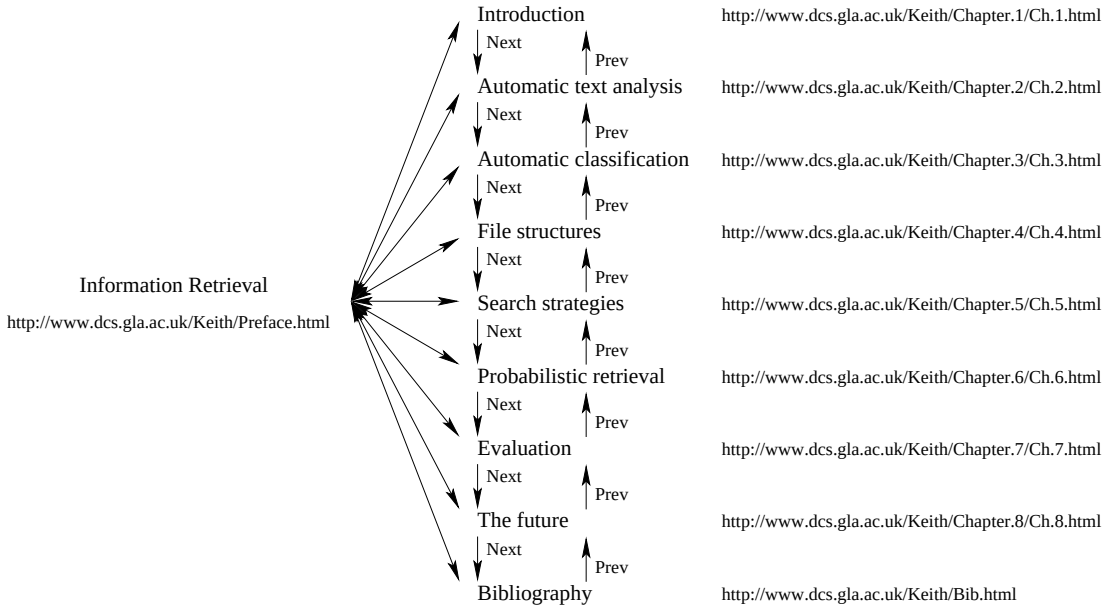

FIG. 2.3 – Structure hiérarchique d'un site Web.

2.3.3 Le futur du Web : description de structure hiérarchique

L'abandon progressif de HTML est programmé au profit de XML (eXtended Markup Language) [Bray et al.00] et de ses dérivés, en s'aidant du langage XHTML (eXtensible Hypertext Markup Language, [Pemberton et al.00]) qui est une reformulation de HTML en XML. Avec ces nouveaux standards, une tendance à décrire de plus en plus de structure dans les documents se confirme, avec l'aboutissement du DOM (*Document Object Model*) dont la première version a été finalisée en 1998 (*Document Object Model level 1, version 1.0*, [Wood et al.98]), adaptée spécifiquement aux documents HTML 4.0 et XHTML 1.0 (*DOM Level 2 HTML Specification*, [Stenback et al.01]). La dernière version du DOM est le *DOM Level 3*, basé sur le *DOM Level 3 Core Specification* [Hors et al.02] avec notamment la partie XPath [Whitmer02] qui offre des fonctionnalités pour accéder à un arbre DOM.

2.4 Hypertextes

Le principe des hypertextes a été inventé par Vannevar Bush avec le système *Memex*, en s'inspirant du fonctionnement du cerveau humain par association d'idées [Bush45]. L'idée principale d'un système hypertexte est donc de donner la possibilité à l'utilisateur de gérer (consulter et modifier) un document ou un ensemble de documents de manière non linéaire (par opposition au livre qui se lit de manière linéaire), en organisant les informations de manière associative. L'intérêt est de pouvoir naviguer dans un espace d'information en choisissant de suivre les associations que l'utilisateur juge pertinentes au moment de sa lecture.

2.4.1 Définitions

Nous proposons la définition suivante d'un hypertexte :

Définition 3 *un **hypertexte** est une représentation non-linéaire d'une information textuelle sous la forme d'un graphe de **nœuds** connectés par des **liens**. La consultation d'un hypertexte nécessite une phase interactive de **navigation**.*

Nous définissons les notions qui sont à la base de l'hypertexte dans les sections suivantes : le nœud, le lien, l'ancre, la navigation, le cheminement, le tour guidé et l'hypermédia.

Définition : un **nœud** *est une unité d'information textuelle, c'est-à-dire un fragment de texte (chapitre, section, paragraphe, etc.), ou un document entier.*

Un nœud peut contenir un paragraphe, une *carte* (HyperCard), une page, ou même une image, un son ou une vidéo dans le cas d'hypermédia.

Définition : un lien *définit une connexion entre deux nœuds de l'hypertexte, le nœud source et le nœud destination du lien.*

On distingue deux classes de liens :

Liens explicites : ceux qui ont été définis par l'auteur au moment de la création de l'hypertexte, comme modélisant une relation entre deux nœuds.

Liens implicites : ceux qui ne sont pas définis au moment de la création de l'hypertexte, mais qui existent potentiellement. Ils peuvent être extraits de l'hypertexte, et utilisés sur demande de l'utilisateur (exemples : liens de similarité, de co-citation, de co-occurrence de termes, etc.).

Les liens explicites sont décrits à l'aide de URL/URI, qui définissent un espace d'adressage et permettent d'associer un identifiant unique à chaque ressource Web. L'ensemble des nœuds associés aux liens qui les connectent construisent un graphe.

Définition : une **ancre** *matérialise la source ou la destination d'un lien dans un nœud.*

Sur le Web, une ancre source est une zone "cliquable" de la page HTML : une portion de texte ou une image. Une ancre destination peut être définie en une zone quelconque à l'intérieur d'un document, mais peut aussi correspondre à la page Web dans sa globalité.

Définition : la **navigation** *consiste à "activer" un lien hypertexte pour se retrouver "transporté" sur l'objet référencé, à l'aide d'un logiciel de consultation du Web appelé "navigateur" (ou "butineur").*

La navigation est aux hypertextes ce que la lecture est aux documents structurés. On distingue plusieurs types de navigation dans un hypertexte, comme le tour guidé ou le cheminement déambulatoire.

Définition : un **tour guidé** ("*guided tour*") *est une navigation supervisée qui suit un parcours linéaire imposé par l'auteur.*

Définition : le **cheminement déambulatoire** (“*browsing*”, “*butinage*”) *est une navigation non supervisée : le lecteur choisit lui-même les liens qu'il désire suivre, au fur et à mesure de sa “balade”.*

Un hypermédia (comme le Web) est une généralisation du concept d'hypertexte :

Définition : un **hypermédia** *est un hypertexte dont les nœuds d'information sont composés de n'importe quel type de média : texte, image, son, vidéo, etc.*

2.5 Web et hypertextes

Le Web a aussi des caractéristiques d'hypertextes. Nous les présentons dans cette section, en particulier avec la structure hypertexte interne aux sites et la structure macroscopique du Web.

2.5.1 Sites Web

Un site Web est un hypertexte : il possède des nœuds (les pages HTML) connectés par des liens (définis à l'aide d'URLs). Chaque site est un hypertexte distinct, qui peut être consulté indépendamment des autres, et qui représente une information diffusée par une personne ou un groupe de personnes. Généralement, on réduit la notion de site Web à la machine physique qui l'accueille, mais ce n'est pas toujours le cas. En effet, plusieurs sites Web peuvent être hébergés sur la même machine, comme par exemple un ensemble de sites Web personnels hébergés par un fournisseur d'accès à Internet. A l'opposé, un site Web peut être distribué sur plusieurs machines.

Nous considérons donc le Web comme un ensemble d'hypertextes. Ces hypertextes sont eux-mêmes connectés par des liens et peuvent donc être considérés comme les nœuds d'un *hyper-hypertexte*. Le World Wide Web est donc un hyper-hypertexte.

2.5.2 Structure hypertexte des sites Web

Les liens hypertextes peuvent être utilisés pour décrire la structure hypertexte interne d'un site Web, appelée *structure hypertexte intra-site*. Cette structure organise les documents (pages HTML) au sein d'un même site Web, permettant une consultation hypertexte des sites. En effet, les visiteurs ont la possibilité de parcourir le Web à la manière des hypertextes, en choisissant au fur et à mesure les liens à activer (et donc leur chemin de lecture), contrairement aux documents structurés qui comportent un chemin de lecture imposé.

La figure 2.4 montre un exemple simple de graphe hypertexte sur le Web avec les deux types de structure hypertexte : la structure hypertexte intra-site et la structure macroscopique du Web.

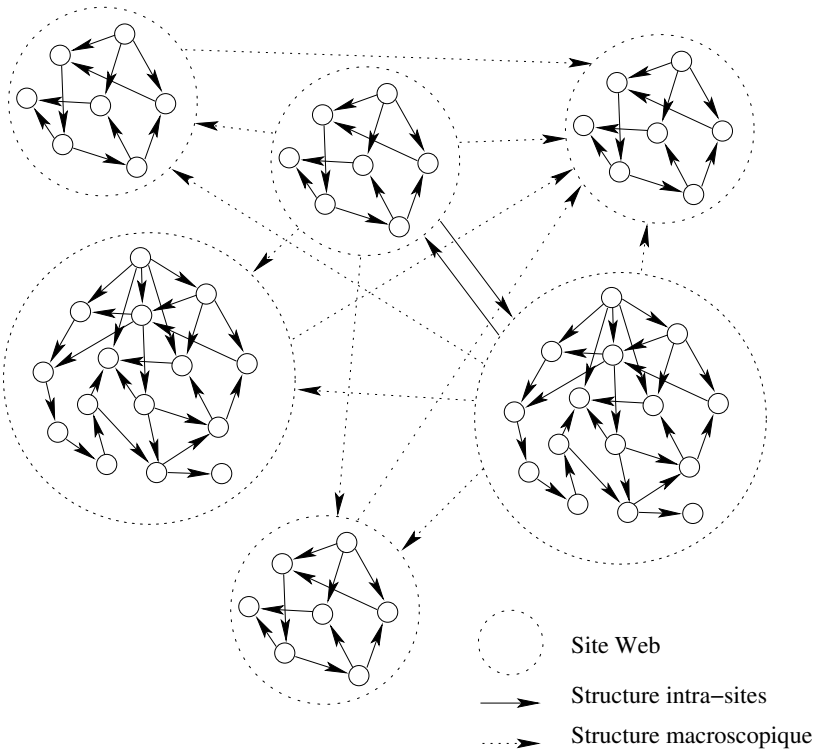

FIG. 2.4 – Structure hypertexte des sites Web.

2.5.3 Structure macroscopique du Web

La structure macroscopique est celle qui organise les sites Web entre eux. En effet, les lecteurs peuvent aussi naviguer de site Web en site Web en suivant des liens de référence, qui à première vue ne semblent pas décrire de structure particulière. La structure macroscopique désigne l'émergence non contrôlée de pages qui jouent un rôle particulier dans l'hypertexte Web, à l'instar des pages considérées comme des pages de référence dans un domaine et qui sont par exemple au centre d'un ensemble de sites Web d'une communauté.

Mais d'autres méthodes d'extraction de structure au niveau macroscopique du Web s'intéressent à des groupes de pages plutôt qu'à des pages prises individuellement, comme par exemple des *grappes de sites* [Bray96] [Carriere et al.97]. Ces grappes ont parfois une structure typique, comme par exemple les *anneaux du Web* ("Web rings"). Ces anneaux sont le fruit du travail d'une communauté d'intérêts, qui a organisé une chaîne de liens permettant de consulter ses sites (initialement indépendants) de manière séquentielle.

2.5.4 Le futur du Web : description de structure hypertexte

De même que pour la structure des documents, l'avènement annoncé de XML et de ses dérivés [Bray et al.00], comme le langage de description de liens XLink [Derose et al.01b],

tend à la description de plus en plus fine de la structure du Web, avec un typage des liens et des mécanismes d'adressage plus complets. En effet, les liens décrits à l'aide des standards HTML et URL/URI se sont révélés trop simples pour exprimer toute la sémantique que l'on voudrait représenter par des liens. XLink permet de décrire des liens bi-directionnels typés, pouvant connecter plus de deux ressources à la fois. Les liens peuvent aussi être définis dans un document à part, séparant le contenu de la navigation. XLink utilise XPointer [Derose et al.01a], qui permet de référencer avec une granularité plus fine des parties de documents. XPointer est un mécanisme d'adressage de structure interne d'un document XML, qui utilise lui-même le langage XPath [Clark et al.99] pour décrire des accès internes à un arbre de document XML.

2.6 Extraction de structure du Web

Les nouveaux standards présentés dans les sections 2.3.3 et 2.5.4 permettent de décrire la structure du Web, avec d'une part la description de la structure hiérarchique des pages Web et d'autre part la description élaborée des liens et donc de la structure hypertexte du Web. Cependant, même si l'utilisation de ces standards est encore relativement peu répandue pour la description de sites Web (qui demeurent principalement basés sur la norme HTML), cela ne signifie pas pour autant que de telles structures n'existent pas sur le Web dans son état actuel. En effet, même si la pauvreté de description de HTML ne permet pas toujours de décrire cette structure dans les pages et au niveau des liens entre les pages, elle y est souvent implicite. En effet, les balises HTML ou les URLs sont souvent riches d'informations à propos de la structure. Il est donc nécessaire de réaliser un traitement supplémentaire pour l'extraire ou l'identifier.

2.6.1 Extraction de la structure hiérarchique

Nous faisons la distinction entre structure hiérarchique des pages et structure hiérarchique des sites en raison de l'existence de la page HTML comme "document" de base du Web, sans que cela corresponde à une réalité du point de vue de la structure. En effet, nous avons vu que la structure *logique* peut être décrite au sein d'une page HTML et entre des pages HTML. Nous présentons dans cette section des approches permettant d'extraire ou d'identifier cette structure logique.

a) Structure hiérarchique intra-page

Nous distinguons trois types d'approches pour extraire la structure hiérarchique intra-page : l'utilisation de la structure logique décrite à l'aide de HTML (ou tout autre type de langage de description structuré, comme SGML), l'utilisation de patrons pour l'intégration de données semi-structurées, et l'utilisation de la similarité des parties d'un document comparées deux à deux.

Le travail de Woodruff [Woodruff et al.96] nous montre que les auteurs de pages HTML font un usage massif des balises HTML de description de structure. Avec l'analyse d'une collection de 40 000 pages Web du site des laboratoire de l'IMAG[2], nous avons montré dans [Gery et al.01] que la structuration interne des pages HTML et des sites Web était massivement utilisée (cf. section 10.4 pour des résultats détaillés). Nous avons analysé la proportion de balises HTML à trois niveaux de granularité HTML : la phrase (constituants élémentaires : les balises *address*, *code*, etc.), le paragraphe (*éléments de bloc* : les balises *p*, *table*, *pre*, *form*, etc.) et la section (les balises *h1*, *h2*, etc.). Par exemple, il existe en moyenne :

Phrases : 17 éléments par page au niveau de la granularité des phrases,

Paragraphes : 46 éléments par page au niveau des paragraphes,

Sections : 3,3 éléments par page au niveau des sections.

Il existe de nombreux travaux qui utilisent la structure interne des pages Web, et qui se basent sur les balises HTML pour l'extraire. Fuller a proposé une telle méthode avant même l'explosion du Web, en fragmentant un document textuel en un ensemble de nœuds et de relations de composition [Fuller et al.93]. Il se base sur la structure du document exprimée à l'aide de SGML et transforme cette structure en un hypertexte, pour permettre la recherche et la navigation. Carchiolo propose de modéliser la structure logique interne des sites Web en combinant la structure décrite à l'aide des balises HTML (comme *p*, *table*, *hr*, etc.) et la similarité structurelle de parties de documents [Carchiolo et al.00]. Un autre exemple d'utilisation de la structure intra-page est la modélisation à l'aide d'une base de données orientée objets basée sur des *unités informationnelles*, qui sont extraites et structurées en fonction des balises HTML [Riahi98]. Toutefois, Riahi met en garde contre l'utilisation abusive des balises *structurelles* à des seules fins de présentation.

D'autres approches consistent à intégrer des données semi-structurées provenant de bases hétérogènes au sein d'un même modèle de documents. Typiquement, une telle approche semi-supervisée consiste à définir un certain nombre de "patrons" (templates), c'est-à-dire des classes de documents ayant une structure précise. Ensuite, l'extraction d'information d'un document semi-structuré consiste à déterminer la classe de document qui lui correspond le mieux, puis à insérer le document dans la structure d'accueil [jH et al.97]. Cette approche a été évaluée sur un corpus homogène de FAQ (*Frequently Asked Questions*, les Questions Fréquemment Posées). Gardarin propose d'identifier le type d'une page HTML (rapport technique, article, etc.) en fonction de l'ordre d'apparition de certaines balises, permettant ainsi de définir un schéma de BD par type de document [Gardarin et al.96]. Atzeni propose le modèle de pages Web ARANEUS (ADM) pour représenter des données semi-structurées et montre un exemple d'application sur un serveur de données bibliographiques homogènes dans [Atzeni et al.97]. On peut aussi utiliser des règles de conversion pour intégrer des documents, dont la structure est hétérogène, au sein d'une Hyperbase [Sedes98]. Ces règles nécessitent de disposer d'une grammaire du document source. L'inconvénient majeur de ce type d'approche est la nécessité de disposer de patrons de structure prédéfinis,

[2]Fédération IMAG : http ://www.imag.fr

ce qui restreint son utilisation à des collections relativement homogènes. Nestorov considère que dans le cas du Web, on ne dispose pas, sauf exception, de patrons établis *a priori* [Nestorov et al.97]. Et selon lui, la taille du corpus et la diversité des documents rendent très complexes et difficiles à utiliser les méthodes précédentes. En effet, même si certains documents du Web sont fortement structurés, cette structure est trop irrégulière pour être modélisée avec un modèle relationnel ou objet.

Enfin, une approche consiste à utiliser la similarité des fragments de textes entre eux [Salton et al.94] pour détecter des liens sémantiques à l'intérieur même d'un document. Ensuite, la distribution de ces liens est utilisée pour extraire une structure entre des fragments reliés. Par exemple, un document dont les liens sémantiques sont uniformément répartis sera jugé comme étant homogène et on ne pourra donc pas en extraire de structuration thématique. Par contre, un document dont les liens sémantiques sont concentrés entre plusieurs paires de fragments de texte, sera jugé comme comportant une structure thématique.

b) Structure hiérarchique intra-site

L'extraction de la structure hiérarchique intra-site est plus délicate. En effet, les standards autorisent la description hypertexte des sites Web, mais ne permettent pas d'expliciter si le site Web représente un grand document structuré, fragmenté pour faciliter la lecture linéaire, ou si le site représente un "vrai" hypertexte avec une lecture par navigation.

Aguiar met l'accent sur la difficulté de la tâche d'identification des "liens structurels" dans un hypertexte, d'autant plus que selon lui, il n'est même pas certain que ces liens existent explicitement [Aguiar et al.00]. Aguiar envisage donc deux hypothèses :

1) Les liens structurels existent mais sont mélangés avec d'autres types de liens. Dans ce cas, il faut envisager une méthode pour trier les liens.

2) Les liens structurels n'existent pas nécessairement. Dans ce cas, il faut les extraire.

Aguiar opte pour la seconde hypothèse, et propose une méthode basée sur l'analyse statistique de la distribution des termes dans les pages et entre les pages, ainsi que la distribution des liens entre les pages pour extraire ces liens structurels [Aguiar et al.00].

Nous avons montré qu'il est possible d'extraire une structure hiérarchique interne des sites Web ([Gery et al.01], cf. section 10.4 pour des résultats détaillés). Le réseau de liens entre les pages d'un même site Web est très dense et il y a peu de liens hors-sites : seulement 2,6% des liens, apparaissant dans 2,4% des pages. Nous en déduisons que l'entité "site Web" a une signification. D'autre part, nous obtenons 30% de relations de composition et 59% de relations de cheminement. La moitié des relations de cheminement sont extraites comme étant linéaires, et l'autre moitié déambulatoires.

La structure intra-site est extraite en analysant la structure du réseau de liens dans son ensemble par Botafogo, qui propose des métriques pour exprimer les propriétés d'un hypertexte [Botafogo et al.91], [Botafogo et al.92]. Il se base sur la matrice M des distances $c_{i\to j}$ entre les nœuds n_i, n_j pris deux à deux au sein d'un même hypertexte[3], pour calculer

[3]Par convention, une distance "infinie" entre deux nœuds est une constante : $c_{i\to j} = K$.

des métriques telles que le *Relative Out Centrality* (ROC, cf. équation 2.1) d'un nœud, qui exprime sa "centralité" dans l'hypertexte.

$$ROC_i = \frac{\sum_{i'} \sum_j c_{i' \to j}}{\sum_j c_{i \to j}} \tag{2.1}$$

Botafogo montre qu'il est possible de différencier automatiquement les liens hiérarchiques (*organizational*) des liens de référence (*cross-reference*), en extrayant une racine et la hiérarchie qui en découle. Ses hypothèses sont qu'une racine permet d'accéder à tous les nœuds sauf ceux qui sont isolés, qu'elle est à une distance faible des autres nœuds, et qu'elle possède un nombre raisonnable de fils. Les deux premières hypothèses sont vérifiées si le nœud possède un ROC élevé. La dernière hypothèse permet d'éliminer les nœuds qui ont uniquement un rôle d'index sans réellement être la racine du site.

2.6.2 Extraction de la structure hypertexte intra-site

L'extraction de la structure hypertexte intra-site revient à déterminer quelle est l'organisation des pages HTML, en dehors d'une organisation "à la documents structurés" dont nous avons déjà parlé. On cherche alors à déterminer le rôle d'une page dans l'hypertexte, plutôt que sa position dans une structure hiérarchique.

Ainsi, Pirolli propose une classification des pages Web d'un site [Pirolli et al.96], selon leur rôle dans l'hypertexte (*functional categories*) :

Head : les pages d'accueil, c'est-à-dire les pages représentant un point d'entrée dans un espace d'information. Cette catégorie se subdivise en page d'accueil d'organisation (*organizational home page*) et en page d'accueil personnelle (*personal home page*).

Index : les pages d'aide à la navigation, comme les tables des matières ou les listes de liens.

Reference : les pages qui sont souvent référencées dans l'hypertexte, comme par exemple une page contenant la définition d'un concept, régulièrement rappelée dans le reste de l'hypertexte.

Content : les pages dont le but n'est pas de faciliter la navigation, mais de délivrer de l'information.

Pirolli montre qu'il est possible de déterminer le type d'une page par une combinaison entre l'analyse de la topologie du réseau de lien, la similarité entre les documents, les statistiques d'utilisation du site (nombre d'accès, navigation, etc.), ainsi que divers autres critères statistiques : titre, auteur, taille de la page, etc. Chaque page est représentée par l'ensemble des *caractéristiques* ("*features*") qui correspondent à ces éléments, et qui sont stockées dans un vecteur ("*Web page feature vectors*"). Les vecteurs sont ensuite comparés à une liste de vecteurs prédéfinis représentant les caractéristiques des différents types de la classification. Par exemple, la page principale d'un site est selon Pirolli caractérisée par un grand nombre de liens entrants ou sortants, une similarité par rapport à ces "pages filles" importante, et un point de passage pour visiter le reste du site.

Ellen Spertus se base sur une classification semblable des pages, et établit un certain nombre de “règles” permettant d’obtenir des informations sur les pages d’un site [Spertus97]. Ces règles se basent sur une information contenue dans les liens, qui permet une classification de ceux-ci par une analyse syntaxique de l’URL.

Dans l’exemple de la figure 2.5, le lien *Down* (respectivement, *Up*) descend (respectivement, monte) dans la structure du site, le lien *Cross* est transversal à l’intérieur d’un site, et le lien *Out* sort du site.

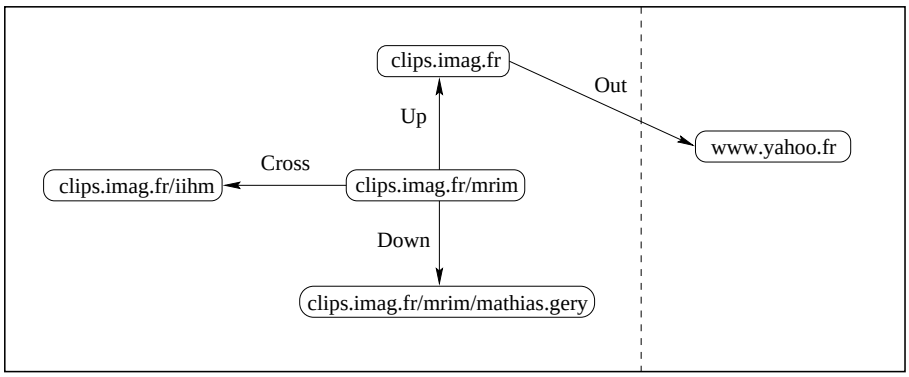

FIG. 2.5 – Direction des liens hypertextes.

Parmi les règles énoncées par Spertus, nous distinguons :

Une page référencée par une page personnelle P à l’aide d’un lien *Down*, est probablement du même auteur que la page P.

Deux liens dont les ancres sources sont proches dans la page HTML traitent probablement d’un sujet similaire, ou possèdent une autre caractéristique commune.

Une page référencée par une page *index* I à l’aide d’un lien *Out*, traite probablement du même sujet que la page I.

Une page référencée par une page *index* I à l’aide d’un lien *Down*, traite probablement d’une spécialisation du sujet traité dans la page I.

Un point intéressant dans cette proposition est l’hypothèse de travail : Spertus considère d’une manière générale, qu’il existe une “structure” du Web, en particulier dans les sites Web, et que cette structure est fortement liée à la hiérarchie des fichiers HTML sur les serveurs Web. Ainsi, la structure des sites pourrait être déduite par les seules URLs.

2.6.3 Extraction de la structure macroscopique du Web

« *Que pouvons nous inférer de l’existence de liens sur le Web ?* » s’interroge Mike Thelwall [Thelwall01], à un niveau macroscopique, c’est-à-dire en prenant en compte uniquement les liens qui sortent des sites Web, donc en considérant les pages Web dans le contexte global du Web et non plus localement à un site.

Nous distinguons deux types de travaux qui tentent de répondre à cette question, en proposant d'extraire une structure macroscopique du Web. Le premier type d'approche s'intéresse à une page ou un site par rapport au reste du Web, tandis que le second type s'intéresse à un groupe de pages ou un groupe de sites.

a) Une page

Cette approche trouve son origine dans l'analyse de citations ou de co-citations dans la littérature scientifique : la bibliométrie [Kessler63] [Small74] [White et al.89]. Il existe un très grand nombre de travaux qui ont proposé une variante de la bibliométrie adaptée au Web [Larson96]. Ces approches cherchent à extraire les pages Web qui jouent un rôle particulier dans le réseau de liens, en se basant sur un "*score*" pour extraire des pages qui font *autorité* (référencées par beaucoup de pages) ou des pages *rayonnantes* (qui référencent beaucoup de pages) [Brin et al.98]. Ce score est éventuellement amélioré en intégrant une notion de *qualité* [Thelwall01] ou de *réputation* [Rafiei et al.00]. Ces notions demeurent toutefois subjectives, puisque l'on ne se base que sur le réseau de liens pour les évaluer. Enfin, on peut aussi se baser sur des scores combinant *autorité* et *rayonnement* [Kleinberg99].

Ces techniques sont directement utilisées pour la Recherche d'Information, généralement en complément d'une approche plus classique, pour réordonner les résultats en considérant par exemple qu'une page est d'autant plus intéressante à retrouver qu'elle joue un rôle d'*autorité* dans l'hypertexte. Ces approches sont présentées plus en détails dans les chapitres 3 et 4.

b) Un groupe de pages

La structure macroscopique du Web est extraite en analysant la connectivité du réseau de liens inter-sites. Il s'agit de détecter des structures qui émergent du Web sans qu'il y ait volonté centralisatrice de les créer. Selon Kleinberg, ce sont des "*communities structures*", des structures qui identifient une communauté d'intérêts [Gibson et al.98] [Kleinberg et al.01].

Typiquement, Bray analyse une collection de 11 millions de pages HTML [Bray96] et montre que, si la densité des liens est importante (en moyenne, une page comporte 14 liens sortants, et seulement 25% des pages sont des "feuilles"), les pages forment des "grappes", qu'il formalise par le concept de *site Web*. Selon Bray, un *site* est un groupe de pages très reliées entre elles, mais peu reliées au reste du Web. En effet, quatre pages sur cinq pointent uniquement sur des pages appartenant à un même *site*. De plus, ces *sites* sont souvent isolés : 80% d'entre eux sont référencés par moins d'une dizaine d'autres *sites*, et 80% d'entre eux n'en référencent aucun. De plus, Bray affirme qu'un *site* correspond généralement à la définition "physique", c'est-à-dire à un ensemble de pages situées sur une même machine.

Les premiers résultats d'une analyse de la topologie du Web à grande échelle ont montré une connectivité forte du réseau de liens, avec une phrase choc : « *Le diamètre du Web est de 19 clics* » [Albert et al.99]. Selon cette étude, portant sur 325 000 pages et 1,5 millions de liens du domaine *.nd.edu*, la moyenne de la plus courte distance entre deux nœuds de la

collection vue comme un graphe orienté serait de $d = 0,35 + 2,06 * log(N)$, avec N le nombre de nœuds. Les auteurs extrapolent cette estimation au Web entier, dont la taille était évaluée à l'époque à 800 millions de documents, pour estimer le diamètre du Web à 18,59 liens et en tirer la rassurante mais quelque peu frustrante conclusion que « *l'information n'est qu'à quelques clics de distance* » :

> « *Fortunately, the surprisingly small diameter of the web means that all information is just a few clicks away* » [Albert et al.99].

Toutefois, d'autres travaux montrent qu'il faut probablement mettre un bémol à cette présumée forte connectivité du Web. Une expérimentation de plus grande envergure a été menée par Broder et a permis de mettre en avant la désormais célèbre macrostructure dite du *nœud papillon* [Broder et al.00] [Kumar et al.00]. Cette expérimentation a bénéficié de données fournies par Altavista [Altavista], ce qui en plus de la taille importante des collections (203 millions de pages HTML et 1,5 milliards de liens) permet d'accéder à des sites "isolés" du reste du Web, dont les URLs ont été fournies à Altavista par "soumission directe" des auteurs de sites.

La richesse de la collection analysée est sans doute un avantage important de cette étude, qui est arrivée à la conclusion que la connectivité du Web est beaucoup moins forte que ce que l'on pensait. Le diamètre de la collection est de 28 liens, mais cette valeur extrapolée à l'ensemble du Web serait de plus de 500 liens selon Broder. De plus, la probabilité qu'il existe un chemin entre deux pages du réseau prises au hasard est seulement de 25%. Si ce chemin existe, alors sa longueur moyenne est de 16 liens.

Broder discerne cinq grands ensembles de pages HTML qui dessinent le fameux *nœud papillon* de la figure 2.6 (source [Broder et al.00]) :

SCC (Strongly Connected Component) : la région centrale du Web, comportant 55 millions de pages. Il existe un chemin pour aller de n'importe quelle page à n'importe quelle autre page de cette région.

IN : la "partie gauche" du Web, comportant 44 millions de pages dites "d'origine". Ces pages peuvent être reliées entre elles, mais, s'il existe des chemins pour aller des pages de IN vers le SCC, il n'existe pas de chemin dans l'autre sens. On peut imaginer qu'il s'agit de nouveaux sites qui n'ont pas encore été "découverts" par le reste du Web.

OUT : la "partie droite" du Web, comportant 44 millions de pages dites "de destination". Il existe des chemins pour aller des pages du SCC vers le OUT, mais il n'existe pas de chemin dans l'autre sens.

Tendrils : les "vrilles" du Web, comportant 44 millions de pages. On peut naviguer des pages de IN vers les vrilles et/ou des vrilles vers les pages de OUT, mais il n'y a pas de liens entre les vrilles et le SCC.

Disconnected : les régions "déconnectées" du Web, comportant au total 17 millions de pages. Ce sont des îlots épars de pages qui ne sont pas connectées aux autres régions.

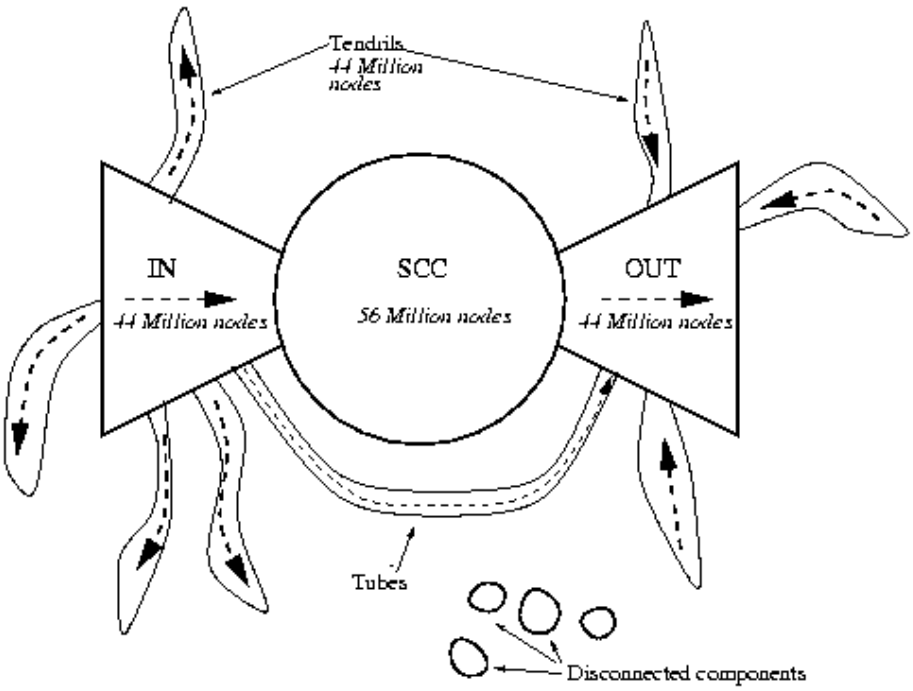

FIG. 2.6 – La théorie du nœud papillon.

Un autre résultat intéressant de cette étude est que si on enlève les liens des pages qui sont référencées par 5 pages ou plus, la taille des différentes régions ne change pas dans de grandes proportions. Cela signifie que la connectivité ne dépend pas d'un petit nombre de pages qui référenceraient un très grand nombre de pages.

2.7 Un exemple concret : le site Web de l'équipe MRIM

Pour mieux comprendre les caractéristiques de la structure du Web que nous venons de présenter, nous utilisons un cas concret : le site Web de l'équipe MRIM[4]. Nous y ferons également référence pour montrer les limites des approches actuelles dans les chapitres 3 et 4, et pour accompagner la description des principes qui régissent notre modèle dans le chapitre 6.

Ce site contient une centaine de pages, pour sa partie principale. Il possède une structure hiérarchique classique (page principale, rubriques, sous-rubriques), avec un bandeau de navigation sur le côté gauche pour naviguer d'une rubrique à l'autre. Il y existe de nombreux liens transversaux entre les rubriques, les sous-rubriques d'une même rubrique, et les sous-rubriques de rubriques différentes. Il existe aussi de nombreux liens externes au site, c'est-à-dire des liens qui référencent d'autres sites Web et décrivent ainsi son contexte.

La page d'accueil du site Web de MRIM est présentée dans la figure suivante :

[4]http ://www-mrim.imag.fr

FIG. 2.7 – La page d'accueil du site Web de l'équipe MRIM.

Nous présentons dans les sections suivantes les différentes structures que nous distinguons sur le site Web de MRIM, à savoir sa structure logique dans la section 2.7.1, sa structure de cheminement dans la section 2.7.2 et sa structure de référence dans les sections 2.7.4 (contexte référençant) et 2.7.3 (contexte référencé).

2.7.1 Architecture du site

La figure 2.8 montre une partie de la hiérarchie du site Web de l'équipe MRIM, qui fait partie du site Web du laboratoire CLIPS (Communication Langagière et Interaction Personne-Système), et regroupe les équipes de recherche ARCADE, GEOD, GETA, IIHM et MRIM ainsi que la plateforme d'expérimentation MULTICOM.

Le site est composé d'une page de présentation (racine du site), à partir de laquelle on peut consulter une description des axes de recherche de l'équipe, des projets, des publications ou encore les pages personnelles des membres.

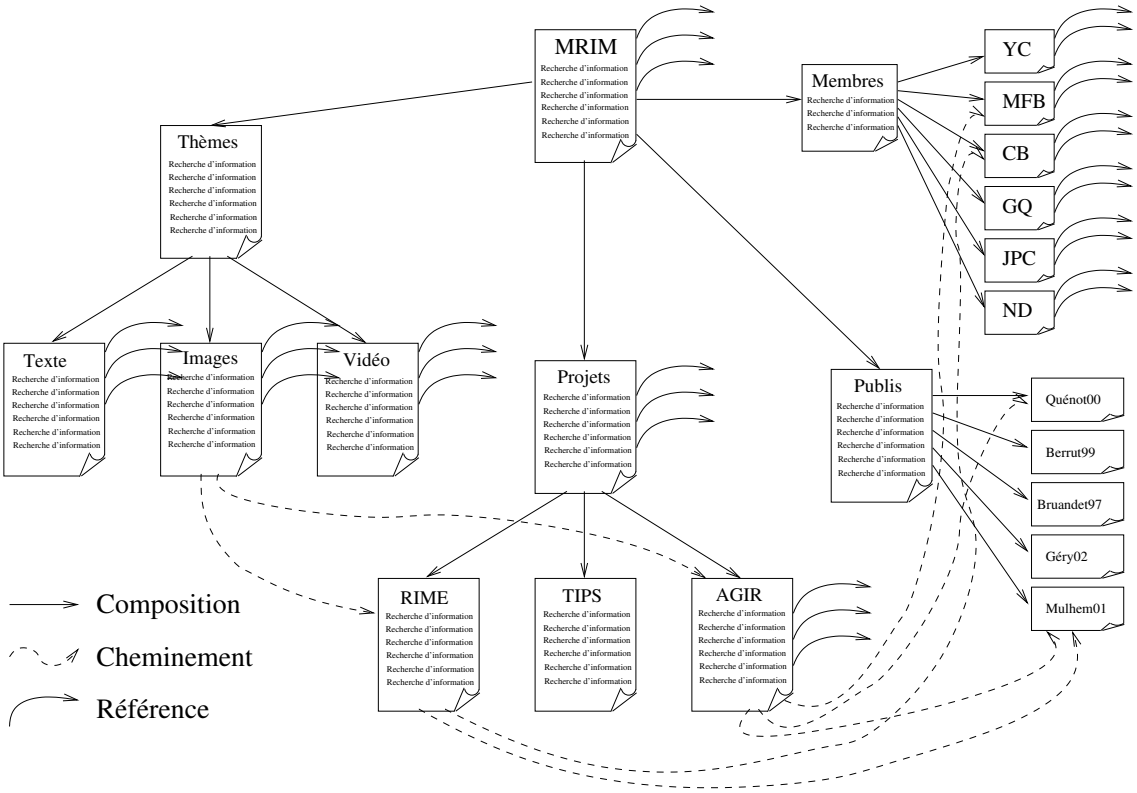

FIG. 2.8 – Architecture (partielle) du site Web de l'équipe MRIM.

Le site de MRIM comporte 9 rubriques (on appelle rubrique les parties du site de premier niveau) :

1) Présentation : la page d'accueil du site, avec un résumé des axes de recherche de l'équipe. Chaque résumé contient la liste des membres de l'équipe impliqués dans l'axe de recherche, et un lien hypertexte vers une page détaillant l'axe de recherche :

1.1) ML : Modèles logiques pour la recherche d'information.

1.2) IF : Indexation d'images fixes.

1.3) DS : Indexation de documents structurés.

1.4) V : Indexation de vidéo.

1.5) RITM : RI textuelle et multilingue.

1.6) FRIC : Filtrage et RI collaborative.

2) Projets : la description des projets en cours ou terminés. Pour certains projets, des pages supplémentaires sont présentes.

3) Membres : la liste des membres de l'équipe, avec des liens vers les pages personnelles.

4) Réalisations : la description des réalisations de l'équipe, dans chacun des axes de recherche, avec parfois des liens vers les projets ou les démonstrations.

5) Démonstrations : la description des démonstrations de l'équipe, avec parfois des liens vers les projets ou les réalisations.

6) Publications : la liste des publications de l'équipe, sans aucun lien sortant.

7) Ressources : des ressources mises à disposition des visiteurs par l'équipe (manuels, etc.).

8) Liens : une liste de liens en rapport avec la RI, presque exclusivement externes.

La structuration présentée organise les informations selon leur nature : projets, personnes, publications, etc. Il n'y a pas d'organisation structurelle suivant le thème des informations présentées. Si on s'intéresse par exemple aux travaux de l'équipe sur la Recherche d'Information appliquée à la vidéo, on trouvera des informations pertinentes aussi bien dans les sous-rubriques de "Projets" (parmi les projets, ceux qui traitent de vidéo) que dans les sous-rubriques de "Membres" (parmi les membres de l'équipe, ceux qui travaillent sur la vidéo), ou encore dans les sous-rubriques de "Réalisations", "Publications", etc. Une telle recherche peut donc être satisfaite par un sous-ensemble organisé des éléments de la structure du site.

2.7.2 Navigation sur le site (chemins de lecture)

Des liens permettent de parcourir les pages de manière linéaire (suivant la structure logique). Sur ce site, pour faciliter la lecture et permettre une autre navigation que celle sur les différents niveaux de la hiérarchie, comme on pourrait le faire pour un document papier, de nombreux liens ont été ajoutés à la structure arborescente. Nous avons observé principalement deux nouveaux types de liens sur ce site : les liens internes au site, qui sont majoritairement transversaux, et les liens externes.

Les liens internes offrent la possibilité d'une navigation thématique dans les différentes sous-parties du site. Par exemple, le tableau suivant montre les liens internes au site qui apparaissent dans les pages décrivant les projets.

Rubrique référencée	Page référencée	Thème
Présentation	Axe RITM	RI textuelle
Présentation	Axe IF	RI images
Présentation	Axe V	RI vidéo
Membres	Chevallet	Personne
Membres	Berrut	Personne
Membres	Quénot	Personne
Réalisations	THEOREME	RI Vidéo
Réalisations	RIME	RI Images
Réalisations	IOTA	RI Textuelle
Publications	Article	RI Vidéo
Publications	Article	RI Textuelle
...	...	...

FIG. 2.9 – Liste des liens sortants internes de la partie "Projets".

Par exemple, un lecteur peut consulter le site Web de MRIM avec un intérêt particulier pour la RI de documents vidéo. Dans ce cas, il suivra les *relations de cheminement* qui ont été définies sur le thème de la vidéo : de la page de présentation du site, il naviguera à la page "Axe de recherche Vidéo", puis choisira d'aller consulter les information sur les projets traitant de vidéo, et il pourra continuer par la consultation des publications sur ce thème, etc.

On appelle "chemin de lecture global" du site Web de MRIM, le chemin de lecture qui passe par chacune des pages du site et permet donc de collecter la totalité de l'information.

2.7.3 Navigation hors du site (information accessible)

Les liens externes permettent de naviguer vers des ressources externes au site, en rapport avec les travaux de l'équipe. Ces ressources, qui font partie du contexte du site, constituent l'*information accessible* du site. Par exemple, le tableau suivant montre les liens externes au site qui apparaissent dans la partie "Projets" :

Page référençante	Site référencé	Thème
Projet TIPS	Laboratoire SISSA	RI Collaborative
Projet TIPS	CERN	RI Collaborative
Projet Théorème	Laboratoire LIMSI-CNRS	RI Vidéo
Projet Théorème	Société VECSYS	RI Vidéo
Projet Perception des Scènes Naturelles	Laboratoire LPE	RI Images Fixes
...	...	...

FIG. 2.10 – Liste de liens sortants externes du site de MRIM.

2.7.4 Référencement du site (méta-information)

Enfin, le site Web de MRIM est référencé par d'autres sites (des laboratoires, des moteurs de recherche, des universités, etc.). L'ensemble des sites référençants fait aussi partie du contexte du site : à ce titre, nous l'appelons la *méta-information* du site. On trouve quelques exemples dans le tableau suivant :

Site référençant	Page référençante	Page référencée	Thème
Projet FERMI	Page principale	Page principale	RI
GDR ISIS	Liste des participants	Page personnelle	RI Multimédia
Équipe IRG (Glasgow)	Liens	Page principale	RI
Laboratoire CSIRO	Réalisations	Page principale	RI Textuelle
Yahoo ! France	Liens (conférences)	Ressources	RI
...	...	...	

FIG. 2.11 – Liste des liens entrants externes du site MRIM.

2.8 Structure du Web et Recherche d'Information

Nous avons montré dans ce chapitre que la structure logique, c'est-à-dire celle qui organise les "documents", était très présente sur le Web. Nous distinguons quatre catégories :

1) La structure intra-page : une structure logique hiérarchique, interne aux pages Web (document structuré).

2-3) La structure intra-site : une structure logique, interne aux sites Web, qui se décompose en deux catégories :

- **2) Hiérarchique :** une structure logique hiérarchique, interne aux sites Web (document structuré).
- **3) Hypertexte :** une structure logique non hiérarchique, interne aux sites Web (hypertexte).

4) Macroscopique : une structure logique macroscopique, externe aux sites Web (hypertexte).

La plupart des moteurs du Web considèrent la structure d'une manière simpliste, qui groupe typiquement les résultats ("clustering" des résultats) : les pages pertinentes provenant d'un même site Web sont rassemblées, pour plus de facilité dans leur consultation. Par ailleurs, les moteurs du Web sont basés sur des modèles de RI qui ont été développés pour des documents textuels classiques depuis déjà plus de 30 ans [Salton71] [vR79] [Salton et al.83b], et qui ne sont pas adaptés aux spécificités du Web. En effet, les documents sont considérés par ces moteurs comme atomiques et indépendants, en prenant l'aspect physique d'un document (la page HTML) comme entité de base. On ne tient pas compte de la structure intra-page, et la structure inter-pages est parfois utilisée mais n'est pas intégrée dans le modèle de documents.

D'un autre côté, il existe de nombreux travaux portant sur l'utilisation de la structure pour la RI, dans le contexte des documents structurés, des hypertextes ou appliqués au Web. Nous présentons dans les chapitres 3 et 4 des travaux qui intègrent cette structure de documents structurés, d'hypertextes ou du Web au processus de RI.

Nous distinguons trois catégories de travaux qui s'intéressent principalement à la phase d'indexation, et trois catégories à la phase d'interrogation :

Phase d'indexation : la modélisation de la structure, la propagation de popularité et la propagation d'information. Ces travaux sont présentés dans le chapitre 3.

Phase d'interrogation : les langages de requêtes sur la structure, l'interrogation structurée et la propagation de pertinence. Ces travaux sont présentés dans le chapitre 4.

Enfin, nous terminerons la présentation de l'état de l'art dans le chapitre 5 avec une synthèse des travaux étudiés et une discussion des limites de ces approches, afin d'esquisser dans la section 5.4 les principes d'un modèle de Recherche d'Information Structurée pour le Web.

Chapitre 3

Intégrer la structure à l'indexation

La phase d'indexation comporte deux parties : la description d'une structure d'accueil pour indexer les documents (le modèle de documents), et le processus d'extraction du contenu des documents pour remplir cette structure d'accueil (l'indexation à proprement parler). Nous distinguons trois types d'approches à l'indexation :

La représentation de la structure : représenter la structure logique au sein du modèle de documents pour pouvoir interroger avec des critères portant sur la structure.

La propagation de popularité : extraire certaines caractéristiques du réseau de liens afin d'en faire une composante à part entière du modèle de documents, pour par exemple privilégier les documents référencés par un grand nombre de documents.

La propagation d'information : prendre en compte le réseau de liens d'un point de vue sémantique, et le répercuter sur l'extraction du contenu sémantique des documents.

3.1 Représentation de la structure logique

La représentation de la structure n'est pas une fin en soi. Ce qui importe c'est l'utilisation qui en est faite par la suite à la phase d'indexation, ou pour l'application d'un langage de requête structuré à la phase d'interrogation. Nous présentons dans cette section les principales approches de modélisation de la structure logique d'un document structuré ou d'un hypertexte, en distinguant les approches qui se basent sur un Système de Gestion de Base de Données (SGBD) relationnel ou à objets, des approches de Recherche d'Information qui se basent sur une représentation moins rigide de la structure.

3.1.1 SGBD et représentation de la structure

De nombreux travaux proposent une modélisation des documents structurés avec un SGBD relationnel ou objets, qui permet une interrogation à l'aide de requêtes SQL/OQL parfois complexes à appréhender. Nous distinguons deux grands types de représentation, selon le type du SGBD utilisé comme structure d'accueil.

L'utilisation de relations (au sens "base de données") pour décrire des documents structurés n'est pas très souple. La difficulté principale réside dans le passage d'une structure hiérarchique à un ensemble de relations BD la représentant, comme le montre l'exemple de modélisation d'une structure hiérarchique ci-dessous [Blake et al.94]. Les documents se conforment à une structure rigide encapsulée dans le schéma de la BD.

TEXT_NODES(nodeid, genid, content) : représentation du contenu des documents, c'est-à-dire un nœud de l'arborescence textuelle, identifié par *nodeid* et de type *genid*.

TEXT_STRUCTURE(a_nodeid, d_nodeid) : représentation des relations de composition. Le nœud identifié par *a_nodeid* est le père (l'ancêtre) du nœud identifié par *d_nodeid*.

TEXT_ATTRIBUTES(nodeid, attr, value) : représentation des attributs externes. Pour le nœud identifié par *nodeid*, l'attribut *attr* prend la valeur *value*.

L'objectif de cette approche est d'intégrer des documents respectant une DTD SGML dans une base de données relationnelle. Le schéma de la BD est rigide, et ne permet qu'une correspondance exacte des requêtes avec les documents.

a) Le langage WebSQL

L'adaptation de cette approche au Web est réalisée avec le système WebSQL, qui stocke les documents et leurs attributs externes dans une table *Document* et le réseau de liens dans une table *Anchor* [Mendelzon et al.96] :

Documents : Document(URL, title, text, type, length, modif)

Liens : Anchor(base, href, label)

b) Le langage POQL

Un SGBD orienté objets permet une description plus riche de la structure. Le langage POQL permet de stocker des documents SGML dans une base de données orientée objets O_2 [Christophides et al.94], [Christophides96]. Dans la représentation fortement typée de POQL, la correspondance stricte est établie entre la DTD et le schéma de la base O_2 : à chaque type de nœud SGML correspond une classe d'objets.

Il existe de nombreux autres travaux fondés sur une base de données relationnelle, qui permettent de représenter la structure. En particulier, on peut citer les nombreux langages de requête adaptés au Web que nous présenterons dans la section 4.1.

3.1.2 SRI et représentation de la structure

Les Systèmes de Recherche d'Information se caractérisent par une plus grande souplesse dans l'indexation et l'interrogation. En particulier, l'utilisateur n'a pas besoin de connaître la structure des documents recherchés. En effet, la structure est représentée pour permettre une indexation structurée transparente pour l'utilisateur. Celui-ci se contente de formuler une requête classique et non structurée, comme par exemple une liste de mots-clés, et le système

se charge de retrouver les meilleurs documents répondant à sa requête, en tenant compte de la structure.

Nous présentons un exemple de SRI qui représente et interroge la structure des documents, dont l'intérêt essentiel réside dans le processus d'indexation (cf. section 3.3) et d'interrogation 4.2.1 plutôt que dans la représentation des documents.

a) Structure arborescente : le système IOTA

Le système IOTA [Kerkouba84], [Defude86] propose d'indexer un document structuré sous la forme d'arbres. Il gère un corpus de documents structurés à un ou plusieurs niveaux (partie, sous-partie, chapitre, section, paragraphe, etc.). On ne considère plus le document comme une entité atomique, mais comme étant composé d'une ou plusieurs *unités d'indexation*. Le découpage d'un document structuré en unités d'indexation se fait en suivant la structure logique du document (cf. figure 3.1). Au lieu d'un index pour chaque document, IOTA utilise donc un index pour chaque unité d'indexation.

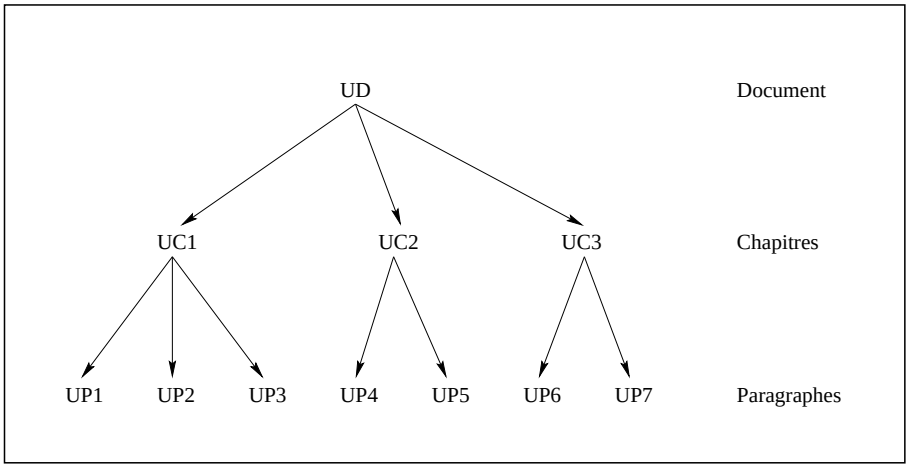

FIG. 3.1 – IOTA : arborescence structurelle d'unités d'indexation.

b) Utilisation de la structure

La structure interne des pages HTML est utilisée dans certains moteurs de recherche du Web pour affiner l'indexation. Typiquement, Boyan propose de considérer plus attentivement les termes présents, par exemple, dans le titre, les en-têtes, les méta-données, les mots écrits en italique ou en gras, les ancres, etc. La pondération de ces termes est alors multipliée par un facteur prédéfini, pour tenir compte de la plus grande représentativité supposée de ceux-ci pour le document [Boyan et al.96].

Une méthode similaire est utilisée par Brin [Brin et al.98], qui considère la fonte de caractères utilisée. Certains moteurs de recherche commerciaux (HotBot, InfoSeek, WebCraw-

ler, etc.) utilisent aussi des méthodes similaires, par exemple en considérant les attributs externes (mots-clés, titre, etc.).

3.2 La propagation de popularité : *PageRank*

Nous appelons *propagation de popularité* (parfois appelée "*macroscopic distillation*" [Chakrabarti01]) une approche initialement dédiée à l'analyse de citations ou de co-citations dans la littérature scientifique : la bibliométrie [Kessler63] [Small74] [White et al.89], adaptée au Web dans [Larson96]. Au lieu de modifier directement l'index des documents, cette méthode consiste à mettre en avant les documents qui jouent un rôle particulier dans le réseau de liens.

Typiquement, il s'agit de la notion de popularité, qui se base sur l'hypothèse : "*une page référencée par un grand nombre de pages est une bonne page*". Cette approche a été popularisée avec le moteur de recherche Google [Google] qui utilise l'algorithme *PageRank* [Brin et al.98] que nous présentons ici.

Une analyse de la connectivité du réseau de liens permet d'extraire des propriétés des pages, comme par exemple le *PageRank* utilisé par Google. L'hypothèse sous-jacente au *PageRank* est récursive : "*une page référencée par un grand nombre de pages populaires est une bonne page*".

Un algorithme itératif est utilisé pour calculer un "score de prestige" PR qui est défini récursivement [Brin et al.98] :

$$PR(p) = (1 - d) + d. \sum_{q \to p} \frac{PR(q)}{C(q)} \tag{3.1}$$

Avec $C(q)$ le nombre de liens sortants de la page q, qui est une page référençant p.

Brin justifie le *PageRank* comme une modélisation du comportement d'un utilisateur "aléatoire", c'est-à-dire un utilisateur choisissant au hasard le lien à suivre sur chaque page q, avec une probabilité $\frac{1}{C(q)}$ pour chaque lien sortant de la page. Le paramètre d représente la probabilité que l'utilisateur arrête sa navigation pour repartir d'une autre page prise au hasard dans le graphe. Avec ces hypothèses, la probabilité qu'un utilisateur visite une page est égale au *PageRank* de la page.

Le PR est propagé le long des liens, jusqu'à ce que la convergence soit atteinte. La figure suivante 3.2 montre un exemple de propagation du *PageRank*.

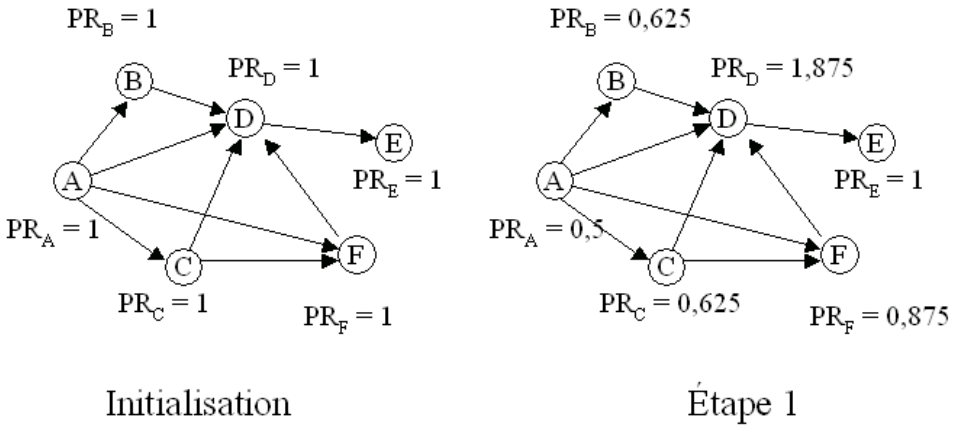

FIG. 3.2 – Exemple de propagation du *PageRank*.

Dans cet exemple, avec le facteur $d = 0,5$, nous calculons le *PageRank* du nœud D à l'étape 1 en propageant le *PageRank* des nœuds A, B, C et F (cf. équation 3.2).

$$\begin{aligned} \text{Etape 1 : } PR_D &= 0,5 + 0,5 * (\tfrac{PR_A}{4} + \tfrac{PR_B}{1} + \tfrac{PR_C}{2} + \tfrac{PR_F}{1}) \\ PR_D &= 0,5 + 0,5 * (\tfrac{1}{4} + \tfrac{1}{1} + \tfrac{1}{2} + \tfrac{1}{1}) \\ PR_D &= 1,875 \\ \text{Etape 2 : } PR_D &= 0,5 + 0,5 * (\tfrac{0,5}{4} + \tfrac{0,625}{1} + \tfrac{0,625}{2} + \tfrac{0,875}{1}) = 1,46875 \end{aligned} \tag{3.2}$$

Pour clarifier l'exemple, nous avons initialisé les valeurs de *PageRank* à 1. Cependant, on remarque que le *PageRank* représente une distribution de probabilité sur les nœuds : la somme de tous les *PageRank* doit donc être égale à 1. Il suffit pour cela d'initialiser chaque *PageRank* à la valeur $\frac{1}{nb_{nœuds}}$.

Le *PageRank* est utilisé pour réordonner la liste de résultats du système. Ainsi, même si le *PageRank* est calculé dès l'indexation des documents, il n'est utilisé qu'à l'interrogation.

3.3 La propagation d'information

La propagation d'information consiste à modifier l'index d'un document en fonction du contenu des documents reliés. Comme la propagation de popularité, cette propagation est donc indépendante de la requête et peut donc être effectuée à la phase d'indexation.

3.3.1 Propagation dans les documents structurés

La propagation d'information peut suivre la structure hiérarchique des documents le long des relations de composition. Par exemple, avec le système IOTA, les feuilles d'un arbre sont indexées de manière classique, et les pondérations des termes des feuilles "remontent" le long de l'arborescence pour indexer les nœuds non feuilles, considérant la composition comme une agrégation de contenus.

a) Propagation statistique : le système IOTA

Le principe est l'interprétation des relations de composition comme une agrégation de contenus : pour chaque unité d'indexation qui n'est pas une feuille, on interprète son contenu sémantique comme étant l'agrégation des contenus de ses unités d'indexation filles. Cette indexation peut être considérée comme une adaptation des fonctions de pondération classiques au cas des documents structurés.

Le processus d'indexation est donc récursif : la première étape consiste à indexer les unités d'indexation minimales (les feuilles de l'arbre). Puis, l'indexation des unités non minimales (nœuds de l'arbre) se fait récursivement, en utilisant l'indexation des descendants. C'est une "remontée" des termes d'indexation dans la hiérarchie du document.

Pour cela, IOTA se base sur les informations suivantes :

Base : Corpus : D_i ; Langage d'indexation : T = $\{t_i\}$, Card(T) = n.

Fréquence totale : FTOT(t_i) est le nombre total d'occurrences de t_i dans le corpus.

Fréquence locale : FLOC(t_i,u) est le nombre d'occurrences de t_i dans l'unité d'indexation u.

Taille : Taille(u) est la taille de l'unité d'indexation u (en nombre de termes).

A partir de ces informations, l'indexation d'une feuille f calcule la pondération de chaque terme t_i de f, et évalue la représentativité mutuelle entre t_i et f :

$$F(t_i, f) = \frac{REP(t_i, f) + REP(f, t_i)}{2} \in [0, 1] \tag{3.3}$$

Le calcul de $F(t_i, f)$ est donc une combinaison de deux critères :

Représentativité de t_i par rapport à f : $REP(t_i, f)$ est la représentativité du terme t_i par rapport à f, qui exprime dans quelle mesure t_i représente l'information contenue dans f. On l'appelle *pouvoir résumant*, calculé de la manière suivante :

$$REP(t_i, f) = \frac{FLOC(t_i, f)}{Taille(f)} \in [0, 1] \tag{3.4}$$

Représentativité de f par rapport à t_i : $REP(f, t_i)$ est la représentativité de f par rapport au terme t_i, qui exprime dans quelle mesure t_i caractérise f par rapport au reste du corpus. On l'appelle *pouvoir discriminant*, calculé de la manière suivante :

$$REP(f, t_i) = \frac{FLOC(t_i, f)}{FTOT(t_i)} \in [0, 1] \tag{3.5}$$

Ensuite, la remontée des pondération se fait avec l'indexation d'un nœud N non feuille. On calcule la pondération de chaque terme t_i de F(t_i,N), à l'aide des calculs effectués précédemment sur les unités d'indexation filles (cf. équations 3.6, 3.7).

$$REP(t_i, N) = \frac{\sum_{u_j \in fils(N)} FLOC(t_i, u_j)}{\sum_{u_j \in fils(N)} Taille(u_j)} \tag{3.6}$$

$$REP(N, t_i) = \sum_{u_j \in fils(N)} REP(u_j, t_i) \tag{3.7}$$

Il y a donc au cours de l'extraction du contenu sémantique une propagation de l'information du "bas" de la structure logique (les feuilles) vers le "haut" (le document).

b) Généralisation de la propagation d'information

L'approche présentée met en avant l'intérêt de la propagation du contenu des documents le long des relations de composition. Chiaramella s'est interrogé sur l'intérêt de généraliser la propagation d'information [Chiaramella et al.96] [Chiaramella97]. Il s'agissait d'affiner le principe développé dans IOTA, selon lequel la composition réalisait une agrégation des contenus sémantiques.

Le modèle développé met en œuvre la notion de propagation d'attribut [Chiaramella97], dans une modélisation du document basée sur la relation de composition et la relation de séquence. Le contenu est lui-même considéré comme étant un attribut. La propagation des attributs est définie selon des classes d'attributs, selon leur comportement lors de la propagation :

Attributs statiques : ce sont les attributs qui ne se propagent pas, comme par exemple le titre d'un document.

Attributs dynamiques descendants (DDA) : ce sont les attributs qui se propagent en descendant dans la hiérarchie, comme par exemple la date de création d'un document.

Attributs dynamiques ascendants (ADA) : ce sont les attributs qui se propagent en remontant dans la hiérarchie, comme par exemple l'auteur d'un document. En effet, on considère que l'auteur (ou les auteurs) du document d composé des documents d_1 et d_2 est l'union de l'auteur (ou des auteurs) des deux documents d_1 et d_2.

Un attribut dynamique ascendant particulier est évidemment l'attribut de contenu, pour lequel un opérateur particulier de composition est défini.

Le système MyPDN [Fourel98], développé dans la droite lignée de ces travaux, propage différents attributs du document le long des relations de composition et de séquence, tout en formalisant la notion de portée des attributs. Par exemple, la portée des attributs liée à la relation de composition réalise une agrégation en remontant, et une dissémination en descendant. Mais la portée est aussi liée aux relations de séquence, auquel cas elle s'intéresse à la propagation en avant/en arrière. L'application de cette notion sur une collection de pages Web nécessite une résolution des conflits sur les portées des attributs.

c) Propagation : théorie de Dempster-Schaffer

Lalmas se fonde sur la théorie de l'évidence de Dempster-Schaffer [Lalmas et al.98] [Lalmas et al.00] pour modéliser la remontée des termes d'indexation dans l'arborescence des documents structurés. Un "*bpa*" ("*basic probability assignment*") est assigné comme

pondération des termes dans les documents (qui sont les *cadres de discernement* de la théorie de Dempster-Schaffer), accompagné d'une *valeur d'incertitude* (“*uncommited belief*”) qui représente l'incertitude sur les *bpas*. Ensuite, la *règle de combinaison* de Dempster-Schaffer est utilisée pour l'agrégation des composants. Il existe une *contrainte de dépendance* : la représentation d'un document doit contenir celle de ses fils.

d) Autres approches

On trouve d'autres travaux proposant une indexation des documents structurés en propageant des informations le long des relations de composition. Lee propose 5 stratégies différentes de remontée d'information le long des relations de composition [Lee et al.96].

D'autres approches se basent sur différents formalismes pour modéliser la propagation d'information. Par exemple, Kazai se base sur l'agrégation floue (*fuzzy aggregation*) [Kazai et al.01]. Enfin, Picard adapte un modèle probabiliste PAS (*Probabilistic Argumentation System*) à la propagation d'information [Picard00] [Picard et al.01].

3.3.2 Propagation dans les hypertextes

Certains systèmes propagent de l'information le long des relations de référence. L'information propagée peut être simplement l'ancre ou un choix de mots-clés dans les documents référençants. L'objectif est d'utiliser le *contexte* des pages pour affiner leur indexation. Nous présentons plusieurs travaux significatifs mettant en œuvre cette approche.

a) Propager les ancres

Google [Brin et al.98] et [Amitay98] considèrent que les ancres[1] donnent souvent une meilleure description de la page référencée que la page elle-même, et intègrent les ancres à l'index de la page :

> « *Anchors often provide more accurate descriptions of Web pages than the pages themselves* » [Brin et al.98].

Davison a montré qu'il y avait une similarité textuelle d'une ancre avec la page référencée, et que l'ancre était discriminante pour la page en question, ce qui souligne l'utilité des ancres [Davison00a], [Davison00b]. La figure 3.3 présente un exemple d'ancres qui montre l'intérêt de leur utilisation [Amitay98].

[1] L'ancre est le mot ou la phrase sur lequel on clique pour activer un lien.

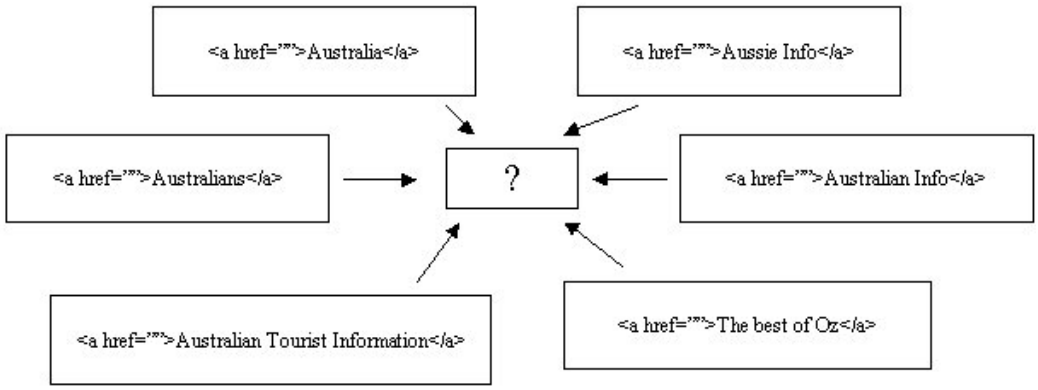

FIG. 3.3 – Exemples d'ancres.

Cette méthode a été expérimentée sur le Web dès 1994 avec le système WWWW (World Wide Web Worm, [Mcbryan94]), en particulier pour pouvoir indexer des documents non textuels automatiquement, et pour élargir l'espace de recherche. En effet, l'utilisation des ancres comme seule description des documents permet d'indexer des documents collectés sur le Web, mais aussi des documents absents de la collecte mais référencés par les documents de la collecte. On trouve un autre exemple de l'utilisation du contexte pour l'indexation automatique de documents non textuels (en l'occurrence des images) dans [Harmandas et al.97] et [Dunlop et al.93], qui combine le contenu des pages et la structure du Web pour trouver des termes en relation avec les images. Ces termes sont principalement sélectionnés parmi le texte entourant les images et dans les pages les référençant.

L'intégration des ancres a été mise en œuvre à grande échelle et popularisée par le moteur de recherche Google [Google], avec une expérimentation portant dès 1998 sur 24 millions de pages et 259 millions d'ancres. Cette technique, appelée "utilisation de critères *off the pages*", se généralise maintenant aux grands moteurs de recherche du Web.

b) Fenêtrage de l'information à propager

La problématique du fenêtrage se pose alors : si l'ancre d'un lien est utile pour indexer la page référencée, on peut se demander s'il n'existerait pas d'autres informations dans la page source, qui pourraient elles aussi aider à la description de la page destination. Ainsi, des travaux proposent d'étendre la "fenêtre" (la zone de texte) des termes à propager, en utilisant par exemple un paragraphe entier.

Ainsi, avec le système ARC, Chakrabarti propose d'améliorer l'algorithme de propagation de pertinence HITS du système système CLEVER ([Kleinberg98], cf. section 4.3) en propageant des ancres [Chakrabarti et al.98]. Les résultats de Chakrabarti montrent que les termes en relation avec la page référencée semblent être concentrés à une distance de 50 caractères de l'ancre elle-même.

c) Extraction du contexte

Enfin, des travaux proposent d'affiner encore cette approche avec l'extraction du contexte d'une page. Par exemple, Aguiar applique des techniques de clustering en utilisant le contenu et la structure, et utilise les clusters de documents ainsi formés pour extraire des termes appartenant au contexte d'une page [Aguiar et al.00]. Aguiar considère donc que le contexte d'une page n'est pas constitué uniquement par ses voisines, mais aussi par les documents sémantiquement reliés. Il n'utilise donc pas directement les liens pour propager l'information, mais il les utilise indirectement par le biais des clusters dont ils participent à la construction.

Pour cela, Aguiar fait intervenir une similarité structurelle entre deux nœuds d_i et d_j, qui tient compte du nombre d'ancêtres en commun, du nombre de descendants en commun, de la connexité des nœuds (en fonction du nombre de chemins qui existent entre eux) et de la longueur du plus court chemin entre eux.

On peut aussi citer le système HyPursuit [Weiss et al.96] qui combine la structure et le contenu pour le clustering hiérarchique de documents Web, et dont la mesure de similarité structurelle a été utilisée dans les travaux précédents.

3.4 Synthèse

La problématique de l'utilisation de la structure pour la RI a été étudiée à la phase d'indexation dans de très nombreux travaux, que ce soit dans un contexte de documents structurés, d'hypertextes, ou plus généralement dans le contexte du Web. Nous présentons une synthèse de ces travaux, suivant les deux axes qui mettent en avant le type d'utilisation de la structure :

La catégorie de structure : ces travaux permettent de représenter la structure au sein du modèle de documents. Il peut s'agir d'une structure hiérarchique intra-page, d'une structure hiérarchique intra-site, d'une structure hypertexte ou d'une structure macroscopique.

La méthode proposée : il peut s'agir de modélisation de la structure, de propagation de popularité ou de propagation d'information.

Nous n'avons pas conservé le point de vue pragmatique distinguant documents structurés et hypertextes, comme cela se fait habituellement dans la littérature, mais nous avons au contraire voulu présenter les travaux selon un point de vue qui permet de mieux saisir les approches scientifiques. Le tableau 3.4 récapitule les approches présentées, qui ont été appliquées aux documents structurés et aux hypertextes, tandis que le 3.5 résume les approches qui ont été appliquées au Web.

Approches	Hiérarchique	
	Intra-page	Intra-site
SMART VSM [Salton71]	- -	- -
SQL-SGML, [Blake et al.94] POQL [Christophides96]	Requêtes structurées	- -
IOTA [Kerkouba84] [Defude86]	Propagation d'information	- -

FIG. 3.4 – Utilisation de la structure à l'indexation : documents structurés et hypertextes.

Approches	Hiérarchique		Hypertexte Intra-site	Macroscopique
	Intra-page	Intra-site		
Moteurs du Web	Importance de certaines balises	- -	- -	- -
WebSQL [Mendelzon et al.96]	Requêtes structurées	- -	- -	- -
Google (*PageRank*) [Brin et al.98]	Importance de certaines balises	Clustering des résultats	- -	Propagation de popularité
MyPDN [Fourel98]	- -	Propagation d'attributs	- -	-
Dempster-Schaffer [Lalmas et al.98]	- -	Propagation d'information	- -	- -
Fuzzy Aggregation [Kazai et al.01], PAS [Picard00]	- - -	Propagation d'information	- - -	- - -
[Amitay98], CLEVER [Chakrabarti et al.98]	- -	- -	- -	Propagation d'information
Aguiar [Aguiar et al.00] HyPursuit [Weiss et al.96]	- -	- -	Extraction du contexte	- -

FIG. 3.5 – Utilisation de la structure à l'indexation : Web.

Chapitre 4

Intégrer la structure à l'interrogation

Nous distinguons deux types d'approches qui s'intéressent à la phase d'interrogation :

Un langage de requêtes sur la structure : il s'agit d'exploiter la structure des documents indexés pour proposer à l'utilisateur une interrogation plus complète avec un langage de requêtes structuré.

L'interrogation structurée : il s'agit d'exploiter la structure des documents indexés sans contraindre l'utilisateur à définir une requête structurée.

La propagation de pertinence : prendre en compte le voisinage des documents en fonction de leur pertinence pour une requête donnée, pour privilégier les documents référencés par un grand nombre de documents **pertinents**.

Nous présentons dans ce chapitre les travaux qui mettent en œuvre des approches de ce type. Certains d'entre eux ont déjà été introduits dans le chapitre 3 avec la présentation du modèle de documents sur lequel ils sont basés.

4.1 Requêtes sur la structure

4.1.1 Exemple de requête structurée

On a vu dans le chapitre 3 un schéma de la BD rigide pour représenter la structure des documents. Cela permet l'utilisation d'un langage de requête ensembliste comme SQL ou OQL pour l'interrogation de données structurées, comme dans l'exemple simple suivant [Blake et al.94] :

```
select nodeid
from TEXT_NODES
where genid=’paragraph’ and content CONTAINS ’Canada’
```

Pour adapter ces techniques à l'hétérogénéité du Web, on parle alors de langage de requête structuré pour l'interrogation de données semi-structurées. Cela nécessite l'intégration de ces données semi-structurées provenant de bases hétérogènes au sein d'un même modèle de documents, comme dans la section 2.3.

Gardarin propose d'identifier le type d'une page HTML (rapport technique, article, etc.) en fonction de l'ordre d'apparition de certaines balises, permettant ainsi de définir un schéma de BD par type de document [Gardarin et al.96]. Il existe de nombreux autres travaux portant sur l'intégration de données semi-structurées [Sedes98], [jH et al.97], [Atzeni et al.97], [Nestorov et al.97], [Riahi98].

a) Correspondance exacte de structure

Le système WebSQL [Mendelzon et al.96] [Mendelzon et al.97] ne permet qu'une correspondance exacte des requêtes avec les documents. Ainsi, on peut poser des requêtes sur la structure comme "*une page qui référence 3 pages et contient 2 sections*" ou des requêtes combinant la structure et le contenu : "*un chapitre contenant un paragraphe qui traite d'ordinateur et qui référence une image*".

La nouveauté de ce type d'approche par rapport à [Blake et al.94] est de permettre d'interroger la structure selon trois types de référence : au sein d'un même document, au sein d'un même serveur Web, et à l'extérieur du serveur. Par exemple, la requête suivante demande tous les documents qui sont référencés par *http ://www.cs.toronto.edu* et qui sont sur le même serveur, grâce à l'utilisation de l'opérateur d'accès local $\rightarrow$.

select x
from Document
x SUCH THAT "http ://www.cs.toronto.edu" $\rightarrow$ x

4.1.2 Requêtes sur les chemins

Le langage POQL permet de définir spécifiquement des requêtes sur les chemins d'un hypertexte [Christophides et al.94], [Christophides96]. Il permet de définir une requête pour trouver les éléments de la première section d'un article "my-article", et qui sont référencés indirectement (par un chemin d'une taille maximum de 2) par un élément de la dernière section :

select y
from last(my-article.sections) PATH_q.reflabel(x), PATH_q.reflabel(y)
where my(articles.section[0] PATH_r(y)

4.1.3 Combinaison structure/contenu

D'autres systèmes permettent de définir des requêtes sur la structure, le contenu ou une combinaison des deux. Par exemple, Proximal Nodes [Navarro95] permet de considérer plusieurs vues sur les documents. Le système MyPDN [Fourel98] permet de combiner contenu, structure et attributs. Ou encore, dans le contexte spécifique des hypertextes, le système $HyperO_2$ définit un langage de requête sur les chemins d'un hypertexte [Amann94].

4.1.4 Les langages de requêtes structurés du Web

On peut citer les nombreux langages de requêtes adaptés au Web comme WebSQL, WebOQL, WebLog, UnQL, W3QL, W3SQL [Konopnicki et al.95], Squeal [Spertus et al.00], ou un langage de requêtes permettant de définir des graphes acycliques [Tan et al.98].

4.2 Interrogation structurée

Parmi les approches qui prennent en compte la structure dans un SRI, certaines ne l'utilisent que pour la représentation et l'indexation, puis mettent en place une interrogation classique. Nous présentons l'interrogation du système IOTA qui interroge la structure des documents à partir d'une requête utilisateur classique. Ensuite, nous présentons une approche qui considère aussi les relations de composition dans l'autre sens, en utilisant l'information globale pour retrouver des sous-parties. Enfin, nous citons une méthode proposant un opérateur "*context*" à l'utilisateur.

4.2.1 SRI et requêtes sur la structure

A partir d'une requête sous la forme d'une expression booléenne de termes (utilisation de ET, OU, SAUF), par exemple (t_1 ET t_2 OU t_3 SAUF t_4), IOTA construit la *liste des références* correspondant à chaque terme de l'expression booléenne : c'est-à-dire la liste des unités d'indexation qui sont indexées par les termes de la requête [Defude86].

On obtient alors une expression booléenne d'unités d'indexation, par exemple : L_1 ET L_2 OU L_3 SAUF L_4. Puis, IOTA en extrait les unités d'indexation vérifiant la requête, en associant à chaque référence résultat une évaluation de pertinence :

Soit $a_1 \in L_1$, $a_2 \in L_2$, avec les pondérations associées : rep_1^{td} et rep_2^{td} la mesure de la représentativité terme-document de a_1 et a_2, rep_1^{dt} et rep_2^{dt} la mesure de la représentativité document-terme de a_1 et a_2.

L_1 **ET** L_2 **:** On renvoie la liste des sous-arbres qui apparaissent dans L_1 et dans L_2 :
- Si $a_1 = a_2$: $a_1 \in$ résultat.
- Si $a_1 \subset a_2$: $a_1 \in$ résultat.
- Si a_1 et a_2 sont disjoints : $a_1 \notin$ résultat, $a_2 \notin$ résultat.

La pondération associée est : $F_{ET^i(a_1,a_2)} = \frac{rep_1^i * rep_2^i}{1 - rep_1^i - rep_2^i + 2 * rep_1^i * rep_2^i}$, avec i ∈ {td,dt}.

L_1 **OU** L_2 **:** On renvoie la liste des sous-arbres qui apparaissent dans L_1, dans L_2 ou dans les deux à la fois :
- Si $a_1 = a_2$: $a_1 \in$ résultat.
- Si $a_1 \subset a_2$: $a_2 \in$ résultat.
- Si a_1 et a_2 sont disjoints : $a_1 \in$ résultat, $a_2 \in$ résultat.

La pondération associée est : $F_{OU^i(a_1,a_2)} = Max(rep_1^i, rep_2^i)$, avec i ∈ {td,dt}.

L_1 **SAUF** L_2 **:** On renvoie la liste des sous-arbres de L_1 qui ne contiennent aucun des sous-arbres apparaissant dans L_2 :

- Si $a_1 = a_2$: $a_1 \notin$ résultat.
- Si $a_1 \subset a_2$: $a_1 \notin$ résultat.
- Si $a_2 \subset a_1$: on enlève a_2 de a_1, et $a_1 \setminus \{a_2\} \in$ résultat.
- Si a_1 et a_2 sont disjoints : $a_1 \in$ résultat.

La pondération associée est : $F_{SAUF^i(a_1,a_2)} = rep_1^i$, avec i $\in$ {td,dt}.

Le système IOTA prend en compte la structure des documentss, tant au niveau de l'indexation que de la phase d'interrogation. Cela constitue une réelle évolution par rapport à un SRI classique.

La granularité de l'indexation est celle d'une unité minimale, qui correspond par exemple à un paragraphe. La phase d'interrogation est une adaptation de l'utilisation d'une requête "booléenne" à ce modèle de documents : la sémantique des opérateurs booléens tient compte du fait que les unités d'indexation sont des arbres, et sont de ce fait susceptibles d'être structurellement dépendantes les unes des autres.

La modélisation d'un document en une arborescence structurelle d'unités d'indexation est donc exploitée au cours de la phase d'interrogation : d'une part pour retrouver les unités d'indexation vérifiant la requête, et d'autre part pour leur affecter une valeur de pondération.

4.2.2 Utilisation bidirectionnelle de la relation de composition

Wilkinson [Wilkinson94] présente des arguments en faveur de l'utilisation des relations de composition dans les deux sens : comment utiliser la structure logique pour retrouver des documents structurés, mais aussi des sous-parties de documents. Le corpus utilisé est constitué de **documents structurés** volumineux divisés en **sections** typées (résumé, sommaire, titre...), correspondant donc à un seul niveau de profondeur.

Wilkinson se propose d'évaluer dans quelle mesure on peut retrouver des documents en se basant uniquement sur les sections, retrouver des documents en utilisant à la fois l'information locale (le contenu des sections) et l'information globale (le contenu du document dans son ensemble), retrouver des sections en se basant uniquement sur les documents, et retrouver des sections en utilisant à la fois l'information locale et l'information globale.

Toutes ces combinaisons mènent à l'évaluation de 18 fonctions de correspondance. Pour la recherche de documents, une fonction "témoin" est comparée à 18 autres fonctions qui combinent de différentes manières le contenu des documents, le contenu des sections, le type des sections, la taille du document, etc. Bien qu'une fonction se basant uniquement sur les sections donne des résultats proches de ceux de la fonction témoin, les meilleurs résultats sont obtenus avec une fonction de correspondance qui combine à la fois le contenu des documents, le contenu des sections et le type des sections. La pertinence d'un document composé de n sections ($S_i \mid i \in [1..n]$) par rapport à une requête Q est évaluée par la formule suivante :

$$Sim(\vec{doc}, \vec{Q}) = \alpha \,.\, Cos(\vec{doc}, \vec{Q}) + \beta \,.\, \sum_{i=1}^{n} [0, 5^{i-1} * type(S_i) * Cos(\vec{S_i}, \vec{Q})]$$

Avec : α et β tels que la partie de l'équation se basant sur le document et celle se basant sur les sections soient du même ordre, i représente le rang des sections, ordonnées selon leur pertinence par rapport à Q, et $type(S_i)$ est un poids associé au type de la section.

En ce qui concerne la recherche de sections, une fonction se basant uniquement sur les documents donne une précision très faible. Les meilleurs résultats sont ceux d'une fonction combinant le contenu des documents, le contenu des sections et le type des sections, qui obtient une bien meilleure précision que la fonction témoin.

Ces résultats montrent que l'extraction du contenu sémantique d'un document peut se faire à partir du contenu de ses sous-parties uniquement, mais que le contenu sémantique d'une section n'est que faiblement relié au contenu du document entier. Un SRI retrouvant des documents et des sections peut donc ne stocker que les sections et les relations de composition, évitant ainsi d'indexer deux fois le même contenu. De plus, l'utilisation à la fois de l'information locale (contenu des sections) et globale (contenu des documents) donne les meilleurs résultats, que ce soit pour retrouver des documents ou pour retrouver des sections.

Il y a donc une propagation de l'information du "bas" de la structure logique (les sections) vers le "haut" (le document) pour retrouver des documents, mais aussi du "haut" vers le "bas" pour retrouver des sections.

4.2.3 Opérateur "*context*"

Dans le cadre de l'extraction du contexte présenté dans la section 3.3, Aguiar offre à l'utilisateur la possibilité de spécifier le contexte de l'information recherchée, avec l'opérateur "*context :*", en plus de l'information recherchée elle-même. On peut alors définir une requête comme donné en exemple dans [Aguiar et al.00] :

requête : "mémoire"
context : diplôme d'études approfondies

4.3 La propagation de pertinence

La propagation de popularité présentée dans la section 3.2 met en avant les documents qui jouent un rôle particulier dans le réseau de liens, avec l'hypothèse : "*une page référencée par un grand nombre de pages populaires est une bonne page*". Cette propagation est donc indépendante de la requête : la propagation est réalisée quelle que soit la pertinence des pages mises en jeu.

4.3.1 Principes de la propagation

On peut améliorer la propagation de popularité en prenant en compte la pertinence des pages. L'hypothèse devient alors une hypothèse de propagation directe de la pertinence : "*une page référencée par un grand nombre de pages* ***pertinentes*** *est une bonne page*".

La propagation se fait soit de la page référencée vers la page référençante comme le suppose cette hypothèse, soit dans l'autre sens, ou encore en combinant les deux possibilités. On peut donc considérer aussi l'hypothèse de propagation inverse de pertinence :

Hypothèse 1 *Une page qui référence un grand nombre de pages* ***pertinentes*** *est une bonne page.*

Contrairement à la propagation de popularité, cette propagation est donc dépendante de la requête. Le revers de la médaille est qu'elle doit donc être effectuée à la phase d'interrogation, et donc à chaque fois que l'utilisateur interroge le système.

Au lieu de modifier directement l'index des pages, on modifie la pertinence d'une page en fonction de la pertinence des pages voisines. On distingue deux variantes : celle qui propage une "vraie" pertinence, c'est-à-dire une valeur non discrète calculée en comparant la requête et les documents, et celle qui propage une pertinence binaire, c'est-à-dire une valeur qui est à 1 si le document est pertinent, et à 0 s'il ne l'est pas.

Les premières applications ont porté sur des hypertextes avec les travaux de Frisse, qui propose de propager la *pertinence* : au lieu d'initialiser le "score de prestige" à 1 comme avec la propagation de popularité (cf. *PageRank*), Frisse initialise le score de chaque page n_i avec sa valeur de pertinence $RSV_{i,0}$ pour la requête [Frisse88] [Frisse et al.89], comme montré dans l'équation suivante :

$$RSV_{i,fin} = RSV_{i,0} + \alpha * \sum_{j \in fils(i)} RSV_{j,0} \tag{4.1}$$

$RSV_{i,fin}$ est la pertinence finale du nœud n_i, $RSV_{i,0}$ est la pertinence initiale du nœud n_i basée sur le contenu, et α est un paramètre destiné à atténuer la propagation.

Cette approche a été expérimentée sur un hypertexte. Toutefois, Frisse n'utilise que des liens hiérarchiques pour propager la pertinence. En cela, cette technique pionnière s'approche plus de la propagation d'information présentée dans la section 3.3. Cette approche a également été généralisée à tous les liens d'un hypertextes par Croft [Croft et al.89a], [Croft et al.93], Frei [Frei et al.92] et Savoy [Savoy96] (*spreading activation*).

4.3.2 Propagation de pertinence pour la génération de "tours guidés"

Une application intéressante de la propagation de pertinence est la méthode de recherche dynamique de "*tours guidés*" dans un hypertexte. Etant donné une requête, Guinan et Smeaton proposent de générer une séquence de nœuds appelée "*tour guidé*" [Guinan et al.92].

Pour cela, la méthode employée se base sur le type des liens[1] (établi manuellement) qui, combiné avec la pertinence des nœuds pour la requête, permet de faire des "choix de navigation" pour construire le "tour guidé". Le calcul de la pertinence des nœuds utilise une méthode de propagation de pertinence inspirée des travaux de Frisse, pour évaluer le "*goodness of an area*", à opposer au "*goodness of a node*".

Cette approche a été expérimentée sur un hypertexte de 551 nœuds[2], à une époque antérieure au Web. En combinant la propagation de pertinence et le concept de "tour guidé", les auteurs ont développé une approche innovante, qui, bien qu'expérimenté sur un hypertexte relativement réduit, préfigure l'évolution des SRI sur le Web.

4.3.3 Algorithme de propagation de pertinence

Les algorithmes mettant en œuvre la propagation de pertinence utilisent un modèle de RI classique pour indexer les documents, qui sont alors considérés comme non structurés, atomiques et indépendants. Puis, ces méthodes enchaînent les étapes suivantes :

1) Root Set : il s'agit de choisir un sous-ensemble de nœuds sur lequel appliquer l'algorithme, c'est-à-dire un ensemble de points de départ de la propagation de pertinence. On peut choisir la totalité des nœuds, ou un sous-ensemble de nœuds sélectionnés, éventuellement augmenté des plus proches voisins. Généralement, un SRI classique est utilisé pour sélectionner les nœuds.

2) Calcul de la pertinence initiale : à chaque nœud est associé une valeur de pertinence par rapport à la requête, indépendamment de son contexte, à l'aide d'une fonction de correspondance classique. Cette valeur peut être binaire ou non, on l'appelle RSV_0 (*Relevance Status Value*).

3) Propagation : on *active* les liens sortant des points de départ, en propageant les RSV_0 pour calculer les RSV_1.

4) Itérations : la propagation est éventuellement réitérée jusqu'à ce qu'une condition de terminaison soit rencontrée. Typiquement, quand un état stable est atteint.

5) Résultats : la liste de nœuds résultat est ordonnée suivant la valeur de pertinence des nœuds à la fin de la propagation.

Les approches étudiées diffèrent en plusieurs points de cet algorithme, comme le SRI sous-jacent, le mode de construction du "*Root Set*", la pertinence initiale binaire ou non, le mode d'arrêt de la propagation, etc. Les variantes de la propagation de pertinence elle-même portent principalement sur les points suivants :

Propagation uniforme : si la propagation est toujours identique, sans considérer certains facteurs comme le type du lien activé, le nombre de liens sortant ou entrant dans la page, etc., on réalise alors une propagation uniforme.

Distance : la propagation peut se limiter aux nœuds du voisinage, ou continuer plus loin.

[1] Il existe des règles de priorités entre les types de liens.

[2] Un hypertexte contenant un cours sur les Bases de Données.

Direction : selon l'hypothèse précédente sur laquelle on se base, la propagation se fait en suivant les liens, en les remontant, ou par une combinaison des deux.

Nous présentons maintenant quelques travaux mettant en œuvre une propagation de pertinence, principalement dans le contexte du Web.

4.3.4 Exemple de propagation de pertinence sur le Web

L'application de la propagation de pertinence au Web est souvent une propagation de pertinence binaire. Cela consiste donc à propager la popularité uniquement parmi un "*Root Set*" de n pages qui ont été jugées pertinentes par le système pour leur contenu, à l'aide d'un système classique, d'où l'appellation de "pertinence binaire" à l'initialisation : RSV_0 de 1 pour les n pages présélectionnées, et de 0 pour toutes les autres. Cette méthode permet de réduire considérablement la taille du graphe sur lequel on applique l'algorithme.

a) Propagation unidirectionnelle

Un SRI classique est utilisé pour obtenir un ensemble initial de n pages Web auquel on ajoute éventuellement les pages du voisinage immédiat, c'est-à-dire celles directement référencées par ces n pages. Ensuite, on réordonne le classement des pages en appliquant une propagation semblable à celle du *PageRank*, restreinte à l'ensemble de pages.

On peut citer les travaux de Jun, qui propage la pertinence selon l'hypothèse de propagation inverse de pertinence, en se basant sur un modèle probabiliste [Jun et al.97]. Seules les pages les plus proches sont considérées pour la propagation de pertinence, et ceci de manière uniforme. Le système de Boyan est similaire, mais se base sur un modèle vectoriel [Boyan et al.96]. Toutefois, il apporte plusieurs améliorations : la propagation se fait en plusieurs itérations, et de cette manière, la pertinence d'une page dépend de ses voisines immédiates dans l'hypertexte, mais aussi de pages plus éloignées. De plus, Boyan introduit un paramètre important : il considère que la "quantité de pertinence" propagée par un lien doit être inversement proportionnelle au nombre de liens sortant de la même page.

Marchiori propose une méthode similaire à celle de Boyan [Marchiori97], avec une propagation inversement proportionnelle au nombre de liens sortants et à l'éloignement dans l'hypertexte des nœuds considérés. Marchiori choisit une interprétation des liens hypertextes comme apportant une information supplémentaire, en considérant qu'une page contient deux types d'information : l'*information textuelle* (le contenu), et l'*hyperinformation* (l'information accessible en suivant les liens hypertextes) :

$$\text{Info(A)} = \text{Contenu(A)} + \text{HyperInfo(A)}$$

b) Hubs et Authorities

Avec les *Hubs* (pages "rayonnantes") et les *Authorities* (pages qui font "autorité") du système HITS (Hypertext Induced Topic Search, [Kleinberg98] [Kleinberg99]), le voisinage

composé des pages référençantes et le voisinage composé des pages référencées sont combinés :

> « Un bon Hub référence beaucoup de bonnes Authorities, et une bonne Authority est référencée par beaucoup de bons Hubs » [Kleinberg99].

La figure 4.1 montre des exemples typiques de Hubs et Authorities (source [Kleinberg99]).

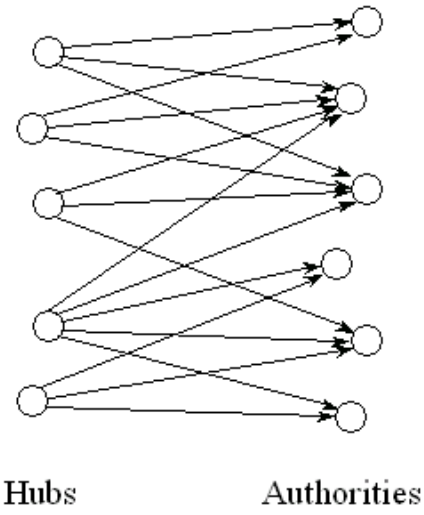

FIG. 4.1 – Pages rayonnantes et pages autorités.

Ces deux concepts sont dépendants (“*mutually reinforcing relationships*”). Ce principe de renforcement mutuel de la notion de pages rayonnantes et de pages autorités se traduit de la manière suivante :

$$Auth_p = \sum_{\forall q \to p} Hub_q \tag{4.2}$$

$$Hub_p = \sum_{\forall p \to q} Auth_q \tag{4.3}$$

Avec une normalisation appropriée, par exemple $\frac{1}{C(p)}$, C(p) représente le nombre de liens sortant de la page p. Il s'agit d'une propagation de pertinence avec une initialisation binaire : tous les $Auth_p$ et les Hub_p sont initialisés à 1.

c) Variantes des Hubs et Authorities

Il existe de nombreux travaux dérivés du *PageRank* et des *Hubs* et *Authorities*, dont on trouve une interprétation théorique dans [Borodin et al.01]. Lempel propose de modéliser

les notions de *Hubs* et d'*Authorities* en utilisant des chaînes de Markov [Lempel et al.00], avec l'algorithme SALSA. Ainsi, les chaînes de Markov des Hubs et des Authorities ont les probabilités de transition suivantes :

$$P_h(i,j) = \sum_{\forall k \mid i \to k \text{ et } j \to k} \frac{1}{|OutLinks_i|.|InLinks_k|} \tag{4.4}$$

$$P_a(i,j) = \sum_{\forall k \mid k \to i \text{ et } k \to j} \frac{1}{|InLinks_i|.|OutLinks_k|} \tag{4.5}$$

Cet algorithme est combiné avec le *PageRank* par Rafiei pour calculer la "*réputation*" d'une page Web sur un sujet donné [Rafiei et al.00].

Enfin, Abchiche propose une autre méthode basée sur le système Mercure [Abchiche01] pour réordonner les résultats en fonction des liens entrants et/ou sortants, en propageant de la pertinence à plusieurs liens de distance.

d) Affiner la propagation de pertinence

Les méthodes que nous venons de présenter ne distinguent généralement que deux sortes de liens : les liens internes et les liens externes à un site. Gurrin s'interroge sur la nécessité de typer les liens, et distingue les liens structurels des liens fonctionnels, proposant de n'utiliser que ce premier type de liens pour la propagation [Gurrin et al.00].

On peut aussi considérer une description des liens pour décider s'il y a lieu de continuer la propagation de pertinence entre deux nœuds [Frei et al.92], ou encore évaluer l'utilité de différents types de liens pour la propagation (citation, co-citation, similarité, cf. [Savoy96]).

Chakrabarti propose un algorithme de *Hubs et de d'Authorities* non binaire, en initialisant une pondération des liens intégrant trois composantes : une valeur par défaut (paramètre du système), une valeur liée à l'appartenance du nœud source et du nœud destination au "*Root Set*", et une valeur calculée en fonction de la présence des termes de la requête dans les nœuds source et destination. Cette pondération est prise en compte dans la propagation.

Enfin, Chakrabarti s'interroge sur le bien fondé de la restriction à la granularité de la page HTML, et présente un "*modèle à grain fin*" en représentant les pages Web suivant leur arbre DOM (Document Object Model). Il propose un algorithme de propagation ("*topic distillation*") adapté à cette modélisation [Chakrabarti01].

4.3.5 Les réseaux d'inférence Bayésiens étendus

Enfin, une approche originale qui revient à faire de la propagation de pertinence est basée sur des réseaux d'inférence Bayésiens. Ces réseaux sont utilisés en RI pour modéliser les documents atomiques. Ils sont étendus par Croft [Croft et al.89b] et Savoy [Savoy et al.90], [Savoy92] pour modéliser les relations existantes entre les nœuds d'un hypertexte.

La figure 4.2 montre un réseau d'inférence qui représente 4 documents d_1, d_2, d_3, d_4 indexés par un ensemble de 7 mots clés $t_1, ...t_7$.

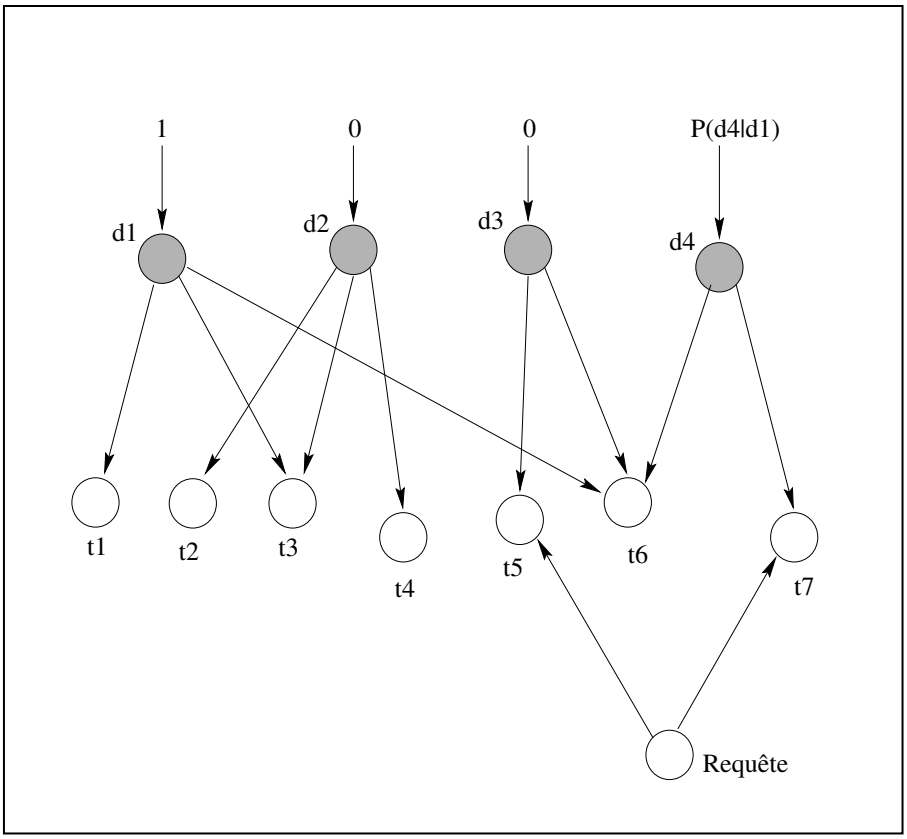

FIG. 4.2 – Réseau d'inférence Bayésien étendu.

Le réseau d'inférence est étendu par l'introduction de probabilités de la forme : $P(d_j|d_k)$, représentant un lien hypertexte entre le nœud d_j et le nœud d_k. Il serait délicat d'introduire directement des dépendances probabilistes entre les nœuds d_i pour représenter les liens hypertextes. En effet, cela nécessiterait de contrôler le formation de cycles dans le réseau, ceux-ci n'étant pas supportés par les réseaux d'inférence. Croft propose donc d'utiliser les probabilités $P(d_j|d_k)$ comme des contraintes sur les nœuds, à la place du 0 ou du 1 habituel.

Dans l'exemple de la figure 4.2, un lien hypertexte de d_1 vers d_4, représenté par la probabilité $P(d_4|d_1)$, est exprimé par une contrainte $P(d_4)$ sur le nœud d_4. Ainsi, au moment du calcul de la pertinence de d_1 par rapport à la requête, il y aura propagation de probabilité à partir de d_1 et, dans une certaine mesure, de d_4 : le document d_4 joue donc un rôle dans l'évaluation de la pertinence de d_1.

4.4 Synthèse

La problématique de l'utilisation de la structure pour la RI à l'interrogation a été étudiée dans de très nombreux travaux, que ce soit dans un contexte de documents structurés, d'hypertextes, ou plus généralement dans le contexte du Web. Nous présentons une synthèse de

ces travaux dans cette section, suivant les deux axes qui mettent en avant le type d'utilisation de la structure, comme pour la synthèse précédente :

La catégorie de structure : ces travaux permettent de représenter la structure au sein du modèle de document. Il peut s'agir d'une structure hiérarchique intra-page, d'une structure hiérarchique intra-site, d'une structure hypertexte ou d'une structure macroscopique.

La méthode proposée : il peut s'agir d'un langage de requêtes structurés, d'interrogation structurée ou de propagation de pertinence.

Chapitre 5

Structure du Web et RI

Nous avons présenté une synthèse de l'état de l'art des travaux qui ont utilisé la structure pour la RI, selon la phase du processus de RI concernée. Il peut s'agir de la phase d'indexation (les travaux intégrant la structure directement dans le modèle de document, cf. section 3.4) et/ou de la phase d'interrogation (les travaux utilisant la structure au moment de la phase d'interrogation, cf. section 4.4).

Nous présentons les avantages et les inconvénients de ces travaux à la section 5.2, en nous appuyant sur l'exemple concret de site Web dont nous avons présenté les caractéristiques, en particulier la structure, dans la section 2.7. Pour cela, nous décrivons dans la section 5.1 un cas concret de RI sur le Web. Nous proposons plusieurs scénarios de RI, pour lesquels des réponses pertinentes existent au sein du site Web de MRIM, afin de dégager les limites des approches actuelles, que nous récapitulons dans la section 5.3. Enfin, ces enseignements nous permettront d'esquisser dans la section 5.4 les grands axes de notre modèle de Recherche d'Information Structurée pour le Web.

5.1 Exemple de RI sur le site de MRIM

Voici une requête que l'on peut poser sur un moteur de recherche généraliste du Web, et pour laquelle certaines des pages du site Web de MRIM sont pertinentes :

Besoin de l'utilisateur : il concerne des informations sur "*les travaux, les publications, les développements et les résultats des équipes universitaires de recherche françaises sur la modélisation et la recherche de vidéo*".

Requête classique : "travaux équipe publications développement résultats modélisation recherche information vidéo".

Le résultat d'une telle recherche peut se présenter sous diverses formes, selon le système utilisé et le modèle de documents sur lequel il se base. Déterminer quelle serait la meilleure réponse à apporter à une telle requête est une problématique complexe, dépendant d'une multitude de paramètres, et à laquelle nous ne prétendons pas apporter de réponse définitive. La connaissance détaillée du site de MRIM tel que nous l'avons présenté dans la section 2.7 nous permet cependant d'avancer plusieurs éléments de réponse.

5.1.1 Réponse pertinente : un document atomique

Une réponse classique serait une liste ordonnée de pages HTML, comme en produisent les moteurs de recherche actuels du Web. Une réponse pertinente, pourra donc être une page HTML contenant à elle seule toutes les informations recherchées.

5.1.2 Réponse pertinente : un document structuré

Nous constatons que les informations recherchées ne sont pas concentrées en une seule et unique page HTML, ni même en une seule sous-partie identifiée du site, mais qu'elles sont au contraire disséminées dans plusieurs pages HTML à travers le site. Une réponse satisfaisante à une requête doit contenir le maximum d'information pertinente pour remplir le critère de *rappel* et éviter le *silence*, mais doit aussi contenir le minimum d'information non pertinente, pour remplir le critère de *précision* et éviter le *bruit*. Une réponse pertinente pourra donc être présentée sous la forme d'un document structuré, dont le choix de granularité optimise le compromis entre silence et bruit.

5.1.3 Réponse pertinente : un chemin de lecture

Nous proposons comme résultat d'une telle recherche un "*chemin de lecture*", c'est-à-dire un point d'entrée dans l'hypertexte et un enchaînement des pages pertinentes à consulter, en respectant les liens de cheminement initialement proposés par l'auteur. De cette manière, le résultat pourra être assimilé au choix d'un document virtuel parmi toutes les combinaisons initialement proposées par l'auteur pour consulter le site. Dans notre exemple, ce pourrait être le chemin de lecture représenté par la figure 5.1, débutant par le point d'entrée de la page "Vidéo" et poursuivant par les pages "Projets vidéo", "Publications vidéo", etc.

5.1.4 Réponse pertinente : un chemin de lecture en contexte

Enfin, nous pensons qu'il est aussi nécessaire de prendre en compte le contexte des chemins de lecture. Par exemple, le chemin de lecture présenté précédemment passe par la page Web "Projets vidéo" qui référence et qui est référencée par plusieurs sites Web de laboratoires travaillant sur la RI vidéo. C'est un argument supplémentaire en faveur du choix de ce chemin de lecture en réponse à la requête de l'utilisateur, en supposant que celui-ci ait manifesté le désir de naviguer à partir des résultats. En effet, si l'utilisateur a d'ores et déjà choisi de ne pas consulter l'information accessible, il n'est pas utile de la considérer pour évaluer la pertinence des chemins de lecture.

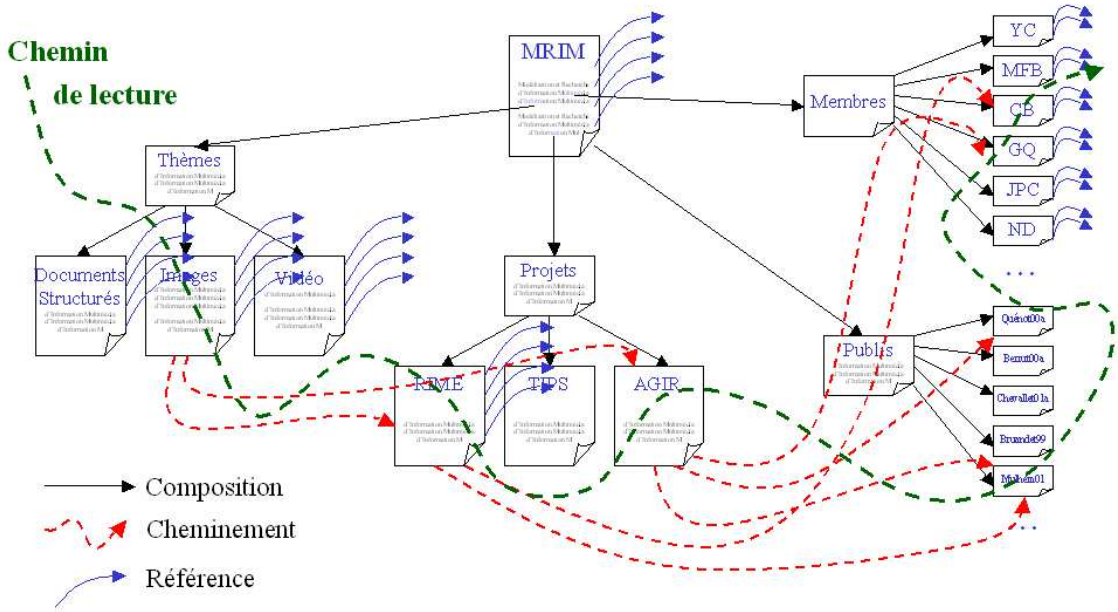

FIG. 5.1 – Un exemple de résultat : un chemin de lecture.

5.2 Discussion des approches de l'état de l'art

5.2.1 RI atomique

Le thème dans cet exemple est suffisamment précis pour définir une requête sans ambiguïté : un SRI devrait pouvoir retrouver les documents pertinents aisément. L'hypothèse simplificatrice des modèles de RI classiques et de la plupart des moteurs de recherche du Web, qui indexent les pages HTML de manière atomique et indépendante, entraîne pour ce genre de requête des résultats insatisfaisants. En effet, les informations recherchées ne sont pas matériellement concentrées dans une même page HTML, mais sont au contraire disséminées dans plusieurs pages du site : la page de présentation, la page des publications, la page des projets, la page des réalisations, etc. Il n'existe donc aucune page HTML qui contienne à elle seule tous les termes de la requête, et les systèmes atomiques sont incapables de faire le lien entre ces pages.

Les résultats obtenus souffrent alors de deux défauts :

Rappel faible : soit le système ne retrouvera aucune des pages pertinentes, car aucune d'entre elle ne contient tous les termes de la requête.

Précision faible : soit une partie des pages pertinentes seront retrouvées, mais dans ce cas elles seront jugées peu pertinentes car ne contenant qu'une partie des termes de la requête, et elles seront noyées parmi des milliers d'autres réponses.

5.2.2 Requêtes sur la structure

Dans le contexte du Web, les approches de requêtes sur la structure (cf. section 4.1) imposent une interrogation contrainte par le schéma de la BD, ce qui demande une connaissance du schéma pour pouvoir définir une requête. Le Web est hétérogène, y compris dans sa structure : il est donc difficile d'indexer les documents selon une structure rigide encapsulée dans le schéma de la BD. Enfin, les approches d'interrogation de la structure orientées SGBD sont coûteuses : entre autres, la correspondance de chemins dans un graphe est un problème complexe, et le calcul se fait à la phase d'interrogation, c'est-à-dire au moment où le temps de réponse doit être faible. Par exemple, WebSQL a besoin de plusieurs minutes pour résoudre une requête sur un graphe de 500 nœuds [Mendelzon et al.96], ce qui rend la méthode inapplicable à grande échelle sur le Web. Ces approches sont très utiles dans le cas de collections de données relativement homogènes, mais ne sont pas adaptées à l'hétérogénéité et/ou la taille du Web.

Dans notre exemple, une requête sur la structure pourrait permettre de retrouver un chemin comme celui proposé, mais pour cela il faudrait que l'utilisateur connaisse avec précision, d'une part la structure du chemin recherché (une page d'entrée suivie d'une page de présentation, etc.) et d'autre part la dissémination et l'enchaînement des informations qu'il recherche (présentation générale, puis projets, puis publications, etc.). Ce type de requête peut être utilisé dans un contexte très spécifique, mais n'est pas adapté au Web. En effet, il est rare de connaître à l'avance la structure des informations que l'on recherche.

5.2.3 Intégrer la structure à l'indexation

L'intégration de la structure à l'indexation (cf. chapitre 3) se heurte sur le Web au problème du "sac de liens" : les liens ne sont pas typés. Les approches s'intéressant au Web appliquent la propagation sans discernement, se contentant au mieux de distinguer les liens internes au site des liens provenant de l'extérieur. Or, nous pensons qu'il est indispensable d'analyser la nature des liens plus finement pour pouvoir les utiliser pour l'indexation. D'autre part, il n'y a pas d'intégration des relations de composition, cheminement et référence au sein d'un modèle de documents : la propagation n'est qu'une simple surcouche à l'indexation classique. Seuls des travaux comme ceux de Weiss [Weiss et al.96] et Aguiar [Aguiar et al.00] proposent de prendre en compte le contexte à l'indexation.

Dans notre exemple, les méthodes de propagation d'information suivant une arborescence pourraient indexer le site entier comme étant un seul document, qui contiendrait alors tous les termes de la requête, mais noyés au milieu des autres informations du site. La propagation d'information le long des relations de référence donnerait soit une propagation trop rapprochée (restreinte au voisinage immédiat d'une page), soit une propagation diffuse de tous les termes dans tout le site, soit une absence de propagation à l'intérieur d'un site.

5.2.4 Intégrer la structure à l'interrogation

Les approches du chapitre 4 sont limitées elles aussi par le "sac de liens" : le Web est modélisé par un graphe orienté, avec les pages HTML comme nœuds et les liens hypertextes comme arcs. Très peu d'approches tentent d'analyser les liens et de comprendre leur rôle en terme de description de l'information. Une illustration amusante de l'impact négatif du "sac de liens" est la requête "more evil than satan himself", pour laquelle Google proposait, en octobre 1999, le site Web de Microsoft comme réponse la plus pertinente[1]. Cette réponse est probablement la conséquence de l'existence à proximité du terme "evil" de nombreux liens de la part d'internautes voulant dénoncer les pratiques commerciales de Microsoft.

Avec ces approches se pose aussi le problème du "démarrage à froid" : un site Web nouvellement créé sera peu compétitif du fait de l'absence d'autres sites le référençant, et cela, même si le site est de très bonne qualité. Il peut alors rentrer dans un cercle vicieux : un site mal classé sera moins souvent visité, aura donc moins de chances d'être référencé par d'autres sites, ce qui entraînera un mauvais classement. De plus, une page qui est mal classée du fait des insuffisances du SRI classique sous-jacent, n'apparaîtra pas dans les n premiers résultats préliminaires, et ne pourra donc pas être "repêchée" par la propagation de pertinence (binaire ou non). Pourtant, il existe des *pages de liens* pas forcément très pertinentes pour leur contenu, mais très pertinentes pour leurs liens, qu'il serait intéressant de retrouver.

Le malheur des uns faisant le bonheur des autres, un inconvénient comparable est le risque de renforcement auto-accéléré des positions dominantes, selon le principe du "*rich get richer*" (auto-accélération), comme l'explique Bourdoncle [Bourdoncle et al.00] :

> « *De plus, les techniques fondées exclusivement sur la popularité présentent un danger réel de renforcement auto-accéléré des positions dominantes, puisqu'il suffit d'être déjà visible sur le réseau pour le devenir encore plus* ».

La propagation de pertinence a néanmoins le mérite de se restreindre à un petit nombre de pages "pertinentes", ce qui évite la propagation sans discernement à travers tout le Web. Mais ces calculs se font au moment de l'interrogation, ce qui limite la distance de propagation, qui est alors le plus souvent restreinte au voisinage immédiat d'une page. Des travaux propagent à "plusieurs liens de distance", comme Marchiori qui choisit finalement, pour des raisons de performance, de se limiter à une distance d'un seul lien [Marchiori97].

Depuis plusieurs années, le moteur Google est le moteur du Web le plus innovant en matière d'utilisation des liens. Son succès vient principalement de deux avantages qu'il conserve sur ses concurrents. Tout d'abord, la taille de sa base le place au premier rang en terme de nombre de pages indexées, ce qui lui permet de retrouver des pages que les autres moteurs n'ont pas. Ensuite, Google privilégie la précision des résultats en retournant des pages très référencées, et profite de la profusion d'information qui fait que pour la plupart des requêtes, il existe une telle page. Mais la comparaison de Google avec les systèmes académiques est délicate. Un grand nombre d'expérimentations ont été menées dans le cadre

[1] Voir "More Evil Than Dr. Evil ?", http ://searchenginewatch.com/sereport/99/11-google.html

de la conférence TREC[2], avec comme objectif l'évaluation de méthodes dérivées de la propagation d'information [Savoy et al.00b], de popularité [Gurrin et al.00], ou de pertinence [Crivellari et al.00]. Ces expérimentations ont montré que ces méthodes n'apportent pas un gain de qualité significatif [Hawking00], ce qui amène Savoy à s'interroger sur l'utilité des hyperliens [Savoy et al.00a], [Savoy et al.01].

Enfin, le fait de considérer un document comme étant une page HTML pose le problème du "sac de nœuds", non typés et de granularité arbitraire, qui peut expliquer le peu d'amélioration dans la qualité des résultats. Il y a pourtant de grandes différences dans la nature d'un paragraphe, d'un livre, d'une page de liens, d'une page principale, d'une page personnelle, etc. Une granularité différente est utilisée par Craswell dans le cadre d'une recherche de sites de la piste Web de TREC (seule les pages principales sont considérées comme pertinentes), qui montre une amélioration des résultats [Craswell et al.01].

La propagation de popularité ou de pertinence pourrait juger comme pertinente la page principale du site, qui est probablement la plus référencée mais pas forcément la plus pertinente. Mais les méthodes de propagation de pertinence ne travaillent généralement pas à l'intérieur d'un même site (seuls les liens externes sont utilisés).

5.3 Limite des approches actuelles

Nous récapitulons dans cette section les insuffisances et les inconvénients des approches présentées, dans la perspective de la recherche d'une information structurée sur le Web.

Les moteurs actuels du Web sont basés sur des modèles de RI qui ont été développés pour des documents textuels classiques depuis déjà plus de 30 ans [Salton71] [vR79] [Salton et al.83b]. Ces modèles ont été très étudiés dans le contexte de documents classiques : atomiques, "plats" et indépendants. De ce fait, la plupart des moteurs considèrent le Web comme un ensemble de documents atomiques et indépendant, dont la granularité est celle d'une page HTML.

Le choix de la granularité a été fait pour des raisons pratiques : on fait alors l'hypothèse que l'auteur d'un page Web cherche à communiquer des informations de la granularité d'une page HTML, comme on le fait avec des documents classiques et des documents papiers. Mais ce n'est pas toujours le cas, et cette hypothèse est souvent prise en défaut. De plus, beaucoup de moteurs ignorent purement et simplement les liens au cours de leur processus de RI. D'autres approches considèrent le Web comme un graphe orienté : les nœuds sont des pages HTML et les arcs sont des liens hypertextes, mais peu d'entre eux utilisent la structure du Web avec plus de finesse. Les systèmes ne tiennent donc pas compte de la structure intra-page, et si la structure inter-pages est parfois utilisée, elle n'est pas intégrée dans le modèle de documents. Les pages HTML étant indexées indépendamment les unes des autres, elles perdent leur contexte.

[2]TREC (Text REtrieval Conference) : http ://trec.nist.gov

Parmi les limites des systèmes présentés précédemment, qui intègrent la structure des documents et/ou de l'hypertexte dans le processus de RI, nous pouvons citer :

Le sac de mots : l'indexation des documents utilise un langage d'indexation simpliste à base de mots-clés, plus ou moins finement sélectionnés sur des critères statistiques, et sans tenir compte des éventuelles dépendances entre les termes.

Le sac de nœuds : les documents ne sont pas typés, et sont tous indexés de la même manière. Il y a pourtant de grandes différences dans la nature d'un paragraphe, d'un livre entier, d'une page de liens, d'une page d'entrée, d'une page personnelle, etc.

Le sac de liens : les systèmes utilisant les liens distinguent uniquement les liens internes à un site de ceux provenant de l'extérieur du site, ou plus généralement ne différencient pas les liens. Or, il y a aussi de grandes différences entre les liens de composition, de référence, les liens purement organisationnels, etc.

L'atomicité des documents : la plupart de ces systèmes ne tiennent pas compte de la **structure** intra-page, qu'elle soit implicite ou décrite à l'aide de HTML.

L'indépendance des documents : les pages HTML sont indexées indépendamment les unes des autres, et perdent donc leur contexte.

La structure hypertexte : ces systèmes ne tiennent pas compte de la structure inter-pages, les relations implicites ou explicites qui existent entre ces pages.

Nous avons longuement disserté sur l'inadaptation des modèles de RI classiques, qui considèrent les "documents" comme étant atomiques, "plats" et indépendants, au cas de la RI sur le Web, qui est structuré, hétérogène dans son contenu comme dans sa présentation et dans sa structure, et dont les documents sont interconnectés. Au delà de cette constatation, nous pensons que les insuffisances des approches présentées sont des conséquences directes du manque de considération de l'aspect "sens" dans la modélisation de la RI sur le Web. En effet, un index doit représenter l'information relative à un document, et mettre en évidence sa sémantique en vue d'une requête. L'objectif d'un modèle de RI structuré pour le Web est de prendre en compte la structure, ce qui nécessite de s'interroger sur la sémantique de la structure, et donc des relations, pour pouvoir comprendre son impact sur la description de l'information.

Le même constat est fait par Bourdoncle, qui considère les méthodes comme celles proposées par Brin avec le moteur Google [Brin et al.98] ou Kleinberg avec le système CLEVER [Kleinberg99] comme étant basées sur une notion *ad hoc* de "popularité" :

> « Ainsi, des techniques comme l'utilisation des liens hypertextes ou les analyses comportementales[3] reposent, pour filtrer et hiérarchiser l'information fournie à l'utilisateur, sur une notion *ad hoc* de "popularité" qui est parfois contestable comme mécanisme de validation du savoir » [Bourdoncle et al.00].

Cette popularité, en raison du problème de l'auto-accélération que nous avons évoqué, est même un réel danger pour la diffusion "démocratique" de l'information, selon Bourdoncle :

[3]Utilisation de statistiques d'accès aux pages pour privilégier les pages les plus souvent choisies par les utilisateurs.

> « Ils menacent directement, si l'on n'y prend pas garde, une certaine forme de démocratie sur le réseau, et compromettent ce que l'on pourrait qualifier de "service universel" d'accès à l'information » [Bourdoncle et al.00].

De plus, Chakrabarti constate qu'il y a de plus en plus d'éléments perturbateurs sur le Web pour ces algorithmes, comme les bandeaux et les liens publicitaires, ou le *spam* de liens. Ces considérations amènent Chakrabarti à prôner le développement d'une architecture "propre" (c'est-à-dire permettant de s'abstraire du bruit occasionné par le *spam*) pour indexer du contenu et de la structure, mais aussi pour pouvoir appliquer ces algorithmes de propagation sur un graphe de nœuds et de liens "propre" et adapté [Chakrabarti01].

5.4 Vers un modèle de RI adapté au Web

Les moteurs de recherche actuels ne sont donc pas adaptés aux caractéristiques des documents du Web. Un axe de recherche prometteur consiste à étudier l'impact de la structure du Web sur l'indexation et l'interrogation. Nous pensons qu'il est nécessaire d'intégrer la structure au sein du modèle de documents. En effet, il ne suffit pas de rajouter une opération à la correspondance ou de répercuter la structure sur l'index classique des documents. L'hypertexte apporte une nouvelle dimension à la diffusion de l'information, en particulier sur le Web : pas seulement dans la présentation de l'information ou dans la structure logique des documents, mais aussi dans la structure même de l'information, à un niveau sémantique. Par exemple, la lecture d'un document structuré est linéaire, alors qu'un hypertexte permet une lecture non linéaire.

De plus, nous allons dans le sens de Chakrabarti, en considérant comme indispensable le développement d'une architecture "propre" pour appliquer des méthodes de propagation de popularité, d'information ou de pertinence.

Pour toutes ces raisons, nous proposons d'intégrer les relations de composition, de séquence et de référence au sein même du modèle de documents. Cette intégration ne doit pas se contenter d'une simple surcouche à un modèle existant, mais doit répercuter l'apport de la structure du Web au niveau sémantique afin de permettre une réelle indexation structurée.

D'un côté, les documents du Web ont des caractéristiques de documents structurés grâce à l'utilisation de langages comme HTML, et d'un autre côté, nous avons présenté le Web comme étant un hypertexte distribué à l'échelle planétaire grâce à l'utilisation de la norme URL pour définir des liens. La dualité documents structurés/hypertextes implique non seulement l'existence d'une structure du Web, mais l'existence de plusieurs structures : structure hiérarchique, structure hypertexte et structure macroscopique. Chacune des structures du Web est une composante essentielle de la description de l'information.

Selon l'utilisation de HTML et/ou de URL pour la décrire, nous distinguons donc plusieurs niveaux de structure : les pages Web possèdent une structure interne (grâce au langage HTML) et sont connectées par un réseau de liens hypertextes (grâce à la norme URL). Ce réseau de liens décrit une structure externe, composée de la structure des sites Web (interne

à un site) et de la structure macroscopique du Web (externe aux sites). Nous faisons donc la distinction entre la structure de type “document structuré” (structure arborescente, sens de lecture linéaire) et la structure de type “hypertexte” (structure de graphe, lecture non linéaire). De nombreux travaux ont porté sur l’extraction de structure sur le Web, comme nous l’avons vu dans les sections 2.3 (documents structurés) et 2.5 (hypertextes).

Notre problématique consiste à intégrer la structure du Web (ou **les** structures du Web) au sein d’un modèle de RI Structurée : quelle structure peut-on trouver, comment l’extraire et l’identifier, comment la modéliser au sein d’un modèle de documents, et comment l’utiliser à la phase d’interrogation ? Cela nécessite de s’interroger sur la sémantique de la structure, et donc des relations, pour pouvoir comprendre son impact sur la description de l’information. En d’autres termes, comment l’auteur d’un site Web utilise-t-il les relations pour décrire le message qu’il veut faire passer ? Est-ce que le fait de référencer une page Web indique une appréciation de la part de l’auteur ? Une similarité entre les documents ? Un conseil de lecture ? Un contre-exemple ? Une composition des contenus ?

Nous considérons les sites Web à la fois du point de vue des documents structurés et du point de vue des hypertextes. Un document structuré possède une structure hiérarchique basée sur la relation de composition. Un hypertexte possède une structure de graphe, basée sur les relations de cheminement et de référence. Les relations de cheminement sont des références internes au site : l’auteur propose au lecteur de poursuivre sa lecture dans un autre nœud du graphe. Les relations de référence sont externes au site : l’auteur propose au lecteur d’aller consulter d’autres sites. Cela nous permet de définir une typologie simple : les relations de composition, de cheminement et de référence.

Ces trois types de relations jouent un rôle majeur dans la construction de l’information, en raison de leur impact sur la lecture des “documents”. La prise en compte de cette typologie est donc essentielle pour la RI sur le Web, et permet de répondre à notre problématique de RI Structurée. Cependant, elle pourrait être affinée et décomposée en plusieurs sous-types, en particulier dans le cas de la relation de référence. Un modèle de RI adapté au Web doit prendre en compte ces trois types et la structure associée, et les répercuter sur le modèle de documents.

Deuxième partie

Un modèle de Recherche d'Information Structurée en contexte

Chapitre 6

L'information structurée sur le Web

Le sens d'un mot n'est autre que l'écheveau scintillant de concepts et d'images qui luisent un instant autour de lui. La rémanence de cette clarté sémantique orientera l'extension du graphe lumineux déclenché par le mot suivant, et ainsi de suite, jusqu'à ce qu'une forme particulière, une image globale brille un instant dans la nuit du sens. Elle transformera peut- être imperceptiblement la carte du ciel, puis disparaîtra pour laisser place à d'autres constellations.

Pierre Lévy - Les technologies de l'intelligence

6.1 Documents du Web

Le modèle de Recherche d'Information (RI) est construit autour de la notion de **document**. La définition d'un document est un problème ouvert dans le contexte du Web. Pour des raisons de simplicité de mise en œuvre, nous avons vu dans le chapitre 3 que cette notion est souvent réduite à la notion physique de page HTML. Nous préférons adopter une définition plus générale :

Définition 1 *Un **document** du Web est un support informatique qui véhicule une information produite par une source (un auteur ou un groupe d'auteurs) à destination des lecteurs du Web, en utilisant un code approprié (comme le langage HTML). Le **document** est le terme générique pour désigner aussi bien les documents atomiques que les documents structurés, les chemins de lecture ou les hyperdocuments.*

Dans le cadre de cette thèse, nous restreignons le modèle d'hyperdocuments au média "texte". Un document du Web peut être un paragraphe d'une page HTML, un chapitre ou un site Web entier. Un document structuré associé à un ou plusieurs chemins de lecture, placé dans le contexte d'autres hyperdocuments, est appelé **hyperdocument en contexte**.

Le rôle de ce chapitre est de donner de manière informelle notre point de vue sur la description et la compréhension de l'information sur le Web, afin d'introduire les notions utilisées dans la description formelle de notre modèle de RI dans le chapitre 7.

6.2 Schéma général du modèle de RI

Le modèle de RI proposé se place dans un cadre de description de l'information respectant les principes établis en théorie de transmission de l'information. De ce point de vue, des documents sont écrits par un auteur à destination de lecteurs pour transmettre un message. On distingue alors trois niveaux successifs de l'information : le niveau syntaxique du **signifiant** et les niveaux sémantiques du **signifié** et de la **pragmatique**.

Dans ce cadre général, nous nous intéressons à la description de l'information, matérialisée par un **document**. Cette description est décomposée en quatre couches : l'**atome** (atomes d'information), la **structure logique** (information structurée), le **cheminement** (chemins de lecture) et la **mise en contexte** (méta-information et information accessible).

L'atome d'information relatif à un **document atomique** est représenté de manière non décomposable et indépendante. Il s'agit de la première couche de notre modèle, avec l'ensemble $\mathcal{A}$ des documents atomiques a_i.

L'information structurée relative à un **document structuré** comporte une structure arborescente basée sur la relation de **composition** $\mathcal{R}_{comp}$ entre les documents. Il s'agit de la deuxième couche du modèle, avec l'ensemble $\mathcal{DS}$ des documents structurés ds_i. Les documents atomiques sont des cas particuliers de documents structurés : $\mathcal{A}$ est donc un sous-ensemble de $\mathcal{DS}$.

Le cheminement d'un document structuré est basé sur la relation de **cheminement** $\mathcal{R}_{chem}$ entre les nœuds du document. Cette troisième couche du modèle s'intéresse à l'information telle que le lecteur est susceptible de l'appréhender, en ajoutant la dimension de la lecture à la description hiérarchique de l'information. En effet, l'utilisation de technologies hypertextes permet la construction d'un ensemble de **chemins de lecture** pour la consultation d'un document structuré. Ainsi, le lecteur peut butiner les nœuds du document au gré des relations de cheminement suivies : on parlera alors de **cheminement déambulatoire**, par opposition au **cheminement linéaire** classique. La description de l'ensemble $\mathcal{CH}$ des chemins de lecture ch_i sur les documents structurés ds_i permet de construire l'ensemble $\mathcal{HD}$ des hyperdocuments hd_i : un hyperdocument est un document structuré parcouru par des chemins de lecture.

Mise en contexte : enfin, la quatrième couche du modèle aborde la problématique de la mise en contexte de l'information, basée sur la relation de **référence** $\mathcal{R}_{ref}$ entre les documents. Nous étudions la modélisation du **contexte textuel** du document, par opposition au contexte situationnel[1]. Le contexte textuel est composé de la **méta-information** (l'espace d'information dans lequel on peut trouver une référence vers le document) et de l'**information accessible** (l'espace d'information accessible par navigation à partir du document). On modélise les ensembles $\mathcal{AC}$, $\mathcal{DSC}$ et $\mathcal{CHC}$ des documents atomiques, des documents structurés et des chemins de lecture en contexte.

[1] Le contexte situationnel englobe tout ce qui a trait à la situation "physique" du document (environnement de travail, outils utilisés, etc.) ou au contexte personnel de l'utilisateur (aspects psycho-cognitifs, connaissances personnelles du thème, de l'auteur, etc.).

Le modèle de documents est schématisé dans la figure 6.1, qui récapitule les différentes couches : l'atome, l'information structurée, le cheminement et la mise en contexte.

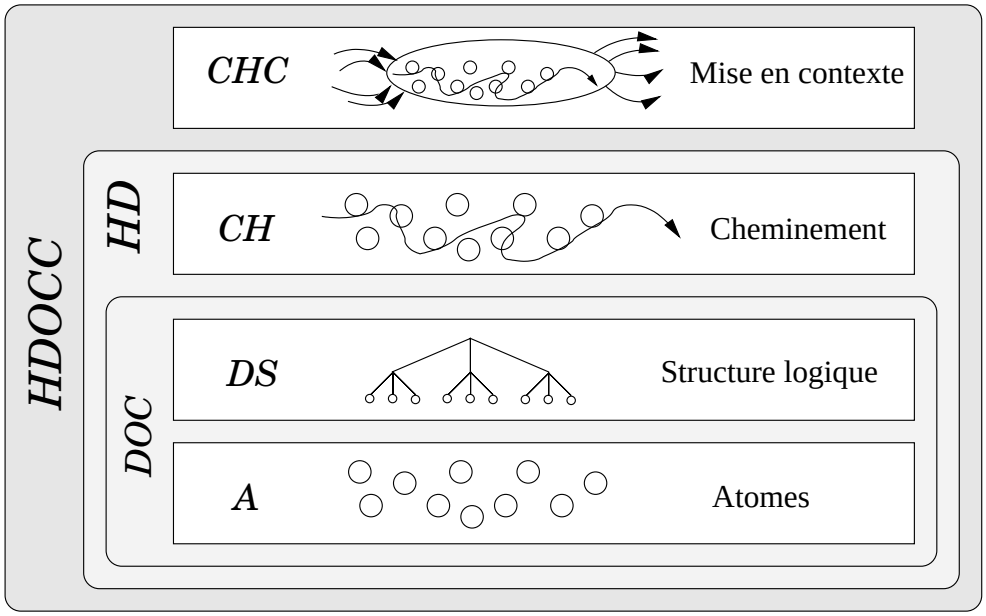

FIG. 6.1 – L'information structurée sur le Web.

Le modèle proposé pour décrire l'information structurée du Web est le modèle $\mathcal{HDOCC}$ d'hyperdocuments en contexte, dont les principaux ensembles sont récapitulés dans le tableau suivant :

	Couche	Type de document	Ensemble	Relation
$\mathcal{HDOCC}$	Atome	Document atomique	$\mathcal{A} \subseteq \mathcal{DS}$	$\mathcal{R}_{comp}$
	Structure logique	Document structuré	$\mathcal{DS}$	
	Cheminement	Hyperdocument	$\mathcal{HD}$	$\mathcal{R}_{chem}$
		Chemin de lecture	$\mathcal{CH}$	
	Contexte	Atome	$\mathcal{AC}$	$\mathcal{R}_{ref}$
		Document structuré	$\mathcal{DSC}$	
		Chemin de lecture	$\mathcal{CHC}$	

FIG. 6.2 – Les composants du modèle d'hyperdocuments. $\mathcal{HDOCC}$.

Le plan de ce chapitre suit la description des différentes couches :

Signifiant, signifié et pragmatique : l'information est modélisée à chacun de ces niveaux de description, qui sont introduits dans le cadre de théories sur la "transmission de l'information" dans la section 6.3, et synthétisés dans la section 6.4 avec un **modèle de transmission de l'information** sur lequel se base notre modèle de RI.

Atomes : nous présentons la brique de base, le document atomique, dans la section 6.6.

Relations : le modèle distingue plusieurs types de relations : composition, cheminement et contexte, que nous présentons dans la section 6.7.

Composition et documents structurés : la composition d'atomes permet de créer des informations de granularité plus élevée possédant une structure hiérarchique. Il s'agit des **documents structurés**, que nous présentons dans la section 6.8.

Cheminement et hyperdocuments : la relation de cheminement permet de décrire des **chemins de lecture** et des **hyperdocuments**, que nous présentons dans la section 6.9.

Référence et mise en contexte : la relation de référence permet de définir le **contexte** des documents, composé de la **méta-information** et de l'**information accessible**, que nous présentons dans la section 6.10.

Impact des relations : enfin, nous terminons par une discussion à propos de l'impact des relations sur l'indexation et la pertinence dans les sections 6.12 et 6.13.

Le modèle de RI que nous décrivons dans ce chapitre est formalisé dans les chapitres 7 (modèle d'hyperdocument), 8 (phases d'indexation et d'interrogation).

6.3 Transmission de l'information

Pour aborder la problématique de la RI structurée sur le Web, nous nous intéressons en premier lieu au concept d'**information**, et plus particulièrement d'**information structurée**. Le concept d'information peut prendre différentes significations selon le contexte dans lequel il est employé, et il est encore plus délicat à définir que le concept de document. Généralement, les définitions proposées restent très vagues. Par exemple, on peut citer la définition suivante : "*Information : élément de connaissance concernant un phénomène et qui, pris dans un contexte déterminé, a une signification particulière*" [Gdt]. L'élaboration d'une définition universelle de cette notion sort du cadre de cette thèse.

6.3.1 Signifiant, signifié et pragmatique

Sur le Web, un SRI doit retrouver une information structurée produite par un auteur à destination de lecteurs. La "théorie de l'information" et ses développements nous proposent différents points de vue sur l'information et sa transmission, en faisant la distinction fondamentale entre signifiant, signifié et pragmatique.

Le dictionnaire Larousse définit le signifiant comme la "*forme concrète (image acoustique, symbole graphique) du signe linguistique, par opposition à signifié*", et le signifié comme le "*contenu sémantique du signe linguistique, concept, par opposition à signifiant*" [Larousse]. Le signifiant d'une information est son "encodage" (que ce soit sur un support papier, en HTML, sous la forme d'un son, etc.). Il est alors porteur d'une sémantique, appelée le signifié : il s'agit d'information concrète, décrite indépendamment de tout contexte.

Mais ces deux aspects de l'information sont indissociables de la pragmatique, qui est le signifié pris dans un contexte. La pragmatique est "*l'ensemble des relations entre les caractères ou groupes de caractères et la signification qui leur est attribuée dans le contexte où ils sont employés*" [Gdt]. Le signifié ne prend tout son sens que s'il est replacé dans un contexte. Par exemple, la phrase "I have a dream" a une signification, mais qui ne peut être entièrement déterminée que si elle est replacée dans un contexte, comme par exemple "un discours de Martin Luther King" ou "une déclaration de Mr Smith au saut du lit".

Le rôle de la tâche d'indexation d'un SRI est d'extraire une représentation du contenu sémantique (signifié et pragmatique) des documents (signifiant) pour pouvoir les retrouver. Un système se basant uniquement sur l'aspect signifiant des documents, comme par exemple un moteur du Web basé sur des mots-clés, rencontre des difficultés face à l'ambiguïté de ce formalisme.

Un modèle de RI doit donc considérer les trois aspects de l'information. Notre travail s'intéresse plus particulièrement à l'utilisation du contexte comme une méta-information qui permet de désambiguïser les documents en apportant une information supplémentaire. Afin de mieux comprendre cette problématique, nous présentons des travaux modélisant l'aspect signifiant [Shannon et al.49] [Shannon et al.75], l'aspect signifié [BH64] et l'aspect pragmatique [Barwise89] dans le cadre de la théorie de la communication. Enfin, nous présentons les travaux de Jakobson [Jakobson63] et Kerbrat-Orecchioni [KO80] qui mettent l'accent sur la notion de contexte dans le cadre d'une communication humaine.

6.3.2 Le signifiant et la transmission de l'information

Les travaux de Shannon sur la transmission de l'information avec sa *théorie mathématique de la communication* [Shannon et al.49] [Shannon et al.75] traitent de l'aspect signifiant de l'information. L'information est considérée comme une donnée quantifiable, du point de vue de la transmission de l'émetteur au récepteur sans tenir compte de la sémantique du message. Cette "quantité" d'information H est calculée à partir de p_i, qui est la probabilité de sélection d'un message i parmi tous les messages possibles : $H = -\sum p_i.log_2(p_i)$.

En conséquence, une phrase parfaitement bien formée sur le plan grammatical mais inacceptable sur le plan sémantique pourra être considérée comme porteuse d'une grande quantité d'information. Un exemple célèbre est la phrase de Noam Chomsky :

« *Colorless green ideas sleep furiously* »
(« *D'incolores idées vertes dorment furieusement* »)

Ce message n'a aucun sens, mais est pourtant porteur d'une grande quantité d'information dans le modèle de Shannon, car sa probabilité d'apparition est faible. Cet exemple montre qu'il est insuffisant de considérer uniquement l'aspect "signifiant" de l'information pour la RI.

6.3.3 Le signifié et l'information sémantique

Les travaux de Carnap et Bar-Hillel sur *l'information sémantique* [BH52] [BH64] critiquent les applications de la théorie statistique de la communication. Carnap et Bar-Hillel développent une "théorie de l'information sémantique" basée sur la logique des propositions, et traitent de l'aspect signifié de l'information, indépendamment de toute transmission. L'information sémantique existe en tant que telle, se suffit à elle-même et peut être décrite. Le modèle permet de calculer une "mesure de contenu" $cont(p) \in [0, 1]$ ("content-measure") d'une proposition p ("statement") en fonction des disjonctions logiquement déduites de p, obtenues à partir des propositions atomiques "logiquement vraies" si la proposition p est vérifiée.

Bien que Carnap et Bar-Hillel s'emploient à considérer l'aspect "signifié" de l'information, la mesure du contenu est analogue à la notion de quantité d'information de Shannon. L'information sémantique n'est pas associée à une valeur de vérité : une information fausse pourra aussi être considérée comme exprimant une grande quantité d'information. L'aspect pragmatique de l'information n'est pas considéré dans ce modèle : l'information y est considérée indépendamment de tout contexte. Par exemple, "I have a dream" sera considéré comme exprimant la même information, que cette phrase soit prononcée par Mr Smith ou par Martin Luther King.

6.3.4 La pragmatique et la théorie des situations

Les travaux de Barwise sur *la théorie des situations* [Barwise89] se situent au niveau de la pragmatique, et considèrent l'information *en contexte*. Barwise franchit encore une étape dans la modélisation de la communication, par rapport à Shannon et Bar-Hillel. Il considère que la signification d'un texte n'est pas complètement déterminée par l'énoncé (le texte seul) comme c'est le cas pour la théorie de l'information sémantique, mais qu'elle est fondamentalement dépendante du contexte, et en premier lieu de l'auteur et du lecteur. Ainsi, il tente de séparer la signification d'un énoncé de sa signification dans un contexte, qu'il appelle le *contenu*. Ensuite, ce contenu est susceptible de prendre différentes significations, selon le contexte dans lequel il est placé : l'esprit de l'auteur, les connaissances partagées par l'auteur et le lecteur, l'esprit du lecteur, etc.

Barwise propose l'unité de base de la théorie des situations : l'*infon*, qui représente une information indépendamment de tout contexte. Il existe une relation de composition sur les infons, qui permet de fusionner des infons, sous réserve que les informations soient compatibles entre elles. Il existe aussi une relation d'ordre sur les infons, qui exprime la déduction logique entre les infons. Puis Barwise définit des *situations*, qui permettent de mettre un infon en contexte (en situation). On dit que la situation s *supporte* l'infon i.

La théorie des situations montre l'intérêt de considérer la situation (le contexte) d'une information, et permet de représenter une information qui peut être vraie dans une situation mais fausse dans une autre. On remarquera que Barwise n'attache pas de valeur de vérité à un infon donné. Par contre, l'énoncé "*D'incolores idées vertes dorment furieusement*", qui

peut être représenté par un infon, ne trouvera probablement aucune situation (contexte) dans laquelle il soit vrai.

6.3.5 Le schéma de la communication humaine

Dans l'optique de considérer l'information au cours d'une communication humaine, l'approche linguiste de Jakobson [Jakobson63], reformulée par Kerbrat-Orecchioni [KO80] propose un "schéma de la communication humaine" qui fait intervenir le contexte de la communication : « *L'émetteur envoie un message au destinataire. Pour être opérant, le message requiert d'abord un contexte auquel il renvoie, contexte saisissable par le destinataire, et qui est, soit verbal, soit susceptible d'être verbalisé* » [Jakobson63].

Le schéma "descriptif" comprend les éléments suivants : un émetteur, un récepteur (destinataire), un contexte, un contact entre eux (canal), un code commun, un enfin un message. Kerbrat-Orecchioni reformule ce modèle en ajoutant une notion d'*univers du discours* : les conditions concrètes de la communication, des contraintes sur le thème du discours, de la nature particulière de l'émetteur et du destinataire, etc. Le schéma de la communication reformulé par Kerbrat-Orecchioni est décrit dans la figue 6.3.

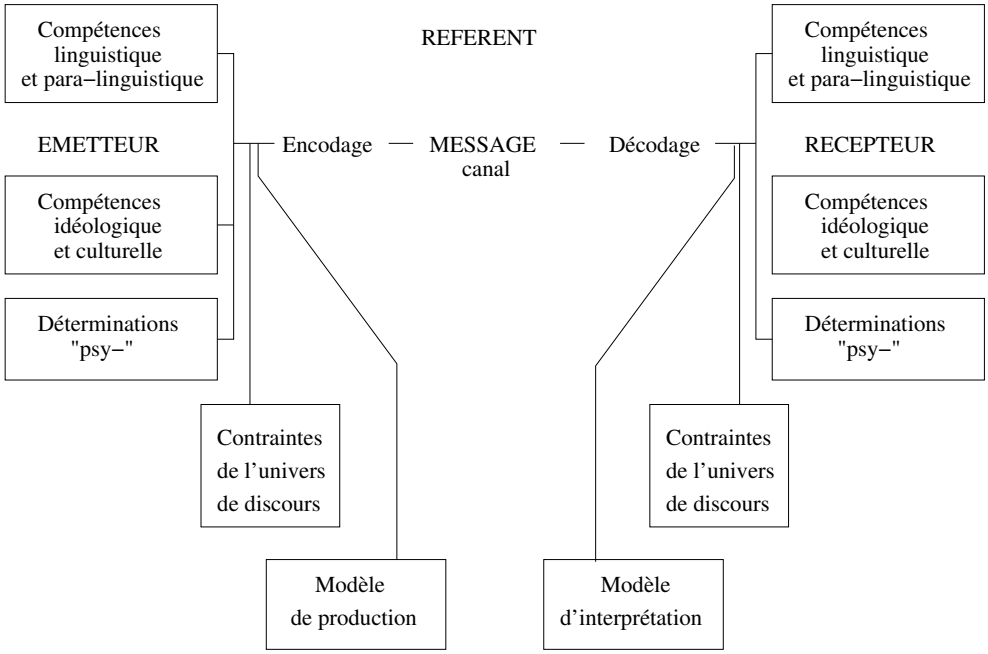

FIG. 6.3 – Schéma de la communication humaine de Kerbrat-Orecchioni.

Dans un contexte de communication humaine, on retrouve sous la dénomination d'*univers du discours* la notion de *situation* formalisée par Barwise. Les travaux s'attachant à la formalisation de l'information rejoignent donc les travaux modélisant la communication humaine d'un point de vue linguistique et psycho-cognitif, et s'accordent à penser qu'une information ne prend pas tout son sens tant qu'elle n'est pas replacée dans un contexte.

6.4 Un modèle de transmission de l'information

Le schéma de Jakobson et celui de Kerbrat-Orecchioni mettent en avant l'importance du contexte d'une communication : l'univers du discours, les connaissances partagées ou non, le code employé, etc. Pour la problématique de la RI, nous pensons que le contexte est au moins aussi important que dans le cas d'une communication humaine et doit donc être pris en compte dans le modèle de RI. Le Web est un moyen d'échange d'informations, et la production de documents Web est un mode de communication écrite basé sur les mêmes principes que la communication parlée. Nous retrouvons donc les principes de la communication humaine dans la publication de pages Web et leur consultation.

6.4.1 L'information : quatre types et deux niveaux de description

Nous présentons dans cette section notre modèle de transmission de l'information adapté à la communication via le média "Web". Ce modèle se base sur la distinction fondamentale de quatre types d'information à deux niveaux de description. Ces niveaux sont décrits dans la section 6.4.3. A chaque niveau, nous retrouvons les quatre types de document déjà présentés dans la figure 6.2 : le document atomique, le document structuré, l'hyperdocument (chemin de lecture) et l'hyperdocument en contexte.

Signifiant	Signifié/pragmatique
Document atomique	Information atomique
Document structuré	Information structurée
Hyperdocument	Hyperinformation
Hyperdocument en contexte	Hyperinformation en contexte

FIG. 6.4 – Les niveaux de description des documents et de l'information.

6.4.2 Schéma général de transmission de l'information

Le processus de communication via le média "Web" est la transmission d'une **information** en provenance d'un **émetteur** ("auteur", source d'information), sous la forme d'un **document**, vers un **récepteur** ("lecteur", destinataire). Nous le résumons en 5 phases :

Extraction : l'auteur extrait l'**information** de son **contexte**, passant du niveau de la "pragmatique" au niveau du "signifié".

Ecriture et encodage : l'**information** est encodée en un **document**, passant du niveau du signifié au niveau du signifiant.

Transmission et décodage : le **document** est transmis par le biais d'un canal, puis il est décodé et **présenté** au lecteur. Cette phase se déroule au niveau du signifiant.

Lecture : le lecteur acquiert l'**information** en "décodant" le document qui lui est **présenté**, qui revient donc au niveau du signifié.

Interprétation : et enfin le lecteur replace cette **information** dans son propre **contexte**, passant du niveau du signifié à celui de la pragmatique.

6.4.3 Signifiant et signifié

L'information au niveau du signifiant est représentée par un *message*, défini par le GDT comme la « *communication d'une information ou d'un renseignement, d'une source vers une ou plusieurs destinations, dans une langue ou dans un code approprié* » [Gdt].

On parlera de **document atomique** exprimant un contenu atomique, dans le cas d'un élément atomique non structuré. La notion de message sur le Web est toutefois plus complexe : en effet, le code utilisé (par exemple HTML) permet une description structurée qui nous amène à définir la notion plus générique de **document** dans la section 6.2.

L'information au niveau du signifié (l'information sémantique de Bar-Hillel, cf. [BH64]) est représentée par un *texte*[2]. Or, un principe essentiel de la textualité est la cohérence, c'est-à-dire la « *continuité sémantique que le texte constitue en vertu de son organisation propre* » [Sarfati97]. Un texte a donc une certaine "texture", c'est-à-dire une « *organisation formelle du texte dans la mesure où cette organisation assure sa continuité sémantique* » [Sarfati97], qui lui donne une "cohésion" sémantique. L'organisation d'un "texte" (une information) décrit une structure, nommée structure du discours, qui est souvent dépendante de la structure logique. On parle alors d'**information structurée**, et le pendant de l'hyperdocument au niveau du signifié est une **hyperinformation**.

6.4.4 Pragmatique : information et contexte

L'information au niveau de la pragmatique est représentée par une information interprétée dans un contexte : « *une information est une donnée qui a été interprétée (ou réinterprétée). Le cadre de référence qui détermine cette interprétation est constitué de la somme des connaissances et des expériences de la personne qui effectue l'interprétation* » [Gdt].

Le dictionnaire Larousse nous donne trois définitions possibles du contexte :

Contexte [Larousse] : n.m. (du lat. contexere, tisser ensemble).

1. Texte à l'intérieur duquel se situe un élément linguistique (phonème, mot, phrase, etc.) et dont il tire sa signification ou sa valeur.
2. Circonstances, situation globale où se situe un événement : replacer un fait dans son contexte historique.
3. Conditions d'élocution d'un discours, oral ou écrit.

[2]Texte : la définition de Sarfati se situe au niveau du signifié : une « *unité de base de la signification dans le langage* » [Sarfati97].

On distingue le contexte textuel (cotexte, intertexte, paratexte) du contexte situationnel (l'univers du discours de Kerbrat-Orecchioni). Il y a eu une évolution de la notion de contexte, qui désignait initialement « *l'ensemble d'un texte précédant ou suivant un mot, une phrase, un passage* ». Cette notion a ensuite été utilisée pour désigner tout ce qui, de manière générale, peut donner une autre interprétation à un texte/une idée. Par exemple, les mots ou les phrases qui précèdent ou qui suivent font partie du contexte, mais aussi les chapitres et les nœuds d'un hypertexte. Tous les documents référencés implicitement ou explicitement, ou encore les connaissances de l'auteur et du lecteur, font aussi partie du contexte.

Nous distinguerons donc le contexte textuel du contexte situationnel. Le contexte textuel englobe tout ce qui est matérialisé dans les documents, alors que le contexte situationnel regroupe les aspects "physiques" du contexte, c'est-à-dire la situation physique du lecteur ou de l'auteur au moment où la communication s'établit, ainsi que tout ce qui concerne les connaissances de l'auteur, du lecteur, etc. Le contexte situationnel est important pour la RI, mais nous nous intéressons uniquement à l'information qu'il est possible d'extraire des documents, à l'exclusion de toute autre information externe. Dans notre modèle, nous avons donc limité nos travaux au **contexte textuel**, que nous définissons de la manière suivante :

Définition 2 ***Contexte textuel*** *: il se définit comme le contexte au niveau du signifiant, c'est-à-dire l'ensemble des éléments extérieurs à un document, matérialisés dans les documents, et qui sont susceptibles de modifier la perception et/ou l'interprétation du document.*

Tous les types de documents (atome, document structuré, hypertexte, chemin de lecture) peuvent être placés en contexte. Par exemple, un atome peut être mis dans le contexte d'autres atomes au sein d'un même document. Ainsi, on distingue trois types de contexte textuel :

Cotexte textuel : le cotexte est une restriction du contexte aux éléments d'information à l'intérieur d'un même document. Il s'agit donc des documents qui suivent ou précèdent un atome ou un document structuré dans la structure logique d'un autre document.

Contexte hypertextuel : le contexte hypertextuel est la transposition du cotexte textuel au cas des hypertextes, c'est-à-dire en considérant les relations de cheminement. Il s'agit donc des documents qui suivent ou précèdent un document dans la structure de cheminement d'un autre document.

Contexte référentiel : le contexte référentiel est le contexte d'un document au niveau de la relation de référence, c'est-à-dire l'ensemble des documents qui référencent ou qui sont référencés par un document.

Le contexte est un élément essentiel de la description de l'information. Il est indispensable de le considérer pour la RI, aussi bien dans la représentation de l'information (avec le modèle de documents) que dans l'extraction du contenu sémantique des documents (la phase d'indexation).

6.4.5 Phase d'extraction : contexte et information

L'émetteur (l'auteur) se fait une idée de l'**information** (*signifié*) qu'il désire transmettre, fortement liée au contexte. La première étape du processus de transmission consiste à extraire

une information (niveau pragmatique) de son contexte, pour produire une information qui se suffit à elle-même (niveau signifié), comme présenté dans le tableau 6.5.

Pragmatique	Signifié
Atome d'information en contexte	Atome d'information
Information structurée en contexte	Information structurée
Hyperinformation en contexte	Hyperinformation

FIG. 6.5 – Phase d'extraction : de la pragmatique au signifié.

L'auteur essaie de faire en sorte que le lecteur, qui n'a pas le même contexte situationnel que lui (connaissances et expériences), puisse interpréter et comprendre l'information transmise. Il est donc nécessaire de transmettre ce contexte. Cela peut se faire de plusieurs manières, comme présenté dans la figure 6.6.

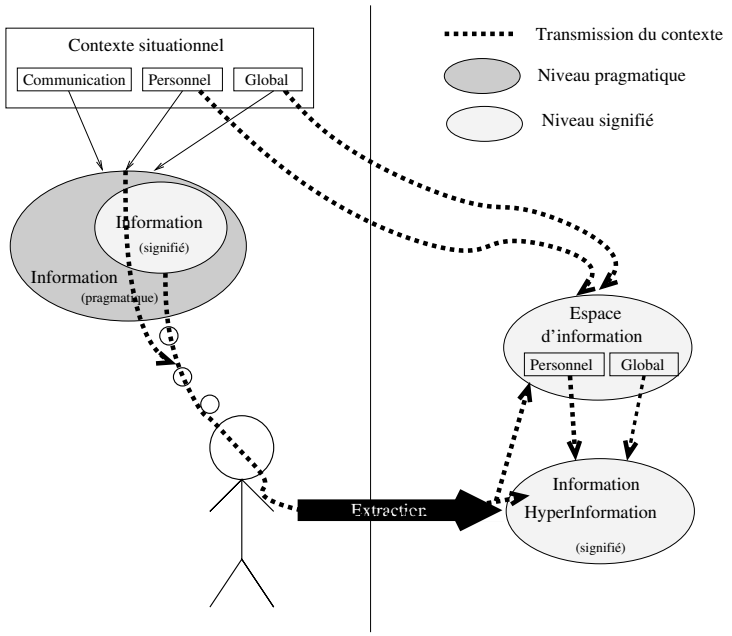

FIG. 6.6 – Phase d'extraction : de la pragmatique au signifié.

La description de références faisant appel au contexte global ne pose pas de problème si l'auteur a une perception correcte des connaissances partagées. Ainsi, un historien pourra omettre de rappeler certains faits historiques auxquels il fait référence dans son discours, ce qui nécessitera de disposer d'un contexte personnel adéquat pour comprendre le message. Par contre, s'il s'agit d'une information portant sur son contexte personnel, l'auteur devra explicitement intégrer cette information à son document, alors même qu'elle ne fait pas partie de l'information initiale à transmettre. Il pourra l'inclure directement dans l'information

(ou l'hyperinformation) transmise, ou par le biais une référence (implicite ou explicite) vers une information externe. Son contexte personnel peut aussi être décrit par une autre information qui référence l'information extraite. Enfin, d'autres composantes du contexte, comme le contexte de communication, ne sont pas transmis.

6.4.6 Phase d'encodage, de décodage et de lecture

La phase d'encodage permet de passer du signifié (l'information) au signifiant (le document), comme présenté dans le tableau 6.7. Il s'agit de l'utilisation des possibilités de description mises à la disposition de l'auteur pour représenter ses idées, comme par exemple le langage HTML, XML ou RTF. Dès l'instant où l'information est encodée, on peut alors la manipuler : il s'agit d'un **document**. L'utilisation de liens hypertextes dans un document permet de résumer une partie de l'information à transmettre sous la forme d'une référence vers un autre document explicitant la partie référençante. Mais une telle référence permet aussi d'inclure une partie du contexte personnel de l'auteur (ou du contexte qui n'appartient pas au contexte personnel du lecteur). Le résultat de l'encodage est que le document produit possède son contexte propre, constitué de son contexte textuel et hypertextuel, ainsi que des références implicites.

Signifié	Signifiant
Atome d'information	Document atomique
Information structurée	Document structuré
Hyperinformation	Hyperdocument

FIG. 6.7 – Phase d'encodage : du signifié au signifiant.

Puis, il y a transmission, décodage et présentation du **document**, généralement en utilisant le code utilisé pour l'encodage. La présentation doit permettre au lecteur d'accéder au contexte explicité dans le document. Les informations sont présentées en fonction des moyens disponibles pour la visualisation, mais aussi en fonction des choix de présentation de l'auteur.

Le lecteur prend connaissance du document, avec les contraintes de la présentation, et se fait une certaine idée de l'information (niveau signifié) que l'auteur a voulu transmettre. Cette opération est l'opération inverse de l'encodage présenté dans le tableau 6.7. Le lecteur essaie de reconstituer l'information représentée par le document, c'est-à-dire l'information sémantique (le texte) de ce document en tant qu'énoncé indépendant de tout contexte.

6.4.7 Phase d'interprétation : information et contexte

La dernière étape consiste à replacer l'information qui se suffit à elle-même (niveau signifié) dans un nouveau contexte (niveau pragmatique). Le lecteur l'**interprète** à la lumière de son propre contexte situationnel : c'est l'opération inverse de l'extraction (cf. tableau

6.5). Il recrée donc les références faisant appel au contexte global, et construit aussi sa vision du contexte situationnel de l'auteur à partir de l'information, de l'hyperinformation, mais aussi du contexte global et de son propre contexte situationnel. Il utilise aussi les références (implicites ou explicites) vers des informations externes (l'information accessible).

6.4.8 Synthèse

Les 5 phases de la transmission (extraction, encodage, décodage, lecture, interprétation) sont des transitions, récapitulées dans le tableau 6.8, entre 6 niveaux de description : l'information en contexte (signifié), l'information (signifié), le document (signifiant), la présentation (signifiant), l'information (signifié) et l'information en contexte (signifié).

Etape de transmission	Représentation de l'information	Niveau de description
Auteur en contexte	Information en contexte	Pragmatique (sémantique)
Auteur	Information	Signifié (sémantique)
Encodée	Document	Signifiant (syntaxique)
Décodée	Présentation	Signifiant (syntaxique)
Lecteur	Information	Signifié (sémantique)
Lecteur en contexte	Information en contexte	Pragmatique (sémantique)

FIG. 6.8 – Les étapes de la transmission d'information.

6.5 Le modèle de documents $\mathcal{HDOCC}$

Notre modèle s'articule autour des deux niveaux de l'information : syntaxique (signifiant) et sémantique (signifié et pragmatique). Le schéma de l'évolution de l'information de ce point de vue est représenté par la figure suivante :

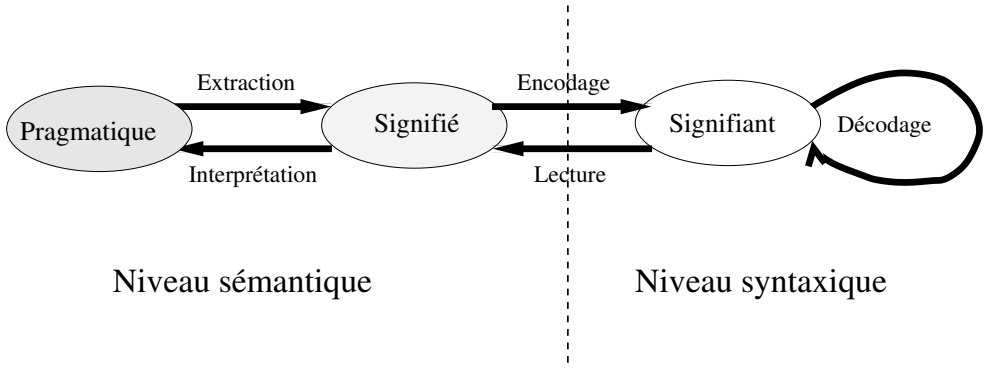

FIG. 6.9 – L'information et le signifiant/signifié/pragmatique.

Au cours de la transmission, les quatre types de document (atome, document structuré, hyperdocument et hyperdocument en contexte) sont décrits au niveau du signifiant ou du signifié. En conséquence, les composants du modèle $\mathcal{HDOCC}$ présentés dans le tableau 6.10 se situent au niveau du signifiant ($\mathcal{HDOCC}_{doc}$) et au niveau du signifié ($\mathcal{HDOCC}_{inf}$)[3].

	Signifiant ($\mathcal{HDOCC}_{doc}$)		Signifié ($\mathcal{HDOCC}_{inf}$)	
Type de documents	Document atomique	$\mathcal{A}_{doc}$	Information atomique	$\mathcal{A}_{inf}$
	Document structuré	$\mathcal{DS}_{doc}$	Information structurée	$\mathcal{DS}_{inf}$
	Hyperdocument	$\mathcal{HD}_{doc}$	Hyperinformation	$\mathcal{HD}_{inf}$
	Chemin de lecture	$\mathcal{CH}_{doc}$	Cheminement	$\mathcal{CH}_{inf}$
	Atome	$\mathcal{AC}_{doc}$	Atomes	$\mathcal{AC}_{inf}$
	Document structuré	$\mathcal{DSC}_{doc}$	Information structurée	$\mathcal{DSC}_{inf}$
	Chemin de lecture	$\mathcal{CHC}_{doc}$	Cheminement	$\mathcal{CHC}_{inf}$

FIG. 6.10 – Les composants du modèle $\mathcal{HDOCC}$: niveau du signifiant $\mathcal{HDOCC}_{doc}$ et niveau du signifié $\mathcal{HDOCC}_{inf}$.

Les modèles classiques de RI se fondent généralement sur l'hypothèse simplificatrice de la bijection entre le niveau du signifiant $\mathcal{HDOC}_{doc}$ et le niveau du signifié $\mathcal{HDOC}_{inf}$. Par exemple, à chaque document structuré de $\mathcal{DS}_{doc}$ correspond une information et une seule de $\mathcal{DS}_{inf}$. L'aspect de la pragmatique prend alors toute sa dimension : en effet, cette hypothèse n'est pas valide en raison des multiples interprétations que l'on peut faire d'une même information de $\mathcal{HDOC}_{inf}$ en fonction de son contexte. Pour la RI, l'intérêt de la pragmatique réside dans la possibilité de désambiguïsation que permet le contexte d'un document, comme le montre la figure suivante :

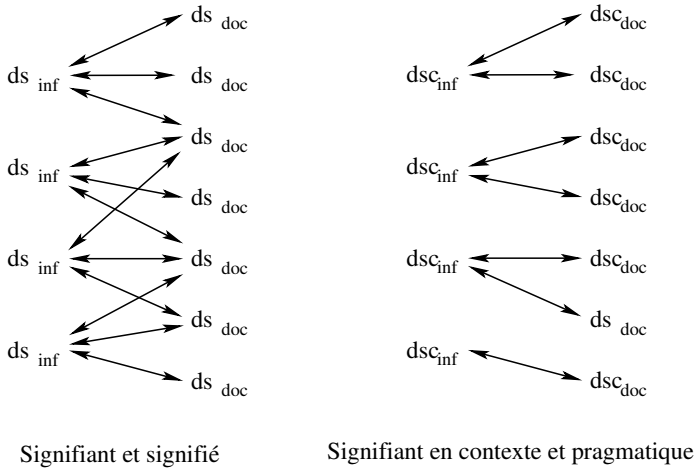

FIG. 6.11 – L'information et le signifiant/signifié/pragmatique.

[3]Le tableau 6.10 complète le tableau 6.2.

On remarque que le contexte permet de déterminer quelle est l'unique information dsc_{inf} associée à un document en contexte dsc_{doc} donné, mais qu'en revanche plusieurs documents peuvent représenter la même information. Cependant, malgré l'absence de bijection, les éléments de $\mathcal{HDOCC}_{doc}$ sont décrits de la même manière que ceux de $\mathcal{HDOCC}_{inf}$. En conséquence, les éléments de $\mathcal{HDOCC}_{inf}$ ne sont pas utilisés par la suite, et nous simplifions la description du modèle en restant au niveau du signifiant. Dans les sections suivantes, nous présentons donc le modèle d'hyperdocuments en contexte $\mathcal{HDOCC}_{doc}$: les documents atomiques, les relations, les documents structurés, les hyperdocuments, et le contexte de chaque type de documents. Nous simplifierons l'écriture dans le chapitre 7, en formalisant le modèle $\mathcal{HDOCC}_{doc}$ avec la notation simplifiée $\mathcal{HDOCC}$.

6.6 Les documents atomiques $\mathcal{A}_{doc}$

Le modèle d'hyperdocuments en contexte est fondé sur la représentation des atomes, qui sont les briques de base des documents et des informations. La phase d'indexation est fondée sur l'indexation des atomes.

Définition 3 ***Document atomique :*** *un document $a \in \mathcal{A}_{doc}$ est un fragment de texte, un ensemble insécable de phrases délimité dans le document (par exemple par un retour à la ligne ou une balise HTML). Un document atomique est porteur d'une signification qui se suffit à elle-même, indépendamment de tout contexte.*

Dans le modèle de transmission de l'information, un document atomique représente une information atomique exprimée par l'auteur, qui est encodée en un paragraphe. Ensuite, le paragraphe est présenté au lecteur, qui le lit et l'interprète pour construire sa vision de l'information atomique.

6.7 Les liens et les relations

Toute la force du Web réside dans son accessibilité, dans la facilité à définir des liens entre les documents, et à tisser ainsi une toile mondiale sur laquelle s'enchevêtrent des informations sur tous les sujets imaginables. Les différentes études de la connectivité du réseau de liens du Web montrent l'utilisation massive qui en est faite. Ainsi, Woodruff estimait en 1996 à 13,9 la moyenne du nombre de liens par page [Woodruff et al.96]. Ce chiffre atteignait 16,15 dans une étude de Beckett [Beckett97], et même 28,6 dans une étude plus récente de Murray [Murray et al.00]. Les statistiques que nous avons récemment extraites montrent une moyenne qui se situe entre 11,45 et 38 selon les collections [Gery02]. Cette connectivité très importante du Web joue un rôle central dans la description de l'information.

L'objectif d'un modèle de RI structurée pour le Web est de prendre en compte la structure de l'information. Cela nécessite de s'interroger sur la sémantique de la structure et donc des relations. En d'autres termes, comment l'auteur d'un site Web utilise-t-il ces relations, c'est-à-dire les liens qui les matérialisent, pour décrire l'information qu'il veut communiquer ?

6.7.1 Définitions

Pour décrire les liens du Web, nous reprenons les définitions de l'état de l'art sur les hypertextes (cf. section 2.4.1). Les pages Web sont des **nœuds**, connectés par des **liens** hypertextes qui sont ancrés dans la page source et parfois dans la page destination. Les **ancres** se trouvent sous une forme textuelle (un mot ou une phrase) ou sous la forme d'une image cliquable. Elles permettent d'activer le lien dans un butineur et de se retrouver "transporté" sur la page référencée par une action de navigation.

Un lien hypertexte entre deux pages Web matérialise une relation entre deux documents du Web. Il existe aussi des relations qui ne sont pas matérialisées par des liens, comme par exemple des relations rendues implicites par la structure d'un site.

Une relation dans notre modèle d'hyperdocuments est définie comme suit :

Définition 4 *La relation $\mathcal{R}$ est une relation binaire entre deux documents des ensembles $\mathcal{A}_{doc}$, $\mathcal{DS}_{doc}$, $\mathcal{HD}_{doc}$ et $\mathcal{CH}_{doc}$.*

Nous avons vu dans le chapitre 2.4 différents travaux utilisant ces liens. Notre principale critique portait sur le "sac de liens" : dans le meilleur des cas, les liens internes à un site Web sont différenciés des liens référençant un document extérieur au site. Il s'agit d'une simple analyse de la connectivité du réseau de liens, sans essayer de comprendre le rôle qu'ils jouent effectivement dans la description de l'information sur le Web, et dans la propagation de l'information dans les index. Il est au contraire nécessaire d'analyser le rôle des liens (et donc des relations) dans la description de l'information, ce qui nous amène au problème sous-jacent du typage des liens.

6.7.2 Rôle des liens dans la description de l'information

Nous avons vu que, dans la littérature, différentes intentions sont prêtées à l'auteur lors de la création des liens (cf. chapitre 2). Les liens internes à un site, quand ils ne sont pas purement et simplement occultés, sont généralement considérés comme des liens représentant une structure hiérarchique des documents (relation de composition), l'organisation interne d'un site (liens "Retour", "Table des matières", etc.) ou encore des liens proposant un sens de lecture (liens "Page suivante", relations de séquence). Ce dernier type de lien est adapté aux documents structurés, pour lesquels on fait l'hypothèse qu'un sens de lecture est proposé, favorisant (et parfois même imposant) une lecture linéaire des documents. Cette restriction ne convient pas au contexte des hypertextes (nous en discutons dans la section 6.9), et c'est pourquoi nous proposons à la place la notion de relation de cheminement.

Nous considérons les sites Web à la fois du point de vue des documents structurés et du point de vue des hypertextes. Un document structuré possède une structure hiérarchique (arborescente) basée sur la relation de composition. Un hypertexte possède une structure de graphe, basée sur les relations de cheminement (référence interne au site : l'auteur propose au lecteur de poursuivre sa lecture dans un autre nœud du graphe) et de référence (référence externe au site : l'auteur propose au lecteur d'aller consulter d'autres sites).

En ce qui concerne les liens référençant une page externe au site, nous avons présenté plusieurs travaux sur l'analyse de la connectivité du réseau de liens et l'extraction d'information à partir de la structure hypertexte du Web dans la section 4.3, avec les notions de *page centrale* et de *page de référence* ("*hubs*" et "*authorities*"), de *réputation*, de *popularité*, etc. Les hypothèses communément avancées pour justifier la création de ces liens sont nombreuses. Il est généralement considéré que la création d'un lien montre un intérêt pour la page référencée de la part de l'auteur. Les relations sous-jacentes à ces liens entrent dans la catégorie "relation de référence".

Cette vision du Web accorde une grande importance au typage des liens. En effet, nous faisons l'hypothèse forte que le typage conditionne le modèle de documents. Ainsi, le choix de l'auteur dans l'utilisation de relations de composition, de cheminement ou de référence, permet de définir les hyperdocuments.

6.7.3 Typologie des relations

Le modèle d'hyperdocuments représente les relations $\mathcal{R}$, qui ne sont pas toujours matérialisées par des liens. Nous en distinguons trois types, qui jouent un rôle pour la description de l'information aux niveaux du signifiant, du signifié et de la pragmatique : les relations de composition $\mathcal{R}_{comp}$, les relations de cheminement $\mathcal{R}_{chem}$ et les relations de référence $\mathcal{R}_{ref}$, comme présenté dans la figure suivante :

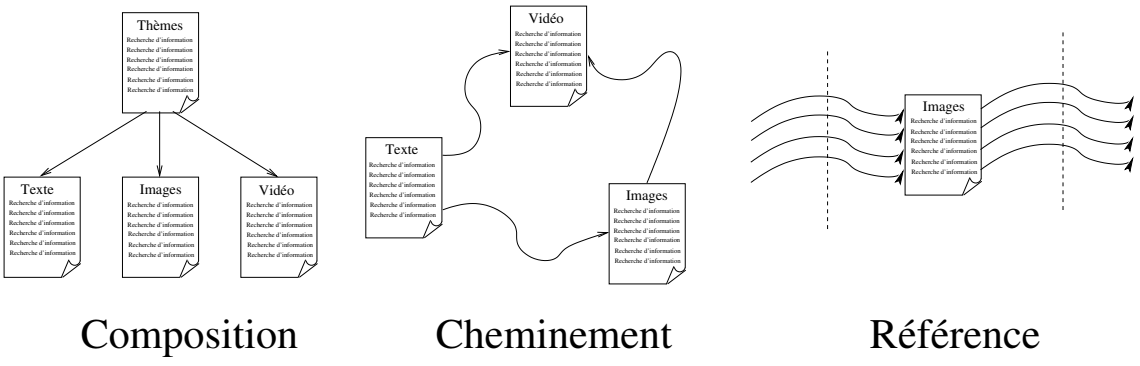

FIG. 6.12 – Les trois types de relations.

Les relations de composition décrivent l'organisation structurelle des documents (la structure logique). La relation de composition implique l'existence d'un sens de lecture (l'introduction est à lire avant le premier chapitre) imposé par l'auteur, que le lecteur devra suivre en l'absence d'autres solutions pour consulter le document.

Les relations de cheminement décrivent une structure délinéarisée du sens dans un hypertexte et définissent une lecture non linéaire. Ces relations sont représentées par des liens internes à un site : l'auteur propose au lecteur de poursuivre sa lecture dans un autre nœud du graphe du site, définissant ainsi différentes possibilités de parcours.

Nous parlerons alors de **relations de cheminement déambulatoires** (non linéaires) par opposition aux **relations de cheminement linéaires** des documents structurés.

Les relations de référence décrivent une **mise en contexte** d'un document et sont matérialisées par des liens hypertextes externes au document : l'auteur propose au lecteur d'aller consulter d'autres sites. Le contexte est composé de l'**information accessible** à partir du document (les pages référencées choisies par l'auteur) et de la **méta-information** (les pages référençant le document, indépendamment de la volonté de l'auteur).

6.7.4 Visibilité des relations

La **visibilité** est une propriété exprimant le caractère implicite ou explicite d'une relation vis-à-vis de l'homme et de la machine. Une relation peut être **implicite**, **explicite** ou **activable** :

Relation implicite : une relation implicite nécessite une interprétation humaine pour déterminer quelles sont les entités en relation. Une telle relation n'est pas décrite au niveau syntaxique, elle n'est donc pas utilisable directement par le système.

Relation explicite : une relation explicite est une relation qui est représentée à l'aide du langage utilisé pour la description des documents, et qui est donc utilisable par le système.

Relation activable : une relation activable est une relation explicite dont le lien permet de naviguer dans l'espace d'information.

Par exemple, beaucoup de relations de composition ne sont pas explicites dans les documents, mais sont rendues implicites par la structure logique du document. Certaines citations, références, séquences, similarités entre documents, sont des relations implicites. Les liens hypertextes sont des références explicites. Un exemple de référence explicite mais non activable peut être représentée en HTML à l'aide de la balise LINK :

<LINK rel="Index" href="../index.html">.

Enfin, un exemple classique de relation activable est une référence vers l'index d'un site Web qui peut être représentée en HTML à l'aide de la balise A :

<A href="../index.html"> index du site .

6.8 Relation de composition

Nous avons présenté les documents structurés dans le chapitre 2.2. Ce type d'organisation des documents est le plus courant, dans les normes (SGML, HTML, XML, etc.) comme dans l'usage qui en est fait. Il est donc naturel que la notion de composition et de structure hiérarchique qui en découle se retrouvent à la base de notre modèle d'hyperdocuments.

6.8.1 Agrégation et composant/composé

La relation de composition $\mathcal{R}_{comp}$ est la relation qui permet d'agréger plusieurs entités d'une granularité donnée, pour obtenir une nouvelle entité de granularité supérieure. On retrouve ici les notions de composant (l'entité destination de la relation) et de composé (l'entité source de la relation).

Ainsi, la notion de **document atomique** étant définie (cf. section 6.6), la relation de composition permet de construire des entités structurées, dont la granularité va croissant avec l'application de la composition. Au niveau de description du signifiant, on trouve typiquement des paragraphes, des sections, des chapitres, etc. L'agrégation des entités avec la relation de composition est récursive : il est possible de décrire une entité composée de plusieurs entités qui sont elles-mêmes composées de plusieurs entités, etc. Il existe des contraintes sur les relations de composition, de manière à ne pas décrire de cycle et que le résultat final soit un arbre. On retrouve ainsi la structure hiérarchique arborescente classique.

6.8.2 Signifiant et signifié

La relation de composition existe au niveau du signifiant (c'est la relation de composition logique) et au niveau du signifié (c'est la relation de dominance de Fourel [Fourel98]). Elle permet donc de décrire la structure de l'information, appelée structure logique au niveau du signifiant et structure de discours au niveau du signifié. L'auteur d'un document construit généralement la structure logique en fonction de la structure de discours de l'information qu'il désire transmettre.

6.8.3 Composition et hypertextes

Du point de vue des hypertextes, la relation de composition a aussi un rôle à jouer. En effet, à l'instar d'un document structuré, un hypertexte est composé de plusieurs nœuds. Il existe donc une relation de composition entre un hypertexte et l'ensemble de ses nœuds. Il s'agit d'une relation d'*hypercomposition*, par opposition à la relation de *composition documentaire*.

L'hypercomposition construit elle aussi une structure hiérarchique, au niveau du signifiant et au niveau du signifié. En effet, la structure documentaire (structure logique) n'est qu'une spécialisation de la structure hypertexte (structure hyperlogique), avec des restrictions sur les cheminements associés.

En conséquence, du point de vue de la composition, un hyperdocument n'est rien d'autre qu'un document structuré. Nous détaillons dans la section 6.9 comment les chemins de lecture permettent de définir des hyperdocuments à partir des documents structurés. Le tableau 6.13 récapitule la liste exhaustive des différentes déclinaisons de la relation de composition :

Relation		Niveau de description	Relation	Type de Structure
Type	Sous-type			
Composition ($\mathcal{R}_{comp}$)	Composition documentaire ($\mathcal{R}_{Dcomp}$)	Syntaxique	$\mathcal{R}_{Dcompdoc}$	Logique
		Sémantique	$\mathcal{R}_{Dcompinf}$	Discours
	Hypercomposition ($\mathcal{R}_{Hcomp}$)	Syntaxique	$\mathcal{R}_{Hcompdoc}$	Hyperlogique
		Sémantique	$\mathcal{R}_{Hcompinf}$	Hyperdiscours

FIG. 6.13 – La relation de composition.

Dans la suite, nous ne parlerons donc plus d'hypercomposition mais uniquement de composition. De plus, la description du modèle réduite à $\mathcal{HDOCC}$ simplifie encore la composition à la seule relation $\mathcal{R}_{comp}$.

On remarque que l'hypercomposition pourrait admettre le partage des composants. Par exemple, le site Web de l'équipe MRIM présente les projets de recherche dans lesquels des membres de l'équipe sont impliqués. Ces projets se déroulent souvent en coopération avec une autre équipe : dans ce cas là, la présentation d'un projet peut être commune aux deux participants et hébergée sur le site Web d'un seul d'entre eux. Cette présentation pourrait être considérée comme un composant des deux sites Web des équipes impliquées. Dans notre modèle d'hyperdocuments, nous préférons une composition sans partage, et nous tenons compte des cas particuliers de ce type à l'aide de la relation de référence (cf. section 6.10). On représentera alors le projet commun comme étant un composant du premier site et comme faisant partie du contexte du second.

6.8.4 Définitions : la relation de composition $\mathcal{R}_{comp}$

Ces considérations nous amènent à la définition de la **relation de composition** $\mathcal{R}_{comp}$:

Définition 5 ***Relation de composition*** *: $\mathcal{R}_{comp}$ est la relation binaire "est-composé-par", interne à l'ensemble $\mathcal{DS}$ des documents structurés. Elle décrit l'organisation structurelle des documents aux différents niveaux de granularité.*

La relation de composition définit un ordre partiel sur les composants d'un document structuré et construit une structure hiérarchique arborescente. Elle permet de construire des documents structurés ds. L'ensemble des documents structurés ds est appelé $\mathcal{DS}_{doc}$. On remarque qu'un document atomique a est un document structuré réduit à sa racine : l'ensemble des documents atomiques est donc un sous-ensemble de $\mathcal{DS}_{doc}$.

Définition 6 ***Documents structurés*** *: un document structuré ds_i est une structure arborescente organisée par la relation de composition, avec un ensemble de documents atomiques $a_j \in \mathcal{A}_{doc} \subseteq \mathcal{DS}_{doc}$ comme feuilles et un ensemble de documents structurés $ds_j \in \mathcal{DS}_{doc}$ comme nœuds intermédiaires.*

Il est nécessaire de distinguer les documents selon leur taille et leur degré de composition. En effet, en terme de résultat d'un SRI, un document atomique n'est pas comparable à un document structuré. Nous utilisons les notions classiques de hauteur et de taille de l'arbre des documents. Cependant, ces mesures sont indépendantes du reste du corpus. Pour pouvoir comparer l'aspect "*degré de composition*" de deux documents, il est nécessaire de définir une mesure qui soit relative au reste du corpus. Nous définissons la "*granularité*" des documents comme suit :

Définition 7 ***Granularité :*** *la granularité "$gran$" d'un document représente son degré de composition. Un document atomique est un élément de granularité minimale, et la granularité augmente avec la composition.*

On définit les fonctions $taille$, $hauteur$ et *granularité* sur les documents de $\mathcal{DOCC}$. Alors que la taille (respectivement la hauteur) est relative au nombre de feuilles (respectivement, à la hauteur de l'arborescence), la granularité combine les aspects hauteur et taille d'un document en fonction des autres documents du corpus.

6.8.5 Exemples

Il existe une relation de composition entre un nœud non feuille de la structure hiérarchique d'un document et chacun des nœuds (ou feuilles) qui le composent. Selon que l'auteur a voulu décrire un "document" consultable à la manière des documents structurés ou à la manière des hypertextes, cette relation de composition est une relation d'hypercomposition ou une relation de composition documentaire.

Exemple 1 *sur le site Web de MRIM, il existe une relation de composition entre le document structuré (la page) "Présentation" et chacun de ses paragraphes.*

Chaque page du site a été décrite à la manière d'un document structuré. Par contre, les sous-parties du site de MRIM n'ont pas été conçues pour obligatoirement être lues dans un ordre prédéterminé, ce qui leur donne leur caractère hypertextuel :

Exemple 2 *sur le site Web de MRIM, il existe une relation d'hypercomposition entre l'entité "Site MRIM" et les pages "Présentation", "Projets", "Membres", etc. L'hyperdocument ainsi construit est composé des documents structurés "Présentation", "Projets", "Membres", etc.*

La figure suivante montre l'arborescence syntaxique du site Web de MRIM, avec les niveaux de composition documentaire et d'hypercomposition, qui sont unifiés dans notre modèle en un seul niveau de composition.

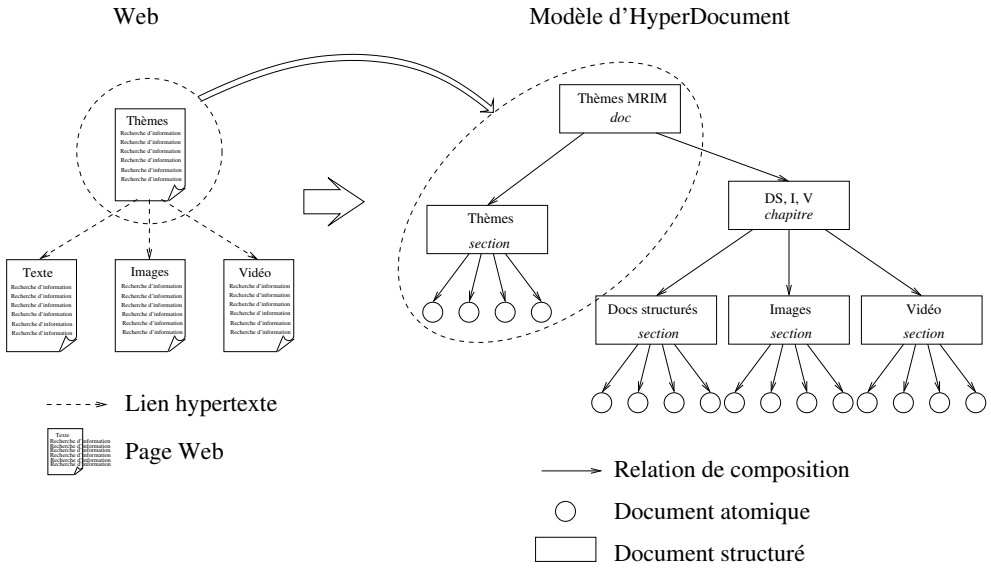

FIG. 6.14 – L'arborescence syntaxique du site Web de MRIM.

La structure physique de l'arborescence du site Web est modifiée dans sa représentation : l'entité physique "Thèmes de recherche" composée de 4 pages Web "Thèmes", "Docs structurés", "Images", "Vidéo", est représentée par un document structuré "Thèmes" composé de 5 documents structurés (nœuds) et de 16 documents atomiques (feuilles). On remarque que la page Web "Thèmes" est représentée par le document structuré "Thèmes" composé de 4 documents atomiques, et que les pages "Docs structurés", "Images", "Vidéo" sont représentées par 3 autres documents structurés. Le document structuré "DS, I, V" (respectivement, "Thèmes MRIM") représente la composition des documents structurés "Docs structurés", "Images" et "Vidéo" (respectivement, "Thèmes" et "DS, I, V").

Le document "Thèmes" est d'une granularité plus importante que les documents feuilles qui le composent et d'une granularité plus faible que le document "DS, I, V". En effet, ce dernier possède un degré de composition plus élevé et la hauteur de l'arbre le représentant est plus importante.

6.9 Relation de cheminement

Dans le contexte des documents structurés, nous avons vu dans le chapitre 2.2 qu'un typage classique des relations comme celui proposé par Fourel [Fourel98] distinguait les relations de composition, les relations de référence et les relations de séquence. Nous pensons que le type "relation de séquence" est particulièrement adapté aux documents structurés : il en est même un des principes essentiels. Mais ce type de relation est moins pertinent dans le contexte des hypertextes, du moins sous sa forme classique qui est basée sur une notion d'ordre total entre les fragments d'un document structuré.

6.9.1 Lecture de textes et d'hypertextes

L'innovation des hypertextes porte principalement sur la possibilité d'organiser des textes de manière non linéaire, pour imiter le fonctionnement du cerveau humain, comme l'a proposé Vannevar Bush dans "*As we may think*" [Bush45], sur le modèle du travail intellectuel du chercheur. La différence entre texte et hypertexte est mise en avant par Laufer d'un point de vue syntaxique :

> « *Le texte est un ensemble de paragraphes successifs, réunis en articles ou chapitres, imprimés sur du papier et qui se lisent habituellement depuis le début jusqu'à la fin. Un hypertexte est un ensemble de données textuelles numérisées sur support électronique, et qui peuvent se lire de diverses manières. Les données sont réparties en éléments ou nœuds d'information - équivalents à des paragraphes. Mais ces éléments, au lieu d'être attachés les uns aux autres comme les wagons d'un train, sont marqués par des liens sémantiques qui permettent de passer de l'un à l'autre lorsque l'utilisateur les active. Les liens sont "ancrés" à des zones, par exemple à un mot ou une phrase.* » [Laufer92].

L'espace d'information d'un hypertexte est donc radicalement différent de celui des documents structurés. Avec des documents structurés ou un discours oral, le discours est linéaire, comme le montre d'ailleurs l'expression "perdre le fil de son discours", d'où l'idée de représenter ce "fil" par une relation de séquence entre les fragments du discours. Clément considère que « *Le texte imprimé introduit une deuxième dimension. Aux deux repères de* ***l'avant*** *et de* ***l'après*** *du discours oral, il ajoute ceux du* ***plus haut*** *et du* ***plus bas*** » [Clement95a]. Le lecteur gagne ainsi une liberté : celle de feuilleter un livre dans le désordre, de sauter certains chapitres, etc. Mais cette liberté est relative : le dispositif de lecture utilisé (livre sur papier ou même consultation d'un livre électronique) limite généralement les possibilités de vagabondage du lecteur.

Par contre, avec l'hypertexte, la lecture linéaire imposée par la séquentialité du document structuré est seulement une des possibilités de lecture d'un "document", comme l'exprime Clément :

> « *L'œuvre hypertextuelle, en effet, compense les limites de l'écran en offrant au lecteur de nouvelles possibilités que n'a pas le livre. Car derrière le cadre rectangulaire qui limite notre champ de lecture, l'ordinateur offre une profondeur qui n'est pas seulement celle de notre espace familier à trois dimensions mais celle, beaucoup plus vertigineuse, d'un espace multidimensionnel, de ce que l'on appelle désormais un "hyperespace". Tel passage que je suis en train de lire sur mon écran n'est plus enchaîné à celui qui lui succède immédiatement. Il s'inscrit dans une structure hypertextuelle qui tisse entre les divers fragments un réseau complexe de liens potentiels. Ma lecture n'est donc plus soumise à l'ordre immuable des pages, elle s'ouvre sur un nouvel espace que je parcourrai désormais au gré de mes humeurs ou de mes curiosités, lecteur-explorateur d'un nouveau type de texte aux perspectives sans cesse en mouvement.* » [Clement95b].

Nous appelons ce type de lecture un **cheminement déambulatoire**. De tels chemins de lecture avaient déjà été imaginé par Vannevar Bush, l'inventeur du concept d'hypertexte, qui les avait nommé "*trails*" dans "*As we may think*" : « *It is exactly as though the physical items had been gathered together to form a new book. It is more than this, for any item can be joined into numerous trails* » [Bush45].

6.9.2 Hyperfiction et lecture non linéaire

Le domaine littéraire s'est intéressé très tôt à la problématique de l'écriture hypertextuelle. Par exemple, la technique d'inspiration collective dite des "cadavres exquis"[4] des surréalistes s'apparente à une écriture hypertextuelle, en raison des combinaisons possibles, de l'aspect collectif de l'écriture, et du fait que la signification naît de l'enchaînement de fragments. L'informatique a facilité la tâche des écrivains dans leur tentative de faire éclater les limites de la linéarité et de s'affranchir des contraintes de l'écrit imprimé.

Les "*livres dont vous êtes le héros*" ont préfiguré la notion d'hyperfiction. Un livre dont vous êtes le héros est un livre composé d'un grand nombre de fragments dont le lecteur tient le rôle principal, et qui comporte des variantes en fonction des choix du lecteur sur certains nœuds du récit. Les dés peuvent être utilisés pour déterminer le résultat d'une action du lecteur/joueur. Ces livres ne sont pas des hypertextes, mais plutôt des récits multilinéaires.

Dans les années 1980, Michael Joyce a lancé le courant de la littérature dite "de fiction hypertextuelle" (hyperfiction) avec la nouvelle électronique "*Afternoon, a Story*" [Joyce85]. Le texte de cette nouvelle est composé de 539 fragments reliés par 950 liens, qu'on ne peut pas lire de manière linéaire. Clément analyse la lecture de "Afternoon, a Story" dans [Clement94]. Le lecteur avance dans le récit en faisant des choix, comme dans un "*livre dont vous êtes le héros*", mais avec un cheminement beaucoup plus complexe. L'hypertexte évolue au fur et à mesure de la lecture, en fonction des choix faits par le lecteur, et les nouveaux choix proposés sont fonction des choix antérieurs. Le lecteur peut donc faire différentes interprétation à la lecture d'un fragment, selon les fragments qu'il a déjà lus. Par exemple, un fragment contenant la seule phrase « *Are you sleeping with her ? he asks* » peut être prononcé par différents personnages de l'intrigue, selon le chemin de lecture emprunté (les fragments qu'on vient de parcourir) : cela peut changer radicalement la compréhension de l'histoire. Joyce a déconstruit entièrement la linéarité de la narration, créant une histoire sans début ni fin, et dont chaque lecture peut donner lieu a une interprétation différente.

6.9.3 Navigation dans un hypertexte

Mais cette liberté qu'a le lecteur de vagabonder de nœud en nœud ne doit pas suggérer que l'espace d'information est complètement désorganisé. En effet, on se place toujours dans le contexte d'une transmission d'information de l'auteur vers le lecteur, et si l'auteur veut faire passer un message, il devra assurer une certaine logique dans son hypertexte. Bien sûr

[4]Le jeu des "cadavres exquis" consiste à élaborer un texte collectivement : chaque participant propose un mot ou une phrase, sans connaître les mots ou les phrases précédemment proposés.

il est difficile d'avoir une maîtrise totale de l'hypertexte que l'on est en train d'écrire : le nombre de chemins potentiels dans le graphe de nœuds augmente très vite avec le nombre de combinaisons et d'enchaînements possibles des relations de cheminement. Toute la difficulté de l'écriture d'hypertextes réside dans la construction d'un espace de navigation suffisamment ouvert pour permettre au lecteur de vagabonder, mais comportant des repères et des lignes directrices permettant au lecteur de reconstituer un fil directeur.

C'est pour cette raison que la métaphore de la navigation est adaptée à l'exploration d'un espace d'information hypertexte. Le lecteur "navigue" dans un océan d'information, aidé d'une carte et d'une boussole pour parcourir les zones d'information qui l'intéressent. Ainsi, un "bon" hypertexte contient des repères et des panneaux indicateurs, comme les liens indiquant le retour à la page principale ou le passage au chapitre suivant. Un hypertexte peut aussi contenir une carte qui le décrit dans sa globalité, permettant au lecteur de se faire une idée de l'espace d'information proposé avant de l'explorer. Sans cela, la navigation s'apparente plus à une dérive, une promenade aléatoire parmi des informations qui n'ont pas de lien apparent entre elles.

Cette vision de la navigation dans un espace d'information linéaire (document structuré) ou non linéaire (hypertexte) comporte donc un aspect "libre", qui peut aller jusqu'à la navigation aléatoire, mais comporte aussi un aspect "supervisé" qui peut aller jusqu'à la contrainte d'une lecture linéaire. Le juste milieu consiste peut-être, pour l'auteur, à proposer un ensemble de chemins potentiels décrivant une information structurée, en imaginant le réseau complexe de liens qui les organise. Le lecteur a ensuite son rôle à jouer dans la construction du sens : « *La lecture ne fait surgir qu'une des potentialités de parcours, elle ne trace qu'un chemin parmi d'autres possibles* » [Clement95a].

6.9.4 Aspects hypertextuels du Web

Les hyperfictions illustrent parfaitement la construction délinéarisée du sens poussée à l'extrême avec un hypertexte. Ce type de description de l'information n'est pas le plus courant sur le Web[5]. En effet, la plupart des auteurs éprouvent encore des difficultés à percevoir un site Web autrement que comme un document structuré arborescent, c'est pourquoi la plupart des sites du Web sont encore construits en suivant ce modèle. Une étude sur le processus cognitif de lecture de documents non linéaires [Rouet92] montre les difficultés rencontrées par différents types d'utilisateurs, dans la lecture et l'apprentissage d'un hypertexte classique. En particulier, il ressort que l'utilisation efficace de la non linéarité pour ces tâches nécessite de se familiariser avec les particularités des hypertextes, et que pour une grosse majorité des utilisateurs, l'adjonction d'une aide à la navigation est indispensable (par exemple sous la forme de cartes arborescentes de l'hypertexte).

Mais on voit de plus en plus de sites Web laissant une liberté plus grande au lecteur, et nous pensons que ce type de construction est appelé a avoir une importance encore plus grande dans le futur. Les auteurs de pages Web prennent conscience de l'intérêt de proposer

[5] Voir le *portail de la littérature hypertextuelle francophone* : http ://www.hypertextes.com

différents scénarios de lecture, selon par exemple l'expertise du lecteur ou selon ses centres d'intérêt, comme le montre le "*visitor profiling*". Cette branche de l'e-commerce s'attache à extraire des profils types pour classifier les visiteurs de sites marchands, afin d'adapter le contenu du site. Le site Web du futur ne sera pas une simple hiérarchie de pages, mais possédera plusieurs vues différentes de son contenu avec plusieurs chemins de lecture possibles, adaptés au profil du visiteur.

Dans le cas général, les chemins de lecture ne sont pas définis explicitement par l'auteur du site, et c'est alors au lecteur de reconstituer un chemin qui lui permette de retirer de l'information cohérente de sa lecture. Il est possible d'identifier de tels chemins en utilisant des techniques de *usage mining*. Le *usage mining* est une branche du *Web mining*, qui est lui-même une application de techniques de *Data Mining* au Web en vue d'extraire des connaissances. Le *usage mining* est le processus d'extraction de modèles ou de patrons à partir des statistiques d'accès à un site Web : les *Web access logs*[6] [Masseglia02]. Par exemple, on pourra déterminer quels sont les chemins couramment suivis par les internautes sur un site Web marchand, afin de construire des profils types de consommateurs, et éventuellement de remodeler le site si on constate qu'il n'est pas ergonomique ou inadapté au besoin des utilisateurs.

Au cours d'expérimentations sur les statistiques de 560 000 accès au serveur Web du laboratoire CLIPS, nous avons extrait des relations de cheminement en identifiant plus de 40 000 "visites" (c'est-à-dire des séquences d'accès aux pages Web par une même machine). Par exemple, la relation de cheminement la plus fréquemment suivie sur la page d'accueil est celle menant à l'annuaire des membres du laboratoire, viennent ensuite les relations vers les pages d'accueil des différentes équipes.

6.9.5 Cheminement et chemins de lecture

Dans le cas des documents structurés, la relation de cheminement des documents est la relation de séquence classique, que nous appellerons *relation de cheminement documentaire*. Dans le cas des hypertextes, la relation de cheminement prend toute sa dimension. Nous l'appellerons *relation de cheminement déambulatoire* pour refléter le fait qu'une grande liberté est laissée au lecteur pour construire son itinéraire dans l'espace d'information. D'autre part, la relation de cheminement existe au niveau de description syntaxique comme au niveau de description sémantique, selon si elle décrit un cheminement dans une information/hyperinformation ou dans un document/hyperdocument.

Un site Web comporte généralement une multitude de relations de cheminement entre les nœuds de l'hypertexte, qui se traduit par différents "chemins possibles" pour sa consultation. Un enchaînement de liens de cheminement, prévu ou non par l'auteur, est appelé *chemin de lecture*. Le tableau suivant récapitule la liste exhaustive des différentes déclinaisons de la relation de cheminement :

[6] Les fichiers dans lesquels chaque accès à un site Web est enregistré : date, heure, page demandée, etc.

Relation		Niveau de description	Relation
Type	Sous-type		
Cheminement ($\mathcal{R}_{chem}$)	Cheminement documentaire ($\mathcal{R}_{Dchem}$)	Syntaxique	$\mathcal{R}_{Dchemdoc}$
		Sémantique	$\mathcal{R}_{Dcheminf}$
	Hypercheminement ($\mathcal{R}_{Hchem}$)	Syntaxique	$\mathcal{R}_{Hchemdoc}$
		Sémantique	$\mathcal{R}_{Hcheminf}$

FIG. 6.15 – La relation de cheminement.

La relation de cheminement documentaire n'est qu'une spécialisation de la relation de cheminement déambulatoire. Dans la suite, nous parlerons donc uniquement de la relation de cheminement $\mathcal{R}_{chem}$, le cheminement documentaire étant distingué par la modélisation du *chemin standard* d'un document structuré.

6.9.6 Chemins de lecture standard

Nous faisons l'hypothèse que la structure hiérarchique des documents structurés, construite avec la relation de composition, entraîne l'existence d'un *chemin standard* de lecture :

Hypothèse 2 *Impact de la composition sur le cheminement : pour chaque document structuré, il existe un chemin de lecture standard et un seul pour le parcourir.*

Par opposition au *chemin de lecture déambulatoire*, on appelle *chemin de lecture standard* d'un document structuré le chemin de lecture qui passe une fois et une seule par chacun de ses documents atomiques, et qui permet de collecter la totalité de l'information en considérant le document dans sa globalité. Il existe un chemin standard à l'intérieur des simples pages Web, même si les relations qui le représentent ne sont pas matérialisées par des liens hypertextes. Ce chemin reflète le développement progressif et cohérent de l'information, à la manière des documents structurés classiques.

6.9.7 Définitions : la relation de cheminement $\mathcal{R}_{chem}$

Ces considérations nous amènent à la définition de la **relation de cheminement** $\mathcal{R}_{chem}$:

Définition 8 ***Relation de cheminement*** *: $\mathcal{R}_{chem}$ est la relation binaire qui permet de définir une possibilité de lecture entre deux documents atomiques à l'intérieur d'un document structuré.*

L'ensemble des documents atomiques d'un document structuré peuvent donc être mis en relation deux à deux par la relation de cheminement. Ils constituent ainsi un ou plusieurs chemins de lecture parcourant tout ou partie des composants sans contrainte sur la linéarité ou l'ordre de lecture des nœuds. La relation de cheminement permet donc de construire des chemins de lecture ch_i^k sur un document structuré ds_i. L'ensemble des chemins de lecture ch est appelé $\mathcal{CH}_{doc}$.

Définition 9 ***Chemins de lecture*** *: un chemin de lecture ch_i^k associé à un document ds_i est un chemin sans cycle dans un graphe dont les nœuds sont les documents atomiques de ds_i et les arcs sont des relations de cheminement. Un tel chemin débute par un document atomique initial a_i^k.*

Nous avons également évoqué l'existence d'un *chemin de lecture standard* ch^s pour chaque document structuré ds :

Définition 10 *On appelle* ***chemin de lecture standard*** ch^s *d'un document structuré ds, le chemin de lecture qui passe une et fois et une seule par chacun de ses documents atomiques.*

Enfin, l'association de documents structurés et de chemins de lecture permet de définir les hyperdocuments. Il existe une bijection entre l'ensemble des documents structurés $\mathcal{DS}_{doc}$ et l'ensemble des hyperdocuments $\mathcal{HD}_{doc}$.

Définition 11 ***Hyperdocuments*** *: un document structuré ds_i associé à un chemin de lecture standard ch_i^s, et éventuellement à plusieurs chemins de lecture déambulatoires ch_i^k, est appelé un* ***hyperdocument*** hd_i.

La notion de **granularité** est également pertinente dans le cas des chemins de lecture. Elle se base sur la longueur des chemins (en nombre de nœuds) et permet de les distinguer selon le nombre de nœuds parcourus.

Nous considérons la notion de rupture sémantique sur les arcs d'un chemin, afin de prendre en compte les changements de thème au cours d'une lecture :

Définition 12 ***Rupture sémantique*** *: la rupture sémantique β associée à un arc entre deux documents atomiques a_i et a_j représente le degré de changement de thème entre a_i et a_j, et permet d'indiquer une rupture dans le développement thématique de l'information.*

Nous nous basons sur la dissimilarité sémantique des deux nœuds pour estimer s'il y a ou non une rupture sémantique, avec l'hypothèse suivante :

Hypothèse 3 *Continuité et similarité : la continuité du discours est fonction de la similarité des nœuds d'informations successifs qui composent le document véhiculant ce discours.*

Nous faisons l'hypothèse que deux nœuds successifs ayant une similarité faible marquent un changement de sujet, et donc une rupture dans le développement thématique du discours.

6.9.8 Exemples

Il existe souvent un lien de cheminement explicite entre la page principale d'un site et la première page dans la hiérarchie, ainsi qu'entre une page du site et la page "suivante".

Exemple 3 *Sur le site Web de MRIM, il existe une relation de cheminement entre la page principale et les pages "Thèmes", puis "Projets", etc. Ainsi, l'auteur préconise un sens de lecture pour une meilleure compréhension : la page de présentation est à lire avant la page décrivant les projets.*

L'enchaînement de ces relations de cheminement permet de construire le chemin de lecture standard du site :

Exemple 4 *Le chemin de lecture standard du site Web de MRIM commence par la page "Thèmes", puis "Projets", puis "Membres", etc.*

Il existe aussi d'autres relations de cheminement permettant une lecture non linéaire du site. On peut par exemple consulter un site Web en prenant un critère thématique pour décider des liens à suivre, au lieu de se laisser imposer un sens de lecture.

Exemple 5 *Un lecteur peut consulter le site Web de MRIM avec un intérêt particulier pour la RI vidéo. Dans ce cas, il suivra les relations de cheminement déambulatoire définies sur le thème de la vidéo : de la page de présentation du site, il naviguera à la page "Axe Vidéo", puis choisira d'aller consulter les informations sur les "Projets" traitant de vidéo, et il pourra continuer par la consultation des publications sur ce thème, etc.*

L'enchaînement de ces relations de cheminement permet de construire un chemin de lecture déambulatoire. Un tel chemin existe sur le site Web de MRIM, commençant par la page "Présentation", puis "Axe de recherche Vidéo", "Projets sur la vidéo", puis des publications sur la vidéo, etc. La figure 6.16 montre la structure de cheminement de l'hyperdocument représentant le site Web de MRIM.

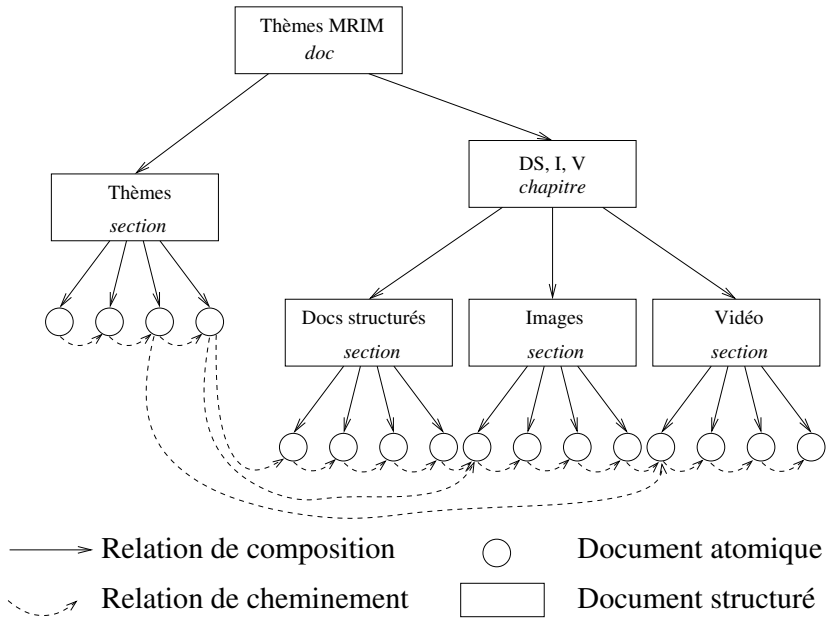

FIG. 6.16 – Structure de cheminement de l'hyperdocument "site Web de MRIM".

6.10 Relation de référence

6.10.1 L'information et le contexte

Les derniers développements de la RI sur le Web montrent l'importance du contexte, avec de nombreux travaux qui proposent de l'intégrer dans le processus de RI (cf. partie I). Ces approches utilisent les notions de popularité, d'autorité (*authorities*) ou de rayonnement (*hubs*). Cependant, peu d'approches considèrent le contexte d'un point de vue sémantique, principalement à cause de la modélisation atomique des documents (une page HTML) et surtout de l'application de méthodes statistiques sur de grandes quantités d'information. Ces méthodes conduisent souvent à appliquer un algorithme calculatoire à la manière "*force brute*", et à déterminer les paramètres optimum du système par apprentissage. Nous avons développé dans la section 6.3 des arguments en faveur de l'utilisation du contexte pour la RI. Nous pensons que dans le cas du Web, où tout peut être relié à tout, le contexte est un élément essentiel de la description de l'information qu'il est nécessaire de représenter au sein même du modèle de documents.

Le contexte situationnel n'est pas modélisé en tant que tel. Par contre il peut l'être par le biais des informations que l'auteur intègre dans ses documents. Par exemple, un lien référençant la page personnelle de l'auteur nous donne une information sur son contexte personnel. Ainsi, notre modèle d'hyperdocuments intègre les trois composantes du contexte textuel présentées dans la section 6.4.4 : le cotexte textuel, le contexte hypertextuel et le contexte référentiel.

6.10.2 Cotexte textuel et contexte hypertextuel

Le cotexte textuel, relevant de l'organisation des informations au sein d'un même document structuré ds_i, est pris en compte au niveau des relations de composition et de cheminement. Par exemple, le cotexte textuel d'un paragraphe p_1 contenu dans une section s_1 et suivi par un autre paragraphe p_2 est pris en compte par la relation de composition entre la section s_1 et le paragraphe p_1, et la relation de cheminement entre le paragraphe p_1 et le paragraphe p_2, comme le montre la figure suivante :

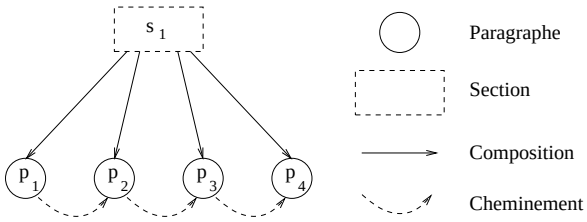

FIG. 6.17 – Un exemple de cotexte textuel au sein d'un document structuré.

La notion de contexte hypertextuel, qui est la transposition du cotexte textuel au cas des hypertextes, est aussi prise en compte au niveau de la composition et du cheminement. Par

contre, la notion de contexte référentiel (cf. figure 6.18) doit être traduite par un autre type de relation. Il s'agit de la relation de référence $\mathcal{R}_{ref}$, qui permet de mettre en relation deux entités n'appartenant pas à un même hyperdocument. Ces relations permettent de décrire les quatre composantes du contexte référentiel des hyperdocuments : l'autorité, le rayonnement, la méta-information et l'information accessible.

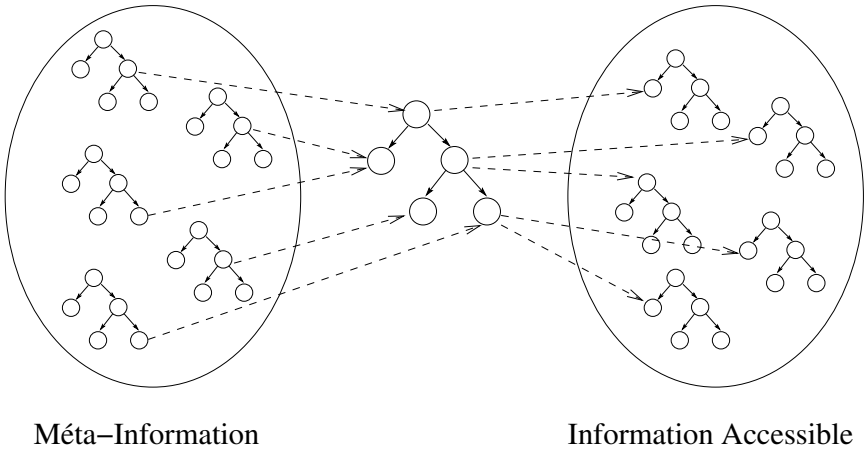

FIG. 6.18 – Un exemple de contexte référentiel.

6.10.3 Autorité et rayonnement

Les notions dérivées de la popularité sont des éléments importants du contexte, qui ont été abondamment utilisés dans la littérature. Il s'agit des notions d'*autorité* et de *rayonnement*, ainsi que de toutes les notions (réputation, qualité, etc.) relevant du principe d'utilisation du réseau de liens pour mettre en avant des pages particulières en tenant compte de leur voisinage, c'est-à-dire des pages référençantes, des pages référencées, et éventuellement de leur pertinence pour une requête donnée. L'autorité et le rayonnement d'un document ds_i sont donc liés au rôle que ds_i joue dans l'hypertexte du Web considéré comme un graphe, et à la quantité et à la popularité des documents référençant ds_i. Ces deux notions sont basées sur des hypothèses qui ont été utilisées par certains travaux présentés dans la partie I en exploitant les liens hypertextes pour la RI. En premier lieu, la notion de popularité présentée dans la section 3.2 participe à l'autorité d'une ressource Web : « *une page référencée par un grand nombre de pages est une bonne page* ».

L'objection que l'on peut faire à la notion de popularité comme preuve de la "qualité" d'une page Web, est le fait que la "qualité" des pages reliées n'est pas prise en compte. En effet, une page référencée par un grand nombre de pages de mauvaise qualité est certes une page populaire, mais que pouvons nous dire de la "qualité" de cette page ? Pour pallier cet inconvénient, la notion d'autorité est utilisée. Il s'agit d'une popularité tenant compte du voisinage des pages : « *une page référencée par un grand nombre de pages pertinentes est une bonne page* ». La notion duale de l'autorité est le rayonnement, qui considère non plus

les références vers la page comme représentatives de sa "qualité", mais les références qui sont décrites dans la page vers d'autres pages. Si on tient compte de la pertinence des pages référencées, l'hypothèse sous-jacente à la notion de rayonnement est : « *une page référençant un grand nombre de pages pertinentes est une bonne page* ».

Enfin, pour combiner ces deux notions, on retrouve les *Hubs* et les *Authorities* de Kleinberg [Kleinberg99] présentées dans le chapitre 4. Ces notions sont semblables aux notions d'autorité et de rayonnement, avec en plus une interdépendance : « une page *rayonnante* référence beaucoup de pages *autorités*, et une page *autorité* est référencée par beaucoup de pages *rayonnantes* ».

6.10.4 Méta-information et information accessible

Les notions d'autorité et de rayonnement permettent de considérer une certaine "qualité" des ressources Web en fonction de leur rôle dans le réseau de liens de l'hypertexte. Cela permet de considérer le voisinage des pages dans une certaine mesure, grâce à la propagation d'une pertinence binaire. Ainsi, la pertinence initialement propagée d'une page A vers une pages B est égale à 1 si A est dans le voisinage de B (de même, si B est dans le voisinage de A), et à 0 sinon.

Mais ces notions ne sont pas suffisantes pour prendre en compte les aspects sémantiques du contexte. Pour cela, il est nécessaire de considérer la pertinence des pages voisines, relativement ou non à la requête. Afin que cela soit possible indépendamment de la requête, notre modèle d'hyperdocuments autorise la représentation de l'espace d'information accessible à partir d'un hyperdocument. Ainsi, l'**information accessible** représente l'espace d'information que l'utilisateur peut consulter par navigation à partir de la page (cf. figure 6.18). Nous faisons l'hypothèse suivante :

Hypothèse 4 *Information accessible : l'espace d'information accessible par navigation à partir d'un document "doc" fait partie de l'information décrite dans "doc" en tant que potentialité de lecture pour l'utilisateur, et peut être utilisé pour l'extraction du contenu sémantique de "doc".*

De plus, avec les relations de référence, on retrouve les mêmes principes de compréhension de l'information que ceux que nous avons décrits au sujet des hyperfictions, mais à un niveau macroscopique. On parle alors du contexte référentiel global d'une information, par opposition au contexte de lecture interne à une information. Avec le contexte de lecture, on s'intéresse à l'accumulation de l'information qui influence la compréhension au cours de la lecture. Au niveau macroscopique, le contexte référentiel joue un rôle semblable : nous l'appelons *méta-information*. La méta-information d'un document donné représente l'information que peuvent nous apporter les documents référençant ce document (cf. figure 6.18). Nous faisons l'hypothèse suivante :

Hypothèse 5 *Méta-information : l'espace d'information à partir duquel sont définies une ou plusieurs relations de référence vers un document "doc" représente une information à propos de "doc", et peut être utilisé pour l'extraction du contenu sémantique de "doc".*

6.10.5 Relation de référence et contexte

Les documents atomiques sont donc placés dans un contexte référentiel, basé sur la relation de référence. Nous venons de voir que le contexte référentiel comporte différents aspects : l'autorité, le rayonnement, la méta-information et l'information accessible. Ces aspects sont définis pour chaque document atomique, mais ils sont aussi définis pour les documents structurés, les hyperdocuments et les chemins de lecture. En effet, nous considérons qu'il existe une relation de référence entre deux documents structurés ds_i et ds_j, à partir du moment où il existe deux documents atomiques a_j et a_k, respectivement composants de ds_i et ds_j, qui sont eux-mêmes connectés par une relation de référence. Il en est de même pour les hyperdocuments et les chemins de lecture. Nous distinguons donc la relation de référence pour chaque type de documents : la relation de référence atomique $\mathcal{R}_{Aref}$, la relation de référence documentaire $\mathcal{R}_{Dref}$, la relation d'hyperréférence $\mathcal{R}_{Href}$ et enfin la relation de référence chemin $\mathcal{R}_{Cref}$. Le tableau suivant récapitule la liste exhaustive des différentes déclinaisons de la relation de référence :

Relation		Niveau de description	Relation
Type	Sous-type		
Référence ($\mathcal{R}_{ref}$)	Référence atomique ($\mathcal{R}_{Aref}$)	Syntaxique	$\mathcal{R}_{Arefdoc}$
		Sémantique	$\mathcal{R}_{Arefinf}$
	Référence documentaire ($\mathcal{R}_{Dref}$)	Syntaxique	$\mathcal{R}_{Drefdoc}$
		Sémantique	$\mathcal{R}_{Drefinf}$
	Hyperréférence ($\mathcal{R}_{Href}$)	Syntaxique	$\mathcal{R}_{Hrefdoc}$
		Sémantique	$\mathcal{R}_{Hrefinf}$
	Référence chemin ($\mathcal{R}_{Cref}$)	Syntaxique	$\mathcal{R}_{Crefdoc}$
		Sémantique	$\mathcal{R}_{Crefinf}$

FIG. 6.19 – La relation de référence.

Cependant, un hyperdocument contient les mêmes documents atomiques que le document structuré sur lequel il se base. On ne distingue donc pas la relation d'hyperréférence de la relation de référence documentaire. Ainsi, une relation de référence documentaire définie entre deux documents structurés indiquera implicitement une relation de référence entre les deux hyperdocuments associés. Dans la suite, nous parlerons donc uniquement de la relation de référence $\mathcal{R}_{ref}$.

6.10.6 Définitions : la relation de référence $\mathcal{R}_{ref}$

Ces considérations nous amènent à la définition de la **relation de référence** $\mathcal{R}_{ref}$:

Définition 13 ***Relation de référence*** *: $\mathcal{R}_{ref}$ est la relation binaire qui permet de définir une référence entre deux documents atomiques appartenant à deux documents structurés distincts. Par extension, la relation $\mathcal{R}_{ref}$ peut aussi définir une référence entre deux documents structurés, deux chemins de lecture ou deux hyperdocuments.*

La relation de référence $\mathcal{R}_{ref}$ construit le contexte d'un document "doc", qui se traduit au niveau du modèle d'hyperdocuments par la méta-information *méta-info* et l'information accessible *info-acc* de "doc", que nous définissons de la manière suivante :

Définition 14 ***Méta-information*** *: la méta-information méta-info d'un document "doc" est l'ensemble des documents du même type ($\mathcal{A}$, $\mathcal{DS}$, $\mathcal{CH}$, ou $\mathcal{HD}$) que "doc" qui le référencent.*

Définition 15 ***Information accessible*** *: l'information accessible info-acc d'un document "doc" est l'ensemble des documents du même type ($\mathcal{A}$, $\mathcal{DS}$, $\mathcal{CH}$, ou $\mathcal{HD}$) qu'il référence.*

Ces notions nous permettent de définir l'ensemble $\mathcal{DOCC}_{doc}$ des documents en contexte :

Définition 16 ***Document en contexte*** *: l'association d'un document doc et de son contexte référentiel (méta-information méta-info et information accessible info-acc) est appelé un document en contexte : $docc =< doc,$ méta-info, info-acc $>$.*

Le contexte d'un document est défini pour tous les types de documents. Ainsi, les documents structurés comme les hyperdocuments et les chemins de lecture sont plongés dans un contexte, obtenu à partir du contexte de l'ensemble de leurs documents atomiques. Par exemple, l'information accessible d'un document structuré ds est composée de l'espace d'information accessible (un ensemble de documents structurés ds) à partir de chacun des documents atomiques qui le composent. On définit l'ensemble $\mathcal{DSC}_{doc}$ des documents structurés en contexte, l'ensemble $\mathcal{HDC}_{doc}$ des hyperdocuments en contexte, et l'ensemble $\mathcal{CHC}_{doc}$ des chemins de lecture en contexte, comme présenté dans le tableau suivant :

Document	Méta-information	Information accessible	Documents en contexte
$a \in \mathcal{A}$	*méta-info*	*info-acc*	$ac \in \mathcal{AC}$
$ds \in \mathcal{DS}$	*méta-info*	*info-acc*	$dsc \in \mathcal{DSC}$
$ch \in \mathcal{CH}$	*méta-info*	*info-acc*	$chc \in \mathcal{CHC}$
$hd \in \mathcal{HD}$	*méta-info*	*info-acc*	$hdc \in \mathcal{HDC}$

FIG. 6.20 – Les documents en contexte.

6.10.7 Exemples

Dans notre exemple, le site Web de l'équipe MRIM est représenté par un seul hyperdocument hd_{mrim}. Donc, tous les liens hypertextes sortants ou entrants de cet hyperdocument représentent autant de relations de référence.

Exemple 6 *Le site Web de MRIM hd_{mrim} est placé dans le contexte des sites Web qui le référencent (méta-information) et des sites Web qu'il référence (information accessible). Les sites qui référencent hd_{mrim} sont les sites contenant au moins une page Web qui référence une des pages Web de hd_{mrim}, et les sites que hd_{mrim} référence sont les sites contenant au moins une page Web référencée par une des pages Web de hd_{mrim}.*

On retrouve les composantes du contexte à chacun des niveaux de granularité du site Web de MRIM. Par exemple, au niveau des documents structurés :

Exemple 7 *La page Web du site de MRIM qui décrit les projets de l'équipe sur la RI vidéo est placée dans un contexte matérialisé par l'ensemble des liens entrants et sortants de la page, qui référencent par exemple le site Web de laboratoires associés.*

6.11 Synthèse

Le tableau 6.21 synthétise les différents types de relations : les relations de composition $\mathcal{R}_{comp}$, les relations de cheminement $\mathcal{R}_{chem}$ et les relations de référence $\mathcal{R}_{ref}$.

Relation	Niveau	Type de document			
		Atome	Document	Hyperdocument	Chemin
Composition	Syntaxique	-	$\mathcal{R}_{Dcompdoc}$	$\mathcal{R}_{Hcompdoc}$	-
	Sémantique	-	$\mathcal{R}_{Dcompinf}$	$\mathcal{R}_{Hcompinf}$	-
Cheminement	Syntaxique	-	$\mathcal{R}_{Dchemdoc}$	$\mathcal{R}_{Hchemdoc}$	-
	Sémantique	-	$\mathcal{R}_{Dcheminf}$	$\mathcal{R}_{Hcheminf}$	-
Référence	Syntaxique	$\mathcal{R}_{Arefdoc}$	$\mathcal{R}_{Drefdoc}$	$\mathcal{R}_{Hrefdoc}$	$\mathcal{R}_{Crefdoc}$
	Sémantique	$\mathcal{R}_{Arefinf}$	$\mathcal{R}_{Drefinf}$	$\mathcal{R}_{Hrefinf}$	$\mathcal{R}_{Crefinf}$

FIG. 6.21 – Typologie de relations : composition, cheminement et référence.

La description du modèle, réduite à $\mathcal{HDOCC}$, nous permet de simplifier la description des relations. Nous parlerons donc uniquement, dans le modèle d'hyperdocuments, des relations $\mathcal{R}_{comp}$, $\mathcal{R}_{chem}$ et $\mathcal{R}_{ref}$. Nous garderons à l'esprit les sous-types de relations, mais ces trois types suffisent à la description de $\mathcal{HDOCC}$.

6.11.1 Hyperdocuments en contexte

Nous proposons donc le modèle d'hyperdocuments en contexte ($\mathcal{HDOCC}$) intégrant les principes développés dans ce chapitre, pour représenter l'information structurée du Web. La figure 6.22 présente les deux points de vue d'un site Web selon le modèle $\mathcal{HDOCC}$. Ce site est composé de 14 pages HTML, que nous considérons comme autant de documents atomiques. Parmi les liens hypertextes existant entre ces nœuds, certains sont interprétés comme représentant une relation de composition, d'autres comme représentant des relations de cheminement. Les liens sortant du site Web sont interprétés comme représentant une relation de référence. Ce site Web peut donc être représenté du point de vue des documents structurés par deux $ds_i, ds_j \in \mathcal{DS}_{doc}$, et du point de vue des hyperdocuments par deux $hd_i, hd_j \in \mathcal{HD}_{doc}$ associés à ds_i et ds_j. Les hyperdocuments hd_i et hd_j contiennent chacun plusieurs chemins de lecture, de longueur variable.

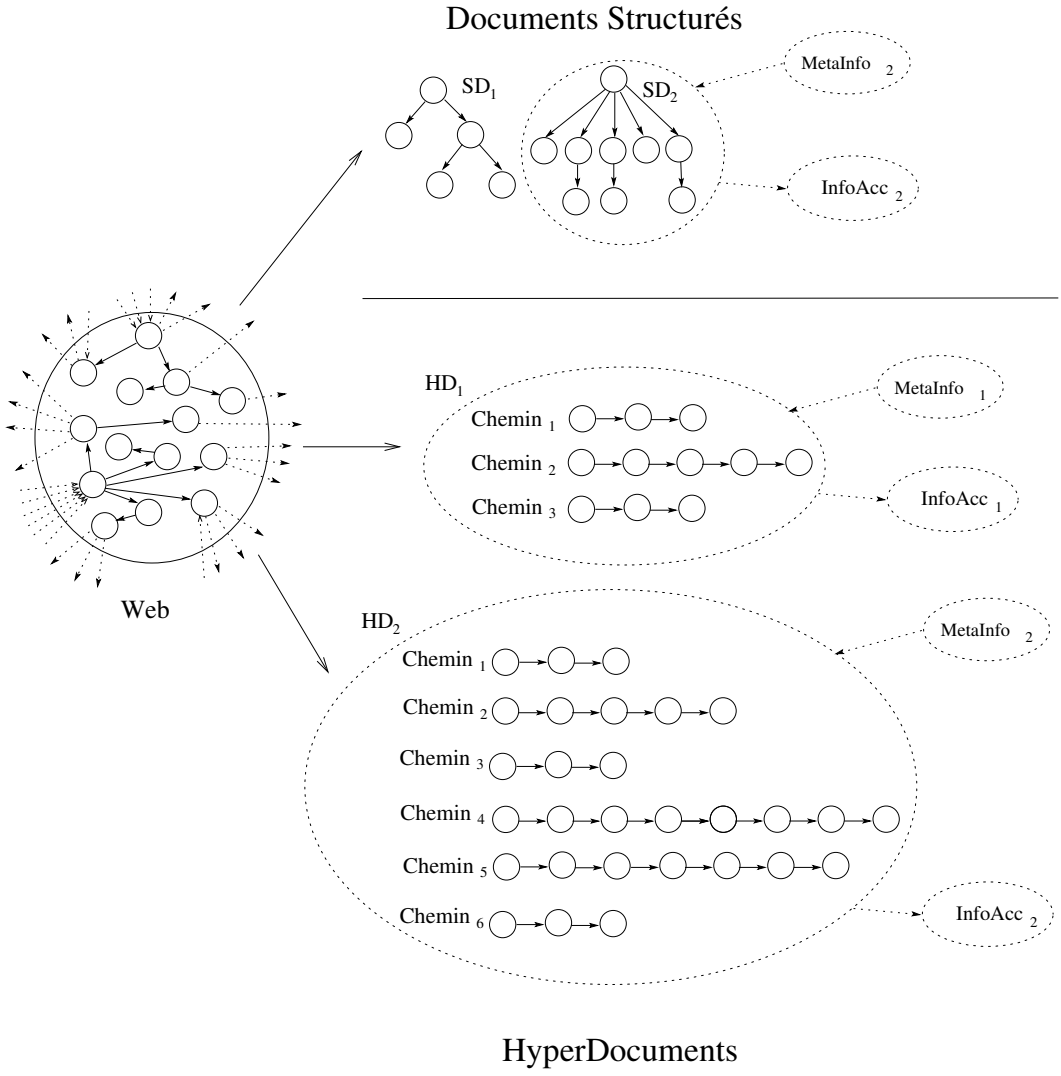

FIG. 6.22 – Modèle d'hyperdocuments $\mathcal{HDOCC}$.

6.12 Impact des relations sur l'indexation

Nous avons présenté la sémantique que nous attachons à chacun des trois types de relations qui sont représentés dans le modèle d'hyperdocuments en contexte, en détaillant les raisons des choix de modélisation. Ces choix sont basés sur les principes que nous avons présentés dans les sections précédentes, visant à représenter l'information en prenant en compte au niveau sémantique les relations qui existent entre les différents types d'information. Les principes régissant la structure d'accueil des hyperdocuments en contexte étant présentés, il reste à décrire l'impact des relations sur l'indexation des hyperdocuments avec comme problématique sous-jacente la question de l'évaluation de la pertinence dans ce contexte que nous abordons dans la section 6.13.

6.12.1 Composition et niveaux de granularité

L'aspect sémantique de la composition, pour l'extraction des index, a été étudiée par exemple dans [Kerkouba84], [Defude86] [Lalmas et al.98] [Picard et al.01] comme présenté dans le chapitre 3. Que ce soit dans le cas de méthodes statistiques de remontées de termes dans l'arborescence du document ou dans le cas de l'utilisation d'un modèle probabiliste, la problématique de l'utilisation de ces relations pour l'indexation est identique. Il s'agit d'utiliser l'indexation des sous-parties ds_j pour aider à l'indexation de l'ensemble ds_i, et éventuellement vice-versa.

Au niveau sémantique, l'objectif de ces diverses méthodes de remontée des pondérations est d'extraire le contenu informationnel d'un document structuré ds_i dans son ensemble. On peut aborder le problème du point de vue de la lecture des documents : quelle sera la quantité d'information que le lecteur pourra assimiler au cours de la lecture de ds_i, par rapport à la quantité d'information qu'il pourra assimiler au cours de la lecture de chacun des ds_j ?

L'application de la composition à l'indexation consiste en une propagation d'information, avec la remontée des pondérations le long de la hiérarchie des documents structurés. Cette propagation doit conserver les hypothèses de la représentativité des termes tout en intégrant l'information de la granularité des documents structurés. Nous proposons un algorithme approprié dans le chapitre 8. Du point de vue du lecteur, l'agrégation suppose que le nœud sera lu de manière "atomique" : le lecteur appréhende le contenu entier du document d'un coup d'œil, c'est une opération instantanée et indépendante du contexte.

6.12.2 Cheminement et construction de l'information

La problématique de l'aide à la navigation dans les hypertextes a été étudiée dans la littérature, mais à notre connaissance il n'existe pas de système de RI qui permette de retrouver des "chemins de lecture" à la manière de ce que proposaient Guinan et Smeaton sur un hypertexte (cf. section 4.2, [Guinan et al.92]). Certains langages de requêtes structurés permettent à l'utilisateur de définir des requêtes sur la structure de l'hypertexte, mais ces travaux n'abordent pas la problématique des chemins de lecture d'un point de vue de l'extraction sémantique de leur contenu.

a) Indexation de chemins de lecture

La construction du sens par le lecteur dépend de la description hypertextuelle de l'information par l'auteur, mais dépend aussi des choix du lecteur. En effet, on peut faire différents parcours d'un même texte, et en tirer différents sens : chaque parcours recontextualisant le texte lu en l'insérant dans une perspective nouvelle.

L'étude de la construction du sens par l'utilisateur qui avance dans sa lecture nous montre l'importance d'une extraction du contenu sémantique qui ne se base pas uniquement sur des moyennes statistiques des pondérations des termes. En effet, il faut considérer les nouvelles notions mises en œuvre dans un cheminement, par rapport au cas de la composition. En particulier, deux chemins de lecture différents qui parcourent le même ensemble de documents

atomiques ne doivent pas produire le même index, en raison de l'importance du sens de lecture et de l'ordre dans lequel les documents sont lus.

b) Progression thématique

Pour permettre au système de retrouver des chemins de lecture, notre objectif est d'extraire leur contenu informationnel en un index. Nous nous basons sur des principes de progression thématique dans un texte intégrant le contexte pour la compréhension au cours d'une lecture [Vandendorpe91b] [Vandendorpe91a]. Nous entendons ici par "contexte" uniquement le contexte de lecture, c'est-à-dire le cotexte textuel interne à un hyperdocument. Tout texte comporte un thème (ce dont il est question), qui est un point de départ. A partir de là, le texte amène des informations nouvelles : c'est le rhème, ou le propos. Le texte est donc une suite de séquences : un développement progressif et cohérent de l'information, communiquée à partir d'un thème donné.

Danes distingue différents schémas de progression thématique dans [Danes74]. Par exemple, avec la progression à thème constant d'un texte, chaque phrase part du même thème en développant des propos (rhèmes) successifs différents :

Phrase p_1 **:** Thème $th_1 \rightarrow$ Rhème rh_1,

Phrase p_2 **:** Thème $th_1 \rightarrow$ Rhème rh_2,

Phrase p_3 **:** Thème $th_1 \rightarrow$ Rhème rh_3,

etc.

Avec la progression à thème linéaire, le propos d'une phrase est repris comme thème de la phrase suivante. Ce nouveau thème fait l'objet d'un nouveau propos, repris lui-même avec le statut de thème :

Phrase p_1 **:** Thème $th_1 \rightarrow$ Rhème rh_1,

Phrase p_2 **:** Thème th'_1 (=Rhème rh_1) $\rightarrow$ Rhème rh_2,

Phrase p_3 **:** Thème th'_2 (=Rhème rh_2) $\rightarrow$ Rhème rh_3,

etc.

Notre proposition d'indexation de chemins de lecture a pour objectif de modéliser la progression thématique, avec les schémas à thème linéaire et à thème constant. Il existe d'autres schémas de progression thématique, comme la progression à thème divisé (thème dérivé d'un hyperthème) dans laquelle le thème d'ensemble (hyperthème) est divisé en sous-thèmes à partir desquels les phrases successives développent de nouveaux propos. Ce type de progression est pris en compte indirectement dans notre modèle, avec l'indexation de plusieurs chemins de lecture pour le même hyperdocument.

c) Indexation et progression thématique

L'indexation d'un chemin de lecture réalise la simulation d'une lecture du chemin et consiste en l'acquisition successive de l'information contenue dans les nœuds du chemin.

Tout d'abord, il nous faut prendre en compte le sens de lecture. Cela signifie qu'un hypertexte composé de 4 nœuds A, B, C et D ne donnera pas le même index selon que les relations de cheminement indiquent une lecture dans l'ordre A, B, C, D ou dans l'ordre D, C, B, A (cf. section 6.9). Nous introduisons pour cela la notion de *mémoire de lecture* :

Hypothèse 6 *Mémoire de lecture : la lecture et la compréhension d'un document atomique a_i dépend de son cotexte textuel, et en particulier des document atomiques $a_1, a_2, ...a_{i-1}$ lus pour arriver jusque-là.*

Nous définissons ensuite le principe d'accumulation, qui permet de prendre en compte la prépondérance des informations du début de la lecture. Par exemple, dans le cas d'un article scientifique comportant un résumé et une introduction à la problématique dans les deux premiers nœuds, on considère que la lecture du reste de l'article est conditionnée par cette entrée en matière.

Hypothèse 7 *Le principe d'accumulation : les informations lues au début ont plus d'importance que les autres, étant donné qu'elles sont réutilisées par la suite en tant que mémoire de lecture. Il y a une accumulation d'information au cours de la lecture, et la mémoire de lecture elle-même bénéficie d'un effet d'accumulation.*

Il faut aussi considérer la possible discontinuité du discours, qui se répercute sur la sémantique extraite d'un chemin de lecture. Pour cela, la rupture sémantique permet de réduire l'impact de la mémoire de lecture et de l'accumulation, en considérant qu'un changement dans le thème du récit (une rupture sémantique) revient à remettre à zéro la mémoire de lecture :

Hypothèse 8 *La rupture sémantique : une rupture sémantique β dans le chemin de lecture indique une discontinuité du récit et entraîne une perte de la mémoire de lecture.*

Enfin, des travaux ont montré l'importance du cotexte textuel pour retrouver des sous-parties de documents (cf. section 3.3, cf. [Wilkinson94]). L'hypothèse de la *mémoire de lecture* permet de considérer le cotexte textuel d'un document atomique au cours de l'indexation d'un chemin, mais le cotexte textuel relatif aux autres documents atomiques du même document structuré, qui ne sont donc pas parcourus par le chemin, n'est pas pris en compte. Nous proposons donc l'hypothèse du *cotexte textuel* :

Hypothèse 9 *Le cotexte textuel : tous les documents atomiques d'un document structuré sont susceptibles de fournir une information pour l'indexation d'une sous-partie de document structuré (comme par exemple un chemin parcourant une partie des documents atomiques).*

Ces hypothèses restent des hypothèses générales, qui nécessitent une validation expérimentale. C'est pourquoi l'algorithme d'indexation décrit dans la section 8.4 propose plusieurs paramètres permettant de faire varier l'importance accordée à chacune des hypothèses.

6.12.3 Référence et mise en contexte

Dans notre modèle d'hyperdocuments, nous représentons les différentes composantes du contexte, en distinguant la méta-information et l'information accessible pour chaque document. Le rôle de l'indexation est d'extraire le contenu sémantique de la méta-information et de l'information accessible d'un document, mais aussi d'évaluer un score d'autorité et de rayonnement. Outre la distinction entre ces quatre composantes de l'indexation, nous proposons aussi de les considérer à des granularités variables. En effet, nous avons montré l'intérêt pour un modèle de RI de s'abstraire de la granularité du Web qui est habituellement utilisée, et qui est imposée pour des raisons pratiques : celle de la page Web. Cet intérêt s'accroît encore dans le cas de propagation de pertinence, d'information (comme avec l'extraction de l'information accessible) ou de popularité (comme dans le calcul des scores d'autorité et de rayonnement), afin d'assurer une propagation qui tienne compte de la granularité des documents.

L'information accessible représente l'information qu'un "surfeur aléatoire" pourrait collecter à partir d'un document doc, et la méta-information représente une information supplémentaire sur doc (cf. section 6). L'extraction de l'information accessible et de la méta-information dépend du *coût de navigation* $\beta_{coût}$ pour accéder au nœud référencé, avec l'hypothèse suivante :

Hypothèse 10 ***Coût de navigation et quantité d'information potentiellement propagée :*** *la quantité d'information potentiellement propagée de chaque fragment du contexte (une relation de référence) est inversement proportionnelle au coût de navigation nécessaire pour activer le lien correspondant.*

On associe également à la méta-information et à l'information accessible le "score" qui lui correspond, c'est-à-dire le score d'autorité "aut" pour la méta-information, et le score de rayonnement "ray" pour l'information accessible. L'impact de la référence sur l'indexation comprend l'indexation de ces composantes au niveau de chaque document atomique, mais aussi à chaque niveau de granularité identifié à l'indexation. Il est donc nécessaire de développer des méthodes d'extraction de chacune de ces quatre composantes en fonction de la valeur de ces composantes pour les niveaux inférieurs.

6.13 Impact des relations sur la pertinence

L'indexation structurée du Web pose le problème de l'évaluation d'un SRIS (Système de Recherche d'Information Structurée). En effet, les schémas classiques d'évaluation, basés sur les notions de rappel et de précision, ne sont pas adaptés, car ils se fondent sur une représentation de documents atomiques, non structurés et indépendants les uns des autres. Avant de revenir sur cette problématique dans le chapitre 9.2, nous introduisons brièvement dans cette section l'impact des relations sur la pertinence d'un "document".

6.13.1 Composition et pertinence

La prise en compte de différents niveaux de granularité des documents rend les méthodes traditionnelles d'évaluation des SRI inadaptées. Il est toujours possible d'évaluer un système qui retrouve des documents en utilisant leur structure, mais l'évaluation ne portera que sur la recherche de documents d'une granularité donnée afin de pouvoir utiliser les critères de rappel/précision.

Si on désire évaluer réellement une RI structurée, il est nécessaire de redéfinir la notion de pertinence, qui doit alors considérer le paramètre de la granularité pour ne pas traiter sur un même plan des documents de granularités différentes. Il s'agit donc d'un problème de normalisation de la pertinence en fonction de la granularité des documents, de la même manière que, dans le cas d'une RI classique, on normalise en fonction de la taille des documents pour ne pas privilégier exagérément les documents de tailles plus importantes.

Il est aussi envisageable de considérer la granularité des documents recherchés comme un paramètre du système, en demandant à l'utilisateur sa préférence en matière de granularité. Dans ce cas, l'évaluation d'un système doit se faire en considérant la dimension supplémentaire de la granularité, en se plaçant donc dans un espace à trois dimensions : rappel, précision et granularité.

6.13.2 Cheminement et pertinence

La composition nécessite de revoir les méthodes d'évaluation et la notion de pertinence. Si l'indexation de documents structurés ne peut pas se satisfaire des méthodes d'évaluation traditionnelles, *a fortiori* l'indexation de chemin de lecture ne peut s'en satisfaire non plus. La pertinence d'un chemin de lecture pour une requête doit intégrer la quantité d'information relative à la requête qui est disséminée le long du chemin. On doit aussi considérer la taille de ce chemin pour normaliser.

6.13.3 Référence et pertinence

Notre modélisation de l'information permet de considérer les dépendances entre les entités de différentes granularités, avec les quatre composantes du contexte. L'évaluation de la pertinence d'une entité doit donc prendre en compte chacune de ces composantes, et peut considérer plusieurs extensions au modèle de requête classique, destinées à répercuter aussi ces composantes sur le modèle de requête.

En premier lieu, l'information accessible est très importante pour l'évaluation de la pertinence d'un cheminement en contexte. On doit aussi considérer la quantité d'information que le lecteur pourra collecter en explorant cette information accessible, en fonction du caractère "focalisé" ou "dé-focalisé" de la requête. L'évaluation d'un SRIS doit se faire en considérant la dimension supplémentaire du focus : en se plaçant donc dans un espace à quatre dimensions : rappel, précision, granularité et focus.

6.14 Organisation du modèle de RI Structurée

Nous avons présenté dans ce chapitre les principes de base de notre modèle de Recherche d'Information visant à représenter et à retrouver des *hyperdocuments en contexte*. Les chapitres suivants sont consacrés à la formalisation de ce modèle : nous présentons dans les chapitres 7 et 8 les trois composants du modèle de RI :

Modèle d'hyperdocuments en contexte : nous détaillons dans le chapitre 7 l'aspect "syntaxe" du modèle de RI, c'est-à-dire la formalisation du modèle d'hyperdocuments en contexte, intégrant la représentation des documents atomiques, de la structure logique, des chemins de lecture et du contexte.

Processus d'indexation structurelle : nous présentons dans le chapitre 8 la formalisation de l'extraction du "contenu sémantique des documents", c'est-à-dire le processus d'indexation structurelle dans la structure d'accueil, qui dans notre cas ne se limitera pas à une extraction d'un contenu atomique mais sera complété par l'extraction des chemins de lecture et leur mise en contexte.

Processus d'interrogation : enfin, le chapitre 8 contient également la formalisation du besoin de l'utilisateur, c'est-à-dire le modèle de requête, ainsi que la description du processus d'interrogation, c'est-à-dire la fonction de correspondance.

Chapitre 7

Modèle d'hyperdocuments en contexte

Dans le chapitre 6, nous avons présenté les principes sur lesquels se base notre modèle de Recherche d'Information visant à représenter et à retrouver des *hyperdocuments en contexte*. Le chapitre 7 résume la formalisation du modèle de documents $\mathcal{HDOCC}$ appelé **modèle d'hyperdocuments en contexte**, qui combine le contenu avec les relations de composition (structure logique), de cheminement (chemins de lecture) et de référence (contexte) pour l'indexation de l'*hypertexte Web*.

7.1 Schéma général du modèle d'hyperdocuments

Le schéma général des informations qui sont modélisées a été introduit dans la figure 6.1 (cf. section 6.2). On y retrouve les quatre couches qui sont à la base de notre modèle d'hyperdocuments en contexte. La première couche du modèle décrit le document réduit à sa plus simple expression (avec les **documents atomiques** $\mathcal{A}$, cf. section 7.2), et les autres couches intègrent la composition, le cheminement et le contexte. Ainsi, la deuxième couche représente les **documents structurés** $\mathcal{DS}$ comportant une structure arborescente basée sur la relation de **composition** (cf. section 7.3). La troisième couche s'intéresse au **cheminement**, et représente les **hyperdocuments** $\mathcal{HD}$ qui sont des documents structurés auquel sont associés des **chemins de lecture** $\mathcal{CH}$ (cf. section 7.4). Enfin, la quatrième couche de notre modèle aborde le problème de la **mise en contexte** de l'information, et représente la **méta-information** *méta-info* et l'**information accessible** *info-acc* d'un document (cf. section 7.5).

7.1.1 Signifiant, signifié, et pragmatique

L'élaboration du modèle suit la logique définie dans le modèle de transmission de l'information (cf. section 6.4), avec les niveaux du signifiant ($\mathcal{HDOCC}_{doc}$, syntaxique), et du signifié ($\mathcal{HDOCC}_{inf}$, sémantique/pragmatique). Cependant, bien qu'il n'y ait pas de bijection entre $\mathcal{HDOCC}_{doc}$ et $\mathcal{HDOCC}_{inf}$, nous avons choisi de décrire le seul niveau du signifiant, c'est-à-dire $\mathcal{HDOCC}_{doc}$. Ce choix simplificateur est justifié par la symétrie qui existe entre les deux niveaux de description.

7.1.2 Les composants de $\mathcal{HDOCC}$

Les composants du modèle d'hyperdocuments en contexte $\mathcal{HDOCC}$ sont les ensembles décrivant les différents types de documents ($\mathcal{A}$, $\mathcal{DS}$, $\mathcal{CH}$, $\mathcal{HD}$) et la mise en contexte de chaque type ($\mathcal{AC}$, $\mathcal{DSC}$, $\mathcal{CHC}$, $\mathcal{HDC}$). Cette modélisation se base sur les types de relations entre les documents ($\mathcal{R}$), dont les caractéristiques ont été détaillées dans le chapitre 6.

On distingue les relations de composition, de cheminement et de référence :

$$\mathcal{R} = \mathcal{R}_{comp} \cup \mathcal{R}_{chem} \cup \mathcal{R}_{ref} \tag{7.1}$$

Les ensembles utilisés pour la description du modèle d'hyperdocuments $\mathcal{HDOCC}$ sont récapitulés dans le tableau 7.1 qui vient compléter le tableau 6.2. La symétrie avec le niveau du signifié est rappelée dans ce tableau.

		Syntaxique (signifiant)	Sémantique (signifié)
$\mathcal{DOC}$	Document atomique $\mathcal{A}$	$\mathcal{A}_{doc} \in \mathcal{DS}_{doc}$	$\mathcal{A}_{inf} \in \mathcal{DS}_{inf}$
	Document structuré $\mathcal{DS}$	$\mathcal{DS}_{doc}$	$\mathcal{DS}_{inf}$
	Chemin de lecture $\mathcal{CH}$	$\mathcal{CH}_{doc}$	$\mathcal{CH}_{inf}$
	Hyperdocument $\mathcal{HD}$	$\mathcal{HD}_{doc}$	$\mathcal{HD}_{inf}$
$\mathcal{DOCC}$	Document atomique $\mathcal{AC}$	$\mathcal{AC}_{doc}$	$\mathcal{AC}_{inf}$
	Document structuré $\mathcal{DSC}$	$\mathcal{DSC}_{doc}$	$\mathcal{DSC}_{inf}$
	Chemin de lecture $\mathcal{CHC}$	$\mathcal{CHC}_{doc}$	$\mathcal{CHC}_{inf}$
	Hyperdocument $\mathcal{HDC}$	$\mathcal{HDC}_{doc}$	$\mathcal{HDC}_{inf}$
$\mathcal{R}$	Relations	$\mathcal{R}_{doc}$	$\mathcal{R}_{inf}$
	de composition	$\mathcal{R}_{compdoc}$	$\mathcal{R}_{compinf}$
	de cheminement	$\mathcal{R}_{chemdoc}$	$\mathcal{R}_{cheminf}$
	de référence	$\mathcal{R}_{refdoc}$	$\mathcal{R}_{refinf}$

FIG. 7.1 – Les composants du modèle d'hyperdocuments $\mathcal{HDOCC}$.

Dans la suite, nous utiliserons le terme générique *document* pour désigner les éléments de $\mathcal{A}$, $\mathcal{DS}$, $\mathcal{HD}$ ou $\mathcal{CH}$, comme introduit dans la définition 1 (cf. chapitre 6). Nous définissons l'ensemble des documents comme suit :

Définition 17 *L'ensemble des* ***documents*** doc *est appelé* $\mathcal{DOC}$ *:*

$$\mathcal{DOC} = \mathcal{A} \cup \mathcal{DS} \cup \mathcal{CH} \cup \mathcal{HD} \tag{7.2}$$

L'ensemble des **documents en contexte** est défini de la manière suivante :

Définition 18 *L'ensemble des* ***documents en contexte*** $docc$ *est appelé* $\mathcal{DOCC}$ *:*

$$\mathcal{DOCC} = \mathcal{AC} \cup \mathcal{DSC} \cup \mathcal{CHC} \cup \mathcal{HDC} \tag{7.3}$$

Nous décrivons dans les sections suivantes l'aspect signifiant du modèle d'hyperdocuments en contexte en simplifiant l'écriture comme dans le chapitre 6 (par exemple, $\mathcal{HDOCC}_{doc}$ est simplifié en $\mathcal{HDOCC}$).

7.2 Documents atomiques $\mathcal{A}$

La modélisation des documents atomiques est similaire à celle d'un modèle de RI classique, c'est-à-dire de manière indépendante et non structurée. Le modèle d'hyperdocuments est basé sur l'ensemble des documents atomiques $\mathcal{A}$. On rappelle la définition 3 d'un document atomique :

Définition 3 ***Document atomique :*** *un document atomique a est un fragment de texte, un ensemble insécable de phrases délimité dans le document. Un document atomique est porteur d'une signification qui se suffit à elle-même, indépendamment de tout contexte.*

Nous définissons un document atomique comme étant un cas particulier de document structuré ($\mathcal{A} \subseteq \mathcal{DS}$). Nous y reviendrons au moment de la définition des documents structurés. Dans la suite, nous continuerons cependant, pour plus de clarté, à distinguer les éléments de $\mathcal{A}$ des autres éléments de $\mathcal{DS}$.

7.3 Composition et documents structurés

La relation de composition $\mathcal{R}_{comp}$ est une relation binaire interne sur l'ensemble $\mathcal{DS}$:

$$\mathcal{R}_{comp} \times \mathcal{DS} \cup \mathcal{DS} \tag{7.4}$$

La relation de composition peut être définie entre deux documents structurés ds_i et ds_j. On dit alors que ds_i est le père de ds_j, et réciproquement ds_j est le fils de ds_i. La relation de composition construit une structure arborescente des documents, avec les documents structurés comme nœuds et les documents atomiques comme feuilles. Alors, un document atomique est un document structuré qui ne comporte aucun fils : il s'agit d'un arbre réduit à sa racine.

7.3.1 Propriétés de la relation de composition

En plus des contraintes liées à la structure arborescente des documents structurés, on précise que la relation de composition est une relation anti-réflexive, transitive et asymétrique sur l'ensemble $\mathcal{DS}$:

Anti-réflexivité : un document ne peut pas être composé de lui-même.

Transitivité : si un document ds_i est composé d'un document ds_j lui-même composé de ds_k, alors il existe aussi une relation de composition entre ds_i et ds_k. On appellera ds_j un *composant direct* de ds_i (un fils), par opposition à ds_k, qui est un *descendant*.

Asymétrie : si un document doc_i est composé d'un document doc_j, alors doc_j ne peut pas être composé de doc_i.

7.3.2 Documents structurés $\mathcal{DS}$

a) Définition

Les éléments que nous avons déjà définis nous permettent de décrire les documents structurés $\mathcal{DS}$. Un document structuré ds est un nœud d'un arbre n-aire dont les feuilles sont des documents atomiques. On le définit comme un triplet <A, DS, R_{comp}> tel que :

$$ds =< A, DS, R_{comp} > \in \mathcal{DS}, \quad Avec \begin{cases} A = \{a_i\} \subseteq \mathcal{A} \\ DS = \{ds_i\} \subseteq \mathcal{DS} \\ R_{comp} \subseteq \mathcal{R}_{comp} \end{cases} \tag{7.5}$$

L'ensemble $A \subseteq \mathcal{A}$ est l'ensemble des documents atomiques a_i qui sont des composants directs de ds (ses fils, dans la hiérarchie). Ce sont des feuilles de l'arborescence qui ne comportent donc aucun fils. L'ensemble $DS \subseteq \mathcal{DS}$ est l'ensemble des documents structurés ds_i qui sont composants directs de ds. R_{comp} est la restriction de $\mathcal{R}_{comp}$ aux relations de composition qui existent entre le document structuré et ses composants directs (documents atomiques et documents structurés).

b) Feuilles et nœuds

Le modèle de documents structurés décrit uniquement les fils directs. Cela ne permet pas, pour un document structuré donné, de manipuler l'ensemble de ses descendants (par application de la transitivité de la relation de composition). Nous définissons les opérateurs d'accès à l'ensemble des documents atomiques et structurés qui sont composants directs et indirects d'un document structuré donné. Nous appelons ces opérateurs '$feuilles$' et '***nœuds***', qui sont définis récursivement comme suit :

$$\begin{array}{l} feuilles : \mathcal{DS} \longrightarrow 2^{\mathcal{A}} \\ \forall\, ds \in \mathcal{DS},\ feuilles(ds) = A \cup (\bigcup_{ds_i \in DS} feuilles(ds_i)) \end{array} \tag{7.6}$$

$$\begin{array}{l} \textit{nœuds} : \mathcal{DS} \longrightarrow 2^{\mathcal{DS}} \\ \forall\, ds \in \mathcal{DS},\ \textit{nœuds}(ds) = DS \cup (\bigcup_{ds_i \in DS} \textit{nœuds}(ds_i)) \end{array} \tag{7.7}$$

Nous définissons également l'opérateur '***père(ds)***' permettant d'accéder à l'ascendant direct (le père, dans l'arborescence) d'un document ds :

$$\begin{array}{l} \textit{père} : \mathcal{DS} \longrightarrow \mathcal{DS} \\ \forall\, ds_i \in \mathcal{DS},\ \textit{père}(ds_i) = ds_j \mid (ds_j, ds_i) \in \mathcal{R}_{comp} \end{array} \tag{7.8}$$

On note que l'opérateur '$\textit{père}(ds)$' n'est pas défini pour les documents structurés qui sont à la racine de l'arborescence, et que tous les documents qui ne sont pas des racines possèdent un et un seul père.

c) Un document structuré particulier : le document atomique

Nous avons défini un document atomique comme étant un cas particulier de document structuré. La définition des documents structurés que nous avons donnée nous permet de formaliser les documents atomiques de la manière suivante :

$$< A, DS, R_{comp} > \in \mathcal{A}, \quad Avec \begin{cases} A = \{a\} \\ DS = \emptyset \\ R_{comp} = \emptyset \end{cases} \tag{7.9}$$

d) Racine d'un document structuré

L'arborescence d'un document structuré possède un élément racine, qui est nécessairement un document structuré. Nous définissons la racine d'un document structuré donné :

Définition 19 ***Racine :*** *un document structuré ds est appelé* ***document racine*** *s'il n'admet aucun prédécesseur selon la relation de composition :*

$$\forall\, ds_i \in \mathcal{DS}, (\textit{est-racine}(ds_i) \Leftrightarrow \neg\exists\, ds_j \in \mathcal{DS} \mid (ds_j, ds_i) \in \mathcal{R}_{comp}) \tag{7.10}$$

L'opérateur d'accès associé à l'élément racine d'un document s'appelle '$racine$', nous le définissons comme suit :

$$racine : \mathcal{DS} \longrightarrow \mathcal{DS}$$
$$\forall\, ds \in \mathcal{DS}, racine(ds) = \begin{cases} \textit{père}(ds) \textit{ si est-racine}(\textit{père}(ds)) \\ racine(\textit{père}(ds)) \textit{ sinon} \end{cases} \tag{7.11}$$

Enfin, à chaque document atomique et à chaque document structuré, correspond un et un seul élément racine :

$$\forall\, ds_i \in \mathcal{DS}, \exists_1\, ds_j \in \mathcal{DS} \mid ds_j = racine(ds_i) \tag{7.12}$$

e) Granularité

Comme nous l'avons introduit dans la section 6.8, la granularité $gran$ est définie sur l'ensemble des documents $\mathcal{DOCC}$:

$$\forall\, ds \in \mathcal{DS}, gran = \textit{granularité}(ds) \tag{7.13}$$

$$\textit{granularité} : \mathcal{DOCC} \longrightarrow \mathbb{R} \tag{7.14}$$

La granularité représente le degré de composition d'un document par rapport aux autres documents du corpus. La fonction "*granularité*" est basée sur les fonctions "*taille*" et "*hauteur*", que nous détaillons ultérieurement à la phase d'indexation (cf. chapitre 8).

f) Contraintes sur les documents structurés

Il existe un certain nombre de contraintes inhérentes à la structure arborescente des documents. Par exemple, les documents atomiques ne peuvent être que des feuilles d'un arbre et possèdent obligatoirement un et un seul père, et de leur côté les documents structurés ne peuvent être que des nœuds non feuilles et ne peuvent pas avoir simultanément $A = \emptyset$ et $DS = \emptyset$.

7.4 Cheminement et hyperdocuments

La relation de cheminement $\mathcal{R}_{chem}$ est une relation binaire interne sur l'ensemble des documents atomiques $\mathcal{A}$:

$$\mathcal{R}_{chem} \subseteq \mathcal{A} \times \mathcal{A} \tag{7.15}$$

La relation de cheminement représente un sens de lecture potentiel entre deux documents atomiques a_i et a_j. On dit alors que a_j est le successeur de a_i, et réciproquement a_i est le prédécesseur de a_j. La relation de cheminement est anti-réflexive : on ne peut pas cheminer d'un document à lui-même.

7.4.1 Chemins de lecture $\mathcal{CH}$

La relation de cheminement permet de construire des chemins de lecture. A chaque document structuré ds_i est associé un ensemble de chemins de lectures $ch_i^k \in \mathcal{CH}$. Du point de vue du cheminement, les feuilles d'un même document structuré construisent un graphe orienté, avec les documents atomiques a_j comme nœuds et les relations de cheminement R_{chem} comme arcs. Un chemin de lecture est donc un chemin au sens des graphes sur les documents atomiques d'un document structuré. D'autre part, le fait de construire plusieurs chemins de lecture pour un même document structuré revient à considérer les documents structurés du point de vue des hyperdocuments. L'ensemble des lectures potentielles d'un document structuré vu comme un hyperdocument peut se représenter avec un ensemble de chemins de lecture.

a) Définition

Un chemin de lecture $ch \in \mathcal{CH}$ est défini sur l'ensemble A des documents atomiques du document structuré ds auquel il est associé. Un chemin est formalisé par un triplet $<A, Arcs, R_{chem}>$ tel que :

$$\begin{array}{l} ds = < A, DS, R_{comp} > \in \mathcal{DS}, \\ ch^k = < A_k, Arcs_k, R^k_{chem} > \in \mathcal{CH}, \end{array} \quad Avec \begin{cases} A_k = \{a_j\} \subseteq A \\ Arcs_k = [arc_1, arc_2, ...arc_n] \\ R^k_{chem} \subseteq R_{chem} \subseteq \mathcal{R}_{chem} \end{cases} \tag{7.16}$$

L'ensemble A_k est l'ensemble des documents atomiques parcourus par le chemin de lecture, parmi les documents atomiques A du document structuré. L'enchaînement des $a_j \in A_k$ est défini par la séquence d'arcs $Arcs_k$. Enfin, R^k_{chem} est une restriction de la relation de cheminement R_{chem} aux documents atomiques de A (et R_{chem} est lui-même une restriction de $\mathcal{R}_{chem}$ aux nœuds de A). L'ensemble de relations R^k_{chem} organise les nœuds de A_k en un chemin sans cycle.

b) Arcs

Un arc représente la possibilité de naviguer au cours de la lecture d'un document atomique source vers un document atomique destination. L'existence d'un arc est conditionnée par l'existence d'une relation de cheminement entre le document source et le document destination (le premier étant le prédécesseur du second). Un arc est un triplet $< a_{src}, a_{dest}, \beta >$ (document source, document destination et coefficient de rupture sémantique) tel que :

$$\forall\, arc \in Arcs_k, arc\ = (a_{src} \in A_k, a_{dest} \in A_k, \beta) \mid (a_{src}, a_{dest}) \in R^k_{chem} \tag{7.17}$$

β est le coefficient de rupture sémantique entre le nœud source a_{src} et le nœud destination a_{dest} d'une relation de cheminement. Ce coefficient exprime dans quelle mesure la continuité du discours est assurée au cours du cheminement. Un coefficient de rupture β élevé signifie qu'il y a une rupture dans la continuité du discours, entre les deux nœuds (cf. section 6.9).

c) Document initial/final

Un chemin de lecture possède un et un seul "*document initial*", noté a^{init}, qui est le document atomique source du premier arc. Nous définissons le document initial d'un chemin de lecture donné :

Définition 20 ***Document initial :*** *un document atomique a est appelé* ***document initial*** a^{init} *d'un chemin de lecture donné s'il n'admet aucun prédécesseur (sur ce chemin) selon la relation de cheminement :*

$$\begin{array}{l} \forall\, ch^k \in \mathcal{CH}, \\ \forall\, a_i \in \mathcal{A}, \textit{est-initial}(a_i) \Leftrightarrow \neg\exists\, a_j \in \mathcal{A} \mid (a_j, a_i) \in \mathcal{R}^k_{chem} \end{array} \tag{7.18}$$

A chaque chemin de lecture correspond un et un seul document initial :

$$\forall\, ch^k \in \mathcal{CH}, \exists_1\, a_i \in A_k \mid \textit{est-initial}(a_i) \tag{7.19}$$

De la même manière, nous définissons le "*document final*", noté a^{final}, qui est le document atomique destination du dernier arc. A chaque chemin de lecture correspond également un et un seul document final.

d) Granularité

Comme présenté dans la section 6.9, on définit la granularité $gran$ d'un chemin de lecture, dont le calcul se base sur la fonction *granularité*. La granularité $gran$ est relative à la taille d'un chemin par rapport aux autres chemins du corpus.

e) Chemin de lecture standard

On rappelle que le chemin de lecture standard d'un document structuré est un chemin qui passe une fois et une seule par chacun de ses documents atomiques feuilles (cf. section 6.9.6). Il existe un et un seul chemin de lecture standard par document structuré, par opposition aux multiples chemins déambulatoires. On appelle ce chemin ch^s :

$$\begin{array}{l}\text{Chemin de lecture standard :} \quad \forall\, ds =< A, DS, R_{comp} > \in \mathcal{DS}, \\ \qquad\qquad\qquad\qquad\quad \exists_1\, ch^s =< A^s, Arcs^s, R^s_{chem} > \in \mathcal{CH} \mid A = A_s\end{array} \tag{7.20}$$

Un chemin de lecture standard doit passer une fois et une seule par chacun des documents de A_s, sans constituer de cycle et sans passer deux fois par un même nœud : il s'agit de la définition d'un *chemin hamiltonien* dans un graphe orienté.

f) Contraintes sur les chemins de lecture

La relation de cheminement se définit entre deux documents atomiques qui sont feuilles d'un même document structuré :

$$\forall\, (a_i, a_j) \in \mathcal{R}_{chem}, racine(a_i) = racine(a_j) \tag{7.21}$$

On remarque que la relation n'est pas restreinte aux fils directs d'un même document structuré. Cela signifie qu'une relation de cheminement peut être définie entre deux documents atomiques d'un même document structuré appartenant à deux niveaux de granularité distincts.

Les autres contraintes sont exprimées par le fait qu'un chemin de lecture est un chemin *hamiltonien* : un chemin linéaire, sans cycle, qui ne repasse pas deux fois par le même nœud.

Le même chemin ne peut pas parcourir deux fois le même document atomique, mais par contre il peut exister plusieurs chemins distincts parcourant le même document atomique. Il n'y a donc pas de disjonction entre les chemins de lecture "déambulatoires". Les ensembles de documents atomiques A_k et A_p parcourus par deux chemins distincts ch^k et ch^p sur un même document structuré ds peuvent éventuellement comporter des éléments en commun ($A_k \cap A_p \neq \emptyset$), voire être identiques ($A_k = A_p$).

g) Contraintes sur les chemins de lecture standard

Les chemins de lecture standard nécessitent la définition de contraintes supplémentaires. Nous avons défini l'existence d'un et un seul chemin de lecture standard par document struc-

turé. Un chemin ch^s doit parcourir la totalité des documents atomiques du document structuré associé ($A = A_k$). De plus, contrairement aux cas des chemins de lecture déambulatoire, il y a disjonction des chemins de lecture standard. Il existe un cas particulier où deux chemins de lecture standard ch_i^s et ch_j^s peuvent parcourir deux sous-ensemble de documents non disjoints : quand le document structuré ds_i associé à ch_i^s est un descendant du document structuré ds_j associé à ch_j^s. La contrainte de la disjonction se formalise alors de la manière suivante :

$$\begin{array}{ll} \text{Disjonction :} & \forall\, ch_i^s, ch_j^s \in \mathcal{CH}, \\ & i \neq j \Rightarrow A_i \cap A_j = \emptyset \\ & \vee\, ds_i \in \textit{nœuds}(ds_j) \\ & \vee\, ds_j \in \textit{nœuds}(ds_i) \end{array} \tag{7.22}$$

7.4.2 Hyperdocuments $\mathcal{HD}$

La description des chemins de lecture associés à des documents structurés nous amène à la définition de l'ensemble des hyperdocuments $\mathcal{HD}$. Un hyperdocument hd, élément de $\mathcal{HD}$, est un triplet $< ds, Chemins, R_{chem} >$ tel que :

$$hd = < ds, Chemins, R_{chem} > \in \mathcal{HD}, \quad Avec \begin{cases} ds = < A, DS, R_{comp} > \in \mathcal{DS} \\ Chemins = \{ch^k\} \subseteq \mathcal{CH} \\ R_{chem} \subseteq \mathcal{R}_{chem} \end{cases} \tag{7.23}$$

Un hyperdocument exprime une association entre un document structuré ds et un ensemble de chemins de lecture $Chemins$ sur les documents atomiques de ds. Les chemins vérifient donc : $\forall\, ch^k \in Chemins, A_k \subseteq A$. Enfin, R_{chem} est la restriction de $\mathcal{R}_{chem}$ aux documents atomiques du document structuré ds, qui organise ses documents atomiques en un chemin de lecture standard et zéro ou plusieurs chemins de lecture déambulatoires.

7.5 Référence et contexte

La relation de référence est une relation binaire sur l'ensemble des éléments de $\mathcal{DOC}$:

$$\mathcal{R}_{ref} \subseteq \mathcal{DOC} \times \mathcal{DOC} \tag{7.24}$$

La relation de référence peut être définie entre deux documents doc_i et doc_j (deux documents atomiques, deux documents structurés ou deux chemins de lecture, comme présenté dans la section 6.10). On dit alors que doc_i est le référenceur de doc_j, et réciproquement doc_j est le référencé de doc_i.

7.5.1 Propriétés de la relation de référence

La relation de référence est une relation binaire anti-réflexive, non transitive et non symétrique sur l'ensemble $\mathcal{DOC}$:

Anti-réflexivité : un document ne peut pas se référencer lui-même.

Non transitivité : la relation de référence décrit les références directes d'un document.

Non symétrie : si un document doc_i référence un document doc_j, alors doc_i peut éventuellement être référencé par doc_j.

7.5.2 Contraintes sur la relation de référence

Une relation de référence entre les documents structurés ds_i et ds_j signifie qu'un des documents atomiques a_p de ds_i est en relation avec un des documents atomiques a_q de ds_j. Le document a_p (respectivement a_q) fait partie des feuilles de ds_i (respectivement de ds_j).

$$\begin{array}{l}\forall\, ds_i, ds_j \in \mathcal{DS}, \\ (ds_i, ds_j) \in \mathcal{R}_{ref} \Leftrightarrow \exists\, a_p \in feuilles(ds_i), a_q \in feuilles(ds_j) \mid (a_p, a_q) \in \mathcal{R}_{ref}\end{array} \quad (7.25)$$

Contrairement à la relation de composition ou de cheminement, la relation de référence n'a aucune contrainte de linéarité, de recouvrement des composants d'un document (les nœuds du graphe des documents comportent 0, 1 ou plusieurs relations de référence) ou de construction d'une structure arborescente, sans cycle, etc.

Contrairement à la relation de cheminement, la relation de référence ne peut pas être définie entre deux documents atomiques (ou deux documents structurés) ayant une racine commune. Cette contrainte se formalise de la manière suivante :

$$\forall\, ds_i, ds_j \in \mathcal{R}_{ref}, racine(ds_i) \neq racine(ds_j) \quad (7.26)$$

7.5.3 Les documents en contexte $\mathcal{DOCC}$

La relation de référence $\mathcal{R}_{ref}$ permet de décrire le contexte des documents de $\mathcal{DOC}$ de différents types. A chaque document de $\mathcal{DS}$ (respectivement de $\mathcal{CH}$, de $\mathcal{HD}$) correspond un document en contexte de $\mathcal{DSC}$ (respectivement de $\mathcal{CHC}$, de $\mathcal{HDC}$).

$$\begin{array}{c}\text{Mise en contexte : } \mathcal{DOC} \rightarrow \mathcal{DOCC} \\ \mathcal{A} \rightarrow \mathcal{AC} \\ \mathcal{DS} \rightarrow \mathcal{DSC} \\ \mathcal{CH} \rightarrow \mathcal{CHC}\end{array} \quad (7.27)$$

Un document en contexte est un triplet $< doc, MI, IA >$ défini comme suit :

$$\forall\, doc \in \mathcal{DOC}, \exists_1\, docc = < doc, MI, IA > \in \mathcal{DOCC} \quad (7.28)$$

On appelle MI la méta-information du document doc, et IA son information accessible. Par exemple, le contexte d'un document atomique a_i est composé de la méta-information MI_i et de l'information accessible IA_i. MI_i désigne l'ensemble des documents atomiques qui référencent a_i, et IA_i désigne l'ensemble des documents atomiques qui sont accessibles par navigation à partir de a_i. Ces deux notions sont symétriques :

$$\begin{array}{ll} \forall\, a_i \in \mathcal{A}, & MI_i = \{(a_j, \beta) \mid a_j, a_i \in \mathcal{A} \wedge (a_j, a_i) \in \mathcal{R}_{ref}\} \\ & IA_i = \{(a_j, \beta) \mid a_j, a_i \in \mathcal{A} \wedge (a_i, a_j) \in \mathcal{R}_{ref}\} \end{array} \tag{7.29}$$

On retrouve β qui représente le coût de navigation pour activer le lien hypertexte (cf. chapitre 6). Le contexte existe aussi au niveau de granularité des documents structurés et des chemins de lecture et se base sur la relation de référence entre les documents atomiques. Par exemple, la méta-information d'un document structuré ds_i est composé de l'ensemble des ds_j qui le référencent, c'est-à-dire de l'ensemble des ds_j dont un des éléments feuilles a_j référence un des éléments feuilles de ds_i. De manière similaire, la méta-information d'un chemin de lecture ch^k est l'ensemble des chemins dont au moins un des documents atomiques parcourus référence un des documents atomiques de ch^k.

7.5.4 Les hyperdocuments en contexte $\mathcal{HDOCC}$

Enfin, la description d'un ensemble de chemins de lecture en contexte associés à un document structuré en contexte permet de définir l'ensemble des hyperdocuments en contexte $\mathcal{HDC}$. A la manière de l'ensemble des hyperdocuments en contexte, on définit un élément hdc de $\mathcal{HDC}$ comme un triplet $< dsc, CheminsC, R_{chem} >$ tel que :

$$\begin{array}{l} \forall\, dsc \in \mathcal{DSC}, \\ \exists_1\, hdc = < dsc, CheminsC, R_{chem} > \in \mathcal{HDC}, \\ \qquad\qquad Avec \left\{ \begin{array}{l} dsc = < ds, MI, IA > \in \mathcal{DSC} \\ CheminsC = \{chc^k\} \subseteq \mathcal{CHC} \\ R_{chem} \subseteq \mathcal{R}_{chem} \end{array} \right. \end{array} \tag{7.30}$$

La définition des hyperdocuments n'introduit pas de nouveau concept par rapport aux documents structurés ou aux chemins de lecture. Son rôle est d'associer documents structurés et chemins de lecture.

7.6 Le modèle d'hyperdocuments : signifié

Dans le chapitre 6, la description du modèle de documents débutait par une discussion à propos de la symétrie (et non bijection) entre le niveau du signifiant et celui du signifié, et à propos de l'utilisation de la pragmatique (du contexte) pour la désambiguïsation. Maintenant que nous avons décrit le modèle du signifiant, nous revenons sur cette problématique.

7.6.1 Symétrie signifiant/signifié

Avec la formalisation du processus de transmission de l'information, nous avons axé notre modèle de RI selon les trois niveaux de description de l'information : signifiant, signifié et pragmatique (cf. section 6.3.1). Le passage du niveau du signifié au niveau du signifiant, lors de l'étape d'encodage, construit la représentation des idées de l'auteur dans le modèle du signifiant que nous venons de décrire. Nous faisons l'hypothèse que l'auteur cherche à organiser ses documents en utilisant le même type de structure que pour la représentation mentale de ses idées. Ainsi, le modèle du signifié $\mathcal{HDOCC}_{inf}$ est symétrique du modèle du signifiant $\mathcal{HDOCC}_{doc}$, comme présenté dans le chapitre 6. Du fait de la symétrie entre les deux niveaux, nous n'entrons pas dans les détails de la description de $\mathcal{HDOCC}_{inf}$.

7.6.2 Passage du signifiant au signifié

Le passage du signifié au signifiant se déroule, dans notre modèle, à la phase d'encodage. L'opération inverse est le décodage. Nous formalisons ces opérations par les fonctions $encodage$ et *décodage*, qui se décomposent en fait en plusieurs fonction d'encodage pour chaque ensemble du modèle.

$$\begin{array}{l} encodage : \mathcal{HDOC}_{inf} \rightarrow \mathcal{HDOC}_{doc} \\ \textit{décodage} : \mathcal{HDOC}_{inf} \rightarrow \mathcal{HDOC}_{doc} \end{array} \tag{7.31}$$

Comme il n'y a pas de bijection entre $\mathcal{HDOC}_{doc}$ et $\mathcal{HDOC}_{inf}$, la fonction d'encodage n'est pas bijective. En effet, un élément du signifié peut être encodé en plusieurs éléments du signifiant, et inversement :

$$\begin{array}{l} \forall\ doc_i \in \mathcal{HDOC}_{inf}, \exists\ \textit{1 ou plusieurs}\ \ doc_j \in \mathcal{HDOC}_{doc} \mid doc_j = encode(doc_i) \\ \forall\ doc_i \in \mathcal{HDOC}_{doc}, \exists\ \textit{1 ou plusieurs}\ \ doc_j \in \mathcal{HDOC}_{inf} \mid doc_j = \textit{décode}(doc_i) \end{array} \tag{7.32}$$

Plus concrètement, du point de vue de la Recherche d'Information, cela signifie que, pour un document du corpus, il peut y avoir ambiguïté sur l'information représentée.

7.6.3 Désambiguïsation

Comme présenté dans la figure 6.11 (cf. section 6.5), la pragmatique permet une désambiguïsation des éléments du signifiant par rapport au niveau du signifié. Formellement, cela est réalisé par les fonctions d'encodage et de décodage entre les éléments de $\mathcal{HDOCC}_{doc}$ et de $\mathcal{HDOCC}_{inf}$:

$$\begin{array}{l} encodage_{prag} : \mathcal{HDOCC}_{inf} \rightarrow \mathcal{HDOCC}_{doc} \\ \textit{décodage}_{prag} : \mathcal{HDOCC}_{inf} \rightarrow \mathcal{HDOCC}_{doc} \end{array} \tag{7.33}$$

En effet, l'apport de la pragmatique est de lever l'ambiguïté dans un sens. Cela signifie que, pour chaque élément de $\mathcal{HDOCC}_{doc}$, il existe un et un seul élément de $\mathcal{HDOCC}_{inf}$. Par

contre, il reste toujours une ambiguïté au niveau du signifié : à chaque élément de $\mathcal{HDOCC}_{inf}$ peuvent correspondre un ou plusieurs éléments de $\mathcal{HDOCC}_{doc}$:

$$\begin{array}{l} \forall\, doc_i \in \mathcal{HDOCC}_{inf}, \exists \textit{ 1 ou plusieurs } doc_j \in \mathcal{HDOCC}_{doc} \mid doc_j = encode(doc_i) \\ \forall\, doc_i \in \mathcal{HDOCC}_{doc}, \exists_1\, doc_j \in \mathcal{HDOCC}_{inf} \mid doc_j = \textit{décode}(doc_i) \end{array} \tag{7.34}$$

L'utilisation du contexte est intéressante pour la Recherche d'Information. Il permet de résoudre des problèmes de polysémie mais ne permet pas de résoudre les problèmes de synonymie.

7.7 Conclusion

Dans ce chapitre, nous avons présenté le modèle d'hyperdocuments, qui se situe au niveau du signifiant. Le modèle d'hyperdocuments intègre les quatre aspects importants de la description de l'information sur le Web : le **contenu**, la **composition** (relation de composition), la **lecture** linéaire ou déambulatoire (relation de cheminement) et le **contexte** (relation de référence). Celui-ci est composé de l'espace d'information référençant un document et de l'espace d'information accessible à partir d'un document.

L'objectif du modèle de RI est d'extraire une approximation du contenu sémantique des documents pour permettre, à la phase d'interrogation, de retrouver les documents pertinents pour l'utilisateur. Il s'agit donc de considérer l'aspect signifié, tel que nous l'avons décrit dans ce chapitre. Notre modèle d'hyperdocuments considère la structure du Web : il est donc nécessaire de prendre en compte l'impact de cette structure au moment de l'indexation (l'extraction du contenu sémantique des documents).

Le chapitre suivant est consacré à la description de ce processus d'indexation, et de la structure des index qui sont produits. Le processus ainsi que les index doivent intégrer la structure du Web, avec la composition, les chemins de lecture et le contexte.

Chapitre 8

Indexation et interrogation structurées

8.1 Processus d'indexation : extraction du signifié

Ce chapitre présente l'indexation des documents du Web, intégrant les aspects du contenu, de la structure logique, des chemins de lecture et du contexte. Le processus d'indexation se base sur le modèle d'hyperdocuments en contexte présenté dans le chapitre 7. L'objectif est de construire les index des hyperdocuments en vue de l'interrogation structurée décrite dans les sections suivantes (cf. 8.9).

Comme nous l'avons décrit dans le chapitre 6, l'indexation est basée sur le contenu des documents, mais doit également prendre en compte la structure des documents, c'est-à-dire d'intégrer l'impact de la relation de composition sur l'indexation. Il est ensuite important de considérer les caractéristiques hypertextuelles du Web, et en particulier l'impact de la relation de cheminement, en réalisant l'indexation d'un hyperdocument donné comme une *simulation de lecture*, ou plutôt **des** simulations de lecture (un hyperdocument comporte un choix de plusieurs chemins de lecture). Enfin, une information ne prend tout son sens que si elle est placée dans un contexte : il nous faut donc prendre en compte le contexte des hyperdocuments, c'est-à-dire intégrer l'impact de la relation de référence à l'indexation.

8.1.1 Etapes de l'indexation

L'indexation vue comme une simulation de lecture des hyperdocuments pour en extraire le signifié respecte les principes présentés dans le chapitre 6 et suit les étapes proposées :

Contenu : l'indexation des documents atomiques, c'est-à-dire l'extraction de l'information atomique (cf. section 8.2).

Composition : l'indexation des documents structurés (cf. sections 8.3). Cette étape se base sur les index extraits au cours de l'étape initiale, c'est-à-dire les index des documents atomiques.

Cheminement : l'indexation des chemins de lecture (cf. section 8.4). Cette étape se base également sur les index extraits lors de la première étape. On indexe donc les chemins en utilisant l'index des documents atomiques parcourus.

Contexte : l'indexation du contexte des différents types de documents (la popularité, le rayonnement, la méta-information et l'information accessible, cf. section 8.6).

8.1.2 Composants de l'index

Nous utilisons le modèle vectoriel (VSM) de Salton [Salton71] qui a fait ses preuves pour l'indexation et l'interrogation de documents atomiques. Dans ce modèle de RI, un document est représenté par un vecteur de termes pondérés.

Le tableau 8.1 présente les index $\vec{a}$, $\vec{ds}$, et $\vec{ch}$ correspondant à chaque type de document. Chaque index est associé à une valeur $gran$, qui représente la granularité du document. Le modèle vectoriel est utilisé pour l'indexation de tous les types de documents.

	Signifiant	Signifié	Index
$\mathcal{DOC}$	Document atomique $\mathcal{A}_{doc}$	Information atomique $\mathcal{A}_{inf}$	Index atomique $\vec{a}$, gran
	Document structuré $\mathcal{DS}_{doc}$	Information structurée $\mathcal{DS}_{inf}$	Index structuré $\vec{ds}$, gran
	Chemin de lecture $\mathcal{CH}_{doc}$	Cheminement $\mathcal{CH}_{inf}$	Index de chemin $\vec{ch}$, gran
	Hyperdocument $\mathcal{HD}_{doc}$	Hyperinformation $\mathcal{HD}_{inf}$	Index d'hyperdocument <$\vec{ds}$, gran, { $\vec{ch}$, gran }>

FIG. 8.1 – Indexation : extraction des index de documents.

Enfin, l'indexation du contexte se base aussi sur le modèle vectoriel. Le tableau 8.2 présente les index du contexte, c'est-à-dire les deux vecteurs *méta-info* et *info-acc* qui représentent, respectivement, la méta-information et l'information accessible de chaque document $doc \in \mathcal{DOC}$. A ces deux vecteurs sont associées les deux valeurs aut et ray, qui représentent, respectivement, l'autorité et le rayonnement du document.

Signifiant	Index
Méta-information MI	*méta-info*, $aut \in [0..1]$
Information Accessible IA	*info-acc*, $ray \in [0..1]$

FIG. 8.2 – Indexation : mise en contexte (pragmatique).

Nous présentons dans les sections suivantes l'extraction de ces index, avec l'indexation des documents atomiques dans la section 8.2, des documents structurés dans la section 8.3, des chemins de lecture dans la section 8.4 et du contexte dans la section 8.6.

8.2 Indexation des documents atomiques a

8.2.1 Modèle vectoriel

Un document atomique a_i est représenté par un vecteur dans un espace à n dimensions, n étant le nombre de termes du langage d'indexation :

$$\vec{a_i} \ = \ (w_{i1}, \ w_{i2} \ ... \ w_{ij} \ ... \ w_{in}) \tag{8.1}$$

8.2.2 Pondération

L'indexation d'un document atomique a en un vecteur $\vec{a}$ consiste à extraire les termes t_j représentatifs des documents et à leur affecter une pondération $w_{ij} \in [0, 1]$. Cette pondération représente l'importance d'un terme t_j dans un document a_i. Elle est calculée à l'aide d'une fonction de pondération classique, de type $tf * idf$ (*term frequency, invert document frequency*).

Le calcul de la pondération w_{ij} d'un terme t_j par rapport à un document atomique a_i combine l'évaluation de l'importance du terme relativement au document (le tf, appelé "*pouvoir résumant*" du terme) avec l'évaluation de l'importance du terme dans le corpus (l'idf, appelé "*pouvoir discriminant*" du terme). Ainsi, un terme qui obtient une pondération élevée pour un document est un terme qui est à la fois important dans le document (c'est-à-dire qu'il a un fort pouvoir résumant pour le document) et peu important dans le reste du corpus (c'est-à-dire qu'il est discriminant pour le document).

La fonction de pondération se base sur les informations suivantes :

Fréquence locale : tf_{ij} (*term frequency*) est le nombre d'occurrences du terme t_j dans le document a_i.

Fréquence documentaire : df_j (*document frequency*) est le nombre de documents dans lesquels le terme t_j apparaît :

$$df_j = |\{a_i \in \mathcal{A} \mid t_j \in a_i\}| \tag{8.2}$$

Taille du corpus : N_{doc} est le nombre de documents du corpus. Ici, le corpus est l'ensemble des documents atomiques, et N_{doc} est donc le nombre de documents atomiques :

$$N_{doc} = |\mathcal{A}| \tag{8.3}$$

a) Pouvoir résumant : *"Résum"*

La première étape de la pondération consiste à calculer *Résum*(a_i, t_j), la composante normalisée relative au tf (*fréquence locale*), qui exprime dans quelle mesure le terme t_j représente l'information contenue dans le document a_i. On l'appelle *Résum*, le *pouvoir résumant* du terme pour le document, calculée par la formule suivante, bien connue en RI :

$$\textit{Résum}(a_i, t_j) \ = \ \log_2(tf_{ij}) + 1 \tag{8.4}$$

b) Pouvoir discriminant : "*Discr*"

La deuxième étape consiste à calculer *Discr*(t_j), la composante normalisée relative au df (*fréquence documentaire*), qui exprime dans quelle mesure le terme t_j caractérise a_i par rapport au reste du corpus $\mathcal{A}$ des documents atomiques. On l'appelle *Discr*, le *pouvoir discriminant* du terme dans le corpus :

$$\textit{Discr}(t_j) \;=\; \frac{1}{\log_2(\frac{N_{doc}}{df_j})} \tag{8.5}$$

c) Combinaison : $tf * idf$

La combinaison des deux critères *Résum* et *Discr* permet de calculer la pondération w_{ij} :

$$w_{ij} \;=\; \textit{Résum}(a_i, t_j) \;*\; \textit{Discr}(t_j) \tag{8.6}$$

8.2.3 Taille, hauteur et granularité

Le modèle vectoriel permet de représenter un contenu informationnel normalisé, en faisant abstraction de l'aspect, la présentation ou la taille des documents. Mais il est nécessaire de distinguer les documents selon leur taille et leur degré de composition. En effet, nous avons besoin de conserver la granularité des documents à l'indexation, pour intégrer ce paramètre à l'interrogation.

Une information d'une granularité sémantique donnée (en terme de quantité d'information) pourra se traduire par différentes granularités syntaxiques, selon que le document est résumé ou au contraire redondant. Nous pensons que la granularité sémantique est conservée dans l'index du document. Mais la granularité syntaxique est perdue à l'indexation : or, celle-ci aussi nous intéresse comme critère de recherche.

Nous utilisons les notions classiques de hauteur et de taille des documents, qui sont des mesures indépendantes du reste du corpus. Pour pouvoir comparer l'aspect "granularité" de deux documents, il est nécessaire de définir une mesure sur les documents qui soit relative au reste du corpus. On définit les fonctions $taille$, $hauteur$ et *granularité* sur les documents de $\mathcal{HDOCC}$:

$$\begin{array}{l} \textit{taille} : \mathcal{DOCC} \longrightarrow \mathbb{N} \\ \textit{hauteur} : \mathcal{DOCC} \longrightarrow \mathbb{N} \\ \textit{granularité} : \mathcal{DOCC} \longrightarrow \mathbb{R} \end{array} \tag{8.7}$$

La taille d'un document est simplement le nombre de documents atomiques dont il est composé, c'est-à-dire le nombre de feuilles de son arbre. La hauteur d'un document est le nombre maximum de relations de composition qu'il faut suivre à partir de sa racine pour atteindre une feuille, plus 1. La hauteur d'un document atomique est 1 :

$$\forall a \in \mathcal{A}, \textit{hauteur}(a) = 1 \tag{8.8}$$

Nous considérons que la granularité élémentaire est celle du document atomique, avec l'hypothèse simplificatrice suivante : « *les documents atomiques sont tous de la taille et de la granularité minimum* ». Cela signifie que les documents atomiques, de taille égale à 1, sont aussi équivalents du point de vue de la granularité. Un document atomique est de granularité 1 par rapport au reste du corpus :

$$\forall a \in \mathcal{A}, \textit{gran} = \text{taille}(a) = \textit{granularité}(a) = 1 \tag{8.9}$$

Cette hypothèse n'est pas valide dans le contexte des moteurs de recherche du Web car toutes les pages ne sont pas de granularité équivalente. Dans notre modèle, cette hypothèse est valide à la condition qu'il y ait une fragmentation adéquate des pages HTML en documents atomiques.

8.3 Indexation d'un document structuré ds_i

L'indexation d'un document structuré ds considère le contenu sémantique de chaque nœud non-feuille de l'arbre comme étant l'agrégation des contenus de ses fils, et construit les index vectoriels $\vec{ds_i}$. L'indexation du contenu atomique d'un document structuré ds_i en un vecteur $\vec{ds_i}$ suit les mêmes principes que dans le cas des documents atomiques. Elle consiste à extraire les termes t_j représentatifs des documents et à leur affecter une pondération *w-ds*$_{ij} \in [0, 1]$, qui est calculée à l'aide d'une fonction classique, de type $tf * idf$.

$$\forall ds_i \in \mathcal{DS}, \vec{ds_i} = (\textit{w-ds}_{i1},\ \textit{w-ds}_{i2} \ldots \textit{w-ds}_{ij} \ldots \textit{w-ds}_{in}) \tag{8.10}$$

8.3.1 Pondération

De la même manière que pour les documents atomiques, le calcul de la pondération *w-ds*$_{ij}$ d'un terme t_j par rapport à un document structuré ds_i combine l'aspect "*pouvoir résumant*" du terme pour le document, avec l'aspect "*pouvoir discriminant*" du terme. L'indexation d'un nœud ds_i non feuille consiste à calculer récursivement la pondération de chaque terme t_j à partir des calculs effectués précédemment sur les fils (les unités d'indexation filles). Les fils peuvent être des documents structurés ou des documents atomiques. Ces derniers sont indexés comme présenté dans la section 8.2.

La fonction de pondération se base sur les informations suivantes :

Fréquence locale : *tf-ds*$_{ij}$ est le nombre d'occurrences du terme t_j dans le document structuré ds_i :

$$tf_{ij} = \sum_{a_k \in feuilles(ds_i)} (tf_{i,k}) \tag{8.11}$$

Fréquence documentaire : *df-ds*$_j$ est le nombre de documents structurés dans lesquels le terme t_j apparaît :

$$\textit{df-ds}_j = |\{ds_i \in \mathcal{DS} \mid \exists a_k \in ds_i \mid t_j \in a_k\}| \tag{8.12}$$

Taille du corpus : N_{doc} est le nombre de documents du corpus. Ici, le corpus est l'ensemble des documents structurés, et N_{doc} est donc le nombre de documents structurés :

$$N_{doc} = |\mathcal{DS}| \tag{8.13}$$

8.3.2 Le problème du df

Alors que le calcul de *tf-ds*$_{ij}$ (la composante ayant trait au pouvoir résumant du terme) relève d'un simple comptage des termes dans le document structuré et dans sa hiérarchie, le calcul de *df-ds*$_j$ (la composante ayant trait au pouvoir discriminant du terme) est plus délicat. En effet, la calcul du df fait intervenir la fréquence documentaire d'un terme, ce qui implique de savoir dans quel corpus on se place. Dans le cas des documents structurés, on ne peut pas considérer simplement que le corpus est l'ensemble des documents structurés, car ils ne sont pas disjoints (un document structuré peut être composé d'autres documents structurés). Si on procédait de cette manière, à chaque apparition d'un terme dans un document atomique, son df serait augmenté du nombre des documents structurés ascendants.

Pour résoudre ce problème, il nous faut définir un ensemble de "corpus", dans lesquels il est possible de calculer la composante relative au pouvoir discriminant des termes. Les éléments de chaque corpus doivent donc être disjoints deux à deux, ce qui est le cas pour le corpus des documents atomiques $\mathcal{A}$, que nous appelleront *corpus de niveau 1*.

Une solution simple pour obtenir ces ensembles de corpus serait de les partitionner en fonction de la hauteur de l'arbre à laquelle se situent les documents. Ainsi, nous aurions, comme présenté sur la figure 8.3, le *corpus de niveau 1* des documents atomiques, le *corpus de niveau 2* des documents structurés se situant à la hauteur 1 d'un arbre (c'est-à-dire les documents structurés qui possèdent seulement des documents atomiques comme composants directs), le *corpus de niveau 3*, etc.

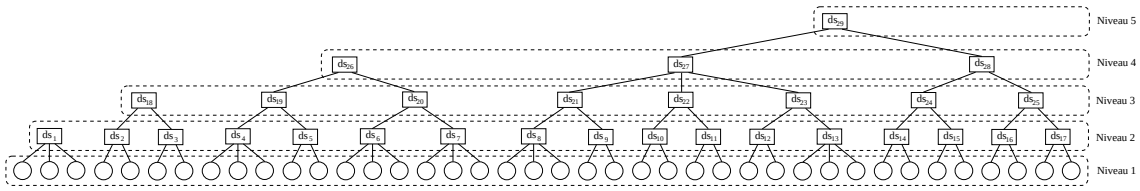

FIG. 8.3 – Partition selon la hauteur.

Mais la hauteur d'un arbre est une mesure locale, qui ne permet pas de partitionner l'ensemble des documents structurés en se basant sur une comparaison des documents entre eux, et qui ne donne donc aucune garantie d'obtenir des partitions équilibrées. De ce fait, il suffit qu'une collection soit composée de documents de tailles très différentes, représentés par des arbres de hauteurs très différentes, pour obtenir par exemple un *corpus de niveau 18* ne contenant qu'un document structuré, un *corpus de niveau 17* ne contenant que trois document structuré, etc.

Le même problème se pose si nous choisissons la *taille* d'un document comme critère pour le partitionnement. Nous avons préféré utiliser la *granularité* des documents, qui est une mesure globale comme défini dans l'équation 8.23. Cette méthode a le mérite de réunir dans une seule et même partition tous les documents de granularité maximale.

8.3.3 Partition des corpus

Nous définissons une fonction qui répartit les documents structurés en nb_{niv} partitions, nb_{niv} étant la hauteur maximale des arbres des documents structurés. Cette fonction se base sur la hauteur et la granularité d'un document, que nous définissons dans la section 8.3.5.

$$\begin{array}{rl} niveau: & \mathcal{DS} \longrightarrow [1..nb_{niv}+1] \\ \text{Avec}: & \forall a_i \in \mathcal{A}, niveau(a_i) = 1 \end{array} \tag{8.14}$$

Ainsi, tous les documents structurés dont la hauteur est nb_{niv} verront leurs composants répartis dans les corpus aux différents niveaux, en fonction de leur granularité. Si tous les documents ont la même hauteur, la partition est triviale. On obtient ainsi des partitions équilibrées, comme avec l'exemple présenté dans la figure 8.4. Le tableau 8.5 montre la taille, la hauteur, la granularité et enfin le niveau de chacun des documents structurés de la figure 8.4.

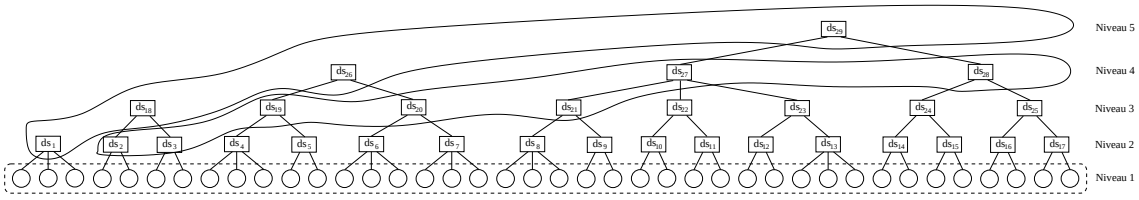

FIG. 8.4 – Partition selon la granularité.

Document					Racine du document	
Numéro	Taille	Hauteur	Granularité	Niveau	Numéro	Taille
ds_1	3	2	4	4	ds_1	2
ds_{18}	4	3	4	4	ds_{18}	4
ds_{26}	11	4	4	4	ds_{26}	11
ds_{29}	22	5	4	4	ds_{29}	22
ds_{19}	5	3	1.82	3	ds_{26}	11
ds_2	2	2	2	3	ds_{18}	4
ds_3	2	2	2	3	ds_{18}	4
ds_{20}	6	3	2.18	3	ds_{26}	11
ds_{27}	14	4	2.55	3	ds_{29}	22

FIG. 8.5 – Taille, hauteur, granularité et niveau des documents structurés de la figure 8.4.

La partition des corpus est horizontale, et les documents structurés d'un niveau donné sont disjoints deux à deux (un document ne peut pas appartenir au même niveau qu'un de ses descendants ou ascendants). Les documents atomiques vérifient cette propriété, car ils constituent à eux seuls le corpus de niveau 1. Les documents structurés, eux, vérifient la contrainte suivante :

$$\forall ds_i, ds_j \in \mathcal{DS}, ds_j \in \textit{nœuds}(ds_i) \Rightarrow niveau(ds_i) > niveau(ds_j) \tag{8.15}$$

8.3.4 Pondération

Dans cette optique de calcul du *df-ds*$_j$ d'un terme dans un corpus d'un niveau $niv \in [1...nb_{niv}]$ donné, la fonction de pondération se base sur les informations suivantes :

Fréquence locale : *tf-ds*$_{ij}$ est inchangée.

Fréquence documentaire : *df-ds*$_{j,niv}$ est le nombre de documents du niveau niv dans lesquels le terme t_j apparaît :

$$\textit{df-ds}_{j,niv} = |\{ds_i \mid niveau(ds_i) = niv \wedge \exists a_k \in feuilles(ds_i) \mid t_j \in a_k\}| \tag{8.16}$$

Taille du corpus : $N_{doc,niv}$ est le nombre de documents du corpus du niveau niv :

$$N_{doc,niv} = |\{ds_i \mid niveau(ds_i) = niv\}| \tag{8.17}$$

Ainsi, la pondération *w-ds*$_{ij}$ du terme t_j dans le document structuré ds_i est calculée à partir de ces informations. Le calcul est similaire au cas des documents atomiques, comme le montre l'équation 8.18, avec une combinaison du pouvoir résumant "*Résum*(ds_i, t_j)" du terme t_j pour le document ds_i, et du pouvoir discriminant "*Discr*(t_j, niv)" du terme dans le corpus du document ds_i.

$$\begin{aligned} \textit{Résum}(ds_i, t_j) &= \log_2(\textit{tf-ds}_{ij}) + 1 \\ \textit{Discr}(t_j, niv) &= \frac{1}{\log_2(\frac{N_{doc,niv}}{\textit{df-niv}_{j,niv}})} \\ \textit{w-ds}_{ij} &= \textit{Résum}(ds_i, t_j) * \textit{Discr}(t_j, niv) \end{aligned} \tag{8.18}$$

8.3.5 Taille, hauteur et granularité

Le calcul de la granularité d'un document structuré ds se base sur la fonction $taille$. La granularité maximale des documents, c'est-à-dire la granularité de tous les éléments racines, est la hauteur maximum des arbres de $\mathcal{DS}$:

$$nb_{niv} = max_{ds \in \mathcal{DS}}(hauteur(ds)) \tag{8.19}$$

On calcule la taille $taille(ds)$ (respectivement, la hauteur $hauteur(ds)$) d'un document ds en appliquant récursivement la fonction $taille$ (respectivement, la hauteur $hauteur(ds)$) :

$$\begin{array}{rl} taille(ds_i) & = \sum_{a_j \in A_i} taille(a_j) + \sum_{ds_j \in DS_i} taille(ds_j) \\ & = |A_i| + \sum_{ds_j \in DS_i} taille(ds_j) \\ hauteur(ds_i) & = 1 \quad si \quad DS_i = \emptyset, \quad \text{c'est-à-dire : } ds_i \in \mathcal{A} \\ & = max_{ds_j \in DS_i}(hauteur(ds_j)) \quad sinon \end{array} \tag{8.20}$$

Ensuite, la granularité d'un document structuré ds composé de plusieurs documents atomiques a se calcule à partir de la taille de ds, relativement à la taille du document racine. Nous avons vu que les documents atomiques ont tous une granularité de 1. D'un autre côté, les documents structurés racines ont une granularité égale à nb_{niv}. Nous pouvons donc préciser le co-domaine de la fonction *granularité* :

$$\textit{granularité} : \mathcal{DOCC} \longrightarrow [1..nb_{niv}] \tag{8.21}$$

Enfin, nous calculons la granularité d'un document structuré, qui est la proportion des documents atomiques du document structuré par rapport à son document racine, rapporté à l'échelle de la granularité de 1 à nb_{niv} :

$$\forall ds_i \in \mathcal{DS}, \textit{gran}_i = \textit{granularité}(ds_i) \tag{8.22}$$

$$\begin{array}{rl} \textit{granularité}(ds_i) & = nb_{niv} \quad si \quad \textit{est-racine}(ds_i) \\ & = max(1, nb_{niv} * \frac{taille(ds_i)}{taille(racine(ds_i))}) \quad sinon \end{array} \tag{8.23}$$

On remarque qu'un document structuré dont le rapport entre la taille et la taille de son document racine est inférieur à $\frac{1}{nb_{niv}}$ sera assimilé, du point de vue de la granularité, à un document atomique.

8.3.6 Remontée d'information et résumé

Du point de vue du lecteur, cette indexation suppose que le document sera lu de manière atomique, en une opération unique et indépendante. Il y a donc au cours de l'extraction du contenu sémantique, une propagation de l'information du "bas" de la structure logique (les feuilles) vers le "haut" (le document structuré). On réalise ainsi une "remontée" des termes d'indexation dans la hiérarchie du document, en tenant compte de la partition en corpus à chaque niveau de granularité : en effet, on calcule le pouvoir discriminant d'un terme relativement au sous-corpus du document qui le contient.

De plus, cette indexation extrait un résumé informationnel du document, et non pas une concaténation, du fait de la fonction de filtrage utilisée pour éliminer les termes trop peu importants pour un document. En effet, les termes sont filtrés pour alléger la taille des index, et leur pondération doit être supérieure à un seuil donné $seuil_{comp}$ pour être conservée :

$$\begin{array}{rl} \forall ds_i \in \mathcal{DS}, \forall t_j, \textit{w-ds}_{ij} & = 0 \quad si \quad \textit{w-ds}_{ij} < seuil_{comp} \\ & = \textit{w-ds}_{ij} \quad sinon \end{array} \tag{8.24}$$

8.4 Indexation d'un chemin de lecture ch^k

L'extraction de la sémantique d'un chemin de lecture revient à représenter à l'aide d'un vecteur de termes pondérés $\vec{ch}$ l'information qui y est décrite de manière non atomique :

$$\forall ch \in \mathcal{CH}, \vec{ch} = (w\text{-}ch_{i1},\ w\text{-}ch_{i2} \ldots w\text{-}ch_{ij} \ldots w\text{-}ch_{in}) \tag{8.25}$$

Une solution simple pour indexer un chemin de lecture serait de le considérer de la même manière qu'un document structuré, et de réaliser une agrégation des documents atomiques. L'algorithme d'indexation consisterait alors à recalculer la pondération des termes d'un chemin dans le corpus $\mathcal{CH}$ des chemins de lecture, en considérant le pouvoir discriminant des termes par rapport à ce corpus de la même manière que l'indexation décrite dans la section précédente.

Mais, afin de mettre en œuvre les principes présentés dans le chapitre 6, nous proposons un algorithme d'indexation de chemins qui prend en compte l'ordre dans lequel sont lus les documents atomiques. Cet algorithme simule une lecture séquentielle des documents atomiques du chemin, comme présenté dans les sections suivantes.

8.4.1 Simulation de lecture

La lecture d'un chemin parmi d'autres est linéaire, mais l'indexation de plusieurs chemins de lecture pour un même hyperdocument $hd \in \mathcal{HD}$ permet de prendre en compte la délinéarisation de la lecture. Ainsi, l'algorithme d'indexation de chemins réalise une *simulation de lecture*, en suivant le chemin pour "*lire*" chaque nœud dans l'ordre proposé et accumuler un index du chemin de lecture. Cet index représente l'information que pourrait retirer un utilisateur à la lecture du chemin, indépendamment du contexte du chemin.

Comme présenté dans le chapitre 6, nous nous inspirons des principes de progression thématique dans un texte intégrant la *mémoire de lecture* pour la compréhension au cours d'une lecture [Vandendorpe91b]. Nous avons présenté le texte comme un développement progressif et cohérent de l'information à partir d'un thème donné, avec les différents schémas de progression thématique proposés par Danes (cf. chapitre 6, [Danes74]) : progression à thème constant, à thème linéaire, etc.

Notre algorithme d'indexation de chemins de lecture a pour objectif de modéliser la progression thématique, en se basant sur les hypothèses de la *mémoire de lecture*, du *principe d'accumulation*, de la *rupture sémantique* et du *cotexte textuel* présentés dans la section 6.12.2.

8.4.2 Algorithme de lecture

Nous décrivons l'algorithme d'indexation (ou de "lecture") des chemins de lecture. On rappelle la formalisation d'un chemin ch_i^k et d'un document structuré ds_i associé (cf. le modèle de documents, section 7.4.1) :

$$\begin{array}{l} ch_i^k = < A_k, Arcs_k, R^k_{chem} > \in \mathcal{CH}, \\ ds_i = < A_i, DS_i, R^i_{comp} > \in \mathcal{DS} \end{array} \tag{8.26}$$

Un arc arc_k est un triplet document source, document destination et coefficient de rupture sémantique :

$$arc_j = (a_{src} \in A_k, a_{dest} \in A_k, \beta_j) \in Arcs_k \tag{8.27}$$

L'algorithme d'indexation d'un chemin ch_i^k composé de n arcs reliant $n+1$ documents atomiques (cf. figure 8.6) extrait un index $\vec{ch_i^k}$. Nous appellerons a_1 le premier nœud visité, a_2 le suivant, etc. L'arc arc_1 sera donc un arc de a_1 à a_2 : $arc_1 = (a_1, a_2, \beta_1)$.

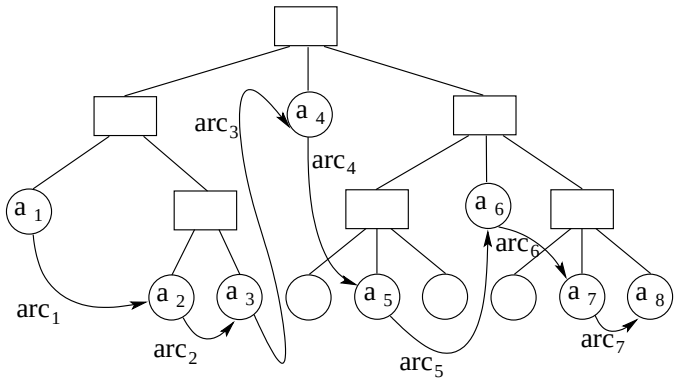

FIG. 8.6 – Exemple de chemin de lecture.

L'algorithme simule une lecture, comme présenté dans l'algorithme de la figure 8.7, pour extraire un index $\vec{ch_i^k}$ en utilisant les vecteurs intermédiaires $\vec{lect}_j$ et $\vec{mém}_j$:

Mémoire de lecture : le vecteur $\vec{mém}_j$ représente la mémoire de lecture à une étape j donnée, c'est-à-dire l'information acquise au cours des étapes 1 à (j - 1), qui est réutilisée par le lecteur pour mieux comprendre l'information du nœud j.

Accumulateur de lecture : le vecteur $\vec{lect}_j$ est l'*accumulateur de lecture*. Son rôle est d'accumuler l'information collectée sur chaque nœud, au fur et à mesure de la lecture. Ainsi, à une étape j donnée de la lecture, le vecteur $\vec{lect}_j$ représente l'information collectée à partir du nœud a_1 jusqu'au nœud a_j. C'est l'accumulateur de lecture qui construit le vecteur final d'index du chemin.

La lecture d'un nœud a_j amène à l'acquisition d'une information propre au nœud, que nous appelons $\vec{local}_j$, et qui est une combinaison de la mémoire de lecture $\vec{mém}_j$ et de l'information $\vec{a_j}$ effectivement présente dans le nœud. Ensuite, le lecteur "accumule" les vecteurs $\vec{a_j}$ sur chaque nœud. De son côté, la mémoire de lecture évolue en fonction de la rupture sémantique et de l'information apportée par le nœud a_j.

Dans les sections suivantes, nous présentons les étapes de l'algorithme de lecture de chemins, qui sont résumées dans la figure suivante :

(a) Initialisation : lecture du premier nœud a_1 :
(a.1) Mise à zéro de la mémoire de lecture : $\vec{mém}_1 = \vec{0}$
(a.2) Acquisition du nœud a_1 : $\vec{lect}_1 = \alpha_{mem}.\vec{mém}_1 + (1 - \alpha_{mem}).\vec{a_1}$
(b) Lecture : activations successives des arcs $\forall arc_j = (a_j, a_{j+1}, \beta_j) \in arcs_k$:
(b.1) Mise à jour de la mémoire de lecture : $\vec{mém}_j = \beta_j.(\vec{mém}_j + \vec{a_j})$
(b.2) Calcul de l'information locale : $\vec{local}_{j+1} = \alpha_{mem}.\vec{mém}_{j+1} + (1 - \alpha_{mem}).\vec{a_{j+1}}$
(b.3) Accumulation : $\vec{lect}_{j+1} = \gamma.\vec{lect}_j + \alpha.\vec{local}_{j+1}$
(c) Activation du dernier arc : $arc_n = (a_n, a_{n+1}, \beta_n)$:
(c.1) Calcul de l'information acquise sur tout le chemin : $\vec{lect}_{n+1}$
(d) Intégration du cotexte textuel :
(d.1) $\vec{ch}_i^k = (1 - \alpha_{cotexte}) * \vec{lect}_{n+1} + \alpha_{cotexte} * \frac{taille(ch_i^k)}{taille(ds_i)} * \vec{ds}_i$

FIG. 8.7 – Algorithme de lecture d'un chemin.

8.4.3 Etapes de l'algorithme de lecture de chemins

a) Initialisation : lecture du premier nœud

La première étape consiste à initialiser les "vecteurs de travail" $\vec{lect}_1$ et $\vec{mém}_1$, en procédant à la lecture du premier nœud du chemin :

Mémoire de lecture : la mémoire de lecture initiale $\vec{mém}_1$ est initialisée avec le vecteur nul, en considérant que le lecteur commence sa lecture avec un "esprit neuf", et donc sans mémoire de lecture. La lecture du nœud a_1 n'est donc influencée par aucune connaissance préalablement accumulée.

Accumulateur de lecture : l'accumulateur de lecture $\vec{lect}_j$ est initialisé par une combinaison de la mémoire de lecture initiale (qui est d'ailleurs nulle) et le vecteur $\vec{a_1}$. L'information accumulée sur un chemin de longueur 1 est simplement le contenu du nœud.

b) Activation d'un arc

L'activation d'un arc ($arc_j = (a_j, a_{j+1}, \beta_j)$, cf. figure 8.8) se décompose en trois étapes. La lecture d'un nœud a_{j+1} consiste à calculer l'information locale au nœud : $\vec{local}_{j+1}$, c'est-à-dire l'information que l'on peut retirer de la lecture de a_{j+1} après la lecture des nœuds précédents. Ce calcul nécessite de disposer de la mémoire de lecture mem_j mise à jour. Et enfin, l'information collectée est accumulée au vecteur $\vec{lect}_{j+1}$.

(b.1) : à chaque itération de l'algorithme, on met à jour la mémoire de lecture, en fonction du nœud précédent. La conservation de la mémoire de lecture au fil du temps dépend des ruptures sémantiques successives β_j : plus la rupture sémantique est importante, moins la mémoire est conservée.

(b.2) : le calcul de l'information locale est une combinaison de la mémoire de lecture et du contenu du nœud. Le paramètre du système α_{mem} est utilisé pour équilibrer l'importance de la mémoire par rapport au contenu.

(b.3) : l'information locale que l'on vient de calculer est accumulée et ajoutée au vecteur $\vec{lect}_{j+1}$. L'information collectée sur les nœuds précédents subit une dégradation ou une augmentation, en fonction du paramètre du système γ. En effet, ce paramètre permet de privilégier soit les nœuds du début du chemin (si γ est supérieur à 1), soit les nœuds de la fin du chemin (si γ est inférieur à 1).

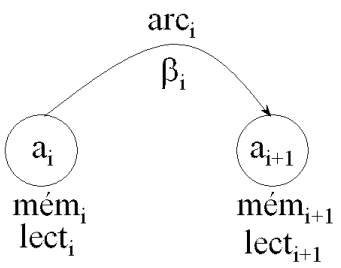

FIG. 8.8 – Exemple d'arc d'un chemin.

c) Activation du dernier arc

L'index du chemin est donc le vecteur d'accumulation de lecture $\vec{lect}_{n+1}$ qui est obtenu après activation de tous les arcs, et donc lecture de tous les nœuds.

d) Intégration du cotexte textuel

Afin de prendre en compte le cotexte textuel des nœuds parcourus, nous proposons d'intégrer à l'index du chemin l'index du document structuré englobant. L'importance du cotexte est fixé à l'aide du paramètre du système $\alpha_{cotexte}$. De plus, la proportion du chemin dans le document structuré est prise en compte avec le facteur $\frac{taille(ch_i^k)}{taille(ds_i)}$ qui permet d'éviter que l'index du chemin soit "écrasé" par son cotexte.

8.4.4 Interprétation de l'algorithme

Le vecteur de l'information locale $\vec{local}_j$ peut être supprimé de l'écriture de l'équation formalisant l'indexation d'un chemin. On obtient alors la formule récursive suivante pour exprimer la mémoire de lecture :

$$\begin{array}{l} \vec{mem}_1 = \vec{0} \\ \vec{mem}_{j+1} = \beta_j * \{\vec{mem}_j + \vec{a_j}\} \\ \vec{mem}_n = \sum_{i=1}^{n-1} \left\{ \vec{a_j} * \pi_{k=1}^{j-1} \beta_k \right\} \end{array} \quad (8.28)$$

Cela nous permet de réécrire également la formulation de l'accumulateur de lecture :

$$\begin{array}{l} \vec{lect}_1 = (1 - \alpha_{mem}) * \vec{a_1} \\ \\ \vec{lect}_{j+1} = \left\{ \begin{array}{c} \alpha_{mem} * \vec{mem}_{j+1} \\ +(1 - \alpha_{mem}) * \vec{a_{j+1}} \end{array} \right\} + \gamma * \vec{lect}_j \\ \\ \vec{lect}_n = \left\{ \begin{array}{c} \alpha_{mem} * \{\sum_{i=1}^{n} (\gamma^{n-i} * \vec{mem}_i)\} \\ +(1 - \alpha_{mem}) * \{\sum_{i=1}^{n} (\gamma^{n-i} * \vec{a_i})\} \end{array} \right\} \end{array} \quad (8.29)$$

a) Evolution du contexte de lecture

La mémoire de lecture est conservée d'un nœud sur l'autre, en fonction du coefficient de rupture sémantique. Plus β_j est élevé, moins la mémoire est reconduite pour la lecture des nœuds suivants. A l'extrême, si $\beta_j = 1$, on considère qu'il y a rupture sémantique totale entre les deux nœuds qui entraîne une remise à zéro de la mémoire de lecture. Ainsi, si tous les coefficients sont égaux à 1, le calcul de l'index $\vec{ch_i^k}$ est simplifié :

$$\forall j \in [i..n], \beta_j = 0 \Rightarrow \left\{ \begin{array}{l} \vec{mem}_n = \vec{0} \\ \vec{lect}_n = (1 - \alpha_{mem}) * \sum_{i=1}^{n} (\gamma^{n-i} * \vec{a_i}) \end{array} \right\} \quad (8.30)$$

Enfin, la participation de chaque nouveau nœud au contexte de lecture (par le biais du vecteur $\vec{info - acc}$) est elle aussi fonction du coefficient de rupture sémantique, et sera d'autant plus importante qu'il est élevé.

On ne considère donc pas le contexte de lecture comme un accumulateur d'information, étant donné que n'importe quelle composante w_{ij} de $\vec{accu_{contexte}}$ relative à un terme t_j peut augmenter ou diminuer et revenir à sa valeur initiale (thème global du chemin) en cas de rupture sémantique brutale.

b) Progression thématique

Dans le cas de coefficients de rupture élevés, on repart d'un nouveau nœud avec une mémoire de lecture vide : on simule alors une *progression à thème constant* [Danes74]. Au contraire, dans le cas de coefficients de rupture faibles, l'essentiel de la mémoire de lecture est conservée : on simule alors une *progression à thème linéaire*.

Cet algorithme tient compte du sens de lecture. Un même hd_j est indexé par un ensemble de chemins $\{\vec{ch_i^k}\}$ et peut donc être lu par n chemins différents qui donneront n index différents, ce qui est le cas dans les hypertextes comme on a pu le voir avec l'exemple des hyperfictions dans la section 6. L'indexation d'un site composé de n fragments $a_1,\ a_2,\ \ldots a_n$

pourra être représentée par un unique chemin de lecture linéaire avec les ds, et par plusieurs chemins de lecture avec les hd, dans un graphe composé de n nœuds avec les relations de cheminement comme arcs.

8.4.5 Taille, hauteur et granularité

La taille d'un chemin de lecture est calculée simplement : il s'agit du nombre de documents atomiques parcourus. La hauteur d'un chemin, par convention, est toujours 1 :

$$\begin{array}{l} \forall ch_i^k \in \mathcal{CH}, \\ \qquad \begin{array}{ll} taille(ch_i^k) & = |A_k| \\ hauteur(ch_i^k) & = 1 \end{array} \end{array} \tag{8.31}$$

La granularité d'un chemin ch_i^k est calculée par rapport à la taille des autres chemins du corpus, de manière semblable au calcul de la granularité des documents structurés. Mais dans ce cas, il n'y a pas de contraintes de dépendance entre les chemins (qui peuvent parcourir des sous-ensembles de documents atomiques non disjoints, mais sans relation d'inclusion entre eux), et il existe un seul corpus de chemins. La granularité d'un chemin par rapport à ce corpus est le ratio entre sa taille et la taille du plus grand chemin du corpus. Nous pouvons donc préciser le co-domaine de la fonction *granularité* sur les chemins de lecture, qui diffère de celui des documents structurés :

$$\begin{array}{l} \textit{granularité} : \mathcal{CHC} \longrightarrow [0..1] \\ \forall ch_i^k \in \mathcal{CH}, \textit{gran}_i^k = \textit{granularité}(ch_i^k) \\ \textit{granularité}(ch_i^k) = \frac{taille(ch_i^k)}{max_{ch_i^k \in \mathcal{CH}}(taille(ch_i^k))} \end{array} \tag{8.32}$$

8.4.6 Chemin et résumé

L'indexation d'un chemin extrait un résumé informationnel, et applique une fonction de filtrage pour éliminer les termes trop peu importants. Leur pondération doit être supérieure à un seuil donné $seuil_{chem}$ pour être conservée :

$$\begin{array}{lll} \forall ch_i^k \in \mathcal{CH}, & \forall t_j, \textit{w-ch}_{ikj} & = 0 \quad si \quad \textit{w-ch}_{ikj} < seuil_{chem} \\ & & = \textit{w-ch}_{ikj} \quad sinon \end{array} \tag{8.33}$$

8.5 Indexation d'un hyperdocument hd_i

L'index d'un hyperdocument $hd = < ds, Chemins, R_{chem} >$ est composé des index construits par les étapes précédentes. En effet, un hyperdocument est un document ds_i auquel est associé un ensemble de chemin $chemins_i$. L'index d'un hyperdocument est donc un vecteur de document $\vec{ds_i}$ auquel est associé un ensemble de vecteurs de chemins :

$$\forall hd \in \mathcal{HD}, \quad index \quad hd : < \vec{ds}, \{\vec{ch^k}\} > \tag{8.34}$$

8.6 Indexation du contexte

On utilise la structure de référence pour extraire le contexte des documents. Pour chaque type de documents ($\mathcal{A}$, $\mathcal{DS}$ et $\mathcal{CH}$), le modèle de documents nous donne le contexte correspondant ($\mathcal{AC}$, $\mathcal{DSC}$ et $\mathcal{CHC}$). Chaque document doc est associé à un ensemble MI et à un ensemble IA qui représentent la méta-information et l'information accessible d'un document.

On rappelle la définition du contexte (MI_i, IA_i) d'un document :

$$\begin{array}{l} \forall doc_i \in \mathcal{DOC}, \quad MI_i = \{(ds_j, \beta_{\text{coût}}) \mid (ds_j, ds_i) \in \mathcal{R}_{ref}\} \\ \qquad\qquad\quad IA_i = \{(ds_j, \beta_{\text{coût}}) \mid (ds_i, ds_j) \in \mathcal{R}_{ref}\} \end{array} \tag{8.35}$$

Le contexte étant défini de manière identique pour chacun des types de documents, l'extraction de l'autorité, du rayonnement, de la méta-information et de l'information accessible se déroule de la même manière. Dans la suite de cette section, nous présentons cette indexation en prenant l'exemple des documents structurés.

8.6.1 Composants du contexte

L'impact du contexte d'un document sur l'indexation produit les index aut, ray, $\vec{\textit{méta-info}}$ et $\vec{\textit{info-acc}}$. Le vecteur $\vec{\textit{méta-info}}$ représente l'espace d'information qui référence le document. Le vecteur $\vec{\textit{info-acc}}$ représente l'information accessible à partir du document. La valeur aut (respectivement, ray) représente son autorité (respectivement, son rayonnement).

Définition 21 *Autorité : pour chaque document $ds \in \mathcal{DS}$, aut représente l'autorité de ds dans le graphe de la structure de référence.*

Définition 22 *Rayonnement : pour chaque document $ds \in \mathcal{DS}$, ray représente le rayonnement de ds dans le graphe de la structure de référence.*

Finalement, le contexte d'un document (document atomique, document structuré, ou chemin de lecture) est indexé par le quadruplet composé de l'autorité, du rayonnement, de la méta-information et de l'information accessible :

Définition 23 *Contexte : pour chaque document $ds \in \mathcal{DS}$, on indexe son contexte :*

$$\forall ds \in \mathcal{DS}, contexte = (aut \in [0..1], ray \in [0..1], \vec{\textit{méta-info}}, \vec{\textit{info-acc}}) \tag{8.36}$$

L'information accessible représente l'information qu'un "surfeur aléatoire" pourrait collecter à partir, par exemple, d'un document structuré ds, et la méta-information représente une information supplémentaire sur ds (cf. chapitre 6). On représente ces deux composantes du contexte par deux vecteurs : $\vec{\textit{méta-info}}$ et $\vec{\textit{info-acc}}$:

$$\forall ds \in \mathcal{DS}, \vec{\textit{méta-info}} = (\textit{w-mi-ds}_1,\ \textit{w-mi-ds}_2 \ldots \textit{w-mi-ds}_j \ldots \textit{w-mi-ds}_n) \tag{8.37}$$

$$\forall ds \in \mathcal{DS}, \vec{\textit{info-acc}} = (\textit{w-ia-ds}_1,\ \textit{w-ia-ds}_2 \ldots \textit{w-ia-ds}_j \ldots \textit{w-ia-ds}_n) \tag{8.38}$$

On associe à chacun un “score” (la popularité) qui lui correspond, c’est-à-dire le score d’autorité “aut” pour la méta-information, et le score de rayonnement “ray” pour l’information accessible. Ainsi, l’importance de la méta-information pour un document structuré est proportionnelle à son score d’autorité. De même, l’information accessible d’un document structuré est d’autant plus importante que le document est “rayonnant”, c’est-à-dire possède un score ray important.

8.6.2 Autorité et rayonnement

Le calcul des scores d’autorité et de rayonnement est basé sur un algorithme classique inspiré des “*Hubs and Authorities*”, qui réalise une propagation de popularité le long des relations de référence. Son objectif est d’extraire, pour chaque document, les deux scores aut et ray du graphe de la structure de référence. L’algorithme fait évoluer, à chaque étape p, les scores $aut_{i,p}$ et $ray_{i,p}$ pour chaque document ds_i, jusqu’à ce qu’un état stable soit atteint.

(a) Initialisation : l’étape 0 initialise les valeurs d’autorité et de rayonnement de tous les documents à une valeur v_{init} commune :

(a.1) $\forall ds_i \in \mathcal{DS}, aut'_{i,0} = v_{init}$

(a.2) $\forall ds_i \in \mathcal{DS}, ray'_{i,0} = v_{init}$

(b) Normalisation :

(b.1) Autorité : $\forall ds_i \in \mathcal{DS}, \quad aut_{i,0} = \frac{aut'_{i,0}}{\sqrt{\sum_{\forall ds_j \in \mathcal{DS}} aut'^2_{j,0}}}$

(b.2) Rayonnement : $\forall ds_i \in \mathcal{DS}, \quad ray_{i,0} = \frac{ray'_{i,0}}{\sqrt{\sum_{\forall ds_j \in \mathcal{DS}} ray'^2_{j,0}}}$

(c) Itération (étape p) : mise à jour des scores d’autorité et de rayonnement, pour chaque étape $p \in [1..n_{stable}]$:

(c.1) Autorité : $\forall ds_i \in \mathcal{DS}, \quad aut'_{i,p} = \sum_{ds_j \in MI_i} ray_{j,p-1}$

(c.2) Rayonnement : $\forall ds_i \in \mathcal{DS}, \quad ray'_{i,p} = \sum_{ds_j \in IA_i} aut_{j,p-1}$

(c.3) Normalisation : normalise les scores en $aut_{i,p}$ et $ray_{i,p}$ (cf. étape b).

(d) Résultats : convergence des scores $aut_{i,p}$, $ray_{i,p}$, et atteinte d’un état stable.

FIG. 8.9 – Calcul de l’autorité et du rayonnement des documents structurés.

a) Initialisation

L’initialisation permet de “distribuer” uniformément un score initial d’autorité et de rayonnement. Le choix de la valeur d’initialisation v_{init} commune à tous les documents structurés est sans conséquence, étant donné que la première normalisation ramène tous les scores à $\frac{1}{\sqrt{|\mathcal{DS}|}}$.

b) Normalisation

La normalisation des scores d'autorité et de rayonnement (cf. figure 8.9) permet de s'assurer que la somme des carrés des autorités (respectivement des rayonnements) soit toujours égale à 1 (cf. équation 8.39). Il s'agit d'une *normalisation globale*, considérant la totalité du graphe pour normaliser la popularité de chaque nœud.

$$\begin{array}{l} \forall \ \textit{étape} \ \ k \in [1..n_{stable}], \\ \sum_{ds_i \in \mathcal{DS}} aut_{i,k}^2 = 1, \quad \sum_{ds_i \in \mathcal{DS}} ray_{i,k}^2 = 1 \end{array} \tag{8.39}$$

c) Itération

Chaque itération réalise une propagation de "popularité" le long des liens sortant et entrant d'un document structuré ds_i. Les scores de rayonnement sont propagés dans le sens des liens de référence, c'est-à-dire de la méta-information vers ds_i, pour calculer son autorité. Inversement, les scores d'autorité sont propagés dans le sens inverse des liens de référence, c'est-à-dire de l'information accessible vers ds_i, pour calculer son rayonnement.

On peut se représenter le mécanisme de propagation de popularité comme l'écoulement d'un liquide suivant le principe des vases communicants. Ainsi, selon le *principe de conservation globale de la popularité*, la "quantité de popularité" distribuée dans le graphe est conservée au cours des itérations successives. Avec notre algorithme, nous remarquons que la popularité (autorité ou rayonnement) d'un document structuré ds_i est utilisée à chaque étape à n reprises pour le calcul de la popularité d'autres documents structurés[1]. Cette "multiplication de la popularité" ne respecte pas le principe de conservation globale de la popularité, d'où la nécessité de la normalisation.

Une normalisation possible serait de contrôler la "quantité de popularité" propagée à chaque étape, en fonction du nombre de liens entrants ou sortants, comme présenté dans la variante détaillée ci-dessous. Mais, dans certains cas de figure, le problème du déficit de popularité intervient. Par exemple, le score de rayonnement d'un document structuré ds_i est "perdu" si ds_i ne contient aucun lien sortant. Pour cette raison, il est nécessaire que la normalisation "réinjecte" la popularité "gaspillée", uniformément dans tous le graphe.

d) Résultats et état stable

Enfin, l'algorithme atteint un état stable après un nombre variable d'itérations n_{stable}. Pour simplifier, on considère qu'un état stable est atteint quand la différence entre les scores d'autorité et de rayonnement de deux étapes successives est inférieure à un seuil $seuil_{aut}$ (et $seuil_{ray}$) donné :

$$si \ \ \textit{étape} \ \ p = n_{stable} \Rightarrow \left\{ \begin{array}{l} \frac{\sum_{ds_i \in \mathcal{DS}} (aut_{i,p} - aut_{i,p-1})}{|\mathcal{DS}|} < seuil_{aut} \\ \wedge \frac{\sum_{ds_i \in \mathcal{DS}} (ray_{i,p} - ray_{i,p-1})}{|\mathcal{DS}|} < seuil_{ray} \end{array} \right\} \tag{8.40}$$

[1] Avec n le nombre de liens entrant ou sortant de ds_i.

La convergence des scores d'autorité et de rayonnement a été démontrée (cf. [Kleinberg99]). L'algorithme est alors stoppé, et on obtient les scores finaux.

e) Variante

On peut aussi utiliser une normalisation "locale", c'est-à-dire en utilisant uniquement le nombre de lien sortants des nœuds, pour pallier le problème de la "multiplication de la popularité". La propagation de popularité (étape (c) de l'algorithme) se calcule alors de la manière suivante :

$$\forall ds_i \in \mathcal{DS}, \quad aut_{i,p} = \sum_{ds_j \in MI_i} \frac{ray_{j,p-1}}{|IA_j|} \tag{8.41}$$

$$\forall ds_i \in \mathcal{DS}, \quad ray_{i,p} = \sum_{ds_j \in IA_i} \frac{aut_{j,p-1}}{|MI_j|} \tag{8.42}$$

La quantité d'information propagée est partagée entre tous les liens entrants ou sortants. Cette manière de calculer les scores respecte le *principe de conservation globale de la popularité*, sans qu'il soit nécessaire de réaliser l'étape (d) supplémentaire de normalisation globale. Cette variante est celle qui s'approche le plus de la métaphore de l'écoulement d'un liquide, mais elle ne traite pas le problème de "déficit de popularité". Il n'y a donc pas d'invariant global tel que celui de l'équation 8.39 pour la variante précédente.

8.6.3 Méta-information et information accessible

a) Indexation

L'indexation de l'information accessible (respectivement de la méta-information) d'un document structuré ds_i a pour objectif de représenter le contenu sémantique de l'espace d'information constitué par les documents structurés de IA_i (respectivement de MI_i). L'indexation de la méta-information est symétrique de l'indexation de l'information accessible. Elle consiste à effectuer une somme vectorielle des documents structurés de MI_i, combinée avec le coût de navigation $\beta_{coût}$:

$$\begin{array}{l} \forall ds_i \in \mathcal{DS}, \\ \text{Méta-information : } \vec{\textit{méta-info}}_i = \sum_{\vec{ds_j} \in MI_i} \frac{1}{\beta_{coût}} * \vec{ds_j} \end{array} \tag{8.43}$$

$$\begin{array}{l} \forall ds_i \in \mathcal{DS}, \\ \text{Information Accessible : } \vec{\textit{info-acc}}_i = \sum_{\vec{ds_j} \in IA_i} \frac{1}{\beta_{coût}} . \vec{ds_j} \end{array} \tag{8.44}$$

b) Normalisation et coût de navigation

L'extraction de la méta-information et de l'information accessible normalise la propagation de l'information le long des relations de référence, elle est basée sur un "coût de navigation". Ainsi, l'apport d'un document structuré à l'index de l'information accessible d'un

autre document structuré est calculé en fonction du "coût de navigation" pour l'atteindre. Cela permet de ne pas avantager outrageusement des pages contenant un grand nombre de liens, de la même manière qu'une fonction de pondération d'un terme dans un document est normalisée par rapport à la taille du document pour ne pas trop avantager les documents de grande taille.

Le coût de navigation permet de prendre en compte l'hypothèse de "quantité d'information potentiellement propagée" (cf. hypothèse 10, section 6.12.3), que l'on peut exprimer en terme de probabilité de navigation : « *la quantité d'information qui se propage le long d'un lien est proportionnelle à la probabilité qu'a un utilisateur "aléatoire" de suivre ce lien* ». Le coût de navigation est alors d'autant plus élevé que cette probabilité est faible. Cette hypothèse est à la base de l'algorithme de calcul du $PageRank$ de Google pour la propagation de popularité (cf. section 3.2, cf. [Brin et al.98]), ou encore de l'algorithme de Marchiori dans le cas d'une navigation aléatoire (cf. section 4.3, cf. [Marchiori97]).

Avec cette hypothèse, si un lien a très peu de chance d'être activé, alors la quantité d'information propagée dans notre modèle est quasi-nulle. Par exemple, nous considérons que très peu d'information est propagée le long d'un lien noyé parmi une liste de 500 autres liens. Dans un premier temps, nous avons considéré les choix entre les différents liens d'une page comme étant équiprobables, et nous avons en conséquence choisi un $\beta_{coût}$, pour un lien entre deux documents ds_i et ds_j, égal au nombre de liens sortants de la page ds_i source du lien :

$$\beta_{coût} = nb_{out-links} = card(IA_i) \tag{8.45}$$

Il existe de nombreux autres facteurs qui influencent la navigation sur le Web. Sans prétendre modéliser le comportement d'un utilisateur au cours de la navigation, il est toutefois possible d'intégrer au paramètre $\beta_{coût}$ d'autres informations, comme par exemple la "surface" du texte ou de l'image cliquable qui sert de point de départ au lien, ou encore sa position dans la page. Pour cela, nous faisons l'hypothèse suivante :

Hypothèse 11 *La probabilité qu'un utilisateur suive un lien est proportionnel à la surface activable du lien dans la page.*

La séparation entre structure physique et structure logique ne nous permet pas de connaître la surface d'affichage des mots de l'ancre, qui dépend (entre autres) de la fonte de caractère utilisée. La seule information dont nous disposons qui soit indépendante du dispositif d'affichage est le nombre de caractères de l'ancre (si elle est textuelle) ou sa surface en pixels (si elle est une image), que nous utilisons dans la suite en l'appelant la surface "surface($ds_i \longrightarrow ds_j$)" du lien de ds_i vers ds_j.

En considérant cette hypothèse, nous proposons un paramètre "coût de navigation" affiné. Par exemple, pour un lien d'un document ds_i vers un document ds_j, on obtient :

$$\beta_{coût} = \frac{\sum_{ds_k \in IA_i} surface(ds_i \longrightarrow ds_k)}{surface(ds_i \longrightarrow ds_j)} \tag{8.46}$$

8.6.4 Contexte et résumé

L'indexation du contexte extrait un vecteur (résumé informationnel), et applique une fonction de filtrage pour éliminer les termes trop peu importants. Les termes sont filtrés pour alléger la taille des index, et leur pondération doit être supérieure à un seuil donné $seuil_{mi}$ (ou $seuil_{ia}$) pour être conservée :

$$\begin{array}{ll} \forall ds_i \in \mathcal{DS}, & \forall t_j, \\ \textit{w-mi-ds}_{ij} & = 0 \quad si \quad \textit{w-mi-ds}_{ij} < seuil_{mi} \\ & = \textit{w-mi-ds}_{ij} \quad sinon \\ \textit{w-ia-ds}_{ij} & = 0 \quad si \quad \textit{w-ia-ds}_{ij} < seuil_{ia} \\ & = \textit{w-ia-ds}_{ij} \quad sinon \end{array} \tag{8.47}$$

8.7 Indexation d'un hyperdocument en contexte

La finalité de la phase d'indexation est l'obtention des index d'hyperdocuments en contexte. Ainsi, l'index de hd est composé de l'index $\vec{ds}$ du document structuré ds, l'index du contexte $Contexte$ de ds (qui est, de fait, le contexte de hd), de l'index $\vec{ch}^k$ de tous les chemins de lecture ch^k qui parcourent hd, et enfin de l'index $contexte^k$ du contexte de chacun de ces chemins. L'équation 8.48 présente un index complet d'hyperdocument en contexte :

$$\begin{array}{ll} \forall hd_i \in \mathcal{HD},\ index \quad hd_i & = < \vec{ds}_i, contexte_i, \{\vec{ch}_i^k\}, \{contexte_i^k\} > \\ & i)\ \vec{ds}_i \\ & ii)\ contexte_i = (aut_i, ray_i, \vec{\textit{méta-info}}_i, \vec{\textit{info-acc}}_i) \\ & iii)\ \vec{ch}_i^k \\ & iv)\ contexte_i^k = (aut_i^k, ray_i^k, \vec{\textit{méta-info}}_i^k, \vec{\textit{info-acc}}_i^k) \end{array} \tag{8.48}$$

8.8 Indexation : synthèse

Dans ce chapitre, nous avons présenté l'extraction de la sémantique des documents considérant le contenu, la structure logique, le cheminement et le contexte, à partir de la modélisation du Web présentée dans le chapitre 7.

Dans un premier temps, nous avons décrit la construction, à partir des documents atomiques a, des documents structurés, des hyperdocuments et des chemins de lecture. L'indexation des a est atomique et sert de base à l'indexation des ds, qui met en jeu des règles de composition de l'information. L'indexation des ch est inspirée de théories de progression thématique dans un texte, avec un algorithme itératif qui simule une lecture linéaire d'un chemin. Ces premières étapes produisent les vecteurs $\vec{a}$, $\vec{ds}$ et $\vec{ch}$. Or, un hyperdocument est un document structuré ds auquel est associé un ensemble de chemin de lecture $Chemins$. L'index d'un hyperdocument est donc un vecteur de document structuré $\vec{ds}$ auquel est associé un ensemble de vecteurs de chemins de lecture $\{\vec{ch}\}$.

Dans un second temps, nous avons décrit la mise en contexte des documents. A chaque niveau d'indexation (a, ds, hd, ch), les trois aspects du contexte sont intégrés comme présenté dans le chapitre 6 : la popularité, la méta-information et l'information accessible. La popularité d'un document est représentée par deux variables : l'autorité aut et le rayonnement ray. La méta-information est représentée par un vecteur $\vec{méta\text{-}info}$, et l'information accessible par un autre vecteur $\vec{info\text{-}acc}$. On note que le contexte d'un hyperdocument est le contexte du document structuré associé.

Nous avons donc intégré à l'indexation les trois aspects de la structure du Web : la composition, le cheminement et la référence. La composition et le cheminement permettent de produire les index des documents structurés et des chemins de lecture, en considérant les caractéristiques spécifiques des relations sous-jacentes. La référence permet d'extraire et d'indexer le contexte des documents.

Les trois composantes du contexte (cf. chapitre 6) sont intégrées au modèle de RI : la popularité, la méta-information et l'information accessible. La popularité est extraite de l'analyse de la connectivité du réseau de liens (cf. *PageRank*, *Hubs et Authorities*, etc.) provenant des méthodes de propagation de popularité ou de pertinence. La méta-information apporte un éclairage supplémentaire pour la lecture d'un document, et permet d'aider à son indexation, selon le même principe que celui utilisé par Brin avec la propagation des ancres [Brin et al.98]. Enfin, l'information accessible représente l'information que le lecteur pourra consulter par navigation à partir d'un document.

La présentation de la phase d'indexation nous amène à la phase d'interrogation, dont l'objectif est de retrouver des documents pertinents pour un besoin d'utilisateur en exploitant les composantes des index de documents.

8.9 Interrogation et besoin de l'utilisateur

Le modèle d'hyperdocuments proposé permet de décrire la structure de l'information du Web et de l'intégrer à l'extraction du contenu sémantique des documents. Ainsi, le SRIS (Système de Recherche d'Information Structurée) basé sur ce modèle a à sa disposition tous les éléments pour proposer une interrogation structurée complexe. Toutefois, il est primordial de conserver la plus grande simplicité d'interrogation possible. Dans le chapitre 4, nous avons évoqué l'avantage que possède un SRI classique, qui permet d'interroger en langage naturel, par rapport à une Base de Données proposant un langage de requête structurée. Cet avantage doit être conservé pour permettre une interrogation structurée qui ne nécessite pas de connaissance *a priori* de la structure des documents, et qui ne nécessite pas non plus la maîtrise d'un langage d'interrogation structuré complexe et difficile à mettre en œuvre.

Nous avons donc choisi de définir un langage de requête simplifié ayant un pouvoir d'expression réduit par rapport à ce qu'offre le modèle d'hyperdocuments. Le modèle de requête représente les principaux aspects du besoin de l'utilisateur, et le système est ensuite chargé d'assurer une interrogation structurée par le biais de la fonction de correspondance, pour profiter de la richesse de l'index. Ainsi, certains paramètres de la fonction de correspondance

sont des paramètres du système, alors que les besoins les plus intuitifs et les plus faciles à décrire sont définis dans le modèle de requête. En relation avec la structure du Web que nous modélisons, nous distinguons quatre axes du besoin de l'utilisateur :

Thème : l'utilisateur doit décrire le thème de l'information recherchée. Il peut le faire à l'aide de mots-clés, ou en langage naturel, à la manière d'un SRI classique.

Contexte : l'utilisateur peut préciser le contexte dans lequel se trouve l'information recherchée. Ici, le contexte désigne l'ensemble des documents qui référencent un document donné, c'est-à-dire la méta-information (la deuxième composante du contexte, l'information accessible, est prise en compte avec l'axe "Focus").

Granularité : l'utilisateur peut donner sa préférence en matière de granularité des résultats : est-il prêt à consulter un document structuré, voire un site entier, ou préfère-t-il une information concentrée dans un document simple ? En d'autres termes, quelle longueur de chemin préfère-t-il ?

Focus : l'utilisateur peut indiquer s'il est prêt à naviguer pour consulter l'information pertinente : est-ce qu'il préfère un document atomique (pertinence focalisée) ou une *zone de pertinence* (pertinence défocalisée) ?

8.10 Modèle de requête

Ces quatre aspects du besoin de l'utilisateur sont intégrés au modèle de requête, que nous définissons par un 4-uplet :

$$Req = < \vec{contenu}_{req}, \mu_{gran}, \vec{\textit{méta-info}}_{req}, \mu_{nav} > \tag{8.49}$$

Thème : le thème de l'information recherchée est représenté par un vecteur $\vec{contenu}_{req}$.

Contexte : la méta-information de l'information recherchée est représentée par un vecteur $\vec{\textit{méta-info}}_{req}$.

Granularité : la granularité recherchée est choisie à l'aide d'une réglette graduée, représentée par un paramètre $\mu_{gran} \in [0, 1]$.

Focus : le caractère focalisé ou défocalisé de la recherche est choisi à l'aide d'une réglette graduée (cf. figure 8.10) représentée par un paramètre $\mu_{nav} \in [0, 1]$, qui indique l'intérêt de l'utilisateur pour l'information accessible.

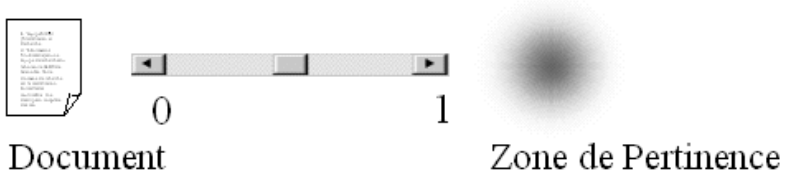

FIG. 8.10 – Choix du focus (importance de l'information accessible).

8.11 Fonction de correspondance

Le rôle de la fonction de correspondance est de comparer les différents éléments d'un hyperdocument et d'une requête afin d'évaluer une "pertinence système" la plus proche possible de la "pertinence utilisateur". La fonction de correspondance évalue la pertinence d'un index d'hyperdocument ($hd_i = < ds_i, chemins_i, \mathcal{R}^i_{chem} >$) pour une requête Req en combinant les différents éléments disponibles :

$$\begin{aligned} hd_i &= < \vec{ds_i}, contexte_i, \{\vec{ch_i^k}\}, \{contexte_i^k\} > \\ & i)\ \vec{ds_i} \\ & ii)\ contexte_i = (aut_i, ray_i, \vec{\textit{méta-info}}_i, \vec{\textit{info-acc}}_i) \\ & iii)\ \vec{ch_i^k} \\ & iv)\ contexte_i^k = (aut_i^k, ray_i^k, \vec{\textit{méta-info}}_i^k, \vec{\textit{info-acc}}_i^k) \\ Req &= (\vec{contenu}_{req}, \mu_{gran}, \vec{\textit{méta-info}}_{req}, \mu_{nav}) \end{aligned} \tag{8.50}$$

8.11.1 Objectifs de l'interrogation

L'objectif de l'interrogation est de retrouver les meilleurs chemins de lecture en contexte par rapport aux quatre aspects du besoin de l'utilisateur que nous considérons. Pour cela, le filtre de la première étape ne doit pas éliminer d'hyperdocuments contenant des chemins pertinents. Ensuite, la deuxième étape doit retrouver les chemins les plus pertinents : un chemin de lecture pertinent est un chemin qui contient un maximum d'informations pertinentes, et qui est placé dans un contexte pertinent. De plus, la taille du chemin doit être d'autant plus réduite que la granularité demandée est faible. Enfin, la pertinence de l'information accessible par rapport à la requête est considérée avec d'autant plus de force que le paramètre de focus de la recherche est grand.

8.11.2 Etapes de filtrage et de recherche

Les étapes de l'évaluation de la pertinence s'intéressent au contenu, aux chemins de lecture et au contexte des hyperdocuments. La dualité document structuré/hypertexte des documents du Web est utilisée au cours des deux étapes de l'interrogation :

Filtrer les hyperdocuments : dans un premier temps, le calcul de la pertinence des documents indexés comme des documents structurés est utilisé comme un filtre préliminaire qui permet d'éliminer la plus grande partie des documents non pertinents.

Retrouver des hyperdocuments en contexte : dans un second temps, les hyperdocuments sélectionnés sont examinés de plus près, et on recherche le meilleur "chemin de lecture en contexte" par rapport à la requête. Pour cela, on utilise les documents indexés comme des hyperdocuments comportant un ou plusieurs chemins de lecture.

Ainsi, la pertinence d'un hyperdocument hd pour une requête Req est calculée comme présenté dans l'équation 8.51. L'étape de filtrage calcule la pertinence $Pert_{ds}$ de hd indexé du point de vue des documents structurés, et vérifie si cette pertinence atteint le seuil $seuil_{ds}$. Ensuite, l'étape de recherche calcule la pertinence $Pert_{chem}$ d'un hyperdocument qui a passé le filtre, c'est-à-dire la pertinence du plus pertinent des chemins de lecture de hd. Cela consiste à trouver le maximum de la pertinence de hd indexé comme un ensemble de chemins de lecture.

$$Pert(Req, hd) = \begin{cases} 0 \ si \ Pert_{ds}(Req, ds) \ < \ seuil_{ds} \\ max_{ch^k \in Chemins}(Pert_{chem}(Req, ch^k)) \ sinon \end{cases} \quad (8.51)$$

8.11.3 Filtrer les hyperdocuments en contexte

La première étape de l'interrogation consiste à évaluer la pertinence d'un hyperdocument hd vu comme un document structuré, en considérant la méta-information et l'information accessible. L'indexation des ds, basée sur la composition des contenus est une indexation qui agrège les contenus en résumant l'information (en raison du seuil $seuil_{comp}$), et le vecteur $\vec{ds}$ ainsi obtenu est réduit. Cela constitue un premier filtre qui élimine les hyperdocument hd dont le "résumé" $\vec{ds}$ est peu pertinent.

L'évaluation de la pertinence entre hd vu comme un document structuré et la requête Req suit les étapes ci-dessous, en utilisant les pertinences intermédiaires $Pert_{locale}$, $Pert_{\textit{info-acc}}$ et $Pert_{\textit{méta-info}}$. Ces trois pertinences concernent trois aspects de l'information du Web considérés dans le modèle d'hyperdocuments. La pertinence $Pert_{locale}$ est la pertinence locale du document ds pour son contenu, qui est représenté par le vecteur intermédiaire $\vec{local}$. Les pertinences $Pert_{\textit{info-acc}}$ et $Pert_{\textit{méta-info}}$sont les pertinences de l'information accessible et de la méta-information de l'hyperdocument par rapport à la requête.

Filtre-1) Intégration de la méta-information de ds (paramètre du système α_{mi}) :

$$\vec{local} = (1 - \alpha_{mi}) * \vec{ds} + \alpha_{mi} * \vec{\textit{méta-info}} \quad (8.52)$$

Filtre-2) Similarité vectorielle pour calculer la pertinence focalisée de ds :

$$Pert'_{locale}(Req, ds) = sim_{vec}(\vec{contenu_{req}}, \vec{local}) \quad (8.53)$$

Filtre-3) Intégration de la granularité et de l'autorité de ds (paramètres aut, μ_{gran} et *gran*, cf. équation 8.64) :

$$Pert_{locale}(Req, ds) = aut * sim_{gran}(\mu_{gran}, \textit{gran}) * Pert'_{locale}(Req, ds) \quad (8.54)$$

Filtre-4) Similarité vectorielle pour calculer la pertinence défocalisée de ds :

$$Pert'_{\textit{info-acc}}(Req, ds) = sim_{vec}(\vec{contenu_{req}}, \vec{\textit{info-acc}}) \quad (8.55)$$

Filtre-5) Utilisation du rayonnement de ds (paramètre du document ray) :

$$Pert_{\textit{info-acc}}(Req, ds) = ray * Pert'_{\textit{info-acc}}(Req, ds) \tag{8.56}$$

Filtre-6) Similarité vectorielle pour calculer la pertinence de la méta-information de ds :

$$Pert'_{\textit{méta-info}}(Req, ds) = sim_{vec}(\vec{\textit{méta-info}}_{req}, \vec{\textit{méta-info}}) \tag{8.57}$$

Filtre-7) Utilisation de l'autorité de ds (paramètre du document aut) :

$$Pert_{\textit{méta-info}}(Req, ds) = aut * Pert'_{\textit{méta-info}}(Req, ds) \tag{8.58}$$

Filtre-8) Intégration des trois aspects de la pertinence (paramètre utilisateur μ_{nav} et paramètre du système $\alpha_{contexte}$) :

$$\begin{aligned} Pert_{ds}(Req, ds) = \quad & (1 - \alpha_{contexte}) \left\{ \begin{array}{r} (1 - \mu_{nav}) * Pert_{locale}(Req, ds) \\ +\mu_{nav} * Pert_{\textit{info-acc}}(Req, ds) \end{array} \right\} \\ & +\alpha_{contexte} * Pert_{\textit{méta-info}}(Req, ds) \end{aligned} \tag{8.59}$$

On remarque le double rôle de la méta-information, qui est présente aux étapes Filtre-1 et Filtre-6. Elle est prise en compte comme une description supplémentaire de ds à l'étape Filtre-1 (cf. équation 8.52), et comme une information du contexte à l'étape Filtre-6 (cf. équation 8.57). Dans le premier cas, la méta-information est considérée comme décrivant le contenu de ds, en conséquence de quoi elle est intégrée à l'information locale. Le paramètre du système α_{mi} permet de fixer l'importance de la méta-information par rapport à l'information locale du document ($\vec{ds}$). Cela permet d'évaluer $Pert_{locale}(Req, ds)$, la pertinence focalisée de ds, en utilisant la fonction de similarité vectorielle sim_{vec}. La fonction sim_{vec}, très utilisée en RI, calcule le cosinus entre deux vecteurs pour évaluer leur similarité :

$$sim_{vec}(a_i, a_k) = \frac{\sum_{t_j} w_{ij} * w_{k_j}}{\sqrt{\sum_{t_j} w_{ij}^2 + w_{kj}^2}} \tag{8.60}$$

La pertinence $Pert_{locale}(Req, ds)$ est corrigée par le score d'autorité du document (aut, cf. équation 8.54), et par la similarité de la granularité *gran* du document par rapport au paramètre μ_{gran} de la requête (cf. équation 8.54).

Ensuite, l'information accessible est prise en compte en fonction du caractère défocalisé de la recherche, c'est-à-dire du paramètre μ_{nav} (cf. étape Filtre-4, équation 8.55). La pertinence défocalisée $Pert_{\textit{info-acc}}(Req, ds)$ du vecteur d'information accessible par rapport au vecteur de la requête $\vec{\textit{méta-info}}_{req}$ est calculée à l'aide de la fonction sim_{vec}, puis corrigée en fonction du score de rayonnement ray de ds (cf. étape Filtre-5, équation 8.56).

Enfin, la méta-information de ds est prise en compte en fonction du paramètre système $\alpha_{contexte}$ (cf. étape Filtre-6, équation 8.57). La pertinence $Pert_{\textit{méta-info}}(Req, ds)$ du vecteur de méta-information $\vec{\textit{méta-info}}$ par rapport au vecteur de méta-information $\vec{\textit{méta-info}}_{req}$ de

la requête est calculée à l'aide de la fonction sim_{vec}, puis corrigée en fonction du score d'autorité ray de ds (cf. étape Filtre-8, équation 8.59).

L'étape de filtrage de l'interrogation est résumée par l'équation 8.61 :

$$
\begin{aligned}
&Pert_{ds}(Req, ds) = \\
&\quad (1-\alpha_{contexte}) \quad * \left\{ \begin{array}{ll} & (1-\mu_{nav}) \\ * & aut * sim_{gran}(\mu_{gran}, \textit{gran}) \\ * & sim_{vec}\left\{ \vec{contenu}_{req}, \left\{ \begin{array}{r} (1-\alpha_{mi}) * \vec{ds} \\ +\alpha_{mi} * \vec{\textit{méta-info}} \end{array} \right\} \right\} \\ & \\ + & \mu_{nav} * ray * sim_{vec}(\vec{contenu}_{req}, \vec{\textit{info-acc}}) \end{array} \right\} \\
&\quad + \alpha_{contexte} \quad * sim_{vec}(\vec{\textit{méta-info}}_{req}, \vec{\textit{méta-info}})
\end{aligned}
\tag{8.61}
$$

Pour pouvoir retrouver des hd pertinents pour leur contexte, il est nécessaire de faire intervenir la méta-information et l'information accessible dès la première étape de l'interrogation (avec les paramètres aut et ray associés). La popularité joue un rôle au niveau du contenu des documents, pour donner plus d'importance aux documents populaires (étape Filtre-1), et au niveau de la méta-information à laquelle elle est associée (étape Filtre-6). Le rayonnement, combiné au paramètre μ_{nav}, est utilisé pour donner de l'importance à l'information accessible d'un hyperdocument. Ainsi, un paramètre μ_{nav} élevé privilégie l'information accessible aux dépens du contenu du chemin et de la méta-information.

8.11.4 Retrouver de l'hyperinformation

Il s'agit ensuite, pour chaque hyperdocument hd qui a passé le filtre, de trouver le meilleur chemin pour le consulter, en fonction de la requête. Pour cela, la fonction de correspondance $Pert_{chem}(Req, ch^k)$ est une version étendue de la fonction de filtrage, qui évalue la pertinence d'un chemin de lecture. Cette pertinence est utilisée pour évaluer la pertinence de hd vu comme un ensemble de chemins de lecture, c'est-à-dire le maximum de la pertinence des chemins.

Le calcul de pertinence $Pert_{chem}(Req, ch^k)$ fait intervenir l'index des chemins de lecture $\vec{ch}^k$ ainsi que les vecteurs $\vec{\textit{méta-info}}^k$ et $\vec{\textit{info-acc}}^k$ de méta-information et d'information accessible, la granularité *gran* et les scores de popularité aut^k et ray^k des chemins. Cette deuxième étape s'applique sur tous les chemins de lecture d'un hyperdocument (d'où l'importance de la première étape pour sélectionner les meilleurs candidats dont il faut évaluer la pertinence plus finement).

Les 8 étapes de la phase de recherche sont en tout point identiques aux 8 étapes de filtrage de l'évaluation de $Pert_{ds}(Req, ds)$. L'évaluation de $Pert_{chem}(Req, ch^k)$ est résumée par l'équation 8.62 :

$$
\begin{aligned}
&Pert_{ch}(Req, ch^k) = \\
&\quad (1-\alpha_{contexte}) \quad * \left\{ \begin{array}{ll} & (1-\mu_{nav}) \\ * & aut^k * sim_{gran}(\mu_{gran}, \textit{gran}) \\ * & sim_{vec}\left\{ \vec{contenu}_{req}, \left\{ \begin{array}{r} (1-\alpha_{mi}) * \vec{ch}^k \\ +\alpha_{mi} * \vec{\textit{méta-info}}^k \end{array} \right\} \right\} \\ & \\ + & \mu_{nav} * ray^k * sim_{vec}(\vec{contenu}_{req}, \vec{\textit{info-acc}}^k) \end{array} \right\} \\
&\quad + \alpha_{contexte} \quad * sim_{vec}(\vec{\textit{méta-info}}_{req}, \vec{\textit{méta-info}}^k)
\end{aligned}
\tag{8.62}
$$

La fonction sim_{gran} évalue si le chemin de lecture ch^k est d'une granularité proche du besoin de l'utilisateur exprimé par μ_{gran}. Ce document peut être un document structuré ds à l'étape de filtrage, ou un chemin de lecture ch^k à l'étape de recherche.

La granularité d'un chemin telle que nous l'avons définie dans la section 8.4.5 est le pourcentage de la taille du chemin par rapport à la taille du plus grand chemin du corpus. On considère donc que le paramètre μ_{gran} exprime le besoin de l'utilisateur par rapport à ce chemin maximum, ce qui nous amène à définir une fonction de similarité de la granularité :

$$
\begin{array}{c} \forall ch \in \mathcal{CH}, \quad sim_{gran}(\mu_{nav}, gran) = 1 - |\mu_{nav} - gran| \\ avec \ \ \mu_{nav} \in]0..1] \ \ et \ \ gran \in]0..1] \end{array}
\tag{8.63}
$$

Cette similarité doit être adaptée au cas des documents structurés, pour lesquels la granularité appartient à l'intervalle $[1..nb_{niv}]$:

$$
\begin{array}{c} \forall ds \in \mathcal{DS}, \quad sim_{gran}(\mu_{nav}, gran) = 1 - |\mu_{nav} - \frac{gran}{nb_{niv}}| \\ avec \ \ \mu_{nav} \in]0..1] \ \ et \ \ gran \in]0..nb_{niv}] \end{array}
\tag{8.64}
$$

8.12 Conclusion

La phase d'interrogation permet de calculer la pertinence des hyperdocuments par rapport à une requête permet d'exprimer différentes facettes du besoin de l'utilisateur : le thème, le contexte, la granularité et le focus. Cette interrogation réalise dans un premier temps un filtrage pour éliminer les hyperdocuments peu pertinents du point de vue de l'index des documents structurés (index $\vec{ds}$). Ensuite, l'interrogation recherche les chemins de lectures les plus pertinents qui parcourent les hyperdocuments pré-selectionnés.

La phase d'interrogation, outre les éléments de la requête définie par l'utilisateur, se base sur le paramètre du système $seuil_{ds}$, qui est le seuil de pertinence des hyperdocuments évalués comme des documents structurés. Cette phase utilise aussi les paramètres du système α_{mi}, $\alpha_{contexte}$ ainsi que la fonction de similarité vectorielle sim_{vec} et la fonction de similarité de granularité sim_{gran}.

Troisième partie

Mise en œuvre : un Système de RI Structurée sur le Web

Chapitre 9

Expérimentations et évaluation

9.1 Objectifs

Nous souhaitons expérimenter et valider l'approche théorique proposée. Le modèle d'hyperdocuments en contexte comporte quatre principaux axes : la description du contenu (documents atomiques), de la structure hiérarchique (relation de composition, documents structurés), de la structure hypertexte (relation de cheminement, chemins de lecture) et du contexte (relation de référence, méta-information et information accessible).

1) Documents atomiques : l'expérimentation de l'indexation des documents atomiques a pour but d'obtenir une indexation "de base" de bonne qualité. Nous l'avons optimisée afin d'éviter le scénario d'une amélioration spectaculaire des résultats qui seraient davantage une compensation des médiocres performances de l'indexation "de base" qu'une amélioration significative de la qualité du système.

2) Documents structurés : l'expérimentation de l'indexation des documents structurés a un double objectif :

- **2.1) Importance du cotexte textuel :** nous désirons déterminer si le cotexte textuel d'un document est important pour son indexation et sa recherche, ou si au contraire il est tout aussi facile à retrouver indépendamment de tout contexte.
- **2.2) Schéma de pondération :** il faut également évaluer la pondération multi-niveaux que nous proposons comme adaptation aux documents structurés du schéma classique de $tf.idf$, par rapport à une propagation plus simple de l'information.

3) Chemins de lecture : l'expérimentation des chemins de lecture a pour objectif de répondre aux questions suivantes :

- **3.1) Sous-ensemble :** est-il intéressant de considérer un chemin parmi les atomes d'un document structuré, au lieu d'indexer le document en entier ?
- **3.2) Ordre :** quel est l'intérêt de considérer l'ordre des documents atomiques d'un chemin à l'indexation ?

4) Contexte : enfin, il faut aussi expérimenter et valider l'utilisation du contexte, c'est-à-dire de la méta-information et de l'information accessible (relation de référence).

Dans ce chapitre, nous commencerons par évoquer les possibilités classiques d'évaluation d'un SRI dans la section 9.2 pour mettre en évidence le fait que les collections de test existantes ne sont pas adaptées à la tâche d'évaluation de notre système.

Puis, après avoir évoqué dans la section 9.3 les campagnes d'évaluation de référence et présenté plus particulièrement la collection de test OFIL de la campagne Amaryllis (que nous avons utilisée dans nos expérimentations), nous dégagerons dans la section 9.3.4 les principes généraux d'une collection de test pour l'évaluation d'un SRI structurée.

Nous discuterons alors des difficultés de la construction manuelle d'une collection de test, et en conséquence nous proposons une méthode simple de construction automatique d'une collection de test structurée (cf. section 9.5), que nous avons mise en œuvre en utilisant la collection OFIL. Nous présenterons ensuite l'ensemble de collections de test construites selon cette approche.

Nous terminerons par les résultats d'expérimentations de notre approche de RI structurée. Dans un premier temps nous présenterons les résultats obtenus avec une collection construite manuellement (pour l'évaluation de l'apport de l'information accessible), et dans un second temps nous présenterons les résultats obtenus avec chacune des collections construites automatiquement (pour l'évaluation des aspects "documents structurés" et "chemins de lecture").

9.2 Évaluation classique d'un SRI

Les modèles de RI classiques disposent depuis longtemps de méthodes d'évaluation des résultats d'un SRI, comme par exemple les méthodes développées au cours de la conférence TREC[1]. Le principal inconvénient de ces méthodes pour expérimenter notre approche est qu'elles sont basées sur la notion de documents atomiques et indépendants, et donc sur la notion d'une *pertinence atomique*.

9.2.1 Pertinence atomique

Un SRI présente généralement une liste de documents, ordonnés selon leur valeur de correspondance. Un document est jugé pertinent par le système relativement à une requête, si la fonction de correspondance entre le document et la requête donne une valeur élevée. Mais un document pertinent pour le système ne l'est pas toujours pour l'utilisateur. On distingue donc la pertinence utilisateur : quand un document est jugé pertinent par l'utilisateur, de la pertinence système : quand un document est jugé pertinent par le système pour une requête.

On appelle *pertinence atomique* la pertinence d'un document non structuré, jugé seulement sur la base de son contenu, donc sans tenir compte du reste du corpus. Cette notion de pertinence n'est pas adaptée à l'évaluation d'un SRI modélisant des documents structurés (ou des chemins de lecture) en contexte.

[1] Text REtrieval Conference : http ://trec.nist.gov

9.2.2 Rappel, précision et courbes de R/P

Un SRI parfait est un système qui, pour toute requête, retrouve **tous** et **uniquement** les documents pertinents pour l'utilisateur à cette requête : la pertinence système est la même que la pertinence utilisateur. On utilise deux critères classiques pour évaluer les performances d'un SRI [Salton71] : le rappel et la précision. Le rappel représente la capacité d'un système à retrouver **tous** les documents pertinents, et la précision représente sa capacité à ne retrouver **que** des documents pertinents. Pour calculer ces deux critères, on utilise l'ensemble $\mathcal{R}$ des documents retrouvés par le système et l'ensemble $\mathcal{P}$ des documents pertinents pour l'utilisateur :

$$\textbf{Rappel} = \frac{||\mathcal{R} \cap \mathcal{P}||}{||\mathcal{P}||} \qquad \textbf{Précision} = \frac{||\mathcal{R} \cap \mathcal{P}||}{||\mathcal{R}||}$$

9.2.3 Collection de test

Pour faire une évaluation statistique de la qualité d'un système, il faut disposer d'une collection de test : typiquement, un corpus de plusieurs milliers de documents et quelques dizaines de requêtes, auxquelles sont associés des jugements de pertinence utilisateur, établis par des experts ayant une grande connaissance du corpus.

On calcule le rappel et la précision sur les résultats renvoyés par le système pour une requête donnée (à chaque document renvoyé). Pour chaque requête, on établit alors une **courbe de Rappel/Précision** (la précision en fonction du rappel) : on choisit n points de rappel $r_1, r_2, ...r_i, ...r_n$, et pour chaque point r_i on prend comme valeur de précision correspondante le maximum de précision obtenue pour tout point de rappel supérieur ou égal à r_i :

$$precision(r_i) = max_{r_j \geq r_i}(precision(r_j)) \tag{9.1}$$

La moyenne de ces courbes permet d'établir un profil visuel de la qualité d'un système, et on calcule souvent la "précision moyenne à n points de rappel" (typiquement : $n = 11$), qui est un critère "résumant" de la qualité du système :

$$AvgPrec_n = \frac{\sum_{i=0}^{n-1}(\textit{Précision} \quad | \quad Rappel = \frac{i}{n})}{n} \tag{9.2}$$

9.2.4 Évaluation d'un SRI sur le Web : la précision comparative

Une limitation de l'évaluation à base de rappel/précision se situe au niveau du rappel. En effet, la quantité de documents présents sur le Web ne permet pas de déterminer quels sont **tous** les documents pertinents pour une requête donnée : il est impossible de porter un jugement sur chacun des documents d'une collection de plusieurs milliards de pages HTML.

Il existe une méthode simple permettant de s'affranchir du *problème du rappel* et de construire des collections de test alors même qu'il n'est pas possible de porter un jugement

de pertinence sur chacun des documents. Il s'agit de la méthode d'évaluation dite de la *précision comparative* (ou *pooling*). En effet, s'il n'est pas possible de calculer le rappel, il est par contre envisageable de calculer la précision à n documents.

Cette méthode consiste à comparer un ensemble de systèmes entre eux. Pour cela, elle se base sur l'hypothèse que tous les documents pertinents existants sont retrouvés par au moins un des systèmes, avec un rang r inférieur à une limite donnée ($r \leq n_p$). Les n_{SRI} SRI que l'on veut évaluer sont alors utilisés comme filtres grossiers pour pouvoir établir les jugements de pertinence sur une sous-collection d'une taille raisonnable. Il suffit de demander à chacun des n_{SRI} SRI les n_p premières pages (par exemple, $n_p = 100$) Web en réponse à une requête, constituant l'ensemble de pages R_{SRI}. On obtient un ensemble R_{retr} de n_{retr} pages, avec $n_{retr} \leq n_{SRI} * n_p$. Les juges peuvent alors consulter chacune de ces pages pour émettre un jugement de pertinence et déterminer les p documents pertinents pour la requête, constituant l'ensemble P_{retr} avec $p \leq n_{retr}$. Pour chaque système s, on pourra alors calculer la précision à n_p documents :

Précision à n_p documents = $\frac{\|R_{SRI} \cap P_{retr}\|}{\|R_{SRI}\|}$

A défaut d'évaluer la qualité d'un système dans l'absolu en considérant la collection dans sa globalité, cette méthode permet de comparer plusieurs systèmes entre eux. Les collections de test pour le Web créées dans le cadre de la piste Web de la conférence TREC que nous mentionnons dans la section suivante sont basées sur la précision comparative.

9.3 Exemples de collections de test

Nous présentons une collection de la conférence TREC, qui est une référence pour l'évaluation des SRI. Nous présentons également la collection OFIL de la campagne d'évaluation Amaryllis[2], que nous avons utilisée pour nos expérimentations. Enfin, nous terminons ce rapide tour d'horizon avec la collection de test *Shakespeare*, dédiée à l'évaluation de SRI dans le contexte des documents structurés.

9.3.1 La piste Web de la conférence TREC

Parmi les différentes *pistes* de TREC, il existe une piste dédiée à l'évaluation de la RI sur le Web : la *piste Web* [Hawking00]. Deux échantillons de respectivement 2 Go et 10 Go on été extraits d'une collection de 100 Go collectée sur le Web. Un jeu de requêtes inspiré de celui utilisé dans un contexte plus classique a été utilisé sur ces collections avec plusieurs SRI de la conférence TREC. Puis, des jugements de pertinence ont été faits sur ces réponses. Un des critères de qualité de ces SRI est la précision à 20 documents, permettant ainsi de calculer la précision comparative.

[2]Campagne d'évaluation Amaryllis : http ://amaryllis.inist.fr/

La collection de 2 Go (WT2g), dont un des objectifs était d'évaluer les méthodes de RI utilisant les liens hypertextes, n'a pas permis de mettre en avant de réels progrès lors de TREC-8. Une raison avancée à cela est que le réseau de liens inter-sites, qui n'est qu'un petit échantillon du réseau réel, est sans doute trop clairsemé pour que les méthodes qui exploitent les liens puissent donner pleine mesure de leur efficacité. La collection de 10 Go (WT10g) a ensuite été créée pour tenter de corriger les défauts de WT2g. Malgré la qualité du réseau de liens de WT10g et le jugement de pertinence ternaire[3] adopté spécifiquement pour détecter une amélioration dûe à l'utilisation des liens, les résultats sont restés décevants lors de TREC-9 [Hawking00] comme de TREC-10 [Hawking et al.01a].

Malgré l'importance accordée aux liens, les jugements de pertinence de la collection proposée sont toujours basés uniquement sur le contenu des pages, sans tenir compte de leur voisinage, à la manière des collections de test classiques. Les collections WT2g et WT10g sont "orientées liens", mais n'en conservent néanmoins qu'une partie, et se basent sur la notion de pertinence classique (ternaire) en considérant uniquement le contenu des documents. Ce problème a été évoqué par Craswell [Craswell et al.01] qui propose de rechercher uniquement des "pages principales" de sites, ce qui sous-entend que la page est intéressante parce qu'elle permet de visiter un site. Il existe maintenant une piste "*homepage finding*" depuis TREC-10 [Hawking et al.01a]. Avec ce nouveau type de jugement de pertinence, les méthodes utilisant les liens donnent enfin de meilleurs résultats [Craswell et al.01]. Ce type d'évaluation est une avancée vers la définition d'une notion de pertinence adaptée au Web.

9.3.2 La collection OFIL de la conférence Amaryllis

La campagne d'évaluation Amaryllis, organisée par l'INIST[4] a été l'occasion de constituer un corpus francophone dans le style de TREC. Elle propose trois collections de test : OFIL (article du journal *Le Monde*), INIST (résumés scientifiques) et LRSA (monographies sur la culture Mélanésienne). LRSA, bien que structurée, a été jugée trop petite (environ 2 Mo) pour évaluer notre approche. INIST est la plus grosse des trois collections, mais elle est très spécialisée, les documents sont petits et les thèmes scientifiques. Elle est donc moins intéressante que OFIL au niveau du développement progressif de l'information que l'on est susceptible d'y trouver.

Nous avons donc choisi la collection OFIL pour nos expérimentations. Un document de cette collection est un article, purement textuel, auquel sont associés un titre et un identifiant unique. L'article lui-même n'est pas structuré et ne comporte aucune coupure de section ou de paragraphe, ni même de coupure de ligne (un article entier est stocké sur une seule ligne). A titre d'exemple, on trouvera en annexe B le premier document de la collection au format TEI, ainsi que les deux premières requêtes et les jugements de pertinence associés. Le tableau 9.1 présente les principales caractéristiques de cette collection.

[3] Jugement de pertinence ternaire : non pertinent/pertinent/très pertinent.

[4] INIST : Institut National de l'Information Scientifique et Technique.

Documents	
Nombre de documents	11 016
Dont pertinents pour au moins une requête	576
Pertinence globale moyenne ($pert^{glob}_{moy}$)	5,33 %
Taille du corpus TEI/texte	32,9 Mo/30,3 Mo
Nombre de mots	6,27 millions
Nombre de mots de l'index	3,45 millions
Nombre de termes distincts	106 700
Taille moyenne des documents	569 mots, 2,82 Ko
Requêtes	
Nombre de requêtes	26
Nombre de jugements de pertinence	587

FIG. 9.1 – Caractéristiques de la collection OFIL.

Pour faciliter la description, nous appelons $a_1 ... a_n$ les documents de base d'OFIL, avec $n = 11\,016$, et $r_1 .. r_{nbreq}$ les requêtes, avec $nbreq = 26$. On définit la pertinence globale d'un document par rapport à l'ensemble de la collection (toutes requêtes confondues) comme étant le nombre de jugements de pertinence qui lui sont associés :

$$pert^{glob}(a_i) = card(\{req_j \mid pertinent(req_j, a_i)\}) \tag{9.3}$$

On peut alors définir la *pertinence globale moyenne* des documents de la collection :

$$pert^{glob}_{moy}(OFIL) = \frac{\sum_{i \in [1..n]} (pert^{a_i}_{glob})}{n} = 5,3\% \tag{9.4}$$

On remarque que la *pertinence globale moyenne* des documents d'OFIL est de 5,3 %, c'est-à-dire 587 documents jugés pertinents sur 11 016.

9.3.3 La collection Shakespeare

Il existe une collection de test spécifiquement construite pour évaluer des SRI sur des documents structurés : la collection *Shakespeare* qui a été développée par le groupe QMIR[5] de l'université Queen Mary (University of London) dans le cadre du projet d'indexation de document structuré *FOCUS*[6].

Cette collection est composée de 37 pièces de théâtre de Shakespeare. Chaque pièce est structurée en actes/scènes/dialogue/ligne, et les jugements de pertinence associés sont portés sur le niveau le plus bas de granularité, celui des lignes, pour les 43 requêtes de la collection. Différentes stratégies de "remontée de pertinence" peuvent être utilisées pour

[5]Groupe QMIR : http ://qmir.dcs.qmw.ac.uk/
[6]Projet FOCUS : http ://qmir.dcs.qmul.ac.uk/Focus/

obtenir la pertinence d'un document structuré (un dialogue, une scène ou un acte) en fonction de la pertinence de ses descendants.

Un point particulièrement intéressant a été développé dans cette collection : la notion de "points d'entrée" (BEP : "*Best Entry Points*"). En complément des jugements de pertinence atomiques, des documents de tous les niveaux de granularité ont été identifiés comme étant pertinents pour une requête : ce sont des BEPs.

Cette collection est cependant très particulière, étant composée de pièces de théâtre. La structuration est très détaillée, jusqu'à la ligne, et le dialogue y est omniprésent. Et surtout, on n'y retrouve pas l'aspect "développement progressif et cohérent de l'information" qui est important dans notre travail. Nous n'avons donc pas opté pour cette collection, qui est adaptée à l'évaluation de SRI dans le contexte de documents structurés, mais qui n'est pas adaptée au cas des chemins de lecture.

9.3.4 Limites des collections de test classiques

Les méthodes d'évaluation des SRI ont été développées dans le contexte des modèles classiques de RI, et donc pour des collections de documents atomiques, non structurés et indépendants. Dans le contexte du Web, cela pose de nombreux problèmes :

L'hétérogénéité des documents du Web, dans son contenu comme dans sa présentation, ne permet pas d'établir un jugement de pertinence "universel", valable pour tous les documents.

Granularité : la granularité des documents n'est pas prise en compte. Cela empêche de juger comme pertinent un document d'une granularité plus importante ou plus faible que la granularité adoptée par le système.

Hypertexte : il n'est pas suffisant d'évaluer un document uniquement en considérant son contenu, il est aussi nécessaire de considérer l'espace d'information auquel le document permet d'accéder. Par exemple, une page Web constituée exclusivement de liens pourra ne contenir que très peu d'information, dans le sens ou peu de termes pertinents font partie du contenu textuel pur du document : la pertinence du document sera jugée faible. Mais si cette liste de liens représente une "compilation" soigneuse des meilleurs pointeurs traitant du thème de la requête, alors cette page devrait être jugée comme étant très pertinente.

Pertinence binaire : un jugement de pertinence binaire (un document est pleinement pertinent ou ne l'est pas du tout) est utilisé dans les collections de test classiques. Cette problématique est récurrente dans notre domaine, mais comme le soulignent Greisdorf dans [Greisdorf et al.99] ou Mizzaro dans [Mizzaro01], il est très utile d'utiliser un jugement de pertinence non binaire. Cela nous semble d'autant plus vrai dans le contexte hétérogène du Web.

Les inconvénients majeurs des méthodes d'évaluation classiques de SRI sont l'atomicité des jugements de pertinence et l'indépendance des documents dans le jugement de pertinence. En effet, la consultation de documents dans un hypertexte ne se fait pas de manière

atomique, la pertinence ne doit pas être atomique. Un document est jugé pertinent pour son contenu uniquement, sans tenir compte de son voisinage. La piste de TREC qui propose de rechercher uniquement des “pages principales” de sites est intéressante, mais n’offre toujours pas de pertinence structurée et ne tient pas compte de l’ordre des pages ni des relations entre elles.

Il n’existe donc pas à notre connaissance de collection de test réellement adaptée à nos besoins d’expérimentation et d’évaluation. C’est pour cette raison que nous proposons de construire une telle collection, soit manuellement, soit automatiquement par adaptation d’une collection existante.

9.4 Construction manuelle : la collection CLIPS

La tâche de construction manuelle d’une collection de test classique est déjà extrêmement lourde et dans notre cas la complexité de la notion de pertinence rend la tâche encore plus difficile. Dans cette section, nous discutons brièvement de ce problème, et nous décrivons une collection de test que nous avons développée afin d’expérimenter un aspect de notre modèle (l’information accessible). Ces expérimentations sont présentées dans la section 9.4.3.

9.4.1 Méthode de construction

La construction manuelle d’une collection de test comporte les étapes suivantes :

Choix du corpus : le choix des documents repose sur le type de collection à construire : des pages Web, des documents structurés, des documents techniques, des articles de journaux, des articles scientifiques, etc.

Choix des requêtes : les requêtes doivent être représentatives du besoin de l’utilisateur auquel est censé répondre le système. On peut choisir les requêtes parmi une collection de requêtes “réelles” (par exemple les *fichiers de log* des moteurs de recherche), ou en s’inspirant des documents (auquel cas il faut maîtriser les domaines abordés).

Définir un critère de pertinence : selon le type d’indexation ou de recherche que l’on cherche à évaluer, ou selon le type de besoin auquel le système doit répondre, il faut définir un critère de pertinence. Par exemple, la pertinence d’une page peut prendre en compte son information accessible.

Jugements manuels : le juge (ou, le plus souvent, plusieurs juges) doit passer en revue l’ensemble du corpus, et examiner chaque document en portant un jugement de pertinence. Le mode de lecture doit prendre en compte les critères de la pertinence : par exemple, une pertinence prenant en compte l’information accessible nécessite de prendre connaissance du contenu d’une page, puis de naviguer dans les pages voisines pour vérifier si l’information y est pertinente.

9.4.2 Construction de la collection CLIPS

Dans [Gery99], nous présentons la mise en œuvre de l'aspect "information accessible" du modèle d'hyperdocuments, avec l'évaluation du prototype SmartWeb sur une collection de test construite manuellement à partir du site Web du laboratoire CLIPS[7]. Le prototype SmartWeb est basé sur un SRI de référence, le système **SMART**, qui a été développé à l'Université de Cornell[8] et dont le code source est disponible gratuitement[9]. En annexe D, nous présentons une copie d'écran de l'interface d'interrogation de SmartWeb, qui est accessible sur le Web[10].

a) Choix d'un corpus

En raison de l'effort demandé par l'évaluation d'un jugement de pertinence, et bien que des expérimentations aient porté sur un corpus de 40 000 pages de l'IMAG, l'évaluation de SmartWeb que nous présentons a été faite sur le corpus IMAG restreint aux 2 500 pages du laboratoire CLIPS. On y trouve des manuels techniques, des documents scientifiques, et même des pages sur le cinéma.

b) Choix des requêtes

Nous avons expérimenté le système SmartWeb à l'aide d'un petit nombre de requêtes traitant de sujets divers. Par exemple : "Une présentation du laboratoire CLIPS", "La musique dans les films de François Truffaut", etc.

c) Définir un critère de pertinence

Pour chacune de ces 12 requêtes, un ou plusieurs documents ont été jugés comme étant pertinents. L'objectif de l'expérimentation était de valider notre approche permettant de retrouver de l'information accessible. Nous avons donc défini la pertinence de la manière suivante : "*un document est un document pertinent s'il est pertinent pour son contenu, mais aussi s'il permet d'accéder en une action de navigation, à d'autres documents pertinents*".

d) Jugements manuels

A la différence du jugement de pertinence d'un simple document textuel, il est nécessaire pour l'évaluation de SmartWeb d'établir ces jugements en considérant le contenu du document et le contenu des documents accessibles par navigation. Cette tâche est fastidieuse et nécessite une bonne connaissance du corpus. En effet, il existe dans cette collection 6,79 liens (non typés) par page Web : le nombre de "pages vues" pour établir chaque jugement peut aller jusqu'à 17 000 (= 6,79 * 2 500) !

[7] Site Web du CLIPS : http ://www-clips.imag.fr
[8] Cornell University : http ://www.cs.cornell.edu/
[9] SMART : ftp ://ftp.cs.cornell.edu/pub/smart/
[10] SmartWeb : http ://smartweb.imag.fr

e) Caractéristiques de la collection CLIPS

Le tableau suivant présente les caractéristiques des différents corpus, avec le nombre moyen de liens par page, la taille du corpus, de l'index, et le temps nécessaire à l'indexation.

Collection	Pondération	Nb Docs	Nb liens	Taille Corpus (texte)	Taille Corpus (index)	Temps Indexation
CLIPS	nnn	2 500	6,79	9,4 Mo	4,1 Mo	45 s
CLIPS InfoAcc	ltc	2 500	6,79	9,4 Mo	17.8 Mo	2 mn 24
IMAG	nnn	40 000	8,46	163 Mo	74 Mo	15 mn
IMAG ltc	ltc	40 000	8,46	163 Mo	221 Mo	25 mn
IMAG InfoAcc	ltc	40 000	8,46	163 Mo	936 Mo	2 h 25

FIG. 9.2 – Caractéristiques des collections CLIPS et IMAG.

Le lecteur trouvera en annexe A une description des fonctions de pondérations proposées dans le système SMART, que nous avons utilisées dans nos expérimentations. Nous présentons par exemple des résultats obtenus avec les fonctions 'nnn' et 'ltc'. Le code utilisé pour identifier une pondération est composé de trois lettres, chacune d'entre elles identifiant une méthode de calcul du df, du tf et de la normalisation. Par exemple, la fonction nnn pondère un terme selon son nombre d'occurrences dans le document, et la lettre 'c' désigne la normalisation utilisant le cosinus.

La taille de l'index et la durée de l'indexation des collections sont beaucoup plus importantes quand l'information accessible est indexée, mais restent relativement raisonnables. En effet, l'espace disque est très bon marché et la durée de l'indexation n'est pas un aspect critique d'un SRI. De plus, l'étude du calcul de l'information accessible, en particulier des seuils à utiliser pour la normalisation, permet de réduire la taille de l'index : ainsi, nous pensons qu'il est envisageable de traiter de très gros corpus avec cette méthode.

9.4.3 Évaluation de l'indexation de l'information accessible

Au cours de ces expérimentations, nous avons simplifié le modèle d'indexation du contexte. Les pages Web sont représentées par deux facettes principales : leur contenu et l'information accessible par navigation, et chaque facette est indexée par un vecteur. A la phase d'interrogation, les deux vecteurs $\vec{contenu}$ et $\vec{\textit{info-acc}}$ sont combinés linéairement, dans une simplification du modèle de requête, selon deux paramètres du système α et β :

$$Pert(Q, a) = cos(\vec{req}, \alpha * \vec{contenu} + \beta * \vec{\textit{info-acc}}) \tag{9.5}$$

L'évaluation est basée sur un jugement de pertinence binaire, défini manuellement : les résultats sont donc présentés de manière classique, par le biais de la précision moyenne et de courbes de rappel/précision. Le tableau récapitulatif de la précision moyenne obtenue

(précision moyenne à 11 points) avec différents paramétrages du système permet d'apprécier les améliorations apportées :

Méthode	Pondération	Anti-dico	Att. Externes	InfoAcc	$AvgPrec_{11}$
1	nnn	non	non	non	6,23%
2	nnn	oui	non	non	16,05%
3	ltc	oui	non	non	33,59%
4	ltc	oui	oui	non	51,77%
5	ltc	oui	oui	oui	61,85%

FIG. 9.3 – SmartWeb : indexation de l'information accessible : résultats.

Ces résultats montrent l'intérêt d'utiliser un anti-dictionnaire et une fonction de pondération "évoluée" comme **ltc**. Et surtout, cette évaluation montre l'intérêt de l'utilisation de l'information accessible, malgré le fait que, pour les besoins de l'expérimentation, son importance par rapport au contenu soit fixée (paramètre $\alpha = \beta = \frac{1}{2}$). Lors d'une recherche avec SmartWeb, l'utilisateur peut faire varier ce paramètre en fonction de ses besoins : en effet, dans le cas des requêtes de la collection de test, le meilleur réglage varie énormément en fonction des requêtes, de $\alpha = 0.1$ à $\alpha = 1$ (avec $\beta = 1 - \alpha$).

La courbe de Rappel/Précision (cf. figure 9.4) permet de visualiser les améliorations des méthodes 3, 4 et 5 par rapport à la méthode 1.

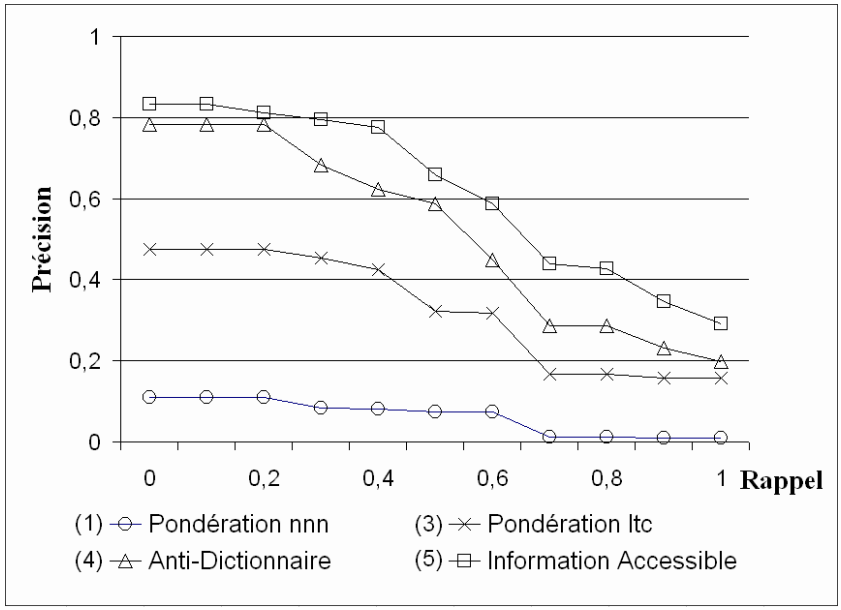

FIG. 9.4 – Courbes de Rappel/Précision : méthodes 1, 3, 4 et 5.

L'évaluation de SmartWeb montre une amélioration importante de la précision des réponses quand les deux facettes sont combinées. Nos expérimentations montrent la faisabilité

et l'utilité de l'information accessible dans le modèle de documents. Ainsi, la structure de l'hypertexte est utilisée pour le calcul de la pertinence de **tous** les documents du corpus. De plus, l'utilisateur a la possibilité, à la phase d'interrogation, de privilégier l'une ou l'autre des facettes, selon qu'il recherche un *document* ou une *zone de pertinence*.

Le calcul de l'information accessible basé sur le coût de navigation doit concilier deux facteurs importants : les répercussions sur la qualité des réponses du système et le coût matériel de l'indexation, d'où l'importance de l'étude de la fonction de normalisation et des seuils à appliquer. L'expérimentation de SmartWeb sur un corpus plus important (de l'ordre du million de pages) pour observer la robustesse du système, permettrait d'analyser son comportement en "grandeur nature". Il serait alors intéressant d'évaluer SmartWeb par rapport à un moteur de recherche du Web, du point de vue de l'utilisateur.

9.5 Construction automatique d'une collection structurée

La construction manuelle d'une "vraie" collection de test pour la RI structurée sur le Web est un projet d'envergure, que nous ne pouvons réaliser dans le cadre d'une thèse. Nous avons donc opté pour la solution d'une construction automatique de la collection de test à partir d'une collection existante (la collection OFIL d'Amaryllis). Cette solution présente l'avantage de pouvoir faire varier à volonté les caractéristiques de la collection afin d'expérimenter différents aspects du système.

Nous présentons dans cette section la méthode de construction de collection, avant de présenter les collections elles-mêmes et les expérimentations que nous avons menées dans les sections suivantes.

9.5.1 Méthode

La construction automatique d'une collection de test se base sur une collection classique existante. Nous avons utilisé la collection OFIL d'Amaryllis. L'objectif est de construire de manière contrôlée une nouvelle collection à partir de la collection existante.

Nous avons expérimenté deux types de construction : la première méthode consiste à considérer les documents de base comme étant les documents atomiques de notre future collection, et la deuxième méthode considère au contraire que les documents de base sont les documents structurés. Dans le premier cas, nous allons donc construire la structure "au-dessus" des documents de base (par agrégation), et dans le second cas nous allons déterminer une structure à l'intérieur de ces documents (par fragmentation). Les collections ainsi fabriquées sont nommées, respectivement, $OFIL_{agreg}$ et $OFIL_{frag}$.

La collection OFIL est purement textuelle et ne comporte aucun lien hypertexte. Cette collection nous sert donc à évaluer la notion de chemin de manière indépendante du contexte. Nos efforts ont donc porté sur la création de documents structurés (donc de relations de composition) et de chemins de lecture (donc de relations de cheminement). Les informations

dont nous disposons, c'est-à-dire les documents existants et des indices comme leur similarité deux à deux, sont plus adaptés à l'extraction des deux premiers types de liens qu'à l'extraction de liens de référence.

9.5.2 Propriétés des collections

La construction d'une collection structurée cohérente nécessite de conserver (ou de recréer) deux propriétés des documents :

Cohérence de la pertinence : les jugements de pertinence de OFIL sont des jugements atomiques. Il faut définir une stratégie de propagation de pertinence cohérente (des documents vers leurs parents pour $OFIL_{agreg}$, et des documents vers leurs fragments pour $OFIL_{frag}$). Par exemple, il faut définir la pertinence d'un document structuré quand la moitié seulement de ses composants ont été jugés pertinents.

Cohérence des documents structurés : les relations de composition créées doivent construire des structures cohérentes. Les documents atomiques d'un document structuré doivent être cohérents les uns par rapport aux autres, par exemple en développant le même thème.

Cohérence des chemins : les relations de cheminement créées doivent construire des chemins cohérents. Par exemple, un chemin de lecture doit suivre une progression logique dans le développement des idées.

Nous proposons un ensemble de mesures sur les documents structurés et les chemins de lecture, avec comme objectif de mettre en avant les propriétés de cohérence d'une collection. Ces propriétés sont au nombre de trois : la propriété de distribution de la pertinence, la propriété de cohésion sémantique et la propriété de la progression thématique. Elles devront être validées sur une collection de test existante : au cours de nos expérimentations, nous nous sommes assurés que ces propriétés concordaient avec celles de la collection OFIL.

a) Paramètres de la construction

Nous utilisons les données suivantes pour paramétrer la création des collections de test :

n **:** nombre de documents de la collection OFIL originale.

$nbreq$ **:** nombre de requêtes de OFIL.

$taille_{ds}$ **:** taille des documents structurés fabriqués, en nombre de documents atomiques.

nb_{ds} **:** nombre de documents structurés fabriqués.

$taille_{ch}$ **:** taille des chemins fabriqués.

$nbchds$ **:** nombre de chemins fabriqués par document structuré.

On rappelle que $taille(ds)$ (respectivement $taille(ch)$) est la fonction qui donne le nombre de documents atomiques d'un document structuré (respectivement d'un chemin).

b) Mesures sur les documents structurés

Nous définissons la pertinence d'un document structuré pour une requête donnée, comme étant la proportion de documents atomiques pertinents pour cette même requête qu'il contient :

$$pertinence(req, ds) = \frac{\parallel a_i \in ds \mid pertinent(req, a_i) \parallel}{taille(ds)} \tag{9.6}$$

Un document de taille $taille_{ds} = 10$ composé de 10 documents pertinents pour la même requête, aura une pertinence de 1. Si seulement la moitié de ses composants sont pertinents, il sera donc jugé comme étant partiellement pertinent et se verra affecter une valeur de pertinence de 0,5. Il s'agit donc d'un choix de pertinence non binaire.

Dans ce cas, la pertinence globale d'un document structuré se calcule de la manière suivante :

$$pert^{glob}(ds) = \sum_{i \in [1..nbreq]} pertinence(req_i, ds) \tag{9.7}$$

On définit aussi la *pertinence globale moyenne* des documents structurés de la collection :

$$pert^{glob}_{moy}(OFIL) = \frac{\sum_{i \in [1..nb_{ds}]} (pert^{glob}(ds_i))}{nb_{ds}} \tag{9.8}$$

Une première condition à la cohérence d'une collection de test est que la pertinence globale moyenne des documents structurés soit voisine de la pertinence globale moyenne des documents atomiques $pert^{glob}_{moy}(OFIL)$. Il s'agit de la propriété de distribution de la pertinence :

Propriété de distribution de la pertinence :

si *coll* est une collection de test, alors : $pert^{glob}_{moy}(coll) \sim pert^{glob}_{moy}(OFIL)$ (9.9)

Nous définissons également la mesure $sim2a2(ds)$, qui est la moyenne des similarités des documents atomiques d'un document structuré comparés deux à deux :

$$sim2a2(ds) = \frac{\sum_{(i,j) \in [1..taille_{ds}]^2} (sim_{vec}(a_i, a_j))}{taille(ds)} \tag{9.10}$$

Cette mesure représente la cohésion sémantique d'un document structuré : quand $sim2a2$ est important, cela signifie que ses composants sont similaires les uns par rapport aux autres. La fonction de similarité sim_{vec} utilisée est le cosinus entre les vecteurs, qui sont indexés à l'aide d'une fonction de pondération $tf * idf$. Nous avons utilisé le schéma de pondération ltc[11]. Par la suite, nous présenterons les valeurs de $sim2a2$ sous la forme de pourcentage, en considérant qu'une similarité $sim2a2$ égale à 1 est maximale (documents identiques).

Enfin, nous définissons la similarité moyenne sur la collection :

[11] Selon la notation du SRI SMART, cf. annexe A.

$$sim2a2_{moy}(OFIL) = \frac{\sum_{i\in[1..n]}(sim2a2(ds_i))}{n} \tag{9.11}$$

Cette mesure nous amène à la deuxième propriété des collections de test : la propriété de cohésion sémantique. Une collection de test doit avoir une similarité moyenne proche de celle de la collection OFIL.

Propriété de cohésion sémantique :

si *coll* est une collection de test, alors : $sim2a2_{moy}(coll) \sim sim2a2_{moy}(OFIL)$ (9.12)

c) Mesure sur les chemins de lecture

Nous définissons $sim2a2ch$, la similarité le long d'un chemin (à distinguer de la moyenne des similarités deux à deux $sim2a2$), comme étant la somme des similarités deux à deux des documents atomiques successifs du chemin :

$$sim2a2ch(ch) = \sum_{arc=(a_i\rightarrow a_j)\in ch}(sim_{vec}(a_i, a_j)) \tag{9.13}$$

Nous définissons la *longueur* d'un chemin comme étant la moyenne des distances (sur les arcs) entre les documents atomiques successifs du chemin. Cette mesure représente la distance sémantique (au sens des vecteurs) parcourue à la lecture d'un chemin, normalisée par le nombre de nœuds (moyenne) :

$$\begin{aligned} longueur(ch) &= \tfrac{1}{taille(ch)} * \textstyle\sum_{arc=(a_i\rightarrow a_j)\in ch}(distance_{vec}(a_i, a_j)) \\ &= \tfrac{1}{taille(ch)} * \textstyle\sum_{arc=(a_i\rightarrow a_j)\in ch}(\left\{ \begin{array}{l} K \;\; si \;\; sim_{vec}(a_i, a_j) = 0 \\ \frac{1}{sim_{vec}(a_i,a_j)} \;\; sinon \end{array} \right\}) \end{aligned} \tag{9.14}$$

La distance utilisée est l'inverse de la similarité, bornée par une constante K (la distance maximale) qui permet de comparer la longueur de deux chemins même s'ils contiennent un arc auquel est affecté une distance théoriquement infinie (quand la similarité entre les documents atomiques est nulle).

On remarque que dans le cas où aucune des similarités entre deux documents atomiques d'un chemin n'est nulle, la longueur est égale à l'inverse de la similarité deux à deux le long du chemin : $longueur(ch) = \frac{1}{sim2a2ch(ch)}$.

Enfin, nous définissons la longueur (respectivement, la similarité deux à deux) moyenne des chemins de la collection :

$$longueur_{moy}(OFIL) = \frac{\sum_{i\in[1..nb_{chds}*nb_{ds}]}(longueur(ch_i))}{nb_{chds} * nb_{ds}} \tag{9.15}$$

$$sim2a2ch_{moy}(OFIL) = \frac{\sum_{i\in[1..nb_{chds}*nb_{ds}]}(sim2a2ch(ch_i))}{nb_{chds} * nb_{ds}} \tag{9.16}$$

La troisième propriété des collections de test impose une longueur courte des chemins, et donc une similarité deux à deux importante, pour refléter le développement progressif des idées. Cela signifie que la rupture sémantique entre deux atomes successifs ne doit pas être brutale. Un collection de test doit avoir une longueur moyenne proche de celle de la collection OFIL :

Propriété de cohérence des chemins :

si *coll* est une collection de test, alors : $longueur_{moy}(coll) \sim longueur_{moy}^{OFIL}$ (9.17)

9.5.3 Construction de collections et évaluation

a) Collections

Nous présentons dans les sections suivantes les différentes collections de test construites automatiquement par agrégation ou par fragmentation. Ensuite, nous présentons les résultats des expérimentations que nous avons menées sur chacun de ces types de collections.

La présentation des collections et des expérimentations associées est organisée de la manière suivante :

Collections par agrégation : les collections $OFIL_{agreg}^{req}$ et $OFIL_{agreg}^{sim}$ (cf. section 9.6).

Expérimentations : la collection $OFIL_{agreg}^{req}$ (cf. section 9.7).

Expérimentations : la collection $OFIL_{agreg}^{sim}$ (cf. section 9.8).

Collection par fragmentation : la collection $OFIL_{frag}$ (cf. section 9.9).

Expérimentations : la collection $OFIL_{frag}$ (cf. section 9.10).

b) Évaluation

L'évaluation de la section 9.9 (construction par fragmentation) est basée sur une pertinence binaire. En conséquence, les résultats sont présentés de manière classique, par le biais de la précision moyenne et de courbes de rappel/précision, comme lors de l'évaluation présentée dans la section 9.4.3.

Par contre, l'évaluation de l'indexation de documents structurés et de chemins des sections 9.7 et 9.8 se base sur un jugement de pertinence non binaire, que nous définissons comme la proportion des documents atomiques pertinents dans le document structuré. En conséquence, même si l'aspect visuel des courbes de rappel/précision reste le même, leur interprétation n'est pas classique. Par exemple, un point de rappel à 0,6 et de précision à 0,3 ne signifie pas que 60% des documents structurés pertinents existants ont été retrouvés et renvoyés parmi 70% d'autres documents structurés non pertinents. Il faut plutôt comprendre : 60% des documents atomiques pertinents existants ont été retrouvés par le biais de leurs documents structurés père, parmi 70% de documents atomiques non pertinents.

La nuance est importante car une conséquence directe est qu'il n'est pas possible, sauf cas particulier, d'obtenir la "courbe de rappel/précision parfaite" (c'est-à-dire avec une précision égale à 1 quel que soit le rappel). En effet, s'il existe un document structuré qui n'est

que partiellement pertinent, alors même le “SRI parfait” n’aura d’autre solution que de le renvoyer, et ainsi renvoyer en même temps que ses documents atomiques pertinents, tous ceux qui ne sont pas pertinents et qui vont diminuer la précision des résultats.

Avec cette définition de la pertinence des documents structurés, nous faisons implicitement l’hypothèse simplificatrice que tous les documents atomiques d’un document structuré sont nécessaires à la compréhension, car ils ont tous une pertinence non nulle.

9.6 Construction par agrégation : $OFIL_{agreg}$

La première étape consiste à construire les documents structurés, et ensuite la deuxième étape se charge de construire des chemins de lecture qui parcourent les nouveaux documents.

9.6.1 Construction de documents structurés

A partir des n documents de OFIL, nous construisons nb_{ds} documents structurés de taille $taille_{ds}$. En choisissant $taille_{ds} = 10$, nous avons préféré limiter nb_{ds} à environ 2 000 documents, pour éviter trop de redondance dans l’utilisation des documents.

Nous avons expérimenté deux méthodes pour tenter de construire des documents cohérents. La première construit les documents en se basant sur les requêtes alors que la seconde se base sur la similarité des documents entre eux.

a) Construction basée sur les requêtes : $OFIL^{req}_{agreg}$

La construction basée sur les requêtes consiste à produire un ensemble de documents à partir de chaque requête. Afin de vérifier la propriété de distribution de la pertinence, nous construisons artificiellement des documents structurés non pertinents (contenant zéro document atomiques pertinent), partiellement pertinents (contenant un ou plusieurs documents atomiques pertinents) ou entièrement pertinents (composés uniquement de documents atomiques pertinents). Ainsi, il existera pour chaque requête le même nombre de documents pertinents et partiellement pertinents. Par exemple, pour une taille $taille_{ds}$ des documents égale à 10 documents atomiques, nous pouvons choisir de construire, pour chaque requête :

- **2** documents structurés comportant **0** document atomique pertinent pour la requête.
- **2** documents structurés comportant **1** document atomique pertinent pour la requête.
- **2** documents structurés comportant **2** documents atomiques pertinents pour la requête.

... ...

- n_i documents structurés comportant nb_i documents atomiques pertinents pour la requête.
- etc.

Les documents atomiques pertinents sont choisis aléatoirement parmi les documents atomiques pertinents pour la requête, en minimisant le nombre de doublons au sein du même document structuré. Les documents atomiques non pertinents sont choisis aléatoirement dans

le reste du corpus. Le nombre de documents structurés pertinents créés par requête et la distribution des documents atomiques pertinents doivent mener à un ensemble de document structurés dont la pertinence moyenne est voisine de la pertinence moyenne des documents atomiques de la collection OFIL :

$$pert^{glob}_{moy}(OFIL^{req}_{agreg}) = \frac{\sum_{nb_i \in 1..taille_{ds}}(n_i * nb_i)}{\sum_{nb_i \in 0..t_{ds}}(n_i * taille_{ds})} \sim pert^{glob}_{moy}(OFIL) \quad (9.18)$$

Pour nos expérimentations, nous avons choisi la distribution suivante de documents atomiques pertinents (n_i, nb_i) : (70,0), (2,1), (2,4), (2,6), (2,8), (2,10). Cette distribution produit 2 080 documents structurés composés de 20 800 documents atomiques (dont 1 508 pertinents), et qui ont une pertinence globale moyenne de $pert^{glob}_{moy}(OFIL^{req}_{agreg}) = \frac{1\ 508}{20\ 800} = 7,25\%$ (valeur théorique). Le tableau suivant présente les caractéristiques de la collection $OFIL^{req}_{agreg}$:

Documents structurés, collections $OFIL^{req}_{agreg}$ et $OFIL^{sim}_{agreg}$		
Nombre de documents structurés (nb_{ds})	2 080	2 080
Dont pertinents pour au moins une requête	996	562
Nombre de da par ds ($taille_{ds}$)	10	10
Nombre de documents atomiques	20 800	20 800
Nombre de mots	11,3 millions	11,36 millions
Nombre de termes distincts	67 900	70 400
Taille du corps textuel	40,4 Mo Mo	41,2 Mo
Taille de l'index (nnn)	19,01 Mo	19,12 Mo
Taille moyenne des documents	5 430 mots, 19,9 Ko	5 461 mots, 20,3 Ko
Nombre de mots de l'index	6,33 millions	6,38 millions
Pertinence globale moyenne ($pert^{glob}_{moy}$)	12,27 %	5,74 %
Pertinence moyenne des ds pertinents	25,6	21,2
Similarité moyenne ($sim2a2_{moy}$)	2,6	13,41

FIG. 9.5 – Caractéristiques des collections $OFIL^{req}_{agreg}$ et $OFIL^{sim}_{agreg}$.

On remarque qu'un grand nombre de documents structurés sont pertinents ou partiellement pertinents. La cause est l'absence de contrôle sur les documents atomiques choisis comme étant non pertinents. En effet, quand on crée un document non pertinent pour une requête, on ne contrôle pas si les documents atomiques ajoutés sont pertinents pour d'autres requêtes. En conséquence, la pertinence moyenne de la collection $OFIL^{req}_{agreg}$ est supérieure à celle de la collection d'origine (12,37% contre 5,33%). Par ailleurs, on remarque que les documents structurés pertinents sont en moyenne peu pertinents : seulement 25% de leurs documents atomiques sont pertinents.

b) Construction basée sur la similarité : $OFIL^{sim}_{agreg}$

La collection $OFIL^{req}_{agreg}$ n'offre aucune garantie quand à la cohérence sémantique des documents structurés, mis à part que certains contiennent des documents atomiques pertinents pour la même requête. Avec la collection $OFIL^{sim}_{agreg}$, l'objectif est de renforcer la cohérence sémantique des composants. Pour cela, les documents structurés sont construits comme des clusters. On choisi n_{ds} documents atomiques distincts qui servent de "référence" pour chaque cluster. Puis, on ajoute à chaque "cluster" les $(taille_{ds} - 1)$ documents qui sont les plus similaires au document de référence. Comme pour calculer la similarité moyenne sim_{2a2}, la fonction de similarité utilisée est le cosinus entre les vecteurs.

Pour construire une collection de taille comparable à la précédente, nous avons choisis de générer 2 080 documents suivant la stratégie de clustering. Le tableau 9.5 présente les caractéristiques de la collection $OFIL^{sim}_{agreg}$. On remarque que la pertinence globale moyenne des documents structurés est comparable avec celle de la collection initiale :

$$pert^{glob}_{moy}(OFIL^{sim}_{agreg}) = 5,74 \sim pert^{glob}_{moy}(OFIL) = 5,33\% \tag{9.19}$$

La moyenne des similarités deux à deux des documents atomiques d'un même document structuré est de $sim2a2_{moy}(OFIL^{sim}_{agreg}) = 13,41$ avec cette stratégie, contre seulement $sim2a2_{moy}(OFIL^{req}_{agreg}) = 2,6$ avec la précédente.

9.6.2 Construction de chemins de lecture

Selon l'utilisation de la collection de documents structurés $OFIL^{req}_{agreg}$ ou $OFIL^{sim}_{agreg}$, nous fabriquons la collection de chemins de lecture correspondante. Cependant, la stratégie de fabrication reste la même. Elle consiste à construire, pour chaque document structuré, nb_{chds} chemins composés chacun de $taille_{ch}$ documents atomiques (avec $taille_{ch} \leq taille_{ds}$). Les $taille_{ch}$ atomes sont choisis aléatoirement parmi les $taille_{ds}$ composants du document structuré. Ainsi, la pertinence moyenne des chemins sera comparable à celle des documents structurés.

Au cours des expérimentations présentées, nous avons simplifié les collections de chemins de lecture, en construisant seulement un chemin de taille $taille_{ch} = 10$ par document structuré. Nous n'avons donc pas expérimenté méthodiquement l'indexation de plusieurs chemins de lecture pour parcourir le même document structuré. Ce choix facilite la comparaison entre les méthodes d'indexation de documents structurés et de chemins de lecture : en effet, chaque chemin parcourt la totalité du sous-ensemble des documents atomique du document correspondant.

La difficulté dans la construction des chemins consiste à choisir l'ordre de lecture des documents atomiques. L'objectif est de simuler un "vrai" chemin de lecture, ce qui nécessite une cohérence et une progression dans l'enchaînement des nœuds. Il est difficile de mesurer cette cohérence, et il est *a fortiori* encore plus difficile d'assurer une cohérence aux chemins construits. Nous avons expérimenté différentes stratégies :

Hasard : la première stratégie est destiné à produire des chemins "témoins" dont l'ordonnancement est choisi aléatoirement (collections $OFIL_{agreg}^{req,hasard}$ et $OFIL_{agreg}^{sim,hasard}$).

Plus court chemin (pcc) : cette stratégie consiste à minimiser la somme des distances entre les documents atomiques successifs du chemin, c'est-à-dire la longueur du chemin. On fait ainsi l'hypothèse qu'un chemin de lecture, dans sa progression, relie les nœuds les plus proches avec le minimum de rupture sémantique (collections $OFIL_{agreg}^{req,pcc}$ et $OFIL_{agreg}^{sim,pcc}$). L'algorithme utilisé est un des plus simples visant à résoudre le problème du "voyageur de commerce". Il met en œuvre la "méthode des plus proches voisins[12]" qui consiste à choisir en premier lieu la paire de documents ayant la plus grande similarité, et à itérer en ajoutant le document le plus similaire au dernier choisi.

Plus long chemin (plc) : il s'agit de la stratégie inverse de la précédente, qui maximise la longueur (collections $OFIL_{agreg}^{req,plc}$ et $OFIL_{agreg}^{sim,plc}$).

Pertinents au début : cette stratégie "tasse" tous les documents atomiques pertinents au début du chemin. On fait ainsi l'hypothèse que les chemins de lecture pertinents commencent par aborder le sujet demandé, puis continuent en développant l'information d'un point de vue différent (collections $OFIL_{agreg}^{req,pertdeb}$ et $OFIL_{agreg}^{sim,pertdeb}$).

Pertinents à la fin : cette stratégie est l'inverse de la précédente. On fait l'hypothèse que l'information pertinente d'un chemin se trouve à la fin du chemin, et que le début du chemin est une progression thématique menant au sujet recherché (collections $OFIL_{agreg}^{req,pertfin}$ et $OFIL_{agreg}^{sim,pertfin}$).

Les collections de chemins $OFIL_{agreg}^{req}$ et $OFIL_{agreg}^{sim}$ sont particulières en raison du choix de construction d'un unique chemin par document. En conséquence, les caractéristiques générales des chemins de lecture des deux collections sont les mêmes que celles des documents structurés correspondants, présentées dans le tableau 9.5. Le tableau 9.6 présente la similarité moyenne des documents atomiques successifs et la longueur moyenne des chemins des sous-collections dérivées de $OFIL_{agreg}^{req}$ et de $OFIL_{agreg}^{sim}$.

[12]Cet algorithme ne trouve pas toujours la solution optimale, mais marche raisonnablement bien sur un petit nombre de nœuds.

Chemins de lecture dérivés de $OFIL^{req}_{agreg}$			
	$sim2a2_{moy}$	$sim2a2ch_{moy}$	$longueur_{moy}$
$OFIL^{req,hasard}_{agreg}$	2,6	2,6	5 193
$OFIL^{req,pcc}_{agreg}$	2,6	4,5	915
$OFIL^{req,plc}_{agreg}$	2,6	1,9	22 148
$OFIL^{req,pertdeb}_{agreg}$	2,6	2,8	5 200
$OFIL^{req,pertfin}_{agreg}$	2,6	2,8	5 200
Chemins de lecture dérivés de $OFIL^{sim}_{agreg}$			
	$sim2a2_{moy}$	$sim2a2ch_{moy}$	$longueur_{moy}$
$OFIL^{sim,hasard}_{agreg}$	13,41	13, 45	1 065
$OFIL^{sim,pcc}_{agreg}$	13,41	19,41	172
$OFIL^{sim,plc}_{agreg}$	13,41	10,41	3 440
$OFIL^{sim,pertdeb}_{agreg}$	13,41	13,7	1 012
$OFIL^{sim,pertfin}_{agreg}$	13,41	13,7	1 012

FIG. 9.6 – Caractéristiques des chemins dérivés de $OFIL^{req}_{agreg}$ et $OFIL^{sim}_{agreg}$.

On observe des valeurs élevées pour la longueur des chemins, malgré des valeurs de similarité assez importantes pour la collection $OFIL^{sim}_{agreg}$. Cela s'explique par une carence en documents atomiques : comme nous limitons au maximum le nombre d'utilisations d'un document atomique pour construire plusieurs documents structurés, le processus de fabrication n'a donc pas toujours assez de documents similaires pour compléter un document structuré. Quand il arrive qu'une similarité soit nulle, la constante K est utilisée comme *distance maximale* et augmente fortement la longueur. La différence est donc importante entre la mesure de similarité et la longueur : en effet, un document hétérogène mais comportant une partie de documents atomiques très similaires, pourra être affecté d'une longueur élevée en raison de quelques uns de ses composants ayant une similarité de zéro avec le reste.

Dans la collection $OFIL^{req}_{agreg}$, la longueur est plus courte avec la stratégie *pcc* qu'avec les autres stratégies : en moyenne, 915 contre environ 5 200 pour les stratégies *hasard, pertdeb* et *pertfin*. Nous faisons le même constat avec la collection $OFIL^{sim}_{agreg}$. Ces chiffres sont cohérents avec l'objectif de création des collections concernant la longueur des chemins. Cette cohérence est aussi vérifiée dans les deux collections avec la stratégie *plc*. Par ailleurs, on constate que les chemins sont toujours légèrement plus courts quand les documents atomiques pertinents sont placés à une extrémité du chemin, par rapport à la stratégie entièrement aléatoire, en raison de la similarité élevée des documents pertinents entre eux.

Enfin, l'observation de la mesure $sim2a2ch_{moy}$ aboutit au même constat : par exemple, pour la stratégie *hasard*, la similarité moyenne des chemins est équivalente à la similarité moyenne. Par contre, elle est plus faible pour la collection *plc* et légèrement supérieure quand les documents pertinents sont tassés en début ou en fin de chemin.

9.7 Évaluation d'un SRI structurée : collection $OFIL^{req}_{agreg}$

Nous présentons brièvement dans la première section les résultats de l'optimisation des paramètres pour l'indexation et la recherche des documents atomiques, puis nous détaillons les résultats obtenus avec les différentes stratégies d'indexation de documents structurés et de chemins sur les collections que nous venons de décrire.

9.7.1 Évaluation de l'indexation de documents atomiques

Un grand nombre d'évaluation ont été menées, en faisant varier tous les paramètres de base de l'indexation et de l'interrogation atomiques. Il en ressort que les meilleurs réglages sont les suivants :

Lemmatisation : utilisation d'un lemmatiseur basé sur le dictionnaire ABU[13] pour extraire la racine des mots.

Utilisation d'un anti-dictionnaire, élimination des accents et de la casse.

Titre : l'importance du titre des documents a été multipliée par un facteur 5.

Champs de la requêtes : tous les champs de la requête (cf. annexe B pour voir les différents champs) sont utilisés, avec un facteur multiplicateur pour chacun.

Pondération des documents : le meilleur schéma de pondération est *lfc* (cf. annexe A pour les détails de la fonction *lfc*).

Pondération des requêtes : le meilleur schéma de pondération des requêtes est *ltm*.

L'optimisation de ces paramètres permet d'atteindre une précision moyenne $AvgPrec_{11} = 44,23\%$, et produit la courbe de rappel/précision de référence présentée en annexe C.1.

9.7.2 Évaluation de l'indexation de documents structurés

Dans l'objectif de valider l'approche de calcul de la discriminance des termes au niveau des documents structurés, nous présentons des résultats de la méthode "témoin". Cette méthode consiste à indexer un document structuré par la moyenne des vecteurs de ses composants. Puis, nous présentons les résultats de notre approche d'indexation, qui consiste à recalculer le df, la discriminance des termes dans le corpus des documents structurés.

a) Indexation : moyenne des indexations atomiques

Le tableau suivant récapitule les pondérations qui donnent les meilleurs résultats, parmi les 150 combinaisons évaluées du schéma de pondération. On constate que la pondération **mtc** donne de meilleurs résultats. La figure 9.7 présente la courbe de rappel/précision de la pondération 'mtc' correspondante.

[13] ABU : Association des Bibliophiles Universels, http ://abu.cnam.fr

Pondération	$AvgPrec_{11}$
mtc	56,08 %
ltc	56,06 %
mfc	56,02 %
lfc	55,72 %

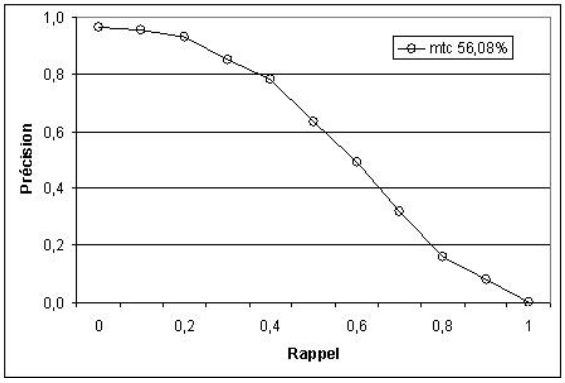

FIG. 9.7 – Indexation documents structurés : moyenne ($OFIL^{req}_{agreg}$).

Ce sont les premiers résultats sur une indexation de documents structurés, que nous pouvons difficilement comparer aux résultats d'une indexation de documents atomiques. En effet, le fait que la construction des documents structurés totalement ou partiellement pertinents soit basée sur les requêtes a deux conséquences. Un effet positif est que la tâche du SRI est facilitée : s'il retrouve un document structuré contenant un document atomique pertinent, alors il sera le plus souvent accompagné d'autres documents atomiques pertinents. D'un autre côté, un effet négatif est que tous les documents structurés partiellement pertinents retrouvés vont introduire leur lot de documents atomiques non pertinents dans les résultats. Les résultats de l'indexation *moyenne* donnent une courbe de référence pour l'évaluation de nos algorithmes d'indexation de documents structurés ou de chemins.

b) Indexation : calcul du df au niveau des documents structurés

Nous avons expérimenté deux variantes de notre algorithme : la première (dfs_{ds}) recalcule la pondération des termes dans les documents structurés en les considérant comme des documents atomiques, comme décrit dans le chapitre 8, et la deuxième (dfs_{da}) calcule aussi les pondérations au niveau des documents structurés, mais en utilisant les dfs provenant de l'indexation des documents atomiques.

Les variantes dfs_{ds} et dfs_{da} ont été évaluées à l'aide d'un total de 300 indexations (schéma de pondération des documents structurés). Le tableau 9.8 récapitule les pondérations qui donnent les meilleurs résultats.

dfsda		dfsds	
Pondération	$AvgPrec_{11}$	Pondération	$AvgPrec_{11}$
mpc	54,25%	mtc	53,42%
mtc	54,23%	mfc	53,41%
mfc	53,41%	lfc	51,73%

FIG. 9.8 – Indexation documents structurés : pondération dfs_{ds} et dfs_{da} ($OFIL^{req}_{agreg}$).

La figure 9.9 présente les courbes de rappel/précision correspondantes. On trouve en annexe C.2 toutes les courbes obtenues avec la variante dfs_{ds}.

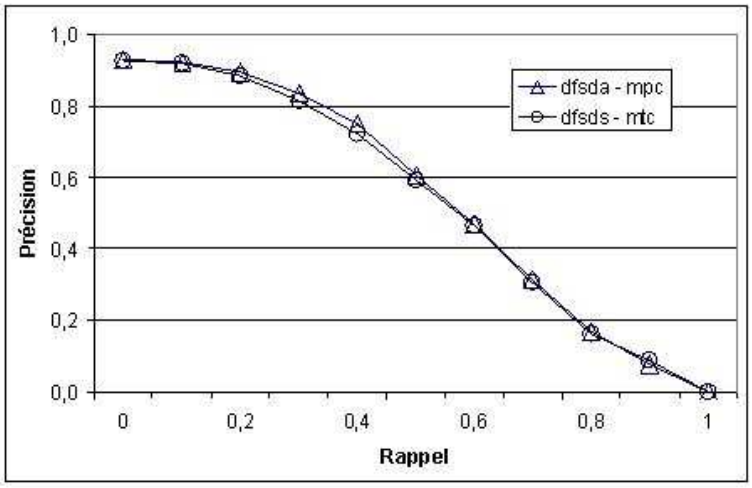

FIG. 9.9 – Indexation documents structurés : pondérations dfs_{da} et dfs_{ds} ($OFIL^{req}_{agreg}$).

Les résultats montrent une légère dégradation de la précision par rapport à la pondération “moyenne” quand le calcul du df au niveau des documents structurés est utilisé, que ce soit avec la variante dfs_{da} (54,25% contre 56,08%) ou avec la variante dfs_{ds} (53,42% contre 56,08%). Qui plus est, plus nous utilisons de l’information au niveau des documents structurés (variante dfs_{ds}), moins bons sont les résultats.

Nous pensons que cette baisse de précision provient du fait que les documents sont agrégés aléatoirement pour la plupart (à part, dans une certaine mesure, ceux contenant des documents pertinents). Cela nous amène à remettre le df en question dans le cas de documents peu cohérents. De plus, le nombre de documents structurés dans $OFIL^{req}_{agreg}$ est plus faible que celui des documents atomiques dans $OFIL$: l’échantillon est statistiquement trop petit pour qu’il soit possible d’obtenir une aussi bonne précision dans le calcul de la discriminance.

9.7.3 Évaluation de l’indexation de chemins de lecture

Les paramètres optimaux des expérimentations précédentes sont fixés afin de pouvoir expérimenter l’influence de l’indexation de chemins de lecture sur la précision des résultats. Ainsi, l’algorithme s’appuie sur l’indexation atomique utilisant une pondération *mtc* qui donne les meilleurs résultats quand on fait la moyenne des documents atomiques.

a) Paramètres α, γ et $coeff_\beta$

Nos expérimentations de l’indexation des chemins de lecture ont consisté à faire varier les paramètres (α, γ et $coeff_\beta$) de l’algorithme d’indexation et à observer leur impact sur les

résultats obtenus avec chacune des stratégies d'ordonnancement des chemins.

Pour chaque stratégie d'ordonnancement, nous avons d'abord fait varier le paramètre γ en inhibant l'effet de la mémoire de lecture (paramètre $\alpha = 0$). En particulier, nous évaluons l'indexation témoin des chemins : quand $\gamma = 1$, cela revient à l'algorithme d'indexation des documents structurés qui réalise la moyenne des vecteurs des documents atomiques. Enfin, en inhibant l'impact de γ sur l'indexation ($\gamma = 1$), nous faisons varier α et $coeff_\beta$.

b) Calcul de la similarité β

Quand $\alpha \neq 0$, les coefficients de rupture sémantique β interviennent dans l'indexation. Le calcul de la similarité (cosinus) pour obtenir le coefficient β rend des valeurs très petites $\in [0..1]$. Comme l'effet du β est élevé à la puissance 1, puis 2, puis, ..., puis $taille(ch)$ au fur et à mesure de l'indexation, il est très vite réduit à néant.

Nous avons donc utilisé une version du calcul des ruptures sémantiques qui dépend d'un coefficient $coeff_\beta$ afin d'obtenir une distribution plus uniforme des ruptures sémantiques entre 0 et 1, de la manière suivante :

$$\beta_i = cosinus(a_i, a_{i+1})^{\frac{1}{coeff_\beta}} \tag{9.20}$$

c) Évaluation de l'accumulation de lecture (γ varie)

On rappelle que le rôle du paramètre γ est de privilégier le début du chemin (quand $\gamma < 1$) ou la fin du chemin (quand $\gamma > 1$), en fixant l'importance de l'accumulateur de lecture.

La figure 9.11 montre l'évolution de la précision moyenne quand γ varie de 0 à 2. Les résultats de chacune des cinq stratégies d'ordonnancement des chemins sont présentés, et le tableau 9.10 présente le choix optimal pour chaque stratégie.

Stratégie	γ_{opt}	$AvgPrec_{11}$
Hasard	1	52,92%
Pertdeb	1,5	73,55%
Pertfin	0,6	74,21%
PCC	1	52,92%
PLC	1	52,92%

FIG. 9.10 – Choix optimaux de γ.

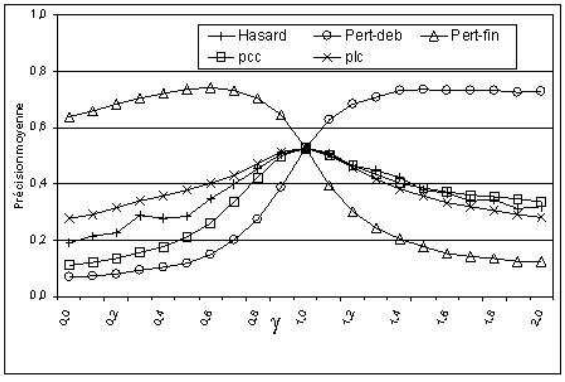

FIG. 9.11 – Gamma varie (collection $OFIL^{req}_{agreg}$).

Ces résultats nous montrent que γ joue très bien son rôle quand les documents atomiques sont placés au début ou à la fin (stratégies $pertdeb$ et $pertfin$). Les valeurs de γ pour lesquelles la précision moyenne est la meilleure sont $\gamma_{pertdeb} = 1,5$ avec la stratégie $pertdeb$ et $\gamma_{pertfin} = 0,6$ avec la stratégie $pertfin$. Conformément à nos attentes, nous trouvons $\gamma_{pertdeb} > 1$ et $\gamma_{pertfin} < 1$.

Par contre, les trois autres stratégies donnent la meilleure précision pour $\gamma = 1$, c'est-à-dire quand γ est inactivé. Nous en déduisons qu'aucune de ces stratégies ne tend à positionner les documents pertinents au début ou à la fin des chemins. Il est intéressant de constater que la stratégie *pcc* est celle qui souffre le plus de la prise en compte de l'accumulateur de lecture quand $\gamma < 1$ (mis à part la stratégie *pertdeb*). Cela s'explique par l'algorithme de calcul du plus court chemin que nous avons employé, qui commence par choisir les documents les plus similaires. Comme les documents pertinents pour une requête sont similaires entre eux, cette méthode favorise donc le positionnement des documents pertinents en tête.

d) Évaluation de la mémoire de lecture (α et $coeff_\beta$ varient)

Le paramètre γ étant inactivé ($\gamma = 1$), nous avons ensuite observé l'effet des paramètres α et $coeff_\beta$. La figure 9.12 montre l'évolution de la précision moyenne quand α varie entre 0 et 1 tandis que $coeff_\beta$ reste à 1. La figure 9.12 représente les mêmes résultats, mais avec le paramètre $coeff_\beta = 2$.

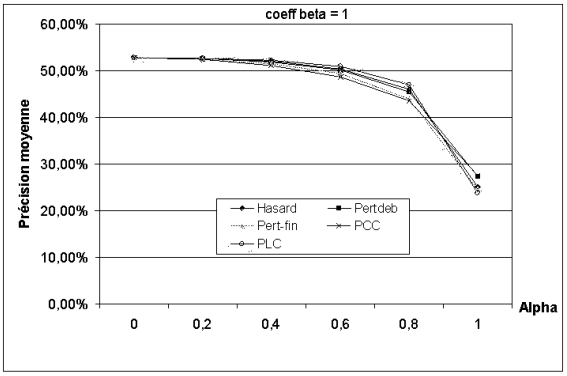

FIG. 9.12 – α varie et $coeff_\beta$ = 1.

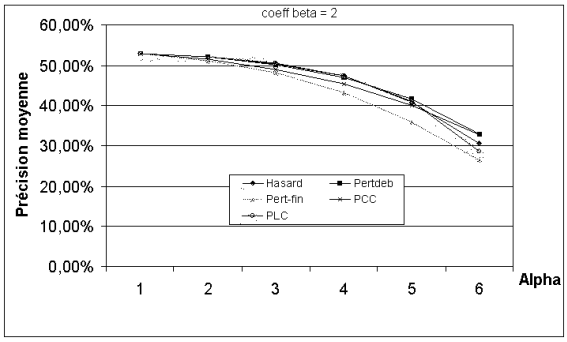

FIG. 9.13 – α varie et $coeff_\beta$ = 2.

Dans toutes nos autres expérimentations avec des combinaisons les plus variées de α, γ et $coeff_\beta$ (non présentées ici), nous arrivons au même constat : la précision est strictement décroissante avec l'augmentation de α. Dans aucun cas de figure, la prise en compte de la *mémoire de lecture* de notre algorithme ne permet d'améliorer les résultats.

Etant donné que les chemins sont construits automatiquement, nous ne pouvons pas déterminer si ces résultats sont causés par l'inadaptation de notre algorithme à la recherche de chemins de lecture, ou bien par la construction même de ces chemins. Nous avons donc

cherché à introduire un peu plus de cohésion sémantique dans les chemins, au cours de nos expérimentations sur la collection $OFIL^{sim}_{agreg}$.

9.8 Évaluation d'un SRI Structurée : collection $OFIL^{sim}_{agreg}$

La stratégie d'expérimentation suivie est identique à celle utilisée avec $OFIL^{req}_{agreg}$. Cette section récapitule les résultats les plus importants, avec comme objectif principal d'observer l'impact de nos méthodes sur ce corpus spécifique, alors que cet impact est négatif dans le cas de $OFIL^{req}_{agreg}$.

L'indexation atomique optimale utilisée avec la collection $OFIL^{sim}_{agreg}$ est bien sûr identique à celle utilisée avec $OFIL^{req}_{agreg}$, étant donné que l'indexation atomique est la même.

9.8.1 Évaluation de l'indexation de documents structurés

a) Indexation : moyenne des indexations atomiques

Le tableau 9.14 récapitule les pondérations qui donnent les meilleurs résultats, parmi les 150 combinaisons évaluées du schéma de pondération des documents atomiques. La figure associée présente la courbe de rappel/précision correspondant à la pondération optimale 'lfc'.

Pondération	$AvgPrec_{11}$
lfc	28,22 %
ltc	27,70 %
mfc	27,60 %

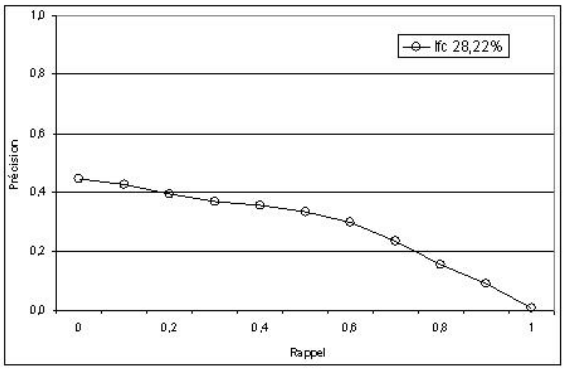

FIG. 9.14 – Indexation documents structurés : moyenne ($OFIL^{sim}_{agreg}$).

Les résultats montrent une dégradation importante de la précision par rapport à la même stratégie d'indexation sur la collection $OFIL^{req}_{agreg}$: 28,22% contre 56,08%. Cette dégradation est causée par le nombre plus faible de documents structurés pertinents dans $OFIL^{sim}_{agreg}$ que dans $OFIL^{req}_{agreg}$ (562 contre 996) alors que la pertinence moyenne des documents pertinents est comparable (21,2% contre 25,6%).

b) Indexation : calcul du df au niveau des documents structurés

Nous avons expérimenté les deux variantes dfs_{ds} et dfs_{da}. Le tableau 9.15 récapitule les pondérations qui donnent les meilleurs résultats et la figure 9.16 présente les courbes de rappel/précision correspondantes.

dfsda		dfsds	
Pondération	$AvgPrec_{11}$	Pondération	$AvgPrec_{11}$
mfc	27,41 %	mfc	27,41 %
lpc	26,79 %	lpc	26,76 %
ltc	26,73 %	lfc	26,55 %

FIG. 9.15 – Indexation documents structurés : pondération dfs_{ds} et dfs_{da} ($OFIL^{sim}_{agreg}$).

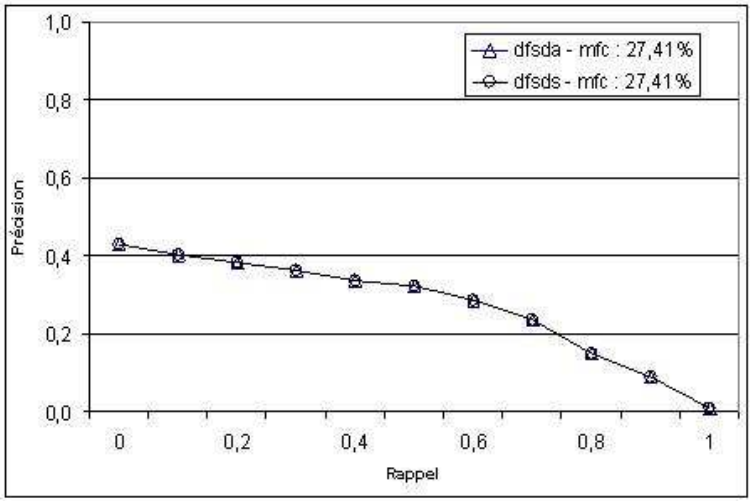

FIG. 9.16 – Indexation documents structurés : pondération dfs_{da} et dfs_{ds} ($OFIL^{sim}_{agreg}$).

On constate que pour les deux stratégies de calcul du df, les meilleurs résultats sont obtenus avec le schéma de pondération mfc. Ce schéma ne fait pas intervenir le df des termes : il est donc logique d'obtenir exactement les mêmes résultats dans ce cas là.

Comme pour la collection précédente, on observe une légère dégradation des résultats par rapport à l'indexation "moyenne". La stratégie de grouper des documents atomiques en documents structurés selon leur similarité ne permet donc pas de mettre en avant une amélioration des résultats avec notre algorithme d'indexation des documents structurés. Cela nous amène à la même conclusion : la cause est soit de la stratégie de fabrication des documents structurés, soit de l'algorithme d'indexation lui-même.

9.8.2 Évaluation de l'indexation de chemins de lecture

Nous avons conduit la même série d'expérimentations que pour la collection $OFIL^{req}_{agreg}$.

a) Évaluation de l'accumulation de lecture (γ varie)

La figure 9.18 montre l'évolution de la précision moyenne quand γ varie de 0 à 2. Les résultats de chacune des cinq stratégies d'ordonnancement des chemins sont présentés, et le tableau 9.17 présente le choix optimal de γ pour chaque stratégie.

Stratégie	γ	$AvgPrec_{11}$
Hasard	1,1	53,42%
Pertdeb	2	68,8 %
Pertfin	0,4	72,05%
PCC	0,9	54,10%
PLC	1,1	54,13%

FIG. 9.17 – Choix optimaux de γ

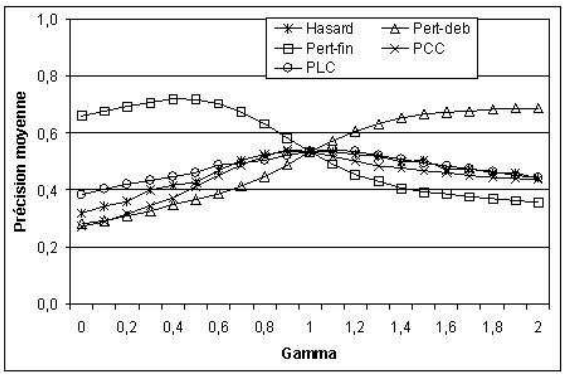

FIG. 9.18 – Gamma varie ($OFIL^{sim}_{agreg}$).

Comme avec $OFIL^{req}_{agreg}$, ces résultats nous montrent que γ joue très bien son rôle quand les documents atomiques sont placés au début ou à la fin (stratégies $pertdeb$ et $pertfin$).

Un résultat intéressant de cette série d'expérimentations est que la valeur optimale de γ est différente de 1 pour chacune des stratégies. Cela signifie que l'accumulation de lecture améliore l'indexation des chemins. Cependant, exceptées les stratégies *pertdeb* et *pertfin* qui font apparaître, avec cette collection aussi, une amélioration très importante de la précision, les différentes stratégies montrent une amélioration peu significative (+1% pour la stratégie *plc*). Nous ne pouvons pas tirer de conclusion définitive de ces résultats, mais ils sont encourageants et laissent penser que cette stratégie de construction de collection est meilleure que la précédente.

b) Évaluation de la mémoire de lecture (α et $coeff_\beta$ varient)

Nous procédons comme pour la collection précédente. Le paramètre γ étant inactivé ($\gamma = 1$), nous avons ensuite observé l'effet des paramètres α et $coeff_\beta$. La figure 9.19 montre l'évolution de la précision moyenne quand α varie entre 0 et 1 tandis que $coeff_\beta$ reste à 1. La figure 9.20 représente les mêmes résultats, mais avec le paramètre $coeff_\beta = 2$.

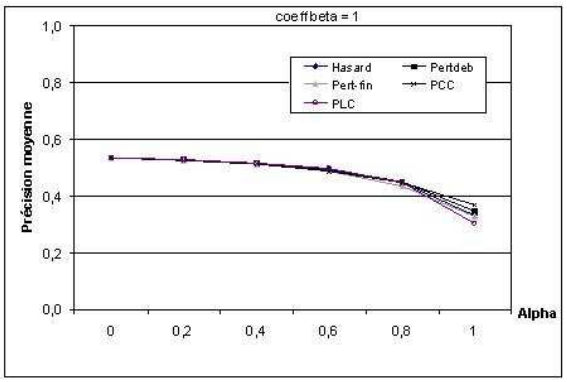

FIG. 9.19 – α varie et $coeff_\beta = 1$.

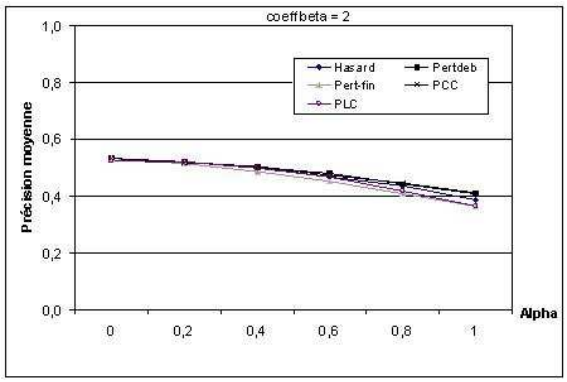

FIG. 9.20 – α varie et $coeff_\beta = 2$.

Ces résultats montrent que la collection $OFIL^{sim}_{agreg}$ ne permet pas non plus de mettre en avant une amélioration des résultats quand la mémoire de lecture est utilisée.

9.9 Construction par fragmentation : $OFIL_{frag}$

Avec les collections $OFIL_{agreg}$ et la mesure de similarité, il est possible de construire des documents structurés thématiquement cohérents. Par contre, il est difficile d'assurer une continuité sémantique entre les nœuds qui se suivent, et il est encore plus difficile de traiter le problème d'ordonnancement des informations étant donné que la mesure de similarité est symétrique.

Pour introduire dans nos collections cette notion d'ordre entre les documents atomiques, nous proposons donc d'utiliser de réels enchaînements : ceux que fait l'auteur quand il écrit un paragraphe, puis un autre, etc. Pour cela, notre stratégie de construction de collection consiste à fragmenter les documents existants en plusieurs "documents" qui seront considérés, dans la collection créée, comme des documents atomiques. Ainsi, ce sont les documents de la collection OFIL de départ qui seront considérés comme des documents structurés.

La méthode consiste à découper les documents en fragments d'une taille supérieure à un seuil en nombre de caractères (fixé ici à 300), en prenant garde à ne pas couper de phrase. Le titre du document initial constitue un document atomique à lui tout seul. Ainsi, mis à part les titres (et éventuellement les fins de documents initiaux), tous les documents atomiques de la collection auront une taille supérieure ou égale à 300 caractères. Par contre, les documents structurés auront une taille différente en nombre de documents atomiques. Le lecteur trouvera un exemple de document fragmenté en annexe B.4.

9.9.1 Construction de documents structurés

Le tableau suivant présente les caractéristiques de la collection $OFIL_{frag}$:

Documents structurés, collection $OFIL_{frag}$	
Nombre de documents structurés (nb_{ds})	11 016
Dont pertinents pour au moins une requête (pertinence non binaire)	576
Nombre de documents atomiques (da)	86 751
Nombre de da par ds	7,87
Taille moyenne des da	72,3 mots, 0,36 Ko
Pertinence globale moyenne ($pert^{glob}_{moy}$)	5,33%
Pertinence moyenne des ds pertinents	100%
Similarité moyenne ($sim2a2_{moy}$)	10,33

FIG. 9.21 – Fragments de la collection $OFIL_{frag}$.

La construction des documents structurés réalise l'opération inverse de la fragmentation, en recomposant le document initial sous la forme d'un document structuré. Reconstruire les documents d'OFIL n'est pas d'une grande utilité en soi, mais cela nous permet d'exploiter leur structure, que nous supposons cohérente étant donné qu'elle a été établie par l'auteur.

La stratégie de propagation de pertinence consiste à affecter à chaque document atomique une pertinence proportionnelle à sa taille par rapport à celle du document englobant. Par exemple, un document atomique de 320 caractères qui est composant d'un document structuré de 6 400 caractères se verra affecter la valeur de pertinence de 0,05. Inversement, la pertinence d'un document structuré est la somme de la pertinence de ses composants : on retrouve la pertinence binaire des documents structurés.

9.9.2 Construction de chemins de lecture

Il est possible de construire des chemins de lecture de la même manière que pour la collection $OFIL_{agreg}$. Toutefois, si nous voulons profiter de la cohérence du chemin de lecture implicite constitué par le sens de lecture du document initial, il nous faut expérimenter cette nouvelle stratégie d'ordonnancement qui consiste à réutiliser le chemin de lecture original prévu par l'auteur.

La création des chemins consiste donc à retrouver pour chaque document structuré, l'ordre dans lequel étaient présentés les fragments dans la collection initiale. Ce chemin correspond au *chemin de lecture standard* du document. La nouvelle stratégie d'ordonnancement correspondante est appelée "*initial*". A des fins de comparaison, nous avons également construit, pour chaque *chemin de lecture standard*, une version du chemin suivant les stratégies proposées dans la section 9.6.2 : *hasard*, *pcc*, *plc*, *pert-deb* et *pert-fin*.

Les caractéristiques générales de la collection de chemins $OFIL_{frag}$ sont les mêmes que celles des documents structurés correspondants (cf. tableau 9.21), en raison du choix de construction d'un unique chemin par document. Le tableau suivant présente les caractéristiques des chemins de chaque collection dérivée de $OFIL_{frag}$ en rappelant la similarité moyenne deux à deux à des fins de comparaison avec la similarité chemin :

Chemins de lecture dérivés de $OFIL_{frag}$			
	$sim2a2_{moy}$	$sim2a2ch_{moy}$	$longueur_{moy}$
$OFIL_{frag}^{initial}$	10,33	14,14	9,8
$OFIL_{frag}^{hasard}$	10,33	10,32	10,6
$OFIL_{frag}^{pcc}$	10,33	16,65	4,6
$OFIL_{frag}^{plc}$	10,33	8,88	19,9
$OFIL_{frag}^{pertdeb}$	10,33	10,38	10,4
$OFIL_{frag}^{pertfin}$	10,33	10,38	10,4

FIG. 9.22 – Caractéristiques des chemins dérivés de $OFIL_{frag}$.

La similarité deux à deux des documents atomiques d'un même document est élevée (10,33), mais n'atteint pas la valeur obtenue avec la collection $OFIL_{agreg}^{sim}$ (13,41). Il est difficile de comparer la longueur des chemins dans cette collection avec la longueur dans les collections $OFIL_{agreg}^{req}$ et $OFIL_{agreg}^{sim}$, étant donné que $OFIL_{frag}$ ne connaît pas le problème de carence rencontré lors de la construction de ces deux collections.

Cependant, nous pouvons quand même utiliser ces mesures pour comparer les collections dérivées de $OFIL_{frag}$ entre elles. On retrouve alors les mêmes rapports que pour les collections précédentes : par rapport à la stratégie *hasard*, les chemins sont plus long avec *plc* et plus courts avec *pcc* (en terme de $longueur_{moy}$). La stratégie *initial* produit des chemins plus courts que *hasard*. Cela indique que les documents atomiques ne sont pas organisés au hasard par l'auteur : il y a une cohésion sémantique entre les nœuds successifs.

9.10 Évaluation d'un SRI Structurée : collection $OFIL_{frag}$

9.10.1 Évaluation de l'indexation de documents atomiques

L'évaluation des documents atomiques montre les paramètres optimaux suivants :

Lemmatisation : utilisation d'un lemmatiseur basé une troncature simple des suffixes courants du français.

Utilisation d'un anti-dictionnaire, élimination des accents et de la casse.

Titre : sans objet (les fragments n'ont pas de titre).

Champs de la requêtes : tous les champs de la requête sont utilisés.

Pondération des documents : le meilleur schéma de pondération est *lnc*.

Pondération des requêtes : le meilleur schéma de pondération est *lsm*.

L'optimisation de ces paramètres permet d'atteindre une précision $AvgPrec_{11} = 17,86\%$, et donne la courbe de rappel/précision de référence présentée dans la figure suivante :

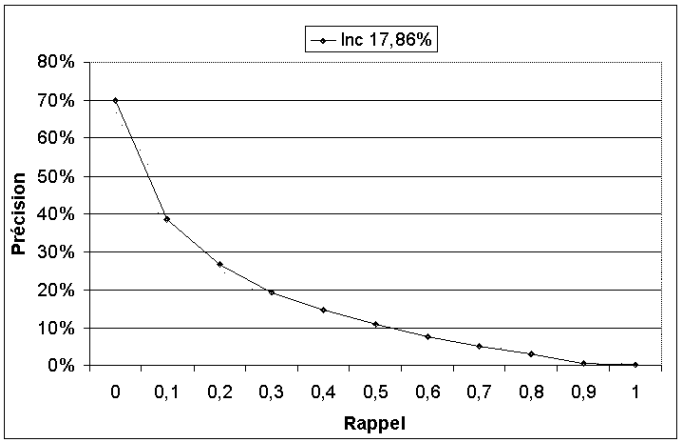

FIG. 9.23 – Indexation atomique (collection $OFIL_{frag}$).

Les résultats donnent une précision faible, qui montre la difficulté de retrouver des fragments de documents de petite taille (environ 300 caractères) indépendamment de leur contexte.

9.10.2 Évaluation de l'indexation de documents structurés

Nous conservons le même schéma d'évaluation : indexation par moyenne, puis en calculant le df au niveau des documents structurés, et ensuite évaluation des chemins de lecture.

a) Indexation : moyenne des indexations atomiques

Le meilleur schéma de pondération pour l'indexation par moyenne est le schéma *lnc*, comme montré par la courbe de rappel/précision de la figure suivante :

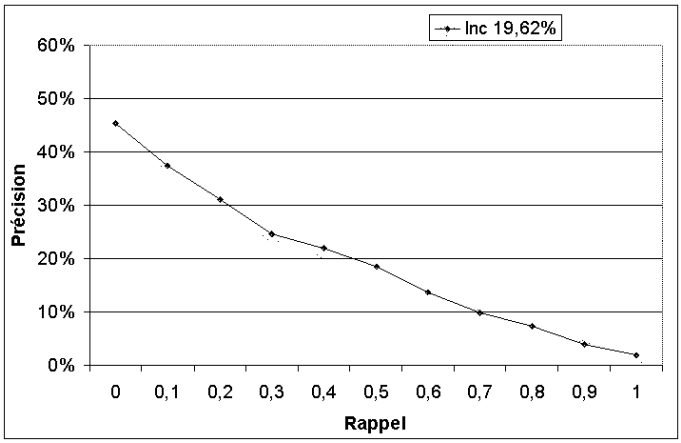

FIG. 9.24 – Indexation documents structurés : moyenne lnc (collection $OFIL_{frag}$).

Il est intéressant de comparer ces résultats avec l'indexation atomique. On constate que l'indexation de documents structurés améliore légèrement les résultats. Tout en étant basée sur l'indexation atomique, cette indexation permet une recherche au niveau du document structuré, ce qui avantage légèrement la méthode. Les résultats de cette indexation *moyenne* donnent une courbe de référence pour la suite des expérimentations.

b) Indexation : calcul du df au niveau des documents structurés

Le tableau 9.25 récapitule les trois pondérations qui donnent les meilleurs résultats avec les variantes de l'indexation dfs_{ds} et dfs_{da}, et la figure 9.26 présente les courbes de rappel/précision correspondantes.

dfsda		dfsds	
Pondération	$AvgPrec_{11}$	Pondération	$AvgPrec_{11}$
lnc	38,19%	lnc	38,19%
lfc	38,19%	lfc	38,19%
lpc	37,37%	ltc	36,78%

FIG. 9.25 – Indexation documents structurés : pondérations dfs_{ds} et dfs_{da} ($OFIL_{frag}$).

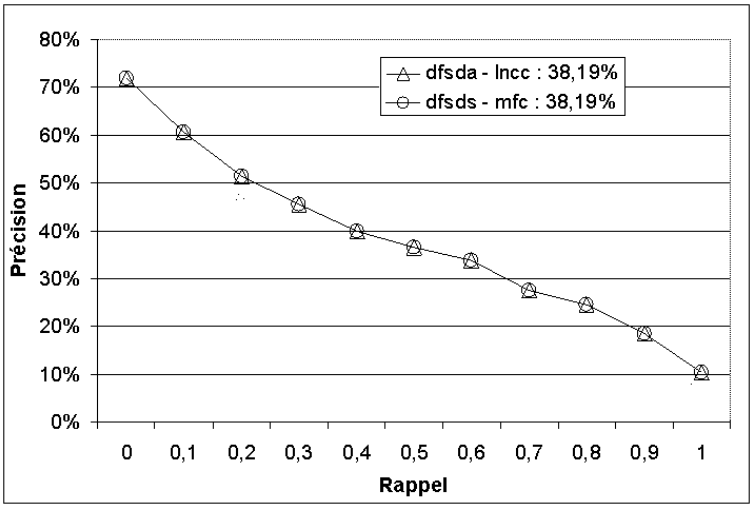

FIG. 9.26 – Indexation documents structurés : pondérations dfs_{da} et dfs_{ds} ($OFIL_{frag}$.)

Les deux meilleures pondérations n'exploitent pas le df, ce qui explique que les résultats soient identiques. Ces résultats montrent une amélioration très importante de la précision quand l'indexation se déroule au niveau du document structuré, par rapport à l'indexation *moyenne* qui pâtit de la petite taille des documents atomiques et du fait qu'ils soient indexés

indépendamment de leur contexte. Ce résultat est important et montre l'intérêt de l'indexation au niveau des documents structurés quand la fragmentation en documents atomiques est trop fine.

9.10.3 Évaluation de l'indexation de chemins de lecture

a) Évaluation de l'accumulation de lecture (γ varie)

La figure 9.28 montre l'évolution de la précision moyenne quand γ varie de 0 à $1,3$. Les résultats de chacune des six stratégies d'ordonnancement des chemins sont présentés, et le tableau 9.27 montre le choix optimal de γ pour chaque stratégie.

Stratégie	γ	$AvgPrec_{11}$
Initial	0,8	32,26%
Hasard	0,8	36,50%
Pertdeb	0,8	34,69 %
Pertfin	0,8	34,69%
PCC	0,9	34,46 %
PLC	0,8	38,34%

FIG. 9.27 – Choix optimaux de γ.

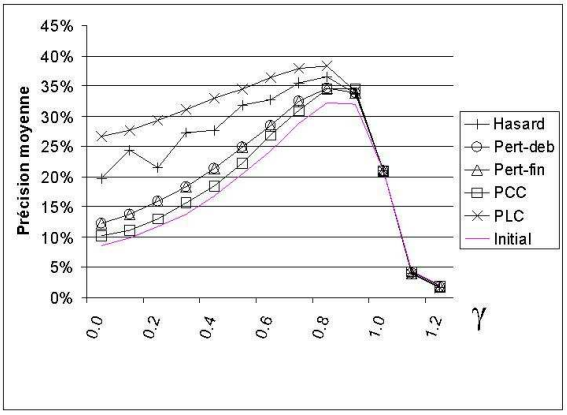

FIG. 9.28 – γ varie.

Ces résultats montrent une amélioration très importante de la précision quand γ est diminué par rapport à l'indexation "témoin" ($\gamma = 1$, correspond à l'indexation "moyenne" : 19,62%). Selon les stratégies d'ordonnancement, le gain est de 11% à 14% quand $\gamma = 0,9$. Par contre, la précision s'effondre dès que γ devient plus grand que 1, à un point tel que la précision passe sous la barre de la pertinence globale moyenne (5,33%). Cela signifie que le SRI donne de moins bons résultats que s'il choisissait aléatoirement les documents. Cela provient de l'absence de normalisation sur la taille du chemin, dans cette version de l'algorithme. En effet, contrairement aux collections précédentes, les chemins n'ont pas tous la même taille dans la collection $OFIL_{frag}$. Les plus longs d'entre eux profitent outrageusement de l'effet cumulatif de l'algorithme d'indexation.

Le résultat intéressant de cette expérimentation est que l'utilisation de l'accumulation de lecture améliore la précision moyenne quelle que soit la stratégie d'ordonnancement choisie (+11% à +17%). Par contre, la stratégie "hasard" est une des stratégies qui profite le plus de cette amélioration, et de surcroît la stratégie "initial" est celle qui en profite le moins. Si notre algorithme utilise l'ordre des documents pour améliorer l'indexation, alors l'effet est renforcé quand les documents successifs sont les moins similaires (stratégies *plc* et *hasard*).

b) Évaluation de la mémoire de lecture (α et $coeff_\beta$ varient)

Nous avons ensuite observé l'effet des paramètres α et $coeff_\beta$ en laissant le paramètre γ à la valeur donnant les meilleurs résultats ($\gamma = 0,8$). La figure 9.29 montre l'évolution de la précision moyenne quand α varie entre 0 et 1 tandis que $coeff_\beta$ reste à 1. La figure 9.30 représente les mêmes résultats, mais avec le paramètre $coeff_\beta = 2$.

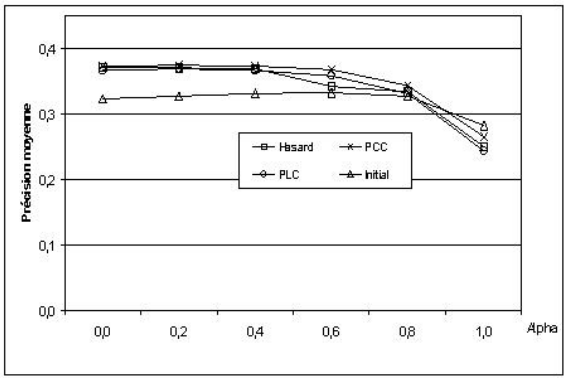

FIG. 9.29 – α varie, $\gamma = 0,8$ et $coeff_\beta = 1$.

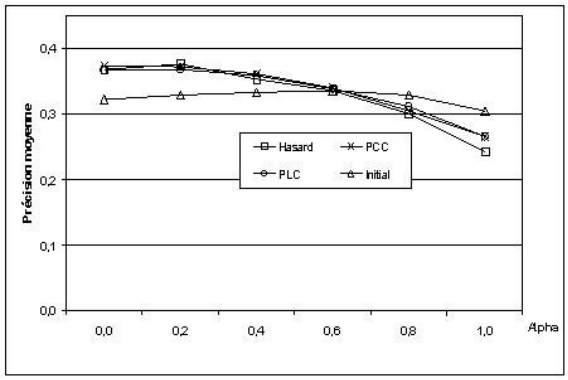

FIG. 9.30 – α varie, $\gamma = 0,8$ et $coeff_\beta = 2$.

Le tableau suivant résume les paramètres optimaux identifiés au cours d'expérimentations faisant varier plus précisément les paramètres :

Stratégie	γ	α	$coeff_\beta$	$AvgPrec_{11}$
Initial	0,85	0,15	1,5	34,51%
Hasard	0,75	0,4	1,2	39,21%
Pertdeb et pertfin	0,8	0,3	1,2	36,34%
PCC	0,85	0,25	1,3	39,47 %
PLC	0,85	0,2	1,5	38,89%

FIG. 9.31 – Choix optimaux de γ, α et $coeff_\beta$.

Ces derniers résultats sont très intéressants : ils montrent que l'utilisation de la mémoire de lecture permet d'améliorer encore les résultats par rapport à l'indexation utilisant un γ optimisé ($\gamma = 0,8$). On observe que le paramètre $coeff_\beta$ est lui aussi utilisé pour arriver à la meilleure précision moyenne obtenue, et cela quelle que soit la stratégie d'ordonnancement employée. Nous constatons malgré cela que l'ordonnancement "initial" est toujours celui qui produit la précision moyenne la moins élevée.

9.11 Conclusion

Nous avons présenté notre démarche d'évaluation d'un SRIS mettant en œuvre le modèle d'hyperdocuments en contexte proposé, dans le cadre de collections de test structurées. Ayant évoqué les limites des collections de test existantes pour évaluer un tel SRIS, nous avons expérimenté l'aspect *information accessible* de notre modèle sur une collection construite manuellement, dans le contexte du Web. Les résultats montrent que la combinaison de la facette "information accessible" avec la facette "contenu" d'un document donne de meilleurs résultats que l'utilisation du contenu seul.

Nous avons également expérimenté les aspects *documents structurés* et *hypertextes* (chemins de lecture) à l'aide de collections de test construites automatiquement. L'expérimentation du SRIS complet serait très coûteuse, car elle nécessite la construction d'une collection de test adaptée dont nous avons présenté les difficultés. Les méthodes de construction proposées nous ont permis d'expérimenter plusieurs stratégies différentes. Toutefois, la meilleure solution parmi toutes celles que nous avons expérimentées est celle de la collection $OFIL_{frag}$, car les "chemins de lecture" ont été créés manuellement par l'auteur. En utilisant cette collection, nous avons montré que plusieurs aspects de nos algorithmes d'indexation améliorent la précision moyenne des résultats. En premier lieu, nous avons observé que l'indexation au niveau des documents structurés donne de bien meilleurs résultats que l'indexation de documents atomiques sortis de leur contexte. Ensuite, l'utilisation de l'accumulation de lecture, comme celle de la mémoire de lecture, a montré une amélioration des résultats. Par contre, nous ne pouvons pas tirer de conclusion de l'expérimentation de différentes stratégies d'ordonnancement, étant donné que le chemin de lecture "*initial*" choisi par l'auteur donne de moins bons résultats que la plupart des autres stratégies.

Il faut poursuivre cette série d'expérimentations afin d'expérimenter certains aspects du problème que nous avons laissé de côté jusqu'à présent. Parmi ceux-ci, on peut citer l'utilisation du paramètre λ mais aussi la création de collections comportant une plus grande variété de chemins : différentes tailles, différentes stratégies d'ordonnancement, et enfin plusieurs chemins par document structuré.

Chapitre 10

Un SRI Structurée sur le Web

10.1 Vers un SRI Structurée sur le Web

Afin de montrer la pertinence et la faisabilité de notre approche dans sa globalité et sur le Web, nous avons mis en œuvre le modèle d'hyperdocuments au sein d'un Système de Recherche d'Information Structurée (SRIS) complet. Cela nous a permis d'identifier les principales difficultés de l'application de notre modèle dans le contexte du Web et à grande échelle. Nous présentons brièvement dans cette section les différents modules qui composent le SRIS.

Puis, nous détaillons des résultats d'analyse de collections du Web indexées par notre système. L'objectif est de déterminer si les données et la structure existante sont en adéquation avec le modèle d'hyperdocuments que nous proposons. En particulier, nous avons mené des expérimentations sur le typage des liens, qui est à la fois un des aspects les plus importants du modèle et une difficulté majeure de sa mise en œuvre.

10.2 Architecture du système

Le SRIS est composé d'un robot pour collecter les pages, d'outils d'analyse des collections, d'outils de typage des pages et des liens, d'un module d'indexation vectorielle, de modules d'extraction des cheminements et du contexte, etc.

Robot : CLIPS-Index[1] est un robot qui parcourt le Web et qui peut collecter jusqu'à 3 millions de pages par jour. Il enchaîne les opérations suivantes : choix d'une URL parmi une base locale d'URLs à collecter, collecte de la page, analyse du HTML et extraction de la liste d'URLs, stockage de la page HTML, et ajout des nouvelles URLs à la base d'URLs à collecter. CLIPS-Index a été développé avec Dr. Vaufreydaz de l'équipe GEOD du laboratoire CLIPS.

Analyseur HTML : nous avons développé un ensemble d'outils d'analyse et d'extraction de statistiques à partir des données collectées (35 000 lignes de PERL). A partir de

[1]CLIPS-Index : http ://clips-index.imag.fr

données brutes HTML, l'analyseur extrait des corpus normalisés (texte, liens, méta-données, etc.), le lexique et diverses statistiques comme le langage, la structure des pages, les types de pages et de liens, etc. L'extraction doit être robuste, malgré les données hétérogènes qui respectent rarement les standards du Web.

Typage : un module important utilisé par l'analyseur HTML est le module de typage, qui est capable de traiter jusqu'à 60 millions de liens par jour, dont nous présentons les premiers résultats dans la section 10.4.

Indexation (1 : noyau du SRIS) : nous avons développé un moteur d'indexation et d'interrogation basé sur le modèle vectoriel (VSM [Salton71]), pouvant indexer des Go de données hétérogènes.

Indexation (2 : documents structurés) : le module d'indexation des documents structurés réalise la propagation d'information le long des liens de composition.

Indexation (3 : lecture de chemins) : le module d'indexation des chemins est basé sur l'algorithme de lecture de chemins décrit dans la section 8.4.

Indexation (4 : extraction de contexte) : le module de mise en contexte réalise la propagation de popularité et d'information le long des liens de référence, comme décrit dans la section 8.6.

Interrogation : le module d'interrogation est basé sur la correspondance proposée dans la section 8.9, et propose une interface Web pour définir les requêtes et consulter les résultats.

La figure 10.1 montre l'architecture du système avec le robot, le module d'extraction, les modules de typage de liens et de pages, et les quatre modules d'indexation : documents atomiques, documents structurés, chemins de lecture et contexte. Par ailleurs, nous présentons en annexe E des copies d'écran de l'interface du robot CLIPS-Index ainsi que de l'interface du SRIS qui permet d'accéder et d'interroger l'ensemble des collections indexées.

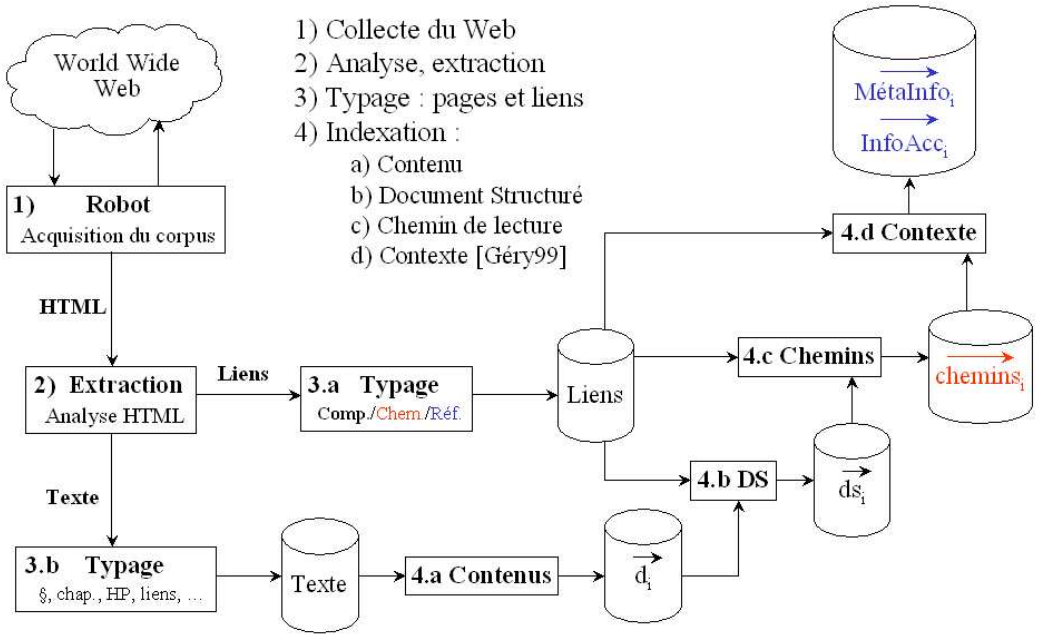

FIG. 10.1 – Architecture du SRIS.

10.3 Collecte de corpus : des échantillons du Web

10.3.1 Des corpus variés

Le robot CLIPS-Index a été utilisé ces dernières années pour collecter divers corpus à différentes dates : le Web francophone (5 millions), le Web Irlandais (0,5 million), les universités françaises (2 millions), le domaine *.museum* (0,6 million), des pages personnelles (0,2 million), des journaux et magazines français (0,4 million), etc. Ces corpus sont collectés à des fins d'expérimentations sur la structure du Web, la modélisation de langage pour la reconnaissance de la parole, l'extraction de connaissances à partir des textes, l'indexation de la structure, etc. Nous présentons dans les sections suivantes les résultats de l'analyse de cinq de ces corpus :

"IMAG" : un corpus de pages provenant des sites Web des laboratoires d'informatique de l'IMAG (le domaine "*.imag.fr*"). L'objectif est de construire un corpus relativement homogène de pages, qui sont pour la plupart des pages institutionnelles ou des documents techniques.

"Tunisie" : un corpus de pages collectées sur le domaine "*.tn*". L'objectif est de construire un corpus qui ne soit pas trop volumineux, qui contienne une majorité de documents francophones, et qui soit représentatif d'un pays.

"PagesPerso" : un corpus provenant d'hébergeurs de sites Web personnels. L'objectif est de construire un corpus permettant de comparer l'utilisation qui est faite de la structure

dans un autre cas d'utilisation du Web, avec des pages personnelles qui sont généralement moins strictes et beaucoup plus hétérogènes.

"Journaux" : un corpus de pages provenant d'un grand nombre de sites Web de journaux. L'objectif est de construire un corpus textuel de grande taille, francophone, et de bonne qualité dans l'utilisation du HTML et de la langue française.

"Irlande" : un autre exemple de collecte d'un domaine entier, mais qui est cette fois-ci anglophone.

10.3.2 Caractéristiques des collections

Les caractéristiques générales des collections sont présentées dans le tableau 10.2. Le "site" et la "page Web" sont les deux granularités couramment utilisées dans les moteurs de recherche. Le tableau 10.3 montre leurs caractéristiques pour chacune des collections.

	IMAG	Tunisie	PagesPerso	Journaux	Irlande
Nombre de sites	86	405	485	235	5 225
Nombre de pages	64 797	43 651	57 730	345 860	311 640
Taille HTML	1 544 Mo	396 Mo	461 Mo	7 728 Mo	6 752 Mo
Taille texte	779 Mo	55 Mo	126 Mo	1 491 Mo	1 741 Mo
Millions de termes	120	8,5	20,2	223	274
Termes distincts	386 000	164 000	273 00	718 000	2 millions
% pages françaises	37%	61 %	87%	93%	0,8%
Millions de liens	2,46	0,5	0,8	7,4	6,2

FIG. 10.2 – Caractéristiques générales des collections.

	IMAG	Tunisie	PagesPerso	Journaux	Irlande
Site					
Taille HTML	17,95 Mo	0,98 Mo	0,95 Mo	32,88 Mo	1,29 Mo
Taille texte	9,05 Mo	0,14 Mo	0,26 Mo	6,34 Mo	0,33 Mo
Nombre de termes	1 400 000	21 000	42 000	949 000	52 000
Nombre de liens	28 600	1 234	1 649	31 489	1 187
Profondeur moy.	6,46	3,37	2,14	2,67	3,42
Page					
Taille HTML	24,40 Ko	9,29 Ko	8,18 Ko	22,88 Ko	22,19 Ko
Taille texte	12,31 Ko	1,29 Ko	2,23 Ko	4,41 Ko	5,72 Ko
Nombre de termes	1 852	195	350	645	879
Nombre de liens	38	11,45	13,85	21,40	19,89
Profondeur moy.	3,22	2,37	2,54	3,52	2,79

FIG. 10.3 – Caractéristiques des pages Web.

On remarque des disparités importantes entre les différentes collections. La taille d'un site et d'une page varie considérablement (par exemple d'un facteur 64 entre "Tunisie" et "IMAG"), aussi bien en terme de taille HTML, que de taille textuelle ou en nombre de termes. Cela entraîne un déséquilibre du nombre de liens par site. On note aussi que la moyenne de la profondeur moyenne des sites varie. Ces différences existent également au niveau de la page HTML, même si elles sont moins marquées. Cette hétérogénéité importante, alors même que les collections contiennent entre 43 000 et 346 000 pages, est un argument en faveur d'une utilisation affinée de la granularité.

10.4 Analyse des corpus et typage automatique de liens

Notre approche nécessite la description de documents et de liens typés, et permettant au lecteur de comprendre l'organisation structurelle, l'enchaînement et la mise en contexte des idées. Or, même si la volonté des auteurs de sites Web est parfois de définir et de typer une telle structure, le langage HTML (qui est majoritairement utilisé sur le Web) ne le permet pas. Un début de typage a pourtant été proposé dans la norme, mais il est très rarement utilisé dans un Web où seulement 7% des pages respectent la norme [Beckett97]. Dans le futur, avec l'avènement de XML [Bray et al.00] et de ses dérivés, la description de pages Web se rapprochera de l'exemple "idéal" présenté dans la section 2.7.1, intégrant un typage de liens explicite. En attendant, nous sommes confrontés à la problématique du typage automatique de liens. Nous travaillons à ce typage en se basant uniquement sur le "*sac de liens*" et le "*sac de nœuds*" qu'est le Web actuel. L'algorithme utilise des heuristiques simples sur la syntaxe des URLs, en accordant de l'importance à la structure hiérarchique des répertoires du serveur Web : nous avons par exemple adopté et adapté certaines propositions de Spertus (cf. section 2.6) en analysant la configuration d'une URL par rapport à la hiérarchie de fichiers du serveur Web. Nous utilisons aussi une liste de patrons fréquents de structures de sites Web, comme par exemple le bandeau de navigation "*Page précédente*", "*Page principale*", "*Page suivante*", etc.

Nous présentons les résultats des expérimentations d'analyse et d'extraction de structure que nous avons menées sur la collection "IMAG", et nous terminons par la présentation du typage de liens des cinq collections analysées.

10.4.1 Analyse de la granularité

Nous avons expérimenté l'extraction de 6 niveaux de granularité différents, de l'ordre de grandeur de la phrase jusqu'à celle du site, résumés dans le tableau 10.4. L'extraction se base sur des éléments syntaxiques des pages HTML et des liens, comme par exemple l'utilisation de balises HTML précises pour délimiter des paragraphes.

Niveau de granularité	Objet syntaxique	#objets
Phrase	Balises HTML “de niveau 1”	663 000
Paragraphe	Balises HTML “éléments de bloc”	659 000
	Balises HTML “séparateurs de paragraphes”	1 142 000
Sections	Balises HTML “séparateurs”	130 000
Documents	Pages HTML	38 994
Sites	Nom de sites	39
Domaines	Noms de domaines	1

FIG. 10.4 – Niveaux de granularité.

Le langage HTML est largement utilisé pour définir une structure intra-page : chaque page contient en moyenne 17 objets du niveau de la phrase, 17 éléments de bloc, 29 séparateurs de paragraphe et 3,3 sections. Les statistiques sur la granularité de la page HTML nous indiquent une taille moyenne de la page HTML de 11,65 Ko qui se réduit à 3,69 Ko si on élimine les balises HTML. Mais ce ne sont que des considérations physiques. Nous devons aussi prendre en compte la connectivité du réseau de lien pour pouvoir conclure de la pertinence de considérer une page HTML comme étant un *document*. Le tableau 10.5 montre qu’il existe en moyenne 37 liens par page dans notre collection, ce qui est largement supérieur au Web dans son ensemble. Mais si nous éliminons les liens redondants, il ne reste plus que environ 14 liens par page, ce qui est proche des autres études (13,9 liens [Woodruff et al.96], 16,1 liens [Beckett97]).

10.4.2 Réseau de liens

Liens	#liens	%	Par page	Par site	Distincts	%	Par page	Par site
Intra-page	118 248	8	2,97	3 128	13 897	2,53	0,36	356
Inter-pages	1 318 490	89,38	33,81	33 807	500 472	90,96	12,83	12 832
Inter-sites	2 093	0,14	0,05	57,67	1 708	0,31	0,04	43,79
Hors-domaines	36 265	2,46	0,93	930	34 130	6,20	0,87	875
Total	1 475 096	100	37,12	39 118	550 207	100	14,11	14 108

FIG. 10.5 – Analyse des liens.

Le réseau de liens entre les pages d’un même site Web est dense, mais sans typage il est difficile de déterminer si les pages HTML représentent des sections, des documents structurés ou des hyperdocuments. Par contre, l’analyse du réseau de liens nous montre qu’il y a peu de liens inter-sites : seulement 2,6% des liens, apparaissant dans 2,4% des pages. Nous en déduisons que l’entité “site Web” a une signification sur le Web d’un point de vue sémantique.

10.4.3 Résultats : types de relations

L'analyse simple des liens montre la distribution des liens internes (à une page), des liens hiérarchiques (qui suivent la hiérarchie du serveur Web), des liens transversaux (internes à un site, mais non hiérarchiques) et enfin des liens inter-sites et hors-domaines. Cette distribution est un début de typage, elle est présentée dans le tableau suivant :

Type de lien	#liens	%
Internes	118 248	8,02
Hiérarchiques	880 421	59,69
Transversaux	438 069	29,70
Inter-sites	2 093	0,14
Hors-domaine	36 265	2,46

FIG. 10.6 – Types de liens.

Finalement, notre algorithme de typage de liens applique les patrons de structure, et identifie le type des liens. Le tableau suivant montre la répartition des différents types de liens extraits :

Type de lien	#liens	%
Liens intra-page	118 248	8,02
Composition (down)	75 573	5,12
Composition (up)	372 489	25,25
Cheminement (séquence)	432 359	29,31
Cheminement	438 069	29,70
Référence (inter-site)	2 093	0,14
Référence (hors-domaine)	36 265	2,46
Total	1'475'096	100

FIG. 10.7 – Types de liens, collection IMAG.

Nous identifions 6% de relations de composition et 59% de relations de cheminement (dont la moitié sont séquentielles et l'autre moitié déambulatoires). Enfin, nous avons analysé d'autres collections afin d'en extraire les liens typés. Le tableau suivant synthétise la répartition des types pour chacune des cinq collections :

	IMAG	Tunisie	PagesPerso	Journaux	Irlande
Liens intra-page	7,49%	0,95%	2,07%	0,21%	1,27%
Composition	25,53%	26,28%	9,37%	15,26%	16,00%
Chemins (séquence)	31,66%	42,65%	50,38%	46,47%	51,82%
Chemins	29,46%	26,18%	22,41%	24,48%	15,09%
Inter-sites (référence)	5,87%	3,95%	15,76%	13,59%	15,83%
Total	100%	100%	100%	100%	100%

FIG. 10.8 – Types de liens.

On remarque que la proportion de liens de composition par rapport aux liens de cheminement est comparable (entre 2,5 et 4), mis à part dans le cas des pages personnelles où elle atteint un facteur de 8. La collection *PagePerso* comporte une structure plus simple, avec peu de liens de composition et un grand nombre de liens de référence ou de liens de séquence de type “page suivante”. Cela est cohérent avec le fait que cette collection est celle qui obtient la plus faible profondeur moyenne des sites (cf. tableaux 10.3). Nous pensons que l’avantage très net, parmi les liens de cheminement, des liens séquentiels (passant d’un nœud de l’arborescence à un de ses frères) sur les liens non séquentiels (qui proposent des “chemins de traverse” pour la consultation des sites) est dû au fait que les auteurs de pages Web, n’étant pas habitués aux principes de l’hypertexte, reviennent souvent à une description arborescente d’un site Web pour rester dans le domaine bien connu des documents structurés. Enfin, nous obtenons un pourcentage de liens inter-sites élevé (entre 4 et 16%) par rapport aux 2,5% obtenus par Gurrin [Gurrin et al.99].

10.5 Validation du typage de liens

Le typage de liens étant un aspect déterminant de l’application de notre modèle d’hyperdocuments au Web, nous avons initié une campagne de validation des méthodes employées pour extraire la structure. Pour cela, nous avons mis en place une plateforme d’évaluation des types de liens, qui propose une interface d’aide au jugement manuel. Une copie d’écran de l’interface se trouve en annexe E.3. L’interface permet de choisir une page Web à évaluer, et l’affichage simultané de la page réelle et de ses liens permet aux juges de naviguer à partir de la page afin d’avoir toutes les informations nécessaires pour porter un jugement sur le type de chaque lien.

Les juges ont le choix entre cinq alternatives : les trois types de liens de notre modèle (composition, cheminement, référence) auxquels nous avons rajouté les solutions “Ne sait pas” et “Autre type”. Les liens internes aux pages ne sont pas considérés dans cette évaluation. Le typage de lien obtenu par notre système n’est pas révélé aux juges, et le dépouillement automatique des résultats évalue le typage à l’aide d’une adaptation de la mesure de rappel/précision pour chacun des trois types de liens qui nous intéressent plus particulièrement. Par exemple, pour le type “composition”, le rappel est la proportion de liens de ce type

qui ont été correctement typés par le système par rapport à ceux existants, jugés manuellement. La précision est la proportion de liens correctement typés parmi tous les liens jugés par le système comme représentant une composition.

Les premiers résultats de notre évaluation, portant sur les jugements par quatre utilisateurs d'environ 100 pages Web contenant 1 750 liens, montrent une bonne qualité du typage automatique de liens. Nous avons utilisé à cet effet un sous-ensemble de la collection IMAG qui a le mérite d'être connue des juges. La majorité des pages sont des pages "de surface", c'est-à-dire des pages se situant à une faible profondeur, comme les pages principales. De ce fait, la répartition des liens n'est pas représentative du reste du corpus, comme le montre en particulier le nombre très important de liens du type "référence". Le tableau 10.9 présente le rappel et la précision pour chaque type de lien :

Type de lien	Nombre de jugements	Précision	Rappel
Composition	339	60%	86%
Cheminement	317	43%	32%
Référence	871	100%	100%
Ne sait pas	91	-	-
Autre type	136	-	-
Tous les types	1 754	78%	79%

FIG. 10.9 – Évaluation du typage de liens.

Ces résultats montrent la difficulté de l'extraction d'information du Web. Les relations de référence sont triviales à identifier dans le cas où un seul site est présent sur le serveur Web, et où tous les liens sortants du site sont effectivement des liens de référence : il y a donc une limite physique des hyperdocuments. Par contre, il est plus difficile de détecter ce type de liens dans le cas d'une limite uniquement logique entre les hyperdocuments. Les relations de composition sont identifiées avec un bon score de rappel et de précision. La méthode reste à améliorer, mais ces résultats montrent qu'il est possible d'extraire une structure hiérarchique de bonne qualité uniquement en se basant sur des aspects syntaxique des URLs ou des patrons fréquents de structure. Il est par contre plus difficile d'identifier les relations de cheminement sans information supplémentaire sur les documents reliés. Par ailleurs, nous nous sommes rendu compte au cours de la validation manuelle des types de liens que cette tâche était très délicate, même manuellement. En effet, la stratégie de description de l'auteur n'est pas toujours très lisible ni très explicite.

Quatrième partie

Conclusion

Chapitre 11

Conclusion

11.1 Synthèse et apport de la thèse

La structure du Web est un aspect essentiel de la description de l'information. Pourtant, bien que les méthodes développées pour intégrer cette structure dans le processus de RI paraissent donner de bons résultats sur le Web (cf. le moteur Google [Brin et al.98]), de nombreuses expériences ont montré qu'il n'y a pas de gain significatif comparé aux SRI académiques (cf. TREC, [Savoy et al.00a], [Hawking00], [Craswell et al.01]). Pour expliquer ces performances décevantes, nous avons mis en cause le "sac de nœuds" et le "sac de liens" du Web (cf. chapitre 5), qui sont souvent utilisés tels quels pour l'application de propagation d'information, de popularité ou de pertinence. Or, nous pensons que les liens et les pages doivent être utilisés avec plus de finesse, en tenant compte de leur nature et du rôle qu'ils jouent dans l'hypertexte "Web".

Notre objectif est de considérer le Web du point de vue de la sémantique, en représentant la structure de l'information et non plus seulement la structure physique des documents (comme une page HTML). Le modèle de RI présenté propose un point de vue original sur l'indexation de documents du Web en utilisant sa structure, fondé sur un modèle d'hyperdocuments en contexte ($\mathcal{HDOCC}$) . Ce modèle considère quatre facettes fondamentales de la description d'information sur le Web : le **contenu** et les différents types de structures du Web. Il s'agit des structure relatives à la **composition** (relation de composition), à la **lecture** linéaire ou déambulatoire (relation de cheminement), et au **contexte** composé de l'espace d'information référençant un document et de l'espace d'information accessible (relation de référence).

La sémantique d'un hyperdocument hd est extraite en tenant compte de la structure du Web dans les deux parties du modèle de RI : le modèle d'hyperdocuments et le processus d'indexation. Un hyperdocument hd est modélisé par un contenu ds au sens des documents structurés (contenu + structure hiérarchique), basé sur un ensemble de documents atomiques a. Un hyperdocument comprend également un ensemble de chemins de lecture $\{\vec{ch}\}$ (contenu + structure de cheminement) et un contexte : aut, ray, *méta-info* et *info-acc* (autorité + rayonnement + méta-information + information accessible).

Le modèle permet la RI structurée en contexte. Le Web n'est plus considéré comme un ensemble de documents atomiques et indépendants, mais comme un continuum d'informations interdépendantes, dont l'indexation est inspirée des théories de compréhension et de construction du sens au cours d'une lecture (progression thématique). De plus, la propagation d'information ou de pertinence avec le modèle d'hyperdocuments $\mathcal{HD}$ considère la structure, en distinguant les quatre types de documents et les trois types de relations. En particulier, le contexte est représenté pour chaque type de document (document atomique, $\mathcal{DS}$, $\mathcal{HD}$ et chemin de lecture).

Le modèle de RI Structurée permet à l'utilisateur de retrouver de l'information en considérant les aspects suivants :

Granularité : la prise en compte des différents niveaux de granularité permet de retrouver des documents structurés qui n'auraient pas été retrouvés par un système classique, comme par exemple un document fragmenté en plusieurs pages HTML dans lesquelles sont dispersés les termes de la requête. La requête intègre un paramètre indiquant la granularité recherchée.

Cheminement : les chemins de lecture permettent de retrouver un sous-ensemble de pages d'un hyperdocument qui n'aurait été retrouvé ni avec un système classique, ni avec une méthode considérant différents niveaux de granularité. En effet, un chemin de lecture parcourt des pages d'un document structuré qui peuvent être noyées parmi plusieurs centaines d'autres pages. Les chemins aident aussi à la consultation des résultats, en évitant les problèmes de désorientation et de surcharge cognitive. L'utilisateur profite des chemins proposés par l'auteur, qui doivent contenir toute l'information recherchée, être de taille minimale et être ordonnés de manière à décrire un développement progressif et cohérent de l'information.

Contexte : la méta-information et l'information accessible sont indexées et utilisées à l'interrogation, qui combine donc les différentes sources de description de l'information. De plus, la requête intègre un paramètre précisant le caractère focalisée ou défocalisée de la recherche.

Autorité et rayonnement : les scores de popularité et de rayonnement sont utilisés aux différents niveaux de granularité, et sont appliqués à des entités de même nature (document atomique, $\mathcal{DS}$, $\mathcal{HD}$ et chemin de lecture).

Le premier apport de notre modèle est la formalisation de l'information structurée du Web qui mène à la description et au typage des liens et de différentes granularités d'information et de leur impact sur la construction du sens (cf. chapitre 6).

Le deuxième apport réside dans l'élaboration d'un modèle intégrant l'indexation de chemins de lecture et prenant en compte la délinéarisation de l'hypertexte Web. Ainsi, pour l'exemple présenté dans la section 2.7.1, le SRIS doit proposer un chemin dans l'hyperdocument "Site de MRIM", permettant de consulter les pages pertinentes pour la requête : "*les travaux, les publications et les développements sur la recherche d'images*". Par exemple :

"Axe Images" $\rightarrow$ "Projets images" $\rightarrow$ "Page perso. (vidéo)" $\rightarrow$ "Publications images"

Le troisième apport est la formalisation de méthodes de propagation d'information, de pertinence et de popularité dans un modèle identifiant différents types de liens et différents niveaux de granularité, ce qui permet une propagation "fine" à l'indexation et à l'interrogation. En effet, on ne fait pas la même propagation le long des liens de composition, cheminement ou référence. De plus, cela permet de diminuer considérablement le coût de l'indexation et de l'interrogation, avec une propagation "ciblée" entre documents de même type. Par exemple, l'indexation d'un document structuré réalise une propagation d'information des feuilles de sa hiérarchie jusqu'à son sommet. L'indexation d'un chemin de lecture réalise également une propagation d'information pour extraire son information accessible.

11.2 Expérimentations et évaluation

Le premier type d'expérimentations, que nous avons présentées dans le chapitre 9, avaient pour objectif l'évaluation des aspects *composition*, *cheminement* et *référence* de notre modèle. Mais les collections de test existantes ne sont pas adaptées à la tâche d'évaluation particulière d'un SRIS basé sur un modèle tel que celui que nous proposons. En effet, la pertinence atomique sur laquelle se basent ces collections ne satisfait pas un modèle pour lequel un *document* pertinent est un chemin de lecture parcourant dans un ordre précis une liste de documents atomiques placés dans un contexte (cf. section 9.3).

Pour cette raison, nous avons choisi de construire nos propres collections de test. La première d'entre elles, la collection CLIPS, construite manuellement à partir de pages du Web, a permis de montrer l'intérêt de l'information accessible combinée avec le contenu. Ensuite, nous avons développé une méthode de construction automatique de collections structurées, dont nous pouvons modifier les paramètres afin de fabriquer des collections variées. Ces collections ont été utilisées afin d'expérimenter notre approche d'indexation de documents structurés et de chemins de lecture. Nous avons montré l'intérêt d'une indexation au niveau des documents structurés, dans le cas de documents atomiques difficiles à retrouver car isolés de leur contexte. Nous avons également montré, dans certains cas, l'intérêt d'une indexation de chemins de lecture basée sur les principes d'accumulation et de mémoire de lecture. Les résultats obtenus sont encourageants malgré les contraintes liées aux collections de test. Cependant, la construction automatique de collections de test a ses limites, et la création d'une collection développée spécifiquement et composée de documents Web structurés est une nécessité pour les expérimentations futures de notre SRI Structurée (SRIS).

Au cours d'autres expérimentations, le modèle d'hyperdocuments a été implanté dans un SRIS complet (robot pour collecter les pages, analyse des collections, typage des pages et des liens, indexation vectorielle, modules d'extraction des chemins de lecture et du contexte, etc.) que nous avons décrit succinctement. Nous avons montré la faisabilité de notre approche dans sa globalité et sur le Web, et cela nous a permis d'identifier les principales difficultés de la mise en œuvre de notre modèle dans le contexte du Web et à grande échelle. En particulier, nous avons mené des expérimentations sur le typage des liens [Gery et al.01], qui est à la fois un des aspects les plus importants du modèle (il conditionne l'indexation des documents structurés, des chemins de lecture et du contexte) et une difficulté majeure de sa mise en

œuvre. Nous avons débuté une expérimentation de validation du typage de liens, dont les premiers résultats montrent la bonne qualité des liens de référence et de composition et la difficulté d'identifier les liens de cheminement.

11.3 Perspectives

La perspective à ce travail qui nous semble la plus intéressante est le développement ou l'adaptation de méthodes de propagation de pertinence ou d'information adaptées aux principes du modèle d'hyperdocuments. Celui-ci formalise la méta-information et l'information accessible qui sont ensuite combinées avec les "scores" d'autorité et de rayonnement. Mais nous n'avons pas exploré l'application de ces techniques (mis à part l'information accessible), faute de collection de test adaptée. Les méthodes de propagation de pertinence fondées sur la granularité de document d'une page HTML ont montré leurs limites [Hawking et al.01a] et nous pensons que des techniques de propagation, appliquées par exemple aux chemins de lecture, représentent une piste de recherche très prometteuse.

Une autre perspective concerne l'évaluation d'un SRI Structurée (SRIS). Nous avons discuté des limitations des collections de test existantes, qui ne sont pas adaptées au Web. Les inconvénients majeurs des méthodes d'évaluation classiques de SRI sont l'atomicité des jugements de pertinence et l'indépendance des documents dans le jugement de pertinence. Un document est jugé pertinent pour son contenu uniquement, sans tenir compte par exemple de son voisinage. Or, la consultation de documents dans un hypertexte ne se fait pas de manière atomique, la pertinence ne doit donc pas être atomique. Ce problème a été évoqué par Craswell [Craswell et al.01], et la piste de TREC qui propose de rechercher uniquement des "pages principales" de sites est intéressante, mais n'offre pas encore de notion de pertinence structurée. L'ordre des pages et les relations entre les pages ne sont pas non plus considérés.

La première solution pour évaluer un SRIS est l'utilisation d'une collection de test existante, ce qui n'est pas satisfaisant en raison des limitations évoquées de ces collections. Une deuxième solution, très coûteuse, consiste à construire manuellement une collection de test. Une troisième solution consiste à fabriquer une collection de test à partir d'une collection existante. Ayant expérimenté ces trois méthodes, nous envisageons une quatrième solution qui consiste à adapter le principe de "précision comparative" à l'évaluation d'un SRIS, en prenant en compte les critères de la pertinence dans ce contexte.

Nous pensons qu'il faut radicalement changer notre vision de l'évaluation des SRI dans le contexte du Web. Plus particulièrement, avec un modèle qui retrouve des chemins de lecture en contexte, il faut s'interroger sur la notion de pertinence d'un chemin en fonction de sa granularité, de son contexte, de son information accessible, etc. Un SRIS devrait être évalué dans le cas d'une recherche de chemins de différentes granularités, focalisée ou défocalisée, etc. L'évaluation d'un SRIS pourrait se faire suivant quatre axes : les deux axes classiques de précision et de rappel, et deux nouveaux axes de granularité et de focus. Ainsi, nous pourrions juger distinctement la qualité d'un système pour une recherche focalisée d'un document atomique, ou pour une recherche défocalisée d'un chemin de lecture.

Cinquième partie

Bibliographie et glossaire

Bibliographie

[Abchiche01] Abchiche (Malika). – Intégration des liens hypertextes dans la recherche d'information. *19ème Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'01)*, pp. 253–266. – Martigny, Suisse, Mai 2001.

[Aguiar et al.00] Aguiar (Fernando) et Beigbeder (Michel). – Des moteurs de recherche efficaces pour des systèmes hypertextes grâce aux contextes des noeuds. *Colloque International : Technologies de l'Information et de la Communication dans les Enseignements d'ingénieurs et dans l'industrie (TICE'2000)*. – Troyes, France, Octobre 2000.

[Albert et al.99] Albert (Réka), Jeong (Hawoong) et Barabási (Albert-László). – The diameter of the World Wide Web. *Nature*, vol. 401, Septembre 1999, pp. 130–131.

[Altavista] http ://www.altavista.com. Altavista.

[Amann94] Amann (Bernd). – *Interrogation d'Hypertextes*. – Paris, France, Thèse de PhD, Centre d'Etude et de Recherche en Informatique (Conservatoire National des Arts et Métiers), Février 1994.

[Amitay98] Amitay (Einat). – Using Common hypertext links to identify the best phrasal description of target Web document. *Workshop on Hypertext IR for the Web (SIGIRW'98)*. – Melbourne, Australie, Août 1998.

[Atzeni et al.97] Atzeni (Paolo), Mecca (Giansalvatore) et Merialdo (Paolo). – Semi-structured and Structured Data in the Web : Going Back and Forth. *1er ACM SIGMOD Workshop on Management of Semistructured Data (MSD'97)*. – Tucson, Arizona, États-Unis, Mai 1997.

[Barwise89] Barwise (Jon). – *The Situation in Logic*. – CSLI Publications, Mars 1989.

[Beckett97] Beckett (Dave). – 30% Accessible - A Survey of the UK Wide Web. *6ème World Wide Web Conference (WWW'97)*. – Santa Clara, Californie, États-Unis, Avril 1997.

[Bergman00] Bergman (Michael K.). – *The Deep Web : Surfacing Hidden Value*. – Rapport technique, BrightPlanet, Juillet 2000.

[BH52] Bar-Hillel (Yehoshua). – Semantic information and its measures. *8ème Cybernetics - circular, causal and feedback mechanisms in biological and social systems (Cybernetics'52)*, pp. 33–48. – New-York, États-Unis, 1952.

[BH64] Bar-Hillel (Yehoshua). – *Language and Information : selected essays on their theory and application*. – Addison-Wesley, Janvier 1964.

[BL et al.92] Berners-Lee (Tim), Cailliau (Robert), Groff (Jean-Francois) et Pollermann (Bernd). – World-Wide Web : The Information Universe. *Electronic Networking : Research, Applications and Policy*, vol. 1, 1992, pp. 74–82.

[BL et al.94] Berners-Lee (Tim), Masinter (L.) et McCahill (M.). – *Uniform Resource Locators (URL) (RFC1738)*. – Rapport technique, IETF, The Internet Engineering Task Force (IETF), Décembre 1994.

[BL et al.98] Berners-Lee (Tim), Fielding (Roy T.), Irvine (U.C.) et Masinter (L.). – *Uniform Resource Identifiers (URI) : Generic Syntax (RFC2396)*. – Rapport technique, IETF, The Internet Engineering Task Force (IETF), Août 1998.

[BL89] Berners-Lee (Tim). – *Information Management : A Proposal*. – Rapport technique, Genève, Suisse, Organisation Européenne pour la Recherche Nucléaire (CERN), Mars 1989.

[Blake et al.94] Blake (G. Elizabeth), Consens (Mariano P.), Kilpeläinen (Pekka), Åke Larson (Per), Snider (T.) et Tompa (Frank Wm.). – Text / Relational Database Management Systems : Harmonizing SQL and SGML. *1er Applications of Databases (ADB'94)*, pp. 267–280. – Vadstena, Suède, Juin 1994.

[Borodin et al.01] Borodin (Allan), Roberts (Gareth O.), Rosenthal (Jeffrey S.) et Tsaparas (Panayiotis). – Finding Authorities and Hubs From Link Structures on the World Wide Web. *10ème World Wide Web Conference (WWW'01)*. – Hong-Kong, Chine, Mai 2001.

[Botafogo et al.91] Botafogo (Rodrigo A.) et Shneiderman (Ben). – *Identifying Aggregates in Hypertext Structures*. – Rapport technique, Maryland, États-Unis, University of Maryland, Avril 1991.

[Botafogo et al.92] Botafogo (Rodrigo A.), Rivlin (Ehud) et Shneiderman (Ben). – Structural Analysis of Hypertexts : Identifying Hierarchies and Useful Metrics. *ACM Transactions on Information Systems*, vol. 10, Avril 1992, pp. 142–180.

[Bourdoncle et al.00] Bourdoncle (François) et Bertin (Patrice). – Recherche d'aiguilles dans une botte de liens. *La recherche*, Février 2000, pp. 66–72.

[Boyan et al.96] Boyan (Justin), Freitag (Dayne) et Joachims (Thorsten). – A machine learning architecture for optimizing Web search engine. *AAAI*

Workshop on Internet-Based Information Systems (W-AAAI'96). – Portland, Oregon, États-Unis, Août 1996.

[Bray et al.00] Bray (Tim), Paoli (Jean), Sperberg-McQueen (C.M.) et Maler (Eve). – *Extensible Markup Language (XML) 1.0 (Second Edition)*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Octobre 2000.

[Bray96] Bray (Tim). – Measuring the Web. *5ème World Wide Web Conference (WWW'96)*, pp. 994–1005. – Paris, France, Mai 1996.

[Brin et al.98] Brin (Sergey) et Page (Lawrence). – The anatomy of a large-scale Hypertextual Web Search Engine. *7ème World Wide Web Conference (WWW'98)*. – Brisbane, Australie, Avril 1998.

[Broder et al.00] Broder (Andrei), Kumar (Ravi), Maghoul (Farzin), Raghavan (Prabhakar), Rajagopalan (Sridhar), Stata (Raymie), Tomkins (Andrew) et Wiener (Janet). – Graph structure in the Web. *9ème World Wide Web Conference (WWW'00)*. – Amsterdam, Pays-Bas, Mai 2000.

[Bush45] Bush (Vannevar). – As We May Think. *The Atlantic Monthly*, vol. 176, Juillet 1945, pp. 101–108.

[Carchiolo et al.00] Carchiolo (Vincenza), Longheu (Alessandro) et Malgeri (Michele). – Extracting Logical Schema from the Web. *International Workshop on Text and Web Mining (PRICAI'00)*, pp. 64–71. – Melbourne, Australie, Août 2000.

[Carriere et al.97] Carrière (Jeromy) et Kazman (Rick). – WebQuery : Searching and Visualizing the Web through Connectivity. *6ème World Wide Web Conference (WWW'97)*. – Santa Clara, Californie, États-Unis, Avril 1997.

[Chakrabarti et al.98] Chakrabarti (Soumen), Dom (Byron E.), Raghavan (Prabhakar), Rajagopalan (Sridhar), Gibson (David) et Kleinberg (Jon M.). – Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text. *7ème World Wide Web Conference (WWW'98)*, pp. 65–74. – Brisbane, Australie, Avril 1998.

[Chakrabarti01] Chakrabarti (Soumen). – Integrating the Document Object Model with Hyperlinks for Enhanced Topic Distillation and Information Extraction. *10ème World Wide Web Conference (WWW'01)*. – Hong-Kong, Chine, Mai 2001.

[Chiaramella et al.96] Chiaramella (Yves), Mulhem (Philippe) et Fourel (Franck). – *A Model for Multimedia Information Retrieval*. – Rapport technique, Grenoble, Laboratoire CLIPS-IMAG, Juillet 1996.

[Chiaramella97] Chiaramella (Yves). – Browsing and Querying : Two Complementary Approaches for Multimedia Information Retrieval. *Confe-*

rence on Hypertext - Information Retrieval - Multimedia (HIM'97). – Dortmund, Allemagne, Septembre 1997.

[Christophides et al.94] Christophides (Vassilis) et Rizk (Antoine). – Querying Structured Documents with Hypertext Links using OODBMS. *European Conference on Hypertext Technology (ECHT'94)*, pp. 186–197. – Edinburgh, Écosse, Septembre 1994.

[Christophides96] Christophides (Vassilis). – *Documents Structurés et Base de Données Objet*. – Paris, France, Thèse de PhD, Centre d'Etude et de Recherche en Informatique (Conservatoire National des Arts et Métiers), Octobre 1996.

[Clark et al.99] Clark (James) et DeRose (Steve). – *XML Path Language (XPath) Version 1.0*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Novembre 1999.

[Clement94] Clément (Jean). – Afternoon, a story, du narratif au poétique dans l'oeuvre hypertextuelle. *A :LITTÉRATURE, numéro spécial des Cahiers du CIRCAV*, 1994.

[Clement95a] Clément (Jean). – Du texte à l'hypertexte : vers une épistémologie de la discursivité hypertextuelle. *Hypertextes et Hypermédias : Réalisations, Outils et Méthodes (HH'95)*. – Paris, France, Mai 1995.

[Clement95b] Clément (Jean). – L'hypertexte de fiction : naissance d'un nouveau genre ? *Littérature et informatique : La littérature générée par ordinateur*, éd. par Alain Vuillmemin et Michel Lenoble. – Artois Presses Université, 1995.

[Craswell et al.01] Craswell (Nick), Hawking (David) et Robertson (Stephen). – Effective Site Finding using Link Anchor Information. *24ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*, pp. 250–257. – Nouvelle Orleans, Louisiane, États-Unis, Septembre 2001.

[Crivellari et al.00] Crivellari (Franco) et Melucci (Massimo). – Web Document Retrieval Using Passage Retrieval, Connectivity Information, and Automatic Link Weighting–TREC-9 Report. *9ème Text REtrieval Conference (TREC'00)*. – Gaithersburg, Maryland, États-Unis, Novembre 2000.

[Croft et al.89a] Croft (W. Bruce), Lucia (T.J.), Gringean (J.) et Willett (Peter). – Retrieving documents by plausible inference : an experimental study. *Information Processing & Management*, vol. 25, Janvier 1989, pp. 599–614.

[Croft et al.89b] Croft (W. Bruce) et Turtle (Howard). – A Retrieval Model for Incorporating Hypertext Links. *2ème ACM Conference on Hypertext (HT'89)*, pp. 213–224. – Pittsburgh, Pennsylvanie, États-Unis, Novembre 1989.

[Croft et al.93] Croft (W. Bruce) et Turtle (Howard). – Retrieval Strategies for Hypertext. *Information Processing & Management*, vol. 29, Mai 1993, pp. 313–324.

[Danes74] Danes (Frantisek). – Functional sentence perspective and the organization of the text. *Papers on functional sentence perspective*, éd. par Frantisek Danes, pp. 106–208. – Academia, Prague, Czech Republic, 1974.

[Davison00a] Davison (Brian D.). – Topical locality in the Web. *23ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00)*, pp. 272–279. – Athènes, Grèce, Juillet 2000.

[Davison00b] Davison (Brian D.). – *Topical Locality in the Web : Experiments and Observations*. – Rapport technique, New Brunswick, États-Unis, University of New Jersey, Juillet 2000.

[Defude86] Defude (Bruno). – *Etude et réalisation d'un système intelligent de recherche d'informations : Le prototype IOTA*. – Grenoble, Thèse de PhD, Institut National Polytechnique de Grenoble, Janvier 1986.

[Derose et al.01a] DeRose (Steve), Maler (Eve) et Daniel (Ron). – *XML Pointer Language (XPointer) Version 1.0*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Septembre 2001.

[Derose et al.01b] DeRose (Steve), Maler (Eve) et Orchard (David). – *XML Linking Language (XLink) Version 1.0*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Juin 2001.

[Dunlop et al.93] Dunlop (Mark D.) et van Rijsbergen (Cornelis Joost). – Hypermedia and free text retrieval. *Information Processing & Management*, vol. 29, Mai 1993, pp. 287–298.

[Encyclopaedia] http ://www.universalis-edu.com. Encyclopaedia Universalis.

[Engelbart63] Engelbart (Douglas C.). – A Conceptual Framework for the Augmentation of Man's Intellect. *Vistas in Information Handling*, éd. par Howerton et Weeks, pp. 1–29. – Spartan Books, Washington, D. C., 1963.

[Estival et al.81] Estival (Robert) et Meyriat (Jean). – La dialectique de l'écrit et du document. Un effort de synthèse. *Schéma et schématisation*, 1981, pp. 82–91.

[Fielding et al.99] Fielding (Roy T.), Gettys (J.), Mogul (J.), Frystyk (H.), Masinter (L.), Leach (P.) et Berners-Lee (Tim). – *Hypertext Transfer Protocol – HTTP/1.1*. – Rapport technique, IETF, The Internet Engineering Task Force (IETF), Juin 1999.

[Fourel98] Fourel (Franck). – *Modélisation, indexation et recherche de documents structurés*. – Grenoble, Thèse de PhD, Université Joseph Fourier, Février 1998.

[Frei et al.92] Frei (Hans Peter) et Stieger (Daniel). – Making use of hypertext links when retrieving information. *4ème European Conference on Hypertext Technology (ECHT'92)*, pp. 102–111. – Milan, Italie, Novembre 1992.

[Frisse et al.89] Frisse (Mark E.) et Cousins (Steve B.). – Information Retrieval from Hypertext : Update on the Dynamic Medical Handbook Project. *2ème ACM Conference on Hypertext (HT'89)*, pp. 199–212. – Pittsburgh, Pennsylvanie, États-Unis, Novembre 1989.

[Frisse88] Frisse (Mark E.). – Searching for Information in a Hypertext Medical Handbook. *Communications of the ACM*, vol. 31, Juillet 1988, pp. 880–886.

[FS97] Fayet-Scribe (Sylvie). – Chronologie des supports, des dispositifs et des outils de repérage de l'information. *Solaris*, Janvier 1997.

[Fuller et al.93] Fuller (Michael), Mackie (Eric), Sacks-Davis (Ron) et Wilkinson (Ross). – Structured Answers for a Large Structured Document Collection. *16ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93)*, pp. 204–213. – Pittsburgh, PA, États-Unis, Juin 1993.

[Gardarin et al.96] Gardarin (Georges) et Yoon (S.). – HyWeb : un système d'interrogation orienté objet pour le Web. *12ème Journées Bases de Données Avancées (BDA'96)*, pp. 205–224. – Cassis, France, Août 1996.

[Gdt] http ://www.granddictionnaire.com/. GDT : le grand dictionnaire terminologique.

[Gery et al.01] Géry (Mathias) et Chevallet (Jean-Pierre). – Toward a Structured Information Retrieval System on the Web : Automatic Structure Extraction of Web Pages. *1er Workshop on Web Dynamics (WebDyn'01)*. – Londres, Royaume-Uni, Janvier 2001.

[Gery99] Géry (Mathias). – SmartWeb : Recherche de Zones de Pertinence sur le World Wide Web. *17ème Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'99)*, pp. 133–147. – La Garde, France, Juin 1999.

[Gery02] Géry (Mathias). – Un modèle d'hyperdocument en contexte pour la recherche d'information structurée. *RSTI-ISI, numéro spécial "Recherche et filtrage d'information"*, vol. 7, 2002, pp. 11–44.

[Gibson et al.98] Gibson (David), Kleinberg (Jon M.) et Raghavan (Prabhakar). – Inferring Web Communities from Link Topology. *9ème ACM Conference on Hypertext (HT'98)*, pp. 225–234. – Pittsburgh, États-Unis, Juin 1998.

[Google] http ://www.google.com. Google.

[Greisdorf et al.99] Greisdorf (Howard) et Spink (Amanda). – Regions of Relevance : Approaches to Measurement for Enhanced Precision. *21ème Information Retrieval Specialist Group Annual Colloquium on IR Research (IRSG'99)*. – Glasgow, Ecosse, Avril 1999.

[Guinan et al.92] Guinan (Catherine) et Smeaton (Alan F.). – Information Retrieval from Hypertext using Dynamically Planned Guided Tours. *4ème European Conference on Hypertext Technology (ECHT'92)*, pp. 122–130. – Milan, Italie, Novembre 1992.

[Gurrin et al.99] Gurrin (Cathal) et Smeaton (Alan F.). – A Connectivity Analysis Approach to Increasing Precision in Retrieval from Hyperlinked Documents. *8ème Text REtrieval Conference (TREC'99)*. – Gaithersburg, Maryland, États-Unis, Novembre 1999.

[Gurrin et al.00] Gurrin (Cathal) et Smeaton (Alan F.). – Dublin City University Experiments in Connectivity Analysis for TREC-9. *9ème Text REtrieval Conference (TREC'00)*. – Gaithersburg, Maryland, États-Unis, Novembre 2000.

[Gutenberg54] Gutenberg (Johannes). – *The Gutenberg Bible*. – Johannes Gutenberg, Mainz, Germany, 1454.

[Harmandas et al.97] Harmandas (V.), Sanderson (Mark) et Dunlop (Mark D.). – Image Retrieval by Hypertext Links. *20ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'97)*, pp. 296–303. – Philadelphie, PA, États-Unis, Juillet 1997.

[Hawking et al.99] Hawking (David), Craswell (Nick), Thistlewaite (Paul) et Harman (Donna). – Results and Challenges in Web Search Evaluation. *8ème World Wide Web Conference (WWW'99)*. – Toronto, Canada, Mai 1999.

[Hawking et al.01a] Hawking (David) et Craswell (Nick). – Overview of the TREC-2001 Web Track. *10ème Text REtrieval Conference (TREC'01)*, pp. 61–67. – Gaithersburg, Maryland, États-Unis, Novembre 2001.

[Hawking et al.01b] Hawking (David), Craswell (Nick), Bailey (Peter) et Griffiths (Kathleen). – Measuring Search Engine Quality. *Journal of Information Retrieval*, vol. 4, Avril 2001, pp. 33–59.

[Hawking00] Hawking (David). – Overview of the TREC-9 Web Track. *9ème Text REtrieval Conference (TREC'00)*. – Gaithersburg, Maryland, États-Unis, Novembre 2000.

[Hors et al.02] Hors (Arnaud Le), Hégaret (Philippe Le), Nicol (Gavin), Wood (Lauren), Champion (Mike) et Byrne (Steve). – *Document Object Model (DOM) Level 3 Core Specification*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Janvier 2002.

[Jakobson63] Jakobson (Roman). – *Essais de linguistique générale*. – Editions de minuit, Paris, 1963.

[jH et al.97] jen Hsu (Jane Yung) et tau Yih (Wen). – Template-Based Information Mining from HTML Documents. *14ème National Conference on Artificial Intelligence (AAAI'97)*, pp. 256–262. – Providence, Rhode Island, États-Unis, Juillet 1997.

[Joyce85] Joyce (Michael). – *Afternoon, a story*. – Eastgate Systems, Watertown, 1985.

[Jun et al.97] Jun (Young-Mi), Yook (Hyun-Gyoo) et Park (Myong-Soon). – A link based information retrieval model in WWW. *4ème International Conference on Multimedia Modeling (MMM'97)*, pp. 397–402. – Singapour, Novembre 1997.

[Kazai et al.01] Kazai (Gabriella), Lalmas (Mounia) et Rölleke (Thomas). – A Model for the Representation and Focussed Retrieval of Structured Documents based on Fuzzy Aggregation. *9ème Conference on String Processing and Information Retrieval (SPIRE'01)*. – Laguna de San Rafael, Chili, Novembre 2001.

[Kerkouba84] Kerkouba (Dalila). – *Une méthode d'indexation automatique des documents fondée sur l'exploitation de leurs propriétées structurelles. Application à un corpus technique*. – Grenoble, Thèse de PhD, Institut National Polytechnique de Grenoble, Novembre 1984.

[Kessler63] Kessler (M.M.). – Bibliographic coupling between scientific papers. *American Documentation*, vol. 14, Janvier 1963, pp. 10–25.

[Kleinberg et al.01] Kleinberg (Jon M.) et Lawrence (Steve). – The Structure of the Web. *Science*, vol. 294, Novembre 2001, pp. 1849–1850.

[Kleinberg98] Kleinberg (Jon M.). – Authoritative Sources in a Hyperlinked Environnement. *9ème Symposium on Discrete Algorithms (SODA'98)*, pp. 668–677. – San Francisco, Californie, États-Unis, Janvier 1998.

[Kleinberg99] Kleinberg (Jon M.). – Authoritative Sources in a Hyperlinked Environnement. *Journal of the ACM*, vol. 46, Septembre 1999, pp. 604–632.

[KO80] Kerbrat-Orecchioni (Catherine). – *L'énonciation de la subjectivité dans le langage*. – Armand Colin, Paris, 1980.

[Konopnicki et al.95] Konopnicki (David) et Schmueli (Oded). – W3QS : A Query System for the World-Wide Web. *21ème International Conference on Very Large Data Bases (VLDB'95)*, pp. 54–65. – Zurich, Suisse, Septembre 1995.

[Kumar et al.00] Kumar (Ravi), Raghavan (Prabhakar), Rajagopalan (Sridhar), Sivakumar (D.), Tomkins (Andrew) et Upfal (Eli). – The Web as a Graph. *19ème Symposium on Principles of Database Systems (PODS'00)*, pp. 1–10. – Dallas, Texas, États-Unis, Mai 2000.

[Lalmas et al.98] Lalmas (Mounia) et Ruthven (Ian). – Representing and Retrieving Structured Documents using the Dempster-Shafer Theory of Evidence : Modelling and Evaluation. *Journal of Documentation*, vol. 54, Décembre 1998, pp. 529–565.

[Lalmas et al.00] Lalmas (Mounia) et Moutogianni (Ekaterini). – A Dempster-Shafer indexing for the focussed retrieval of a hierarchically structured document space : Implementation and experiments on a Web museum collection. *6ème Conference on Recherche d'Informations Assistée par Ordinateur (RIAO'00)*. – Paris, France, Avril 2000.

[Larousse] http ://larousse.compuserve.com/larousse/dico.htm. Larousse : dictionnaire 72.000 mots.

[Larson96] Larson (Ray R.). – World Wide Web : an exploratory analysis of the intellectual structure of cyberspace. *Annual Meeting of the American Society for Information Science (ASIS'96)*. – Baltimore, Maryland, États-Unis, Octobre 1996.

[Laufer92] Laufer (Roger). – *Texte, Hypertexte, Hypermédia*. – Presses universitaires de France, Paris, 1992.

[Lawrence et al.98] Lawrence (Steve) et Giles (C. Lee). – Context and Page Analysis for Improved Web Search. *IEEE Internet Computing*, vol. 2, Juillet 1998, pp. 38–46.

[Lawrence et al.99] Lawrence (Steve) et Giles (C. Lee). – Accessibility of information on the Web. *Nature*, vol. 400, Juillet 1999, pp. 107–109.

[Lee et al.96] Lee (Yong Kyu), Yoo (Seong-Joon), Yoon (Kyoungro) et Berra (P. Bruce). – Index Structures for Structured Documents. *1er ACM International Conference on Digital Libraries (DL'96)*, pp. 91–99. – Bethesda, Maryland, États-Unis, Mars 1996.

[Lempel et al.00] Lempel (Ronny) et Moran (Shlomo). – The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC Effect. *9ème World Wide Web Conference (WWW'00)*. – Amsterdam, Pays-Bas, Mai 2000.

[Lyman et al.00] Lyman (Peter) et Varian (Hal R.). – *How Much Information ?* – Rapport technique, Berkeley, États-Unis, School of Information Management and Systems, University of California, Octobre 2000.

[Marchiori97] Marchiori (Massimo). – The Quest for Correct Information on the Web : Hyper Search Engines. *6ème World Wide Web Conference (WWW'97)*. – Santa Clara, Californie, États-Unis, Avril 1997.

[Masseglia02] Masseglia (Florent). – *Algorithmes et applications pour l'extraction de motifs séquentiels dans le domaine de la fouille de données : de l'incrémental au temps réel*. – Montpellier, Thèse de PhD, Université de Montpellier, Janvier 2002.

[Mcbryan94] McBryan (Oliver A.). – GENVL and WWWW : Tools for Taming the Web. *2ème World Wide Web Conference (WWW'94)*. – Chicago, États-Unis, Octobre 1994.

[Mendelzon et al.96] Mendelzon (Alberto), Mihaila (George A.) et Milo (Tova). – Querying the World Wide Web. *4ème International Conference on Parallel and Distributed Information Systems (PDIS'96)*, pp. 80–91. – Miami Beach, Floride, États-Unis, Décembre 1996.

[Mendelzon et al.97] Mendelzon (Alberto), Mihaila (George A.) et Milo (Tova). – Querying the World Wide Web. *Journal of Digital Libraries*, vol. 1, 1997, pp. 68–88.

[Mizzaro01] Mizzaro (Stephano). – A new measure of retrieval effectiveness (Or : What's wrong with precision and recall). *1er International Workshop on Information Retrieval (IR'01)*, pp. 43–52. – Oulu, Finlande, Septembre 2001.

[Murray et al.00] Murray (Brian H.) et Moore (Alvin). – *Sizing the Internet*. – Rapport technique, Washington, Cyveillance, Inc., Juillet 2000.

[Navarro95] Navarro (Gonzalo). – *A language for Queries on Structure and Contents of Textual Database*. – Santiago, Chili, Thèse de PhD, University of Chile, Avril 1995.

[Nelson65] Nelson (Ted H.). – The Hypertext. *World Documentation Federation (WDF'65)*. – 1965.

[Nelson80] Nelson (Ted H.). – *Literary Machines*. – Mindful Press, Sausalito, 1980.

[Nelson93] Nelson (Ted H.). – *Literary Machines*. – Mindful Press, Sausalito, 1993.

[Nestorov et al.97] Nestorov (Svetlozar), Abiteboul (Serge) et Motwani (Rajeev). – Inferring Structure in Semistructured Data. *1er ACM SIGMOD Workshop on Management of Semistructured Data (MSD'97)*. – Tucson, Arizona, États-Unis, Mai 1997.

[Nie90] Nie (Jian-Yun). – *Un modèle logique général pour les Systèmes de Recherche d'Informations. Application au prototype RIME*. – Grenoble, Thèse de PhD, Université Joseph Fourier, Juillet 1990.

[Nua01] NUA. – *Internet Surveys, http ://www.nua.ie/surveys*. – Rapport technique, Dublin, Irlande, NUA, Août 2001.

[Paradis96] Paradis (François). – *Un modèle d'indexation pour les documents textuels structurés*. – Grenoble, Thèse de PhD, Université Joseph Fourier, Novembre 1996.

[Pemberton et al.00] Pemberton (Steven) et et al. – *XHTML[tm] 1.0 : The Extensible HyperText Markup Language - A Reformulation of HTML 4 in XML 1.0*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Janvier 2000.

[Picard et al.01] Picard (Justin) et Savoy (Jacques). – Using Probabilistic Argumentation Systems to Search and Classify Web Sites. *IEEE Data Engineering Bulletin*, vol. 24, Septembre 2001, pp. 33–41.

[Picard00] Picard (Justin). – *Probabilistic Argumentation Systems Applied to Information Retrieval*. – Neuchâtel, Suisse, Thèse de PhD, Université de Neuchâtel, Institut d'informatique, Mai 2000.

[Pirolli et al.96] Pirolli (Peter), Pitkow (James) et Rao (Ramana). – Silk from a Sow's ear : extracting usable structures from the Web. *ACM Conference on Human Factors in Computing Systems (CHI'96)*, pp. 118–125. – Vancouver, Canada, Avril 1996.

[Rafiei et al.00] Rafiei (Davood) et Mendelzon (Alberto). – What is this Page Known for ? Computing Web Page Reputations. *9ème World Wide Web Conference (WWW'00)*. – Amsterdam, Pays-Bas, Mai 2000.

[Raggett et al.97] Raggett (Dave), Connolly (Dan), Berners-Lee (Tim), Maloney (Murray) et Quin (Liam). – *Hypertext Links in HTML*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Mars 1997.

[Raggett et al.99] Raggett (Dave), Hors (Arnaud Le) et Jacobs (Ian). – *HTML 4.01 Specification*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Décembre 1999.

[Riahi98] Riahi (Farshad). – Elaboration automatique d'une base de données à partir d'informations semi-structurées. *16ème Congrès Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'98)*, pp. 327–341. – Montpellier, France, Mai 1998.

[Rocchio71] Rocchio (J.J.). – Relevance feedback in information retrieval. *The SMART retrieval system : experiments in automatic document processing*, éd. par Gerald Salton, pp. 313–323. – Prentice Hall, 1971.

[Rouet92] Rouet (Jean-François). – Cognitive Processing of Hyperdocuments : When Does Nonlinearity Help ? *4ème European Conference on Hypertext Technology (ECHT'92)*, pp. 131–140. – Milan, Italie, Novembre 1992.

[Salton et al.83a] Salton (Gerald), Fox (Edward A.) et Wu (Harry). – Extended Boolean Information Retrieval. *Communications of the ACM*, vol. 26, Décembre 1983, pp. 1022–1036.

[Salton et al.83b] Salton (Gerald) et McGill (Michael J.). – *Introduction to Modern Information Retrieval*. – McGraw-Hill, Janvier 1983.

[Salton et al.90] Salton (Gerald) et Buckley (Chris). – Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, vol. 41, Juin 1990, pp. 288–297.

[Salton et al.94] Salton (Gerald) et Allan (James). – Automatic Text Decomposition and Structuring. *4ème Conference on Recherche d'Informations Assistée par Ordinateur (RIAO'94)*. – New-York, États-Unis, Octobre 1994.

[Salton71] Salton (Gerald). – *The SMART retrieval system : experiments in automatic document processing*. – Prentice Hall, 1971.

[Sarfati97] Sarfati (G.E.). – *Eléments d'analyse du discours*. – Nathan, Paris, 1997.

[Savoy et al.90] Savoy (Jacques) et Desbois (Daniel). – *Réseaux d'inférence bayésiens dans un système hypertexte : principes et premiers résultats*. – Rapport technique, Montréal, Université de Montréal, Département d'informatique et de recherche opérationnelle, Janvier 1990.

[Savoy et al.00a] Savoy (Jacques) et Picard (Justin). – Recherche documentaire sur le Web : Les hyperliens sont-ils vraiment utiles ? *5ème Journées Internationales d'Analyse Statistique des Données Textuelles (JADT'00)*, pp. 27–34. – Lausanne, Suisse, Mars 2000.

[Savoy et al.00b] Savoy (Jacques) et Rasolofo (Yves). – Report on the TREC-9 Experiment : Link-based Retrieval and Distributed Collections. *9ème Text REtrieval Conference (TREC'00)*. – Gaithersburg, Maryland, États-Unis, Novembre 2000.

[Savoy et al.01] Savoy (Jacques) et Picard (Justin). – Retrieval effectiveness on the web. *Information Processing & Management*, vol. 37, 2001, pp. 543–569.

[Savoy92] Savoy (Jacques). – Bayesian Inference Networks and Spreading Activation in Hypertext Systems. *Information Processing & Management*, vol. 28, Janvier 1992, pp. 389–406.

[Savoy96] Savoy (Jacques). – Citation schemes in hypertext information retrieval. *Information Retrieval and Hypertext*, éd. par Maristella Agosti (Alan F. Smeaton), pp. 99–120. – Kluwer Academic Publishers, Janvier 1996.

[Schwartz98] Schwartz (Candy). – Web Search Engines. *Journal of the American Society for Information Science*, vol. 49, Septembre 1998, pp. 973–982.

[Sedes98] Sèdes (Florence). – *BASES DOCUMENTAIRES - HYPERBASES : Proposition d'un modèle générique et contribution à la spécification d'un langage pour l'intégration et la manipulation d'informations semi-structurés*. – Rapport technique, Toulouse, Université Paul Sabatier, Décembre 1998.

[Shannon et al.49] Shannon (Claude Elwood) et Weaver (Warren). – *The Mathematical Theory of Communication*. – University of Illinois Press, Urbana, Illinois, Janvier 1949.

[Shannon et al.75] Shannon (Claude Elwood) et Weaver (Warren). – *Théorie mathématique de la communication*. – Retz-Centre d'Études et de Promotion de la Lecture, Paris, Janvier 1975.

[Small74] Small (Henry). – Co-citation in the Scientific literature : A New Measure of the Relationship Between Two Documents. *Essays of an Information Scientist*, vol. 2, Février 1974, pp. 28–31.

[Spertus et al.00] Spertus (Ellen) et Stein (Lynn Andrea). – Squeal : A Structured Query Language for the Web. *9ème World Wide Web Conference (WWW'00)*. – Amsterdam, Pays-Bas, Mai 2000.

[Spertus97] Spertus (Ellen). – ParaSite : Mining Structural Information on the Web. *6ème World Wide Web Conference (WWW'97)*. – Santa Clara, Californie, États-Unis, Avril 1997.

[Stenback et al.01] Stenback (Johnny), Hors (Arnaud Le), Hégaret (Philippe Le), Wilson (Chris), Jacobs (Ian), Champion (Mike), Isaacs (Scott) et Apparao (Vidur). – *Document Object Model (DOM) Level 2 HTML Specification*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Décembre 2001.

[Tan et al.98] Tan (Chen-Hai), Lim (Ee-Peng), Ng (Wee-Keong) et Lim (Boon-Wan). – *Structured Information Retrieval for Web Documents*. – Rapport technique, Nayang Technological University, Singapour, Centre for Advanced Information Systems (CAIS), Janvier 1998.

[Thelwall01] Thelwall (Mike). – Extracting macroscopic information from Web links. *Journal of the American Society for Information Science*, vol. 52, 2001, pp. 1157–1168.

[Vandendorpe91a] Vandendorpe (Christian). – Contexte, compréhension et littérarité. *RS/SI*, vol. 11, 1991, pp. 9–25.

[Vandendorpe91b] Vandendorpe (Christian). – Lecture et quête de sens. *Protée*, vol. 19, 1991, pp. 95–101.

[vR79] van Rijsbergen (Cornelis Joost). – *Information Retrieval*. – Butterworths, London, Janvier 1979.

[vR86] van Rijsbergen (Cornelis Joost). – A new theoretical framework for Information Retrieval. *9ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'86)*, pp. 194–200. – Pise, Italie, Septembre 1986.

[Weiss et al.96] Weiss (Ron), Vélez (Bienvenido), Sheldon (Mark A.), Namprempre (Chanathip), Szilagyi (Peter), Duda (Andrzej) et Gifford (David K.). – HyPursuit : a hierarchical network search engine that exploits content-link hypertext clustering. *7ème ACM Conference on Hypertext (HT'96)*, pp. 180–193. – Washington, DC, États-Unis, Mars 1996.

[White et al.89] White (H.D.) et McCain (K.W.). – Bibliometrics. *Annual Review of Information Science Technology*, vol. 24, 1989, pp. 119–165.

[Whitmer02] Whitmer (Ray). – *Document Object Model (DOM) Level 3 XPath Specification*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Février 2002.

[Wilkinson94] Wilkinson (Ross). – Effective Retrieval of Structured Documents. *17ème ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*, pp. 311–317. – Dublin, Irlande, Juillet 1994.

[Wood et al.98] Wood (Lauren) et et al. – *Document Object Model (DOM) Level 1 Specification*. – Rapport technique, MIT, INRIA, Keio, W3C : World Wide Web Consortium, Octobre 1998.

[Woodruff et al.96] Woodruff (Allison), Aoki (Paul M.), Brewer (Eric), Gauthier (Paul) et Rowe (Lawrence A.). – An Investigation of Documents from the WWW. *5ème World Wide Web Conference (WWW'96)*. – Paris, France, Mai 1996.

Chapitre 12

Glossaire

12.1 Paramètres (document, système ou utilisateur)

- $\alpha_{cotexte}$ **:** importance du cotexte d'un chemin, c'est-à-dire le document structuré sur lequel il est défini (paramètre du système, indexation).
- α_{mem} **:** importance de la mémoire de lecture (paramètre du système, indexation).
- α_{mi} **:** importance de la méta-information (paramètre du système, interrogation).
- $\beta_{\textbf{\textit{coût}}}$ **:** coût de navigation entre deux documents (relation de référence).
- β_i **:** rupture sémantique entre deux documents (relation de cheminement).
- μ_{gran} **:** granularité demandée dans la requête (paramètre utilisateur).
- μ_{nav} **:** focus de la requête (paramètre utilisateur).
- $seuil_{comp}$ **:** seuil de pondération lors de la composition (paramètre du système, indexation).
- $seuil_{chem}$ **:** seuil de pondération lors de l'indexation des chemins de lecture (paramètre du système, indexation).
- $seuil_{aut}$ **:** seuil de pondération lors de la mise en contexte (paramètre du système, indexation).
- $seuil_{ray}$ **:** seuil de pondération lors de la mise en contexte (paramètre du système, indexation).

12.2 Fonctions

- sim_{gran} **:** similarité de granularité.
- sim_{vec} **:** similarité vectorielle.

Sixième partie

Annexes

Annexe A

Fonctions de pondérations

SMART propose un grand nombre de fonctions de pondération (150 combinaisons possibles), dont le résultat est une valeur exprimant l'**importance** ($weight$) d'un terme par rapport à un document, en fonction de :

N = le nombre total de documents dans le corpus.

n = le nombre de termes distincts dans le corpus.

n_{loc} = le nombre d'occurrences du terme dans le document.

tf_{max} = le nombre maximum d'occurrences d'un terme dans le document.

n_{doc} = le nombre de documents dans lesquels le terme apparaît.

Une telle fonction est désignée par un code de trois lettres (par exemple "nnn"), qui signifient :

1. Le premier caractère spécifie la procédure utilisée pour calculer la composante "**term frequency (tf)**" de la pondération, par rapport au document :

   none : $tf = n_{loc}$

   binary : $tf = 1$

   max-norm : $tf = \frac{n_{loc}}{tf_{max}}$

   aug-norm : $tf = \frac{1+\frac{n_{loc}}{tf_{max}}}{2}$

   square : $tf = n_{loc}^2$

   log : $tf = log_2(n_{loc}) + 1$

2. Le second caractère spécifie la procédure utilisée pour prendre en compte la composante "**inverted document (idf)**" de la pondération, c'est-à-dire normaliser le tf par rapport au reste du corpus.

   none : $idf = 1$

   tfidf : $idf = log_2(\frac{N}{n_{doc}})$

prob : $idf = log_2(\frac{N-n_{doc}}{n_{doc}})$
freq : $idf = \frac{1}{n}$
squared : $idf = log_2^2(\frac{N}{n_{doc}})$

3. Le troisième caractère spécifie la procédure utilisée pour normaliser le tf :

none : $weight = tf * idf$
sum : $weight = \frac{tf*idf}{\sum_{i=1}^{n}(tf_i*idf_i)}$
cosine : $weight = \frac{tf*idf}{\sqrt{\sum_{i=1}^{n}(tf_i*idf_i)^2}}$
fourth : $weight = \frac{tf*idf}{\sum_{i=1}^{n}(tf_i*idf_i)^4}$
max : $weight = \frac{tf*idf}{\sum_{i=1}^{n}(tf_i*idf_i)}$

Le tf exprime la représentativité du terme par rapport au document, et doit être d'autant plus fort (faible) que le terme exprime mieux (moins bien) le contenu du document. Le tf peut être normalisé, par exemple pour ne pas avantager outrageusement les documents volumineux.

L'idf exprime la discriminance du terme par rapport au document, et doit être d'autant plus fort (faible) que le terme permet mieux (moins bien) discriminer le document parmi les autres documents du corpus.

Enfin, une normalisation est appliquée au produit $tf * idf$, par rapport au reste du vecteur. Généralement, l'objectif est que la somme des composantes (ou des composantes élevées au carré) du vecteur soit égale à 1.

Annexe B

Collection OFIL d’Amaryllis

B.1 Requêtes

Voici les deux premières requêtes de la collection OFIL d’Amaryllis :

```
<record>
<num>1</num>
<dom>International</dom>
<suj>La séparation de la Tchécoslovaquie</suj>
<que>Pourquoi et comment avoir divisé la Tchécoslovaquie,
et quelles ont été les répercussions économiques et
sociales ?</que>
<cinf>Prendre en compte les différentes versions présentées</cinf>
<ccept>
<c>Partition de la Tchécoslovaquie</c>
<c>Causes et modalités de la partition</c>
<c>Création de la Slovaquie et de la République Tchèque</c>
<c>Points de vue</c>
<c>Economie</c>
</ccept>
</record>
<record>
<num>2</num>
<dom>International</dom>
<suj>Le conflit yougoslave</suj>
<que>Comment ont été traités les civils pendant le conflit ?</que>
<cinf>Les documents pertinents devront préciser les conditions de
vie et les sévices subis par les populations civiles, de même que
l’aide qui leur a été apportée</cinf>
<ccept>
<c>Guerre en ex-Yougoslavie</c>
```

```
<c>Conditions des civils</c>
<c>Viols systématiques en Bosnie</c>
<c>Serbes, Musulmans, Croates</c>
<c>Victimes</c>
</ccept>
</record>
```

B.2 Jugements de pertinence

Voici les jugements de pertinences associés aux deux requêtes :

```
<record>
  <qid>1</qid>
<rep><docno>2276407</docno></rep>
<rep><docno>2271490</docno></rep>
<rep><docno>2271537</docno></rep>
<rep><docno>2273519</docno></rep>
<rep><docno>2276407</docno></rep>
<rep><docno>2271491</docno></rep>
<rep><docno>2274996</docno></rep>
<rep><docno>2271492</docno></rep>
<rep><docno>2271493</docno></rep>
</record>
<record>
  <qid>2</qid>
<rep><docno>2274388</docno></rep>
<rep><docno>2271825</docno></rep>
<rep><docno>2274238</docno></rep>
<rep><docno>2275740</docno></rep>
<rep><docno>2271538</docno></rep>
<rep><docno>2271822</docno></rep>
<rep><docno>2272958</docno></rep>
<rep><docno>2273062</docno></rep>
<rep><docno>2273176</docno></rep>
<rep><docno>2275487</docno></rep>
<rep><docno>2275928</docno></rep>
<rep><docno>2276194</docno></rep>
</record>
```

B.3 Documents

Voici le premier document de la collection :

<TEI.2>

<text>

<body>

<title>Les plumes de l’ange : Pour un texte écrit par Pasolini en harmoniques avec son film " Théorème ", Baudoin invente de belles assonances dessinées</title>

" Théorème a été créé comme sur un fond or : je le peignais de la main droite tandis que, de la gauche, je travaillais à une fresque sur une grande paroi (le film homonyme) ", écrivait Pier Palo Pasolini en présentation du livre qui, comme il vient de le dire, a été conçu en même temps que le célèbre film avec Terence Stamp, Silvana Mangano, Laura Betti, Anne Wiazemsky et Massimo Girotti, et publié (en Italie) en 1968, avant même la sortie en salle. C’est cet ouvrage " littéraire " (paru en France dix ans plus tard, déjà chez Gallimard), mais dont l’auteur indique la nature composite en se référant à la peinture (et en particulier la peinture religieuse), qui reparaît dans la singulière collection Futuropolis/Gallimard, consacrée à l’édition de grands textes accompagnés de dessins par des auteurs de BD (dont trois mémorables Céline-Tardi). Baudoin, le dessinateur invité dans la maison de Théorème, avait déjà réussi, pour la même collection, l’improbable exploit de faire danser ses images d’encre noire et de mystère silencieux autour du Procès-verbal de Le Clezio. Intervention fort éloignée de ce qu’on entend d’ordinaire par " illustration ", entretenant avec le texte une relation plutôt comparable à ce que devrait être celle qui unit musique de film et images : ni description, ni commentaire, ni surenchère, mais des harmonies et des contrepoints qui ouvrent un espace nouveau, et de nature différente. De ce texte, qui tourne autour de l’irruption dans une famille bourgeoise milanaise d’un étranger luciférien, Pasolini écrivait : " Notre propos consiste moins en un récit qu’en ce qu’on pourrait appeler, en langage scientifique, un " relevé ". L’écrivain y mêle morceaux de chroniques, analyses, poèmes, extraits de journal intime, descriptions romanesques. Et les interventions de Baudoin retrouvent cette mobilité, images composites où se mêlent photos, lambeaux de film, croquis, planches de BD avec ou sans dialogues, ébauches suggestives et dessins achevés. Plus naturellement encore, Pasolini lui-même débarque dans ces dessins, fraternel et distant, inquiétant et séduisant comme l’Hôte dans la demeure milanaise. Coup de force ou coquetterie de l’ " illustrateur " ? Non, tant paraît nécessaire la présence de l’observateur dans le compte-rendu de l’expérience scientifique dont, comme on sait, il modifie le résultat. Le résultat est explosif, et bizarrement tendre. La violence des noirs et blancs, la folie des paysages-visages, l’étrangeté des répétitions et des glissements, l’ironie et la sensualité des recadrages et des indications graphiques à l’intérieur des dessins riment avec les mots de PPP. Dans les interstices de ces jeux de miroirs, effectivement, un ange passe.

</body>

</text>

</TEI.2>

B.4 Documents fragmentés

2271448.1

Les plumes de l ange Pour un texte écrit par Pasolini en harmoniques avec son film Théorème Baudoin invente de belles assonances dessinées

2271448.2

Théorème a été créé comme sur un fond or je le peignais de la main droite tandis que de la gauche je travaillais à une fresque sur une grande paroi le film homonyme écrivait Pier Palo Pasolini en présentation du livre qui comme il vient de le dire a été conçu en même temps que le célèbre film avec Terence Stamp Silvana Mangano Laura Betti Anne Wiazemsky et Massimo Girotti et publié

2271448.3

en Italie en 1968 avant même la sortie en salle C est cet ouvrage littéraire paru en France dix ans plus tard déjà chez Gallimard mais dont l auteur indique la nature composite en se référant à la peinture et en particulier la peinture religieuse qui reparaît dans la singulière collection Futuropolis Gallimard consacrée à l édition de grands textes accompagnés de dessins par des auteurs de BD

2271448.4

dont trois mémorables Céline Tardi Baudoin le dessinateur invité dans la maison de Théorème avait déjà réussi pour la même collection l improbable exploit de faire danser ses images d encre noire et de mystère silencieux autour du Procès verbal de Le Clezio Intervention fort éloignée de ce qu on entend d ordinaire par illustration entretenant avec le texte une relation plutôt comparable à ce que devrait être celle qui unit musique de film et images ni description ni commentaire ni surenchère mais des harmonies et des contrepoints qui ouvrent un espace nouveau et de nature différente

2271448.5

De ce texte qui tourne autour de l irruption dans une famille bourgeoise milanaise d un étranger luciférien Pasolini écrivait Notre propos consiste moins en un récit qu en ce qu on pourrait appeler en langage scientifique un relevé L écrivain y mêle morceaux de chroniques analyses poèmes extraits de journal intime descriptions romanesques Et les interventions de Baudoin retrouvent cette mobilité images composites où se mêlent photos lambeaux de film croquis planches de BD avec ou sans dialogues ébauches suggestives et dessins achevés

2271448.6

Plus naturellement encore Pasolini lui même débarque dans ces dessins fraternel et distant inquiétant et séduisant comme l Hôte dans la demeure milanaise Coup de force ou coquetterie de l illustrateur Non tant paraît nécessaire la présence de l observateur dans le compte rendu de l expérience scientifique dont comme on sait il modifie le résultat Le résultat est explosif et bizarrement tendre

2271448.7

La violence des noirs et blancs la folie des paysages visages l étrangeté des répétitions et des glissements l ironie et la sensualité des recadrages et des indications graphiques à l intérieur des dessins riment avec les mots de PPP Dans les interstices de ces jeux de miroirs effectivement un ange passe

Annexe C

Courbes de Rappel/Précision, collections OFIL

C.1 RI atomique

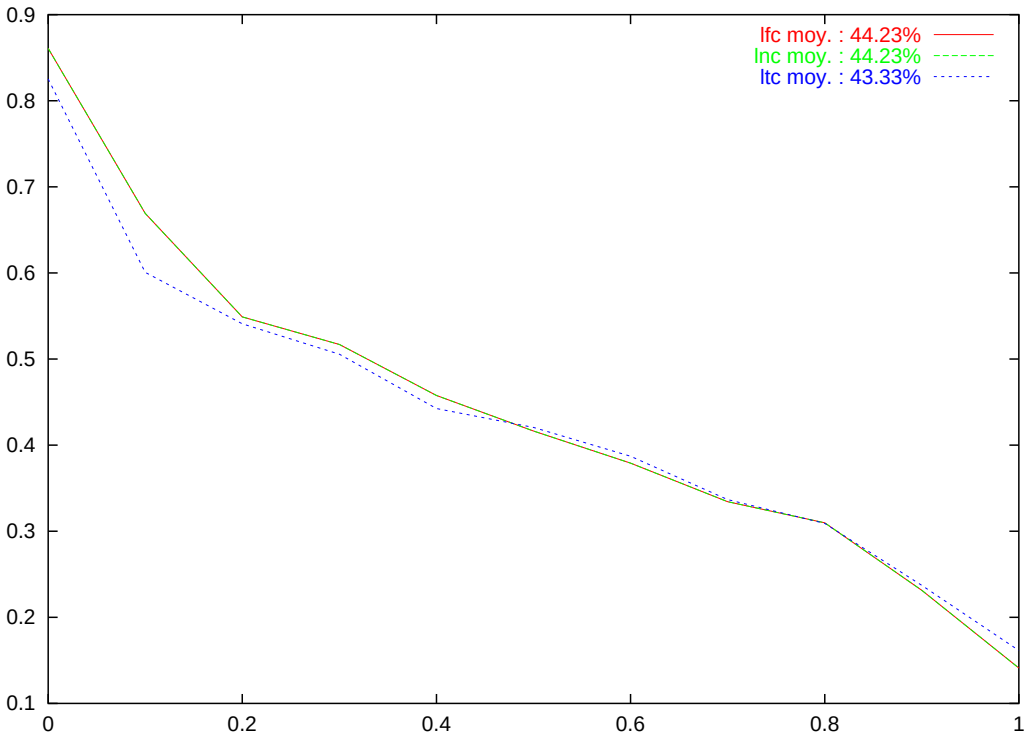

FIG. C.1 – RI atomique : courbe de référence (collection $OFIL^{req}_{agreg}$).

C.2 Pondération dfs_{ds}

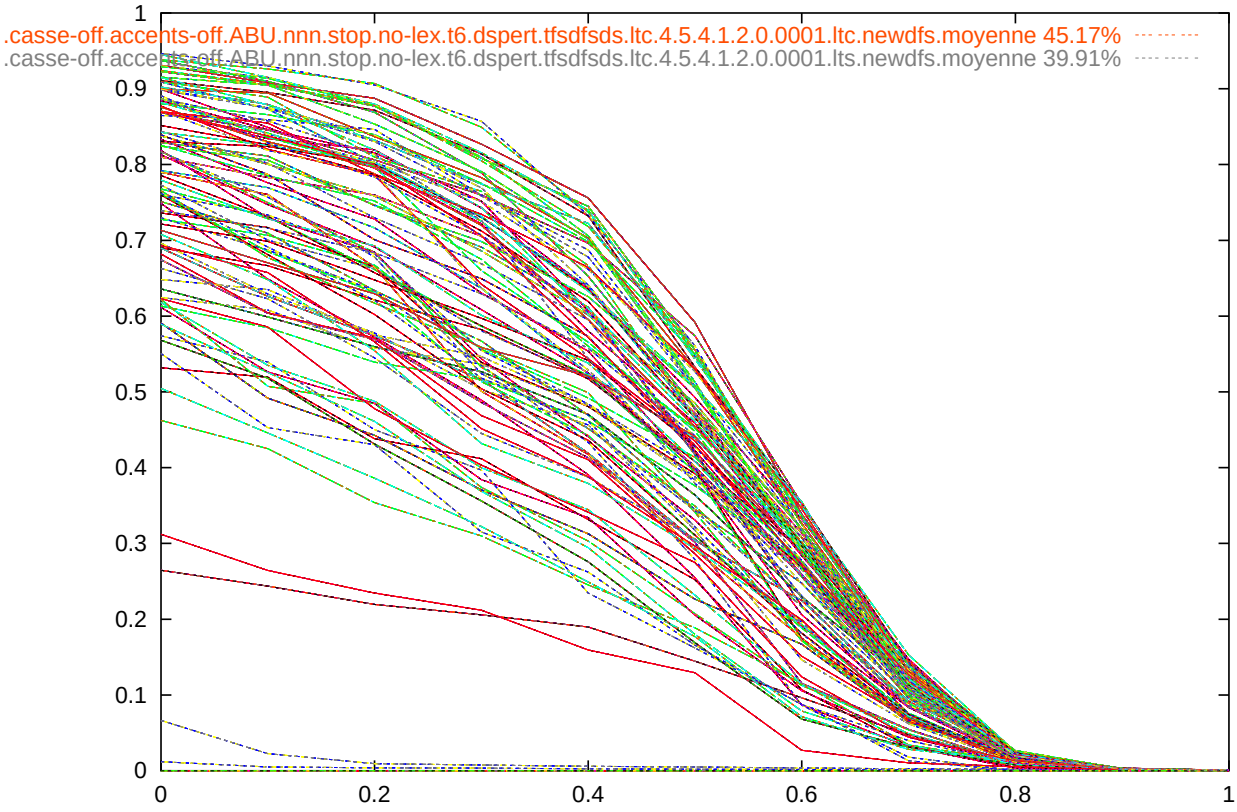

FIG. C.2 – Indexation documents structurés : pondération dfs_{ds}, collection $OFIL^{req}_{agreg}$.

Annexe D

SmartWeb

La figure suivante montre l'interface d'interrogation du prototype SmartWeb. On peut voir la réglette permettant de choisir une recherche focalisée ("*page HTML*") ou défocalisée ("*zone de pertinence*").

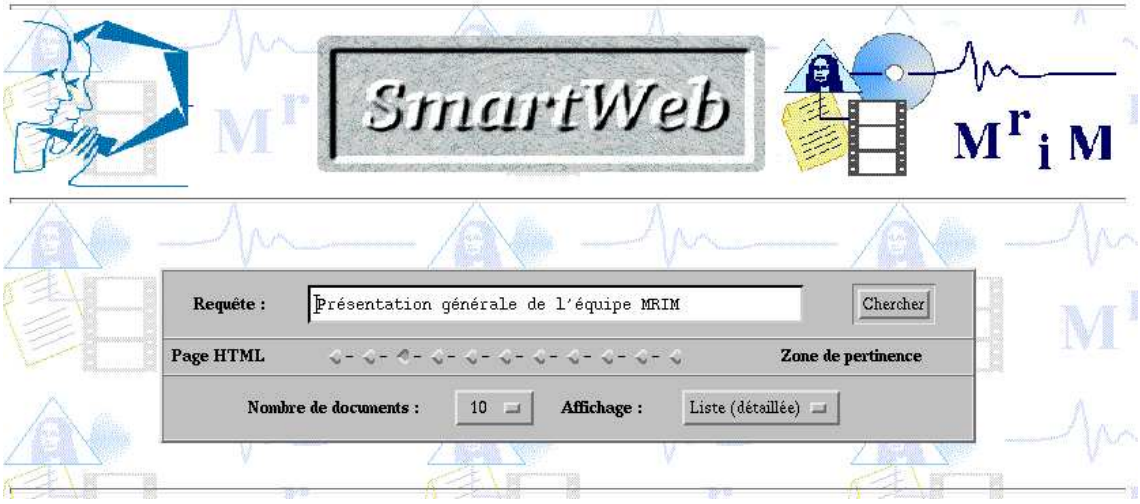

FIG. D.1 – Interface d'interrogation du prototype SmartWeb.

Annexe E

SRIS

E.1 Collecte du Web

La figure suivante montre la fenêtre de paramétrage d'une collecte à l'aide du robot CLIPS-Index.

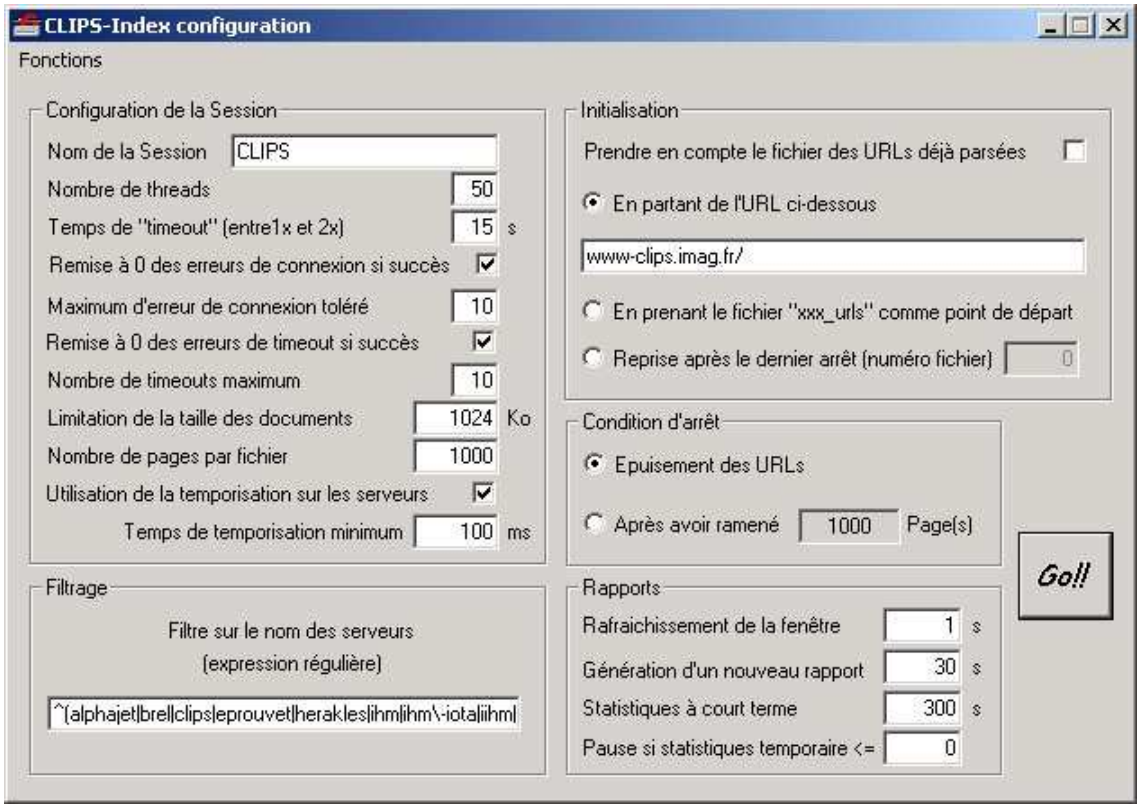

FIG. E.1 – Interface de lancement du robot CLIPS-Index.

La figure suivante montre l'interface de visualisation de l'état d'une collecte en cours avec le robot CLIPS-Index.

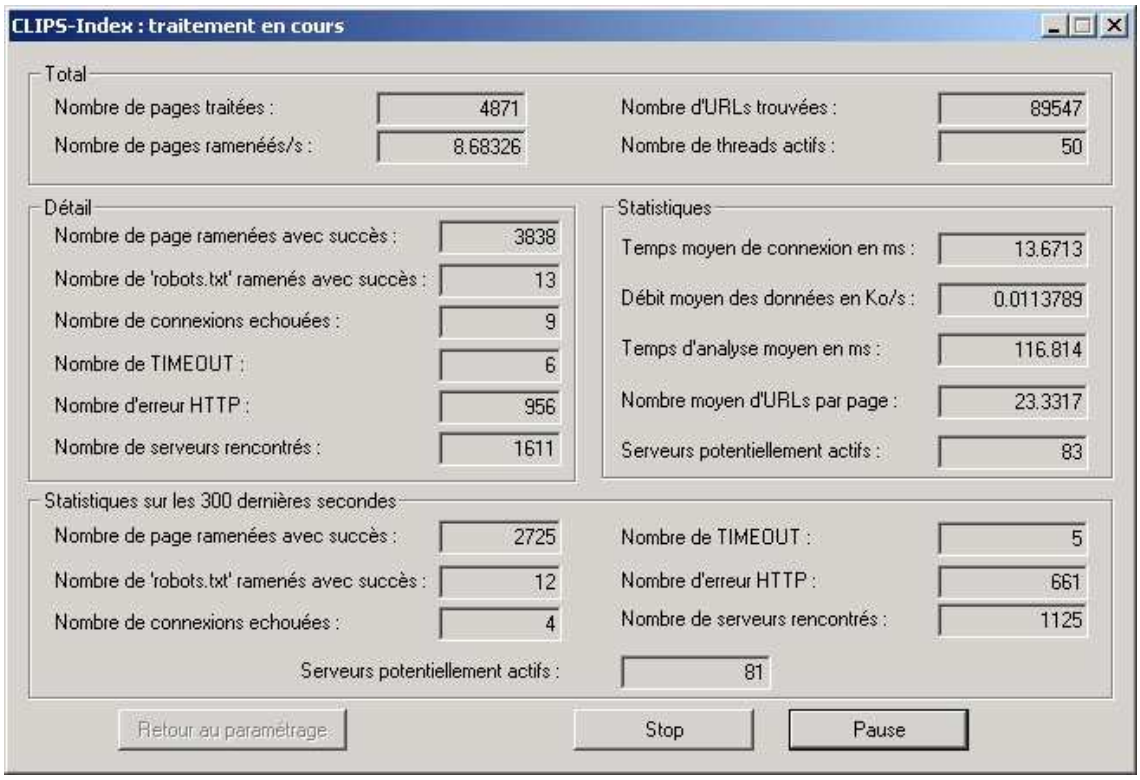

FIG. E.2 – Interface d'affichage du robot CLIPS-Index.

E.2 Visualiser les collections

La figure suivante montre l'interface du SRIS permettant d'accéder aux collections, d'examiner les documents, les liens, leurs indexations, etc., et enfin d'interroger le SRIS.

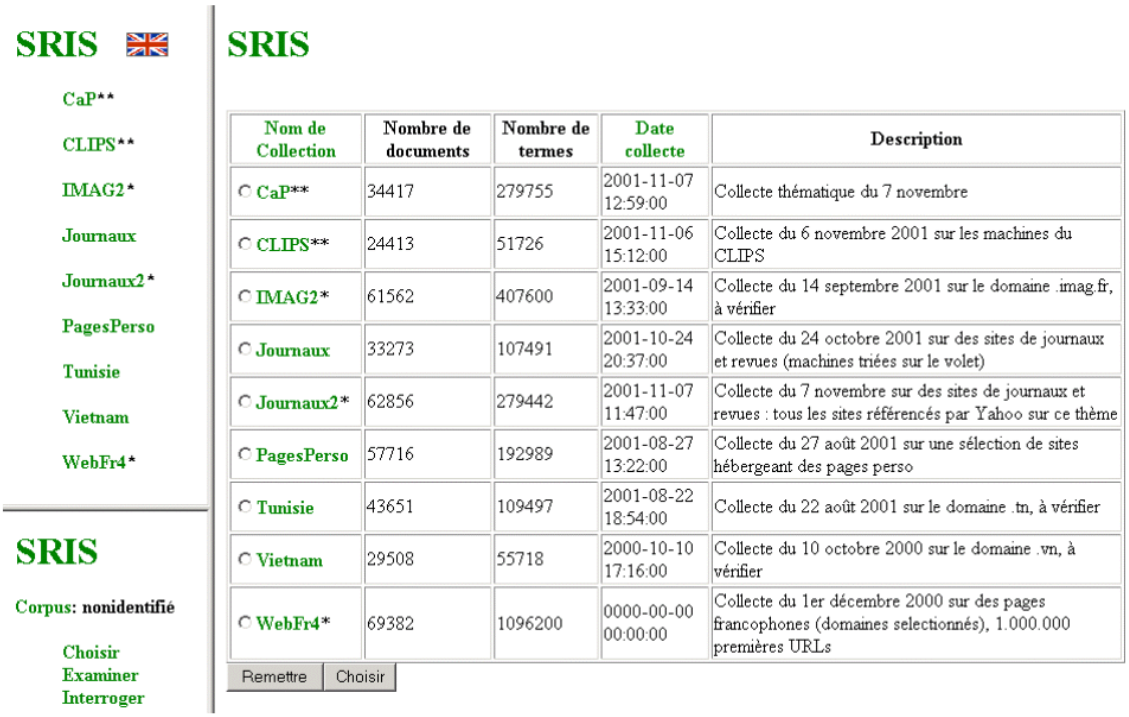

FIG. E.3 – Interface d'accès aux collection indexées.

La figure suivante montre l'interface de consultation des documents d'une collection donnée (ici, la collection IMAG).

FIG. E.4 – Examiner une collection.

La figure suivante montre l'interface de consultation des liens entrants et sortants d'une page. Cette interface permet également de naviguer dans le graphe des documents indexés, en suivant les liens entrants ou sortants.

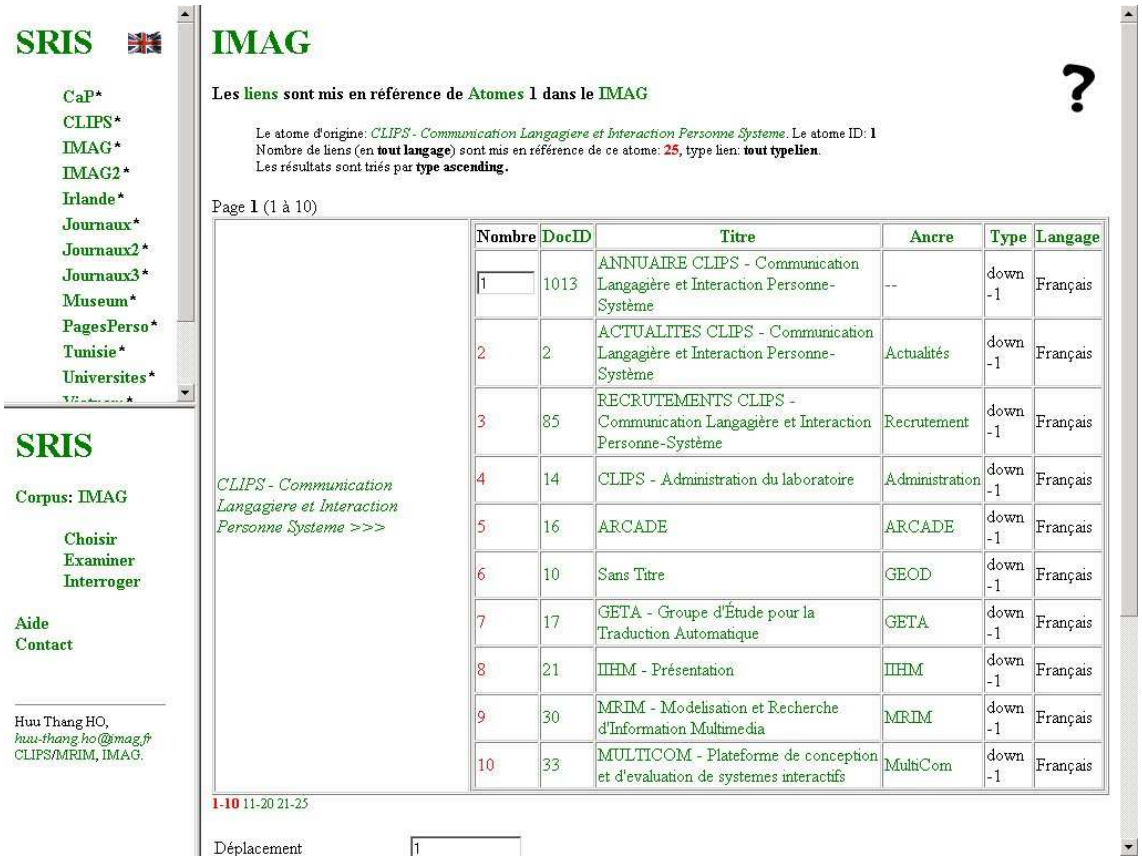

FIG. E.5 – Examiner le réseau de liens.

E.3 Validation du typage de liens

La figure suivante montre l'interface de validation du typage de liens, décomposée en trois "cadres". Le bandeau de gauche montre toutes les pages à évaluer. Le bandeau supérieur permet de donner un jugement utilisateur sur les types des liens d'une page. Enfin, le cadre central permet de naviguer à partir des documents étudiés, afin d'établir son jugement.

FIG. E.6 – Interface d'évaluation du typage de liens.