HAL
open science

Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo

Brigitte Simonnot

To cite this version:

Brigitte Simonnot. Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo. Informatique [cs]. Université Henri Poincaré - Nancy 1, 1996. Français. NNT : 1996NAN10017 . tel-01748617

HAL Id: tel-01748617

https://hal.univ-lorraine.fr/tel-01748617

Submitted on 20 Jun 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE LORRAINE

AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : ddoc-theses-contact@univ-lorraine.fr

LIENS

Code de la Propriété Intellectuelle. articles L 122. 4
Code de la Propriété Intellectuelle. articles L 335.2- L 335.10
http://www.cfcopies.com/V2/leg/leg_droi.php
http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

Département de Formation Doctorale en Informatique

École Doctorale IAE + M

Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo.

THÈSE

présentée et soutenue publiquement le 16 Janvier 1996

pour l'obtention du

Doctorat de l'Université Henri Poincaré – Nancy I

(Spécialité Informatique)

par

Brigitte Simonnot

Composition du jury

Président :	Marie-Christine Haton
Rapporteurs :	Jean-Paul Barthès Yves Chiaramella Amedeo Napoli
Examinateurs :	Marion Créhange (Directrice de thèse) Rose Dieng

Centre de Recherche en Informatique de Nancy

Remerciements

Je tiens à remercier chaleureusement chacun des membres du jury et en particulier :

Jean-Paul Barthès qui a bien voulu rapporter ce travail de thèse;

Yves Chiaramella pour le temps consacré à l'examen de ce travail, et pour ses remarques très pertinentes qui ont aidé à améliorer ce mémoire;

Marie-Christine Haton qui a accepté avec gentillesse de présider le jury et d'y représenter l'Université Henri Poincaré;

Amedeo Napoli pour sa relecture minutieuse et ses remarques enrichissantes sur mon travail;

Marion Créhange, qui a accepté de diriger cette thèse dans des conditions administratives peu faciles; ton amitié et ton soutien ont été très importants pour l'aboutissement de cette thèse;

Rose Dieng qui, malgré un agenda bien chargé, a bien voulu examiner ce travail.

Ce travail de thèse s'est déroulé au sein de l'équipe EXPRIM dont je remercie tous les membres pour l'ambiance sympathique qui leur est due.

Un remerciement tout particulier à Malika Smaïl : nous n'avons pas seulement partagé le même bureau pendant ces années, mais aussi des discussions souvent passionnées et enrichissantes; tes relectures patientes m'ont aidée à corriger ce mémoire.

Il m'est impossible de citer toutes les personnes du laboratoire que j'ai eu plaisir à cotoyer pendant ces années. Que chacun trouve ici l'expression de mon amitié. Un merci tout particulier à Sylvain pour ta disponibilité et ton aide pour résoudre certains problèmes dans l'utilisation de LaTeX, et Denis pour le style TheseCrin qui a facilité la mise en page de ce mémoire.

Je remercie enfin le CCETT, et en particulier Alain Léger, dont les contrats d'étude ont permis indirectement de financer une partie de ce travail.

À Dominique, à Pierre,
à mes proches ...

Table des matières

1 Problématique de la Recherche d'Information Multimédia **5**

1.1 Problématique générale de la Recherche d'Information 5

1.2 Particularités liées aux documents multimédias 7

1.2.1 Les données textuelles 8

1.2.2 Les images fixes 8

1.2.3 La vidéo 10

1.2.4 Choix des niveaux de granularité pertinents pour la RI vidéo 12

1.3 Les connaissances et leur représentation dans les SRI 14

1.3.1 Connaissances sur les documents 15

1.3.2 Connaissances sur les concepts du domaine 16

1.3.3 Connaissances sur l'utilisateur 21

1.3.4 Le besoin d'information de l'utilisateur 21

1.4 Évaluation en Recherche d'Information 23

1.4.1 La pertinence 23

1.4.2 Évaluation des résultats d'un SRI 24

1.5 Bilan 26

2 Modèles de Recherche d'Information **27**

2.1 Les premiers modèles de mise en correspondance pour l'accès par requête . . . 27

2.1.1 Le modèle booléen 28

2.1.2 Le modèle vectoriel 29

2.1.3 Les modèles booléens étendus 36

2.1.4 Approches alternatives 39

2.1.5 Thésaurus pour la recherche d'information 43

2.2 Le modèle logique 47

2.2.1 Principes du modèle logique 48

2.2.2 *À-propos* et principes d'incertitude 49

2.2.3 Comparaison du modèle logique avec les précédents modèles présentés 51

2.2.4 Comparaison inductive de modèles de RI 53

2.2.5 Bilan sur le modèle logique 57

2.3 Le bouclage de pertinence 58

2.3.1 Bouclage de pertinence dans le modèle vectoriel 58

2.3.2 Bouclage de pertinence dans les modèles probabilistes 61

2.3.3 Remarques sur le bouclage de pertinence 62

2.4 Conclusion 63

3 Propositions pour la structuration des connaissances sur les documents 65

3.1 Facettes de documents 65

3.1.1 Catégories de facettes 66

3.1.2 Indépendance des facettes 67

3.1.3 Choix d'un formalisme de représentation : les graphes conceptuels . . . 68

3.2 Présentation de la théorie des graphes conceptuels 69

3.2.1 Les concepts et les relations conceptuelles 69

3.2.2 Les graphes canoniques 71

3.2.3 Règles de formation de graphes 72
3.2.4 L'opérateur de projection de graphes 74
3.2.5 Ordre sur les graphes conceptuels 75
3.2.6 Généralisation/spécialisation commune 76
3.2.7 Utilisation de l'ordre sur les graphes pour évaluer la pertinence 77
3.2.8 Contraction de graphes 78
3.2.9 Négation et graphes conceptuels 80
3.2.10 Notion de contexte et graphes conceptuels 81
3.2.11 Méta-représentation des graphes conceptuels 81
3.3 Modélisation des connaissances d'un SRI sous forme de graphes conceptuels . 82
3.3.1 Les documents 82
3.3.2 Les descripteurs 83
3.3.3 Les concepts relationnels 84
3.3.4 Définition des descriptions de documents 84
3.3.5 Les attributs 86
3.3.6 Les facettes 88
3.3.7 Le thésaurus 88
3.4 Travaux liés 91
3.4.1 Autres définitions de facettes 91
3.4.2 RIME et Hyper-RIME 91
3.4.3 ELEN, recherche de composants en génie logiciel 92
3.5 Conclusion 96

4 Primitives génériques d'un processus de recherche d'information 97

4.1 Quelques modèles hybrides de processus de recherche d'information 97

4.1.1 Processus exploitant le modèle vectoriel : SMART et SIRE 97

4.1.2 Recherche interactive d'images : EXPRIM et RIVAGE 99

4.2 Le modèle paramétré de processus de recherche d'information 100

4.3 Le modèle paramétré revisité 104

4.3.1 Impact du traitement par facette sur les étapes de la résolution 104

4.3.2 Formulation de la requête 105

4.3.3 Interprétation de la requête 106

4.3.4 Formalisation de la requête 108

4.3.5 Proaction : Évaluation a priori 112

4.3.6 Déformation de la requête 113

4.3.7 Mise en correspondance requête-document 113

4.3.8 Synthèse des résultats 121

4.3.9 Préparation de la visualisation des résultats 122

4.3.10 Visualisation des résultats et choix 123

4.3.11 Reformulation 123

4.3.12 Reformulation en présence de descriptions par graphes conceptuels . . 127

4.3.13 Bilan 131

4.4 Les documents structurés et la vidéo 133

4.4.1 Spécificités de la recherche d'information dans des documents structurés 133

4.4.2 Structuration des documents vidéo 134

4.4.3 Attribution de types aux éléments de la structure du document 135

4.4.4 Structuration par sujet principal/secondaire 136

4.4.5 Conséquences sur le langage de requête 137

4.4.6 Nouveau type de besoin d'information 139

4.4.7 Niveaux de structure traités dans les différentes phases du processus . . 139

4.5 Résumé 146

5 Les agents d'un Système de Recherche d'Information 147

5.1 Les Agents : principaux concepts 148

5.1.1 Qu'est ce qu'un agent ? 148

5.1.2 Propriétés d'un agent 150

5.2 Systèmes multi-agents 153

5.2.1 Notion de système 153

5.2.2 L'organisation du système 154

5.3 La communication 154

5.3.1 Moyens de communication 154

5.3.2 La coopération 159

5.3.3 Les différents types de contrôle possibles 162

5.4 Les buts du système de recherche d'information 165

5.5 Les principaux agents qui gèrent les connaissances dans un SRI 168

5.5.1 L'application de recherche d'information 168

5.5.2 Le thésaurus 168

5.5.3 La base descriptive 169

5.5.4 La base de connaissances sur les individus et les événements 169

5.5.5 Le serveur de documents 170

5.5.6 L'utilisateur 170

5.5.7 L'interface utilisateur 171

5.6 Les principaux agents qui détiennent le savoir-faire du SRI 171

5.6.1 Les agents d'indexation 171

5.6.2 Les agents statistiques 172

5.6.3 Les agents de recherche d'information 173

5.6.4 Les outils de visualisation des documents 175

5.7 Stratégie et organisation dynamique des agents 175

5.7.1 Introduction 175

5.7.2 Variations et dynamicité de la stratégie 176

5.7.3 Auto-conception d'organisation 177

5.7.4 Composition/décomposition en agents-facettes 177

5.7.5 La connaissance organisationnelle 178

5.8 Relations de dépendance entre les agents du SRI 179

5.8.1 Consultation de la collection 180

5.8.2 Formulation de la requête 181

5.8.3 Interprétation sémantique de la requête 181

5.8.4 Proaction 183

5.8.5 Mise en correspondance entre requête et documents 187

5.8.6 Synthèse des résultats : élection des documents pertinents 188

5.8.7 Préparation de la visualisation 190

5.8.8 Visualisation 190

5.8.9 Analyse des choix et reformulation 190

5.9 Les conflits 191

5.9.1 Utilité des conflits 192

5.9.2 Types de conflits au sein d'un SRI 192

5.10 Caractérisation de la stratégie de recherche 193

5.11 Conduite du processus de recherche 196

5.11.1 Les cycles du processus de recherche 196

5.11.2 Critères sociaux de choix d'un agent pour la résolution 198

5.11.3 Architecture globale du système 199

5.12 Alternatives possibles pour l'organisation des agents 201

5.12.1 Interactions plus fortes entre les agents de mise en correspondance . . . 201

5.12.2 Granularité des agents 201

5.13 Résumé 201

6 Le modèle de système multi-agent retenu et sa mise en œuvre 205

6.1 Typologie des agents du SRI 205

6.1.1 Les agents de recherche de documents 205

6.1.2 Les agents de reformulation de la requête 207

6.1.3 Les agents de service 207

6.1.4 Les agents de coordination de l'organisation 208

6.2 Scénario général d'une session de recherche d'information 208

6.3 Critères de décision pour la composition/décomposition des agents 211

6.4 Évaluation des résultats d'un agent 212

6.4.1 Auto-Évaluation des agents de mise en correspondance 212

6.4.2 Auto-évaluation des agents de bouclage de pertinence 215

6.5 Évaluation globale du système 217

6.5.1 Les mesures d'évaluation de la société des agents 217

6.5.2 Choix d'un mode de scrutin pour l'élection des documents pertinents . 217

6.5.3 Notion de complétude de la recherche 217

6.5.4 Conséquences de l'auto-évaluation sur le fonctionnement du système . 218

6.5.5 Évaluation de la progression d'une session de recherche 218

6.6 Le prototype implanté : SAFARI-VIDEO 219

6.6.1 Macro-architecture du système 219

6.6.2 Le Thésaurus 220

6.6.3 La Base Descriptive 220

6.6.4 Les descriptions de documents 221

6.6.5 Le modèle de description 222

6.6.6 Transmission de connaissances entre agents 223

6.6.7 L'interface 224

6.6.8 Le serveur de documents 225

6.7 Déroulement d'une session de recherche 225

6.7.1 Vue de la session de recherche par l'utilisateur 225

6.7.2 L'application de recherche de séquences d'actualités 227

6.7.3 Évaluation du prototype réalisé 228

6.8 Travaux liés : agents de RI 228

6.9 Vers un langage de communication entre agents de recherche 230

6.10 Résumé 231

7 Conclusion et Perspectives **233**

7.1 Pour conclure 233

7.2 Perspectives 235

7.2.1 Perspective d'évaluation d'autres critères du système 235

7.2.2 Apprentissage à long terme 236

7.2.3 Aide à l'indexation automatique 237

7.2.4 Amélioration de la gestion des descriptions par graphes conceptuels . . 237

7.2.5 Les croyances 238

7.2.6 Le raisonnement temporel 238

7.2.7 Langage de requête visuel 239

7.2.8 Amélioration de l'interface de consultation des documents vidéo 239

Table des figures **241**

Bibliographie **243**

Introduction

La Recherche d'Information (RI) consiste à permettre à un utilisateur de retrouver des documents susceptibles d'être pertinents vis-à-vis d'un besoin d'information généralement formulé par une requête. Les applications de la RI se diversifient, tant du point de vue des médias de représentation des documents – ils ne sont plus seulement textuels, mais comprennent aussi images fixes, images animées et sons – que de celui des domaines d'application : recherche documentaire mais aussi génie logiciel, applications médicales, etc. Les modèles développés dans ce domaine deviennent, eux, de plus en plus nombreux et complexes.

Les utilisateurs des Systèmes de Recherche d'Information (SRI) ont également changé avec l'apparition des systèmes de recherche en ligne. Auparavant, les SRI étaient manipulés par des documentalistes, c'est à dire des personnes ayant reçu une formation à la recherche documentaire. À présent, la grande majorité des utilisateurs de ces systèmes n'ont aucune formation particulière à la recherche d'information. Pour ces personnes, la complexité des systèmes est un obstacle à leur utilisation.

Avec le développement des bibliothèques électroniques et du traitement automatisé des documents, le nombre de documents disponibles devient de plus en plus important. Les connaissances prises en compte pour améliorer l'accès à ces collections s'élargissent. Le développement des réseaux de communication laisse imaginer une vaste médiathèque électronique interconnectant différents systèmes de recherche, comme World Wide Web le préfigure.

C'est dans ce contexte que se situent nos réflexions. L'équipe EXPRIM, au sein de laquelle s'est déroulée cette thèse, s'est depuis ses débuts posé le problème de l'accès à des collections d'images fixes. Dans le travail que nous présentons, cette problématique est délibérément élargie aux images animées, et plus particulièrement à la vidéo. En effet, les applications utilisant ce type de média sont nombreuses : diffusion d'informations culturelles ou journalistiques, de résultats scientifiques, communication externe ou interne dans les entreprises, formation, enregistrement de données ethnologiques ou production de documents historiques, ou même télésurveillance. La construction des autoroutes de l'information voit le mariage des réseaux de communication traditionnels et de ceux de diffusion de la télévision, et une place de plus en plus importante est faite aux documents animés. Par ailleurs, la prise en compte de ce type de média permet de mettre en relief les caractéristiques de la RI propres à l'image fixe.

La première prise de contact avec ce nouveau média nous a surprise : si la technologie évolue rapidement autour de celui-ci, les documents vidéo sont la plupart du temps traités comme des points terminaux, comme s'il n'y avait plus rien à dire après avoir proposé un document vidéo à un utilisateur. Cette première impression nous a guidée vers une approche résolument interactive et analytique pour la recherche de documents représentés dans ce type de média.

Le problème qui nous occupe est donc de concevoir un SRI qui permette de rechercher des documents multimédias, et en particulier des vidéos. Face à un problème complexe, le réflexe

de l'informaticien est de le diviser en sous-problèmes plus simples. C'est ce que nous avons fait pour aborder notre problème. Nous avons donc commencé par distinguer les différents types de connaissances qui peuvent être enregistrées à propos des documents. Ceci nous a ammené à définir un modèle de description par facettes des documents, chaque facette pouvant être vue comme le point de vue d'un spécialiste sur ce document. En effet, les techniques mises en œuvre pour retrouver et présenter des informations multimédias font appel à des connaissances et des savoir-faire hétérogènes (celles du documentaliste, celles des spécialistes de chaque média impliqué, celles des spécialistes du domaine de l'application, etc.) dont la plupart évoluent rapidement : le traitement de la vidéo numérique en est un exemple.

Comment exploiter un modèle de document si complexe ? Notre approche a consisté à concevoir un ensemble d'agents qui se mette au service de l'utilisateur pour l'aider dans sa recherche d'information. La «société» d'agents ainsi constituée doit être capable d'évoluer. Définir les agents d'un tel système et leur organisation, étudier les types d'interactions entre les agents d'une part, et entre agents et utilisateur(s) du système d'autre part, en fonction des informations ou connaissances qu'ils peuvent échanger, tel a été l'objet principal de notre travail. Une approche distribuée permet d'affiner les différents aspects qui entrent en jeu dans une recherche d'information : elle oblige à expliciter la façon dont les différents éléments s'harmonisent entre eux ou s'opposent les uns aux autres.

Lorsque nous avons commencé cette étude, peu de travaux existaient sur la recherche d'information vidéo. Les chercheurs qui s'intéressaient à ce type de documents adoptaient en général une approche *reconnaissance automatique de formes*. Nous pensons que les techniques traditionnelles de recherche d'information, qui s'appuient sur des descriptions textuelles des documents, peuvent être utilisées avec profit pour les documents vidéo et sont complémentaires des approches de type *traitement automatique d'images*. Si nous n'avons pas pu, dans le cadre de cette thèse, aborder plus en détail les approches exploitant des techniques de reconnaissance automatique de formes, nous ne les avons pas ignorées pour autant : l'architecture distribuée du système a été conçue pour pouvoir les accueillir.

Nous allons d'abord présenter plus en détail la problématique et les principaux concepts de la Recherche d'Information (RI) multimédia (chapitre 1), puis les principaux modèles classiques ou plus avancés de RI (chapitre 2).

Nous présentons le formalisme de représentation que nous adoptons pour ces connaissances au chapitre 3.

Nous analysons ensuite un processus de recherche d'information interactive, et l'exploitation qui peut être faite des connaissances représentées (chapitre 4) lors des différentes phases de ce processus. Cette analyse met en évidence non seulement la diversité des connaissances mais aussi des savoir-faire qui sont mis en œuvre pour une RI. Nous adoptons donc une approche distribuée également pour leur analyse, et les passons au crible de la grille des concepts qui émergent des théories multi-agents. Ceci nous ammène à définir une architecture multi-agents pour notre système de recherche d'information (chapitre 5).

Nous présentons enfin le prototype implanté (chapitre 6) pour une application de recherche d'extraits d'actualités télévisées et faisons des propositions pour son évaluation avant de conclure et d'évoquer les prolongements possibles de ce travail.

Nous précisons que la distribution physique a constitué une hypothèse importante pour la définition de l'architecture du système, mais elle n'a pas été étudiée dans ses aspects techniques particuliers (accès réseaux, etc).

Chapitre 1

Problématique de la Recherche d'Information Multimédia

Nous présentons ici la problématique générale de la recherche d'information, en essayant d'abord de mettre en évidence la façon dont la nature des documents à traiter peut influencer la conception d'un modèle de recherche. Nous introduisons également les principaux concepts qui sont manipulés par un système de recherche d'information (SRI). Nous présentons enfin la problématique particulière de ce travail de thèse.

1.1 Problématique générale de la Recherche d'Information

La problématique de la recherche d'information (RI) consiste à donner aux utilisateurs accès à des documents tirés d'une collection, ces documents étant susceptibles de répondre à un besoin d'information. Contrairement à l'interrogation d'une base de données, où une requête permet de retrouver des faits stockés dans la base, une recherche d'information cherche à exploiter un certain nombre de connaissances (voir §1.3) pour déterminer si un document est ou non susceptible de contenir des informations pertinentes par rapport à ce besoin. L'utilisateur n'est pas obligé de connaître la façon dont les informations sont organisées dans le système pour formuler sa requête ; l'accès aux informations est donc plus flexible. Des techniques de recherche diverses qui correspondent au savoir-faire du système, ont été élaborées pour tenter d'améliorer les performances de la recherche, en qualité et en temps de réponse par rapport au volume de documents traité.

Les collections de documents interrogeables étaient à l'origine presque exclusivement des ensembles de références bibliographiques ; les textes eux-mêmes n'étaient pas directement accessibles. À présent, les SRI traitent des **documents qui sont en ligne** ; la recherche d'information s'adresse au contenu des documents proprement dit et est dirigée directement par les utilisateurs. Ceci a bouleversé les habitudes des chercheurs d'information : auparavant, ils s'adressaient à un bibliothécaire ou à un documentaliste qui servait d'intermédiaire de recherche. Un dialogue s'instaurait entre le documentaliste et la personne à la recherche d'une information

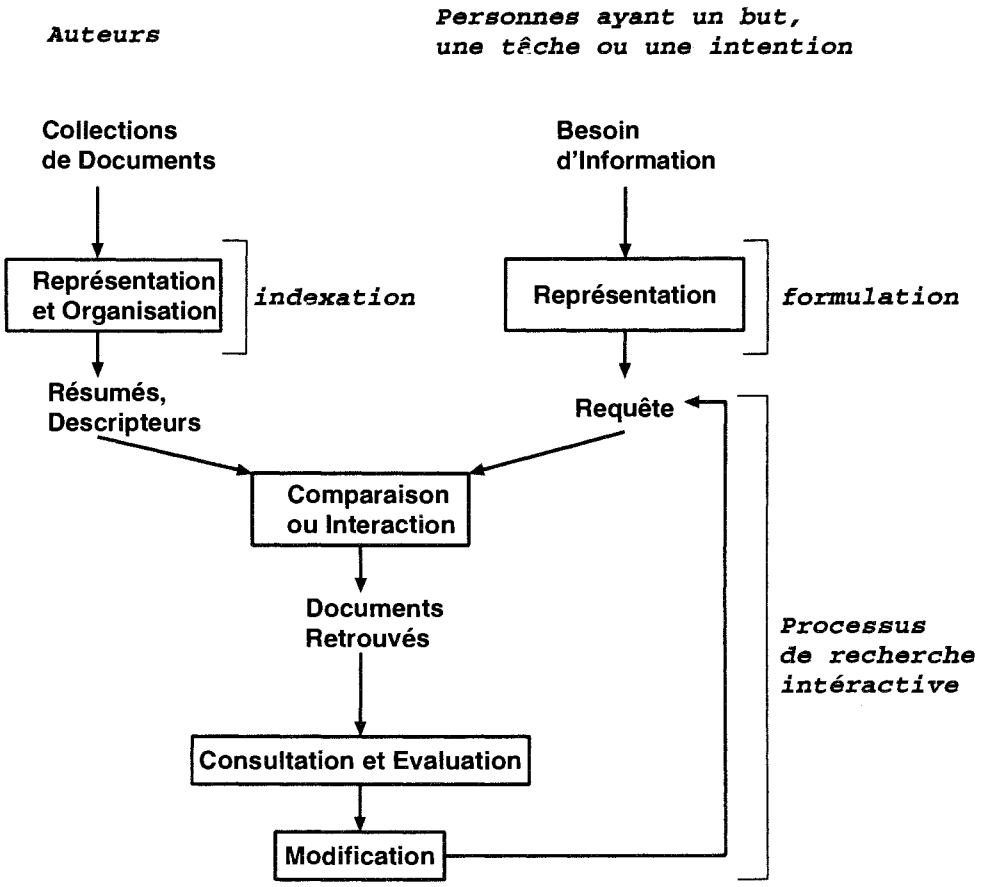


FIG. 1.1 – *Problématique générale de la recherche d'information interactive.*

pour aider à mieux cerner le besoin. L'utilisateur n'avait pas besoin de se préoccuper de la façon dont la recherche allait se dérouler : l'intermédiaire de recherche choisissait la démarche de recherche la plus appropriée au cas qu'il devait résoudre. Avec les systèmes de recherche en ligne, le rôle des intermédiaires de recherche a évolué [Hart 91]. Les utilisateurs principaux des SRI ne sont plus des documentalistes ayant reçu une formation spéciale, mais divers utilisateurs qui, le plus souvent, n'ont aucune connaissance en recherche documentaire. Le rôle du SRI en est complexifié : il doit suggérer à l'utilisateur une stratégie de recherche correspondant à la situation.

D'autre part, les SRI de la nouvelle génération traitent non seulement des documents plein texte, mais aussi multimédias. Par document **multimédia**, nous entendons un document comportant plusieurs médias : texte, son, images fixes ou images animées. Chaque média a ses propres caractéristiques qui ont des conséquences sur leur traitement par un SRI. De plus, il est possible d'exploiter la synergie produite par la co-existence de plusieurs médias au sein d'un même document ; par exemple, le texte d'une affiche peut contredire l'image qui y est représentée. Les systèmes hypermédias, qui permettent un accès associatif à l'information grâce à un simple clic de souris, sont un cas particulier de SRI. Dans ces systèmes, l'accès aux informations n'est pas conditionné par la formulation d'une requête, mais repose sur la navigation dans une collection, grâce à des liens en général pré-établis par l'auteur des documents hypermédias.

Cependant, ces systèmes intègrent de plus en plus les fonctionnalités d'un SRI, soit pour la génération automatique de liens hypermédias à partir d'une requête comme dans Microcosm [Davis 92, Li 92], soit pour permettre à l'utilisateur de formuler une requête dont le résultat sera le point de départ de la navigation dans des hyperdocuments [Kheirbek 95].

La **taille des collections** de documents auxquels les SRI permettent d'accéder subit également une inflation considérable, nos sociétés étant de plus en plus prolifiques en documents de toutes sortes. Bien plus que les textes, les images fixes et la vidéo représentent des volumes importants à stocker, même si les techniques de compression progressent considérablement. La possibilité d'accéder à une information précise dans ces collections volumineuses est une aide précieuse pour un chercheur d'information.

Ces évolutions dans l'utilisation des SRI interactifs provoquent de nombreux problèmes : la désorientation de l'usager, l'inefficacité de la navigation et la surcharge cognitive [1] en constituent quelques uns. Par exemple, pour une étude récente [Oliver 95], il a été demandé à un public d'étudiants de choisir le média dans lequel ils préféraient retrouver une information. Les choix ont été fortement influencés non pas par des critères objectifs vis-à-vis des besoins d'information, mais plutôt par le pouvoir d'attraction de certains médias, comme le son et la vidéo. Cependant, le succès de ces nouveaux médias a aussi un revers : quelle que soit la pertinence de l'information que contiennent de tels documents, le désintérêt gagne rapidement l'utilisateur qui a tendance à se laisser distraire. Les photographies ou les images fixes, au contraire, stimulent l'intérêt et l'attention ; elles entraînent une certaine activité chez l'utilisateur. Quant au texte, son usage en ligne est encore restreint à des extraits courts, les utilisateurs préférant en général imprimer un texte long pour le lire en différé à tête reposée.

Le domaine de la recherche d'information doit tenir compte de ces nouveaux critères. La diffusion massive des textes permise par l'invention de l'imprimerie avait fait évoluer de façon considérable la transmission des connaissances dans nos sociétés. Le développement des techniques de numérisation des images et des réseaux de communication vont jouer également un rôle considérable pour la diffusion des informations. Il reste cependant beaucoup à faire pour faciliter l'accès en ligne aux vastes quantités de documents produits chaque jour, et notamment pour éviter aux usagers des SRI le stress lié à une surcharge en information.

1.2 Particularités liées aux documents multimédias

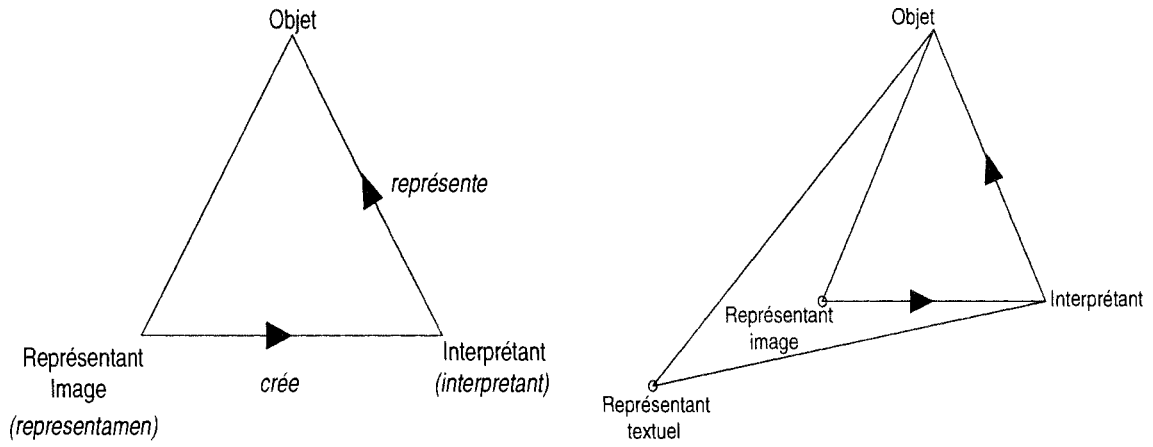
Le traitement des documents images par les SRI s'est d'abord beaucoup inspiré des approches adoptées pour les textes. C'est pourquoi nous présentons d'abord rapidement les caractéristiques du média « texte », pour mieux mettre en relief celles des autres médias.

1. Par surcharge cognitive, nous désignons le phénomène selon lequel l'utilisateur, confronté à un trop grand nombre d'informations, ne peut en assimiler aucune.

1.2.1 Les données textuelles

Les documents textuels ont longtemps été la seule « matière première » des SRI. Ils ont été d'abord représentés par des descriptions (textes courts ou listes de mots-clés) qui résumaient les connaissances que l'on pouvait exploiter (voir 1.3.1). Les premières approches de RI se sont appuyées sur des méthodes statistiques appliquées aux termes rencontrés dans le texte. Lors d'une phase d'indexation, les termes les plus significatifs du document sont extraits et réduits à leur racine. Des listes de « mots-vides » servent à exclure les mots jugés non significatifs, comme les articles et les prépositions. Ces techniques simples, si elles permettent un stockage économique des descriptions, introduisent aussi beaucoup d'ambiguïté. Cependant, de façon surprenante, elles sont assez efficaces, notamment lorsqu'elles sont couplées par exemple à l'utilisation d'un thésaurus [Croft 87]. Les approches linguistiques, elles, cherchent à prendre en compte les relations sémantiques entre les termes d'un document ou d'une requête. Leur but est souvent l'interrogation en langage naturel. Quelquefois elles sont utilisées pour constituer des résumés indicatifs qui sont présentés à l'utilisateur dans le but de lui permettre de mieux juger de la pertinence du document vis-à-vis du besoin formulé [Hahn 90, Paice 93].

1.2.2 Les images fixes



(a) La relation triadique de C.S. Pierce

(b) Relation lecture image/lecture texte

FIG. 1.2 – *Relations entre objet, représentant et interprétant lors de la lecture d'une image.*

La recherche d'images fixes est très différente de la recherche de documents textuels. En effet, le langage des images repose sur un ensemble de symboles plus équivoques que le langage textuel. Plus encore qu'un mot, un symbole iconique n'a pas une signification unique qui permettrait une interprétation univoque, mais des niveaux d'interprétations qui varient en fonction de la culture et de la sensibilité du sujet qui l'observe. La relation triadique de Charles Sanders Pierce, le père des *graphes existentiels*, représente bien cette distance entre l'objet même qui est représenté (Figure 1.2.a), son représentant (que Pierce nomme *representamen*) – par exemple

l'image et ses caractéristiques techniques (cadrage, lumière) – et l'interprétant, c'est à dire le code qui permet à un individu d'associer un représentant à un objet [Hudrisier 83, Sowa 95]. Ces codes reposent sur des connaissances de bon sens, souvent implicites, et sont propres à chaque individu en fonction de son but ou de ses intentions. Nous avons souvent tendance à oublier cette distance, et à confondre l'objet représenté et son représentant par une sorte de métonymie. Les peintres de ce siècle ont pris conscience de l'existence de ce code. Par exemple Picasso, dans sa période cubiste, respectait un certain nombre d'invariants quant aux caractéristiques physiques des personnages représentés ; ces invariants permettaient de les identifier même si leur portrait était éloigné de la réalité.

La distance entre l'objet et son représentant image peut être illustrée par cette anecdote relatée par Heinz Pagels dans *Les Rêves de la Raison* :
Pablo Picasso, alors qu'il voyageait en train, est interpellé par le voyageur assis en face de lui qui l'a reconnu. Ce dernier lui demande pourquoi il ne peint pas les femmes « comme elles sont réellement ». Picasso répond au voyageur qu'il ne comprend pas ce qu'il veut dire. L'homme sort alors une photographie de son portefeuille et la montre au peintre en lui disant :
« - Voici ma femme ! »
« - Elle est plutôt petite et plate ! », lui répond Picasso avec un sourire.

Pour des images, la phase d'indexation ne peut pas, comme pour le texte, exploiter des mots tirés directement du contenu ou du titre, sauf lorsqu'elle est accompagnée d'une légende. La description textuelle d'une image introduit un autre représentant dans la production du sens (Figure 1.2.b). Plusieurs œuvres du peintre surréaliste belge René Magritte illustrent parfaitement ces différents niveaux de production de sens (*« L'usage de la parole »*, 1927 ; *« La Trahison des images (Ceci n'est pas une pipe) » ; « Ceci n'est pas une pomme »* fig. 1.3).

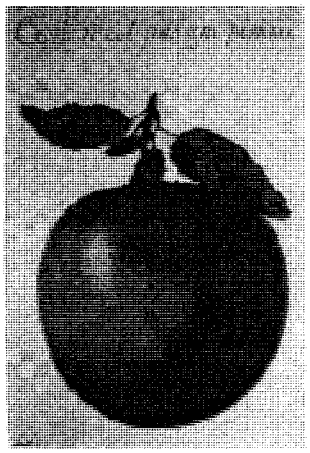


FIG. 1.3 – «Ceci n'est pas une pomme» *René Magritte, 1964.*

En outre, une description textuelle peut traiter deux niveaux de signification dans l'image. Le **dénoté** comporte le niveau pré-iconographique qui s'adresse à ce qui peut être objectivement vu dans l'image (objets, personnes, lieux, etc.) et le niveau iconographique, c'est-à-dire l'interprétation des faits, actions ou gestes selon des conventions sociales. Le **connoté** concerne

Ainsi, H. Hudrisier [Hudrisier 83] relate cette anecdote : Robert Doisneau, lors d'un reportage photographique sur Paris, rencontre à la terrasse d'un café un professeur d'architecture d'un âge respectable en train de deviser avec une jeune étudiante devant deux ballons de rouge. Après leur avoir demandé leur autorisation, il les prend en photo pour illustrer « *Les Bistrots de Paris* ». La photographie est vendue à une agence de presse. Quelque temps plus tard, elle paraît dans la presse pour illustrer une campagne anti-alcoolique. Alors que le procès intenté par le professeur n'est pas même terminé, elle paraît à nouveau dans un magazine à scandale avec la légende « *Prostitution aux Champs-Elysées* » ...

le message que l'auteur veut transmettre, le sens intrinsèque qui repose sur une interprétation de l'image à partir de connaissances et de sensibilité. La plupart des bases d'images comportent une description du dénoté. L'utilité de descriptions de la connotation est sujette à débat : l'interprétation ainsi enregistrée est souvent partielle et peut être erronée. Ainsi, certains arguent qu'il vaut mieux ne pas donner d'information du tout que des informations fausses. L'interprétation ou le sens donné à une image varie en effet énormément d'une personne à une autre. Cette interprétation est très sensible au contexte dans lequel se fait la visualisation. L'anecdote relatée par Henri Hudrisier (voir encadré) illustre bien la sensibilité de l'interprétation d'une image au contexte dans lequel elle est présentée.

Par ailleurs, une demande d'image émise sous la forme d'une requête textuelle traduit de façon très approximative le besoin d'information sous-jacent, c'est pourquoi des chercheurs se sont penchés sur le problème de donner accès aux images par le graphique (Query By Visual Example [Hirata 92, Lee 94a]). Toutefois, ce type de formulation de requête n'est approprié qu'à certaines applications particulières. Il est en effet difficile de demander à un journaliste qui recherche une photographie pour illustrer un article de dessiner l'image qu'il cherche. Pour la recherche de documents vidéo, demander à l'utilisateur de faire un croquis des objets qu'il cherche ou de leur mouvement semble peu réaliste. La description des images par des mots ou un texte court reste donc un moyen privilégié non seulement pour formuler une requête, mais aussi pour orienter l'interprétation qui sera faite des documents lors de leur visualisation.

Les images ont sur le texte plusieurs avantages qui peuvent être exploités par un SRI. Leur visualisation est en effet plus rapide, un coup d'oeil étant souvent suffisant à un utilisateur pour déterminer si l'image qui lui est présentée répond à son besoin. Les images peuvent également être présentées en damier ; leur juxtaposition, accélérant la rapidité de perception, permet au chercheur d'information une analyse comparative qui enrichit sa recherche.

1.2.3 La vidéo

Nous prenons le terme *vidéo* dans un sens élargi, pour désigner les documents constitués d'images animées : films, reportages, émissions télévisées, animations, etc. La vidéo est un média riche, lui-même multimédia puisqu'il associe à des images une bande sonore. De façon brute, une vidéo peut être vue comme une succession d'images fixes dont le défilement à une certaine vitesse restitue l'impression de mouvement, si nous faisons abstraction de la bande sonore qui accompagne les images. Par rapport à l'image fixe, les documents animés ont un

certain nombre de caractéristiques.

La linéarité : une vidéo se regarde séquentiellement. Par cet aspect, elle s'apparente plus au texte qu'à l'image fixe : la visualisation des documents n'est pas instantanée, l'ordre de lecture du document est imposé. Toutefois, contrairement au texte, le sens du parcours peut varier : les images peuvent être projetées en respectant l'ordre dans lequel elles ont été filmées ou dans l'ordre inverse.

La temporalité : les images projetées imposent une vitesse de lecture au spectateur. Une image en chasse une autre. L'image filmée n'a pas forcément la même échelle temporelle que l'objet représenté ou l'événement relaté : la technologie permet de faire des ralentis, des accélérations pour montrer en quelques secondes une fleur qui s'épanouit par exemple, ou des arrêts sur image qui figent le mouvement.

L'accompagnement sonore : les images animées sont en général accompagnées d'une bande sonore. L'information délivrée par les images et la bande sonore sont complémentaires, chacune peut même n'avoir de sens qu'en présence de l'autre. Il peut s'agir de musique, de commentaire parlé, ou de bruits divers.

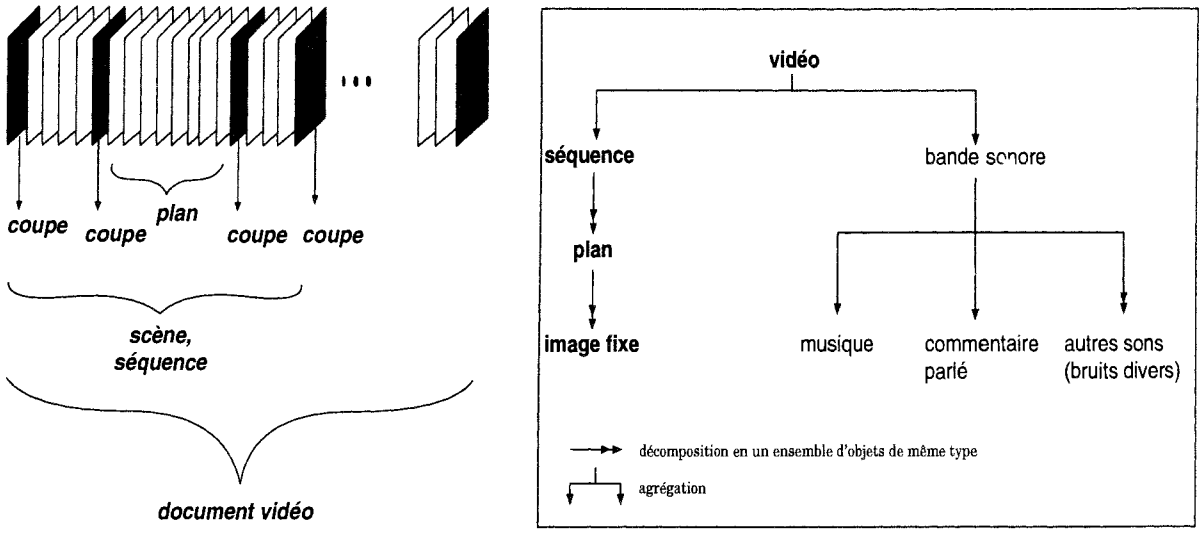
La production cinématographique ou télévisuelle a largement hérité de la tradition narrative littéraire : le générique, le titre et les sous-titres insérés lors du montage en sont des traces tangibles. Ces indices peuvent être utilement exploités lors d'une recherche d'information. Toutefois, le rapport du titre au document lui-même dépend de la catégorie de production de l'œuvre : descriptif pour un documentaire, quelquefois très éloigné du contenu pour un film.

Dans certaines approches, par exemple OVID [Oomoto 93], un document comporte un certain nombre d'*objets vidéo*, chaque objet étant défini par un ensemble d'intervalles de temps ; un intervalle est délimité par l'image où l'objet apparaît et celle où l'objet disparaît. Par exemple, un objet vidéo particulier correspond à l'apparition d'un personnage dans un film. Ainsi, les objets vidéo peuvent être juxtaposés ou se chevaucher. Cette approche est plutôt adaptée aux applications d'identification, où une chose précise est recherchée pour elle-même. Pour d'autres applications comme la recherche de séquences d'actualités, elle présente l'inconvénient de couper l'objet de son contexte. Seuls les liens temporels sont préservés.

En fait, la vidéo a son propre langage, et le travail de montage détermine une partie de sa syntaxe. Nous avons choisi de considérer dans un premier temps la vidéo comme un document structuré. Les différents niveaux de structure identifiés sont :

- l'image simple,
- **le plan**, constitué d'une succession d'images filmées en continu par la caméra,
- **la scène** ou **séquence**, qui est une suite de plans ayant une unité sémantique propre,
- **le document global** qui est une succession de séquences et correspond à l'unité de production.

Ces différents niveaux de structure peuvent être déterminés par des considérations très différentes d'un niveau à l'autre. Le niveau document correspond à l'unité de production, il est donc



(a) Structure d'un document vidéo

(b) Nature composite des documents

FIG. 1.4 – *La vidéo vue comme un objet structuré composite.*

aisément identifiable. De nombreux travaux de recherche en cours en traitement d'image s'intéressent au problème de la structuration automatique des documents vidéo. Les différents plans qui constituent un document peuvent être isolés les uns des autres par identification des plans de coupe, grâce à des algorithmes qui exploitent des histogrammes de couleur, des formes, ou des traits caractéristiques [Lee 94a, Cherfaoui 95, Picard 95]. Par contre, le niveau séquence ou scène est plus difficile à déterminer de manière automatique. En effet, puisque c'est le sens qui détermine ce niveau de structure, l'interprétation en est plus subjective. Toutefois, des indices à la fois visuels et sonores peuvent permettre de le déterminer. Pour des applications particulières, une régularité peut être observée dans la structuration en séquences [Swanberg 92]. Par exemple, dans les journaux télévisés, chaque séquence d'information est précédée et suivie d'une image d'un journaliste présentateur qui introduit le sujet.

1.2.4 Choix des niveaux de granularité pertinents pour la RI vidéo

La structure présentée plus haut, inspirée de l'analyse du discours narratif cinématographique [Colin 92], amène à se poser naturellement certaines questions. Ainsi, quel niveau de granularité doit être retenu pour la recherche et la visualisation des documents ? À quel niveau de granularité doit-on décrire ces documents ?

Dans le contexte d'une recherche interactive d'information, l'utilisateur va devoir visualiser plusieurs documents ou parties de documents que le système a retrouvés, afin d'être à même de décider s'ils répondent vraiment à son besoin d'information. Le visionnement successif de plusieurs documents entiers représente une charge cognitive très importante qui risque d'étourdir le spectateur. De plus, le pouvoir hypnotique de la vidéo est bien connu, ce phénomène s'ex-

plique en partie à cause du fait que ce média impose son propre rythme et sa propre vitesse de lecture. Le chercheur d'information devra donc avoir la possibilité de visionner uniquement des extraits courts de documents, pour conserver toute sa capacité d'action et d'analyse. Le plan, en règle générale, est trop court pour que le spectateur puisse en saisir le sens. Il faut donc pouvoir définir un **niveau séquence**, qui sera associé à l'unité sémantique minimale retenue pour la **présentation des documents** à l'utilisateur. Une solution simple consisterait à faire correspondre cette unité à une scène, c'est-à-dire à imposer au spectateur la structure élaborée par l'auteur. Cette solution peut être acceptable pour certaines applications, comme la recherche d'extraits de films par exemple. Elle ne convient pas pour l'ensemble des applications possibles où l'utilisateur doit pouvoir visualiser uniquement la portion de document qui l'intéresse. Par exemple, les chercheurs en sciences humaines qui analysent des comportements de groupes à travers des séquences filmées peuvent identifier des «tournants» décisifs dans le document qui ne se traduisent pas par des ruptures visuelles mais par le comportement des individus filmés [Duval 95].

Le problème se pose également de définir le niveau de granularité du document adéquat pour l'indexation. La scène n'est pas adéquate : il suffit pour s'en convaincre de considérer uniquement les dimensions spatiales et temporelles du discours. Une scène peut comporter différents plans qui n'ont pas d'unité de temps ou d'espace entre eux : le procédé des *flash back* (retour dans le temps passé par rapport à l'action principale) dans les films en est un exemple bien connu. La description de l'espace et du temps du discours, si elle est liée au niveau scène, risque fort d'introduire des ambiguïtés : il ne sera pas possible de distinguer une séquence tournée entièrement dans un même espace d'une autre mettant en parallèle des plans tournés dans des lieux éloignés. Ce problème est à rapprocher de celui de la référence dans les textes : un auteur peut utiliser un mot pour évoquer dans l'esprit du lecteur un ensemble de concepts. De même, au cinéma, des contextes peuvent être mis en parallèle pour que le spectateur les traite par analogie ou par contraste. Par exemple, des images de nature intercalées dans une scène montrant un prisonnier dans sa cellule évoquent l'idée de rêve de liberté.

La description du contenu au niveau scène serait très imparfaite : les objets qui apparaissent dans des plans différents seraient ramenés, par le biais de telles descriptions, à être contigus ou à cohabiter au sein d'un même espace virtuel. Le plan, tel que nous l'avons défini, a par nature une unité spatiale et temporelle : filmé en continu, il montre un même espace et se situe dans une seule et même unité de temps. C'est pourquoi dans la plupart des cas, la description du contenu peut être attachée au niveau des différents plans.

Le niveau image fixe peut, dans certains cas exceptionnels, être une unité d'indexation pertinente et se voir attacher des descriptions. Mais d'une part, à raison de 25 images par seconde pour les standards audiovisuels européens (30 images par seconde pour le standard NTSC des télévisions américaines ou japonaises), la quantité de descriptions à fournir serait énorme si elle était généralisée ; d'autre part, dans la plupart des cas, il existe une redondance importante entre les différentes images d'un même plan. D'ailleurs, certains chercheurs (par exemple [Zhang 95]) ont choisi d'indexer une ou plusieurs images représentatives par plan. La hiérarchie de composition qui représente une vidéo (voir figure 1.4) comporte alors au niveau le plus bas les images-clés choisies pour représenter le plan ; par exemple, un zoom est représenté par l'image de début, l'image du milieu et l'image de fin. Ces images clés peuvent ensuite être

analysées par des algorithmes de reconnaissance de formes.

Nous avons donc besoin d'un niveau de granularité variable auquel seront attachées les descriptions.

FIG. 1.5 – *Icone vidéo (zoom avant).*

L'image fixe peut être utilisée pour représenter une séquence d'image. Une série d'images fixes peut rendre compte sommairement d'un document entier : c'est le principe des photogrammes qui résument les films. M. Cherfaoui [Cherfaoui 95, p152] a aussi proposé le concept d'icone vidéo, *videoSpaceIcon*, (Figure 1.5) pour représenter un plan par une image iconifiée sur laquelle des indices graphiques incrustés indiquent le mouvement de la caméra.

Si les descriptions des documents sont exclusivement attachées aux plans (et éventuellement aux images), il risque d'y avoir une forte redondance, notamment en ce qui concerne les descripteurs qui se rapportent au document global : titre et genre de l'œuvre, nom de l'auteur. Par ailleurs, si nous voulons traiter la connotation d'un extrait, il peut être traité à différents niveaux : un document peut avoir une connotation générale, mais certains passages peuvent en avoir une différente. Nous devons donc établir un modèle de descriptions flexible qui repose sur la structure composite des documents vidéo, mais nous laisse la liberté d'attacher des descriptions à chaque niveau de granularité des documents : plan, scène, document global ou séquence quelconque. Nous utilisons donc la notion de **segment** pour l'indexation et la présentation, qui est définie par l'identification d'un document, le numéro de l'image de début du segment et le numéro de l'image de fin :

$$segment = (IdDoc, \#debut, \#fin)$$

Ces choix étant établis quant à la structure logique des documents, d'autres questions viennent à l'esprit. Quelles types de connaissances peut-on exploiter sur des documents vidéo ? Comment les représenter ?

1.3 Les connaissances et leur représentation dans les SRI

Les connaissances prises en compte par les SRI peuvent être regroupées en trois classes principales : les connaissances sur les documents d'une part (1.3.1), les connaissances sur les concepts du domaine de l'application (1.3.2), et les connaissances sur les utilisateurs du système (1.3.3). Ces connaissances sont destinées à être exploitées pour mieux répondre à un besoin d'information. Nous spécifions enfin les connaissances relatives à ce besoin 1.3.4.

1.3.1 Connaissances sur les documents

La limitation des ressources informatiques, la mise au point tardive des méthodes de recherche plein-texte[2] et la difficulté de faire interpréter par un ordinateur le langage naturel sont quelques-unes des raisons pour lesquelles, souvent, les documents des collections textuelles ne sont pas directement les objets de recherche : des représentants de ces documents sont élaborés lors d'une phase d'**indexation**. L'indexation a pour but principal de faire ressortir les sujets spécifiques des documents.

Les descripteurs ont d'abord été des mots-clés isolés extraits du titre et du texte, qui caractérisent les concepts importants évoqués dans le document, les objets, personnes, etc. qui y apparaissent. Mais le document ne reflète pas nécessairement l'état des connaissances de l'auteur : par exemple, un article dans un domaine scientifique pointu ne cite pas forcément le domaine scientifique en question. Pour obtenir une bonne indexation, il est souvent nécessaire d'ajouter des concepts qui ne sont pas explicitement dans le document. En outre, les mots-clés isolés n'ont pas forcément la même sémantique que lorsqu'ils sont regroupés. Les syntagmes, groupes de mots ayant un sens particulier par rapport aux mots isolés qui les composent (exemple : `intelligence artificielle`), captent davantage de sémantique. Pour les documents images (fixes ou animées), les descripteurs textuels peuvent traiter des éléments concrets représentés, mais aussi du représentant lui-même de ces éléments (lumière, type de plan par exemple). Les représentants peuvent être aussi des indices graphiques qui sont repérés par des traits caractéristiques, des formes, des couleurs ou des textures [Lee 94a, Gorkani 94].

Avec la disponibilité et le volume croissant des documents en ligne, l'indexation automatique s'est imposée. Elle procède par extraction de termes pour les documents textuels, d'indices visuels (traits, couleurs, textures) pour les documents images. Les chercheurs qui travaillent sur ce thème retrouvent les problèmes liés à l'interprétation du langage naturel (anaphores, ellipses, références, etc.) et à la reconnaissance automatique de formes pour les images.

Depuis l'avènement de la recherche plein texte, certains chercheurs ont trouvé pertinent de tenir compte de la structure des documents pour les décrire. La structure permet en effet d'identifier des concepts importants, qui se trouvent plus souvent dans certaines parties du document que dans d'autres (le titre, les en-têtes de sections ou la conclusion d'un article par exemple). Ces travaux s'attachent également à identifier de façon précise la partie de document qui répond le mieux au besoin d'information (un article précis dans une encyclopédie par exemple). La description structurelle des documents concerne leurs caractéristiques externes comme la segmentation en titre, auteur, et parties significatives du texte [Quint 94]. Pour les images fixes, la structure repose sur le découpage de la surface en régions ou sur les niveaux de profondeur dans lesquels se trouvent les objets représentés (premier plan, arrière-plan). Pour les images animées, la structure reflète le travail de montage des plans filmés, comme nous l'avons dit en au §1.2.3.

Lorsqu'elle est réalisée par un intermédiaire humain, l'indexation exige de l'indexeur[3] une bonne connaissance du domaine dans lequel le document a été produit. Le problème typique lié à

2. Une recherche plein-texte s'adresse au texte lui-même et non à des mots-clés choisis lors de l'indexation pour le représenter et stockés séparément de ce texte.

3. Nous désignons par ce néologisme la personne qui indexe les documents.

l'indexation manuelle est l'inconsistance : deux indexeurs différents vont-ils indexer de la même façon un même document ? [Ingwersen 92a] C'est pourquoi il est souhaitable que l'indexation s'appuie, au moins, sur un ensemble de connaissances communes.

1.3.2 Connaissances sur les concepts du domaine

Les thésaurus

Les connaissances sur les concepts du domaine sont collectées dans des structures de classification diverses dont les plus répandues sont les thésaurus. Un thésaurus est un réseau sémantique regroupant une collection de termes et de relations sémantiques entre ces termes. La constitution d'un thésaurus est une tâche lourde et difficile, pour laquelle des recommandations sont données par plusieurs normes (ISO 2788, ISO 5964, ISO/R639, Afnor NF Z 47-10). Ils peuvent être créés a priori ou incrémentalement – au fur et à mesure de l'arrivée de documents apportant de nouveaux concepts –, très spécialisés ou au contraire pluridisciplinaires. Les thésaurus sont utilisés quelle que soit la nature des documents stockés et indexés : d'abord développés pour les documents écrits, ils ont été adoptés également pour aider à la recherche de documents images.

Le plus souvent, les mots et concepts que les thésaurus désignent sont regroupés par domaines ou champs sémantiques. La définition d'un champ sémantique est abstraite : il s'agit d'un ensemble de mots qui ont un rapport de signification, cet ensemble permettant de donner le contexte conceptuel dans lequel se situe le mot. L'ensemble est structuré selon une organisation hiérarchisée.

Les relations entre les termes d'un thésaurus sont de plusieurs natures.

- Des relations d'équivalence font correspondre les termes **synonymes**. Cette relation est symétrique lorsque tous les termes d'une classe de synonymes peuvent être utilisés indifféremment lors de l'indexation (ceci permet d'utiliser le mot le plus juste pour la description d'un document), ou non symétrique lorsque, c'est le cas le plus répandu, un seul élément d'une classe d'équivalence a le statut de descripteur de document, les autres termes de la classe pouvant être uniquement descripteurs de requêtes (ils sont appelés *non-descripteurs*) et "traduits" automatiquement par le système. La construction a posteriori de ces classes d'équivalence à partir de l'analyse du langage d'interrogation des utilisateurs permet d'enrichir le thésaurus tout en le rapprochant de la population d'utilisateurs qui y accèdent. C'est une des manières de faire évoluer dynamiquement un thésaurus afin qu'il ne reste pas un fonds de connaissance statique et rigide.

- Les relations de hiérarchie que l'on trouve dans les thésaurus sont de deux sortes. La plus répandue est la **relation spécifique/générique**, qui a la sémantique de la relation de spécialisation *sorte_de* (ou *is_a*) en intelligence artificielle. Un champ sémantique est organisé sous forme d'un arbre dont la racine est le nom du domaine (le terme le plus générique) ; au fur et à mesure que l'on chemine vers les feuilles, les termes rencontrés sont de plus en plus spécifiques. Certains thésaurus sont organisés en **taxonomie** [4], une

4. La taxonomie est la science traitant de la classification des plantes et des animaux; par extension, ce terme

rubrique "fourre-tout" étant ajoutée à chaque niveau de l'arbre pour compléter les autres termes de même niveau. Ceci permet d'utiliser la relation inverse, la généralisation, en remplaçant par exemple un terme trop général par l'ensemble de ses «spécifiques».

Espèce animale	Localisation géographique
Invertébré	Afrique
...	Amérique
Mammifère	...
Mammifère marin	Asie
Mammifère terrestre	Europe
ongulé	France
antilope	Alsace
...	Aquitaine
boviné	Auvergne
bovin	Bourgogne
bœuf	Bretagne
génisse	...
taureau	Lorraine
vache	Meurthe & Moselle
veau	Meuse
bison	Moselle
buffle	Vosges
yack	...
zébu	Provence - Alpes - Côte d'Azur
carnivore	...
insectivore	Italie
...	...

Extrait du thésaurus Garnier, relations *sorte_de* (à gauche) et *partie-tout* (à droite)

- Une autre forme de hiérarchie relève plutôt de la **méronymie** ou relation *partie-tout*. C'est le cas des thésaurus géographiques, ainsi que de certains thésaurus architecturaux : les descendants d'un nœud sont les différents composants de l'entité associée au nœud. Par exemple, sous le concept `maison`, on trouve `toiture`, `murs`, `porche`, etc.

- Les relations d'association d'idées permettent une navigation transversale dans le thésaurus. Ce type de relation associe deux termes évoquant des idées proches mais qui ne sont pas reliés directement ou transitivement par une relation spécifique-générique, partie-tout ou synonyme-de. La relation qui existe entre des termes associés représente par exemple

  - l'appartenance à une classe (`ouvrier` → `prolétariat`), ou à un groupe (`sénateur` → `sénat`) ;
  - un lien de causalité (exemple : `feu` → `incendie`),
  - de localisation (`éléphant` → `savane` ou `voiture` → `garage`),
  - d'instrumentation (`film` → `caméra`),
  - de matière ou matériau (`vêtement` → `textile` ou `pneu` → `caoutchouc`),

désigne une structure de classification hiérarchique.

- d'origine (`lait` → `vache`)
- ou même d'antonymie (`écriture` → `oralité` ou `triste` → `gai`).

Ces relations sont très liées au domaine d'application du thésaurus et facilitent la navigation entre les concepts en fournissant des chemins de traverse. Elles sont particulièrement importantes pour la recherche d'images car elles fournissent des passerelles entre les concepts des différents domaines.

D'autres systèmes classificatoires pratiquent la pré-coordination des mots: les termes simples sont combinés en expressions plus ou moins complexes qui servent de descripteurs, répertoriés dans des polyhiérarchies (hiérarchies multiples). C'est le cas par exemple de la liste d'autorités matières de la *Library of Congress* des États-Unis. Par exemple, les termes `pollution` et `rivière` sont coordonnés pour former l'expression `pollution des rivières`. L'inconvénient de ces systèmes est leur explosion combinatoire. Ainsi, RAMEAU (Répertoire d'autorité matière encyclopédique et alphabétique unifié), utilisé au Cabinet des estampes et de la photographie de la Bibliothèque Nationale de France et qui provient d'une évolution de la liste des vedettes-matières établie par la Bibliothèque du Congrès américaine, est augmenté chaque année d'un tome supplémentaire [Dauzats 94].

L'utilisation d'un thésaurus n'est pas toujours possible, du fait de la difficulté liée à sa construction. Toutefois, cette base de connaissance est précieuse pour un SRI pour aider l'utilisateur à formuler sa requête comme dans HELGON [Fischer 89] ou RIVAGE [Halin 89]. De plus, l'affichage du thésaurus donne un retour conceptuel à l'utilisateur sur les connaissances du système. Il peut être aussi le support de l'inférence proprement dite comme nous le verrons au §2.3.1 page 60.

Dans le contexte de serveurs d'information distribués se pose le problème de l'échange de cette connaissance entre les systèmes. Il est difficile d'utiliser conjointement deux thésaurus différents d'une part à cause de l'amalgame qui est souvent fait entre les types de relations hiérarchiques (taxonomie et méronymie) et d'autre part parce que le choix des liens est souvent propre au domaine d'application pour lequel le thésaurus a été construit. C'est pourquoi nous allons formaliser les différentes relations du thésaurus, cette formalisation mettant en évidence le modèle de thésaurus utilisé par une application.

Formalisation des relations d'un thésaurus

Formellement un thésaurus TH peut être représenté par un ensemble fini de termes $\{t\}$ qui satisfait plusieurs conditions.

1. Il existe une relation $\leq$ de TH dans TH, la relation de spécialisation/généralisation, telle que

$$t_1 \leq t_1 \quad \text{réflexivité} \tag{1.1}$$

$$\forall t1 \neq t2,\ t_1 \leq t_2 \Rightarrow \neg(t_2 \leq t_1) \quad \text{asymétrie} \tag{1.2}$$

$$(t_1 \leq t_2) \wedge (t_2 \leq t_3) \Rightarrow t_1 \leq t_3 \quad \text{transitivité} \tag{1.3}$$

Cette relation est donc une relation d'ordre partiel. En général, les thésaurus pour cette relation sont des mono-hiérarchies : un terme a un seul générique. Si un terme t_1 a un homonyme t_2 dans un domaine D_i différent, ils sont considérés comme des objets distincts :

$$(t_1 \in D_1) \wedge (t_2 \in D_2) \wedge (D_1 \neq D_2) \Rightarrow t_1 \neq t_2$$

2. La relation *partie-tout* est aussi une relation d'ordre partiel[5]. Toutefois, elle doit être distinguée de la relation *sorte-de* car leurs propriétés ne sont pas les mêmes notamment en cas de combinaison de ces relations (voir les axiomes définis par exemple dans [Schiel 89]). Une étude détaillée des différents types de relations de composition est donnée dans [Napoli 92, p76]. Les hiérarchies *partie-tout* que l'on trouve dans les thésaurus traduisent dans la plupart des cas une relation d'inclusion spatiale entre les objets. C'est le cas des thésaurus géographiques ou architecturaux.

3. Lorsque tous les termes synonymes peuvent être utilisés pour l'indexation, la relation de synonymie est une relation d'équivalence dans l'ensemble T des termes du thésaurus. Lorsque les synonymes ont un représentant unique pour l'indexation, il existe un sous-ensemble non vide de T, $T_D \subset T$, qui regroupe les termes descripteurs[6] ; le sous-ensemble complémentaire dans T, noté T_{nD}, comprend tous les termes synonymes *non descripteurs* :

$$(T_{nD} \subset T) \wedge (T_{nD} \cap T_D = \varnothing) \wedge (T_D \cup T_{nD} = T).$$

De plus, si nous notons $\overset{syn}{\rightarrow}$ la relation de synonymie,

$$(\forall t \in T_{nD}), \exists t_D \in T_D (t \overset{syn}{\rightarrow} t_D).$$

Dans ce dernier cas, la relation de synonymie peut être appelée *employé pour*, et constitue une relation d'ordre partiel sur une classe de synonymes.

4. Nous avons vu que la relation d'association (ou *voir aussi*) entre les termes, notée ta, regroupe un certain nombre de sous-relations aux propriétés très différentes. Le tableau suivant détaille les propriétés de quelques-unes d'entre elles :

Type de relation	réflexivité	symétrie	transitivité(*)
causalité	non	non	non
matière/matériau	non	non	non
origine	non	non	non
localisation	non	non	non
instrumentation	non	non	non
antonymie	non	oui	non
(*) Seules les relations directes sont retenues.			

5. Certains considèrent que la relation *partie-tout* n'est pas réflexive, comme [Winston 87] cité dans [Napoli 92]. Pour notre part, nous faisons le choix de considérer le tout comme une partie.

6. Les termes *descripteurs* sont ceux qui peuvent être utilisés pour décrire les documents.

Paradoxalement, la relation générale ta (*termes associés*) est utilisée comme une relation d'équivalence dans les thésaurus que nous connaissons, avec les propriétés suivantes :

Type de sous-relation	réflexivité	symétrie	transitivité
terme associé	non	oui	non

La symétrie observée pour la relation *terme associé* provient d'une hypothèse implicite de conception des thésaurus : toute sous-relation a une sous-relation réciproque. Ainsi, la relation *cause* a pour relation réciproque *effet* par exemple. La symétrie est donc obtenue par composition de relations plus détaillées réciproques, sauf pour la relation *antonymie*. Par ailleurs, nous pouvons remarquer que les sous-relations ne sont en général pas transitives. L'explication que nous en donnons est qu'en général, ces relations sont multi-valuées : une chose a plusieurs causes, plusieurs effets, elle peut avoir plusieurs matières, et nécessiter plusieurs intruments pour sa réalisation. La transitivité n'est juste que si elle permet de conserver la multiplicité des relations. En ce sens, le comportement de la plupart de ces relations peut être rapproché de celui de la relation de composition. Si la relation ta était détaillée en sous-relations, il serait possible d'admettre une certaine transitivité. La composition des différentes sous-relations mériterait alors d'être étudiée au cas par cas.

5. Il reste un point que nous n'avons pas encore abordé, et qui est pourtant important à notre sens : le cas des **noms propres**. Dans le thésaurus de RIVAGE, les noms propres (de personnes, de lieux, de monuments) sont stockés artificiellement sous forme d'une hiérarchie plate dont le domaine et le générique sont "Précisions sur l'objet". Ces noms propres sont isolés des noms communs en ce sens qu'aucune relation n'existe entre les deux ensembles. Ceci implique une certaine redondance dans l'indexation, des photographies montrant la *Tour Eiffel* par exemple devant être indexées par *Tour Eiffel* pour l'attribut *Précisions sur l'objet* et par le mot *tour* pour l'attribut *Contenu*. Non seulement cette redondance pourrait être évitée, mais de plus il est possible d'exploiter judicieusement des connaissances sur les objets, lieux ou personnes à partir de leur nom propre. Si nous utilisons la terminologie objet, un nom propre permet d'identifier une instance particulière d'un objet. Par exemple, Rosine Bernard, dont le pseudonyme était Sarah Bernhardt, est un individu particulier de la classe « femme » et un individu particulier de la classe « acteur ». La relation qui existe entre un nom propre et les noms communs qui décrivent l'objet est une relation d'*instance de classe*. Le fait de rattacher une instance d'objet (au sens large) aux classes auxquelles il appartient permettrait de minimiser la redondance des descriptions, d'assurer leur cohérence (puisque chaque photographie de Sarah Bernhardt serait liée aux mêmes connaissances générales sur l'actrice). La relation *instance de* est stable pour sa composition avec la relation spécifique/générique :

$$instance - de(e, C) \wedge (C \leq C') \Rightarrow instance - de(e, C')$$

Cette relation pourrait être le support d'inférences sur les connaissances détenues par le système. Par exemple, à partir de la photographie d'une personne qui aurait été tour à tour chef d'entreprise et personnage politique, il serait possible d'accéder aux différents épisodes de sa vie pour constituer une rétrospective biographique. Par ailleurs, les pseudonymes peuvent être gérés comme les synonymes pour les noms communs. Toutefois, il

n'est pas nécessaire de stocker les noms propres dans le thésaurus ; ils peuvent constituer une base de connaissance distincte.

1.3.3 Connaissances sur l'utilisateur

De nombreux systèmes prennent en compte des connaissances sur l'utilisateur pour concevoir un processus de recherche adaptatif. Deux approches principales existent : un modèle de l'utilisateur est établi grâce à des questionnaires et est utilisé sous la forme de stéréotypes, ou le modèle utilisateur est construit incrémentalement par le système qui mémorise certaines informations durant les sessions de recherche d'un usager du système [Daniels 85, Thompson 89, David 90]. Ces deux approches ne sont pas incompatibles.

Les stéréotypes appliqués à l'utilisateur concernent le plus souvent :

- son niveau d'expérience vis-à-vis des SRI en général,
- son niveau d'expérience vis-à-vis du système concerné,
- son niveau d'expertise dans le domaine d'application (novice/expert),
- ses préférences, par exemple ses stratégies de recherche privilégiées ou le genre de ses besoins habituels (techniques, artistiques, émotionnels, etc).

Ces stéréotypes peuvent aider à déterminer quelques paramètres du processus de recherche comme par exemple dans IOTA [Defude 86] ou dans I^3R [Thompson 89] où le niveau d'expertise de l'utilisateur permet de contrôler le nombre de cycles maximum pour le bouclage de pertinence[7].

À notre avis, la modélisation de l'utilisateur est un problème délicat, et la généralisation des connaissances sur l'utilisateur est difficile : le danger est grand de l'enfermer dans des stéréotypes. Par contre, ces connaissances pourraient être exploitées par un SRI pour le choix et la présentation personnalisée des documents ; un *profil* de l'utilisateur par exemple pourrait permettre un filtrage initial de l'information, si c'est l'utilisateur lui-même qui gère ce profil.

1.3.4 Le besoin d'information de l'utilisateur

L'inconnue des SRI

Dans la problématique de la RI, il y a ce que l'on connaît - les documents, leurs représentants, le thésaurus, le processus de mise en correspondance - et ce que l'on ne connaît pas : le besoin d'information de l'utilisateur. Peter Ingwersen parle de «matière sombre» (*Dark Matter* [Ingwersen 92b]), que nous pourrions traduire par «anti-matière» par analogie avec l'astronomie, l'anti-matière étant active mais ne pouvant être observée qu'à travers ses effets sur son

7. Le bouclage de pertinence consiste à modifier l'expression de la requête suite à l'analyse des jugements de l'utilisateur sur les documents proposés par le système; nous reviendrons sur cette notion au §2.3.

environnement. En RI, un besoin d'information ne peut être observé qu'indirectement car la requête formulée par l'utilisateur ne représente que très imparfaitement son besoin d'information. D'autres indices peuvent être exploités comme ses réactions aux documents que le système lui propose, qu'il s'agisse de jugements de pertinence ou de choix de navigation.

Ainsi, la problématique de la RI interactive ne correspond pas aux critères énumérés par H. A. Simon [Simon 73] pour caractériser les problèmes bien structurés : en particulier, la présentation d'un ensemble de documents susceptibles de répondre à une demande imprécise ou exploratoire peut modifier l'état d'esprit du chercheur d'information ou même le but de sa recherche, ce changement ne se reflétant pas avec une précision totale sur le système. La RI peut donc être vue comme un cas particulier de résolution de problème complexe, le problème ici étant double : trouver des documents qui répondent au besoin d'information de l'utilisateur mais aussi déterminer à quel point la requête représente bien ce besoin d'information.

Diversité des besoins d'information

En recherche d'information, il faut parler *des* besoins d'informations de l'utilisateur au pluriel. Muriel Cluzeau-Ciry [Cluzeau-Ciry 88] a élaboré une typologie des demandes d'information en recherche d'image, et suggéré pour chaque type de besoin d'information une stratégie de recherche pouvant être mise en œuvre par un documentaliste. Quatre grandes catégories de demandes ont été identifiées :

- une **demande précise** est caractérisée par le fait que l'utilisateur sait très exactement ce qu'il cherche, peut-être même connaît-il déjà le document dont il a besoin. La demande comprend souvent un ou des noms propres, de personnes ou de lieux. Les paramètres de ce type de demande doivent être modifiés le moins possible par le documentaliste. Le chercheur d'information, dans ce type de situation, est souvent pressé par le temps, et souhaite trouver rapidement le(s) document(s) cherché(s). Si le document n'existe pas dans la collection auquel la demande est adressée, le documentaliste peut proposer de relâcher la demande (par exemple en traitant un seul critère à la fois plutôt que tous simultanément) ou de l'étendre à un sujet voisin.

- une **demande exploratoire** naît lorsque l'utilisateur veut se faire une idée sur le contenu d'une collection donnée. C'est une demande d'information de type culturel, l'utilisateur pouvant connaître ou non le sujet autour duquel son exploration va s'organiser. Le documentaliste peut lui proposer des "échantillons" qui représentent la collection demandée, et doit éventuellement pouvoir répondre aux questions qui pourraient naître lors de la visualisation de certaines images en fournissant des informations supplémentaires. Ce type de besoin est donc par nature inconstant, car il peut devenir thématique, connotatif ou même précis dans le cours de la session de recherche.

- une **demande thématique** est destinée à illustrer un thème ; ce type de demande est large et amène l'utilisateur à explorer la collection de documents sur un thème précis, par exemple l'œuvre d'un auteur, ou une école de pensée particulière, mais tous les éléments de la demande ne sont pas définis. Dans ce cas, le rôle du documentaliste est d'aider

l'utilisateur à développer sa question, de lui donner des idées. Le type de raisonnement en cause dans ce type de démarche est un raisonnement par association d'idées, stimulé par la visualisation des images.

- un **besoin connotatif** concerne l'expression d'un visage, une atmosphère, une ambiance. Les termes utilisés sont abstraits et subjectifs. Ce type de demande est celui qui fait le plus appel à des valeurs psychologiques et culturelles (voir page 8). Dans ce cas, un dialogue entre documentaliste et utilisateur essayera de déterminer l'utilisation prévue pour les images recherchées, en s'appuyant sur des images pour tester les impressions de l'utilisateur. La requête devra être reformulée en remplaçant les notions abstraites par des expressions plus concrètes. Par exemple, des images illustrant l'insécurité dans les villes pourront être des images représentant des scènes de violence collective (émeutes) ou individuelle (braquage d'une banque, vol à l'arraché).

Cette typologie peut être généralisée à d'autres documents que les images fixes. Elle suggère une façon d'adapter le comportement d'un SRI au type de demande auquel il doit répondre, le système se comportant alors comme un véritable assistant pour une recherche d'information. Malheureusement, il est très difficile de déterminer des critères dans une requête qui permettent de définir objectivement quel est le type du besoin [McCall 86, Croft 84]. C'est de ce constat qu'est née l'idée d'un modèle paramétré de recherche d'information, susceptible de s'adapter à divers types de demandes (nous présenterons ce modèle au chapitre 4).

1.4 Évaluation en Recherche d'Information

L'évaluation est un problème crucial et qui revient régulièrement sous les feux de l'actualité en RI. En effet, les évolutions technologiques remettent en cause les critères établis dans ce domaine pour les SRI non interactifs.

1.4.1 La pertinence

La **pertinence** est un concept abstrait souvent utilisé en recherche d'information. Ce concept recouvre des notions différentes selon que l'on se place du point de vue du système ou de l'utilisateur. Du point de vue du système, la pertinence est la correspondance dans le contexte entre l'énoncé d'un besoin d'information (une requête) et un document, c'est à dire le point auquel le document couvre la matière de l'énoncé du besoin [Jones 73]. Le problème est d'anticiper, lors de la conception du système, tous les besoins auxquels le SRI devra répondre. Ceci est bien évidemment impossible a priori, puisque l'apparition d'un système informatisé fait en général naître de nouveaux besoins.

Lorsque les documents passent par une phase intermédiaire de représentation de leur contenu, comment déterminer à quel point un document traite d'un sujet particulier et comment le refléter lors de l'indexation ? Faut-il d'ailleurs refléter ces degrés de pertinence lors de

l'indexation ? Il est possible de dire qu'un document traite d'un sujet s'il contient une information sur ce sujet. Mais on peut simplement dire qu'un document a un rapport avec un concept, qu'il traite d'un sujet à un certain degré ; c'est ce que van Rijsbergen désigne par le néologisme *aboutness* [Rijsbergen 92], que nous traduisons imparfaitement par *à-propos*.

Du point de vue de l'utilisateur, la pertinence dépend de l'utilité de chaque document que lui présente le SRI. Ainsi, un document peut être pertinent du point de vue du système pour la catégorie de sujets dont il traite, mais il peut ne pas être pertinent pour un utilisateur qui est déjà familier avec son contenu, ou qui a déjà retrouvé auparavant d'autres documents qui couvrent le sujet. L'utilité d'un document pour l'utilisateur ne peut être mesurée qu'à travers les jugements qu'il émet lorsque le SRI le lui présente. Elle dépend beaucoup du contexte, c'est-à-dire à la fois du but poursuivi lors de la recherche d'information et du contexte dans lequel naît le besoin d'information. Ainsi, un étudiant en histoire pourra rechercher un passage très précis dans des archives de journaux télévisés ; au contraire, un journaliste voulant réutiliser des images d'archives sur un personnage célèbre voudra obtenir des séquences représentatives où ce personnage apparaît, sans forcément savoir à l'avance lesquelles conviendront le mieux à son besoin. Le contexte culturel est également très important : la représentation d'un même concept sera différente pour des individus ne provenant pas du même milieu culturel. Le concept de *mort* par exemple est représenté dans nos sociétés occidentales par la couleur noire alors qu'en Afrique de l'Ouest, c'est le blanc qui symbolise la mort. De multiples connaissances doivent donc être prises en compte pour permettre d'évaluer de façon plus flexible la pertinence des documents vis-à-vis du besoin d'information d'un utilisateur.

Chaque étape du processus de recherche influence la pertinence des documents qui seront présentés pour l'utilisateur comme nous le verrons au chapitre 4.

1.4.2 Évaluation des résultats d'un SRI

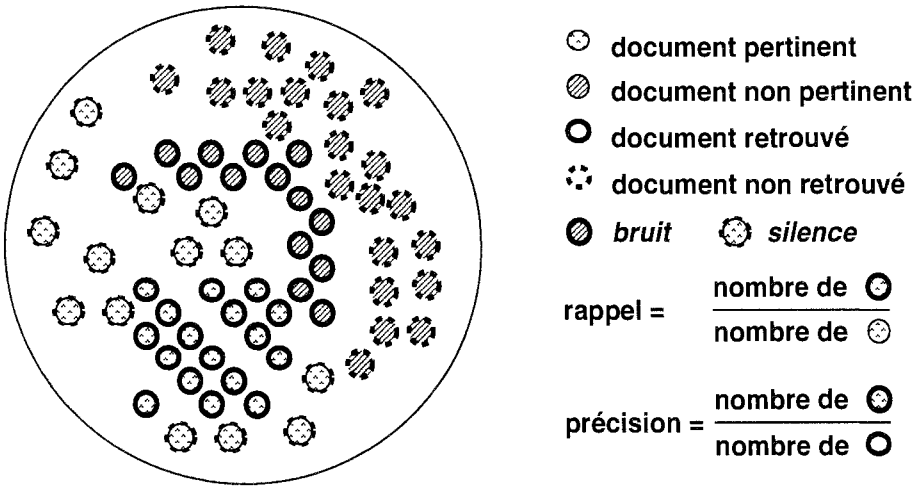


FIG. 1.6 – *Rappel, précision, silence et bruit en Recherche d'Information.*

Les mesures utilisées traditionnellement pour évaluer les SRI sont le rappel et la précision. Le **rappel** mesure la proportion de documents pertinents retrouvés par rapport au nombre total de documents pertinents dans la collection. La proportion complémentaire est le **silence**, c'est-à-dire la proportion de documents pertinents que le système n'a pas retrouvés. Les raisons du silence d'un système peuvent être une mauvaise indexation des documents, ou une mise en correspondance trop stricte et non adaptée à la situation. La **précision** mesure la proportion de documents pertinents retrouvés dans l'ensemble des documents retrouvés. La mesure complémentaire, le **bruit**, représente la proportion de documents non pertinents proposés à l'utilisateur par le système. Ces mesures posent un problème crucial : elles nécessitent de connaître a priori quels sont les documents pertinents pour une requête. Or, seul l'utilisateur est capable de dire si le document que lui présente le système est pertinent pour son besoin d'information. Ainsi, une requête similaire soumise par deux utilisateurs différents peut être satisfaite de manière différente pour chacun d'eux.

Si la précision est mesurable à travers les jugements de pertinence de l'utilisateur, le rappel pose davantage de problèmes : comment évaluer quels sont les documents pertinents non retrouvés par le système ? Ce problème a été résolu par la constitution de collections de tests, où un jeu de requêtes préétabli permet de déterminer, pour chacune d'elles, quels sont les documents pertinents. Ces collections ne sont disponibles actuellement que pour des documents textuels, comme la collection CRANFIELD [Harman 92]. Une initiative récente de la communauté de recherche d'information a consisté à mettre en place des conférences dédiées à l'évaluation : TREC [Harman 93]. Ces conférences sont organisées comme des sortes de tournois : avant la conférence, des requêtes sont soumises à l'ensemble des participants, qui les confrontent à leurs systèmes et envoient en retour la liste classée des documents qu'ils ont retrouvés. Des personnes analysent ensuite ces ensembles de documents et déterminent s'ils sont réellement pertinents vis-à-vis de chaque requête.

La constitution des collections de test est une tâche lourde et difficile : comment s'assurer que les requêtes sont représentatives des différents types de besoins qui pourront se manifester ? De plus, pour des documents multimédias, l'unicité de jugement de pertinence est encore plus improbable : si tout le monde peut être d'accord pour reconnaître un personnage célèbre sur une photographie, des individus différents choisiront certainement des clichés différents le représentant dans des situations variées ; par exemple, un journaliste d'un hebdomadaire satirique et un reporter d'un quotidien géneraliste choisiront des portraits différents d'un même personnage politique.

Par ailleurs, ces mesures, mises en place pour évaluer les SRI non interactifs, ne prennent pas en compte un certain nombre de paramètres complémentaires. En particulier, les SRI présentent souvent seulement une partie des documents ayant une certaine pertinence pour le besoin d'information exprimé, et effectuent une censure qui s'appuie soit sur un seuil minimal de pertinence en-deçà duquel les documents ne sont pas présentés, soit sur un nombre maximum de documents « visualisables » par l'utilisateur. L'évaluation des SRI interactifs est donc encore un problème ouvert. Nous reviendrons ultérieurement sur ce point.

1.5 Bilan

Nous avons présenté de façon générale la problématique de la recherche d'information et les spécificités liées au traitement de collections de documents multimédias. Nous avons essayé de montrer combien la recherche d'information peut être assimilée à une résolution de problème complexe, et de mettre en évidence les différents aspects auxquels elle doit s'intéresser pour mieux répondre aux besoins des utilisateurs : modélisation des connaissances diverses qui sont impliquées dans une recherche d'information, acquisition de connaissances sur le besoin d'information de l'utilisateur, adaptation du système en fonction du contexte.

Depuis le début des recherches dans le domaine de la RI, de nombreux modèles ont été proposés pour représenter les connaissances et effectuer l'opération-clé de la recherche : la mise en correspondance entre documents et requête. Nous analysons dans le chapitre suivant un certain nombre de modèles existants.

Chapitre 2

Modèles de Recherche d'Information

Notre but dans ce chapitre n'est pas de faire un état de l'art exhaustif sur les modèles de recherche d'information. Nous souhaitons plutôt mettre en lumière les avantages et inconvénients liés à certains types de modèles existants de façon à expliquer en partie les propositions que nous serons amenée à faire par la suite.

Nous commençons par la présentation des premiers modèles qui ont permis de retrouver des documents en réponse à une requête : le modèle booléen (§ 2.1.1) et le modèle vectoriel (§ 2.1.2), tous deux expérimentés dans l'équipe EXPRIM ; nous parlerons de modèles plus complexes, comme le modèle booléen étendu (§ 2.1.3). Nous présentons également des approches alternatives, en particulier celles d'inspiration probabiliste. Après ce tour d'horizon, nous nous demanderons, avec C. van Rijsbergen, ce que doit être un modèle de RI (§ 2.2). Enfin, puisque la RI qui nous intéresse est interactive, nous examinons le principe du bouclage de pertinence et les différentes interprétations qui en ont été faites (§ 2.3).

2.1 Les premiers modèles de mise en correspondance pour l'accès par requête

Un modèle de recherche d'information spécifie d'une part les représentations utilisées pour les documents et les demandes d'information, et d'autre part la façon dont ces représentations sont comparées. L'opération principale qui compare documents et requête est la mise en correspondance (*matching* en anglais). Le modèle booléen que nous présentons d'abord effectue une mise en correspondance exacte entre documents et requête. La prise en compte de l'incertitude liée à la représentation des documents et du besoin d'information ont fait naître des modèles de mise en correspondance partielle : modèle vectoriel (§ 2.1.2), modèle probabiliste (§ 2.1.4 page 41). Ces modèles cherchent à obtenir un classement des documents par ordre décroissant de pertinence vis-à-vis de la requête. Une autre tactique cherche non pas à établir un classement ordonné des documents, mais à les regrouper en *clusters*.

2.1.1 Le modèle booléen

Dans le modèle booléen, les documents sont décrits par une conjonction de termes et la requête par des termes connectés par les opérateurs booléens ET, OU et NON.

Exemple :	D = chat ET cheminée
	Q = chat ET (tapis OU panier)

Ce modèle simple s'appuie sur l'algèbre booléenne et les opérations ensemblistes correspondantes. Son implantation est efficace grâce aux **listes inverses**. Une liste inverse associe à chaque terme descripteur les documents qu'il décrit. Les opérations booléennes peuvent alors être interprétées sous forme d'opérations ensemblistes sur les listes de documents associées aux termes de la requête : ET s'obtient par l'intersection des listes inverses associées à chacun des termes, OU s'obtient par l'union, NON s'obtient en prenant le complémentaire.

Les opérateurs booléens permettent de structurer la description de la requête, de faire apparaître les dépendances mutuelles qui existent entre les termes descripteurs [Salton 85] : la relation OU fait apparaître une relation de synonymie ou quasi-synonymie, la relation ET permet de définir un concept complexe par une conjonction de sous-concepts ; par exemple `outils` ET `jardin` peut être interprété comme un concept «composite» désignant des outils de jardinage.

La mise en correspondance booléenne est exacte : un document répond ou non à la requête. De ce fait, les résultats obtenus ne sont pas suffisamment nuancés. En particulier, [Salton 83b]

- La conjonction est trop sévère, trop stricte : un document contenant tous les termes de la requête sauf un est jugé aussi inutile qu'un document n'en contenant aucun. De plus, la conjonction devient de plus en plus restrictive au fur et à mesure que le nombre d'entités considérées croît.
- La disjonction est trop permissive : dans une requête $A\ OU\ B$, les documents contenant un seul des termes sont classés au même niveau que ceux qui contiennent les deux, ce qui est contre-intuitif.
- L'interprétation de l'opérateur booléen NON est délicate : il est souvent interprété comme $SAUF$, c'est-à-dire par rapport au monde fermé des termes de la requête.

À ma connaissance, il n'existe pas de système documentaire booléen indexant les documents par des critères niés ; pour reprendre l'exemple du tableau de Magritte «Ceci n'est pas une pomme» (page 9), comment représenter l'information liée au texte «NON pomme» ? D'autre part, l'indexation des documents est en général considérée comme aussi complète que possible, mais ceci à un instant donné : l'absence d'un terme dans la description d'un document ne prouve pas absolument que le document ne traite pas du concept correspondant, mais que ce concept n'a pas été retenu comme pertinent au moment où l'indexation a été réalisée. L'évolution des connaissances et des faits dans le temps peut remettre en cause une indexation effectuée à l'instant t. Ainsi, une photographie montrant un personnage célèbre au milieu d'un groupe indexée

à l'époque où il n'était pas encore connu ne sera pas indexée par le nom de ce personnage, sauf en cas de révision périodique de l'indexation.

Ce modèle ne permet donc pas de distinguer un document qui comprend quelques termes de la requête de celui qui n'en contient aucun. L'information sur le nombre de termes communs entre requête et descriptions de documents est perdue.

Un autre inconvénient grave du modèle booléen est la difficulté pour un utilisateur de formuler une requête booléenne exprimant exactement son besoin : l'utilisation de ET et de OU pour un utilisateur ne correspond pas à l'interprétation qu'en fait un ordinateur [Avrahami 93]. Cet inconvénient tend à être comblé par les langages graphiques de formulation de requête [Bélières 95] et la construction automatique de requêtes booléennes à partir de mots-clés [Motro 86].

Le modèle booléen reste cependant assez fréquemment utilisé pour effectuer une présélection parmi les documents comme dans les systèmes SMART [Salton 83a] ou SIRE [McGill 76]. Dans ces situations, le connecteur logique ET dans les requêtes est interprété comme un OU (inclusif).

Les systèmes booléens les plus connus sont MEDLARS (1971), devenu MEDLINE, qui permettait d'interroger la National Library of Medicine (Bibliothèque Nationale de Médecine des États-Unis) et TEXTO (1982).

D'autres approches permettent une mise en correspondance partielle, plus proche de l'interprétation intuitive que l'on peut se faire de cette opération.

2.1.2 Le modèle vectoriel

Le modèle vectoriel *(vector space model)* est un des plus anciens en RI [Salton 68]. Simple à mettre en œuvre, il permet de discriminer les documents pertinents en leur affectant un score qui rend leur classement possible.

Principe

Dans le modèle vectoriel, documents et requête sont vus comme des vecteurs dans un espace à n dimensions, une dimension par terme descripteur. C'est l'ensemble T des termes d'indexation qui constitue la base de l'espace vectoriel. Les termes constituant les descriptions des documents ont tout d'abord été vus comme des valeurs binaires : le ième composant d'un vecteur indiquait la présence (1) ou l'absence (0) dans le document du terme ou du concept représenté par le $i$ème terme d'indexation de T. La requête est décrite de la même façon que les documents. Très vite, les termes ont été pondérés dans un intervalle continu [0,1] pour représenter leur degré d'importance dans un document ou une requête. Cet intervalle a été élargi à des valeurs négatives dans certains travaux, comme nous le verrons en page 35.

Le reproche le plus courant fait au modèle vectoriel est l'hypothèse implicite selon laquelle les termes d'indexation forment une base orthogonale de l'espace vectoriel dans lequel docu-

ments et requête sont représentés. En effet, ce modèle ne permet pas de représenter les corrélations qui existent entre certains termes dans les documents ou la requête.

E. Fox a proposé d'étendre le modèle vectoriel pour prendre en compte la diversité des éléments d'information contenus dans un document. Il observe qu'un document est un objet composite, au sens où il peut contenir divers types d'information. Par exemple, un article peut contenir de l'information bibliographique, des citations et des mots-clés représentant les catégories descriptives principales du document. Ainsi, un vecteur document consiste en un ensemble de sous-vecteurs, où chaque sous-vecteur représente une classe de concept différente (il s'agit de classes sémantiques). Pour un article, ces catégories de concepts seraient l'*identification objective du document* comme le nom de l'auteur, le *contenu*, et les *citations*. La similarité entre un couple de vecteurs étendus est calculée par une combinaison linéaire des mesures de similarité des sous-vecteurs correspondants. Ceci revient à introduire des attributs de description, chaque attribut pouvant prendre ses valeurs dans un domaine particulier. De nombreuses stratégies ont été étudiées pour la génération automatique de requêtes vectorielles étendues et la sélection des coefficients de pondération pour chaque catégorie de concepts [Crouch 90].

L'utilisation conjointe d'un thésaurus améliore considérablement les performances de ce modèle. L'hypothèse d'indexation des documents aux feuilles du thésaurus, c'est-à-dire par les termes les plus spécifiques, permet de restreindre la base de l'espace vectoriel considéré, et conforte intuitivement l'hypothèse d'indépendance entre les termes de la base. Cette hypothèse contraint l'indexation mais pas l'expression de la requête dont le vocabulaire peut être élargi à l'ensemble du thésaurus.

Mesures de similarité

De nombreuses mesures de similarité ont été utilisées dans ce modèle. Nous les appelons ici R (comme *Relevance*, mot anglais ayant une signification plus précise que pertinence en français). La plupart de ces mesures sont issues de travaux sur l'analyse de données.

1. Le **produit scalaire** revient à calculer le nombre de correspondances entre les 2 vecteurs. Si w_{il} représente le poids du terme i pour l'entité l (requête ou document), le produit scalaire mesure la correspondance d'un document D_j et d'une requête Q_k sur t termes par :

$$R_{ProdScal}(D_j, Q_k) = \sum_{i=1}^{t} w_{ij}.w_{ik}$$

   Le nombre t de termes correspond en général à la taille du vocabulaire d'indexation. Cette mesure avantage les documents contenant de nombreux descripteurs. Diverses mesures normalisées ont été proposées pour parer à cet inconvénient.

2. La **mesure du cosinus** de l'angle que forment le vecteur document et le vecteur requête permet de retrouver tous les documents dans un cône défini par l'origine - le sommet du cône -, la requête comme axe de définition, le rayon étant défini par un seuil. Les

documents peuvent être classés par la mesure de l'angle qu'ils forment avec le vecteur requête. L'aspect le plus intéressant de cette mesure est l'influence d'un terme isolé sur le score de recherche. Si un descripteur est présent à la fois dans la requête et le document, il contribue au score. S'il est présent uniquement dans l'un des deux, il diminue le score parce que la requête et le document se correspondent moins.

La formule de cosinus, entre un document $D = (d_1, d_2, \ldots, d_n)$ et une requête $Q = q_1, q_2, \ldots, q_m)$ s'écrit :

$$R_{cos}(D,Q) = \begin{cases} \frac{|D \cap Q|}{\sqrt{|D| \times |Q|}} & \text{(écriture ensembliste)} \\ \frac{\vec{D}.\vec{Q}}{||\vec{D}|| \times ||\vec{Q}||} & \text{(écriture vectorielle)} \\ \frac{\sum_{k=1}^{t}(d_{ik} \cdot q_{jk})}{\sqrt{\sum_{k=1}^{n}(d_{ik})^2 \cdot \sum_{k=1}^{m}(q_{jk})^2}} & \end{cases}$$

L'opérateur . représente le produit scalaire et $\| \vec{D} \|$ la norme euclidienne du vecteur $\vec{D}$. Si les poids varient dans l'intervalle [0,1], le cosinus vaut 1 si les vecteurs sont parallèles, 0 s'ils sont orthogonaux. Si certains termes de la requête comportent des poids négatifs (voir 2.1.2.0), ceux-ci vont participer à diminuer le score des documents indexés par ces termes. A noter qu'en présence de poids négatifs, la mesure de similarité varie dans l'intervalle [-1,1].

3. le coefficient de **Dice** s'exprime par :

$$R_{Dice}(D,Q) = \frac{2 \times \mid D \cap Q \mid}{\mid D \mid + \mid Q \mid}$$

4. le coefficient de **Jaccard**, dont l'interprétation vectorielle est appelée mesure de Tanimoto ou S-mesure, est une des mesures les plus utilisées :

$$R_{Jacc}(D,Q) = \frac{\mid D \cap Q \mid}{\mid D \cup Q \mid - \mid D \cap Q \mid} \tag{2.1}$$

Ces différents coefficients n'ont pas le même comportement pour le classement des documents, même si des corrélations existent entre eux. Par exemple, les auteurs de [Hamers 89] ont montré que

$$R_{Jacc}(D_i,Q) \approx 2 \times R_{cos}(D_i,Q) \tag{2.2}$$

Pour les systèmes de recherche d'information qui utilisent une valeur de seuil en-deçà duquel les documents sont estimés non pertinents, la mesure de cosinus peut ainsi trancher plus brutalement que les autres. Par ailleurs, même si les mesures sont normalisées, l'amplitude des valeurs données par ces différentes valeurs peut être assez différente (Figure 2.1).

Par exemple, étant donné la requête d'un utilisateur cherchant des documents sur les tremblements de terre et les incendies, et si possible à San Francisco :

Q = tremblement de terre (1.0), incendie (1.0), San Francisco (0.5)

et la collection de documents :

D1 = tremblement de terre (1.0), incendie (1.0)
D2 = tremblement de terre (1.0), incendie (1.0), San Francisco (1.0)
D3 = tremblement de terre (1.0), incendie (1.0), Japon (1.0)
D4 = incendie (1.0), volcan (1.0)

les valeurs obtenues pour chacune des mesures sont différentes :

Documents	**Valeurs de la mesure de similarité**			
	Produit Scalaire	Cosinus	Dice	Jaccard
D1	2	0,44	0,88	0,80
D2	2,5	0,37	0,90	0,83
D3	2	0,29	0,72	0,57
D4	1	0,22	0,44	0,28

Ces mesures donnent, par ordre décroissant des valeurs, les classement suivants :

Documents	**Rang obtenu pour les différentes mesures**			
	Produit Scalaire	Cosinus	Dice	Jaccard
D1	2	1	2	2
D2	1	2	1	1
D3	2	3	3	3
D4	4	4	4	4

D'une part le classement des documents est différent pour ces mesures, d'autre part, lorsque deux mesures donnent le même classement, leur amplitude varie énormément ; par exemple, la figure 2.1 montre que la mesure de Jaccard donne une distance plus importante entre les documents les moins bien classés que le coefficient de Dice.

Le modèle vectoriel ne permet pas de prendre en compte directement les dépendances qui peuvent exister entre certains concepts. Il n'y a aucun moyen de refléter le fait qu'un terme d'indexation absent de la requête est fortement lié aux termes que l'on trouve dans la requête. Les SRI qui reposent sur ce modèle effectuent donc souvent une phase de traitement préliminaire s'appuyant sur un thésaurus pour tenir compte de ces dépendances.

Quelques remarques sur la pondération des concepts

La représentation vectorielle repose sur des hypothèses implicites quant au sens des poids affectés aux termes, qu'il s'agisse des descripteurs de documents ou des descripteurs de requête. La pondération des concepts permet de nuancer la capacité de description des termes : il est ainsi possible de graduer les différents concepts utilisés. La pondération s'est d'abord appuyée sur

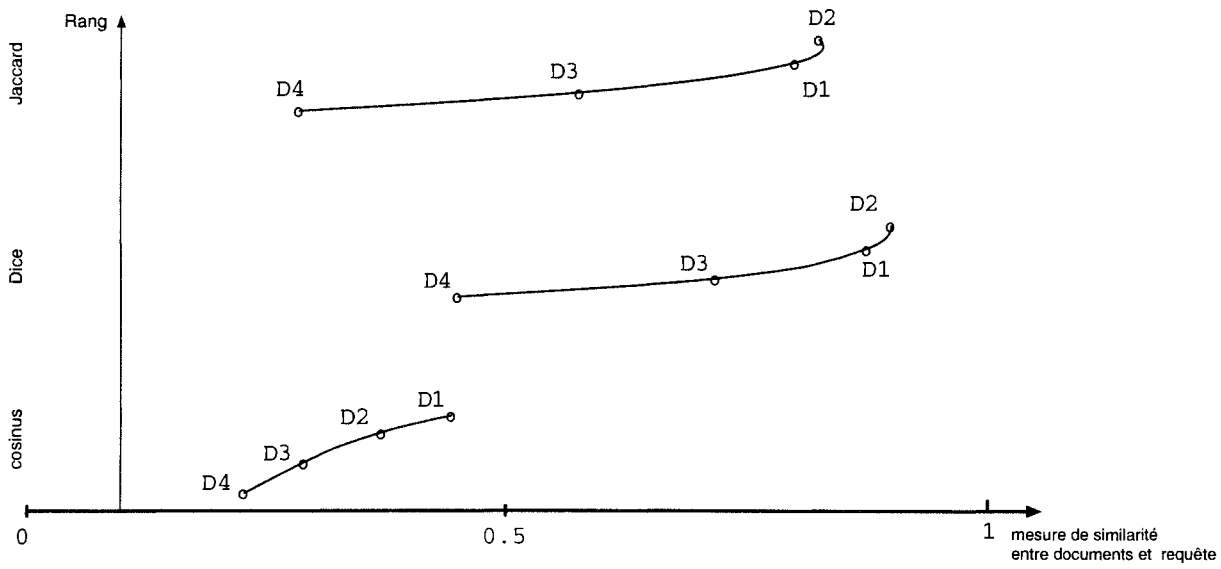


FIG. 2.1 – *Comparaison des résultats obtenus avec des mesures différentes.*

la Loi de Zipf [Zipf 49] : cette loi est expliquée par une loi générale du moindre effort qui encourage un orateur ou un écrivain à réutiliser et répéter certains mots plutôt que d'en forger ou trouver de nouveaux. Elle permet de distinguer des classes de mots dans les textes en fonction de leur fréquence d'occurrence.

Une approche différente de la pondération des termes, plus proche du concept d'entropie, a été introduite par Sparck-Jones [Jones 72]. L'importance d'un terme dans une recherche repose sur la façon dont ce terme permet de distinguer un document d'un autre. Si un mot est utilisé par peu de documents, son **pouvoir de discrimination** est fort. Ceci permet d'introduire la notion de fréquence documentaire d'un terme. En général, l'importance d'un terme est inversement proportionnelle à sa fréquence documentaire : un terme qui se retrouve dans tous les documents de la collection n'est pas un bon candidat pour discriminer ces documents. La fréquence documentaire d'un terme est souvent estimée par un logarithme pour éviter de devoir traiter des nombres très petits. Par ailleurs, pour un document donné, un terme est d'autant représentatif de son contenu qu'il s'y retrouve souvent (c'est la loi de Zipf [Zipf 49]). Le poids d'importance d'un terme k dans un document i peut donc être proportionnel à la fréquence standard d'occurrence de ce terme dans le document. La combinaison de ces deux mesures permet de déterminer l'importance d'un terme pour la représentation d'un document dans une collection. Ce poids est noté $tf.idf$ (term frequency.inverse document frequency). Cette mesure peut être normalisée pour ne pas avantager les documents longs et décrits par des termes plus nombreux. Le poids w_{ik} du terme i pour le document k est calculé par :

$$w_{ik} = \frac{tf_{ik}.log(N/N_i)}{\sqrt{\sum_{j=1}^{t}(tf_{ij})^2.(log(N/N_j))^2}}$$

où tf_{ik} est la fréquence d'occurrence du terme i dans le document k (une fréquence nulle signifie que le terme n'est pas affecté au document), N est la taille de la collection (nombre de

documents), N_i représente le nombre de documents dans la collection décrits par le terme i et t est le nombre total de termes d'indexation du document.

L'avantage de ce système de pondération est qu'il est directement utilisable si l'on veut montrer à l'utilisateur des exemples de documents illustrant un concept unique : il suffit de rechercher les documents pour lesquels le terme ou le concept considéré est le plus fortement pondéré. Du point de vue de l'implantation, elle permet d'optimiser la représentation par listes inverses en classant, pour chaque terme, la liste des documents qu'il décrit par ordre décroissant de mesure *tf.idf* : les documents les plus représentatifs d'un concept se trouvent alors en tête de liste, et la queue de la liste peut être ignorée. Le gain en temps de traitement peut donc être important pour les grosses collections de documents. Cette approche de la pondération est bien adaptée à des collections de documents figées mais peu opérationnelle pour des collections de documents dynamiques (c'est-à-dire où des documents sont ajoutés ou supprimés fréquemment) : en effet, elle oblige à recalculer les poids lorsqu'un document arrive ou sort de la collection. De plus, dans un contexte de serveurs d'informations distribués, les poids d'un terme indexant un même document dans des collections différentes variera d'une collection à l'autre. Des mesures de similarité entre des documents et une requête calculées à partir de ces mêmes poids sur chaque collection donneront un classement différent pour un document donné en fonction de la collection dans laquelle il se trouve, puisque le poids d'importance de chaque terme tient compte de sa fréquence dans les descriptions d'une collection particulière. Ceci contredit en particulier l'hypothèse d'indépendance des termes pour un modèle vectoriel : cette hypothèse n'est remplie qu'à la condition que la distance entre deux documents identiques à un terme près soit la même, quelle que soit l'identité de ce terme. C'est pourquoi l'utilisation des poids $tf.idf$ est délicate du point vue théorique si on utilise le modèle vectoriel pour une recherche dans des collections de documents distribuées. Par contre, ce type de mesure peut donner une photographie instantanée d'*une* collection particulière à un moment donné.

On pourrait imaginer que la **pondération** des termes de description d'un **document image** ait une sémantique différente de celle exposée plus haut. Pour une image fixe, le poids affecté à un terme se rapportant au contenu de l'image pourrait être proportionnel à la surface de l'objet dans l'image (ou à la surface de la région englobante). Cette sémantique de la pondération des descriptions de documents risque d'entrer en conflit avec la sémantique de pondération des termes de la requête. En effet, si, dans la requête, l'utilisateur exprime des degrés d'importance relative pour les concepts, l'importance d'un objet dans une image n'est pas forcément proportionnelle à sa surface. Une autre approche de la pondération des descriptions de documents pourrait prendre en compte le champ dans lequel l'objet s'insère dans l'image (premier plan, second plan, arrière-plan). Il est en effet plausible que les objets apparaissant au premier plan de l'image soient les plus importants. Nous venons donc de mettre en évidence deux interprétations possibles des poids pour les documents image. Ceci montre la difficulté de donner une sémantique explicite aux poids d'indexation. La difficulté a été contournée dans les systèmes que nous connaissons comme Rivage ou Cabri-n en associant à tous les descripteurs un poids maximum (égal à 1) lors de l'indexation.

Pour des vidéos, nous avons envisagé d'affecter au descripteur un poids représentant le nombre d'images dans lesquelles l'objet apparaît par rapport à une séquence globale [Simonnot 95a]. Ainsi, pour un segment de document vidéo S_i, le poids $w_{i,t}$ du descripteur

t associé est calculé au prorata de son apparition :

$$w_{i,t} = \frac{\mid S_i \cap S^t \mid}{\mid S_i \mid}$$

avec :

$$\begin{cases} \mid S_i \mid & \text{: nombre d'images dans le segment courant } i,\ S_i \neq \varnothing \\ \mid S^t \mid & \text{: nombre d'images où apparaît l'objet décrit par le terme } t, \\ \mid S_i \cap S^t \mid & \text{: nombre d'images où l'objet décrit par le terme } t \text{ apparaît dans le segment } i, \end{cases}$$

Cette mesure simple n'est qu'une première approximation. En effet, elle ne permet pas de distinguer si les apparitions de l'objet sont courtes mais fréquentes, ou si l'objet apparaît une seule fois et reste à l'image pour une durée de temps assez longue. Un détail peu important d'un décor sera plus fortement pondéré qu'un objet qui apparaît fortement mais brièvement à l'image. De plus, si la mise en correspondance se fait au niveau des séquences, cette mesure met à plat le contenu des différents plans, ce qui n'est pas forcément souhaitable. Enfin, elle ne correspond pas à la sémantique donnée traditionnellement aux poids pour la formulation de la requête, que ce soit pour une recherche de textes ou d'images fixes, ce qui est gênant si l'on veut faire cohabiter ces différents types de média au sein d'un même système. Une représentation temporelle plus concrète des concepts dans le document serait donc plus riche.

En ce qui concerne la pondération des termes de la requête, les poids sont considérés comme des poids d'importance du concept pour la satisfaction de la demande. Un poids nul représente l'indifférence du concept pour le besoin de l'utilisateur. Lorsque ce poids croît, l'importance du concept augmente jusqu'à un maximum de 1. L'intervalle de valeurs des poids d'importance pour la requête peut être étendu à des **valeurs négatives**. Lorsque le poids est négatif, il exprime l'importance de l'absence du concept dans les documents recherchés. Ceci peut se traduire par deux fonctions : une fonction positive "À sélectionner" prenant les valeurs "absolument, si possible, etc", valeurs qui ont une correspondance dans l'intervalle [1 : 0,1], et une fonction négative "À éviter" prenant les valeurs "absolument, si possible, etc", ces valeurs ayant une correspondance dans l'intervalle [-1 : -0,1]. Ceci permet à l'utilisateur d'exprimer une partie de son besoin par la négative, en exprimant ce qu'il ne souhaite pas retrouver. Encore faut-il s'assurer que les mesures de similarité utilisées traitent bien ces poids négatifs.

Une autre approche de la pondération des termes de la requête a été étudiée qui s'inspire de la théorie de Dempster-Sheffer : le poids attribué par l'utilisateur à un terme représente le degré de croyance de l'utilisateur en la représentativité de ce terme vis-à-vis de son besoin d'information pour la collection à laquelle il s'adresse (voir page 43). Des expériences ont montré que les utilisateurs avaient du mal à manipuler le concept de croyance, qu'ils confondent en général avec le degré d'importance du terme pour le besoin d'information [Croft 90b].

L'interprétation de la pondération dans un modèle doit être examinée avec soin avant qu'un choix ne soit fait. En effet, selon le modèle de mise en correspondance retenu, la pondération des termes est considérée comme absolue ou relative. Par exemple, dans les modèles flous, la pondération est interprétée de manière absolue, à cause de l'intervention des fonctions **minimum** et **maximum**. Dans les modèles booléens étendus et dans le modèle vectoriel, la pondération des termes est relative : le poids d'un terme n'est pris en compte que relativement à celui d'un

autre terme. Ainsi, deux requêtes comportant les mêmes descripteurs avec des poids différents mais proportionnels donneront le même résultat pour ces derniers modèles. Cette remarque est importante si nous souhaitons faire collaborer des méthodes de mise en correspondance différentes : la pondération des termes de la requête est interprétée de manière très différente par le modèle flou par rapport aux autres approches présentées.

Enfin, les concepts ne sont pas les seuls éléments à pouvoir être pondérés dans un SRI. Dans les systèmes où les documents sont regroupés par clusters, des poids attachés au lien qui unit deux descripteurs ou deux documents peuvent représenter leur degré de proximité.

Nous soulignons la diversité des interprétations possibles de la pondération pour montrer qu'il s'agit d'une difficulté parmi d'autres lorsque l'on veut réaliser une recherche d'information sur des sources hétérogènes distribuées. D'un système à l'autre, le poids affecté à un terme peut en effet avoir un sens bien différent.

Bilan sur le modèle vectoriel

Le modèle vectoriel présente l'avantage d'être simple à mettre en œuvre et efficace en temps de traitement. La représentation géométrique en plusieurs dimensions qu'il donne de la requête et des documents est également intéressante, même si pour l'esprit humain, il est difficile de visualiser cette représentation lorsque le nombre de dimensions considéré est supérieur à trois. Le problème est de déterminer avec précision ce qui représente une dimension dans ce modèle. En général, une dimension correspond à un terme ou un concept. Comment, dans ces conditions, traiter les concepts complexes ou composites ? Peut être faudrait-il alors appliquer des théories fractales ? En outre, les seules relations entre termes qu'il est possible de représenter dans ce modèle est leur co-occurrence pour la description d'un document ce qui introduit beaucoup d'ambiguïté. Par exemple, une séquence télévisée montrant un personnage X épinglant une décoration au revers de la veste d'un personnage Y sera décrite, en faisant abstraction des poids, par : { (personnage X), (décoration), (personnage Y)} ; l'information concernant celui qui décore et celui qui est décoré est perdue. Un autre exemple de la faiblesse de représentation de ce modèle concerne l'ajout de qualificatifs à certains termes. Dans le modèle vectoriel, les qualificatifs ne peuvent pas être attachés à un concept précis, mais seulement être inclus en tant que concepts propres dans la requête. À plus forte raison, il n'est pas possible de décrire des relations topologiques entre les éléments pour une image. Nous avons montré que la pondération utilisée pour nuancer les descriptions de documents et de requête peut avoir une sémantique très différente d'un média à l'autre d'une part, et d'un système à l'autre d'autre part. Enfin, ce modèle ne permet pas de structurer la requête, ce qui peut être un grave handicap pour la représentation de requêtes complexes.

2.1.3 Les modèles booléens étendus

De nombreux chercheurs se sont penchés sur le problème de trouver des opérateurs booléens permettant de pondérer les termes de la requête, de façon à conserver les avantages de structuration du modèle booléen simple, tout en donnant la possibilité de classer des documents

d'une part, et d'interpréter les opérateurs de façon plus ou moins stricte d'autre part. En effet, si une collection de documents contient :

$$\begin{aligned} d1 &= (chat, 1), (chien, 1) \\ d2 &= (chat, 1), (chien, 0) \\ d3 &= (chat, 0), (chien, 1) \\ d4 &= (chat, 0), (chien, 0) \end{aligned}$$

une requête $q1 = chat \;\; ET \;\; chien$ traitée par un système booléen traditionnel donnera une valeur de similarité nulle pour les documents $d2$ à $d4$. Or, intuitivement, $d2$ et $d3$ sont *plus proches* de la requête que $d4$: ils y répondent partiellement.

Nous présentons ici le modèle le plus connu : P-norm. D'autres modèles s'appuyant sur la logique floue ont également été développés (voir [Salton 83a]).

Le modèle P-norm

Le modèle booléen étendu P-norm [Salton 83b] subsume le modèle booléen et le modèle vectoriel : il permet d'incorporer des termes pondérés dans la requête et les descriptions de documents, et de classer les résultats par ordre de similarité avec la requête. Il permet aussi d'agréger des termes en des clauses booléennes indépendantes et de pondérer le degré auquel les opérateurs booléens sont interprétés de façon stricte.

L'hypothèse sur laquelle repose ce modèle est que la dissimilarité entre document et requête peut être mesurée par la distance euclidienne normalisée dans l'espace des descripteurs. Ainsi, le modèle de P-norm donnera pour les documents donnés en exemple plus haut les mesures suivantes :

$$\begin{aligned} msim(d1, q1) &= 1 \\ msim(d2, q1) &= 1 - 1/\sqrt{2} \\ msim(d3, q1) &= 1 - 1/\sqrt{2} \\ msim(d4, q1) &= 0 \end{aligned}$$

La notion de *p-norme* peut être vue comme une généralisation de la notion de norme euclidienne. La p-norme d'un document D décrit par n descripteurs dont les poids sont respectivement d_1, d_2, ..., d_n est :

$$||D||_p = (d_1^p + d_2^p + \cdots + d_n^p)^{\frac{1}{p}}.$$

La similarité entre le document D et une requête disjonctive Q dont les clauses disjonctives sont pondérés par $(q_1, q_2, \ldots q_n)$ est définie par :

$$sim(D, Q_{or(p)}) = \left[\frac{q_1^p d_1^p + q_2^p d_2^p + \cdots + q_n^p d_n^p}{q_1^p + q_2^p + \cdots + q_n^p} \right]^{\frac{1}{p}}$$

La disjonction représente les alternatives possibles pour un concept et cette mesure augmente progressivement avec le nombre de termes communs entre requête et documents. La similarité entre le document D et une requête conjonctive Q est définie par :

$$sim(D, Q_{and(p)}) = 1 - \left[\frac{q_1^p(1-d_1)^p + q_2^p(1-d_2)^p + \cdots + q_n^p(1-d_n)^p}{q_1^p + q_2^p + \cdots + q_n^p} \right]^{\frac{1}{p}}$$

La conjonction permet de combiner des concepts pour en former de plus complexes. Intuitivement, cette mesure évalue la distance entre un document et une requête conjonctive ($Q_{and(p)}$), c'est-à-dire composée de tous les termes connectés par des ET ; cette mesure est maximale pour un document qui contient tous les termes de la requête, et décroît avec le nombre de termes communs entre requête et documents. La valeur de p représente le degré de co-occurrence entre les clauses de la conjonction. En faisant varier la valeur de p, il est possible de donner une interprétation différente de chaque clause : une valeur de p proche de l'infini produit une interprétation stricte des relations entre les mots ; le modèle booléen étendu se ramène alors au modèle booléen pur. Des valeurs faibles pour p sont utilisées pour des comparaisons entre documents et requête où les termes sont supposés indépendants les uns des autres. Si $p = 1$, le modèle booléen étendu correspond au modèle vectoriel, puisque les ET sont interprétés comme des OU. Des valeurs intermédiaires fournissent un compromis entre une absence totale de relations entre les termes et une interprétation booléenne stricte. Ainsi, $\mathbf{et}^{(\infty)}$ signifie que le **et** étendu doit être interprété strictement comme un **et** booléen, alors que $\mathbf{et}^{(2)}$ sera pratiquement interprété comme un **ou**. En pratique, dans la plupart des systèmes où ce modèle a été implanté (SMART ou CODER par exemple), la valeur de p varie entre 1,5 et 5.

Le modèle booléen étendu permet de considérer la requête comme composée de sous-requêtes qui s'articulent de façon paramétrée (par la valeur de p) pour former une requête globale. Toutefois, il est complexe et difficile à appréhender pour les utilisateurs. Il nécessite un système de formulation automatique de la requête. Par exemple la requête : *"ouvrages concernant les ordinateurs et la science, de préférence en génie logiciel, et dont l'auteur est exclusivement Dupont"* sera traduite par

$$[(\text{ordinateur } \mathbf{et}^{(2)} \text{ science}) \ \mathbf{or}^{(2)} \ (\text{ordinateur } \mathbf{et}^{(2)} \text{ génie logiciel})] \ \mathbf{et}^{(\infty)} \ (\text{ Dupont})$$

Ainsi, certains termes sont regroupés sémantiquement dans des clauses (**ordinateur** et **science**, **ordinateur** et **génie logiciel**) pour exprimer qu'ils ne sont pas indépendants dans la requête. Dans l'exemple ci-dessus, les termes de la requête ne sont pas pondérés explicitement (de façon implicite, le poids associé à tous les termes est 1). Dans le modèle P-norm, les poids associés aux termes de la requête représentent l'importance relative d'un terme pour le besoin d'information de l'utilisateur. Ainsi, les deux requêtes :

$$\begin{aligned} q2 &= (chat(0,2)ETchien(0,2)) \\ q3 &= (chat(1,0)ETchien(1,0) \end{aligned}$$

donneront la même valeur de similarité pour tous les documents.

Les opérateurs booléens étendus du modèle P-norm ont un grand nombre de «bonnes» propriétés [Lee 94b], à savoir

1. l'idempotence : msim (x,x, ...,x) = x ce qui est important car les trois requêtes

$$t1 \ ET \ t1$$
$$t1 \ OU \ t1$$
$$t1$$

   doivent donner le même résultat ;
2. ils sont strictement monotones sur chaque opérande ; si x, x', y et y' sont des poids :

$$\text{si } x < x' \text{ alors } ET(x,y) < ET(x',y)$$
$$\text{si } y < y' \text{ alors } ET(x,y) < ET(x,y')$$

   ainsi, l'augmentation du poids attaché à un descripteur provoque une augmentation de la valeur de similarité du document, ce qui correspond bien à la définition intuitive de la pondération des termes [8] ;
3. ils sont continus : si un terme voit son poids légèrement augmenter, cela ne doit pas se traduire par une forte augmentation de la mesure de similarité ;
4. ils sont commutatifs : les requêtes $t1 \ ET \ t2$ et $t2 \ ET \ t1$ donnent les même valeurs de similarité pour les mêmes documents.

Par contre, ces opérateurs ne sont pas associatifs, contrairement aux opérateurs booléens simples.

Exprimer toutes les relations de dépendances entre les groupes de termes dans la requête pour former des clauses n'est pas aisé. Lorsqu'il faut, de plus, pondérer les termes de la requête pour indiquer leur importance, la formulation d'une telle requête devient un véritable casse-tête, sauf si elle est assistée par un module expert de formulation automatique. Il semble que ce modèle ait de bonnes performances, comme en attestent les évaluations réalisées sur les collections TREC [Belkin 95]. Il a été implanté dans les systèmes SMART et SIRE, puis dans le système CODER [Fox 87].

2.1.4 Approches alternatives

Utilisation de la théorie des ensembles flous

La théorie des ensembles flous a été développée principalement par Zadeh. Dans le modèle flou de RI, les descripteurs peuvent représenter un élément d'information (un document ou une partie de document) à un certain degré, ce qui est exprimé par une valeur dans [0,1]. Ce poids

8. Ce n'est pas le cas pour d'autres modèles booléens étendus, et notamment pour les modèles inspirés de la théorie des ensembles flous qui utilisent le minimum et le maximum.

reflète l'imprécision de l'information et non son incertitude [Subtil 95]. Les poids peuvent être associés non seulement aux termes d'indexation, mais aussi aux termes de la requête (*poids de pertinence*) pour exprimer le degré auquel ils représentent bien le besoin d'information de l'utilisateur.

Soit $eval()$ la fonction monotone qui associe chaque descripteur à un poids dans [0,1].

Pour des expressions booléennes, les valeurs de recherche sont évaluées par :

- La conjonction de deux constituants de requête $q1$ et $q2$ donne le minimum des deux valeurs de recherche des constituants :
  $eval(d, q1 \quad ET \quad q2) = min(eval(d, q1), eval(d, q2))$
- La disjonction de deux constituants de requête $q1$ et $q2$ donne le maximum des deux valeurs de recherche des constituants :
  $eval(d, q1 \quad OU \quad q2) = max(eval(d, q1), eval(d, q2))$
- La négation d'un terme de la requête donne le complément à 1 de la valeur de recherche qui aurait été obtenue pour le terme non nié :
  $eval(d, NOT \quad q1) = 1 - eval(d, q1)$

Un élément n'est plus soit retrouvé, soit non retrouvé, mais retrouvé avec une certaine valeur. En général, un élément d sera dit retrouvé si sa valeur de recherche dépasse un certain seuil.

Une interprétation possible consiste à considérer les poids des termes de la requête non comme des poids de pertinence mais comme des valeurs de seuils. Étant donné un poids pour un terme de la requête, les éléments d'information sont jugés selon qu'ils atteignent ou dépassent le seuil. Un élément d'information qui atteint exactement la valeur de seuil a le plus haut score. La variation de la valeur de recherche reflète le degré auquel un élément est tombé en-dessous ou a dépassé le seuil. Les seuils n'étant appliqués qu'à des termes isolés, le principe de séparabilité tient. Les fonctions min et max peuvent donc être utilisées pour combiner toutes les valeurs de recherche. Le plus important dans le modèle de seuils est la définition qu'il donne des poids de la requête. Des poids de pertinence faibles dans la requête indiquent que la présence ou l'absence de ces descripteurs particuliers dans l'élément d'information n'est pas très importante (pour le modèle flou). Des poids de seuils faibles dans la requête demandent que ces descripteurs particuliers aient les mêmes poids d'indexations, c'est-à-dire aient la même importance dans les documents que dans la requête. A première vue, cette interprétation des poids est convaincante. Toutefois, pour être consistant avec la signification des poids, les descripteurs absents de la requête mais présents dans les descriptions de documents devraient faire en sorte que le document soit rejeté lui aussi. Dans ces conditions, un document sera rarement retrouvé.

Salton [Salton 83b] reproche au modèle flou de manquer de discrimination parmi les éléments retrouvés, pratiquement au même degré que les SRI booléens traditionnels, puisque le rang d'un élément retrouvé dépend seulement du terme ayant le plus faible (respectivement le plus fort) poids pour une requête ET (respectivement OU).

Pour nous, cette approche est intéressante lorsque l'on s'intéresse à une collection de documents figée pour des critères bien particuliers. Il reste à déterminer ce qu'elle permet d'obtenir

pour adresser une requête à des collections de documents hétérogènes.

Nous ne développons pas plus avant cette approche, le lecteur intéressé pouvant se reporter aux travaux réalisés par ailleurs dans l'équipe EXPRIM, par exemple [Subtil 95].

Les modèles probabilistes

Il y aurait beaucoup à dire sur les modèles probabilistes, sur lesquels reposent de nombreux travaux en RI. Nous n'en donnerons ici que les idées principales.

Dans ces modèles, on cherche les documents qui ont une forte probabilité d'être pertinents et une faible probabilité d'être non pertinents [Salton 85] vis-à-vis de la requête. L'hypothèse sous-jacente est que les termes sont distribués différemment dans les documents pertinents et dans les documents non pertinents.

Le premier modèle probabiliste (de Maron et Kuhns, 1960 [Thompson 89]) reposait sur une théorie probabiliste de l'indexation : étant donné un document D, l'indexeur devait estimer, pour chaque terme quelle était la probabilité qu'un utilisateur, qui juge le document pertinent, choisisse ce terme pour exprimer sa requête, en supposant qu'il n'utilise qu'un seul terme.

Le deuxième modèle probabiliste (de Robertson et Sparck-Jones,1976 [Thompson 89]) s'est appuyé sur une théorie probabiliste de la recherche où les termes d'indexation étaient attribués sans poids. Seuls les termes de la requête étaient pondérés, à partir du bouclage de pertinence (principe que nous examinons au § 2.3) : c'est le bouclage de pertinence qui permet de calculer une probabilité de pertinence pour chaque terme en fonction de sa distribution parmi les documents pertinents et les documents non pertinents. En supposant que des mots non liés sémantiquement sont distribués également parmi les textes, la probabilité pour un mot de se retrouver un certain nombre de fois dans un élément d'information spécifique suit une distribution de Poisson. L'ensemble des documents jugés pertinents par l'utilisateur d'une part et celui des termes jugés non pertinents d'autre part donnent donc deux distributions de Poisson. L'hypothèse principale du modèle de 2-Poisson est que les mots en rapport avec le besoin d'information se retrouvent plus fréquemment dans les éléments pertinents que dans les non pertinents. Les paramètres de ces deux distributions sont choisis pour pondérer les termes d'indexation. Ces probabilités sont propres à un utilisateur particulier et à une classe de documents donnés.

Robertson, Maron et Cooper [Robertson 82] ont unifié les deux premiers modèles, et utilisé un thésaurus pour l'indexation des documents et pour la formulation de la requête. Une pondération éventuelle des termes à l'indexation représente la fréquence des mots dans le texte.

Soit R l'événement correspondant au fait que le document est pertinent pour le besoin d'information courant. La probabilité pour un document D d'être pertinent vis-à-vis d'une requête Q est ainsi représentée par $P(R|Q, D)$ (probabilité de pertinence connaissant la requête Q et le document D). La chance qu'un événement y se produise est estimée par : $C(y) = P(y)/P(\overline{y})$, où $P(\overline{y}) = 1 - P(y)$. Si p_{ik} représente la probabilité de l'événement x_i «le terme t_i est pertinent pour la requête Q_k», $p_{ik} = P(x_i = 1|R, Q_k)$ et $q_{ik} = P(x_i = 1|\overline{R}, q_k)$, alors les valeurs de similarités (ou RSV, Retrieval Status Values) pour les termes t_i communs à la requête et aux

documents sont données par

$$sim(D,Q) = \sum_{t_i \in D \cap Q} log \frac{p_{ik}(1-q_{ik})}{q_{ik}(1-p_{ik})}.$$

Cette mesure permet de classer les documents par ordre de pertinence décroissante. Les paramètres p_{ik} et q_{ik} peuvent être estimés à partir des jugements de pertinence des utilisateurs. Ainsi, si r représente le nombre de documents jugés pertinents par l'utilisateur, dont r_i contiennent le terme t_i, $p_{ik} \approx r_i/r$. Si f représente le nombre de documents présentés à l'utilisateur, dont f_i contiennent le terme t_i, $q_{ik} \approx (f_i - r_i)/(f - r)$ [Fuhr 92].

Les jugements de pertinence étant propres à un utilisateur particulier, l'apprentissage effectué à partir de ces données n'est valable que pour cet utilisateur, à moins que les mêmes requêtes ne soient soumises un grand nombre de fois par des utilisateurs différents.

Le modèle de réseau bayésien d'inférence

Ce modèle utilisé en Intelligence Artificielle a été également introduit en recherche d'information. Un réseau d'inférence bayésien est un graphe orienté sans cycles. Dans ce graphe, les nœuds représentent des *variables propositionnelles* (ou également des constantes) ; les arcs représentent des relations de dépendance entre ces nœuds ou *propositions*. La recherche ici est vue comme un processus de raisonnement incertain où plusieurs sources sont combinées pour estimer la probabilité qu'un document donné satisfasse la requête. Si une proposition représentée par un nœud p cause ou implique la proposition représentée par le nœud q, un arc relie p et q. Des probabilités ou des *croyances* sont associées à chaque nœud [Thompson 89]. Pour les applications de RI, les nœuds du graphe peuvent correspondre à des concepts, à des groupes de termes, ou à des documents. Un nœud particulier représente la requête.

Initialement, tous les documents sont équiprobables : les nœuds documents se voient attribuer une probabilité initiale de $1/N$, où N est la taille de la collection. Chaque nœud de représentation contient la spécification d'une probabilité conditionnelle de pertinence, étant donné l'ensemble des nœuds parents.

Ce modèle repose sur le théorème de Bayes qui permet de calculer la probabilité conditionnelle d'un événement a, connaissant un événement b :

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)}$$

Dans cette formule, a représente une hypothèse et b un fait observé. En RI, on calcule la probabilité de pertinence pour qu'une requête q soit satisfaite connaissant un document, par $P(q|d)$, où la représentation d'un document d est considérée comme certaine.

Dans le modèle de réseau bayésien d'inférence du système I^3R (qui peut être considéré comme un des premiers systèmes multi-agents pour la RI), la stratégie d'activation-propagation utilisée est très représentative du fonctionnement du réseau pour une recherche d'information :

1. Certains nœuds sont identifiés comme points de départ de la recherche, en général grâce à une requête.

2. Tous les nœuds connectés à ces points d'entrée par un lien unique sont activés. Les liens peuvent représenter par exemple des citations ou des co-citations.
3. Le processus de propagation de l'activation des nœuds est répété jusqu'à ce qu'une condition d'arrêt soit satisfaite. Le niveau d'activation décroît avec la longueur du chemin parcouru lors de l'activation, et la recherche se termine quand ce niveau tombe en dessous d'un certain seuil.
4. Les nœuds sont extraits par ordre de leur niveau d'activation.

Des variantes de cette stratégie ont été élaborées, par exemple pour tenir compte du type de lien entre les nœuds, ou pour que la fonction du niveau d'activation tienne compte non seulement du niveau d'activation d'origine, mais aussi de la force du lien, etc. Pour une comparaison du modèle de réseau bayésien d'inférence avec les autres modèles, on peut se référer à [Turtle 92].

Ce modèle peut être combiné avec une recherche basée sur des clusters et l'utilisation d'un thésaurus. Il a été implanté dans les systèmes I^3R et INQUERY [Croft 92a, Croft 92b, Belkin 95], ou encore dans l'interface prototype OFFICER [Croft 90b] qui permet l'accès à une base de données de documents dans un environnement de bureau.

Les reproches qui sont fait à ces modèles sont liés d'une part au problème de l'indépendance des événements pour lesquels on calcule des probabilités, et d'autre part au fait que la représentation d'un document d est sujette à incertitude.

Révision de croyances

C.J. van Rijsbergen [Rijsbergen 92] a proposé de revisiter le modèle probabiliste en s'appuyant sur la théorie de révision de croyance de Dempster-Shafer [Shafer 76].

Selon lui, les probabilités peuvent être vues de deux manières différentes : la mesure de la chance d'un événement ou le degré de croyance d'une proposition. Si un document traite d'un sujet X avec une probabilité α, il peut ou non contenir l'information X. Cette incertitude est représentée par des croyances. Une fonction de croyance $Bel(R, q, d)$ mesure le degré auquel le document d est cru pertinent dans le contexte q, R représentant une relation de pertinence. Le problème est qu'il faudrait demander aux utilisateurs à quel point ils croient qu'un terme représente leur besoin, ce qu'ils sont incapables de faire en général [Croft 90a].

2.1.5 Thésaurus pour la recherche d'information

Nous avons présenté précédemment (voir § 1.3.2) les sources de connaissances importantes pour les SRI que constituent les thésaurus. Certains algorithmes de mise en correspondance s'appuient sur cette structure pour calculer une *distance sémantique* entre un document et une requête.

Dans les approches que nous allons présenter, c'est la connaissance représentée par des nœuds et les arcs du graphe que constitue le thésaurus qui est déterminante pour calculer la

distance entre les termes de la requête et ceux qui décrivent le document. La mise en correspondance s'apparente alors à un mécanisme d'activation-propagation dans un réseau sémantique.

Prise en compte du thésaurus pour le calcul de la pertinence système

Les premiers systèmes sont apparus avec le modèle booléen simple et travaillaient à partir d'une requête exprimée sous forme d'une expression booléenne en forme normale disjonctive : les termes sont regroupés en expressions conjonctives pouvant contenir des négations, les expressions conjonctives étant elles-mêmes regroupées en disjonctions. Pour la recherche dans des textes, où les termes d'indexation peuvent se trouver à un niveau quelconque du thésaurus, ces algorithmes calculent la distance entre deux termes du thésaurus $distance(T, T_i)$ par le nombre minimal de relations *is_a* à traverser dans le thésaurus pour aller d'un terme à l'autre. Si les descripteurs de documents sont limités aux feuilles du thésaurus, l'utilisateur n'est pas contraint à utiliser le vocabulaire d'indexation pour formuler sa requête. Il est donc justifié de chercher à mesurer la distance entre deux termes sémantiquement liés plutôt que de les traiter comme des termes indépendants.

L'algorithme de pertinence de Rada & Humphrey calcule le degré de pertinence (appelé $relevance(q_i, d_j)$) entre un terme q_i décrivant la requête et un descripteur de document d_j par une mesure inversement proportionnelle à la distance qui les sépare dans le thésaurus :

$$\begin{aligned} relevance(q_i, d_j) &= \frac{\alpha}{1 + distance(q_i, d_j)} \\ \text{où } \alpha &= \begin{cases} 1 & \text{si } q_i \text{ est } d_j, \\ -1 & \text{si } q_i \text{ est } NOT\ d_j. \end{cases} \end{aligned}$$

Par exemple, si les termes `animal` et `boeuf` sont reliés dans le thésaurus par le chemin : `boeuf` $\leq$ `bovin` $\leq$ `bovidé` $\leq$ `mammifère-terrestre` $\leq$ `mammifère` $\leq$ `animal`,

$$relevance(animal, boeuf) = 1/6$$

Pour une requête conjonctive $Conj_i = q_{i_1}\ ET\ q_{i_2}\ ET\ \ldots\ ET\ q_{i_{m_i}}$ comportant m_i termes, la pertinence d'un document D décrit par n termes d_k est mesurée par :

$$R(Conj_i, D) = \frac{\sum_{j=1}^{m_i} \sum_{k=1}^{n} relevance(q_{ij}, d_k)}{min(m_i, n) + \frac{1}{2}(m_i.n - min(m_i, n))}.$$

Soit D_1 un document décrit par `boeuf`, `chêne`, `prairie`. Soit la requête q_1 décrite par les termes `animal` et `saule`. Pour chaque terme de la requête, il faut calculer sa distance avec tous les termes du document. Ainsi,

$$R((boeuf\ \ ET\ \ saule), D) = \frac{\sum_{j=1}^{2} \sum_{k=1}^{n} relevance(q_{ij}, d_k)}{min(m_i, n) + \frac{1}{2}(m_i.n - min(m_i, n))} = 1,8.$$

Enfin, la proximité d'un document D par rapport à une requête Q en forme normale disjonctive (composée de p conjonctions $Conj_i$) se mesure par le maximum des mesures obtenues pour chaque expression conjonctive :

$$R(Q, D) = R(Con_1\ OR\ \ldots\ OR\ Con_p, D) = max_{i=1\ldots p}(R(Con_i, D))$$

Cet algorithme a été amélioré par l'**algorithme de R-distance** [McMath 89] et étendu par l'algorithme **K-distance** [Kim 90] qui permet d'attacher des poids aux arcs du thésaurus et aux termes de la requête et du document.

L'**algorithme de E-Pertinence** [Lee 94c] intègre le modèle booléen étendu et l'algorithme de pertinence. Les auteurs constatent que, dans l'algorithme de pertinence, (1) l'utilisation des fonctions min ou max pour évaluer l'opérateur OR détériore l'efficacité de la recherche, (2) la transformation des requêtes en forme normale disjonctive augmente la complexité des calculs, (3) la négation est traitée très inefficacement, cette inefficacité croissant avec la taille du thésaurus. Par ailleurs, la formule du modèle booléen étendu n'utilise pas la dépendance qui peut exister entre les termes du document D. Avec l'algorithme de E-pertinence, une mesure de similarité est calculée entre un terme T et un document D :

$$\begin{aligned} sim(T,D) &= \frac{\sum_{i=1}^{n}(relevance(T,T_i).d_i)}{1+\frac{1}{2}(n-1)} \\ \text{où } n &= \text{nombre de termes indexant } D \\ d_i &= \text{poids du terme } T_i \text{ pour } d. \end{aligned}$$

Cette mesure de similarité est utilisée pour affecter des poids aux termes dans le modèle booléen étendu.

À noter que le nombre de termes indexant un document joue sur la mesure de pertinence calculée. Ainsi, un document D_2 décrit par les mêmes termes que D_1, mais avec en plus le terme `ruisseau` aura une mesure de pertinence inférieure à $D1$. Par ailleurs, le problème se pose de comparer des termes qui n'appartiennent pas au même domaine sémantique, et qui ne sont donc pas reliés, directement ou transitivement, par la relation *is_a*. Dans l'exemple ci-dessus, la distance entre des termes de champs sémantiques différents a été considérée comme maximale (1). Pour être exact, il faudrait prendre en compte les relations d'association *voir aussi* dans ce calcul ; par exemple, `vache` et `lait` sont moins éloignés l'un de l'autre que `vache` et `placard` ...

Le modèle de recherche de document présenté dans [Lu 90] repose aussi sur une analyse lexicale sémantique, et sur un algorithme qui mesure une distance entre arbres. Dans ce modèle, documents et requêtes sont codés sous forme d'une structure qui prend en compte toutes les hiérarchies ou ordres entre les termes ou *coordonnées conceptuelles*. La mise en correspondance est réalisée par un algorithme qui repose sur des opérations de construction et de mise à jour d'arbres.

L'auteur propose en effet de représenter documents et requêtes par des arbres dont les nœuds sont des concepts et les arcs des relations sémantiques entre les concepts. Les relations utilisées sont celles que l'on trouve communément dans un thésaurus (voir § 1.3.2 page 16) : taxonomie, partie-tout, diverses associations d'idées. Toutefois, les auteurs regroupent la relation partie-tout et la relation *membre d'une classe* ou *d'un groupe*, ainsi que certaines relations d'associations comme *matière* ou *origine*. Les types de relations qui ne correspondent pas naturellement à des

hiérarchies, comme les relations d'association d'idées, sont représentés artificiellement par un arbre à deux niveaux, dont le concept est la racine et les termes associés sont les feuilles.

Un type particulier de relations, appelées *hiérarchie sans branchement* (*non-branching hierarchy*), recouvre les chaînes (exemple : épaule-bras-coude-avant-bras), les cycles ou hélices (exemple : printemps-été-automne-hiver), les termes de classement (exemple : simple-double-triple), les graduations (exemple : petit - gros - énorme), les degrés (échec, moyenne, honneur, distinction). Ces relations sont représentées par des chaînes de concepts verticales.

Dans ce modèle, un document D (ou une requête Q) est représenté par des listes de termes, ces termes pouvant être isolés ou dotés de leur contexte sémantique c'est-à-dire du détail des relations sémantiques du thésaurus dans lesquelles ils sont impliqués :

$$D = \{\{S_1^t, \ldots, S_x^t\}\,;\{S_1^c, \ldots, S_y^c\}\,;\{S_1^p, \ldots, S_z^p\}\}$$
$$\text{avec } 0 \leq x, y, z \leq n\,; x + y + z > 0$$

où S_i^t est une relation de taxonomie, partie-tout ou terme associé, S_j^c est une relation de hiérarchie sans branchement, et S_k^p un concept isolé sans relation lexico-sémantique. Concrètement, tous les termes du document (ou de la requête) sont examinés deux à deux ; s'ils appartiennent à une même structure sémantique correspondant à l'une des relations présentées plus haut, le nouveau terme est ajouté à la structure d'arbre déjà existante, sinon il devient une nouvelle structure arborescente.

Le mise en correspondance d'un document et une requête mesure leur *similarité structurelle* par le coût total minimal des transformations nécessaires pour obtenir l'arbre-requête T' à partir de l'arbre-document T. Ce coût est évalué par une fonction δ.

Si X_i et Y_i sont le nombre de structures d'arbres de type i représentant un document D et une requête Q, D et Q contenant alors respectivement $\sum X_i$ et $\sum Y_i$ structures d'arbres, le coût $\delta(D, Q)$ est calculé selon le nombre d'insertions ou de suppressions nécessaires pour passer d'un arbre à l'autre. Ce coût permet de mesurer la similarité $S(D, Q)$ entre document et requête par :

$$S(D, Q) = 1 - \frac{\delta(D, Q)}{\delta(D, Q)_{max}}$$

S prend ses valeurs dans l'intervalle $[0, 1]$: S vaut 1 si $D = Q$, 0 si D et Q sont totalement différents.

Lors de la formulation de la requête, le chercheur d'information sélectionne lui-même quelles sont les catégories de relations sémantiques qui sont pertinentes pour un terme, en fonction du contexte du sujet de sa recherche. Cette méthode, expérimentée sur une petite base de documents, semble donner de meilleurs résultats (rappel et précision) que le modèle vectoriel. L'originalité de cette méthode est la prise en compte de nombreuses relations sémantiques, comme les relations de co-taxonomie[9], pour mesurer la similarité des documents et de la requête. Toutefois, les auteurs pensent qu'il serait plus significatif de distinguer les relations sémantiques hiérarchiques « pures» (par exemple de taxonomie ou partie-tout) des relations d'associations d'idées afin de représenter ces dernières par un réseau et non un arbre.

9. La co-taxonomie est la relation qui lie deux termes qui ont le même générique direct.

Aide à la formulation de la requête

Le thésaurus peut apporter également une aide précieuse lors de la formulation de la requête : il permet à un utilisateur dont le besoin est mal défini de naviguer de façon descendante dans la hiérarchie de termes pour affiner incrémentalement l'expression de son besoin. Ce principe a été implanté notamment dans RIVAGE [Halin 89] et dans le système HELGON [Fischer 89], où à chaque étape de la navigation dans le thésaurus, des exemples de documents sont présentés à l'utilisateur.

Bilan sur l'utilisation d'un thésaurus

Les thésaurus sont des bases de connaissances très utiles pour la RI. Ils permettent d'une part de donner à l'utilisateur un retour conceptuel sur les connaissances détenues par le système, d'autre part d'augmenter le vocabulaire de requête par des termes non directement utilisés dans l'indexation des documents. Toutefois, faire reposer l'inférence sur de telles structures présente plusieurs inconvénients :

- Si l'inférence réalisée durant la recherche repose uniquement sur les thésaurus, d'une part elle peut être difficilement transposable dans un nouveau contexte, l'élaboration d'un nouveau thésaurus étant une tâche difficile et coûteuse (particulièrement en temps). Cette caractéristique de l'architecture peut empêcher l'ouverture du SRI à de nouveaux thèmes d'application.
- Un thésaurus n'est en aucun cas une vérité universelle ; il s'agit d'une base de connaissances adaptée à un domaine particulier où les connaissances sont structurées en fonction d'un contexte applicatif ; certaines inférences réalisées peuvent donc tenir plus d'heuristiques que de mécanismes de déduction proprement dits.
- Dans un univers distribué, le partage de connaissances entre des thésaurus est un problème encore ouvert.

Nous ne rejetons pas pour autant l'utilisation d'un thésaurus pour un SRI, mais les inférences d'un système ne peuvent reposer exclusivement sur ce type particulier de connaissances.

2.2 Le modèle logique

Nous avons présenté jusqu'ici différentes approches pour la modélisation d'un système de recherche d'information. Les éléments communs à ces modèles sont : un modèle de requête, un modèle de description des documents et un modèle de mise en correspondance entre les deux. Les travaux que nous abordons à présent donnent des éléments complémentaires pour définir un modèle de recherche d'information.

2.2.1 Principes du modèle logique

Van Rijsbergen [Rijsbergen 86b, Rijsbergen 86a], après avoir observé que les recherches en RI piétinaient quelque peu au début des années 80, définit les éléments nécessaires à un modèle de recherche d'information. En particulier, il s'intéresse à la relation calculée entre les documents et la requête lors d'une recherche. Il propose d'utiliser la logique pour modéliser la mise en correspondance entre requêtes et documents. Un document et une requête sont vus comme un ensemble de propositions qui sont interprétées dans une sémantique prédéfinie. La primitive unique pour la RI est une opération d'implication incertaine : pour qu'un document réponde à une requête, il doit impliquer logiquement la requête. $D \rightarrow Q$ est vrai si le fait de retrouver le document D satisfait la requête Q. Cette implication n'est pas stricte car un document n'implique jamais strictement une requête, mais plausible, c'est pourquoi il lui est associé une **force** ou **mesure d'incertitude**. Pour C.J. van Rijsbergen, l'implication n'est pas celle de la logique classique mais une évaluation conditionnelle :

$$P(D \rightarrow Q) = P(D|Q) \tag{2.3}$$

Si D et Q sont représentés par un ensemble de termes d'indexation indépendants non pondérés, cette implication peut être interprétée par

$$P(D \rightarrow Q) = \frac{|D \cap Q|}{|Q|}.$$

Exemple :

Q	= (chat, extérieur, jardin)	
$D1$	= (chat, étagères, fenêtre, intérieur)	$P(D1 \rightarrow Q) = 1/3$
$D2$	= (arbre, chat, extérieur, fleurs)	$P(D2 \rightarrow Q) = 2/3$

Si $P(D \rightarrow Q) = 1$, alors le document D contient tous les termes de la requête Q.

Cette mesure ne permet pas de départager un document qui aborderait de façon détaillée les sujets souhaités (et exprimés dans sa requête par le demandeur) d'un document plus général. Elle exprime seulement le degré auquel la requête est satisfaite par le document ou une partie du document. Nie le nomme **critère d'exhaustivité**.

Exemple :

Q	= (chat, jardin)	
$D1$	= (chat, jardin)	$P(D1 \rightarrow Q) = 2/2 = 1$
$D2$	= (oiseau, chien, chat, fleurs, jardin)	$P(D2 \rightarrow Q) = 2/2 = 1$

$D1$ et $D2$ ne sont pas départagés par la mesure d'exhaustivité.

C'est pourquoi Nie ([Nie 88]) introduit le **critère de spécificité**, pour mesurer la force de l'implication réciproque $P(Q \rightarrow D)$. Si $P(Q \rightarrow D) = 1$, alors le document D ne concerne que

la requête Q. Le critère de spécificité permet de mesurer le degré auquel une requête (ou une partie de la requête) satisfait un document.

Exemple :

Q	= (chat, jardin)	
$D1$	= (chat, jardin)	$P(Q \rightarrow D1) = 2/2 = 1$
$D2$	= (oiseau, chien, chat, fleurs, jardin)	$P(Q \rightarrow D2) = 2/5 = 0,4$

$D1$ est plus spécifique que $D2$ par rapport à la requête.

La mesure de correspondance entre une requête Q et un document D peut alors être exprimée par une fonction compromis F entre les deux implications :

$$R(D, Q) = F(P(D \rightarrow Q), P(Q \rightarrow D)) \tag{2.4}$$

Par exemple, une requête sur les accidents peut être satisfaite par un document sur les accidents ferroviaires, alors qu'une requête sur les accidents ferroviaires ne sera pas souvent satisfaite par des documents sur les accidents en général. La connaissance utilisée lors de l'évaluation de ces deux implications est donc différente.

Si la représentation n'est pas la même pour les documents et la requête, par exemple si les termes de la requête sont pondérés alors que ceux qui décrivent les documents ne le sont pas, la formule devient :

$$R(D, Q) = F(P(D \rightarrow Q), P'(Q \rightarrow D)).$$

La définition des fonctions P, P' et F dépend beaucoup des caractéristiques du système et des besoins de l'utilisateur.

Nous pouvons remarquer que la mesure de spécificité n'a de sens qu'en complément de la mesure d'exhaustivité. En effet, si elle est utilisée seule, elle mesure uniquement l'aptitude d'un utilisateur à formuler sa requête exactement dans les mêmes termes que ceux utilisés par l'indexeur pour décrire le document, ce qui n'a pas vraiment de sens pour une recherche d'information en général.

2.2.2 *À-propos* et principes d'incertitude

Pour évaluer $P(D \rightarrow Q)$, C. J. van Rijsbergen se situe dans le contexte de la logique modale et introduit la notion de *proximité* entre les mondes possibles que constituent les documents. Si l'implication n'est pas vérifiée, il faut changer de monde et évaluer l'implication dans ce

nouveau monde, le changement de monde devant être minimal. C'est ainsi qu'est introduit le principe général d'incertitude [Rijsbergen 86a] :

> *"Étant donné deux propositions x et y, une mesure de l'incertitude de $y \rightarrow x$ relativement à un ensemble d'informations est déterminée par le degré minimal auquel de l'information doit être ajoutée à cet ensemble d'informations pour établir la vérité de $y \rightarrow x$."*

Dans la proposition initiale de van Rijsbergen, la dérivation du monde durant l'évaluation de l'implication est considérée comme l'extension de l'environnement pour correspondre à la proposition. Mais elle n'indique pas ce que recouvre l'«ensemble d'informations». Nie propose de la modifier comme suit :

> *"Étant donné deux ensembles d'informations x et y, une mesure de l'incertitude de $y \rightarrow x$ relativement à un ensemble de connaissances K, est déterminée par l'extension minimale E qu'il faut ajouter à l'information y pour établir la vérité de $(y + E) \rightarrow x$."*

C'est le premier principe d'incertitude. Dans ce principe, K et E sont liés à K de la manière suivante :

- E ne peut pas être un sous-ensemble de K, car K étant défini dans le système, cette connaissance est disponible dans tout monde possible. Ainsi, si Q n'est pas satisfaite dans un monde possible D_i, la connaissance supplémentaire nécessaire doit être trouvée en dehors de K.
- E peut être inférée à partir de K ; dans ce cas, la considération de E n'augmente pas l'incertitude (ou ne diminue pas la certitude) de la vérité de $y \rightarrow x$.
- E peut être indépendante de K, c'est-à-dire non inférée à partir des connaissances du système ; c'est le cas de l'ajout pur d'information qui augmente l'incertitude de la vérité de $y \rightarrow x$. Par exemple, si un document est décrit par les termes "chat" et "jardin", et que la requête comporte les termes "chat", "jardin" et "soleil", l'ajout du terme "soleil" à l'ensemble des connaissances sur le document augmente l'incertitude de correspondance, car rien n'indique, dans les connaissances du système, que le document traite aussi de "soleil".

Cela lui permet d'énoncer le principe d'évaluation de l'implication suivant :

> *"Pour deux ensembles d'information A et B, une mesure d'incertitude de $A \rightarrow B$ relativement à un ensemble de connaissances est déterminée par la diminution minimale de B à B' pour établir la vérité de $A \rightarrow B'$."*

Le second principe d'incertitude considère les documents comme des mondes possibles, et propose d'agir par transformations successives des mondes-documents pour correspondre à la proposition (requête) :

> *Étant donné deux ensembles d'informations D et Q ; une mesure de l'incertitude de $D \rightarrow Q$ relativement à un ensemble de connaissances donné K, est déterminée par la transformation nécessaire de D en D' et la certitude de la vérification de $D' \rightarrow Q$.*

Nie a analysé de façon détaillée l'approche réciproque, qui consiste à réduire la proposition (la requête) pour correspondre à l'environnement (les documents). Si l'ensemble des propositions est considéré comme un monde, pour évaluer $D \rightarrow Q$, Q est progressivement changé en Q_1 puis $Q_2, \ldots$ jusqu'à Q_n qui satisfait totalement D ; l'incertitude de l'implication initiale peut être mesurée par la distance entre Q et Q_n. C'est le troisième principe d'incertitude :

> *Étant donné deux ensembles d'informations D et Q ; une mesure de l'incertitude de $D \rightarrow Q$ relativement à un ensemble de connaissances donné K, est déterminée par la transformation nécessaire de Q en Q' et la certitude de la vérification de $D \rightarrow Q'$.*

Le fait de transformer des connaissances par généralisation ou spécialisation des termes grâce à un thésaurus peut être une interprétation des transformations successives évoquées dans les principes d'incertitude de C. K. van Rijsbergen et Nie (§ 2.2.2 page 50).

De nombreux travaux s'appuient désormais sur le modèle logique, qui pose les premières pierres d'un modèle formel pour la recherche d'information. Le modèle logique s'avère suffisamment puissant pour exprimer la plupart des modèles de mise en correspondance existants. Toutefois, les interprétations et les représentations de ce modèle sont très diverses. Par exemple, pour Crestani, chaque terme est un monde possible.

2.2.3 Comparaison du modèle logique avec les précédents modèles présentés

Puisque le modèle logique ne donne que des principes et non des contraintes de représentation de ces principes, il est légitime d'analyser les principaux modèles présentés au début de ce chapitre du point de vue de ce méta-modèle.

Modèle logique et modèle booléen

Dans le modèle booléen classique, un document est retrouvé si les termes du document satisfont l'expression booléenne de la requête. Ceci correspond au critère d'exhaustivité du modèle logique.

$$R(D, Q) = P(D \rightarrow Q)$$

On peut considérer D et Q comme des expressions booléennes ; dans ce cas, l'implication logique $D \rightarrow Q$ est fausse si et seulement si D est vrai et Q est faux.

Cette interprétation de l'implication pose un problème. Si D est vrai (un document D est considéré comme vrai quand sa description est connue), la pertinence est estimée en fonction de la satisfaction de la requête. Si une description de document est fausse, cela signifie que le contenu du document est inconnu, et donc que le document n'est pas retrouvé. Ainsi, tout document non retrouvé peut amener à vérifier l'implication logique. Ceci ne correspond pas vraiment à l'idée intuitive que l'on se fait de la pertinence ! En fait, la procédure d'évaluation de l'implication est la suivante : on suppose D vrai, puis on vérifie la vérité de Q sous cette hypothèse. Ceci correspond plutôt à l'inférence logique qu'à l'implication matérielle, c'est-à-dire vérifier Q en présence de D vrai. Nie conclut que : (1) l'implication matérielle ne convient pas pour la modélisation de la pertinence, (2) le modèle booléen utilisé en RI ne repose pas seulement sur l'implication matérielle.

Modèle logique et utilisation des relations sémantiques d'un thésaurus

Y. Chiaramella et J.P. Chevallet [Chiaramella 92] ont utilisé les principes du modèle logique pour analyser l'utilisation de connaissances sur les relations sémantiques entre les termes. La connaissance utilisée est la relation spécifique/générique du thésaurus qui relie chaque terme à un terme plus général. La notion d'à propos (*aboutness*) correspond assez bien à la notion de modèle en logique [Bruza 94] : si un document D est à propos d'un concept t, on peut écrire $D \models t$. Remplacer un terme de la requête par son générique revient à appliquer l'implication *spécifique* $\rightarrow$ *générique* (notée $(t_{spe} \supset t_{gen})$) et la règle du modus ponens :

$$\frac{D \models t_{spe} \ ; \quad \models (t_{spe} \supset t_{gen})}{D \models t_{gen}} \tag{2.5}$$

Si un terme spécifique t_{spe} décrit un document D et qu'une source de connaissance (le thésaurus) permet de déduire le terme plus générique t_{gen}, alors on peut déduire que t_{gen} décrit aussi le document. Bruza [Bruza 94] considère que la relation *à propos*, qu'il note $i \rightarrow j$, est une relation d'ordre partielle, dans laquelle $i < j$; en effet, i est un terme plus spécifique que j s'il contient plus d'information que j. Par exemple, si un texte parle d'un `oiseau` sur une branche, il nous donne moins d'information que s'il précise qu'il s'agit d'une `mésange`.

Ceci justifie l'indexation des documents par les termes les plus spécifiques possible, puisque le système pourra toujours *déduire* un terme plus générique ; par ailleurs, les relations sémantiques entre les termes peuvent évoluer sans remettre en cause l'indexation des documents.

Modèle logique et modèle vectoriel

Dans le modèle vectoriel, un document D est représenté par un vecteur $\vec{D} = (d_1, d_2, ..., d_n)$, où d_i est le poids du terme i pour le document D. Si nous posons, pour les mesures d'exhaustivité

exh et de spécificité spe,

$$exh(D,Q) = \frac{\sum_j d_{ij}.q_j}{\sum_j q_j} \quad et \quad spe(D,Q) = \frac{\sum_j d_{ij}.q_j}{\sum_j d_{ij}}$$

le coefficient de Jaccard peut être exprimé par :

$$\begin{aligned} R_{Jacc}(D_i,Q) &= \frac{\sum_j d_{ij}.q_j}{\sum_j d_{ij} + \sum_j q_j - \sum_j d_{ij}.q_j} \\ &= \frac{exh \times spe}{exh + spe - exh \times spe} \end{aligned}$$

Le coefficient de Dice correspond lui à :

$$R_{Dice}(D_i,Q) = \frac{2\sum_j (d_{ij}.q_j)}{\sum_j d_{ij} + \sum_j q_j} = 2 \times \frac{exh \times spe}{exh + spe}$$

Quant à la mesure de cosinus, elle pourrait correspondre à :

$$R_{cos}(D_i,Q) = \frac{\sum_j (d_{ij}.q_j)}{\sqrt{\sum_j (d_{ij})^2 . \sum_j (q_j)^2}} = exh \times spe$$

en choisissant

$$exh = \sqrt{\frac{\sum_j d_{ij}.q_j}{\sum_j q_j^2}} \quad et \quad spe = \sqrt{\frac{\sum_j d_{ij}.q_j}{\sum_j d_{ij}^2}}. \tag{2.6}$$

Nous pouvons constater que les mesures classiques de similarité utilisées dans le modèle vectoriel donnent la même importance aux mesures d'exhaustivité et de spécificité.

2.2.4 Comparaison inductive de modèles de RI

La recherche d'information comporte une notion-pivot que nous appelons *relation d'à-propos* pour traduire le terme anglais *aboutness* utilisé par van Rijsbergen. Cette notion prend de nombreuses formes différentes. Cette relation peut en effet lier un document et une requête, mais aussi des documents entre eux, les documents liés sont alors réunis dans des clusters. L'indexation des documents pour un SRI tente de définir les relations d'à-propos entre un document et des termes descripteurs. Par ailleurs, des structures de connaissance comme les thésaurus permettent de représenter des relations d'à-propos entre des termes. Il existe donc plusieurs niveaux différents de relations d'*à-propos* en RI : le niveau concept-concept, le niveau concept-document, et le niveau document-document (par exemple dans les systèmes hypermédias), et le niveau document-requête (figure 2.2).

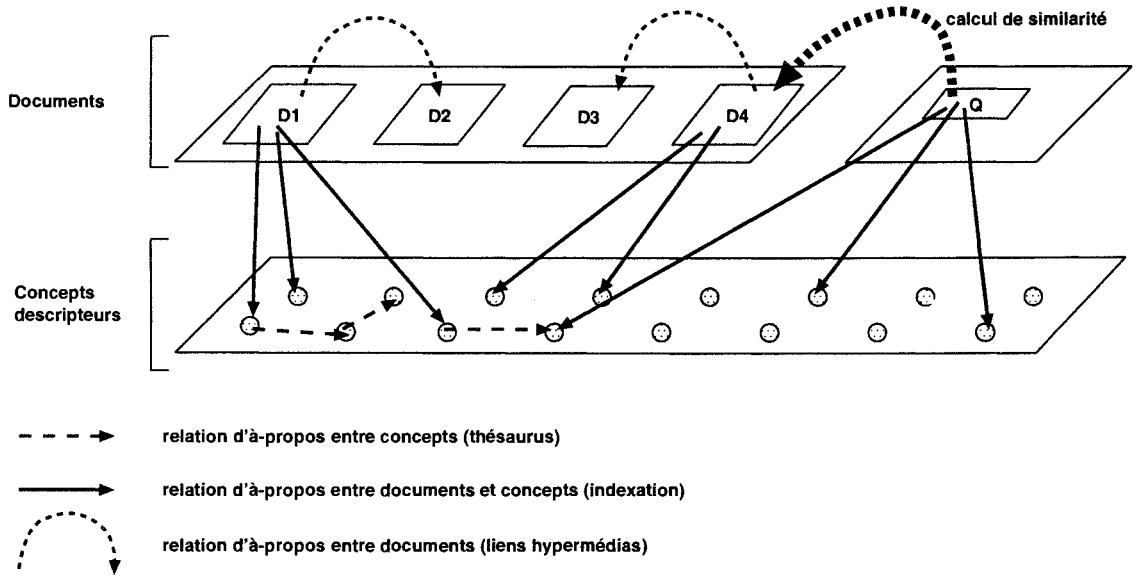


FIG. 2.2 – *Les différents types de relations d'à-propos.*

Bruza [Bruza 94], utilise la notion de «porteur d'information» (*information carrier* en anglais) pour abstraire les notions de descripteurs, documents, et requêtes, c'est-à-dire tous les types d'objets qui sont susceptibles d'être impliqués dans la relation à-propos. Ces travaux s'appuient sur la *théorie des situations*, développée par Barwise et Perry [Barwise 83] pour modéliser la notion de signification ou de sens en langage naturel.

La notation $i \models j$ désigne qu'un porteur d'information i est *à-propos* d'un porteur d'information j. Une définition stricte de l'à-propos (c'est-à-dire s'appliquant entre documents et requête) est la suivante :

> *Un porteur d'information i est dit à-propos d'un porteur d'information j si l'information tirée de i tient dans j.*

Une relation d'ordre partiel peut être définie sur les porteurs d'information :

> $i \rightarrow j$ *si et seulement si i contient l'information que porte j.*

L'ordre sur cette relation est le suivant : $i < j$ si i porte plus d'information que j. Cette relation d'ordre est analogue à la relation spécifique-générique : un spécifique porte plus d'information que son générique. Par exemple, on sait davantage de chose sur un objet s'il est appelé *chaise* que s'il est appelé *meuble* (c'est-à-dire : $chaise \rightarrow meuble$).

Un document peut être vu comme une composition d'information, cette composition respectant certaines règles. La relation de composition sur les porteurs d'information est notée par $\oplus$:

> $i \oplus j$ *est le plus grand porteur d'information (au sens défini par $\rightarrow$) qui contient précisément l'information née de i et de j.*

Par exemple, dans le modèle booléen, la composition est le ET : $air \oplus pollution$ est représenté par $air \wedge pollution$. La composition d'information, appliquée à l'extrême à tous les porteurs d'information, donne l'information totale (notée ∞), qui est l'élément minimal de l'ordre partiel sur tous les porteurs d'information. À noter que l'information totale représente «trop d'information» et est comparable à $faux$ dans le calcul des prédicats, au sens où $faux$ permet de déduire nimporte quoi.

Certains porteurs d'information peuvent être incompatibles : l'information qu'ils portent peut être conflictuelle ou contradictoire. Deux porteurs d'information dans ce cas sont dit s'exclure mutuellement, ce qui est noté $i \perp j$. Par exemple, avec $i = \texttt{uni}$ et $j = \texttt{bicolore}$, $i \perp j$. Un porteur d'information est dit *harmonieux* si et seulement s'il ne porte d'information discordante, c'est-à-dire :

$$\nexists i, j \text{ tels que } k \rightarrow i \text{ et } k \rightarrow j \text{ avec } i \perp j.$$

Deux porteurs d'information s'excluent mutuellement si leur composition donne l'information totale :

$$i \perp j \;\; ssi \;\; i \oplus j = \infty.$$

La duale de la composition d'information est la dégradation d'information. Dans le modèle booléen, la dégradation d'information est représentée par le OU. En effet, $air \vee terre$ est plus vague que air.

Un *champ d'information* est une structure $(\mathcal{I}, \rightarrow, \oplus, \perp, \infty)$ telle que

1. $\mathcal{I}$ est un ensemble non vide de porteurs d'informations,
2. $(\mathcal{I}, \rightarrow)$ est un ensemble partiellement ordonné (*poset*),
3. $\infty \in \mathcal{I}$ et pour tout $i \in \mathcal{I}$, $\infty \rightarrow i$,
4. si $i, j \in \mathcal{I}$ alors $i \oplus j \in \mathcal{I}$ et $i \oplus j$ est le plus grand porteur d'information tel que $i \oplus j \rightarrow i$ et $i \oplus j \rightarrow j$,
5. $\perp \subseteq \mathcal{I} \times \mathcal{I}$.

Bruza définit un certain nombre d'axiomes pour caractériser les hypothèses inhérentes aux mécanismes de RI. Ces axiomes sont

1. la **réflexivité**

$$i \models i$$

2. la **contention** (en anglais *containment*) : un porteur d'information est à-propos de l'information qu'il contient

$$\frac{i \rightarrow j}{i \models j}$$

3. la **monotonicité de contention à droite** : si un document d est à-propos de $chat$ et $chat \rightarrow animal$, d est à propos d'$animal$

$$\frac{k \models i \quad i \rightarrow j}{k \models j}$$

4. le **ET contexte libre** : si un document D est à-propos de air et D est à propos de $pollution$, D est à-propos de $air \oplus pollution$. Cet axiome est le fondement de la RI booléenne, où l'opérateur de composition $\oplus$ est représenté par le ET booléen ($\wedge$).

$$\frac{k \models i \quad k \models j}{k \models i \oplus j}$$

5. le **ET sensible au contexte**, où la composition d'information ne peut se produire que si les porteurs d'information sont dans un même contexte informationnel c.

$$\frac{i \rightarrow c \quad c \models j \quad c \models k}{i \models j \oplus k}$$

6. la **logique de négation** : si un document D n'est pas à propos de $pollution$, il n'est pas à propos de la pollution de l'air.

$$\frac{k \not\models i}{k \not\models i \oplus j}$$

7. la **monotonicité compositionnelle** : le fait de composer deux porteurs d'information i et j, si aucune exclusion mutuelle n'est apparente, ne viole pas la relation d'à-propos.

$$\frac{i \models k \quad j \not\perp k}{i \oplus j \models k}$$

8. l'**exclusion mutuelle** : deux porteurs d'information qui s'excluent mutuellement ne sont pas à-propos l'un de l'autre

$$\frac{i \perp j}{i \not\models j \text{ et } j \not\models i}$$

9. la **négation logique** : si D est à propos d'un dessin uni, D est harmonieux, et uni exclut $bicolore$, alors D n'est pas à propos de bicolore.

$$\frac{i \models j \quad j \perp k}{i \not\models k}$$

10. l'**hypothèse du monde clos** : si un porteur d'information i n'est pas à-propos d'un porteur d'information j, si j exclut k et si i n'exclut pas k, alors i est à-propos de k :

$$\frac{i \not\models j \quad j \perp k \quad i \not\perp k}{i \models k}$$

Bruza utilise ces axiomes pour analyser le modèle booléen de RI et le modèle vectoriel. Il conclut à la supériorité du modèle booléen, qui permet des inférences plus puissantes parce qu'il vérifie un plus grand nombre d'axiomes. Nous pouvons remarquer que cette comparaison ne tient qu'au niveau des inférences possibles dans un modèle donné, pas au niveau de la puissance d'expression offerte par ces modèles. De plus, la supériorité constatée du modèle booléen provient de l'existence de la négation, qui n'existe pas dans le modèle vectoriel. Enfin, le modèle booléen repose sur l'hypothèse du monde clos, c'est-à-dire suppose qu'un document soit examiné pour chaque terme d'indexation possible afin qu'une valeur booléenne lui soit affectée pour ce terme, selon qu'il est présent ou non dans le document. Pour les vastes collections de documents, dont la description repose sur un vocabulaire d'indexation très important, cette hypothèse est peu réaliste.

Toutefois, cette approche est intéressante, car elle propose une méthode formelle de définition d'un modèle de RI par une théorie.

2.2.5 Bilan sur le modèle logique

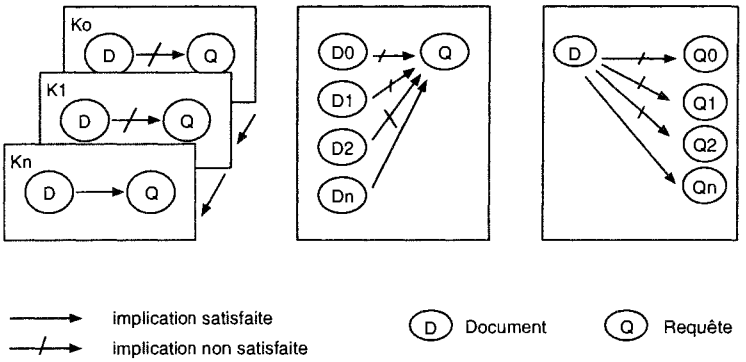


FIG. 2.3 – *Les trois interprétations possibles du principe d'incertitude.*

De nombreux travaux s'appuient désormais sur le modèle logique de RI. Ce modèle en effet est une sorte de méta-modèle, et ne préjuge en rien des représentations et implantations possibles des concepts qu'il met en évidence.

Les trois principes d'incertitude soulignent les trois manières dont l'incertitude peut être introduite durant l'évaluation de la pertinence des documents pour une requête :

- par manipulation des connaissances sur lesquelles repose le système (thésaurus),
- par restriction sur les descriptions de documents, la requête restant la même,
- par évolutions ou reformulations successives de la requête, les descriptions de documents n'étant pas modifiées ; ces évolutions peuvent être dirigées par l'utilisateur (qui raffine progressivement sa requête) ou par le système (par le bouclage de pertinence).

Ces trois principes sont bien résumés par le schéma 2.3 tiré de [Nie 89].

2.3 Le bouclage de pertinence

Le bouclage de pertinence (*relevance feedback*) a été introduit par Rocchio en 1965 [Rocchio 71]. Le principe est d'acquérir davantage de connaissance sur le besoin d'information de l'utilisateur en exploitant ses jugements sur les documents que le SRI lui propose. Si l'on voit les modèles de mise en correspondance comme des moteurs d'inférence fonctionnant en chaînage avant (à partir d'une requête, le SRI cherche à trouver des documents pertinents), le bouclage de pertinence correspond à une sorte de chaînage arrière, puisque, à partir des documents que l'utilisateur estime pertinents pour son besoin d'information, le SRI tente de reformuler la requête ; cette reformulation peut être vue comme l'émission de nouvelles hypothèses sur le besoin d'information. Ce principe est particulièrement adapté à la recherche d'images pour plusieurs raisons : (1) il est plus difficile pour un utilisateur d'émettre une requête sous forme de mots-clés ou de texte pour rechercher une image (voir § 1.2.2), (2) les images ne nécessitent pas un temps de «lecture» trop long ce qui favorise l'interactivité, (3) les critères pour lesquels une image satisfait le besoin d'information de l'utilisateur sont plus subjectifs encore que pour des documents textuels ; l'avis de l'utilisateur est donc plus important encore dans ce cas.

2.3.1 Bouclage de pertinence dans le modèle vectoriel

Dans un premier temps, le bouclage de pertinence permettait de modifier les poids d'importance des termes de la requête en fonction de leur présence dans les documents jugés pertinents ou non pertinents par l'utilisateur. Il a ensuite été utilisé pour étendre la requête avec de nouveaux termes, «découverts» dans les documents choisis.

Les fonctions de choix dont dispose l'utilisateur pour évaluer la pertinence des documents peuvent être ternaires (document choisi/rejeté/neutre) ou plus détaillées. Dans le cadre du modèle vectoriel, une fonction générale de reformulation [Salton 83a, Salton 90] recalcule le poids w affecté aux termes de la requête à chaque étape i de la recherche en fonction de leur présence dans les documents jugés pertinents ou non pertinents par l'utilisateur. Salton propose la formule générale suivante :

$$w_{i+1} = \alpha w_i + \beta \frac{\sum_{j \in \mathcal{C}} D_j}{|D_j|} - \gamma \frac{\sum_{k \in \mathcal{R}} D_k}{|D_k|} \tag{2.7}$$

où i correspond à l'étape d'itération, $\mathcal{C}$ est l'ensemble des documents choisis et $\mathcal{R}$ l'ensemble des documents rejetés, c'est-à-dire jugés non pertinents par l'utilisateur. Cette formule correspond à celle de Rocchio avec un paramètre supplémentaire (α) permettant de moduler l'importance de la requête précédente. La formule 2.7 pose quelques problèmes théoriques. Dans le cas où le système n'utilise pas de poids négatifs pour les termes, le re-calcul des poids par cette formule peut en introduire. Cet écueil est résolu en général en supprimant de la requête les termes dont le poids recalculé devient négatif. Par ailleurs, en l'absence de contrainte sur les paramètres α, β et γ, les nouveaux poids peuvent ne plus être dans l'intervalle [-1,1]. Par exemple, si $\alpha = \beta = \gamma = 1$, les termes ayant un poids maximal à l'étape précédente ($w_i = 1$), présents dans au moins un document choisi et jamais présents dans les descriptions des documents rejetés

vont obtenir un nouveau poids supérieur 1. Il est donc nécessaire d'établir des contraintes sur ces paramètres si l'on souhaite conserver le poids d'un terme dans l'intervalle $[-1, +1]$.

Plusieurs variantes de cette formule ont été utilisées, pour prendre en compte par exemple uniquement le document le plus mal classé par l'utilisateur [Harman 92]. Cette formule peut également être associée à un seuil de pertinence : les termes dont le nouveau poids dépasse ce seuil sont ajoutés à la requête, ceux dont le poids tombe en dessous du seuil en sont retirés.

Le raisonnement utilisé par le bouclage de pertinence peut être formalisée ainsi : le terme t_1 décrit le document D_1, le document D_1 est pertinent pour l'utilisateur, donc le terme t_1 est susceptible d'être pertinent pour l'utilisateur. Soit :

$$(t_1 \in D_1) \wedge (D1 \ \ pertinent) \Rightarrow t_1 \ \ pertinent$$

Une hypothèse implicite est donc : tous les termes décrivant un document pertinent sont pertinents :

$$(D \ \ pertinent) \Rightarrow (\forall t \in D)(t \ \ pertinent)$$

Cette hypothèse n'est évidemment pas toujours vérifiée.

Par contre, un autre hypothèse est plus plausible, à savoir qu'il existe un sous-ensemble de descripteurs d'un document pertinent T_D^p qui sont pertinents pour exprimer le besoin d'information de l'utilisateur :

$$(D \ \ pertinent) \Rightarrow (\forall t \in T_D^p)(T_D^p \subseteq T_D)(t \ \ pertinent).$$

Un autre problème peut survenir si le système permet à l'utilisateur de rejeter des documents estimés non pertinents. Si le besoin d'information est représenté par plusieurs termes, par exemple t_1, t_2, t_3, le chercheur d'information peut choisir des documents où $t1$ apparaît en compagnie de t_2 et t_3, mais rejeter ceux où t_1 apparaît seul. Que se passe-t-il alors vis-à-vis de la reformulation de la requête selon le principe énoncé plus haut ? Le poids du terme t_1 va être réduit, et peut même devenir négatif si ce terme est présent dans un plus grand nombre de documents rejetés que choisis, si $\alpha = 0$ et si $\gamma \geq \beta$. Or, ce terme est malgré tout pertinent pour décrire la requête, mais les jugements négatifs de l'utilisateur n'ont pu porter que sur des documents comportant uniquement ce terme, ce qui fausse l'interprétation liée à la reformulation. En effet, alors que les termes non présents dans la requête ni dans les documents présentés ont un poids nul qui représente leur neutralité vis-à-vis de l'expression du besoin, le terme t_1 peut devenir tout à coup moins pertinent que ces termes neutres ...

Prenons un exemple concret ; un utilisateur veut une photographie montrant un chat sur un divan. Il formule une première requête $q_1 = (chat(1), divan(1))$. La collection d'images ne contient pas d'image de chat sur un divan. Le système, selon la mise en correspondance vectorielle, va présenter des images de chat diverses (par exemple 15), et des images de divan (par exemple 5). Parmi les images de chat, l'une d'elles représente un chat sur un fauteuil. L'utilisateur va choisir cette image, rejeter les images ne comportant pas de chat mais un divan. Il va rejeter également les photographies de chat prises en extérieur (au nombre de 5), car il veut illustrer l'idée de confort douillet lié à une maison. Nous supposons ici que tous les termes

descripteurs d'une image ont un poids de 1. Lors de la reformulation, selon la formule 2.7, le système va modifier les poids des termes ainsi, en supposant $\alpha = 1$, $\beta = 1$, $\gamma = 1$ et $(w_i(chat) = 1)$:

$$w_{i+1}(chat) = (w_i(chat) + 1 - 5/10) = 1 + 1 - 5/10 = 1,5.$$

Ce poids sera ramené à 1 pour rester dans l'intervalle admis [-1,1]. Par contre

$$w_{i+1}(divan) = 1 + 0/1 - 5/5 = 0$$

et

$$w_{i+1}(fauteuil) = 0 + 1/1 - 0/10 = 1.$$

Le concept $divan$ est supprimé de la requête car considéré comme neutre. On peut remarquer que le système a appris le concept $fauteuil$. Pour le même exemple, si $\alpha = 0$, c'est-à-dire si la requête précédente n'est pas prise en compte,

$$w_{i+1}(chat) = 1 - 5/10 = 0,5.$$

Le poids d'importance du concept $chat$ diminue. Or, ce concept est toujours aussi important pour l'utilisateur. La diminution reflète en fait que, **par rapport aux documents de la collection**, ce terme est moins discriminant pour caractériser le besoin d'information de l'utilisateur. Cette technique de reformulation ne permet pas de faire des hypothèses sur la **présence conjointe** de termes pour représenter le besoin d'information de l'utilisateur. Ceci pourrait être exprimé par des opérateurs booléens étendus (voir § 2.1.3) : une conjonction exprimerait une co-occurrence de termes dans la description des documents jugés pertinents. Le poids de chaque terme montrerait l'importance du concept pour le besoin d'information, le «poids» des opérateurs booléens représentant le degré de rigueur pour leur interprétation. Une autre solution serait d'affecter des poids non aux termes, mais aux relations qui lient ces termes dans la description ; nous ferons des propositions en ce sens au § 4.3.12.

Cet exemple montre également qu'ici, le système n'a pas du tout pris en compte le fait que $fauteuil$ et $divan$ sont des sortes de meubles, connaissance qui pourrait être disponible dans un thésaurus, mais les a traités comme des termes indépendants. Pourtant, cette connaissance est certainement d'importance dans le choix fait par l'utilisateur !

Dans RIVAGE [Halin 89], ce problème est abordé par adaptation du principe de construction de concept à partir de l'espace de versions de Mitchell [Michalski 83]. La demande est représentée dans le thésaurus, assimilé à un ensemble de taxonomies, par un niveau d'expression. À chaque nœud de l'arbre est associé un poids d'expressivité, calculé au prorata de l'apparition du terme dans les documents choisis et rejetés. Le thésaurus est ensuite parcouru en partant des termes les plus spécifiques ; le poids d'expressivité d'un terme plus générique est obtenu grâce à une fonction combinant les poids de ses spécifiques. Cette fonction peut être la moyenne des poids de tous les spécifiques. Puis, un nouveau parcours de chaque champ sémantique est réalisé, pour en extraire les concepts *appris*, c'est-à-dire ceux dont le poids est supérieur à un seuil fixé. Ce principe permet non seulement de généraliser la formulation de la demande, mais aussi

de trouver le bon niveau de généralisation en fonction des connaissances du système. Il nécessite toutefois de disposer d'une taxonomie de concepts.

Le bouclage de pertinence peut être vu comme un raisonnement par induction : à partir d'observations sur les documents choisis et rejetés par l'utilisateur, le système tente d'induire une expression représentant mieux son besoin d'information. Ainsi, si un utilisateur formule la requête «*chat, panier*» et ne sélectionne parmi les images de chat, que celles qui représentent des chats persans, le système reformule la requête par «chat persan, panier». S'il rejette des photographies de chat siamois, cela ne signifie pas qu'il ne cherche pas des images de chat. Si l'on dispose d'une taxonomie où chat persan et chat siamois sont des sortes de chat, le système pourra apprendre que c'est précisément ce type de chat qui intéresse l'utilisateur. En l'absence de ces connaissances, si le nombre d'images de chats siamois (rejetées) est supérieur au nombre d'images de chats persans (choisies), le système risque de mal interpréter ces choix en reformulant une requête dans laquelle le terme *chat* aura un poids d'importance faible.

En général, le bouclage de pertinence introduit un certain bruit dans la requête, proportionnel d'une part au nombre de descripteurs attachés à chaque document, et d'autre part au nombre de documents que l'utilisateur peut visualiser. De plus l'incertitude liée à l'induction est grande : non seulement on n'est pas sûr que l'image a été choisie pour tous ses descripteurs, mais elle peut l'avoir été pour des motifs totalement autres, par exemple liés à la sensibilité de l'utilisateur : couleurs, composition par exemple. C'est pourquoi il est important de prendre en compte dans une base d'images le plus grand nombre possible de points de vue sur ces documents, et en particulier ceux liés à la morphologie : une requête sera rarement formulée dans cette facette, mais elle intervient beaucoup dans le jugement de pertinence de l'utilisateur. Ce type de connaissance est donc particulièrement intéressant à traiter par induction.

Une possibilité plus fine, mais aussi plus contraignante pour l'utilisateur serait de lui demander de se prononcer sur chaque descripteur du document choisi. Une solution intermédiaire consisterait à lui demander de se prononcer sur chaque aspect du document.

2.3.2 Bouclage de pertinence dans les modèles probabilistes

Dans le modèle de Robertson et Sparck-Jones, le calcul de la pertinence d'un document est représenté par une probabilité : celle pour chaque terme de la requête d'être présent dans les documents pertinents et d'être absent des documents non pertinents. Les termes, dans les deux sous-ensembles de documents (pertinents et non pertinents) ont une distribution qui suit une loi de Poisson (la loi des petites probabilités). Les jugements de l'utilisateur lors de la phase de visualisation fournissent donc un échantillon sur lequel le système effectue un apprentissage. Le poids d'un terme i dans une requête j, noté w_{ij}, est proportionnel à la proportion de documents dans les documents pertinents qui sont indexés par i par rapport à la proportion dans les documents non pertinents de ceux non indexés par i :

$$w_{ij} = log_2 \frac{\frac{r}{R-r}}{\frac{n-r}{N-n-R+r}}$$

où

$$\begin{cases} r & \text{est le nombre de documents pertinents pour la requête } j \text{ qui contiennent le terme } i, \\ R & \text{est le nombre total de documents pertinents pour la requête } j, \\ n & \text{est le nombre de documents dans la collection qui contiennent le terme } i, \\ N & \text{est le nombre total de documents dans la collection} \end{cases}$$

Ce poids est nul si le terme est autant présent dans les documents pertinents que dans les non pertinents.

La longueur des documents s'est montrée empiriquement importante pour l'efficacité du bouclage de pertinence [Harman 92]. Intuitivement en effet, on peut penser que, lorsqu'un document est choisi, le nombre de nouveaux termes qu'il risque d'introduire est proportionnel à sa longueur. Certaines mesures ont tenté de prendre ce critère en compte, en normalisant le poids d'un descripteur dans un document par la taille de ce document.

Dans le modèle de réseau bayésien d'inférence, le bouclage de pertinence peut se traduire par des tactiques différentes.

1. Augmenter la plausibilité pour les documents choisis, ce qui modifie les croyances. Cette solution est valide lorsque la topologie initiale du réseau est correcte, en général parce qu'elle a été apprise sur un échantillon fiable.

2. Modifier les dépendances, soit en changeant la topologie du réseau soit en modifiant les matrices de liens. Cette approche convient lorsque la topologie initiale du réseau est approximative et que l'on cherche à améliorer l'information sur la nature de la distribution réelle.

3. Modifier l'espace de requête en ajoutant des liens entre les nœuds de la requête et les concepts à ajouter, et ré-estimer les matrices de liens à partir de l'échantillon de documents pertinents. Ceci permet de simuler le processus de bouclage de pertinence du modèle vectoriel [Turtle 92].

2.3.3 Remarques sur le bouclage de pertinence

Le bouclage de pertinence, s'il peut permettre d'affiner la description du besoin de l'utilisateur, présente malgré tout quelques inconvénients. Tout d'abord, lorsqu'il est utilisé après formulation initiale d'une requête par l'utilisateur, **il ne permet d'améliorer la recherche que relativement aux résultats obtenus à partir de la requête initiale**. Si aucun ou très peu de documents pertinents ont été retrouvés et présentés à l'utilisateur, le bouclage de pertinence ne peut pas fonctionner. De plus, il peut **engendrer du bruit** lorsque l'utilisateur évalue globalement un document comme pertinent : de nombreux descripteurs du document qui n'ont aucun rapport avec le besoin d'information peuvent être introduits dans la requête. Certains systèmes comme I^3R donnent la possibilité à l'utilisateur de choisir les concepts les plus pertinents dans le document choisi, ce qui semble améliorer la précision de l'étape de recherche suivante [Croft 90a].

Pour les utilisateurs, le bouclage de pertinence (suivi de la reformulation automatique de la requête par le système) peut être considéré comme un substitut à l'émission d'une nouvelle requête ; si leur besoin d'information change lors d'une session de recherche, les différentes étapes de la session vont chercher alors à cerner non pas UN besoin unique, mais DES besoins successifs, ceux-ci pouvant être d'ailleurs provoqués par la visualisation des documents. Pour éviter que l'utilisateur ne s'égare dans une recherche bouclant un trop grand nombre de fois, le nombre de cycles ou d'étapes de reformulation peut être limité par le système soit par un paramètre (comme dans I^3R, où ce paramètre dépend de l'expérience de l'utilisateur vis-à-vis du SRI) soit, comme nous proposons de l'étudier, par observation de la différentielle des résultats obtenus d'une étape de recherche à l'autre. Enfin, du point de vue psychologique, les utilisateurs peuvent être réticents face à l'effort supplémentaire qui leur est demandé lors de l'étape de visualisation des documents ; ces efforts doivent donc être largement récompensés par l'amélioration des résultats de l'étape de recherche suivante.

2.4 Conclusion

L'analyse de quelques-uns des modèles existants en recherche d'information fait apparaître leurs avantages et inconvénients. Le modèle logique peut être vu comme un méta-modèle qui donne des bases solides pour déterminer ce que doit être un modèle de RI. Il s'avère efficace pour comparer, comme nous l'avons détaillé plus haut, les mesures de similarité existantes qui sont utilisées dans les modèles booléen ou vectoriel. Ce modèle permet aussi de formaliser l'utilisation de connaissances autres que celles directement liées aux documents et à la requête. Par exemple, les relations spécifiques/générique d'un thésaurus peuvent être vues comme des sortes d'implications logiques ([Chiaramella 92]).

Le formalisme de représentation adopté à des conséquences sur le pouvoir d'expression donné aux utilisateurs pour formuler leur requête :

- possibilité de structurer leur demande,
- de s'exprimer par la négative.

Il a aussi des conséquences sur la profondeur de raisonnement du système :

- raisonnement sur les descriptions de documents et
- raisonnement sur les connaissances du domaine (thésaurus),
- raisonnement sur les méta-connaissances du système (attributs de descriptions).

La pondération des termes peut avoir de nombreuses interprétations différentes, en particulier en fonction du type de média traité. C'est pourquoi l'interprétation des poids que nous retenons est celle qui permet de nuancer l'expression du besoin d'information, en donnant des importances relatives aux différents composants de la requête.

Le modèle vectoriel a une efficacité surprenante par rapport à la simplicité des représentations qu'il traite. Mais il a aussi ses limites : il ne permet pas d'exprimer des relations entre les concepts ou des concepts composites, ni de traiter directement des requêtes négatives. Nous ne le rejetons pas pour autant, mais estimons qu'il existe d'autres formalismes de représentation qui méritent d'être examinés.

Tous les modèles que nous avons présentés n'exploitent que des mots-clés descripteurs pour la représentation des documents et de la requête. Pourquoi se limiter à cette représentation simplifiée ? Nous allons examiner d'une part une structuration possible des connaissances sur les documents et d'autre part un formalisme de représentation plus riche, déjà utilisé en recherche d'information [Chevallet 92, Kheirbek 95] : le formalisme des graphes conceptuels. Ce formalisme nous intéresse particulièrement parce qu'il permet de représenter explicitement des relations entre les concepts.

Chapitre 3

Propositions pour la structuration des connaissances sur les documents

Nous avons évoqué au chapitre 1 les connaissances qui sont manipulées par un SRI. Si la quantité d'information contenue dans un document est en général assez importante, elle l'est plus encore pour la vidéo. En outre, selon la personne qui regarde le document, des informations différentes seront mises en relief. Enfin, plus que des objets ou concepts particuliers, un document vidéo représente des actions. Or les actions sont très mal prises en compte dans des structures de classification des connaissances comme les thésaurus d'une part, et sont difficiles à représenter par un mot-clé unique dans les descriptions de documents d'autre part.

Dans ce chapitre, nous proposons de considérer différentes facettes de connaissance sur les documents (§ 3.1). La structuration de ces connaissances nous amène à adopter un formalisme de représentation adéquat, les graphes conceptuels. Après avoir présenté les concepts importants de ce formalisme (§3.2), nous explicitons la modélisation des connaissances du SRI pour ce formalisme (§ 3.3). Nous comparons enfin notre approche à d'autres travaux (§ 3.4).

3.1 Facettes de documents

Deux constats nous ont incitée à proposer un modèle de données particulier pour les documents. Le premier concerne l'ambiguïté d'une description par mots-clés. Par exemple, si un film est décrit par *Clint Eastwood*, il n'est pas possible de savoir si ce descripteur concerne un acteur ou le réalisateur du film, puisque cette personne a joué ces deux rôles dans son existence. Il est donc souhaitable d'attacher les mots-clés descripteurs des documents à des attributs. Dans l'exemple précédent, en fonction du film décrit, le descripteur *Clint Eastwood* serait attaché soit à l'attribut *AUTEUR*, soit à l'attribut *ACTEUR*, ou quelquefois aux deux pour les films où il est intervenu dans ces deux rôles. Le second constat est que les descriptions de documents utiles à la recherche d'information sont relatives à différents domaines conceptuels, que nous pouvons appeler aussi *contextes*. Pour faire face à l'étendue des connaissances qu'il est possible d'exploiter sur les documents, et pour structurer les connaissances décrites dans un SRI, nous avons choisi

une approche de descriptions par **facettes** des documents [Simonnot 94, Simonnot 95b]. Nous donnons au concept de facette la sémantique suivante : une facette correspond à un contexte, au point de vue d'un spécialiste sur le document. Une facette est donc attachée à un domaine d'expertise particulier et affecte un rôle aux concepts qui la décrivent. La gestion des connaissances par facettes permet de regrouper certains attributs liés, plutôt que de les faire apparaître artificiellement comme des attributs indépendants. Elle permet également de distinguer des attributs qui seraient orthogonaux entre eux. Par exemple, la description du contenu d'un document est relativement orthogonale de la description de sa signalétique (auteur, date et lieu de production ou de publication). La distribution logique des connaissances dans plusieurs facettes permet également d'envisager des profondeurs diverses dans les descriptions de documents, et des inférences différenciées en fonction du domaine de connaissance considéré.

> Prenons l'exemple d'un chercheur d'information qui souhaite retrouver un document vidéo montrant une ville du Canada, par exemple Ottawa, en hiver. L'aspect météorologique est le point déterminant de sa demande. La collection à laquelle il adresse sa demande ne contient pas de tels documents sur Ottawa (Ontario), mais par contre comporte des documents tournés en hiver à Québec et à Vancouver (Colombie Britannique). Si la structure de connaissance utilisée pour interpréter la requête est un thésaurus géographique répertoriant les principales régions du Canada selon une relation *partie-tout* et indiquant simplement que l'Ontario, le Québec et la Colombie Britannique sont des provinces du Canada, l'inférence selon laquelle les documents concernant ces trois régions seraient équivalents pour le besoin d'information courant serait fausse. La relation de composition géographique n'est pas utilisable dans ce cas, les phénomènes atmosphériques étant très différents à l'est et à l'ouest du Canada. Par contre, si une personne cherche des documents sur les étudiants et le système éducatif, des documents sur les universités des villes de ces deux régions pourront être utilisés de façon interchangeable. En fonction de la situation, les inférences que peut faire le système sur les connaissances dont il dispose ne sont pas les mêmes.

La description des documents par facettes permet de structurer les différents domaines de connaissance qui sont pris en compte. Au lieu d'être juxtaposés en une structure «plate», les descripteurs peuvent être assemblés au sein de sous-structures, ce qui donne du relief aux descriptions et leur permet d'évoluer de manière relativement indépendante dans chacune des sous-structures. La notion de facette permet d'introduire une autre structure, plus souple et plus proche de l'évolution actuelle de la connaissance qui se spécialise mais dont beaucoup de champs deviennent pluri-disciplinaires. Elle est complémentaire des classifications traditionnelles, dont elle permet de **mettre les concepts en situation**.

3.1.1 Catégories de facettes

Pour des documents multimédias, nous avons identifié plusieurs catégories de facettes, en fonction du type d'expertise qu'elles recouvrent [Simonnot 95b] :

- Les facettes qui décrivent le **contenu** du document, c'est-à-dire les objets, personnes, places qui peuvent y être objectivement vus. Ces éléments peuvent être décrits de façon géné-

rale (par exemple : *femme, actrice*) et/ou en citant des noms propres (par exemple : *Jane Fonda*).

- Les facettes qui détiennent la connaissance propre au domaine d'application ; par exemple pour une application médicale, elle recouvre les connaissances techniques exprimées dans le vocabulaire des disciplines concernées. Pour des films, ce peut être le nom des personnages joués par les acteurs et les récompenses obtenues.

- Les connaissances sur la morphologie du média proprement dit. Pour des médias audiovisuels, cela concerne la lumière, le cadrage, l'angle, la position et les mouvements de la caméra lors des prises de vue, ou la couleur et la texture. Ces éléments ont une sémantique propre : par exemple dans un document vidéo bien réalisé, la caméra ne bouge pas sauf dans une intention bien définie (association, orientation ou exploration). Pour des graphiques, il s'agit des formes géométriques (lignes, cercles, triangles), de leur position et de leur orientation, de leur dimension (2D ou 3D). Pour les textes, cette facette correspondrait à la typographie (taille et type de fonte) et à la mise en page, ces aspects n'étant bien évidemment décrits que s'ils présentent un intérêt pour une recherche d'information ; ce peut être le cas pour une collection de manuscrits anciens.

- Les connaissances signalétiques qui ont trait aux conditions de production du document comme l'auteur, la date de création, le type de média, le titre de l'œuvre, et qui sont gérées habituellement par des spécialistes de la documentation.

Selon le type d'application concernée et le contexte dans lequel elle est utilisée, il pourra être nécessaire d'enregistrer d'autres connaissances comme :

- les connaissances propres à l'archivage, au stockage des documents et à la bonne conservation des supports.

- les connaissances légales, par exemple sur les droits de diffusion et de reproduction des documents.

- la connaissance apportée a posteriori sur les documents, par exemple par l'analyse de critiques. Cette connaissance peut concerner en particulier la connotation des documents, critère subjectif s'il en est. Cet aspect des descriptions peut être très important pour certaines applications comme celles qui exploitent des reproductions de documents artistiques.

3.1.2 Indépendance des facettes

Les types de connaissance énumérés plus haut sont suffisamment orthogonaux les uns par rapport aux autres pour être gérés de façon indépendante. La conception du système, si elle s'appuie sur cette classification, y gagne en souplesse parce que différents experts pourront traiter indépendamment les connaissances qui les concernent. De plus, l'exploitation séparée des

différentes facettes ne peut qu'améliorer la recherche d'information : la facette liée à la description du contenu pourra utiliser une représentation plus riche ou complexe (par exemple sous forme de graphes conceptuels) que la facette décrivant les connaissances factuelles générales où de simples listes de mots-clés suffisent pour décrire le(s) auteur(s), la date de création et le titre éventuel donné au document. Chaque catégorie de facette peut avoir son propre thésaurus, et traiter de façon particulière les termes qui représentent la connaissance du domaine, ou se passer de ces connaissances *a priori* pour exploiter d'autres ressources. Un exemple caractéristique est la description de la couleur des images, qui peut être faite sous forme de mots-clés (bleu, rouge, vert, jaune, ...), ou directement sous forme de codes numériques comme dans le système HLS (Hue-Lightness-Saturation = teinte, luminance, saturation) ou comme dans le système RGB (Red, Green, Blue) de composition des couleurs par addition.

Par contre, les différentes facettes de description des documents peuvent posséder des connaissances les unes sur les autres, de façon non à travailler en parallèle, mais à coopérer et à coordonner leurs résultats.

3.1.3 Choix d'un formalisme de représentation : les graphes conceptuels

Les connaissances, telles que nous souhaitons les représenter, doivent pouvoir être manipulées à différents niveaux d'abstraction. Par ailleurs, le formalisme de représentation utilisé doit permettre de représenter de façon explicite des relations entre les concepts. Nous avons indiqué au § 2.1.2 que le modèle vectoriel nous paraissait insuffisant pour représenter de façon précise des informations. Bien sûr, la structuration en facettes peut être simulée en considérant des sous-requêtes, et en représentant chaque sous-requête dans une facette par un vecteur indépendant. La synthèse effectuée peut alors consister à utiliser une moyenne des mesures de similarité obtenues dans chaque facette. Si les documents que nous traitons sont structurés, comme c'est le cas pour des documents vidéo, ce modèle nous oblige à déterminer des vecteurs documents pré-définis, donc à choisir une fois pour toutes la segmentation des vidéos pour l'indexation et la présentation, ce qui est peu souhaitable comme nous l'avons indiqué au § 1.2.4. L'alternative, qui consiste à considérer qu'un terme a un poids proportionnel à l'apparition de l'objet qu'il décrit, est très simplificatrice et ne permet pas de répondre à des requêtes précises. Le formalisme des graphes conceptuels est mieux adapté à des représentations précises et ne nous impose pas de définir *a priori* la segmentation des documents, c'est pourquoi nous l'avons choisi pour représenter les connaissances du SRI.

Enfin, ce formalisme plus complet peut être ramené au modèle vectoriel en ne tenant compte que des concepts et en faisant abstraction des relations qui existent entre ces concepts. Le choix de ce formalisme pour la représentation des données ne nous interdit donc pas d'avoir recours au modèle vectoriel dans certaines situations où la précision n'est pas requise.

3.2 Présentation de la théorie des graphes conceptuels

Le formalisme des graphes conceptuels permet de manipuler des types de concepts et des relations entre les concepts. Sowa a montré que la théorie des graphes conceptuels est équivalente à la logique du premier ordre [Sowa 84], preuve qui a été complétée par [Wermelinger 95].

3.2.1 Les concepts et les relations conceptuelles

Les graphes conceptuels, introduits par Sowa [Sowa 84], constituent un langage de représentation de la connaissance. Dans les graphes, les nœuds concepts représentent des entités, des attributs, des états ou des événements et les nœuds relations indiquent comment les nœuds concepts sont interconnectés. Graphiquement, un concept est représenté à l'intérieur d'une boîte rectangulaire, alors que les relations conceptuelles sont représentées dans une boîte arrondie. L'arité de la relation est représentée par des flèches. Le premier argument de la relation est symbolisé par un arc qui arrive à la relation, et les suivantes par des arcs (numérotés) qui sortent de la relation. Par exemple, *un chat mange une souris* est représenté par :

$$\boxed{\text{CHAT}} \rightarrow \textcircled{\text{MANGE}} \rightarrow \boxed{\text{SOURIS}}$$

Il est aussi possible d'utiliser une écriture linéaire :

$$[\text{CHAT}] \rightarrow (\text{MANGE}) \rightarrow [\text{SOURIS}].$$

La plupart des relations conceptuelles sont binaires, mais le nombre d'arcs d'une relation peut être arbitraire. Par exemple, un graphe représentant la négation d'un concept serait unaire :

$$\textcircled{\text{NON}} \rightarrow \boxed{\text{CHAT}}$$

alors que la relation conceptuelle ENTRE est ternaire :

$$\begin{array}{ccccc} & & \boxed{\text{Nancy}} & & \\ & & \downarrow & & \\ \boxed{\text{Paris}} & \leftarrow & \textcircled{\text{ENTRE}} & \rightarrow & \boxed{\text{Strasbourg}} \end{array}$$

Un concept isolé peut former un graphe conceptuel a lui seul, mais s'il existe des arcs pour une relation conceptuelle, chaque arc doit être relié à un concept.

La théorie des graphes conceptuels repose sur la notion de **type**. En effet, elle s'appuie sur l'hypothèse selon laquelle une fonction *type* fait correspondre des concepts à un ensemble T, dont les éléments sont appelés *labels* ou *étiquettes de types*. Deux concepts a et b sont de même type si $type(a) = type(b)$. Si t est un label de type, on note δt la dénotation du type t, c'est-à-dire l'ensemble de toutes les entités qui sont instances de tout concept de type t.

Les types sont organisés en hiérarchie et forment un *treillis* : un treillis est une structure formée d'un ensemble L d'éléments, munie d'un ordre partiel $\leq$ et de deux opérateurs binaires $\cap$ et $\cup$. Ainsi, si a et b sont des éléments de L, $a \cap b$ est appelé plus grande borne inférieure ou *infimum* de a et de b ; pour la hiérarchie des types, c'est le sous-type commun maximal de a et de b. $a \cup b$ est appelé plus petite borne supérieure ou *supremum* de a et de b ; pour la hiérarchie des types, c'est le super-type minimal commun de a et de b. Ces opérateurs satisfont les axiomes suivants :

- Pour tout a et b dans L, $a \cap b \leq a$, $a \cap b \leq b$, et si c est un élément quelconque de L pour lequel $c \leq a$ et $c \leq b$, alors $c \leq a \cap b$.
- Pour tout a et b dans L, $a \leq a \cup b$, $b \leq a \cup b$, et si c est un élément quelconque de L pour lequel $a \leq c$ et $b \leq c$, alors $a \cup b \leq c$.

La hiérarchie de concepts est un treillis borné : au sommet de la hiérarchie, on trouve le type universel (noté $\top$) ; en bas de chaque hiérarchie on trouve le type absurde (noté $\perp$). Il existe une dualité d'intension et d'extension sur les concepts : plus un concept est élevé dans la hiérarchie des types, plus le nombre d'individus auxquels il s'applique est important et moins il est besoin de caractéristiques pour le décrire ; plus on descend dans la hiérarchie, moins le concept décrit d'individus, mais plus le nombre de caractéristiques nécessaires pour le décrire croît. Ainsi, plus on progresse vers le haut de la hiérarchie des types, moins ce type est «informatif». Par exemple, si l'on considère une hiérarchie d'espèces animales :

$$\texttt{insectivore} \leq \texttt{mammifère} \leq \texttt{animal}$$

`animal` est moins informatif que `mammifère` qui est lui-même moins informatif que `insectivore`.

Ainsi, si Berlioz est un chat, il est possible d'écrire : [Chat : #Berlioz] (écriture linéaire) ou $\boxed{\text{CHAT : \#Berlioz}}$ (écriture graphique).

Dans la forme d'écriture linéaire, la partie gauche (avant les deux points) est le *champ type* (dans le dernier exemple, "Chat") et celle de droite le *champ référent* (dans le dernier exemple, "Berlioz"). Le nom du type est souvent noté en majuscules (dans l'exemple CHAT) et le référent noté en minuscules (dans l'exemple Berlioz). Le référent d'un concept peut être * pour les concepts génériques, ou # suivi d'un identifiant pour les concepts individuels. Ainsi, $\boxed{\text{CHAT : *}}$ dénote le concept générique CHAT, qui recouvre tous les individus de type CHAT. Cette notation peut être abrégée en $\boxed{\text{CHAT}}$, le référent générique * étant omis. Le concept $\boxed{\text{CHAT : \#12}}$ dénote l'individu particulier de type CHAT identifié par le numéro 12. D'autres symboles informels permettent de représenter que le référent est

- une mesure, dans ce cas le référent est précédé du caractère @, par exemple @1,80 mètres ; il s'agit d'un raccourci d'écriture pour distinguer un nombre désignant une mesure d'un nombre désignant un individu ;
- l'astérisque (*) pour les concepts génériques peut être suivie d'un nom de variable, par exemple $*x$ note la variable nommée x.

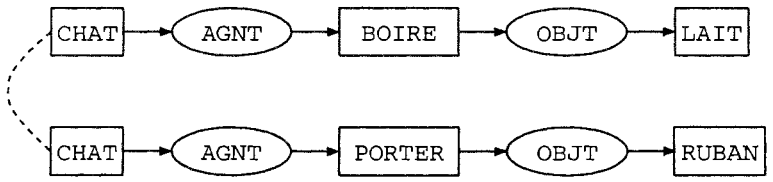


FIG. 3.1 – *Une ligne d'identité regroupant deux concepts complexes sur un concept commun : chat.*

Il existe un ensemble $\mathcal{I} = \{\#1, \#2, \#3, \ldots\}$ dont les éléments sont appelés *marqueurs individuels*. La fonction *référent* peut être appliquée à tout concept c, et l'ensemble d'arrivée de cette fonction est $\mathcal{I} \cup \{*\}$. Lorsque le référent d'un concept c est dans $\mathcal{I}$, c'est un *concept individuel* ; lorsque le référent d'un concept c est *, ce concept est dit *concept générique*.

Une *relation de conformité*, notée ::, relie les noms de types à des marqueurs individuels ; lorsque $t :: i$ est vrai, i est dit conforme au type t. Les contraintes suivantes sont établies sur la relation de conformité :

- Le référent d'un concept doit être conforme à son type ; si c est un concept, type(c)::référent(c).
- Si un marqueur individuel est conforme à un type s, il doit être aussi conforme à tous ses super-types ; si $s :: i$ et $s \leq u$, alors $u :: i$.
- Si un marqueur individuel est conforme aux types s et t, il doit aussi être conforme à leur sous-type commun minimal $s \cap t$; si $s :: i$ et $t :: i$, alors $s \cap t :: i$. Par exemple, Sarah Bernhardt était une FEMME, et elle exerçait la profession de tragédienne, elle est donc conforme au type FEMME-TRAGÉDIENNE.
- Tout marqueur individuel est conforme au type universel $\top$.
- Aucun marqueur individuel n'est conforme au type absurde $\bot$.
- Le marqueur générique * est conforme à tous les types.

Une **ligne d'identité** est un graphe g non orienté dont les nœuds sont des concepts et dont les arcs, appelés *liens de co-référence*, sont des couples de concepts. Les lignes d'identité permettent de représenter les références anaphoriques : un lien de co-référence unit deux concepts identiques qui apparaissent dans des graphes différents. La figure 3.1 illustre le fait que le chat qui boit du lait et celui qui a un ruban sont le même (le lien de co-référence est noté en pointillé entre les deux occurrences du concept CHAT). Pour Esch [Esch 95], deux concepts sont **co-référents** soit parce qu'ils ont le même référent, soit parce qu'ils ont un lien de co-référence.

3.2.2 Les graphes canoniques

Un graphe conceptuel est une combinaison de nœuds concepts et de nœuds relations. Il est donc possible, à partir des concepts et des relations, de créer un grand nombre de graphes. Toutefois, tous les graphes qui peuvent être créés n'ont pas forcément de sens. Les graphes qui

ont un sens sont appelés **graphes canoniques** ; la définition des graphes canoniques permet d'établir des contraintes de sélection sur les relations qui sont permises entre les concepts. La **base des graphes canoniques** est l'ensemble des graphes qui ont un sens. Ils déterminent quelles sont les relations qui sont autorisées sur les types de concepts.

Nous utilisons la **notion de base canonique minimale** [Chevallet 92] pour désigner l'ensemble minimal des graphes canoniques qui permettent de dériver tous les graphes canoniques d'une application :

> Une base canonique est minimale pour un ensemble de graphes donné lorsque toute base obtenue par suppression de l'un de ses éléments ne lui est pas équivalente, c'est-à-dire qu'elle engendre un ensemble de graphes différents.

Un corrollaire qui découle de cette définition est que deux graphes différents d'une base minimale ne sont pas comparables par la relation d'ordre sur les graphes. Nous appellerons quelquefois les graphes de la base canonique minimale les **graphes primitifs**, puisque c'est en partant de ces graphes qu'il est possible de dériver tous les graphes canoniques.

Les graphes de la base canonique spécifient non seulement des contraintes de types sur les concepts impliqués dans une relation, mais déterminent aussi le niveau de généralité maximum des concepts qu'il est possible d'utiliser pour cette relation. Intuitivement, le rôle de la base canonique est de définir des relations prototypes, des sortes de «moules» ou de règles syntaxiques pour la formation des graphes conceptuels. Par exemple, le graphe

$$\boxed{\text{PERSONNE}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{ACTION}}$$

spécifie qu'une action a pour agent un concept de type PERSONNE. S'il n'existe aucun autre graphe dans la base canonique minimale, seuls des concepts de type PERSONNE pourront intervenir comme agents d'une action.

3.2.3 Règles de formation de graphes

Les différentes opérations qui permettent de dériver un graphe sont les suivantes :

- La copie est une duplication du graphe ;
- La restriction de concept consiste à remplacer le type d'un concept du graphe par un de ses sous-types ; si le graphe contient un concept générique, son référent peut être changé en une variable ou un marqueur individuel. Par exemple, $\boxed{\text{CHAT}}$ peut être restreint par $\boxed{\text{CHAT : \#Berlioz}}$; il s'agit d'une spécialisation de concept ; la restriction est une suite d'une ou plusieurs opérations de restrictions sur le même graphe.
- La jointure permet de fusionner deux graphes sur un concept commun ; étant donné deux graphes u et v comportant respectivement un concept c et un concept d, si c et d dénotent le même concept, la jointure de u et v permet d'obtenir un nouveau graphe par copie de

u, suppression de d dans v, et connexion de tous les arcs de d à c. Ainsi, si deux concepts sont dans le même contexte et ne sont pas co-référents, on les rend co-référents. Il s'agit d'une spécialisation de relation ; par exemple la jointure des graphes

$$\boxed{\text{CHAT : \#Berlioz}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{boire}} \rightarrow (\text{OBJT}) \rightarrow \boxed{\text{lait}}$$
$$\oplus_{CHAT}$$
$$\boxed{\text{CHAT : \#Berlioz}} \rightarrow (\text{RACE}) \rightarrow \boxed{\text{Persan}}$$

sur le concept $\boxed{\text{CHAT : \#Berlioz}}$ donne le graphe

$$\begin{array}{ccccccccc} \boxed{\text{CHAT : \#Berlioz}} & \rightarrow & (\text{AGNT}) & \rightarrow & \boxed{\text{boire}} & \rightarrow & (\text{OBJT}) & \rightarrow & \boxed{\text{lait}} \\ \downarrow & & & & & & & & \\ (\text{RACE}) & \rightarrow & \boxed{\text{Persan}} & & & & & & \end{array}$$

La notation adoptée pour la jointure de graphes conceptuels est $\oplus_C$:
$g_1 \oplus_C g_2$ dénote la jointure des graphes g_1 et g_2 sur le concept C. Lorsque les graphes ont un seul concept commun, nous omettrons le nom du concept sur lequel s'effectue la jointure.

- La simplification permet de supprimer dans un graphe une relation redondante, ainsi que tous ses arcs.

Tous les graphes produits par application de ces opérations à des graphes canoniques sont eux-mêmes canoniques. Sowa fait remarquer que ces règles canoniques de formation de graphes ne sont pas des règles d'inférence : elles ne permettent pas de déduire de nouveaux concepts par exemple.

Le **canon** contient l'information nécessaire pour dériver l'ensemble des graphes canoniques. Il est composé

- d'une hiérarchie de type $\mathcal{T}$,
- d'un ensemble de marqueurs individuels $\mathcal{I}$,
- d'une relation de conformité :: qui relie les noms de types de $\mathcal{T}$ aux marqueurs de $\mathcal{I}$,
- et d'un ensemble fini de graphes conceptuels $\mathcal{B}$, appelée base canonique, dont tous les types appartiennent à $\mathcal{T}$ et les référents à $\mathcal{I} \cup \{*\}$.

L'ensemble des graphes canoniques est obtenu par fermeture de $\mathcal{B}$ par les règles de formation canonique.

La restriction est une règle de spécialisation de graphe qui s'applique aux concepts. La jointure est aussi une règle de spécialisation de graphe par ajout d'information dans une relation. Ces deux règles de formation des graphes conceptuels respectent la «vérité» des graphes.

La généralisation de concept est l'opération inverse de la restriction de concept. Un concept peut être généralisé en remplaçant

1. son type par un super-type,

2. un référent individuel par une variable, un label de concept ou le référent générique *,
3. un référent variable par un label de concept ou le référent générique *,
4. un nom de référent par le référent générique *.

De même, il est possible de définir une opération de détachement de concept, en coupant une ligne d'identité de façon à obtenir deux lignes distinctes comprenant chacune un sous-ensemble distinct de concepts [Esch 95]. Cette opération est l'inverse de la jointure.

La généralisation de concept et le détachement permettent de généraliser les graphes. La généralisation ne respecte pas la vérité des graphes : une contrainte qui existe sur un concept relationnel peut ne plus être vérifiée lorsque l'on passe à un concept plus général. Ainsi, si le graphe

1. [CHAT] → (AGNT) → [MANGER] → (OBJT) → [SOURIS]

qui représente que *les chats mangent des souris* est considéré comme vrai, et s'il est généralisé par

2. [ANIMAL] → (AGNT) → [MANGER] → (OBJT) → [SOURIS]

la vérité du graphe général (2) n'est évidemment pas établie : tous les animaux ne mangent pas des souris. À partir du graphe 1, il peut seulement être établi qu'il existe des animaux qui mangent des souris. La généralisation peut être rapprochée d'un raisonnement par induction.

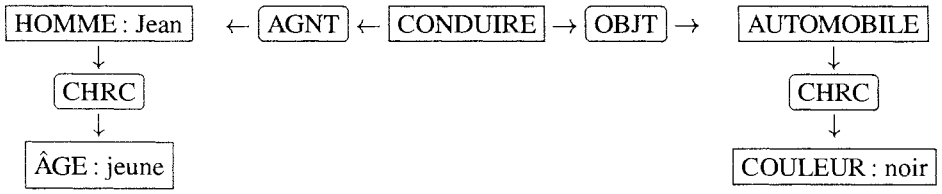
3.2.4 L'opérateur de projection de graphes

Un opérateur $\pi : v \rightarrow u$ de projection de graphes permet de définir un sous-graphe πv d'un graphe u ($u \leq v$) ; πv est alors appelé *projection* de v dans u.

Pour chaque concept c de v, le concept πc de πv, s'il existe, est de même type que le concept c : $type(\pi c) \leq type(c)$. Si c est un concept individuel, alors le référent de πc est le même que le référent de c : $referent(\pi c) = referent(c)$.

Pour chaque relation r dans v, πr, si elle existe, est une relation dans πv où $type(\pi r) = type(r)$. Si le $i$ème arc de r est lié au concept c dans v, le $i$ème arc de πr doit être lié au concept πc de v.

Étant donné une hiérarchie de généralisation de graphes, la projection fait correspondre les graphes les plus élevés dans la hiérarchie avec des graphes qui se trouvent plus bas dans la hiérarchie. Par exemple, étant donné le graphe u :



et le graphe v :

$$\boxed{\text{PERSONNE}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{ACTION}}$$

si CONDUIRE $\leq$ ACTION et HOMME $\leq$ ACTION, la projection de v dans u donne :

$$\boxed{\text{HOMME : Jean}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{CONDUIRE}}$$

Le résultat de la projection n'est pas forcément unique. Par exemple, si un graphe u comporte deux concepts $c1$ et $c2$ de même type TYPE, la projection sur ce type peut donner l'un ou l'autre des concepts de u : $\boxed{\text{TYPE}} \leq u$.

Intuitivement, le graphe v sert en quelque sorte de filtre, appliqué au graphe u.

3.2.5 Ordre sur les graphes conceptuels

Les graphes conceptuels sont ordonnés entre eux par la relation $\leq$, qui permet de définir une hiérarchie de généralisation :

un graphe g_2 est plus général qu'un graphe g_1 (noté $g_1 \leq g_2$) si et seulement si

- g_2 est un sous-graphe de g_1 ;
- g_1 et g_2 ont les mêmes types de concepts, et les labels de types de concepts de g_1 sont plus spécifiques que ceux de g_2 ($g1$ résulte d'une restriction des concepts de g_2) ; par exemple,

$$\begin{array}{ll} g1: & \boxed{\text{HOMME : Dominique}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{CONDUIRE}} \\ \leq & \\ g2: & \boxed{\text{PERSONNE : Dominique}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{CONDUIRE}} \end{array}$$

ou

g_1 est identique à g_2 sauf pour certains concepts génériques de g_2 qui sont restreints à des concepts individuels du même type dans g_1.

$$\begin{array}{ll} g1: & \boxed{\text{PERSONNE : Dominique}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{CONDUIRE}} \\ \leq & \\ g2: & \boxed{\text{PERSONNE}} \rightarrow \boxed{\text{AGNT}} \rightarrow \boxed{\text{CONDUIRE}} \end{array}$$

Si les deux graphes qui entrent dans la relation $\leq$ sont différents (il n'est pas possible d'obtenir l'un par copie de l'autre), on écrira : $g1 < g2$.

3.2.6 Généralisation/spécialisation commune

Esch [Esch 95] introduit les notions d'ancêtre et de descendant d'un graphe. Un *ancêtre* d'un graphe g est tout graphe qui est plus général que g. Le concept universel $\top$ est un ancêtre de tous les graphes. Un *parent* de $g2$ est tout ancêtre $g1$ qui est un prédécesseur immédiat de $g2$; si $g1 = parent(g2)$, alors il n'existe pas de graphe g, tel que $g \neq g1$ et $g \neq g2$ et $g1 \leq g \leq g2$. Un parent est donc un ancêtre direct.

Un *descendant* d'un graphe g est tout graphe qui est plus spécifique que g. Ainsi, le concept absurde $\perp$ est descendant de tous les graphes. Un *enfant* est un descendant direct d'un graphe. Ces notions d'ancêtre et de descendant de graphes nous semblent utiles pour définir la généralisation/spécialisation communes de graphes.

Si u_1, u_2, v et w sont des graphes conceptuels, et si $u_1 \leq v$ et $u_2 \leq v$, alors v est appelé **généralisation commune** de u_1 et de u_2. Le graphe v peut être vu comme l'ancêtre commun de $u1$ et de $u2$. Si v est l'ancêtre commun le plus proche de $u1$ et de $u2$, v est la plus petite généralisation commune ou généralisation commune minimale de $u1$ et $u2$.

Si $w \leq u_1$ et $w \leq u_2$, alors w est appelé **spécialisation commune** de u_1 et u_2. Le graphe w peut être vu comme le descendant commun de $u1$ et de $u2$. Si w est le descendant commun le plus proche de $u1$ et de $u2$, w est appelé plus grande spécialisation commune de u_1 et u_2.

Si les graphes u_1 et u_2 ont une généralisation commune v avec les projections $\pi_1 : v \rightarrow u_1$ et $\pi_2 : v \rightarrow u_2$, les deux **projections** sont dites **compatibles** si pour chaque concept c de v, les conditions suivantes sont respectées :

- $type(\pi_1 c) \cap type(\pi_2 c) > \perp$,
- les référents de $\pi_1 c$ et de $\pi_2 c$ sont conformes au type $type(\pi_1 c) \cap type(\pi_2 c)$,
- si $referent(\pi_1 c)$ est le marqueur individuel i, alors $referent(\pi_2 c)$ est i ou *.

Si les graphes u_1 et u_2 ont une généralisation commune v avec des projections compatibles $\pi_1 : v \rightarrow u_1$ et $\pi_2 : v \rightarrow u_2$, alors il existe un graphe conceptuel unique w qui a les propriétés suivantes :

- w est une spécialisation commune de u_1 et de u_2,
- il existe des projections $\pi'_1 : u_1 \rightarrow w$ et $\pi'_2 : u_2 \rightarrow w$, où $\pi'_1 \pi_1 v = \pi'_2 \pi_2 v$,
- si w' est un autre graphe conceptuel ayant ces deux propriétés, alors $w' < w$.

Le graphe w est appelé **jointure sur des projections compatibles** de u_1 et de u_2, et est canonique si u_1 et u_2 le sont.

Exemple :

u_1 : [PERSONNE : Jean] → (AGNT) → [TRAVAILLE] → (LIEU) → [VILLE : Nancy]

u_2 : [HOMME : Jean] → (AGNT) → [TRAVAILLE] → (OBJT) → [CRISTAL]

v : [PERSONNE] → (AGNT) → [TRAVAILLE]

w : [HOMME : Jean] → (AGNT) → [TRAVAILLE] → (OBJT) → [CRISTAL]
↓
(LIEU)
↓
[VILLE : Nancy]

Le graphe v est une généralisation commune de u_1 et de u_2. La plus petite généralisation commune de u_1 et de u_2 serait

v' : [PERSONNE : Jean] → (AGNT) → [TRAVAILLE]

w est une spécialisation commune de u_1 et de u_2. C'est une spécialisation de u_1 par restriction du concept PERSONNE au concept HOMME, et par jointure sur le concept TRAVAILLE avec le graphe

[TRAVAILLE] → (OBJT) → [CRISTAL]

w est d'ailleurs la plus grande spécialisation commune de $u1$ et de $u2$.

Lorsque deux graphes u_1 et de u_2 ont une généralisation commune v, alors les projections correspondantes $\pi_1 v$ dans u_1 et $\pi_2 v$ dans u_2 sont candidates pour être fondues en une série de jointures, éventuellement avec des restrictions et des simplifications supplémentaires.

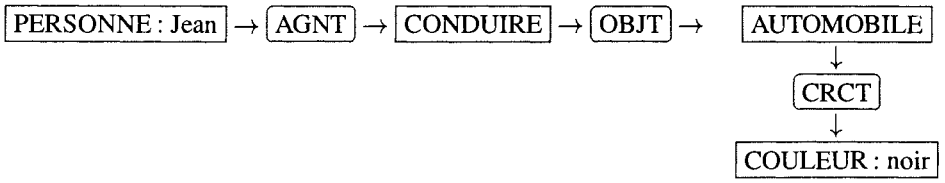
Si les graphes conceptuels u_1 et u_2 ont une généralisation commune v avec des projections compatibles $\pi_1 : v \rightarrow u_1$ et $\pi_2 : v \rightarrow u_2$, et si v' est un sous-graphe propre de v (c'est-à-dire $v \leq v'$), alors v' est aussi une généralisation commune de u_1 et de u_2 avec les projections compatibles $\pi_1 : v' \rightarrow u_1$ et $\pi_2 : v' \rightarrow u_2$. Les projections compatibles $\pi_1 v$ et $\pi_2 v$ sont dites être des **extensions** de $\pi_1 v'$ et $\pi_2 v'$.

Une jointure sur des projections compatibles étendues de façon maximale est appelée **jointure maximale**, elle s'applique au plus grand nombre de concepts communs aux deux graphes joints.

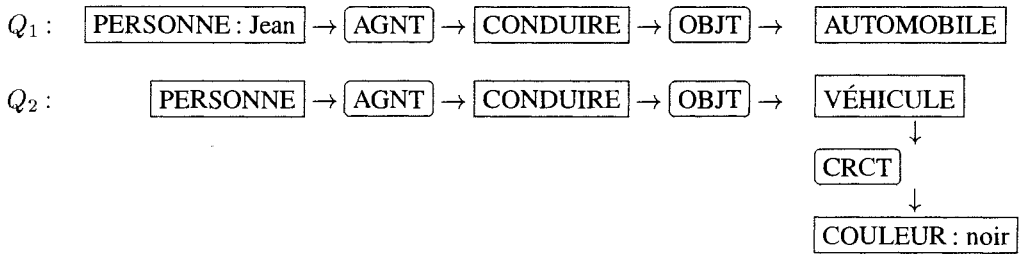
3.2.7 Utilisation de l'ordre sur les graphes pour évaluer la pertinence

Un graphe décrivant le contenu d'un document g_d est pertinent par rapport à un autre graphe décrivant la requête g_q si $g_d \leq g_q$. La pertinence peut donc être liée au fait que g_d et g_q ont les mêmes types de concepts, mais les concepts de g_d sont plus spécifiques que ceux de g_q.

L'hypothèse d'indexation aux feuilles du thésaurus est particulièrement appropriée vis-à-vis du formalisme des graphes conceptuels, car elle permet d'assurer que tous les concepts qui permettent d'indexer le document sont plus spécifiques que (ou tout au plus égaux à) ceux de la requête. En ce qui concerne la spécialisation par jointure, elle suppose que l'indexation des documents est plus «riche» que la description de la requête. Par exemple, le document D_1 dont le contenu est représenté par :



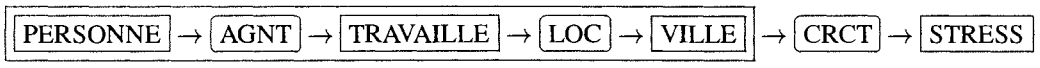
sera pertinent pour les deux requêtes :



La pertinence vis-à-vis de Q_1 est obtenue par spécialisation du graphe de description de D_1 par jointure, la pertinence vis-à-vis de Q_2 est obtenue par restriction du concept VÉHICULE au concept AUTOMOBILE.

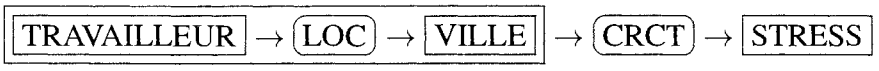
3.2.8 Contraction de graphes

La *contraction relationnelle* permet de remplacer un sous-graphe v d'un graphe conceptuel w avec une relation unique r et les concepts liés à ses arcs. Par exemple, le sous-graphe suivant correspond à l'énoncé «*le stress caractérise les personnes qui travaillent en ville*» :



Il est possible de contracter le sous-graphe

Le premier graphe peut alors être représenté par :



Le nouveau concept [TRAVAILLEUR] est un type construit, qui correspond à une «lambda abstraction» :

$$\lambda x \quad [\text{TRAVAILLEUR}] - [\text{PERSONNE}: *x] \rightarrow (\text{AGNT}) \rightarrow [\text{TRAVAILLE}]$$

De même, il est possible de construire un type [TRAVAILLEUR-URBAIN] qui correspondrait à [TRAVAILLEUR] → (LOC) → [VILLE]. Dans la hiérarchie des types, TRAVAILLEUR-URBAIN est considéré comme plus spécifique que TRAVAILLEUR, il est en effet obtenu par spécialisation de la relation, c'est-à-dire grâce à une jointure. De la même façon, il est possible de définir le type [NOMBRE-POSITIF] par

$$\lambda x\ [\text{NUMBER: } *x] \rightarrow (\geq) \rightarrow [\text{NUMBER: } 0]$$

La contraction de graphe s'appuie sur la *définition de type* qui déclare qu'un nom de t est défini par une abstraction unaire λau, notée **type** $t(a)$ **is** u, où u est le corps ou le *differentia* de t, et $type(a)$ est appelé le *genre* de t.

Nous donnons ci-après le détail de l'algorithme de contraction :

> Si $b_1, b_2, \ldots, b_n$ sont n concepts distincts de v, v n'ayant pas d'arcs liés aux concepts qui sont dans $w - v$, et u une copie de v avec les concepts $b_1, b_2, \ldots, b_n$ remplacés par les concepts génériques $a_1, a_2, \ldots, a_n$ où chaque b_i est un sous-type de a_i, alors la contraction relationnelle consiste à :
>
> 1. supprimer tout de v dans w, sauf les concepts $b_1, b_2, \ldots, b_n$,
> 2. si $type(r) = \lambda a_1, a_2, \ldots, a_n u$, pour chaque type i, lier l'arc i de r au concept b_i.

La contraction relationnelle d'un graphe canonique donne un graphe canonique. Elle permet de supprimer les sous-graphes qui peuvent être reconstitués à partir de l'information différentielle.

La notion de définition de type permet également d'introduire une définition duale de la contraction : l'expansion de type. L'**expansion minimale de type** consiste à joindre deux graphes u et v, tels que u est un graphe canonique contenant le concept a, v est un graphe canonique contenant le concept b, et $type(a) = \lambda bv$, sur les concepts a et b. Par exemple, l'expansion de

$$\text{u}: [\text{TRAVAILLEUR-URBAIN : Nancy}] \rightarrow (\text{CRCT}) \rightarrow [\text{STRESS}]$$

si [TRAVAILLEUR-URBAIN : *x] est la contraction de

$$[\text{TRAVAILLEUR}] \rightarrow (\text{LOC}) \rightarrow [\text{VILLE : *x}]$$

donne

L'**expansion maximale** commence avec une expansion minimale de type, mais y ajoute les étapes suivantes : la jointure de a et de b est étendue à une jointure maximale, puis le label de type du concept a est remplacé par le label de type t, où $type(a) \leq t \leq type(b)$.

Si le graphe original sur lequel on réalise l'expansion est vrai, l'expansion minimale préserve sa vérité, alors que l'expansion maximale est seulement plausible.

3.2.9 Négation et graphes conceptuels

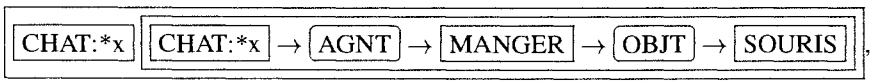
Les graphes conceptuels constituent un formalisme puissant en ce qu'ils permettent de représenter la négation.

Nous rappelons que, pour Sowa, un concept de type **proposition** est un graphe conceptuel dont le référent est un ensemble de graphes conceptuels : $\boxed{PROPOSITION : \{u_1, \ldots u_n\}}$. Chaque graphe conceptuel présent dans l'ensemble référent est dit affirmé par la proposition. La relation unaire (NEG) note une proposition niée, qui correspond à la négation de la conjonction des graphes conceptuels du référent de la proposition. Cette relation est abrégée par le symbole $\neg$. Ainsi, $\neg[u, v]$ est une abréviation de $(NEG) \rightarrow [PROPOSITION : \{u, v\}]$.

La négation permet d'exprimer le quantificateur existentiel $\forall$. Par exemple, si

$$\exists x(CHAT(x) \wedge SOURIS(y) \wedge MANGE(x, y))$$

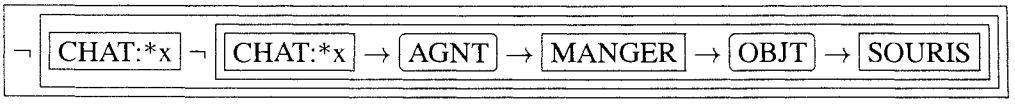
est représenté par



la proposition *tous les animaux mangent des souris* :

$$\forall x(CHAT(x) \wedge SOURIS(y) \wedge MANGE(x, y))$$

peut être représentée par



(il n'existe pas de chat qui ne mange pas de souris).

3.2.10 Notion de contexte et graphes conceptuels

Dans le formalisme des graphes conceptuels, un **contexte** est défini comme un concept dont le champ référent contient des graphes conceptuels imbriqués [Sowa 95]. Par exemple, dans la figure 3.2 (tirée de [Sowa 95]), le graphe représentant *un chat chassant une souris* est imbriqué dans un concept de type *Situation*.

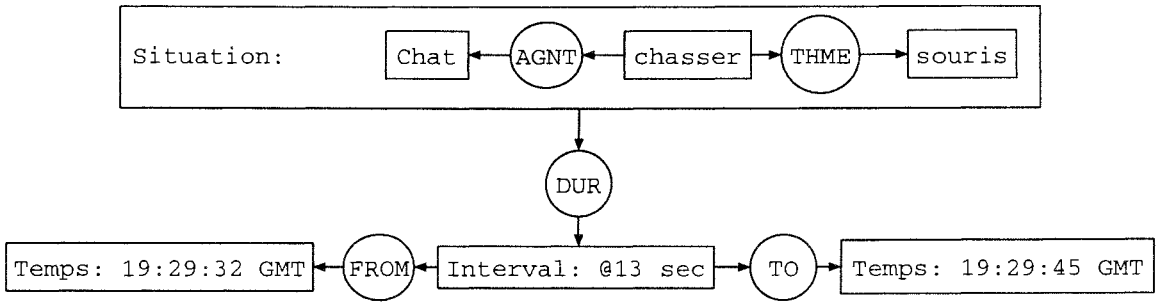


FIG. 3.2 – *Graphe conceptuel pour «un chat a chassé une souris durant 13 secondes de 19:29:32 GMT à 19:29:45 GMT».*

Dans la définition que Sowa donne d'une proposition (concept dont le référent est un ensemble de graphes conceptuels $u_1, \ldots, u_n$), chaque graphe u_i du référent de la proposition est dit se trouver dans le contexte de la proposition. Chaque contexte est aussi un concept, il peut avoir un label de type, des liens de co-référence et des relations conceptuelles qui lui sont attachées.

Sowa introduit la notion d'imbrication et de profondeur de contexte :

- le contexte le plus extérieur, c'est-à-dire la collection de graphes conceptuels qui ne se trouvent pas dans le champ référent d'un autre graphe, est imbriqué à la profondeur 0.
- Si x est un contexte négatif qui se trouve à la profondeur n, alors tout graphe ou concept qui se trouve dans le contexte de x est dit imbriqué à la profondeur $n + 1$.
- Pour tout entier $n \leq 0$, un graphe ou concept de profondeur $2n$ est dit imbriqué de façon paire, et un graphe ou concept imbriqué à la profondeur $2n + 1$ est dit imbriqué de façon impaire.
- Si un contexte y se trouve dans un contexte x, alors on dit que x *domine* y. La domination est transitive pour l'imbrication: si le graphe y domine un autre contexte z, alors x domine aussi z.

3.2.11 Méta-représentation des graphes conceptuels

Pour Wermelinger [Wermelinger 95], les types de concepts sont de deux sortes : les concepts relationnels ($\mathcal{T}^r$) et les concepts non relationnels ($\mathcal{T}^{nr}$). Les concepts non relationnels recouvrent la notion de concept de Sowa, alors que les concepts relationnels recouvrent les relations conceptuelles (qui sont représentés graphiquement dans une boîte ovale ou ronde). Par

exemple, le concept relationnel binaire AGENT sera représenté par $\rightarrow$ (AGNT) $\rightarrow$. Cette appellation nous semble moins ambiguë que celle de «relation conceptuelle» : elle ne peut en effet être confondue avec la dénomination de relation qui, dans le vocabulaire des bases de données, désigne un n-uplets d'attributs. Nous utiliserons donc la dénomination *concept relationnel* plutôt que relation conceptuelle par la suite.

La classification des concepts tient compte de leur ordre. Un type d'ordre supérieur dénote un ensemble de types d'ordre inférieur : chaque type de concept peut être décrit plus précisément par un ensemble de concepts d'ordre inférieur. Il est possible de noter l'ordre d'un ensemble de concepts par un chiffre en bas à droite de l'ensemble. Par exemple, si nous imaginons que les concepts *chat*, *chien*, *arbre* sont d'ordre 1 (ils appartiennent à $\mathcal{T}_1$), les concepts *espèce*, *forme* sont d'ordre 2, le concept *catégorie* est d'ordre 3.

3.3 Modélisation des connaissances d'un SRI sous forme de graphes conceptuels

Les principales notions de la théorie des graphes conceptuels étant présentées, nous allons voir comment elles s'adaptent à notre situation. Nous avons défini intuitivement une facette comme un groupe d'attributs coordonnés. Nous allons définir plus précisément comment représenter les documents, leurs descriptions et le thésaurus selon le formalisme de graphes conceptuels.

3.3.1 Les documents

Les **documents** sont repérés par un identificateur unique pour la collection à laquelle ils appartiennent. Ces identificateurs peuvent correspondre, dans la terminologie des graphes conceptuels, à des marqueurs. La collection de documents, pour le système de recherche d'information, correspond donc à un ensemble de marqueurs qui sont les identifiants des documents.

Un **segment physique de document** est identifié par le document auquel il appartient, et les limites qu'il occupe dans le document. Pour une image, ces limites sont spatiales : elles représentent par exemple le coin supérieur gauche et le coin inférieur droite d'un rectangle délimitant une portion de l'image. Pour une vidéo, les limites sont temporelles ; un segment est constitué d'un intervalle temporel relatif au document, cet intervalle étant repéré par un numéro d'image de début et un numéro d'image de fin. Cette unité peut résulter de la conversion d'un code temps (*time code* en anglais), constitué d'heures (h), des minutes (mn), et des secondes (sec)suivi du numéro d'image (nbi). Ainsi, le code 01:20:35:17 permet d'identifier l'image qui se trouve après une heure vingt minutes et trente cinq secondes du début, et qui est la dix-septième image de cette seconde. Cette unité est convertie en un numéro d'image par la formule simple :

$$h \times 3600 + mn \times 60 + sec \times nbips + nbi$$

où $nbips$ est le nombre d'images par secondes pour ce document défini par le standard utilisé (24 le plus souvent ou 30 pour le standard vidéo NTSC).

3.3.2 Les descripteurs

Nous supposons que les documents sont décrits par un ensemble de graphes conceptuels.

Deux types de termes peuvent être utilisés pour décrire un document. D'une part, il est possible d'utiliser des noms communs ; par exemple, la photographie ci-contre aurait pour descripteurs `comtesse`, `garçon`. Si $\mathcal{T}_n^{nr}$ représente l'ensemble des concepts non relationnels d'ordre n, ces termes descripteurs sont des concepts non relationnels d'ordre 1 dans le formalisme des graphes conceptuels ; si TD est l'ensemble des termes descripteurs,

$$\mathcal{T}_1^{nr} \subset TD.$$

FIG. 3.3 – *«Comtesse portant son fils à califourchon»*, Nadar.

Le deuxième type de descripteur correspond aux noms propres, c'est-à-dire l'ensemble $\mathcal{T}_0^{nr}$ qui représente l'ensemble des «individus» au sens large, cette notion ne recouvrant pas seulement des personnes. Les individus (notion qui était capturée dans le prototype RIVAGE par le champ "Précision sur l'objet") regroupent l'ensemble des noms propres utilisés pour décrire les images : noms de personnes, de lieux, de monuments, etc. Le formalisme des graphes conceptuels nous permet d'établir des relations entre les individus et des concepts plus généraux ; par exemple, il est possible de représenter explicitement que `Mistinguett` (individu, concept non relationnel d'ordre 0) est de type `FEMME` ou `ACTRICE` (concepts non relationnels d'ordre 1) : [FEMME : Mistinguett]. Cette notation est une contraction du graphe :

[FEMME : # 167] → (A POUR NOM) → [NOM: Mistinguett]

Les connaissances dont le SRI dispose sur les individus peuvent être représentées de plusieurs manières : soit une base de connaissance distincte de la base de descriptions donne des informations sur les individus ; soit les informations sur les individus sont rattachées à la base de description.

La première solution permet d'éviter toute redondance : un individu est caractérisé une fois pour toutes par une description extérieure aux descriptions de documents, et les descriptions de documents font référence à cet individu. Cette solution peut toutefois introduire des ambiguïtés. Un même individu peut en effet occuper tour à tour des fonctions différentes ; par exemple, *Ronald Reagan* a été *acteur* et *Président de la République*. Il peut être utile, pour la description d'une photographie ou d'un reportage télévisé, d'indiquer précisément quelle est la fonction qu'il occupait lors de la prise de vue. Une solution peut consister à enregistrer, avec

les connaissances sur l'individu, non seulement les fonctions mais aussi les périodes auxquelles il les a occupées. Ainsi, le rapprochement entre la date de prise de vue et la période associée à chaque fonction pourrait permette de lever l'ambiguïté. Mais si une personne a occupé plusieurs fonctions durant une même période, ce type de déduction échoue. C'est pourquoi il vaut mieux spécialiser la description elle-même ; dans l'exemple que nous avons présenté, cela consiste à préciser la fonction de la personne :

$$\boxed{\text{PERSONNE : Reagan}} \rightarrow \text{(CRCT)} \rightarrow \boxed{\text{FONCTION : président}}.$$

Pour la facette *contenu*, les concepts sont de type PERSONNE, OBJET, ACTION, LIEU, TEMPS, ou CARACTÈRE. Le concept TEMPS peut être décrit par un point de temps (PTMP) comme une date ou une heure par exemple, ou par un intervalle de temps (ITMP) repéré par un moment de début et de fin [(DÉBUT), (FIN)].

3.3.3 Les concepts relationnels

Les relations conceptuelles qui peuvent exister entre les descripteurs des documents dépendent du type de la facette à laquelle ces descriptions sont attachées.

Une liste de concepts relationnels pouvant être retenus pour la description du contenu du document par des mots-clés peut être trouvée en annexe B de [Sowa 84]. L'ensemble $\mathcal{T}_1^r$ des concepts relationnels d'ordre 1 est donc pour la facette CONTENU : {Agent (AGNT), Objet (OBJT), Espace (SPCE), Temps (TMPS), Intervalle de temps (ITMP), Point de temps (PTMP), Manière (MANR), Moyen (MOYN), Instrument (INST), Caractéristique (CRCT)}.

Pour décrire des documents, la mise en œuvre de ces concepts relationnels implique une analyse assez détaillée du contenu. Toutes ces relations sont binaires, sauf la relation *Intervalle de temps* qui est ternaire.

3.3.4 Définition des descriptions de documents

Les descriptions de documents correspondent à un ensemble de graphes conceptuels, répartis par facettes. Les relations conceptuelles qui existent entre les descripteurs de la facette

CONTENU sont centrées autour du concept d'action.

type de concept	rôle dans une relation
Personne	agent de l'action objet de l'action
Objet	agent de l'action objet de l'action
Action	action
Lieu	espace de l'action
Temps	temps de l'action (moment précis ou période)
Manière	manière dont est effectuée l'action
Moyen	moyen dont s'effectue l'action
Instrument	instrument qui permet d'effectuer l'action

De plus, les personnes, objets, lieux, etc. décrits peuvent se voir attacher des caractéristiques. Cette analyse permet de définir un ensemble de graphes canoniques pour l'application.

Par exemple, nous pouvons spécifier que la relation conceptuelle AGENT d'arité 2 relie un concept de type PERSONNE à un concept de type ACTION ; ceci correspond au graphe :

[PERSONNE] → (AGNT) → [ACTION]

Les graphes primitifs qui constituent la base minimale canonique correspondant à la facette CONTENU sont :

[ENTITÉ] → (AGNT) → [ACTION]
[ACTION] → (OBJT) → [ENTITÉ]
[ACTION] → (LOC) → [LIEU]
[ACTION] → (DURÉE) → [INTERVALLE-TEMPS]
[ACTION] → (PTMP) → [TEMPS]
[ACTION] → (MANR) → [MANIÈRE]
[ACTION] → (MOYN) → [MOYEN]
[ACTION] → (INST) → [INSTRUMENT]
[ENTITÉ] → (CRCT) → [CARACTÈRE]

Le concept ENTITÉ est un concept générique qui recouvre PERSONNE et OBJET. Le concept [INTERVALLE-TEMPS] est un concept imbriqué ; il correspond à la contraction de :

INTERVALLE-TEMPS : [TEMPS] → (DÉBUT) → [INTERVALLE] → (FIN) → [TEMPS]

Voici quelques exemples de descriptions que l'on peut trouver dans la facette CONTENU :

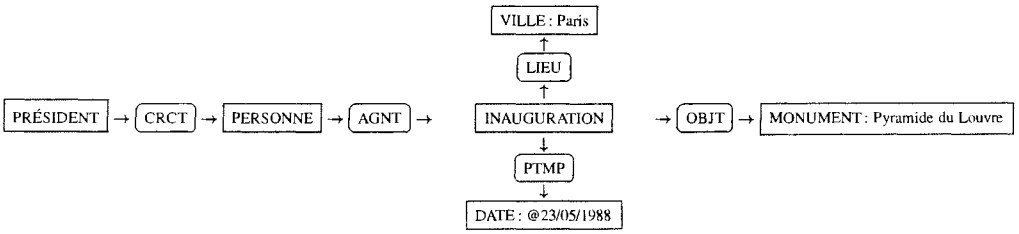
- [PERSONNE] → (AGNT) → [ACTION]
  Exemple : Inauguration par le président
  [FONCTION : président] → (CRCT) → [PERSONNE] → (AGNT) → [INAUGURATION]

- [ACTION] → (OBJT) → [OBJET]
  Exemple : Inauguration d'un monument
  [INAUGURATION] → (OBJT) → [MONUMENT]

- [ACTION] → (LOC) → [LIEU]
  Exemple : Inauguration à Paris
  [INAUGURATION] → (LOC) → [VILLE : Paris]

- [ACTION] → (PTMP) → [DATE]
  Exemple : Inauguration le 23/05/1988
  [INAUGURATION] → (PTMP) → [DATE : @23/05/1988]

L'événement *inauguration à Paris de la pyramide du Louvre par le président le 23/05/1988* sera donc décrit par le graphe conceptuel suivant dans la facette CONTENU :



3.3.5 Les attributs

Un attribut permet de donner un rôle à une partie de description, et d'établir une relation entre l'objet décrit et certains types de descripteurs. Ainsi, le terme *Montesquieu*, s'il est attaché à un attribut AUTEUR n'aura pas le même sens dans la description d'un document que s'il attaché à la facette CONTENU. Les noms des attributs peuvent donc être vus comme des **concepts relationnels** entre un ensemble de termes descripteurs et l'objet décrit. La hiérarchie qui existe sur ces concepts relationnels est représentée en figure 3.4.

Les relations que les attributs établissent entre un document (ou une portion de document) et les descripteurs associés sont binaires. Elles relient un segment de document à un domaine, qui recouvre un ensemble de descripteurs. Ainsi,

[SEGMENT] → (ATTRIBUT) → [ATTRIBUT-VALUÉ]

où [ATTRIBUT-VALUÉ : *x *y] –

[ATTRIBUT-VALUÉ] → (A-POUR-NOM) → [NOM-ATTRIBUT : *x]
[ATTRIBUT-VALUÉ] → (A-POUR-VALEUR) → [DOMAINE : *y]

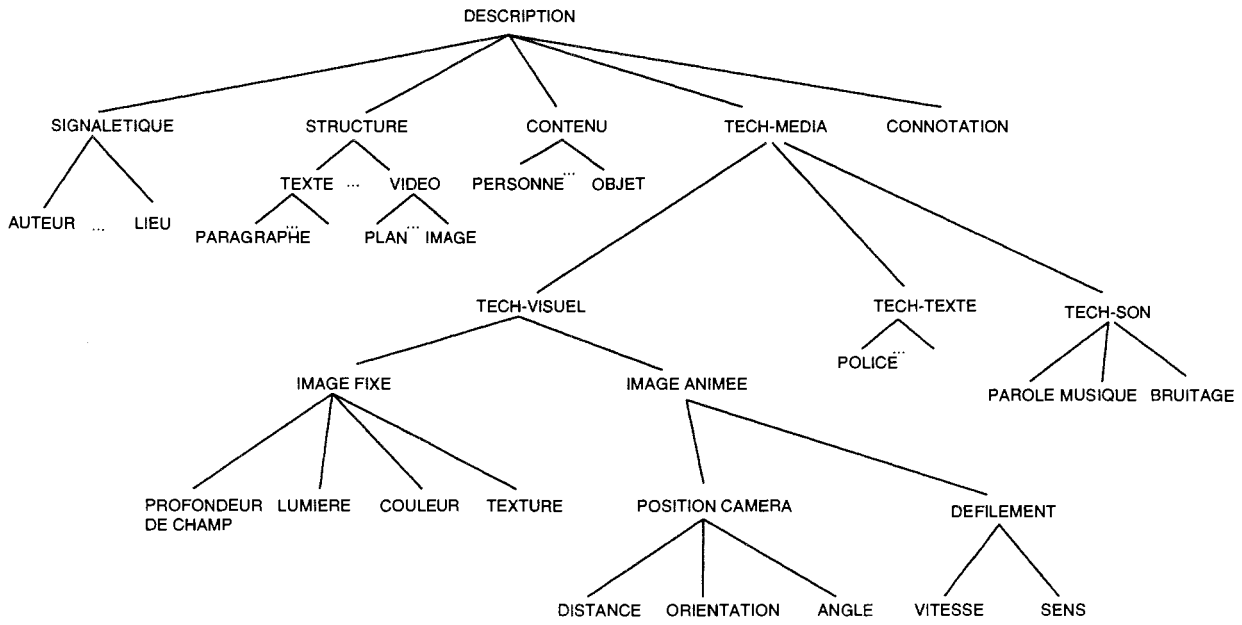


FIG. 3.4 – *Hiérarchie d'attributs de description des documents.*

Le fait de représenter les attributs sous forme de concepts relationnels permet d'envisager de pouvoir les traiter comme les descripteurs, en particulier pour la formulation de la requête, c'est-à-dire de permettre d'interroger le système à un méta-niveau. Par exemple, cela peut permettre de répondre à la requête : *contenu ?* pour savoir quels sont les aspects de contenu qui sont décrits.

Ainsi, l'ensemble des attributs $ATTR$ constitue, dans notre modèle, les concepts relationnels d'ordre 2 : $ATTR = \mathcal{T}_2^r$.

Pour la description des documents, les attributs peuvent être multi-valués : plusieurs relations peuvent être établies pour un même document autour du même attribut (toute occurrence d'une relation multivaluée peut être décomposée en occurrences de relations binaires).

Par exemple, le contenu de la photographie de la page 83 serait décrite, pour l'attribut PERSONNE, par : $\begin{cases} \boxed{\text{PERSONNE} : \texttt{comtesse}} \\ \boxed{\text{PERSONNE} : \texttt{garçon}} \end{cases}$ noté en raccourci : $\boxed{\text{PERSONNE} : \{\texttt{comtesse}, \texttt{garçon}\}}$

À noter que dans notre modèle, nous ne restreignons pas les valeurs des attributs au type *mot-clé*. Il peut s'agir également par exemple de l'ensemble des entiers naturels, ou d'un intervalle de valeurs numériques. La seule contrainte qui existe réellement sur les domaines des attributs est qu'ils soient définissables, en intension ou en extension, et qu'il existe une relation $\leq$ sur les valeurs du domaine. Nous verrons par la suite qu'un domaine peut se voir attacher un certain nombre de propriétés.

3.3.6 Les facettes

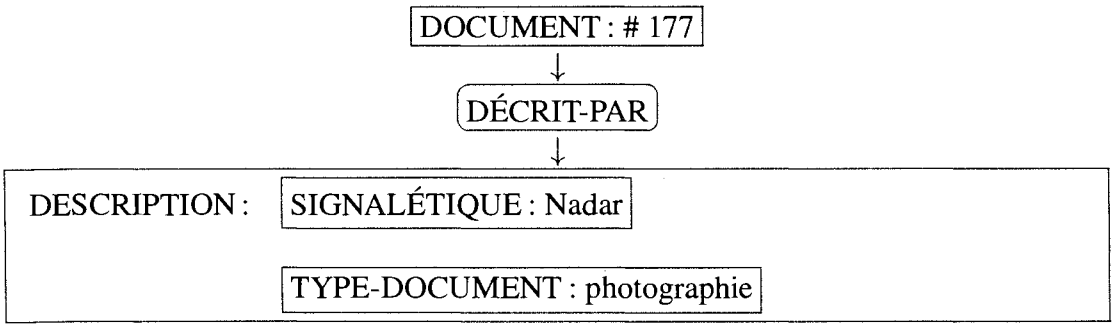
Une facette $\mathcal{F}$ permet de regrouper un ensemble d'attributs par type sémantique. Par exemple, la facette *contenu* comportera les attributs *personne*, *objet*, *lieu*, *période*, *événement*. Il s'agit d'un super-type du type ATTRIBUT, dans la hiérarchie des concepts relationnels. Ainsi, l'attribut AUTEUR peut être considéré comme de type SIGNALÉTIQUE.

Le SRI comporte par exemple les facettes suivantes :

- SIGNALÉTIQUE : auteur, date de production, lieu de production, titre ;
- CONTENU : personnes, objets, lieux, périodes de temps, points de temps, instruments, matières, moyens, caractéristiques ;
- TECHN-MEDIA : techn-image, techn-son, techn-video ; TECHN-VIDEO recouvre les attributs type de plan, angle-caméra, position-caméra ;
- CONNOTATION : connotation, critiques, annotations.

Le formalisme des graphes conceptuels nous permet donc de représenter formellement ce qu'est une facette de description.

Ainsi, TYPE-DOCUMENT est un type générique qui peut être décomposé en IMAGE-FIXE (PHOTOGRAPHIE, GRAVURE, DESSIN, ...), IMAGE-ANIMEE (qui recouvre VIDEO et ANIMATION), SON, et TEXTE. Par exemple, une photographie de Nadar, identifiée par le numéro 177 dans une collection, sera représentée par :



Une facette de description peut être représentée par une *proposition* au sens de Sowa : elle permet de regrouper des graphes conceptuels qui correspondent à des relations autour d'attributs de même type.

3.3.7 Le thésaurus

Le thésaurus est une structure qui regroupe des concepts et des relations entre ces concepts. Les différentes relations qui existent entre les concepts sont *spécifique-de*, *synonyme*, et *voir-aussi*.

Si T_{Descr} est l'ensemble des termes descripteurs, la contrainte d'indexation aux feuilles peut être spécifiée par :

$$T_{Descr} \cap TH = \{t \in TH | \nexists t' \in TH((\text{spécifique-de}(t) = t') \wedge (t' \neq \bot))\}.$$

soit

$$(\forall t, t' \in T_{Descr} \cap TH)(t \leq t') \supset (t = t')$$

(deux termes descripteurs ne peuvent pas être liés par la relation *spécifique-de* du thésaurus) où $\supset$ note l'implication matérielle [10]. La modélisation par graphe conceptuel est tout à fait adaptée à la représentation d'un thésaurus. Le thésaurus peut être vu comme une hiérarchie de concepts, les concepts étant reliés au sein de cette hiérarchie par des liens de spécialisation. La racine du thésaurus est le concept universel ($\top$) et, sous les feuilles du thésaurus, c'est-à-dire les termes plus spécifiques, on trouve le concept absurde ($\bot$), ce qui permet à deux termes d'avoir une borne inférieure commune (voir figure 3.5).

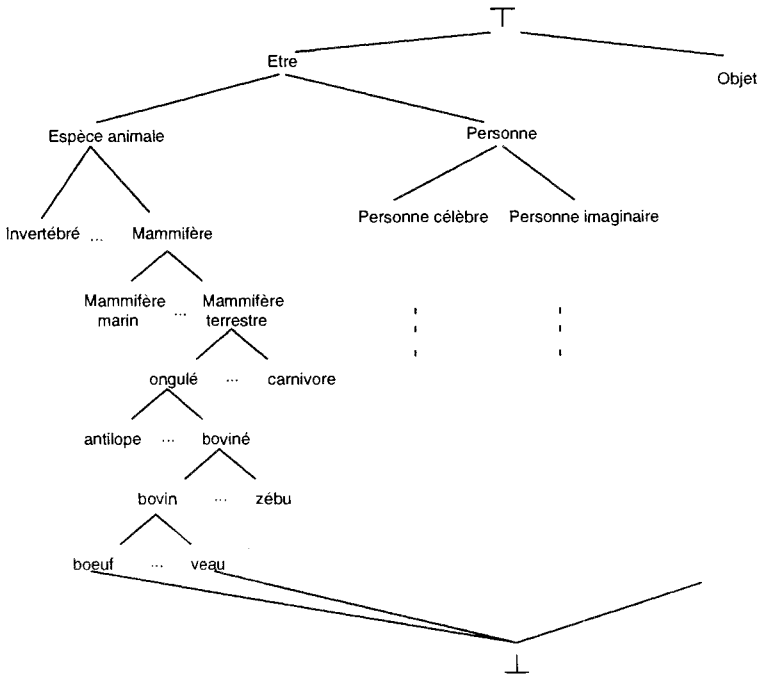


FIG. 3.5 – *Treillis «sorte-de» dans le thésaurus.*

La base canonique des graphes comporte :

$$\boxed{\text{terme}} \rightarrow \boxed{\text{a-pour-domaine}} \rightarrow \boxed{\text{domaine}}$$
$$\boxed{\text{terme}} \rightarrow \boxed{\text{sorte-de}} \rightarrow \boxed{\text{terme}}$$

Un champ sémantique est un treillis de termes tirés du thésaurus, et ordonnés par la relation *spécifique-de*. Un concept de type DOMAINE est la racine d'un champ sémantique :

$$\forall t \in \{domaine\} \nexists t' \text{ tel que } (specifique - de(t') = t) \wedge (t' \neq \top).$$

10. L'implication matérielle $(t \leq t') \supset (t = t')$ est interprétée : si $t \leq t'$ alors $t = t'$.

La relation de composition définit aussi un treillis de termes tirés du thésaurus, et ordonnés par la relation partie-tout : $\mathcal{T}_{part}$.

La relation *synonyme* peut également être considérée comme une relation d'ordre partiel sur un ensemble de synonymes, faisant correspondre à chaque terme d'une classe de synonymes le terme qui est employé à sa place. Dans ce cas, le thésaurus comporte un autre treillis dont le graphe canonique est $\boxed{\text{terme}} \rightarrow (\text{synonyme}) \rightarrow \boxed{\text{terme}}$.
Une classe de synonymes Syn est un treillis de termes liés par une relation *synonyme*, satisfaisant :

$$Syn \subseteq (T_{Descr} \cup T_{nonDescr})$$
et
$$\forall t \in T_{nonDescr}, \exists t' \in T_{Descr}(synonyme(t) = t').$$
et
$$\forall t \in T_{nonDescr}((synonyme(t) = t') \wedge (synonyme(t) = t'') \supset (t' = t''))$$

D'autres relations peuvent exister au sein d'un thésaurus. La relation *voir-aussi* englobe toutes sortes de relations comme nous l'avons détaillé au § 1.3.2 : elle peut être décomposée en *cause*, *origine*, *localisation*, etc. Ces relations ne sont pas des relations d'ordre partiel sur les termes, puisqu'elles ne sont pas transitives. De plus, nous avons montré au § 1.3.2.0 (page 20) que la relation d'antonymie est à exclure de l'ensemble des relations *voir-aussi* car elle ne possède pas les mêmes propriétés. Par contre, ces connaissances peuvent constituer une base de connaissance distincte, une sorte de bases de liens, que le SRI pourra exploiter en dernière ressource lors de la résolution de problème. La hiérarchie des relations permet de spécialiser la relation *voir-aussi* en *cause*, *effet*, *origine*, *localisation*, etc. Ces relations doivent être examinées plus avant pour permettre de constituer un treillis. Par ailleurs, il est également possible de définir des concepts relationnels d'ordre supérieur. Par exemple, la relation $INVERSE-DE$, qui est une relation entre deux concepts relationnels du premier ordre, est une relation du second ordre :

INVERSE-DE(spécifique-de) = (générique-de).

Les noms propres pourraient être intégrés au thésaurus. Ils formeraient alors un réseau de concepts lié aux termes descripteurs : chaque nom propre renvoie en effet à un ensemble de concepts.

En résumé, un thésaurus peut être considéré comme un ensemble composé de trois treillis distincts : $\mathcal{T}_{spe}$ le treillis générique/spécifique, $\mathcal{T}_{part}$ le treillis partie-tout, $\mathcal{T}_{syn}$ le treillis des synonymes. Il peut comprendre, de plus, une base des relations entre termes associés $\mathcal{B}_{ta}$.

3.4 Travaux liés

3.4.1 Autres définitions de facettes

La notion de facette existait déjà dans le prototype RIVAGE [Halin 89], où elle permettait de regrouper plusieurs champs de description des images pour la formulation de la requête par l'utilisateur. Toutefois, le concept de facette était uniquement relié à l'interface utilisateur et n'était pris en compte ni pour la description des documents, ni par le processus de recherche proprement dit. Aucune inférence n'était donc possible à partir de cette structuration.

Ranganathan a utilisé la notion de facette pour nommer les différentes classes de descriptions situationnelles de documents. Constatant que les systèmes classificatoires en vigueur au milieu du siècle étaient assez «figés» - l'introduction d'une nouvelle connaissance non seulement était difficile mais pouvait avoir d'énormes répercussions sur le système tout entier - , il avait proposé de ré-examiner les systèmes statiques de classification traditionnelle des connaissances pour la recherche documentaire en considérant l'univers de la connaissance comme un continuum dynamique [Ingwersen 92b].

Contrairement aux classifications catégorielles déterminées par exemple par les relations générique-spécifiques d'un thésaurus, où les relations concernent essentiellement des substantifs, la notion de facette utilisée par Ranganathan permet d'envisager une classification situationnelle guidée par les actions. Le schéma de facettes de Ranganathan est très proche des grammaires de cas de Fillmore [Fillmore 68]. Il repose sur les concepts d'action, d'agent, d'objet, d'instrument et de lieu de l'action. Les termes sont regroupés en classes (PMEST = Personne, Matière, Énergie, eSpace/Temps), chaque classe étant associée à un rôle, ce qui permet de décrire plus finement le contexte. C'est un moyen supplémentaire de structurer la connaissance que l'on possède sur les documents.

La notion de facette que nous utilisons va un peu plus loin : non seulement elle doit permettre de représenter les concepts en situation, mais elle permet aussi de ne traiter qu'un aspect des connaissances sur les documents, qu'il s'agisse de la représentation des informations données par le document ou de l'interprétation d'une requête de l'utilisateur. Le formalisme que nous utilisons permet d'avoir des descriptions de profondeur variable pour les différents aspects du documents. Pour l'application de recherche de documents vidéos que nous envisageons, la recherche de séquences d'actualités télévisées, la prise en compte de la situation est particulièrement importante. En effet, ce sont des événements qui doivent être décrits. Essayer d'obtenir une description complète des événements par une simple liste de mots-clés relève du tour de force. La notion d'événement est en effet plus complexe à décrire que celle d'objet représenté : elle correspond à une relation entre des concepts.

3.4.2 RIME et Hyper-RIME

La nécessité de travailler sur des représentations de documents captant plus de sémantique s'est déjà imposée dans des travaux précédents. C'est le cas notamment dans des domaines

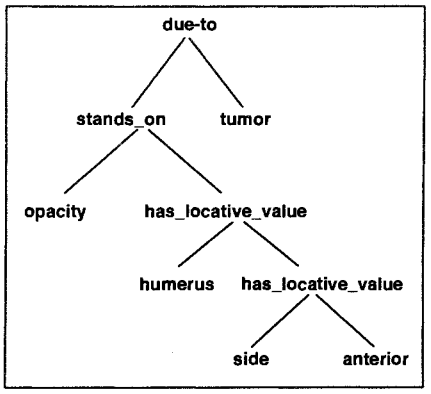


FIG. 3.6 – *Représentation dans RIME de «Opacité sur le côté antérieur droit de l'humerus dû à une tumeur».*

d'applications très spécialisés comme le domaine médical, où s'est posé très vite le problème de travailler sur un vocabulaire unifié et où des structures de classification des dénominations ont été très rapidement créées, comme MeSH. Dans RIME (Recherche d'Informations MEdicales) [Berrut 90], l'objectif est de retrouver des images médicales (radiographies ou images de scanners) grâce aux rapports médicaux qui y sont associés. L'indexation automatique analyse les rapports médicaux en langage naturel ; elle construit des arbres binaires qui représentent la coordination entre des termes. Cette approche s'appuie sur une grammaire sémantique. Les nœuds terminaux d'un arbre représentent les termes techniques du domaine, les nœuds non terminaux correspondent à des opérateurs sémantiques qui explicitent des relations particulières entre les concepts de plus bas niveau (voir l'exemple en figure 3.6). Cette approche convient particulièrement au domaine d'application car le vocabulaire technique utilisé dans les rapports est très spécialisé et limité en volume, et le style des rapports simple et direct.

Un modèle unifié de représentation par graphes conceptuels des rapports décrivant les documents et des images correspondantes a été développé par [Kheirbek 95] ; ce modèle unifié a permis l'intégration du SRI et d'un système hypermédia. Nous reviendrons sur cette approche au § 4.4.7.

3.4.3 ELEN, recherche de composants en génie logiciel

À notre connaissance, la modélisation par graphes conceptuels a été adoptée pour la première fois en RI dans [Chevallet 92]. Dans ces travaux, documents et requête sont représentés par des réseaux de concepts, selon le formalisme des graphes conceptuels [Sowa 91]. Des graphes canoniques représentent les combinaisons de concepts et de relations conceptuelles autorisées. Pour l'indexation, les graphes sont normalisés : ils ne peuvent être simplifiés et ils représentent une classe d'équivalence de graphes égaux pour la relation d'ordre considérée sur les graphes.

Les concepts sont organisés en taxonomie, avec quatre catégories de base pour les termes : noms, verbes, adverbes, adjectifs. Les catégories des objets interrogeables sont des sous-catégories des noms. Ex : dans l'application dédiée à la recherche de composants réutilisables en

génie logiciel, les objets interrogeables sont "procédure", "fonction" ... La catégorie des verbes est divisée en intransitifs, transitifs directs, transitifs indirects, ces derniers se subdivisant à nouveau en fonction des prépositions. Les catégories de liens retenues sont :

- objet-action = manière de lier un verbe à ses compléments (Quoi ?, AQuoi ? SurQuoi ?)
- complément-nom = liaison entre deux substantifs (De, Dans, ParamètreDe)
- auteur-action = "permetDe"
- action-action = relation entre deux verbes "dansLeButDe"

Par rapport au modèle logique de RI, documents et requête représentent des mondes possibles. A chaque monde est associé un graphe conceptuel. Une opération de construction permet de construire tous les graphes qui ont un sens. La mise en correspondance calcule la distance sémantique entre un document et une requête à partir de l'opération de projection de graphes.

Ce modèle a été implanté dans le prototype ELEN de façon efficace en utilisant la technique des signatures. Cette technique de représentation optimisée est souvent utilisée pour les graphes [Pair 88]. Chaque graphe de la base canonique est représenté par un code ayant pour longueur le nombre maximal de graphes à coder. Cela revient à associer chaque graphe de la base canonique à une position du code de représentation des graphes ; le code d'un élément comporte un 1 dans la position qui est affectée à cet élément, à 0 dans toutes les autres positions. Par exemple, si la base canonique contient quatre graphes canoniques, le code sera de longueur 4 :

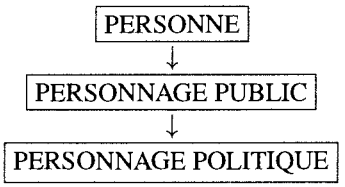
Graphe conceptuel	Signature
PERSONNE → (AGNT) → ACTION	0001
ACTION → (OBJT) → ENTITE	0010
ACTION → (LOC) → LIEU	0100
ACTION → (TIME) → TEMPS	1000

Dans ELEN, chaque graphe est représenté par une signature de jointure et une signature de restriction. Chaque graphe canonique défini a une signature de jointure unique, selon la méthode que nous avons exposée au paragraphe précédent. La signature du graphe correspondant à la jointure de deux autres graphes correspond alors à l'addition de leurs signatures de jointure. Cette représentation provoque une certaine perte d'information, puisque l'opération de projection qui permet de faire l'opération réciproque de jointure, a plusieurs résultats possibles. La signature de restriction représente à quel point les concepts du graphe peuvent être spécialisés. La longueur du code représentant la signature de restriction est égale au nombre de type de concepts minimaux. Chaque position de la signature de restriction correspond à un concept

minimal, il faut donc considérer l'ensemble des termes les plus spécifiques comme une liste ordonnée. A chaque position est affecté un entier qui représente la hauteur dans l'arbre du concept le plus générique autorisé. Par exemple, voyons comment est représenté le graphe :

$$\boxed{\text{PERSONNE}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{ACTION}}.$$

Supposons qu'il faille traverser deux relations depuis le concept PERSONNE pour arriver à PERSONNAGE POLITIQUE, qui est un concept terminal (une feuille) du thésaurus, et le concept PERSONNE n'a pas d'autre spécifique à une plus grande profondeur dans l'arbre :



La hauteur du concept PERSONNE est de 2. La position affectée au concept PERSONNE vaudra 2 + 1 = 3. Si les spécifiques d'un concept développent des branches de profondeurs diverses, c'est la profondeur maximale qui sera considérée. Ainsi,

$$\text{val-position(concept)} = max_i(\text{distance(concept,}spe_i)) + 1$$

où $val - position(concept)$ représente la valeur de la position du concept,
distance(concept, spe_i) représente le nombre de relations à traverser
entre un concept générique et son $i$ème terme le plus spécifique dans le thésaurus.

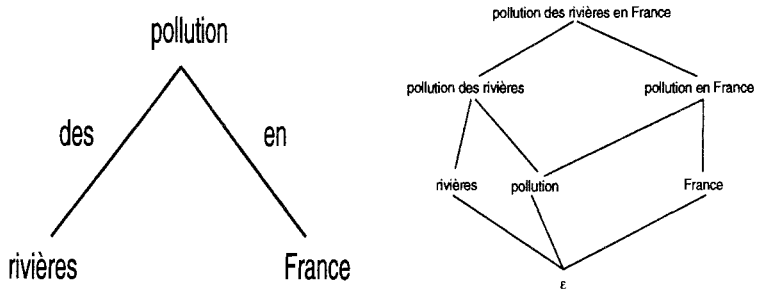
Si deux concepts sont les mêmes, leur distance de restriction est nulle. Ils en est de même s'ils ne sont pas comparables ; c'est le cas par exemple de deux termes appartenant à deux champs sémantiques distincts du thésaurus, comme `armoire` et `mouton`.

Il existe entre les signatures un ordre partiel isomorphe à celui qui existe entre les graphes.

L'interprétation de la requête consiste à la représenter par un graphe conceptuel unique non ambigu. Pour la résolution de la requête, la forme des informations (nom des fonctions, des méthodes, des paramètres, etc.) est distinguée du contenu (ce que fait la méthode).

Treillis d'inférence

Bruza [Bruza 91] a proposé une représentation reposant sur des expressions d'index. La théorie utilisée repose sur des symboles, les termes et les connecteurs entre les termes, des formules bien formées représentées par les expressions d'index, et trois règles d'inférence : le *modus continuens*, le *modus generans* et le *modus substituens*. Le *modus continuens* permet de déduire toute sous-expression d'une expression composée ; par exemple, `pollution` peut être déduit de `pollution des rivières`. Le *modus generans* est une déduction par généralisation : il est possible de déduire un générique à partir d'un de ses spécifiques ; par



(a) Expression d'index (b) Treillis d'inférence

FIG. 3.7 – *Représentation de «pollution des rivières en France».*

exemple, `saumon` permet de déduire `poisson`. Le *modus substituens* permet des déductions par substitution : par exemple, tout document sur la pollution des rivières en France traite aussi des effets de la pollution en France. Ainsi la recherche d'information s'apparente à un mécanisme de preuve automatique par transformations successives des expressions. Un document D a une sémantique $\chi(D)$ qui lui est associée et qui correspond à un ensemble d'axiomes (figure 3.7.a). Une requête q qui caractérise un besoin d'information N peut être notée $q = \chi(N)$. La probabilité pour qu'un document D soit pertinent vis à vis d'une requête est inversement proportionnelle à l'extension minimale des axiomes de D pour permettre de prouver la requête (principe de l'extension axiomatique minimale). Contrairement à l'approche de Nie [Nie 88] qui propose d'utiliser une base de connaissance pour étendre l'ensemble des axiomes, les auteurs choisissent d'exploiter les propriétés structurelles des expressions d'index et introduisent l'opération de *raffinement* pour rendre ces expressions plus spécifiques. L'opération de raffinement permet de parcourir le treillis de concepts en ajoutant de l'information à chaque étape : par exemple, le concept `pollution` est raffiné en `pollution des rivières`, lui-même raffiné en `pollution des rivières en France`. Le raffinement est vu comme un mécanisme d'inférence plausible. La *distance d'évolution* $\delta(I, J)$ entre les expressions d'index est le nombre minimal d'étapes d'inférence plausible nécessaires pour en déduire l'une de l'autre. La similarité $\sigma(I, J)$ entre deux expressions d'index I et J est alors inversement proportionnelle à la distance d'évolution $\delta(I, J)$. Par exemple, $\sigma(I, J) = 2^{-\delta(I,J)}$. La pertinence probable d'un document D pour une requête q est établie en déduisant q à partir d'un axiome unique de D :

$$P_{rel}(D, q) = max_{I \in \chi(D)} \sigma(I, q).$$

Cette approche est séduisante pour le traitement des documents textuels. L'indexation automatique des documents sous cette forme de représentation qui repose sur la détection de prépositions particulières (*de, en, par, ...*) pour générer de nouvelles branches est relativement simple. Toutefois, des cas d'ambiguïtés peuvent se produire, surtout dans les langues où ces prépositions ont des sens multiples. Les auteurs l'ont d'ailleurs appliquée en priorité aux titres et aux en-têtes de sections significatives des documents. Pour la recherche d'images fixes, ou animées, elle pourrait être appliquée aux légendes qui les accompagnent. Enfin, elle est applicable également au calcul d'une distance entre les termes d'un thésaurus.

3.5 Conclusion

Nous avons présenté dans ce chapitre le formalisme des graphes conceptuels et avons montré comment ce formalisme convenait pour représenter notre modèle, les connaissances qu'il utilise et les méta-données comme les attributs ou la notion de *facette* pour décrire les documents. En particulier, il permet de décrire des événements et des objets dans leur contexte (agent, objet, lieu, espace, temps) et d'attacher des qualificatifs et des caractéristiques aux objets dans les descriptions de documents. De plus, la récursivité de ce formalisme, qui permet d'exprimer des concepts complexes et de les traiter ensuite comme un concept simple, semble prometteuse pour traiter les connaissances complexes qui peuvent être introduites dans un SRI. Toutefois, nous n'exploitons pas cette récursivité pour l'instant. Nous nous contentons pour le moment de types de concepts définis dans des facettes distinctes, et de relations conceptuelles propres à chaque facette.

> «*Un modèle idéal de document devrait permettre une représentation multi-niveaux, incluant des vues conceptuelle, logique et de la forme du document. Ces différentes vues devraient être moulées dans un cadre unique de représentation, traitant les diverses représentations du document et leurs composants comme des objets et utilisant des mécanismes d'abstraction pour organiser la base de documents.* » (traduit de [Meghini 91])

Le modèle de documents que nous proposons répond en partie à cette constatation des chercheurs impliqués dans le projet Esprit Multos (MULTimedia Office Server) de 1986 à 1990, qui a donné lieu à la construction d'un prototype pour l'enregistrement et la recherche de documents multimédias. La contre-partie de la richesse de ce modèle est la complexité de traitement qu'il peut impliquer.

Nous allons analyser dans le prochain chapitre un processus de recherche d'information interactive afin de déterminer comment peut être exploité ce formalisme de représentation à chaque phase du processus.

Chapitre 4

Primitives génériques d'un processus de recherche d'information

La plupart des processus de recherche d'information «intelligents et interactifs» sont hybrides : ils utilisent conjointement plusieurs tactiques et quelquefois plusieurs modèles de mise en correspondance pour améliorer la recherche. Nous présentons ici SMART et SIRE, qui représentent deux approches différentes pour la mise en œuvre conjointe de tactiques (§ 4.1.1). Ces SRI sont destinés à la recherche de textes. Nous décrivons ensuite le processus EXPRIM de recherche interactive d'images développé dans notre équipe (§ 4.1.2), et le modèle paramétré *[Smaïl 94] qui le complète et le généralise (§ 4.2). Nous proposons dans cette thèse de compléter le modèle paramétré (§ 4.3) d'une part pour exploiter le formalisme de représentation que nous avons choisi, et d'autre part pour tenir compte du contexte distribué dans lequel se situent les nouveaux SRI. Ceci nous amène à faire des propositions pour mettre en œuvre le bouclage de pertinence dans le formalisme des graphes conceptuels (§ 4.3.12). Puisque nous avons choisi de considérer les documents vidéo comme des documents structurés, nous analysons enfin l'impact de la structure des documents sur le processus de recherche (§ 4.4).*

4.1 Quelques modèles hybrides de processus de recherche d'information

4.1.1 Processus exploitant le modèle vectoriel : SMART et SIRE

Le **système SMART** date des années 1960 [Salton 65], mais il sert encore de banc d'essai pour l'implantation de nombreuses méthodes de recherche d'information. Il utilise des méthodes d'indexation automatique pour affecter des identificateurs de contenu aux documents et aux requêtes qui sont exprimées en langage naturel. Parmi les termes retenus pour l'indexation, les termes trop ou pas assez fréquents sont éliminés. Les termes trop fréquents sont combinés avec d'autres termes en syntagmes, ce qui les rend plus spécifiques. Les termes trop spécifiques (qui se retrouvent seulement dans un ou deux documents) sont généralisés avec l'aide du thésaurus.

Les documents sont rassemblés dans des classes par sujet commun : un calcul de similarité entre vecteurs documents permet de retrouver les éléments voisins dans des champs d'un sujet donné. Chaque document est représenté par un vecteur de poids de termes, pris dans l'intervalle [0,1], 0 signifiant que le terme n'est pas pertinent pour décrire le document, 1 signifiant qu'il l'est totalement. La mesure de cosinus permet de calculer la similarité des documents et de construire les *clusters* de documents. Il existe des recouvrements entre les clusters, puisque les documents peuvent appartenir à plusieurs clusters. Pour améliorer la manipulation des collections clusterisées, un centroïde est calculé pour chaque cluster : il représente la moyenne des vecteurs documents de la classe : pour chaque terme, le centroïde a un poids constitué de la moyenne des poids de ce terme dans la classe de documents.

Pour une recherche d'information, les documents sont classés par ordre de similarité décroissante vis-à-vis de la requête. La mesure de similarité utilisée est le cosinus. Des procédures automatiques permettent d'améliorer la recherche à partir de l'information obtenue dans des opérations de recherche antérieures : le bouclage de pertinence, introduit dans SMART au début des années 70, permet de reformuler la requête à partir des jugements de l'utilisateur sur la pertinence des documents que le système lui a présentés. La mise en correspondance s'effectue en deux étapes : chaque requête est d'abord comparée aux divers vecteurs centroïdes, puis aux documents des classes dont les centroïdes sont les plus proches.

Construit sur le modèle vectoriel, SMART a servi de banc d'essai pour de nombreuses méthodes expérimentales et en particulier des méthodes de recherche statistiques. Il a servi de banc d'essai pour le modèle booléen étendu en 1982 [Fox 87], de façon à permettre d'utiliser différents types de concepts pour représenter les différents aspects des documents composites : le système traitait alors plusieurs sous-vecteurs composants à la place d'un seul (voir § 2.1.2 page 29). Mais SMART ne propose pas de façon simple de représenter les relations entre les composants de documents. En construisant un vecteur différent pour chaque composant, les relations structurelles peuvent être conservées, mais il est difficile de relier les différents vecteurs en un schéma vectoriel de façon à ce qu'un objet qui est une partie d'un autre objet hérite d'une partie des descriptions de l'objet composite.

Le **système SIRE** (Syracuse Information Retrieval Experiment) [Salton 83a] utilise des fichiers inverses pour le traitement de requêtes booléennes et la pondération des termes pour permettre le classement des documents résultats. Le processus de recherche se fait en deux étapes : une requête booléenne est tout d'abord traitée de manière conventionnelle. Cette étape permet d'identifier tous les documents qui répondent de manière précise à la requête. Ce sous-ensemble de documents est ensuite traité en calculant une mesure de cosinus entre chaque document et la requête «mise à plat» c'est-à-dire où tous les termes de la requête d'origine sont connectés par des OU, ce qui permet d'établir un classement entre les documents retrouvés. Les termes sont pondérés par leur fréquence d'occurrence dans les documents.

Ces deux systèmes, qui s'appliquent à des documents et requêtes textuels, sont de bons exemples de la collaboration minimale qui peut être nécessaire entre différents paradigmes pour la recherche d'information.

4.1.2 Recherche interactive d'images : EXPRIM et RIVAGE

Le processus EXPRIM (EXPert pour la Recherche d'IMages) (Figure 4.1) [Créhange 85], implanté dans le prototype RIVAGE [Halin 89] de recherche interactive d'images, utilisait à l'origine quatre composants :

- un système expert qui fonctionne en chaînage avant pour trouver des documents à partir de l'hypothèse sur le besoin de l'utilisateur que représente la requête, puis en chaînage arrière pour modifier l'hypothèse initiale c'est-à-dire reformuler la requête à partir des jugements de pertinence de l'utilisateur sur les documents que le système a retrouvés ;

- un système documentaire qui gère les descriptions d'images (sous forme de mots-clés) et peut être implanté sous forme d'une base de données ;

- un système de visualisation qui permet l'accès à la base d'images, la phase de visualisation des documents étant très importante dans ce processus ;

- un module de dialogue pour piloter l'ensemble du système et activer l'un ou l'autre des composants en fonction des choix de l'utilisateur. En particulier, l'utilisateur peut naviguer dans le thésaurus pour choisir les termes de sa requête, ou entrer directement les termes voulus. Lorsqu'il visualise les documents retrouvés par le système, il peut émettre des choix ou des rejets qui sont ensuite exploités pour reformuler automatiquement la requête.

Le processus EXPRIM repose sur trois phases principales pour chaque étape de la recherche : la phase *avant-visualisation* qui, à partir d'une requête, constitue un ensemble de documents susceptibles d'être pertinents, la phase de *visualisation* durant laquelle les documents sont présentés à l'utilisateur, celui-ci pouvant se prononcer sur leur pertinence vis-à-vis de son besoin d'information, et la phase *après-visualisation* où les jugements de l'utilisateur sont exploités pour reformuler automatiquement la requête. Le prototype RIVAGE [Halin 89] a d'abord été implanté selon le modèle booléen de mise en correspondance. Le modèle de mise en correspondance vectoriel a ensuite été adopté [Halin 90], mais un pré-filtrage booléen tolérant a été conservé pour améliorer les performances du système : dans une première phase, tous les termes de la requête sont connectés par des *OU* et une mise en correspondance booléenne permet de constituer un sous-ensemble de documents parmi lesquels s'effectue ensuite la mise en correspondance vectorielle. Dans RIVAGE, le thésaurus est le support des inférences réalisées par le système dans la phase de reformulation de la requête : les poids d'importance des termes pour la requête sont en effet directement attachés aux nœuds du thésaurus, et propagés dans le thésaurus (voir § 2.3.1 page 58).

Le processus EXPRIM a été généralisé, scindé en plusieurs primitives et paramétré de façon à pouvoir être instancié en une stratégie particulière en fonction du type de besoin d'information de l'utilisateur. Nous allons présenter en détail le modèle paramétré de recherche d'information avant d'y proposer quelques ajouts, pour tenir compte de la structuration des documents, des représentations textuelles plus riches de ces documents et de la distribution du système.

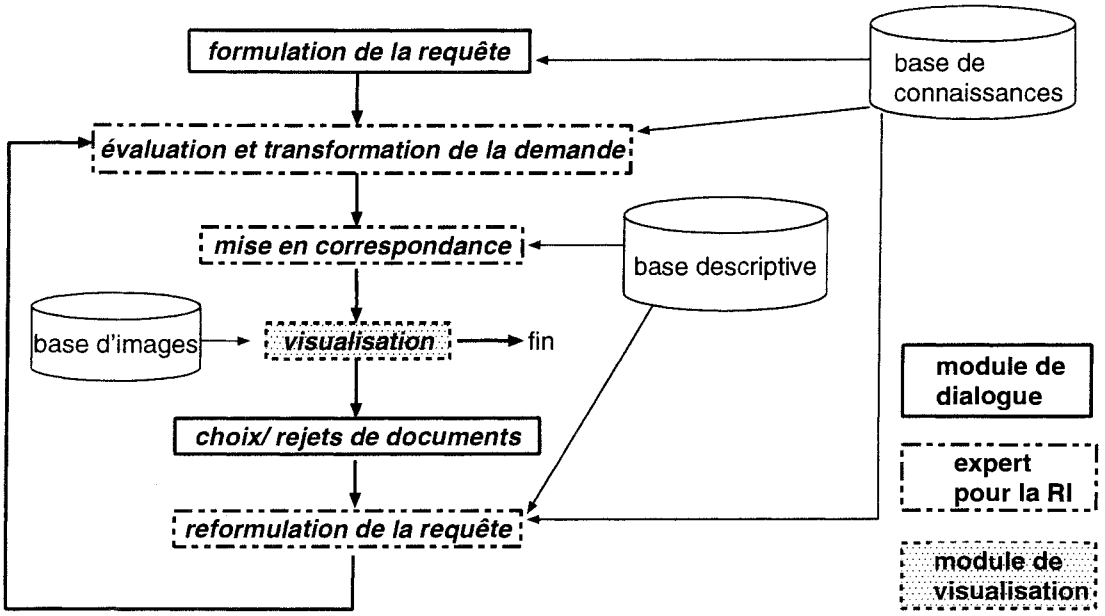


FIG. 4.1 – *Le processus EXPRIM.*

4.2 Le modèle paramétré de processus de recherche d'information

Le modèle paramétré proposé dans [Smaïl 94, Simonnot 95b] met en évidence un certain nombre de paramètres stratégiques qui interviennent dans un processus de RI (Figure 4.2). Certains de ces paramètres avaient été implantés dans le prototype RIVAGE [Halin 89], sans être totalement explicités. Le modèle paramétré les analyse et les complète. Il prend en compte également dans une certaine mesure la notion de facettes des descriptions pour les documents et la requête.

Présentation du modèle paramétré

Le modèle paramétré distingue six primitives d'un processus de recherche interactif d'information : formulation de la requête, interprétation, déformation, mise en correspondance, visualisation et choix, reformulation.

La **formulation de la requête** recouvre la phase durant laquelle l'utilisateur exprime sa demande d'information. Cette primitive n'a pas de paramètre stratégique, sa réalisation dépendant surtout de l'interface qui permet l'interrogation du système.

L'**interprétation** consiste à présélectionner un sous-ensemble de documents dans la base. Cette présélection peut s'appuyer sur une interprétation booléenne de la requête. En présence de facettes, trois *types d'interprétation* sont possibles, en fonction des opérateurs booléens introduits entre les différents composants : une interprétation stricte connecte

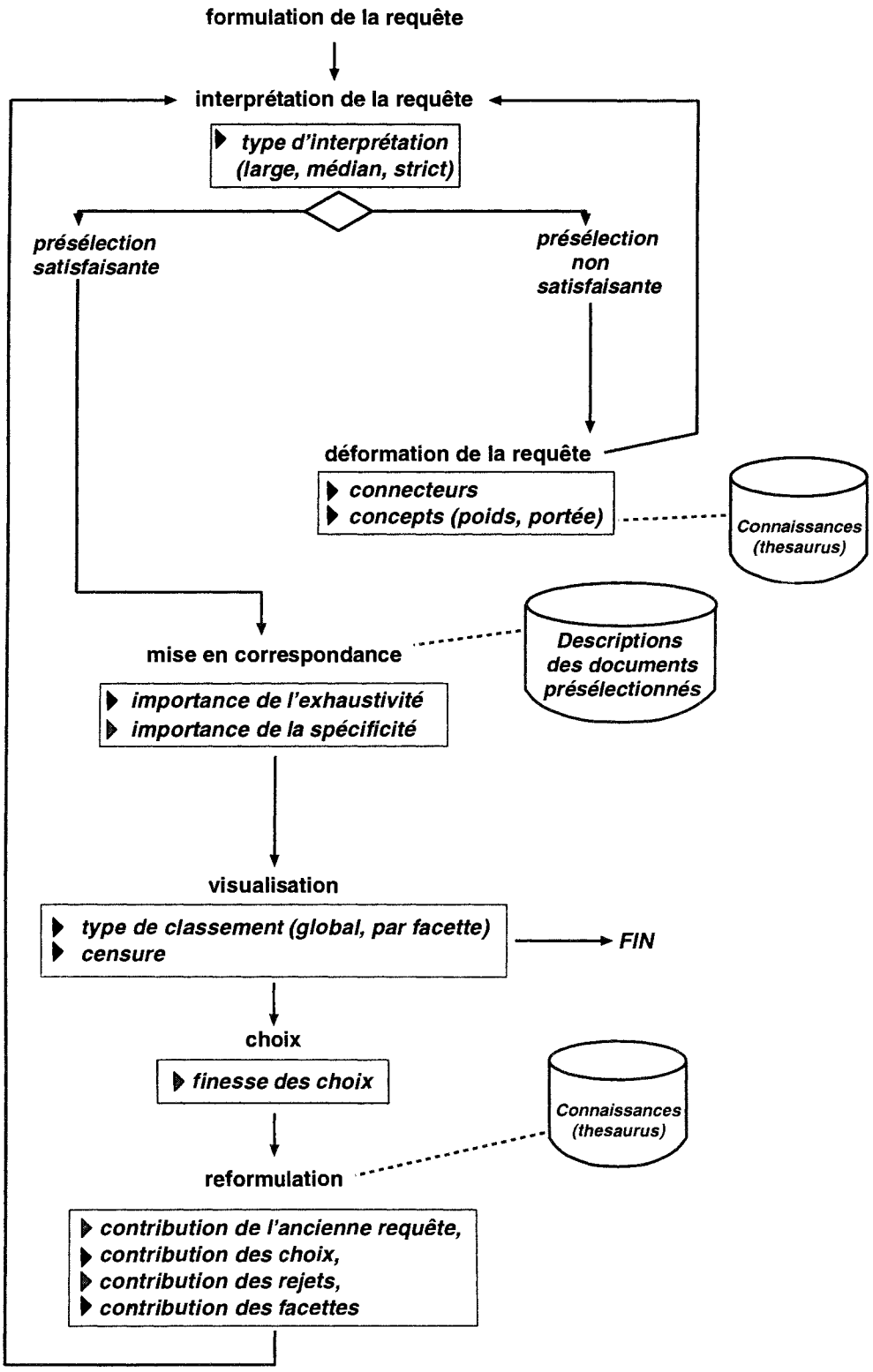


FIG. 4.2 – *Le modèle paramétré de RI interactive.*

tous les termes de la requête par des ET, une interprétation large les connecte par des OU, une interprétation médiane opère la disjonction des concepts d'une même facette et la conjonction des facettes.

La **déformation** de la requête permet d'anticiper un échec si l'ensemble de documents présélectionnés n'est pas satisfaisant. Si elle rapporte trop de documents, la requête doit être restreinte alors qu'elle doit être élargie si le nombre de documents obtenus est nul ou trop faible. Dans ces deux cas, il convient de choisir de modifier soit les *connecteurs logiques* introduits dans la requête soit les *concepts* ; le concept à déformer en priorité peut être celui qui a le poids le plus faible dans la requête ou celui qui rapporte trop (ou trop peu) de documents.

La **mise en correspondance** se fait grâce à une mesure de similarité qui combine mesures d'exhaustivité et de spécificité. Le modèle paramétré propose ainsi une interprétation vectorielle du modèle logique. L'exhaustivité est mesurée par projection orthogonale du vecteur document (comportant m termes pondérés) sur le vecteur requête (comportant n termes pondérés) :

$$exh(D,Q) = \frac{D.Q}{\|Q\|} = \frac{\sum_{j=1}^{n} d_{ij}.q_j}{\sum_{j=1}^{n} q_j^2}$$

et la spécificité par projection orthogonale du vecteur requête sur le document :

$$spe(D,Q) = \frac{D.Q}{\|D\|} = \frac{\sum_{j=1}^{n} d_{ij}.q_j}{\sum_{i=1}^{m} d_i^2}$$

Les paramètres attachés à la primitive de mise en correspondance sont donc les importances respectives des mesures d'exhaustivité et de spécificité, ces paramètres prenant soit la valeur 0, soit la valeur 1. Lorsque ces deux paramètres prennent la valeur 1, une fonction compromis permet de combiner ces deux mesures; elle s'inspire du coefficient de Jaccard :

$$\begin{aligned} sim(D,Q) &= \frac{exh(D,Q).spe(D,Q)}{exh(D,Q) + spe(D,Q) - exh(D,Q).spe(D,Q)} \\ &= \frac{\sum_j d_j.q_j}{\sum_i d_i^2 + \sum_j q_j^2 - \sum_j d_j.q_j} \end{aligned}$$

Cette mesure de similarité, qui donne autant d'importance aux critères de spécificité et d'exhaustivité, est nulle lorsque document et requête n'ont aucun descripteur commun. Elle est maximum (vaut 1) si

$$\sum_j d_j.q_j = \frac{\sum_i d_i^2 + \sum_j q_j^2}{2}$$

c'est-à-dire si document et requête ont le même nombre de descripteurs avec des poids similaires. Cette mesure croît avec le nombre de descripteurs communs entre requête et document, et est d'autant plus forte que les descripteurs communs ont les mêmes poids

dans les deux vecteurs. Enfin, par rapport au coefficient de Jaccard d'origine (voir § 2.1 31 et § 2.6 53), elle permet de traiter correctement les poids négatifs introduits dans la formulation de la requête pour exprimer ce que l'utilisateur veut éviter de retrouver.

Pour la phase de **visualisation des documents** par l'utilisateur, il faut déterminer si le *classement* sera global ou par facette d'une part, et si une *censure* sera opérée, empêchant l'utilisateur de voir les documents les moins bien classés. Lors de cette phase, le chercheur d'information peut opérer des **choix** dont la *finesse* constitue un autre paramètre. Par exemple, une fonction de choix à deux valeurs offre simplement la possibilité de choisir ou de rejeter un document.

Lors de la **reformulation**, il est possible de moduler :

- la *contribution des facettes* c'est-à-dire le poids d'importance de chaque facette dans la reformulation de la requête. Un poids nul indique qu'une facette ne sera pas prise en compte lors de la reformulation.
- la *contribution de l'ancienne requête* ou degré auquel la requête précédente sera prise en compte lors de la nouvelle étape. C'est le paramètre α dans la formule (4.1).
- la *contribution des choix* permet de moduler l'importance des termes présents dans les documents choisis (paramètre β dans la formule 4.1).
- la *contribution des rejets* permet de prendre plus ou moins en compte les documents jugés négativement ou rejetés par l'utilisateur (paramètre γ dans la formule 4.1).

Ces trois derniers critères sont impliqués dans la fonction générale de reformulation qui recalcule chaque composante $R'_{\mathcal{F}}$ de la requête scindée par facettes :

$$\begin{aligned} R'_{\mathcal{F}} &= \alpha R_{\mathcal{F}} \\ &+\beta(\frac{1}{|D_c|}\sum_{d_i \in D_c} max(fchoix(d_i, \mathcal{F}), fchoix(d_i, nil)) * d_i) \\ &+\gamma(\frac{1}{|D_r|}\sum_{d_j \in D_r} max(fchoix(d_j, \mathcal{F}), fchoix(d_j, nil)) * d_j) \end{aligned} \tag{4.1}$$

où

$R_{\mathcal{F}}$ représente le vecteur correspondant à la sous-requête pour la facette $\mathcal{F}$ à l'étape précédente,

D_c est l'ensemble des documents proposés par R et qui ont donné lieu globalement ou partiellement à un choix positif, mais sans être rejetés pour $\mathcal{F}$

D_r est l'ensemble des documents ayant donné lieu globalement ou partiellement à un choix négatif, mais sans être choisis pour $\mathcal{F}$

$\alpha, \beta, \gamma \in [0, 1]$

$|E|$ représente le cardinal de l'ensemble E

$\sum$ est l'addition de vecteurs.

Le modèle paramétré est une première étape vers l'analyse et l'explicitation des dépendances entre les principales phases d'une recherche d'information. Nous allons revisiter ce modèle et le compléter sur un certain nombre de points.

4.3 Le modèle paramétré revisité

Le modèle paramétré n'a pas été élaboré dans un contexte où les différentes unités du système sont distribuées physiquement. D'autre part, la distribution des connaissances par facette n'est prise en compte que de façon très partielle par ce modèle. Nous allons présenter les modifications et les adjonctions qu'il est nécessaire d'apporter à ce modèle dans un contexte distribué, et pour pouvoir exploiter un formalisme de représentation plus riche que celui du modèle vectoriel. Ceci nous amène à décomposer encore les primitives du modèle initial. Par ailleurs, nous analysons les différentes phases de ce modèle par rapport au formalisme plus riche que nous utilisons pour représenter les informations : le modèle de graphes conceptuels.

4.3.1 Impact du traitement par facette sur les étapes de la résolution

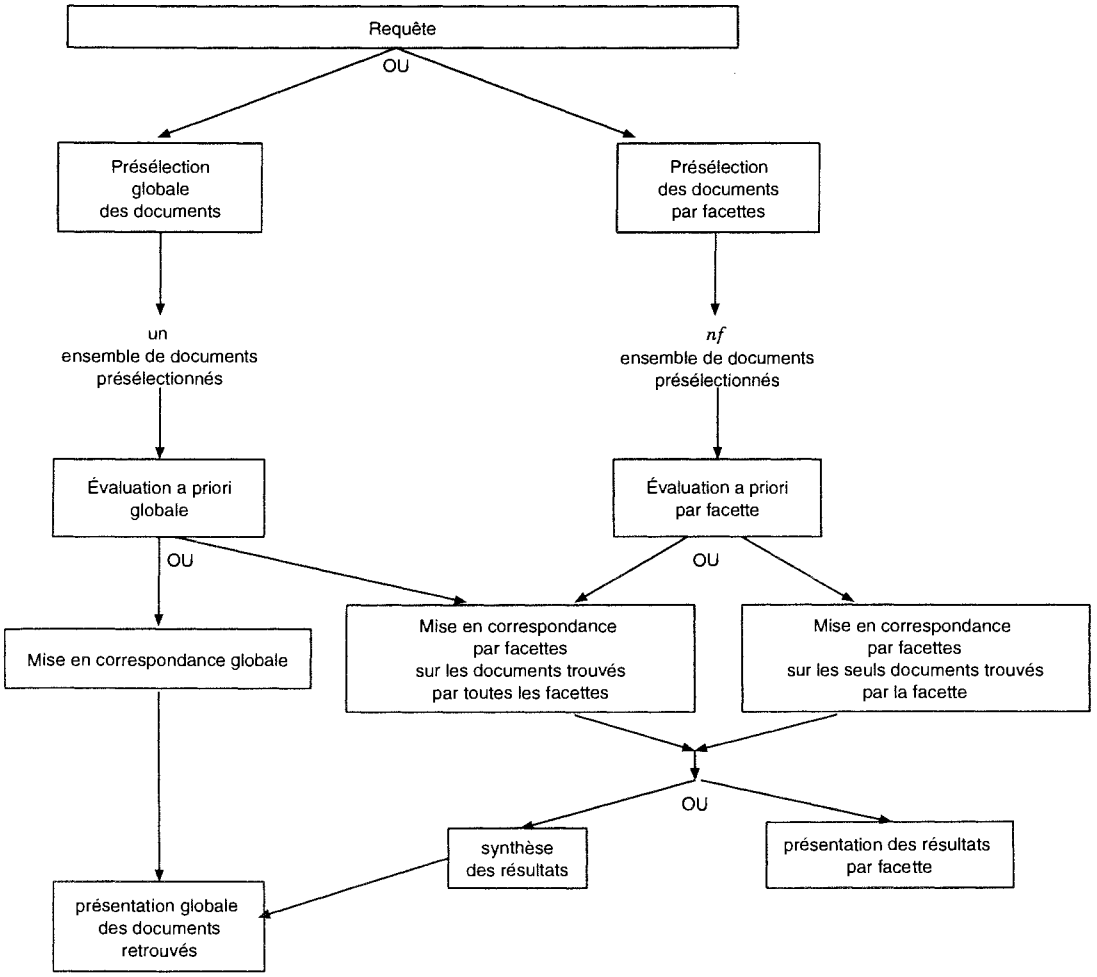


FIG. 4.3 – *Choix de résolution possibles.*

La figure 4.3 montre les alternatives possibles pour chaque phase de la résolution de problème selon que la résolution est distribuée ou non entre les différentes facettes, et qu'elles

coopèrent ou non.

4.3.2 Formulation de la requête

Cette phase du processus n'a pas de paramètre associé dans le modèle paramétré original, car elle est considérée comme dépendante de l'interface. Ceci est en partie vrai, mais nous paraît supposer que tout SRI a une interface dédiée. Or, dans le contexte distribué actuel, ce n'est pas forcément le cas : il peut être souhaitable d'avoir une interface générique capable de permettre la saisie de requêtes qui seront ensuite envoyées à différents SRI.

Dans le cas d'un système distribué, la formulation de la requête se fait sur une **interface distante**. Si l'application de recherche d'information est intégrée à un kiosque [11] permettant l'accès à plusieurs applications, et dans le cas où ces applications sont hétérogènes, l'interface de formulation n'est pas forcément représentative du modèle de requête que chaque application attend. Nous avons étudié ce problème dans le cadre d'un contrat d'étude [12]. Il est possible qu'un «kiosque» propose une interface unique à plusieurs applications de RI : à partir des éléments d'information dont il dispose sur chaque application, il permet à un utilisateur de formuler sa requête selon un des modèles de requête des applications. Cette solution est simple pour les SRI, mais fait peser sur l'utilisateur la charge de formuler sa requête dans chacun des formalismes attendus ; l'interface transmet ensuite à chaque application la requête qui la concerne.

En présence d'un SRI modélisant ses connaissances sous forme de graphes conceptuels, il nous semble difficile de demander à des utilisateurs de formuler directement leur requête sous la forme requise. Une autre solution consiste à permettre à l'utilisateur d'entrer sa requête sous la forme libre d'une liste de mots-clés pondérés, par saisie directe ou grâce à un langage de requête visuel [Bélières 95]. C'est cette seconde alternative que nous choisissons : la requête peut être émise selon le modèle de requête vectoriel : l'utilisateur entre des mots-clés qu'il peut pondérer pour indiquer leurs importances relatives. La pondération de l'importance des termes s'effectue par l'intermédiaire d'une fonction continue qui permet de sélectionner des poids entre -1 et +1. Un poids de -1 indique que le concept correspondant doit être absolument évité, un poids de +1 qu'il doit être absolument sélectionné. Un poids intermédiaire indique le degré auquel le terme est important pour la requête. Ainsi, la requête :

(catastrophe naturelle 1), (inondations -1), (San Francisco 0,5)

indique que l'utilisateur cherche des documents sur les catastrophes naturelles, et si possible celles qui ont eu lieu à San Francisco, mais pas sur les inondations. Si l'interface ne permet pas d'entrer des poids, tous les poids seront automatiquement mis à 1.

Si l'application dispose d'un thésaurus, cette source de connaissance doit pouvoir être exploitée durant la phase de formulation pour donner à l'utilisateur un aperçu des connaissances

11. Un kiosque regroupe plusieurs applications ou services qu'un utilisateur peut lancer grâce à une interface initiale unique.

12. Contrat 93 ME 13 avec le CCETT et Euroclid.

dont dispose le SRI. Plutôt que d'entrer des termes libres, le chercheur d'information doit pouvoir naviguer dans les champs sémantiques du thésaurus. L'interface doit donc avoir la possibilité de demander à une application de RI de lui communiquer un thésaurus.

Il est légitime de se demander si l'utilisateur doit pouvoir spécifier dans sa requête des concepts correspondant à des noms d'attributs. En effet, ces noms d'attributs sont souvent implicites dans des requêtes formulées en langage naturel. Par exemple, dans la demande d'«*œuvres du photographe Nadar montrant les quais de la Seine à Paris*», le groupe de mots *œuvres du photographe Nadar* peut être traduit en une requête `AUTEUR = Nadar` et `MEDIA = photographie`, où `AUTEUR` et `MEDIA` jouent le rôle d'attributs. Sans vouloir imposer à l'utilisateur un langage trop formel, nous supposons que l'interface peut demander à une application les facettes d'interrogation dont elle dispose, chaque facette regroupant plusieurs attributs. Par exemple, la facette SIGNALÉTIQUE regroupe les attributs AUTEUR, DATE et LIEU DE PRODUCTION. Ce type de consultation est facilité par le formalisme de représentation des graphes conceptuels puisque les attributs peuvent être représentés de manière homogène par des concepts relationnels dépendant d'un super-type, la facette.

Le modèle de requête que nous retenons est donc un ensemble de quadruplets : **{(attribut, terme, domaine, poids)}**, étant donné que seule la présence du terme est obligatoire. Si l'utilisateur ne précise pas d'attribut, ce peut être parce qu'il souhaite retrouver les documents où le concept indiqué joue un rôle quelconque : par exemple, tout document à propos de Eugène Atget[13] signifie les photographies montrant Eugène Atget et/ou celles prises par lui. Le SRI peut en effet déterminer le domaine et l'attribut du terme (éventuellement en ayant recours à l'utilisateur si une ambiguïté doit être levée). Les poids doivent appartenir à l'intervalle $[-1, 1]$ (la valeur par défaut est 1). Des valeurs discrètes correspondant à cet intervalle continu sont :

poids	*interprétation*
1	les documents indexés par ce terme doivent être sélectionnés absolument
0,5	les documents indexés par ce terme doivent si possible être sélectionnés
0	terme neutre pour l'expression du besoin
-0,5	les documents indexés par ce terme doivent si possible être évités
- 1	les documents indexés par ce terme doivent être absolument évités

4.3.3 Interprétation de la requête

Des études expérimentales ont montré que des chercheurs d'information différents utilisent une variété de termes pour exprimer un même besoin d'information [Iivonen 95]. Une des études les plus récentes indique qu'il existe seulement 31,2 % de cohérence entre les termes utilisés par des chercheurs différents pour exprimer un même besoin d'information, la cohérence étant bien plus forte en ce qui concerne les concepts de recherche (87,6 %). Un **concept de recherche**, pour Iivonen, regroupe des termes correspondant aux variations grammaticales d'un même terme (par exemple singulier/pluriel), ou des termes liés par une relation d'association "communément reconnue", c'est-à-dire celles enregistrées dans un thésaurus (cf. § 1.3.2). Le

13. Eugène Atget (1857-1927) est célèbre pour les photographies qu'il a prises du Paris du début du siècle.

vocabulaire utilisé dans la requête correspond rarement au vocabulaire d'indexation, ce qui constitue une cause importante d'échec d'une session de recherche. Cette constatation est également faite dans [Croft 90a], d'où l'intérêt d'une phase d'interprétation pendant laquelle le système tente de reconnaître les concepts exprimés dans la requête.

Si l'utilisateur a entré des termes libres dans sa requête, le SRI doit d'abord **identifier les champs sémantiques** de ces termes. Ceci consiste à retrouver dans le thésaurus les concepts de la requête, afin de déterminer quels sont leurs domaines. Le système peut essayer d'affecter à chaque terme un domaine en prenant l'initiative d'un dialogue avec l'utilisateur pour qu'il spécifie de façon plus précise sa requête. Par exemple, si l'utilisateur entre le terme **canal**, l'interface lui proposera la boîte de dialogue suivante :

Choisissez entre ces différents domaines pour le terme **canal** :
→ irrigation
→ communication

La **résolution des synonymes** est un des sous-buts de cette phase du processus : si l'application connaît un dictionnaire de synonymes, elle peut traduire les termes de l'utilisateur en leurs descripteurs correspondants, utilisés pour l'indexation des documents. En général, les synonymes font partie du thésaurus lié à une application de RI.

Certains termes de la requête peuvent ne pas faire partie du vocabulaire d'indexation. Ce sera souvent le cas par exemple si le vocabulaire d'indexation est limité aux termes les plus spécifiques du thésaurus, c'est-à-dire les feuilles si le thésaurus est vu comme une forêt d'arbres pour la relation générique/spécifique. Pour les systèmes de recherche d'information qui s'adressent à des collections d'images, l'hypothèse d'indexation aux feuilles du thésaurus est souvent utilisée : les mots ne sont pas extraits directement du document, et vu qu'il est possible de déduire logiquement la correspondance entre un terme générique et ses spécifiques [Chiaramella 92], le choix des termes les plus spécifiques possible est préférable. L'utilisateur peut malgré tout utiliser un terme assez générique dans sa requête. Un des rôles de l'interprétation de la requête sera de **projeter aux feuilles du thésaurus chaque terme** utilisé pour exprimer la requête.

Les différentes ambiguïtés qui doivent être résolues à ce stade sont, pour résumer :

- ambiguïté du niveau du terme par rapport à l'application : le mot-clé entré est-il une valeur de donnée (terme d'indexation) ou le nom d'une méta-donnée (nom d'attribut ou nom de facette) ?

- homonymie : un même mot-clé peut appartenir à plusieurs domaines avec des sens différents. Exemple : `voile` peut avoir pour domaine `habillement` ou `matériel de navigation`.

- ambiguïté liée au rôle d'un terme dans la requête. C'est le cas lorsque des attributs différents ont des domaines de valeurs qui se recouvrent. Par exemple, les domaines des attributs AUTEUR, ACTEURS ou PERSONNES REPRÉSENTÉES peuvent ne pas être disjoints. Cette ambiguïté est levée si la formulation se fait par *facette* et par attribut, ou bien si tous les attributs ont des domaines disjoints.

- ambiguïté liée aux relations qu'il est possible d'insérer entre les termes de la requête : un terme affectée à l'attribut PERSONNE de la facette CONTENU peut avoir une relation d'agent, d'objet, de caractérisation, etc. avec un autre terme. Si l'ambiguïté est gênante, elle devra être levée, mais nous anticipons ici sur la phase de proaction.

Il nous semble souhaitable de limiter la primitive d'interprétation à l'aspect sémantique, sans tenir compte du modèle de description utilisé par le système. Il s'agit d'une première mise en correspondance des différents vocabulaires utilisés respectivement par l'utilisateur et l'application de RI.

4.3.4 Formalisation de la requête

Puisque nous émettons l'hypothèse que l'interface d'un SRI ne lui est pas forcément dédiée, une étape de **transformation de la requête utilisateur en requête formelle** est nécessaire. Des requêtes formelles différentes peuvent être utilisées pour la mise en correspondance proprement dite (forme vectorielle ou graphes conceptuels) et la présélection de documents dans la collection (forme booléenne).

Formalisation booléenne

Si le SRI utilise un modèle booléen de pré-filtrage de la collection, le système doit rechercher les opérateurs dans la requête ou les y introduire. Le modèle booléen permet en effet de partager la collection de documents en deux classes : documents susceptibles/non susceptibles d'être pertinents. En faisant en sorte que ce classement ne soit pas trop strict, l'interprétation booléenne peut opérer un premier filtrage grossier des documents. Dans un premier temps, les poids affectés aux termes de la requête ne sont pas pris en compte sauf pour leur signe : seuls les concepts pondérés positivement sont traités à cette phase. Si une restriction est nécessaire, il est possible d'exclure les documents dont des descripteurs sont pondérés négativement dans la requête.

La présélection des documents peut être globale ou par facette. Dans ce dernier cas, soit chaque facette présélectionne ses propres documents et travaille de façon indépendante sur le sous-ensemble présélectionné, soit les facettes réunissent les documents qu'elles ont présélectionnés pour augmenter leurs chances de convergence.

Si la présélection est large, les facettes travaillent sur les documents présélectionnés par au moins une facette. Si la présélection est intermédiaire, les facettes travaillent sur les documents qu'elles ont présélectionnés et qui ont été aussi présélectionnés par une ou plusieurs autres facettes. Si la présélection est stricte, les facettes travaillent sur les documents qui ont été présélectionnés par toutes les facettes.

Nous allons étudier l'impact d'une présélection par facette sur un exemple. La requête constituée des concepts {`portrait`, `femme`, `chapeau`, `gai`, `gros plan`} est répartie ainsi

entre les facettes :

r_{genre} = portrait
$r_{contenu}$ = femme, chapeau
$r_{connotation}$ = gai
$r_{morphologie}$ = gros plan

La collection contient les documents suivants (nous omettons les descripteurs non présents dans la requête pour alléger la présentation) :

D_1 = CADRAGE : gros plan
D_2 = CADRAGE : gros plan, CONTENU : femme
D_3 = CONTENU : femme, chapeau
D_4 = CONTENU : femme, CONNOTATION : gai, GENRE : portrait
D_5 = CADRAGE : gros plan, CONTENU : chapeau
D_6 = CONNOTATION : gai
D_7 = GENRE : portrait

Suivant le type de présélection opérée, nous obtenons les ensembles de documents :

Type de présélection	Ensemble de documents présélectionnés
présélection large	$\{D_1, D_2, D_3, D_4, D_5, D_6, D_7\}$
présélection intermédiaire	
- présélection à 2 facettes	$\{D_2, D_4\}$
- présélection à 3 facettes	$\{D_4\}$
- présélection à 4 facettes	$\varnothing$

Nous pouvons constater que le fait d'opérer l'intersection des ensembles présélectionnés par deux ou plusieurs facettes modifie de façon considérable les résultats. Cette présélection est assez «injuste», car elle ne tient pas compte du nombre de concepts pertinents dans chaque facette. Ainsi, D_3 comporte autant de concepts de la requête que D_2, mais ses concepts n'étant pas répartis entre plusieurs facettes, il n'est pas retenu par une présélection intermédiaire. La restriction de l'interprétation par conjonction des présélections des facettes ne peut pas être retenue à notre point de vue, car elle est arbitraire. C'est pourquoi nous préférons adopter l'approche de A. Motro qui n'utilise que la connaissance sur les domaines des termes.

Des chercheurs, comme A. Motro, ont travaillé à la construction automatique d'une requête booléenne à partir d'une liste de mots-clés [Motro 86], dans le contexte de l'interrogation de bases de données relationnelles. En effet, la construction de requêtes booléennes par un utilisateur n'est pas une tâche triviale, et une erreur dans les opérateurs logiques utilisés pour connecter les différents termes de la requête peut avoir des conséquences importantes sur les performances de la recherche. Une interprétation booléenne "standard" d'une requête émise sous forme de liste de mots-clés peut être faite selon les règles heuristiques suivantes :

1. les termes représentant un même concept, c'est-à-dire associés par une relation dans le thésaurus, sont reliés par un *OU* (ceci comprend l'appartenance à un même domaine, les relations générique/spécifique et les relations d'association entre termes)

2. les termes correspondant à des concepts très différents (sans relation dans le thésaurus) sont connectés par un ET.

L'étude expérimentale que nous avons relatée page 106 [Iivonen 95] a permis de constater empiriquement que l'approche proposée par A. Motro était mise en œuvre très souvent dans la réalité par les chercheurs d'information lorsqu'ils formulent une requête booléenne. C'est pourquoi nous considérons que, pour une formulation booléenne standard de la requête, les termes pondérés positivement et qui appartiennent au domaine d'un même attribut seront connectés par des OU, alors que les groupes de termes appartenant à des domaines sémantiques différents seront liés par des ET.

Ainsi, la requête : «*actualités télévisées montrant une catastrophe naturelle ou un accident à San Francisco, mais pas une inondation*» sera interprétée par défaut sous la forme booléenne : `actualités télévisées ET (catastrophe naturelle OU accident) ET San Francisco`.

Ce type d'interprétation est l'interprétation standard de notre modèle. Une interprétation large connectera tous les termes par des OU, une interprétation stricte les connectera tous par des ET.

Par rapport aux types d'interprétation du modèle paramétré, nous ne considérons donc pas les facettes à ce stade, mais seulement les relations sémantiques entre les termes enregistrées dans le thésaurus. L'intervention des facettes fausse les résultats de la présélection à ce stade, et nécessite un traitement supplémentaire de recherche de facettes qui peut être coûteux en temps.

Nous conservons donc les trois types d'interprétation du modèle paramétré, mais leur donnons une signification différente

- formulation stricte : tous les termes de la requête sont connectés par des ET,
- formulation médiane : les concepts d'un même domaine sémantique sont connectés par des OU, ces expressions disjonctives étant connectées par des ET,
- formulation large : tous les concepts sont connectés par des OU.

Par contre, la présélection des documents par facette peut avoir un avantage important sur les temps de traitement : chaque facette n'aura à considérer que les documents susceptibles d'être pertinents pour elle. Un document n'ayant aucun descripteur de la requête initiale dans une facette ne sera pas examiné par cette facette. Cette amélioration des temps de traitement n'est valable que si l'on s'en tient à une compatibilité stricte des termes, c'est-à-dire si l'on ne considère que la subsomption entre les termes. Si la co-taxonomie est prise en compte pour le calcul de la proximité sémantique, il pourrait être intéressant de faire examiner tous les documents par toutes les facettes pour trouver des termes proches sémantiquement de ceux de la requête. Nous reviendrons sur ce point au § 4.3.7.0.

Représentation vectorielle de la requête

La formalisation vectorielle de la requête ne pose aucun problème particulier. Un vecteur de poids est composé par facette de description, ou globalement pour toutes les facettes.

Représentation par des graphes conceptuels

Si les documents sont décrits grâce au formalisme des graphes conceptuels, le SRI doit essayer d'établir quelles sont les relations impliquées entre les concepts de la requête. Pour cela, il faut identifier le type des termes utilisés dans la requête, et rechercher dans la base canonique les graphes comprenant des concepts relationnels d'un type compatible. Plusieurs graphes conceptuels pourront donc être générés pour un même terme (un concept non relationnel). Prenons pour exemple la requête simple : «*femme, robe*» (nous faisons abstraction des poids pour l'instant). Le modèle de description donne pour chaque attribut les domaines sémantiques dans lesquels l'attribut peut prendre sa valeur. Dans le cas que nous prenons en exemple, il indique que le terme `femme` est une valeur possible pour l'attribut PERSONNE, et `robe` est une valeur possible pour l'attribut OBJET. Les graphes de la base canonique où ces attributs sont impliqués sont les suivants :

PERSONNE	→	AGNT	→	ACTION
OBJET	→	AGNT	→	ACTION
ACTION	→	OBJT	→	OBJET
ACTION	→	OBJT	→	PERSONNE
ACTION	→	INST	→	OBJET
PERSONNE	→	CHRC	→	CARACTÈRE
OBJET	→	CHRC	→	CARACTÈRE

Une formulation large de la requête consiste à spécialiser les types de concepts de la base canonique en leur affectant pour référent le terme de type compatible de la requête. Ceci donnerait pour notre exemple l'ensemble de graphes suivants :

PERSONNE : femme	→	AGNT	→	ACTION
OBJET : robe	→	AGNT	→	ACTION
ACTION	→	OBJT	→	PERSONNE : femme
ACTION	→	OBJT	→	OBJET : robe
ACTION	→	INST	→	OBJET : robe
PERSONNE : femme	→	CHRC	→	CARACTÈRE
OBJET : robe	→	CHRC	→	CARACTÈRE

Une formalisation stricte de la requête pourrait chercher à faire intervenir le plus grand nombre de concepts de la requête dans la même relation, en recherchant la jointure maximale des graphes instanciés sur leurs concepts communs, les concepts ne pouvant intervenir qu'une

seule fois dans la relation. Par exemple, une interprétation stricte de la requête «*femme, robe* » serait :

$$\boxed{\text{PERSONNE : femme}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{ACTION}} \rightarrow (\text{OBJT}) \rightarrow \boxed{\text{OBJET : robe}}$$
$$\boxed{\text{OBJET : robe}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{ACTION}} \rightarrow (\text{OBJT}) \rightarrow \boxed{\text{PERSONNE : femme}}$$

Toutefois, l'opération de recherche de jointure maximale est coûteuse, et n'est pas nécessaire à ce stade.

Pour tenir compte des poids affectés par l'utilisateur aux termes de la requête, il est possible de calculer pour chaque relation primitive obtenue par instanciation des graphes de la base canonique un poids correspondant au poids moyen des termes qui interviennent dans la relation. L'instanciation d'un graphe recouvre pour nous le fait d'affecter à un type de concept un référent du type. Un type de concept non instancié dans la requête a un poids neutre (égal à 0).

Les poids négatifs expriment l'exclusion des documents indexés par ce terme, et sont traités comme les poids positifs à ce stade.

Intérêt de la possibilité de formalisation à la demande de la requête

Plus généralement, cette primitive peut permettre de prendre en compte

- les langages d'interrogation visuels en opérant une traduction,
- l'interrogation à distance de serveurs d'information distribués hétérogènes, chaque serveur formalisant à sa façon la requête reçue.

Nous limitons cette primitive à la constitution d'une requête conforme au modèle du système.

L'interprétation et la formalisation de la requête étant réalisées, le système peut *pro-agir*, c'est-à-dire évaluer grossièrement les résultats de la requête pour anticiper un échec de la recherche.

4.3.5 Proaction : Évaluation a priori

Le critère d'évaluation qui permet de décider d'une déformation de la requête est pragmatique : il s'agit de la cardinalité d'un sous-ensemble des documents pré-sélectionnés. Cet ensemble peut être obtenu grâce à une interprétation ensembliste d'une requête booléenne faisant abstraction des poids attribués à chacun des termes.

La phase de pré-sélection de documents peut alors utiliser la correspondance qui existe entre les expressions booléennes et les opérations ensemblistes pour sélectionner un sous-ensemble de documents susceptibles d'être pertinents. La pré-sélection peut être implantée sous forme efficace en s'appuyant sur des fichiers inverses, qui associent à chaque terme l'ensemble de documents décrits par ce terme.

Pour un SRI permettant de rechercher des documents vidéo, le nombre de documents susceptibles d'être pertinents n'est pas un critère suffisant d'évaluation. En effet, la taille de ces documents peut être très importante. Il nous semble donc opportun d'ajouter un critère de taille des documents sélectionnés pour l'évaluation a priori des résultats : un seul document très long peut en effet occuper un chercheur d'information pendant quelques heures ...

4.3.6 Déformation de la requête

Si la requête rapporte pas ou trop peu de documents, elle peut être progressivement élargie dans l'ordre :

- (pour une requête booléenne) en transformant les connecteurs logiques entre les termes (les conjonctions sont transformées en disjonctions pour l'évaluation booléenne),
- (pour une requête sous forme de graphes conceptuels) en élaguant une partie du graphe, c'est-à-dire en supprimant une des projections partielles obtenues par décomposition du graphe en graphes primitifs de la base canonique, en commençant par la projection correspondant à la relation pondérée le plus faiblement,
- en passant au générique pour les termes pondérés le plus faiblement dans la requête ou rapportant trop peu de documents.

Si la requête doit être restreinte,

- les poids négatifs absolus sont pris en compte pour éliminer les documents indexés par le concept correspondant,
- (pour une requête booléenne) les disjonctions pourront être transformées en conjonctions, en commençant d'abord à l'intérieur d'une facette, puis à l'intérieur d'un attribut rapportant trop de documents,
- les termes trop génériques (rapportant trop de documents) pourront être spécifiés plus finement après interaction avec l'utilisateur,
- il pourra être demandé à l'utilisateur de mieux spécifier sa requête en y ajoutant des termes.

L'utilisation d'un thésaurus permet à ce stade de généraliser un concept ou d'aider l'utilisateur à spécifier plus finement son besoin en lui faisant des propositions.

4.3.7 Mise en correspondance requête-document

Les paramètres de la mise en correspondance

Les paramètres proposés pour la primitive de mise en correspondance sont l'importance de la mesure d'exhaustivité et l'importance de la mesure de spécificité. Ces paramètres permettent

d'obtenir des réponses plus ou moins précises à la demande d'information de l'utilisateur.

Remarque sur la spécificité vectorielle

En fait, une mesure de similarité vectorielle qui ne prendrait en compte que la mesure de spécificité privilégierait les documents qui sont indexés exactement avec les mêmes concepts que ceux qui sont exprimés dans la requête (étendue grâce aux connaissances du thésaurus) ; en effet, la mesure de spécificité du modèle vectoriel, considérée seule, mesure surtout le degré auquel un utilisateur est capable d'exprimer sa requête avec les mêmes concepts (au sens large) que l'indexeur. De plus elle suppose que tous les documents similaires sont indexés par exactement le même nombre de concepts. L'expression de la mesure de spécificité vectorielle est :

$$spe(D,Q) = \frac{D.Q}{||D||} = \frac{\sum_j d_{ij}.q_j}{\sum_i d_i^2}$$

où d_{ij} est le poids d'un terme présent à la fois dans la requête et le document pour ce document, et d_i est le poids d'indexation d'un terme décrivant le document. Lorsque tous les poids des termes de description des documents ont une valeur maximum (1), le dénominateur de cette expression revient à compter le nombre de termes qui décrivent le document. Un document décrit par plus de termes que la requête sera considéré comme moins spécifique, ainsi qu'un document décrit plus finement qu'un autre. **La mesure de spécificité vectorielle repose sur une hypothèse d'homogénéité de l'indexation**.

Pour nous, la mesure d'exhaustivité est inévitable, seule la mesure de spécificité peut être optionnelle dans un calcul de mise en correspondance pour une recherche d'information.

Mise en correspondance de graphes conceptuels

Une mise en correspondance de graphes conceptuels est forcément plus précise qu'une mise en correspondance vectorielle : elle permet de considérer une description plus raffinée des documents, comprenant non seulement des concepts mais des relations entre ces concepts. La mise en correspondance dans le modèle de représentation par graphes conceptuels consiste à utiliser l'opérateur de projection que nous avons présenté au § 3.2.4 page 74. Un document est pertinent pour la requête s'il existe une intersection non vide entre

- l'ensemble des projections de la base canonique sur les graphes décrivant les documents, chaque graphe ayant au moins un concept qui comporte un référent – nous nommons ces graphes les *graphes primitifs instanciés* des documents –, et
- l'ensemble des projections de la base canonique sur les graphes pondérés positivement dans la requête, c'est à dire les graphes primitifs de la requête instanciés.

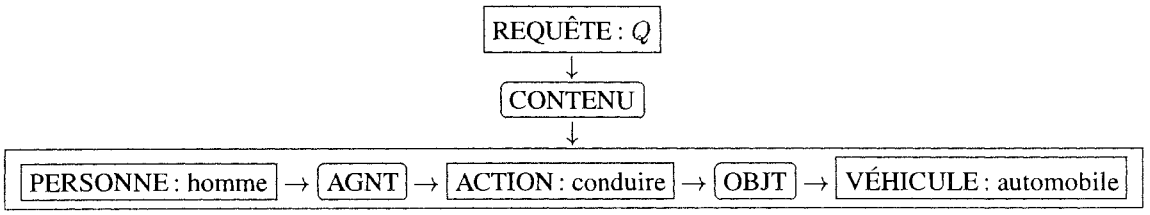
Le document doit être d'autant plus pertinent qu'il comporte davantage de graphes primitifs instanciés pondérés positivement dans la requête; il doit être d'autant moins pertinent qu'il comporte des graphes primitifs instanciés de poids négatif dans la requête.

Cette opération consiste à rechercher tout d'abord les concepts compatibles entre la requête et les documents, puis à vérifier que les relations dans lesquelles ces concepts sont impliqués se correspondent. Il est possible de définir plusieurs degrés de pertinence entre les concepts des graphes conceptuels :

1. les graphes ont des concepts non relationnels compatibles, c'est à dire égaux ou associés par une relation *sorte-de* dans le thésaurus ; par exemple, une photographie ayant pour auteur Nadar et une photographie représentant Nadar ont un concept non relationnel compatible : Nadar. Lorsque seuls les concepts non relationnels sont pris en compte, nous revenons au modèle vectoriel non ventilé par facette.

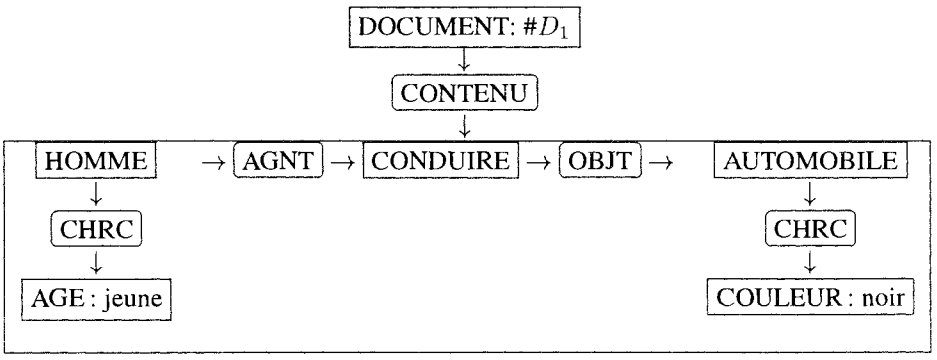
2. les concepts compatibles sont liés par les mêmes concepts relationnels. Il faut alors distinguer le cas où la relation est strictement la même du cas où la relation est de même type. Dans l'exemple ci-dessus, le concept compatible n'est pas impliqué dans le même type de relation. Par contre, deux photographies prises par Nadar ont une relation commune dans la facette signalétique.

Prenons un autre exemple sur la facette CONTENU. L'utilisateur demande des photographies où l'on voit un homme conduisant une automobile. Le graphe qui représente sa requête est le suivant :

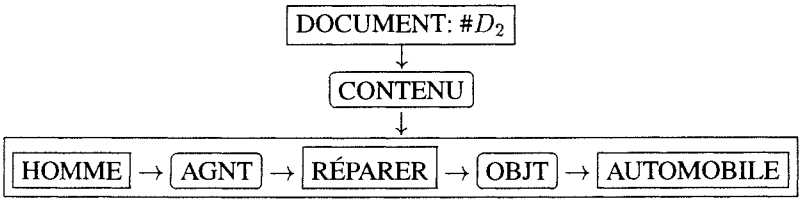


La collection de photographies contient les documents représentés par les graphes suivants :

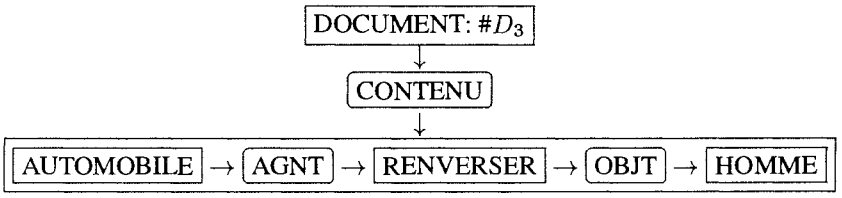
- un document D_1 où l'on voit un homme jeune conduisant une automobile noire



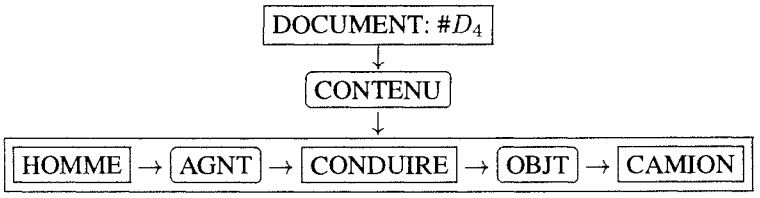
- un document D_2 où l'on voit un homme réparant une automobile



- un document D_3 où l'on voit une automobile renversant un homme



- un document D_4 où l'on voit un homme conduisant un camion



Le modèle vectoriel effectuerait le classement suivant :

rang	document	termes communs	exh	spe	sim
1	D_1	`homme, conduite, automobile`	1	3/5	3/5
2	D_2	`homme, automobile`	2/3	2/3	1/2
2	D_3	`homme, automobile`	2/3	2/3	1/2
2	D_4	`homme, conduite`	2/3	2/3	1/2

Nous pouvons remarquer que dans ce classement, le document D_1 serait pénalisé du point de vue de la spécificité vectorielle, simplement parce qu'il a un descripteur supplémentaire. Or ce descripteur n'ajoute pas un nouveau «concept» à la description du document, il complète simplement un concept existant.

Compatibilité entre les concepts

Il existe plusieurs sortes de compatibilité entre des concepts. Dans l'exemple précédent, les deux termes `camion` et `automobile` étant des feuilles du thésaurus, il ne serait pas fait *a priori* de lien entre eux, la généralisation des termes n'intervenant, dans le modèle paramétré,

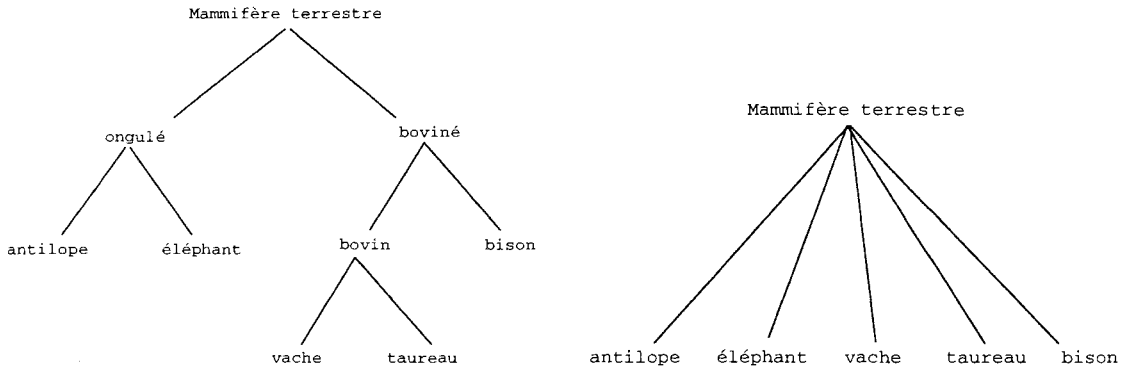
que lorsque le nombre de documents susceptibles d'être pertinents est très faible. La compatibilité entre les concepts non relationnels peut donc être estimée de deux façons :

- Deux concepts sont *strictement compatibles* s'ils sont égaux ou s'ils ont une relation de généricité/spécificité entre eux ; si les documents sont indexés aux feuilles du thésaurus, cela revient à prendre en compte la relation $d_i \leq q_i$ où d_i est un terme de description du document et q_i un terme de la requête. Nous fixons à 1 le degré de compatibilité entre des termes strictement compatibles. Deux termes n'appartenant pas au même chemin générique-spécifique allant d'un domaine à une de ses feuilles ne sont pas strictement compatibles (le degré de compatibilité stricte entre eux est de 0). L'opération de projection aux feuilles du thésaurus des termes de la requête réduit l'interprétation de la compatibilité stricte à l'égalité des concepts.
- Deux concepts sont *compatibles de façon étendue* s'ils ont un générique commun. Le degré de compatibilité des termes peut alors être fixé à l'inverse de la distance entre ces termes dans le thésaurus ; ainsi, deux termes d'un même domaine sémantique dans le thésaurus auront un degré de compatibilité inversement proportionnel au nombre de relations *is_a* qu'il faut parcourir pour atteindre leur plus petit générique commun. Le degré de compatibilité non stricte est donc défini par

$$deg_comp(t_1, t_2) = \begin{cases} 0 & \text{si } domaine(t_1) \neq domaine(t_2) \\ 1 & \text{si } t_1 = t_2 \\ \frac{1}{\sum(l_ch(t_1,t_2))+1} & \text{sinon} \end{cases}$$

  où $l_ch(t_i, t_j)$ est la longueur du chemin entre t_i et t_j, c'est-à-dire le nombre de relations *is_a* à traverser pour aller du terme t_i au terme t_j en passant par t_{gmin}, leur plus petit générique commun. Par exemple, un terme a un degré de compatibilité de 1 avec lui-même ; un terme et son générique ont un degré de compatibilité de 1/2 ; deux termes ayant le même générique direct auront une proximité sémantique de 1/3 ; deux termes incomparables pour la relation générique/spécifique ont un degré de compatibilité de 0. Le problème de ce type de mesure est qu'elle est sensible au degré de raffinement de la taxonomie de termes. Si une des branches de la taxonomie est davantage décomposée qu'une autre, leurs feuilles respectives seront plus éloignées. Par exemple, étant donné la taxonomie représentée figure 4.4.a, un taureau est moins éloigné d'un bison que ne l'est l'éléphant, alors que dans la taxonomie de la figure 4.4.b, ils sont aussi proches les uns des autres.

Si l'on avait pris en compte la compatibilité étendue pour classer les documents donnés dans l'exemple précédent, D_4 serait classé avant D_2 car `automobile` et `camion` sont très proches l'un de l'autre, alors que `conduire` et `renverser` sont plus éloignés. Ce type de compatibilité pourrait donc être utile, à condition d'établir des contraintes fortes sur la construction du thésaurus : chaque branche doit être développée avec précision. Si certaines branches du thésaurus sont des taxonomies artificielles, où les différents concepts sont regroupés «à plat» sous un générique commun uniquement pour pouvoir être intégrés au thésaurus, ce type de mesure est totalement inadapté, puisqu'aucune discrimination ne pourra être faite entre les différents termes, qui se trouvent tous au même niveau. C'est pourquoi nous ne retenons pas cette mesure



(a) Taxonomie détaillée

(b) Taxonomie à un niveau

FIG. 4.4 – *Deux exemples de thésaurus.*

de façon systématique pour notre application. Toutefois, pour certains domaines sémantiques du thésaurus, elle pourra être mise en œuvre. Il faut pour cela vérifier que la branche correspondant à la taxonomie du domaine considéré est suffisamment détaillée (contrainte de hauteur supérieure à n par exemple) et que chaque feuille se trouve à peu près à la même profondeur dans le domaine.

L'utilisation de ce type de mesure peut permettre d'améliorer le classement des documents sans présenter le même inconvénient qu'une généralisation de la requête, qui augmente le nombre de documents qui seront examinés lors de la mise en correspondance.

Exhaustivité et spécificité pour les graphes conceptuels

Pour mesurer la pertinence d'un graphe g_d représentant une description de document vis-à-vis d'un graphe requête g_q, nous nous appuyons sur l'ordre défini au § 3.2.5 pour les graphes conceptuels. Un graphe conceptuel g_d est pertinent pour g_q si $g_d \leq g_q$, c'est-à-dire s'il satisfait une au moins des conditions suivantes :

1° les concepts de g_d sont plus spécifiques que ceux de g_q, au sens de la taxonomie de spécialisation du thésaurus ; puisque l'étape de mise en correspondance est précédée d'une projection aux feuilles du thésaurus pour les concepts de la requête, cela revient à estimer si les concepts de la requête et du document sont les mêmes ;

2° g_q peut être «détaché» de g_d (g_d peut être obtenu par jointure de g_q avec un autre graphe).

Ainsi, un document est pertinent pour une requête si des projections de la requête sur le document sont possibles.

L'exhaustivité du document sera mesurée par le nombre et la taille des projections de la requête Q sur le document D, en ne retenant de l'ensemble des projections possibles que les projections maximales. L'exhaustivité d'un graphe décrivant un document devra être propor-

tionnelle au nombre de graphes primitifs qu'il a en commun avec le graphe représentant la requête.

Ceci revient à exprimer la mesure d'exhaustivité de chacun des graphes représentant le document par :

$$mexh(g_d, g_q) = \frac{\sum_{\pi_j(g) \in \{\pi(g_q) \cap \pi(g_d)\}} d_j \times q_j}{\sum_{\pi_j(g') \in \{\pi(g_q)\}} q_j^2}$$

où

$$\begin{cases} q_j \text{ est le poids du graphe primitif } \pi_j(g) \text{ dans la requête,} \\ d_j \text{ est le poids du graphe primitif } \pi_j(g) \text{ dans le document,} \\ \{\pi(g_q) \cap \pi(g_d)\} \text{ est l'ensemble des graphes primitifs communs entre le graphe requête} \\ \text{et le graphe document,} \\ \{\pi(g_q)\} \text{ est l'ensemble des graphes primitifs de la requête.} \end{cases}$$

Un document pouvant comporter plusieurs graphes de description dans la même facette, et une requête pouvant elle aussi comporter plusieurs graphes (considérés comme des alternatives), l'exhaustivité d'une facette $\mathcal{F}$ du document sera mesurée par l'exhaustivité maximum de ses graphes :

$$mexh(D_{\mathcal{F}}, Q) = max_{g_d \in D_{\mathcal{F}}, g_q \in Q} mexh(g_d, g_q)$$

Un document est spécifique vis-à-vis d'une requête s'il ne comporte que les concepts de la requête dans les mêmes relations.

$$mspe(g_d, g_q) = \frac{\sum_{\pi_j(g) \in \{\pi(g_q) \cap \pi(g_d)\}} d_j \times q_j}{\sum_{\pi_j(g') \in \{\pi(g_d)\}} d_j^2}$$

où

$$\begin{cases} q_j \text{ est le poids du graphe primitif } \pi_j(g) \text{ dans la requête,} \\ d_j \text{ est le poids du graphe primitif } \pi_j(g) \text{ dans le document,} \\ \{\pi(g_q) \cap \pi(g_d)\} \text{ est l'ensemble des graphes primitifs communs entre le graphe requête} \\ \text{et le graphe document,} \\ \{\pi(g_d)\} \text{ est l'ensemble des graphes primitifs du graphe document.} \end{cases}$$

La spécificité d'un document décrit par plusieurs graphes peut être mesurée par le maximum des mesures obtenues pour chacun des graphes. Ainsi, pour une facette $\mathcal{F}$ donnée,

$$mspe(D_{\mathcal{F}}, Q) = max_{g_d \in D_{\mathcal{F}}, g_q \in Q} mspe(g_d, g_q)$$

Une fonction compromis tenant compte des mesures d'exhaustivité et de spécificité des deux mesures précédentes peut être obtenue en faisant leur produit :

$$f_compromis(D, Q) = mexh(D, Q) \times mspe(D, Q)$$

Le tableau suivant donne ces valeurs pour l'exemple précédent. La recherche des projections compatibles donne :

D_1 : trois concepts communs sont engagés dans deux relations compatibles avec la requête, il existe donc deux projections compatibles entre le document D_1 et la requête.

D_2 : `réparer` n'est pas un spécifique de `conduire`, les deux graphes primitifs n'ont donc pas de projection compatible.

D_3 : `renverser` n'est pas un spécifique de `conduire`, les deux graphes primitifs n'ont pas de projection compatible.

D_4 : une seule projection commune est compatible :

$$\boxed{\text{PERSONNE : homme}} \rightarrow (\text{AGNT}) \rightarrow \boxed{\text{ACTION : conduire}}$$

Nous considérons pour le calcul de la spécificité que les graphes donnés dans l'exemple sont les seuls qui indexent les documents. Les mesures données utilisent uniquement le nombre de projections compatibles, tous les poids étant supposés à 1.

Document	exhaustivité GC	spécificité GC	compromis GC (exh $\times$ spe)	rang
D_1	1	2/3	2/3	1
D_2	0	0	0	3
D_3	0	0	0	3
D_4	1/2	1/2	1/4	2

Ces mesures ne portent que sur les relations communes aux documents et à la requête. Elles sont très sévères puisqu'elles éliminent D_2 et D_3 qui ont pourtant des termes communs avec la requête, mais ces termes interviennent dans des relations différentes. Elles sont donc destinées à améliorer la précision de la recherche.

Choix d'un modèle de mise en correspondance

Le choix d'un modèle de mise en correspondance dépend du type de besoin d'information de l'utilisateur. Par exemple, si l'utilisateur cherche des photographies d'églises de style gothique, sa requête sera représentée par :

$$\boxed{\text{PHOTOGRAPHIE}}$$
$$\downarrow$$
$$(\text{CONTENU})$$
$$\downarrow$$
$$\boxed{\boxed{\text{ÉGLISE}} \rightarrow (\text{CHRC}) \rightarrow \boxed{\text{STYLE : gothique}}}$$

Une demande précise ne devra aboutir qu'aux documents montrant des églises de style gothique ; cela correspond à la prise en compte des concepts et des relations entre ces concepts

pour le calcul de la pertinence. Une demande exploratoire pourra trouver des documents montrant des églises de divers styles, et divers bâtiments de style gothique ; ceci correspond à la prise en compte uniquement des concepts pour le calcul de la pertinence, et donc au modèle vectoriel. En fonction du type de besoin, nous pouvons donc mettre en œuvre une méthode de mise en correspondance orientée précision ou au contraire orientée rappel.

Distribution par facette de la mise en correspondance

La distribution par facettes de la mise en correspondance permet d'envisager des modes de mise en correspondance différents sur chacune des facettes. Elle permet également, par rapport au modèle de description par graphes conceptuels que nous avons adopté, d'éviter un niveau de récursivité dans ce traitement. En effet, si un document est décrit dans chaque facette par un graphe dont les relations sont les attributs de description, la description globale du document correspond à un «méta-graphe» qu'il faudrait explorer récursivement pour la mise en correspondance. La traitement par facette permet donc de diminuer la complexité de cette résolution.

Par ailleurs, il est souhaitable de ne comparer que ce qui est comparable : des documents peuvent avoir le même contenu (au sens de la facette CONTENU) mais représenté sous des formes différentes (facette MORPHOLOGIE). Il peut être intéressant pour un chercheur d'information de pouvoir évaluer ces différents aspects des documents de façon séparée.

Si toutes les facettes peuvent être mises en correspondance grâce à la même méthode vectorielle, le fait de distribuer le traitement à ce stade n'est utile que si l'utilisateur veut des résultats par points de vue. Sinon, il résulte de la distribution une perte de précision d'une part, et un allongement des temps de traitement d'autre part puisqu'une étape de synthèse est obligatoire ensuite.

4.3.8 Synthèse des résultats

Une mise en correspondance multi-facettes pose le problème de la synthèse des différents résultats obtenus pour effectuer un classement des documents. Le paramètre **type de classement** permet de déterminer si les résultats sont présentés de façon synthétique pour toutes les facettes, ou s'ils conservent leur indépendance. Par ailleurs, les formules de synthèse proposées dans le modèle paramétré d'origine sont à notre sens insuffisantes : le minimum et le maximum des valeurs obtenues pour les différentes facettes sont des fonctions de synthèse trop restrictives : elles sont équivalentes à ne faire porter la mise en correspondance que sur une seule facette. La moyenne des mesures de similarité obtenues ne peut être calculée que si ces mesures sont comparables et homogènes, ce qui était le cas pour le modèle paramétré qui a été élaboré autour d'une mise en correspondance vectorielle. Nous devons donc élargir le problème de la synthèse pour tenir compte des méthodes de mise en correspondance diverses et pas forcément homogènes pour chacune des facettes.

Plutôt que de tenir compte des mesures de similarité obtenues pour chacune des facettes, nous proposons de ne considérer que le classement obtenu dans chacune d'elles. Ce mode de

synthèse peut être concrétisé par une sorte de vote, où chaque facette disposerait d'un certain nombre de voix qu'elle affecterait en priorité aux documents qui se retrouvent les mieux classés pour elle. La synthèse se réduit alors à une sorte de dépouillement du vote, c'est-à-dire au nombre de voix obtenues par chaque document candidat au classement. Nous reviendrons sur ce vote au chapitre 5 page 188.

Il est possible également d'envisager de faire travailler chaque facette à la fois selon les deux modèles de mise en correspondance existants, chaque modèle donnant des résultats sensiblement différents (voir l'exemple donné en page 115). Des travaux en ce sens ont déjà été entrepris en RI, par exemple [Bartell 94] ou [Thompson 90].

4.3.9 Préparation de la visualisation des résultats

La mesure de similarité permet de déterminer des classes de documents par degré de pertinence. Lors de la visualisation, seuls les documents les plus pertinents sont proposés à l'utilisateur. Ceci amène à considérer un paramètre de **censure** qui détermine le nombre de documents qui pourront être visualisés par l'utilisateur : les classes de pertinence dont le nombre cumulé de documents permet d'atteindre la valeur du paramètre de censure seront présentées. Ainsi, les documents les plus pertinents sont proposés à l'utilisateur, jusqu'à ce que leur nombre cumulé atteignent ou dépasse le seuil fixé. Lorsque le système n'arrive pas à constituer des classes de documents dont le nombre cumulé corresponde au critère de censure (éventuellement à un pourcentage ϵ près), il existe alors un conflit dans le système, et une négociation peut être établie pour le résoudre. Dans RIVAGE, le système fait appel directement à l'utilisateur pour le résoudre : il doit accepter ou non de voir plus de documents que prévu.

Une autre possibilité pour résoudre ce conflit pourrait consister à appliquer une mesure de correspondance complémentaire pour départager les documents d'une même classe. Dans l'exemple que nous avons détaillé, la mise en correspondance selon le modèle vectoriel détermine seulement deux classes de pertinence, alors que le modèle de graphes conceptuels en détermine trois. La mise en œuvre de plusieurs méthodes différentes de mise en correspondance et/ou de synthèse peut donc éviter de faire systématiquement appel à l'utilisateur pour résoudre un tel conflit.

Par ailleurs, en fonction de l'interface de présentation des documents résultats, il peut être nécessaire d'effectuer une préparation de la présentation. C'est le cas en particulier pour une présentation hypermédia des documents résultats : le graphe résultant doit être constitué à ce stade. C'est aussi le cas si les échanges entre l'application et l'interface sont faits selon un standard d'échange comme MHEG ou OMFI : les objets de présentation correspondants doivent être générés.

Cette primitive particulière, de plus en plus pertinente dans les nouveaux environnements de RI, doit être intégrée dans les primitives du processus.

4.3.10 Visualisation des résultats et choix

Lors de la visualisation, l'utilisateur peut se prononcer sur la pertinence des documents vis-à-vis de son besoin. Ces jugements émis par l'utilisateur lui-même permettent au système d'avoir un retour immédiat et d'ajuster sa recherche : le moteur de recherche peut repartir en chaînage arrière pour modifier les hypothèses que représente la requête, et éventuellement émettre de nouvelles hypothèses. Le modèle paramétré propose d'attacher à cette primitive un paramètre de **finesse de choix**, représentant le nombre de valeurs qualitatives que la fonction de choix propose à l'utilisateur et qui correspondent à des valeurs dans l'intervalle [-1,1]. Une valeur de -1 indique une non pertinence totale du document, alors qu'une valeur de 1 indique sa pertinence totale [14]. La fonction de choix peut proposer des valeurs intermédiaires, permettant à l'utilisateur de nuancer son jugement. À noter que la valeur 0 exprime la neutralité, ce qui est équivalent au fait que l'utilisateur ne se prononce pas.

Dans le modèle paramétré, les choix peuvent affecter toutes les facettes d'un document de façon globale, ou seulement quelques-unes d'entre elles. Nous souhaitons que l'utilisateur puisse faire des choix contradictoires d'une facette à l'autre : un document peut être rejeté pour ses qualités techniques mais adopté pour son contenu. De plus, les jugements de l'utilisateur doivent pouvoir porter sur différents degrés de granularité des descriptions de documents : concept, ensemble des concepts d'un attribut, ensemble des concepts d'une facette, description globale du document.

4.3.11 Reformulation

Rappelons que, dans le modèle d'origine, les paramètres attachés à la primitive de reformulation sont au nombre de quatre :

- la **contribution des facettes** représente le poids de chaque facette dans la reformulation de la requête. Un poids nul indique qu'une facette ne sera pas prise en compte lors de la reformulation.

- la **contribution de l'ancienne requête** indique le degré auquel la requête précédente sera prise en compte lors de la nouvelle étape. C'est le paramètre α dans la formule (4.1), page 103.

- la **contribution des choix** permet de moduler l'importance des termes présents dans les documents choisis. C'est le paramètre β dans la formule (4.1).

- la **contribution des rejets** permet de prendre plus ou moins en compte les documents jugés négativement ou rejetés par l'utilisateur. C'est le paramètre γ dans la formule (4.1).

14. Cette fonction de choix de documents ne doit pas être confondue avec la fonction qui permet à l'utilisateur de pondérer l'importance des termes de sa requête.

Importance des facettes pour la reformulation

Il faut décider des facettes qui vont participer à la reformulation. Il peut s'agir uniquement des facettes concernées par les concepts de la requête initiale, ou de toutes les facettes. Cette dernière solution encourage la découverte de nouvelles corrélations entre les documents choisis, et peut permettre de «comprendre» pourquoi un document a été rejeté. Ceci est particulièrement important pour la recherche d'images (fixes ou animées), car nous nous sentons tous habilités à lire des images, mais peu de personnes ont reçu une formation à la lecture de ces images ; ainsi, le langage graphique et audio-visuel peut jouer un rôle important dans les choix et rejets de l'utilisateur sans qu'il en soit conscient. La facette MORPHOLOGIE, peu sollicitée directement en général par les utilisateurs dans leurs requêtes [Turner 94], pourra prendre toute son importance lors de cette phase du processus.

Peut-on émettre une règle qui détermine l'intervention des facettes en fonction du type de besoin ? Pour un besoin précis, il faut favoriser la convergence, mais on ne sait pas a priori quel est l'aspect des documents qui provoquera cette convergence. Il nous semble donc difficile d'émettre une règle dans cette situation. Le fait d'affecter un poids fort à la requête précédente peut effectivement faire converger la recherche si les termes ont été bien choisis par l'utilisateur. Pour un besoin exploratoire, on doit encourager la découverte, mais la découverte peut se faire au sein de la même facette ou au contraire en favoriser de nouvelles. Pour un besoin thématique, on peut encourager la convergence dans la facette correspondant au thème de la recherche, et la découverte dans les autres facettes. Pour un besoin connotatif, on favorise les facettes «objectives», comme celles correspondant à la description du contenu, de la morphologie ou de la signalétique du document. Par rapport au modèle paramétré, nous suggérons donc que chaque facette puisse adopter des paramètres différents pour donner une importance nuancée à l'ancienne requête, aux choix et aux rejets. Les facettes y gagnent alors en autonomie, et les facettes non concernées par la requête initiale ont plus de chance d'intervenir lors de cette étape. Ceci multiplie le nombre de paramètres stratégiques définis pour cette phase par le nombre de facettes.

Prise en compte des choix et des rejets

Dans la fonction de reformulation utilisée par le modèle paramétré, le choix effectué par l'utilisateur est pris en compte sous la forme $max(fchoix(d_i, \mathcal{F}), fchoix(d_i, nil))$ pour indiquer que l'on prend en compte la valeur de choix la plus forte entre un choix portant sur une facette et un choix portant sur le document global (voir formule 4.1 page 103). Ceci ne permet pas de prendre en compte la modération du choix de l'utilisateur pour une facette particulière. Au contraire, il faudrait toujours prendre en compte en priorité le choix de plus fine granularité (dans l'ordre terme, attribut, facette, description globale). En effet, si l'utilisateur spécifie un choix pour une granularité de description plus fine, celui-ci doit toujours être prioritaire. La valeur du choix porté sur un terme ou une facette particulière des documents peut être inférieure à la valeur de choix du document global : cela correspond intuitivement à exprimer une sorte d'opérateur $SAUF$. Par exemple, une photographie peut convenir globalement pour son contenu, mais moins pour son type de cadrage. Si l'utilisateur choisit globalement un document

mais fait un choix plus mitigé pour une facette, le résultat doit consister à augmenter la valeur des poids de tous les termes présents dans les descriptions des documents choisis, sauf pour la facette concernée (modulé par les paramètres β et γ).

Reprenons l'exemple précédent où un utilisateur choisit une photographie qui lui convient très bien globalement, mais moins pour sa facette MORPHOLOGIE. Pour un terme relatif à la facette CONTENU, sur laquelle l'utilisateur ne s'est pas prononcé de façon détaillée, c'est le choix porté sur le document global qui doit être pris en compte. Pour un terme relatif à la facette MORPHOLOGIE, c'est la valeur du choix relatif à cette facette qui doit être retenue. Si la fonction $fchoix()$ est considérée comme monotone sur l'intervalle [-1,1], -1 exprime un rejet absolu, 0 exprime la neutralité, 1 exprime le choix absolu, et des valeurs intermédiaires permettent de nuancer les choix/rejets. Dans l'exemple considéré, le document est choisi globalement et aucun choix n'a été émis au niveau de la facette CONTENU :

$$fchoix(d_i, nil) = 1 \text{ et } fchoix(d_i, \mathcal{F}_{contenu}) = 0,$$

soit

$$max(fchoix(d_i, \mathcal{F}_{contenu}), fchoix(d_i, nil)) = 1.$$

Par contre, pour la facette MORPHOLOGIE, un choix a été émis au niveau de la facette :

$$fchoix(d_i, \mathcal{F}_{morphologie}) = 0,5$$

et cette valeur est inférieure à $fchoix(d_i, nil) = 1$. L'utilisation de la fonction $maximum$ nous semble contre-intuitive pour représenter ces alternatives, et ne permet pas de tenir compte de la modération d'un choix pour une facette du document. En fait, l'examen des choix doit avoir lieu par ordre de granularité croissante. Si la granularité des descriptions est considérée comme ordonnée par :

$$terme \leq attribut \leq facette \leq description \quad globale$$

c'est une fonction $minimum$ qu'il faut utiliser, cette fonction portant sur l'ordre de granularité des descriptions. Nous exprimons donc la fonction $fchoix()$ sous la forme $fchoix(t, g)$ pour dire que le terme t a été choisi au niveau g.

Si la fonction $fchoix()$ est considérée comme monotone continue et à valeur dans l'intervalle [-1,1], l'expression 4.1 page 103 implique qu'il n'est pas possible de choisir un document en rejetant une facette ou vice-versa. Nous souhaitons donner cette possibilité à l'utilisateur, et devons donc distinguer les valeurs positives et négatives de la fonction de choix, en introduisant par exemple deux fonctions distinctes $fchoix_pos()$ et $fchoix_neg()$ qui seraient paramétrées respectivement par les paramètres β et γ. La valeur de ces fonctions, par rapport à une seule fonction $fchoix()$, continue et monotone, et prenant ses valeurs dans l'intervalle [-1,1], peut être calculée de façon simple à chacun des niveaux retenus pour la description. Si nous faisons abstraction ici des facettes, la fonction associée à la prise en compte des choix positifs $fchoix_pos$

pour un terme t s'exprime par :

$$\begin{aligned} fchoix_pos(t) &= fchoix(t) \text{ si } fchoix(t) > 0 \\ &= 0 \text{ si } fchoix(t) \leq 0 \end{aligned}$$

soit

$$fchoix_pos(t) = \frac{fchoix(t) + |fchoix(t)|}{2}$$

De même, pour les rejets, une fonction associée $fchoix_neg$ s'exprime par :

$$\begin{aligned} fchoix_neg(t) &= fchoix(t) \text{ si } fchoix(t) < 0 \\ &= 0 \text{ si } fchoix(t) \geq 0 \end{aligned}$$

soit

$$fchoix_neg(t) = \frac{fchoix(t) - |fchoix(t)|}{2}$$

Ainsi, la fonction de reformulation du poids d'un terme t dans la facette $\mathcal{F}$ s'exprimerait par :

$$\begin{aligned} q_i(t_{\mathcal{F}}) &= \alpha q_{i-1}(t_{\mathcal{F}}) \\ &+\beta(\frac{1}{|D_c|}\sum_{d_j \in D_c} min_{\mathcal{G}}(fchoix_pos(d_{(j,t_{\mathcal{F}})}, niv))) \\ &+\gamma(\frac{1}{|D_r|}\sum_{d_k \in D_r} min_{\mathcal{G}}(fchoix_neg(d_{(k,t_{\mathcal{F}})}, niv))) \end{aligned} \tag{4.2}$$

où
$q_i(t)$ est le poids du terme t à l'étape i,
D_c est l'ensemble des documents ayant donné lieu globalement ou partiellement à un choix positif,
D_r est l'ensemble des documents ayant donné lieu globalement ou partiellement à un choix négatif,
$d_{(j,t)}$ est le poids du terme t dans le document j,
$fchoix_pos(d, niv)$ (respectivement $fchoix_neg(d, niv)$) est la valeur positive (resp. négative) rendue par la fonction de choix proposée au niveau niv,
niv est le niveau auquel le choix est exprimé, $niv \in \mathcal{G} = \{terme, attribut, \mathcal{F}acette, document\}$.
$min_{\mathcal{G}}$ retourne la valeur de choix émise au niveau de granularité le plus fin.

Le fait de pouvoir effectivement récupérer des valeurs pour les différents niveaux de granularité est un problème d'interface d'une part, et de protocole de communication d'autre part : le niveau de granularité auquel le choix a été fait devra être spécifié explicitement.

4.3.12 Reformulation en présence de descriptions par graphes conceptuels

À notre connaissance, aucune proposition n'a été faite à ce jour pour mettre en œuvre une reformulation de la requête en présence de graphes conceptuels. Nous allons donc faire des propositions en ce sens. Une reformulation de la requête à partir des choix de l'utilisateur peut être un moyen d'obtenir une description de la requête sous forme de graphes conceptuels si la requête a été entrée sous forme d'une liste de mots-clés pondérés. Dans tous les cas, le but de la reformulation est de trouver une meilleure expression du besoin.

Lorsque l'utilisateur choisit des documents, les graphes conceptuels qui les décrivent sont considérés comme pertinents pour son besoin d'information (ou une partie d'entre eux si le choix porte sur un niveau de granularité inférieur). Chaque graphe peut être alors associé à un poids de pertinence utilisateur qui prend des valeurs entre - 1 et + 1 : - 1 signifie que le graphe n'est pas pertinent pour le besoin d'information, + 1 signifie qu'il est totalement pertinent.

La reformulation de la requête peut consister à trouver la généralisation commune maximale de tous les graphes pertinents d'une part, puis à élaguer ces graphes en supprimant les portions de graphes non pertinentes. De façon symétrique, il faut trouver la généralisation commune maximale de tous les graphes non pertinents. Toutefois, trouver le graphe maximal commun qui généralise un ensemble de graphes est un problème NP-complet [Bournaud 95, p178] : il faut en effet examiner tous les nœuds du graphes, et pour chaque nœud, examiner toutes les relations qu'il peut avoir avec les autres nœuds. Il faut donc trouver une solution partielle et un algorithme de reformulation plus performant.

Une solution partielle consiste à ne considérer d'abord que les graphes primitifs. Affecter à chaque graphe primitif un nouveau poids tenant compte de son ancien poids, du nombre de fois où il a été choisi, et du nombre de fois où il a été rejeté :

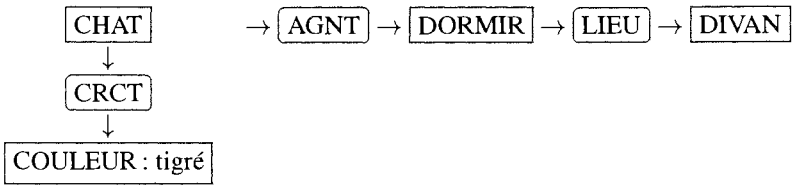
$$\begin{aligned} q_i(gp) &= \alpha q_{i-1}(gp) \\ &+\beta(\frac{1}{|D_c|}\sum_{d_j \in D_c} min_{\mathcal{G}'}(fchoix_pos(gp_{d_j}, ng))) \\ &+\gamma(\frac{1}{|D_r|}\sum_{d_k \in D_r} min_{\mathcal{G}'}(fchoix_neg(gp_{d_k}, ng))) \end{aligned} \tag{4.3}$$

où
$q_i(gp)$ est le poids du graphe primitif gp pour la requête à l'étape i,
D_c est l'ensemble des documents ayant donné lieu globalement ou partiellement à un choix positif,
D_r est l'ensemble des documents ayant donné lieu globalement ou partiellement à un choix négatif,
$fchoix_pos(gp, ng)$ (respectivement $fchoix_neg(gp, ng)$) est la valeur positive (resp. négative) rendue par la fonction de choix proposée au niveau de granularité ng,
$ng \in \mathcal{G}' = \{graphe, facette, document\}$,

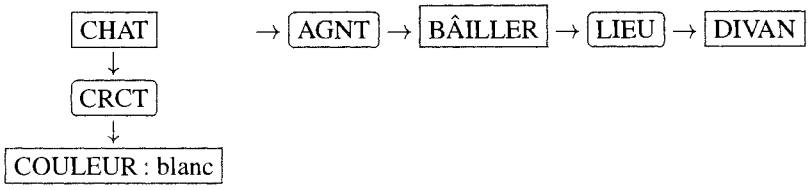
Supposons qu'un utilisateur ait formulé la requête :

CONTENU : (`chat`,1), (`divan, 1`)

Parmi les images présentées par le système, l'utilisateur a choisi les images suivantes :



et rejeté cette image :



Pour cet exemple, nous obtenons l'ensemble de graphes primitifs suivant (les valeurs données pour l'ancien poids résultent de la moyenne des poids des concepts impliqués dans la relation) :

	Graphe primitif	Ancien poids	Valeur moyenne des choix	Valeur moyenne des rejets
g_1	CHAT → AGNT → DORMIR	1/2	1	0
g_2	DORMIR → LIEU → DIVAN	1/2	1/2	0
g_3	DORMIR → LIEU → FAUTEUIL	1/2	1/2	0
g_4	CHAT → AGNT → BÂILLER	1/2	0	1
g_5	BÂILLER → LIEU → DIVAN	1/2	0	1
g_6	CHAT → CRCT → COULEUR : blanc	1/2	0	1
g_7	CHAT → CRCT → COULEUR : tigré	1/2	1/2	0

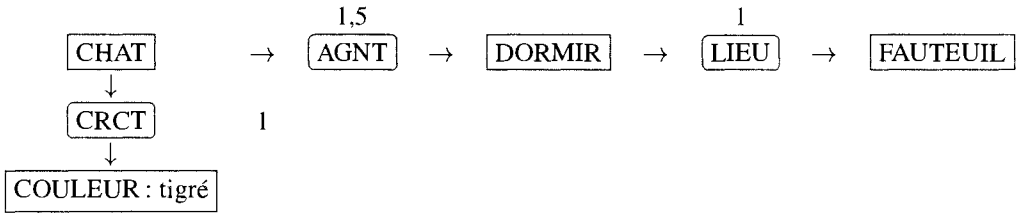
Le graphe g_7 par exemple fait partie de la moitié des documents choisis, et ne fait partie d'aucun document rejeté. Pour chaque graphe primitif, un poids de pertinence peut être calculé en appliquant la formule 4.2 de la page 126 aux graphes primitifs et non plus aux termes.

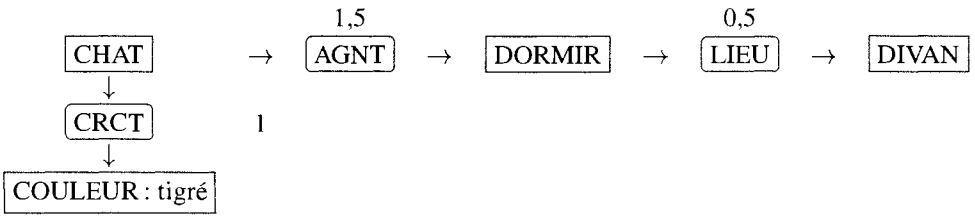
Pour obtenir un ancien poids pour les graphes dans le cas où la requête précédente n'était pas exprimée dans le formalisme des graphes conceptuels, on calcule pour chaque graphe primitif un poids approximatif par la moyenne des poids des concepts qui interviennent dans la relation à l'étape précédente.

La reformulation sur les graphes primitifs, en supposant $\alpha = \beta = \gamma = 1$, affecte les nouveaux poids w_{i+1} suivants :

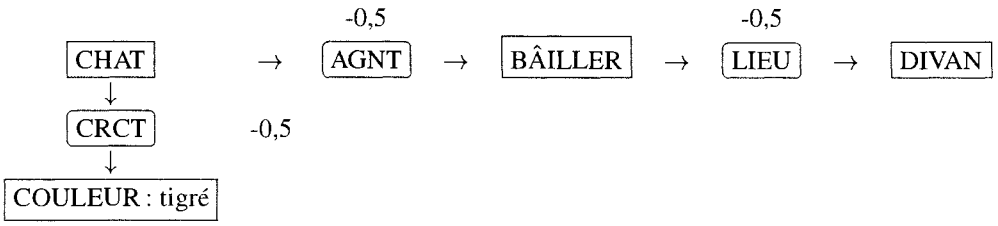
g_i	graphe primitif					w_{i+1}
g_1	CHAT	→	AGNT	→	DORMIR	1,5
g_2	DORMIR	→	LIEU	→	DIVAN	1
g_3	DORMIR	→	LIEU	→	FAUTEUIL	1
g_4	CHAT	→	AGNT	→	BÂILLER	-0.5
g_5	BÂILLER	→	LIEU	→	DIVAN	-0.5
g_6	CHAT	→	CRCT	→	COULEUR : blanc	-0.5
g_7	CHAT	→	CRCT	→	COULEUR : tigré	1

La jointure maximale des graphes pondérés positivement donnerait :





La jointure maximale des graphes pondérés négativement donnerait :



Opérer réellement cette jointure maximale n'est pas nécessaire, sauf pour présenter la requête reformulée à l'utilisateur sous forme de graphes conceptuels. Dans ce cas, nous obtiendrions deux requêtes représentatives de ce que souhaite retrouver l'utilisateur, et une anti-requête représentative de ce qu'il ne souhaite pas retrouver.

La reformulation vectorielle, avec les mêmes valeurs pour les paramètres ($\alpha = \beta = \gamma = 1$), affecte les poids suivants :

t_i	w_i	choix	rejets	poids reformulé w_{i+1}
chat	1	1	1	1
dormir	0	1	0	1
divan	1	1/2	1	0,5
tigré	0	1/2	0	0,5
bâiller	0	0	1	-1
blanc	0	0	1	-1
fauteuil	0	1/2	0	0,5

Tous les concepts pondérés positivement par la reformulation vectorielle se retrouvent dans les graphes pondérés positivement par la reformulation conceptuelle. Mais avec cette dernière, seules certaines relations entre ces concepts sont estimées pertinentes : ainsi, un chat en train de bâiller ne sera pas pertinent, alors qu'un chat en train de dormir le sera. Avec la reformulation conceptuelle, le concept `chat` est considéré pertinent seulement dans le contexte où il apparaît en train de dormir sur un divan ou sur un fauteuil et s'il s'agit d'un chat tigré. Les documents dont les descriptions comprennent des graphes primitifs pondérés négativement par la reformulation seront forcément moins pertinents à l'étape suivante de recherche.

Le même calcul de reformulation sur les graphes primitifs avec $\alpha = 0$ et $\beta = \gamma = 1$ donne :

g_i	graphe primitif					w_{i+1}
g_1	CHAT	→	AGNT	→	DORMIR	1
g_2	DORMIR	→	LIEU	→	DIVAN	0,5
g_3	DORMIR	→	LIEU	→	FAUTEUIL	0,5
g_4	CHAT	→	AGNT	→	BÂILLER	-1
g_5	BÂILLER	→	LIEU	→	DIVAN	-1
g_6	CHAT	→	CRCT	→	COULEUR : blanc	-1
g_7	CHAT	→	CRCT	→	COULEUR : tigré	0,5

Avec les mêmes valeurs pour les paramètres α, β et γ, la reformulation vectorielle aurait affecté les poids suivants aux termes :

t_i	choix	rejets	poids reformulé
chat	1	1	0
dormir	1	0	1
divan	1/2	1	-0,5
tigré	1/2	0	0,5
bâiller	0	1	-1
blanc	0	1	-1
fauteuil	1/2	0	0,5

La reformulation vectorielle considère alors le terme `chat` comme neutre (il est «oublié» par le système) et le terme `divan` comme plutôt non pertinent. Le système découvre les termes assez pertinents `dormir` et `tigré`. Cette interprétation vectorielle des jugements de l'utilisateur

est donc assez violente du point de vue de ce qu'elle oublie ... Elle correspond à un parcours exploratoire d'une collection par changement continu de sujet : un premier sujet exprimé dans la requête initiale est ensuite prolongé en ne considérant qu'un de ses aspects dans la seconde requête, qui par association d'idée, donne un troisième sujet, etc. Ceci peut être rapproché de ce que Udo Hahn nomme une *thématisation par association d'idées* [Hahn 90] dans son analyse de la cohérence de textes par rapport à un thème. S'il s'agit bien d'un type d'exploration possible d'une collection, il amène facilement l'utilisateur à se perdre.

La méthode de reformulation proposée considère la pertinence des relations plutôt que la pertinence des seuls concepts non relationnels. Elle permet donc de découvrir les **contextes pertinents** pour les concepts sous-jacents au besoin d'information. Ce peut être aussi un moyen pour obtenir une expression de la requête sous forme de graphes conceptuels à partir d'une requête initiale formulée sous forme de mots-clés.

Cette solution de reformulation sur les graphes conceptuels n'est que partielle : elle permet seulement d'obtenir un ensemble de graphes primitifs pondérés, et non leur jointure maximale.

Combiner reformulation vectorielle et conceptuelle

Les concepts qui ne se trouvent que dans des images choisies sont estimés pertinents : toutes les relations auxquelles ils appartiennent sont retenues. Les concepts qui ne se trouvent que dans des images rejetées sont estimés non pertinents : toutes les relations auxquelles ils appartiennent pourraient être éliminées ou mieux conservées comme contre-exemples du besoin. Pour les concepts qui se trouvent à la fois dans les descriptions de documents choisies et rejetées, on examine leur rôle dans la relation où ils interviennent. S'il existe des rôles pour lesquels le concept est toujours choisi, le concept est jugé pertinent dans ce rôle. S'il existe des rôles pour lesquels il est toujours rejeté, le concept est jugé non pertinent dans ce rôle. Pour chaque concept, il est possible d'éliminer les relations qui correspondent à des rôles non pertinents pour ce concept, ou mieux de les conserver avec leur poids négatifs : les documents qui contiennent ces relations négatives seront moins bien classés à l'étape suivante de mise en correspondance.

4.3.13 Bilan

Le modèle paramétré de RI constitue un bon point de départ pour l'analyse d'un processus de recherche interactive. Nous avons malgré tout dû le compléter, en envisageant d'autres modes de représentation que des vecteurs pour la requête et les documents et en levant l'hypothèse d'intégration du SRI et de son interface. C'est ainsi que nous avons étudié l'impact de représentations par graphes conceptuels dans ce modèle et que nous y avons ajouté des primitives supplémentaires pour tenir compte de la non intégration de l'interface et du système de recherche proprement dit. Les primitives de notre modèle sont, pour résumer :

- formulation de la requête par l'utilisateur,
- interprétation sémantique de la requête,

- formalisation de la requête,
- proaction : évaluation a priori, déformation éventuelle de la requête (généralisation/ spécialisation), présélection de documents,
- mise en correspondance,
- synthèse des résultats et préparation de la visualisation,
- visualisation des documents et choix,
- analyse des choix de l'utilisateur et reformulation de la requête.

Nous avons proposé également de traiter plus finement les jugements de l'utilisateur sur les documents que lui présente le SRI, de façon à ce qu'il puisse émettre un choix différent sur différents aspects du documents : par exemple choisir globalement un document sauf une de ses facettes.

Par ailleurs, le modèle paramétré de RI a été élaboré pour des documents non structurés. Or, la RI dans des documents structurés pose quelques problèmes spécifiques que nous allons étudier dans la suite de ce chapitre.

4.4 Les documents structurés et la vidéo

La RI dans des collections de documents structurés a déjà donné lieu à de nombreuses applications : recherche d'articles précis dans des encyclopédies, d'informations précises dans les FAQ (*Frequently Asked Questions* – les questions fréquemment posées des forums de discussion électroniques)–, utilisation de techniques de RI pour rechercher une partie d'un hyperdocument. Nous voulons prendre en considération la structure des documents pour permettre la recherche de passages particuliers dans des documents vidéo longs par exemple. Nous allons d'abord détailler les spécificités de cette problématique en RI avant d'envisager de tenir compte de la structure des documents dans notre modèle.

4.4.1 Spécificités de la recherche d'information dans des documents structurés

La recherche d'information dans des documents structurés est une problématique particulière de la RI où la structure des documents est exploitée pour améliorer la recherche. La structure exploitée dans le document peut être sa structure logique ou sa structure physique. Cette problématique concerne, par exemple, la recherche de passages dans des textes volumineux comme les encyclopédies [Salton 94], les collections de messages électroniques [Fox 87] ou la documentation technique [Paice 93].

Plusieurs raisons motivent ces travaux. D'une part, il est souhaitable d'alléger la surcharge cognitive de l'utilisateur confronté à des documents très longs. C'est dans ce but que des chercheurs travaillent à la génération automatique de résumés de textes [Hahn 90, Paice 93], ou à la génération de liens hypertextes dans des collections de documents [Salton 93, Salton 94, Davis 92, Davis 93, Li 92]. D'autre part, une recherche d'information classique sur ce type de documents peut donner des scores trop faibles, parce que ces documents couvrent un certain nombre de sujets différents. Les approches traditionnelles en RI ont été en effet conçues initialement pour rechercher des informations non dans des documents entiers mais dans des représentants courts de ces documents (titres, résumés de textes ou références bibliographiques). À présent, les documents entiers – textes ou images numérisées – sont disponibles en ligne et un accès sélectif à des portions particulières de documents est nécessaire pour répondre à une demande d'information.

Le développement des hyperdocuments (hypertextes/hypermédias), où des portions de textes, des images, des séquences vidéo ou des extraits sonores sont associés au sein d'hypergraphes, a entraîné la constitution de nouvelles collections. Ces documents peuvent être considérés comme des documents structurés [Kheirbek 95]. Un accès associatif aux éléments d'information est possible grâce à des fonctions de navigation. Les techniques de RI peuvent être exploitées pour générer automatiquement des liens hypermédias donnant un accès transversal dans la structure de ces énormes documents ([Salton 94, Davis 92, Davis 93, Li 92, Kheirbek 95]).

Dans la plupart des travaux, l'accent est mis sur le contexte dans lequel un terme est utilisé. En effet, si pour certains documents la perception qu'en a l'utilisateur est fondée sur le contenu,

pour d'autres elle peut reposer sur l'apparence (la présentation) ou encore sur la structure logique [Meghini 91]. Le contexte considéré peut ainsi être physique ou logique. Par exemple, pour une recherche d'information textuelle, plus que le sens intrinsèque d'un terme, c'est la signification d'une portion de texte qui est pertinente. Pour une image, le contexte d'un objet peut être défini comme la partie ou région de l'image dans laquelle il se trouve; c'est l'approche adoptée dans MULTOS [Rabitti 91], ou dans MMIS [Goble 92] où les images sont divisées en *blots*[15]. La division des images en plusieurs blocs est aussi exploitée dans le projet QVE (Query by Visual Example) [Hirata 92], qui étudie l'interrogation d'une base d'images sur le contenu grâce à une requête exprimée sous forme d'un croquis simple.

La définition de la structure d'un document diffère selon les approches : il peut s'agir d'une structure pré-établie et même codée dans le texte (par exemple grâce à des standards comme SGML [ISO-8879 86] ou HTML) ou codée avec le texte (standard ODA [ISO/DIS8613-1 88]), ou d'une structure «motivée», calculée par le système en fonction de la proximité sémantique d'objets élémentaires. Nous constatons que ces deux points de vue cohabitent dans les travaux visant à intégrer SRI et systèmes hypermédias : les liens entre documents ou parties de documents peuvent en effet être prédéfinis ou bien construits *a posteriori* en réponse à une requête.

4.4.2 Structuration des documents vidéo

Nous adoptons les hypothèses suivantes pour la modélisation de la structure des vidéos : la structure pré-définie correspond à celle définie par l'auteur lors du travail de montage. C'est la structure par défaut, qui sera exploitée si aucun autre critère n'intervient pour en imposer une autre. Nous avons retenu une structuration en scènes et en plans qui respecte le travail de montage réalisé sur les documents et avons montré au §1.2.4 (page 12) que pour des documents télévisés cette structuration était pertinente. Toutefois, l'exploitation que font certains chercheurs en sciences humaines de ces documents impose une autre structure, les critères de structuration n'étant plus déterminés par le montage mais par le contenu du document : par exemple, un éclat de rire du public dans un débat politique peut provoquer un tournant dans le débat, sans que cela se traduise par des indices visuels si le public est hors image (*OFF* dans le langage audiovisuel). C'est pourquoi une autre structuration *à la demande* devrait pouvoir être définie par un utilisateur. Cette restructuration peut être obtenue simplement grâce au marquage du document pour définir de nouvelles limites de segments pertinents. L'hypothèse sur laquelle repose notre modèle est que l'ordre de succession des séquences dans un document est immuable, seules les limites de ces séquences peuvent varier. Notre but en effet n'est pas l'édition de documents vidéo, qui est une autre problématique.

Nous devons donc définir un modèle de structure de document par défaut : pour chaque document, définir les séquences qui le composent, et pour chaque séquence, les plans qui la composent. Si le document est considéré comme un intervalle [image_1,image_n], la structuration logique du document revient à identifier des intervalles particuliers, cette opération pouvant être récursive. Un intervalle peut avoir ses propres descriptions, mais bénéficie également des

15. blots (Basic Logical ObjecTS) = régions distinctes de l'image, à ne pas confondre avec les *blobs*, fichiers binaires utilisés dans les SGBD pour stocker les images.

descriptions de ses sous-intervalles. De même, il doit pouvoir «hériter» des descriptions des intervalles qui l'englobent.

Nous décidons d'attacher les descriptions à un niveau quelconque de granularité des documents. Par exemple :

Descriptions	Identification du document	Intervalle concerné
Descr_1	#145	tout le document
Descr_2	#145	seq#1
Descr_3	#145	seq#2
Descr_4	#145	plan#1
Descr_5	#145	plan#2

La description de la structure logique du document doc_id#145 sera d'une part :

Structure par défaut des documents			
doc_id	identification de la scène	composition de la scène	
		plan de début	plan de fin
#145	seq#1	#1	#3
#145	seq#2	#4	#4

et d'autre part la définition des plans :

Composition des plans			
doc_id	identification du plan	image début	image fin
#145	plan#1	#001	#060
#145	plan#2	#061	#120
#145	plan#3	#121	#200
#145	plan#4	#201	#280

Cette façon de prendre en compte les descriptions par rapport à la structure du document permet de retrouver quelles sont les descriptions d'une séquence ou d'un document donnés.

4.4.3 Attribution de types aux éléments de la structure du document

Lorsque la structure du document est figée ou prédéfinie, il existe plusieurs façons de la prendre en compte. Pour certains travaux, seule la modularité ainsi définie est importante, l'objectif étant de raccourcir la partie de document qui sera présentée à un utilisateur en réponse à une requête. C'est le cas des travaux de Salton [Salton 93, Salton 94]. Pour d'autres, les sous-structures sont typées et leur type intervient pour donner une importance nuancée aux concepts qui s'y trouvent. Ainsi, imaginons un document de type article, structuré en en-tête (titre, auteurs, résumé), corps (sections, sous-sections, paragraphes) et conclusion; les descriptions du document auront une importance de représentativité différente en fonction du type de structure

dans laquelle elles se trouvent : les concepts du titre ou de la conclusion sont considérés comme plus représentatifs que ceux du corps de l'article. Cette approche, que l'on trouve par exemple dans [Paice 93], nécessite une base de connaissance détaillée sur les différents types de documents que le système peut traiter : il faut spécifier, pour chaque type de document, les éléments qui permettent de moduler l'importance des concepts. Ceci peut faire partie de la modélisation des connaissances du domaine. Par exemple, dans un journal télévisé, le générique de début et de fin a peu d'importance pour une recherche sur le contenu informatif de ces émissions. Les reportages d'actualités importants sont en général présentés au début du journal, alors que la fin du journal est souvent constituée de séquences magazines. Il s'agit d'une connaissance propre à un domaine d'application particulier, et qui doit donc être enregistrée séparément des autres. Ce type de connaissance peut être rapproché de l'affectation d'un **genre** aux segments de document. La connaissance sur le genre peut être attribuée au niveau d'un document entier (comédie, drame, film noir, journal TV ...), ou d'une séquence (générique, présentateur, reportage, magazine ...).

4.4.4 Structuration par sujet principal/secondaire

La structuration du document peut ne pas concerner la structure logique du document mais la structure des sujets traités par le document; ainsi les auteurs de [Hearst 93b] distinguent sujet principal et sujets secondaires. Le sujet principal constitue alors un contexte dans lequel le sujet secondaire est traité. Ce cas de figure est aussi très pertinent pour une application de recherche de séquences d'actualités télévisées : par exemple, lors du sommet des sept pays les plus industrialisés à Paris en 1989, les réjouissances associées à ce sommet coïncidaient avec celles du bicentenaire de la révolution française. Un sommet des pays les plus pauvres a été organisé en parallèle, ce sommet ne prenant toute sa signification que dans le contexte du sommet du G7. Les séquences d'actualités télévisées qui relatent ces événements sont composées de ces différents contextes imbriqués.

Un autre exemple de ce type de structuration correspond à la réutilisation de plans d'archives dans une séquence d'actualités : alors que le sujet principal d'une séquence est, par exemple, l'élection d'un personnage politique à la tête d'un état, des rappels historiques sur la vie de ce personnage sont faits sous forme de plans d'archives réintégrés au cours du montage. Par exemple, lors de l'élection de Nelson Mandela à la tête de l'Afrique du Sud, les télévisions ont diffusé des séquences retraçant son parcours, qui le montrent tour à tour prisonnier politique, prix Nobel de la paix, puis président élu. La solution qui consiste à indexer un document avec des attributs SUJET PRINCIPAL et SUJET SECONDAIRE (solution adoptée par exemple dans [Hjelsvold 95]) ne nous semble pas satisfaisante. En effet, une séquence prend un nouveau sens dans chacun des documents où elle est réutilisée, et elle peut être réutilisée plus d'une fois. Dans ce cas précis, la séquence d'archive doit pouvoir conserver sa description initiale, tout en étant intégrée à la nouvelle séquence d'actualités où elle est réutilisée. Ce problème est donc celui du partage de composants communs dans des documents différents. Il pourrait être résolu par affectation d'un lien de composition entre le second document et la partie du premier qui est réutilisée (voir schéma 4.5).

En réponse à une requête, il peut être préférable de donner accès à la séquence d'origine,

plutôt que de soumettre au chercheur d'information tous les documents où la séquence a été réutilisée. Une indexation des segments réutilisés par indirection permet ainsi de ne pas prendre en compte ces réutilisations lors d'une recherche. Il doit être possible d'ajouter également des éléments de description à un segment réutilisé, par exemple pour signaler des faits qui n'étaient pas connus au départ. Dans un tel cas, est-il souhaitable de modifier les descriptions du segment dans son document d'origine ? À notre avis, ceci n'est pas souhaitable; la révision de descriptions est en effet un problème important et qui peut avoir des conséquences graves sur l'accès à l'information.

Pour résumer, le partage de descriptions d'un segment réutilisé est souhaitable, mais il doit être possible d'ajouter de nouvelles descriptions au segment dans son nouveau contexte. Par contre, le partage de composants images peut être exploité pour éviter un stockage redondant des segments utilisés plusieurs fois dans des documents différents.

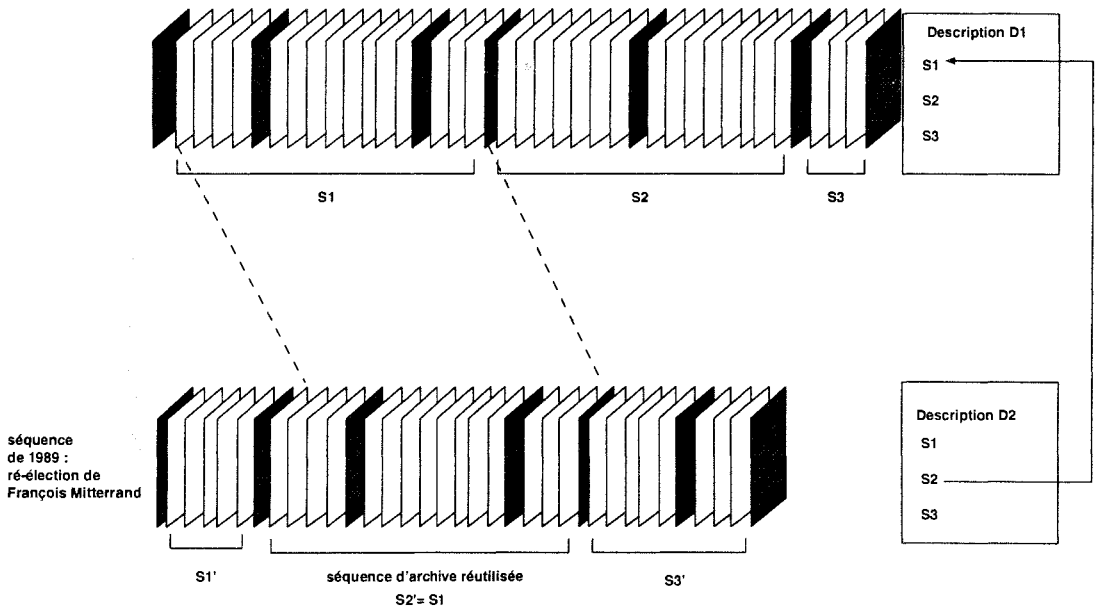


FIG. 4.5 – *Réutilisation d'une partie d'une séquence d'archive dans un document*

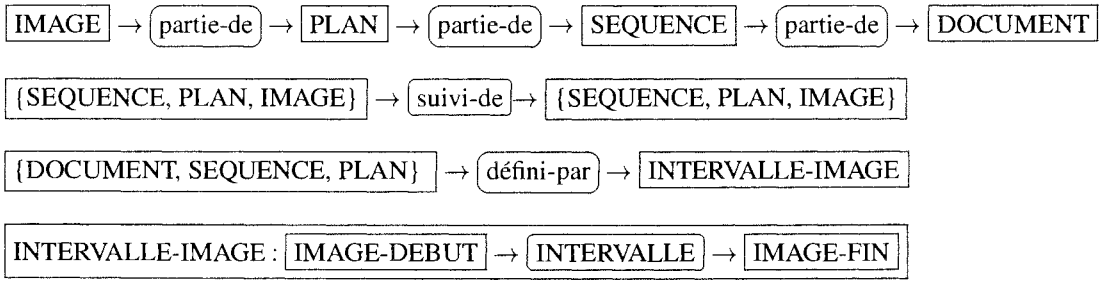
4.4.5 Conséquences sur le langage de requête

Quelles conséquences la structuration des documents peut-elle avoir sur le langage de requête ? Il semble souhaitable que les concepts qui caractérisent la structure logique du document fassent partie du vocabulaire de requête. Toutefois, dans le cadre d'un langage d'interrogation par mots-clés, la façon dont de tels concepts peuvent être utilisés n'est pas évidente. Ce problème ne se pose pas pour les langages de requête structurés. Par exemple, dans Maestro (Management Environment for Structured Text Retrieval and Organization) [Macleod 90], le langage de requête est une extension de SQL qui permet d'utiliser ces concepts de structure : une requête demandant les sous-sections de sections dont le titre comporte le mot "retrieval" et dont un des

paragraphes contient le mot "database" dans la première phrase, sera exprimée par

> List **gets** SousSection (**having any** Paragraphe **where** "database" **in first** Phrase)
> **of** Section **where** "retrieval" **in** Titre

Si l'intégration de critères de structure permet d'étendre le pouvoir d'expression du langage de requête, il ne faut pas en contre-partie que l'interface impose à l'utilisateur de spécifier des informations structurelles. En effet, les chercheurs d'information ne savent que très rarement quel est le type précis des documents qu'ils recherchent, et par conséquent quelle est leur structure. Ceci a été mis en évidence dans une étude résumée dans [Croft 90b]. Par contre, ils sont en général capables de spécifier certains attributs précis. Le fait de pouvoir représenter la description de la structure des documents dans le même formalisme que les descriptions du contenu des documents permet d'envisager d'utiliser les mêmes fonctions d'accès à ces divers types d'informations. Ceci justifie une fois de plus le choix d'un formalisme permettant de représenter des relations entre les différents aspects du modèle : contenu, structure, etc. Dans le cas de documents vidéo, nous pouvons représenter ces relations par :



Les concepts non relationnels qui permettent de définir la structure logique d'un document sont donc {IMAGE, PLAN, SEQUENCE, DOCUMENT}. Ces concepts forment l'ensemble $\mathcal{C}_{sl}$ des concepts de structure logique. Les concepts relationnels liés à la structuration logique sont $\mathcal{R}_{sl}$ = {suivi-de, partie-de}. La relation $suivi - de$ n'est pas une relation d'ordre total sur le contenu des images : une image ou une suite d'images peut être réutilisée à différents moments dans un film, par exemple lorsqu'il s'agit de représenter des faits passés qui reviennent à la mémoire d'un personnage de façon obsédante. Toutefois, si nous considérons les numéros des images qui indiquent la façon dont elles sont montées dans un document, la relation $suivi - de$ devient une relation d'ordre total. En effet, dans l'exemple précédent, les mêmes images qui reviennent à des instants différents auront des numéros différents dans le document. La relation $partie - de$ est une relation d'ordre partiel sur les concepts de structure logique.

Les concepts non relationnels qui permettent de définir la structure physique du document sont : {INTERVALLE-IMAGE, IMAGE-DEBUT, IMAGE-FIN}. La relation $intervalle$ est une relation d'ordre partiel définie par :

1. $\forall i \in NumImage, intervalle(i, i)$, qui dénote un intervalle comprenant une seule image.
2. $\forall i, j \in NumImage,\ intervalle(i, j) \Rightarrow NON\ \ intervalle(j, i)$
   L'ordre sur les numéros d'images est impératif pour le stockage et la description. Le fait

de pouvoir regarder une séquence vidéo en marche arrière n'est lié qu'à des aspects de présentation du document. Les images i et j participent en effet avec des rôles différents (début ou fin) à la définition d'un intervalle.

3. $\forall i, j, k \in NumImage$, $intervalle(i, j)$ et $intervalle(j, k)$ implique $intervalle(i, k)$; la composition de deux intervalles d'images contigus donne un nouvel intervalle.

Enfin, il existe une relation *défini-par* entre les concepts de structure logique et ceux de structure physique qui permet de faire la correspondance entre eux.

Permettre d'intégrer tous ces concepts à la formulation de la requête nécessite l'élaboration d'un langage de requête particulier, et d'une algèbre exploitant la structure temporelle des documents vidéo. Cet aspect du problème mérite d'être analysé plus finement, en s'appuyant sur les travaux déjà réalisés dans ce domaine (par exemple [Duda 95], [Adiba 95]).

4.4.6 Nouveau type de besoin d'information

La prise en compte de documents très longs dans une collection amène à décomposer les demandes exploratoires caractérisées par M. Cluzeau-Ciry dans sa typologie de demandes [Cluzeau-Ciry 88] en deux sous-types : exploration d'une collection et exploration d'un document volumineux. En effet, le deuxième type de demande exploratoire n'existait pas pour les images fixes. Nous remarquons que ce type de besoin peut également être pertinent pour des images fixes, par exemple pour traiter des paquets d'images (photographies d'un même reportage), ou des images très détaillées dont les détails doivent être examinés avec soin (images radiographiques par exemple, ou dessins comportant de nombreux détails).

4.4.7 Niveaux de structure traités dans les différentes phases du processus

Il nous faut déterminer quels sont les niveaux de structure qui sont pris en compte et la façon dont ils le sont lors des différentes phases d'une recherche.

Dans le modèle paramétré, la phase de **présélection** permet de déterminer un sous-ensemble de documents pertinents. Étant donné les quatre niveaux de granularité définis dans un document (document global, scène, plan, image), nous avons, pour un document D, quatre ensembles de descripteurs :

- l'ensemble des descripteurs qui indexent une image du document D, $T_{descr}(image(D))$,
- l'ensemble des descripteurs qui indexent un plan, $T_{descr}(plan(D))$,
- l'ensemble des descripteurs qui indexent une scène, $T_{descr}(scene(D))$,
- l'ensemble des descripteurs qui indexent le document entier, $T_{descr}(document(D))$.

Une présélection large rapporterait tous les documents indexés à un niveau quelconque par au moins un des concepts présents dans la requête :

$$T_{descr}(document(D)) \cup T_{descr}(scene(D)) \cup T_{descr}(plan(D)) \cup T_{descr}(image(D)).$$

Cela revient à ramener tous les descripteurs au niveau le plus général : le document. Ce type de présélection peut être utile pour une demande exploratoire où l'utilisateur ne recherche pas une portion précise du document, mais veut simplement se faire une idée de son contenu.

La présélection peut être plus sévère si elle rapporte les séquences indexées par un des concepts présents dans

$$T_{descr}(document(D)) \cap (T_{descr}(scene(D)) \cup T_{descr}(plan(D)) \cup T_{descr}(image(D))).$$

Dans ce cas, on ne retient du document que quelques séquences susceptibles d'être pertinentes. Ceci revient à ramener tous les descripteurs des différentes parties du document au niveau séquence ou scène, que nous avons choisi comme le mieux adapté à la présentation des résultats (§1.2.4).

Il peut être souhaitable de considérer les niveaux de structure auxquels sont attachés les descripteurs pour déterminer le type de la présélection. Une présélection large ferait l'union des documents décrits par tous les concepts (quel que soit le niveau de structure auquel ils sont attachés); une interprétation médiane ferait l'union des ensembles de documents rapportés par les concepts attachés à un même niveau de structure, et l'intersection des sous-ensembles de documents obtenus à chaque niveau de structure. Par exemple, la requête :

> «*Extrait d'actualités télévisées de 1989 montrant des chefs d'état buvant du champagne*»

traduite par

Facette	Concept	Niveau de structure
Genre	`actualités télévisées`	document
Signalétique	`1989`	document
Contenu	`chef d'état, champagne`	plan

pourra donner lieu à la sélection de documents de genre `actualités télévisées` ou produits en `1989`, et où l'on voit des `chefs d'état` ou du `champagne` :

Doc(`actualités TV`) ∪ Doc(`1989`)) ∩ (Doc(`chef d'état`) ∪ Doc(`champagne`).

Une présélection plus stricte peut faire l'intersection des documents ramenés par des facettes différentes de même niveau, ce qui nous donne en somme un ensemble de parties de documents, le type de partie dont il s'agit correspondant au niveau le plus fin de structure indexé par un des concepts de la requête :

Doc(`actualités TV`) ∩ Doc(`1989`)) ∩ (Doc(`chef d'état`) ∪ Doc(`champagne`).

Dans l'exemple ci-dessus, ce sont des plans qui sont présélectionnés. Ce type de présélection peut être utile si l'utilisateur sait exactement quel est le niveau de structure de document qu'il souhaite visualiser.

Dans le contexte de la recherche d'information structurée, que deviennent les mesures d'exhaustivité et de spécificité du modèle logique ? La **mesure d'exhaustivité** cherche à déterminer le document ou la partie de document qui répond de façon exhaustive à la requête. Un document vidéo peut répondre de façon exhaustive à une requête soit à cause d'une seule de ses séquences, soit à cause de plusieurs séquences complémentaires. Une scène répond exhaustivement à la requête si elle fait partie d'un document répondant aux critères généraux de la requête, et répond elle-même de façon exhaustive aux critères plus spécifiques de la requête. Cette mesure doit donc être calculée au niveau document et au niveau séquence. La **mesure de spécificité** cherche à déterminer à quel point le document ou une de ses parties ne traite que des concepts de la requête. Pour un document long, la mesure de spécificité du document n'a de sens que dans deux cas :

- soit la requête comporte de très nombreux concepts
- soit de nombreux termes de la requête sont relatifs à l'indexation du document global (signalétique, genre du document, ...).

En effet, comme l'ont constaté Salton [Salton 93] et Kheirbek [Kheirbek 95], il faudrait une requête très longue pour qu'un très long document soit vraiment spécifique vis à vis de la requête, si les concepts qu'elle comprend sont des descripteurs des niveaux de structure les plus fins du document.

Le traitement par facette permet d'avoir une autre vision de ces mesures. En effet, une facette est souvent liée à un niveau de structure du document : la facette SIGNALÉTIQUE est liée au niveau document global, les facettes CONTENU et MORPHOLOGIE au niveau plan, la facette CONNOTATION au niveau séquence. Si la mise en correspondance est faite par facette, les mesures d'exhaustivité et de spécificité ont un sens à chacun des niveaux de structure auxquels elles sont calculées. Il est possible de calculer l'exhaustivité de chaque niveau d'un document en ne tenant compte que des descripteurs attachés à ce niveau :

$$mexh_1(u_i) = \frac{\sum_{j \in Tdesc_{u_i}} f_1(d_j, q_j)}{\sum_{j \in Tdesc_{u_i}} f_2(q_j)} \tag{4.4}$$

avec par exemple $f_1(d_j, q_j) = d_j * q_j$ et $f_2(q_j) = q_j^2$. De façon symétrique :

$$mspe_1(u_i) = \frac{\sum_{j \in Tdesc_{u_i}} f_1(d_j, q_j)}{\sum_{j \in Tdesc_{u_i}} f_2(d_j)} \tag{4.5}$$

Ces mesures seront retenues pour les facettes qui correspondent à un niveau unique de structure du document.

Toutefois, il peut exister des facettes qui ne sont pas liées à un niveau de structure particulier du document, comme le GENRE, ou la CONNOTATION. Si des facettes correspondent à plusieurs niveaux de structure, il faut trouver le niveau de structure adéquat pour traiter la facette. Dans ce cas, il est souhaitable de ramener tous les descripteurs à un même niveau pour pouvoir

les examiner. Le critère pris en compte peut être alors la taille de la portion de document qui est indexée par un concept. Ceci nous amène à définir une deuxième mesure d'exhaustivité :

$$mexh_2(u_i) = \frac{1}{2}\left(\frac{\sum_{u_j<u_i} mexh_1(u_j) \times taille(u_j)}{\sum_{u_j<u_i} taille(u_j)} + \frac{\sum_{u_k \geq u_i} mexh_1(u_k)}{|u_k|}\right) \tag{4.6}$$

où

$$\begin{cases} u_j < u_i \text{ signifie que l'unité d'indexation } u_j \text{ est plus fine que } u_i, \\ \text{avec (image} < \text{plan} < \text{scène} < \text{document)}, \\ taille(u_j) \text{ est la taille de l'unité } u_j, \text{ en nombre d'images pour la vidéo}, \\ |u_k| \text{ est le nombre d'unités de structure considérées} \end{cases}$$

À noter que cette mesure peut être rapprochée de celle utilisée dans IOTA [Defude 86] pour calculer la représentativité d'un terme vis à vis d'un document (considéré comme des unités imbriquées) : chaque unité «hérite» des poids d'indexation des unités qui la composent au prorata de leur taille. La normalisation par la taille des unités permet d'éviter qu'une unité comportant de nombreuses parties peu pertinentes soit classée avant une unité contenant une seule partie très pertinente.

De façon symétrique, la spécificité peut être calculée, pour tenir compte des niveaux de structure imbriqués, par :

$$mspe_2(u_i) = \frac{1}{2}\left(\frac{\sum_{u_j<u_i} mspe_1(u_j) \times taille(u_j)}{\sum_{u_j<u_i} taille(u_j)} + \frac{\sum_{u_k \geq u_i} mspe_1(u_k)}{|u_k|}\right). \tag{4.7}$$

En fait, la **synthèse** dépend en grande partie de la façon dont les résultats vont être présentés. Trois sortes principales de présentation peuvent être faites pour des documents structurés :

1. Présentation par document : on veut classer les différents documents entre eux, et pour chaque document les séquences qui le composent par ordre de pertinence décroissante. Dans ce cas, il faut calculer les mesures d'exhaustivité et éventuellement de spécificité au niveau du document (D), mais en tenant compte des niveaux inférieurs du document : $mexh_2(D)$, $mspe_2(D)$. Ces mesures peuvent être ou non combinées.

   Pour classer les séquences d'un même document, il faut calculer leur exhaustivité et leur spécificité. Pour chaque séquence seq du document, on calcule : $mexh_2(seq)$, $mspe_2(seq)$ en les combinant éventuellement en fonction du type de besoin.

2. Il n'existe pas de niveau de structure fixé pour la présentation, la mise en correspondance doit le déterminer. Dans ce cas, on mesure l'exhaustivité, puis la spécificité à chaque niveau de la structure du document et l'on présente le niveau de structure qui maximise ces mesures.

3. C'est le niveau scène qui est retenu prioritairement pour la présentation. Ici, nous avons deux sous-cas :

   - Les scènes sont présentées de façon indépendante de leur appartenance à un document. Cette solution est valide lorsque le niveau document n'est pas important pour

l'application. Par exemple, pour une recherche de séquences d'actualités télévisées, il est peu important de présenter ensemble les reportages (niveau scène) diffusés dans un même journal (niveau document). Dans ce cas, il faut tenir compte, dans les mesures effectuées au niveau de la séquence non seulement des parties qu'elle comprend, mais aussi des parties qui l'englobent : $mexh_2(seq)$, $mspe_2(seq)$.

- Seule la meilleure scène de chaque document est retenue. Il faut alors évaluer la pertinence globale de chaque document de façon récursive, en normalisant par la taille. Puis pour chaque document, rechercher la séquence la meilleure. Ceci correspond à la stratégie globale/locale de Salton. Si la séquence choisie dans le document doit tenir compte de ses séquences voisines, alors

$$\begin{aligned} mexh(scene_i(D)) &= mexh_2(D) \\ & et \\ mspe(scene_i(D)) &= mspe_2(D) \end{aligned}$$

Sinon (on ne tient pas compte des autres séquences présentes dans le document), le niveau document n'intervient que pour les descripteurs attachés à ce niveau :

$$\begin{aligned} mexh(scene_i(D)) &= mexh_2(scene_i(D)) \\ & et \\ mspe(scene_i(D)) &= mspe_2(scene_i(D)). \end{aligned}$$

Enfin, dans un contexte de documents structurés, que faire du **critère de censure**, qui détermine le nombre de documents que l'utilisateur peut visualiser ? Ce critère concerne le nombre de séquences qui seront présentées à l'utilisateur, mais

- soit seule la séquence la plus pertinente est retenue pour chaque document (le critère de censure correspond alors également au nombre de documents qui seront présentés); dans ce cas, toutes les séquences doivent d'abord être classées entre elles, puis n'est retenue que la séquence la plus pertinente de chaque document. Ceci peut répondre à une demande d'exploration d'une collection par exemple, ou à une demande thématique. Dans ce cas, la mesure d'exhaustivité du document est réalisée par $mexh_2(D)$, et on choisit la séquence la plus spécifique, c'est à dire celle qui maximise : $mspe_2(sequence)$.

- soit plusieurs séquences d'un même document sont sélectionnées ensemble; dans ce cas, les séquences les plus pertinentes des documents les plus pertinents sont présentées. Ce cas pose à nouveau deux problèmes :

  1. Il faut déterminer le **nombre de séquences qui sera présenté par document**, par exemple en fixant un seuil de pertinence en-deçà duquel les séquences ne seront pas présentées, afin d'éviter de montrer les séquences les moins pertinentes du document le plus pertinent (au détriment de séquences plus pertinentes dans des documents globalement moins pertinents). Ce seuil de pertinence peut correspondre à la valeur obtenue par la $n$ième séquence (n étant le critère de censure) dans un classement global sur les séquences, tous documents confondus (figure 4.6). Pour

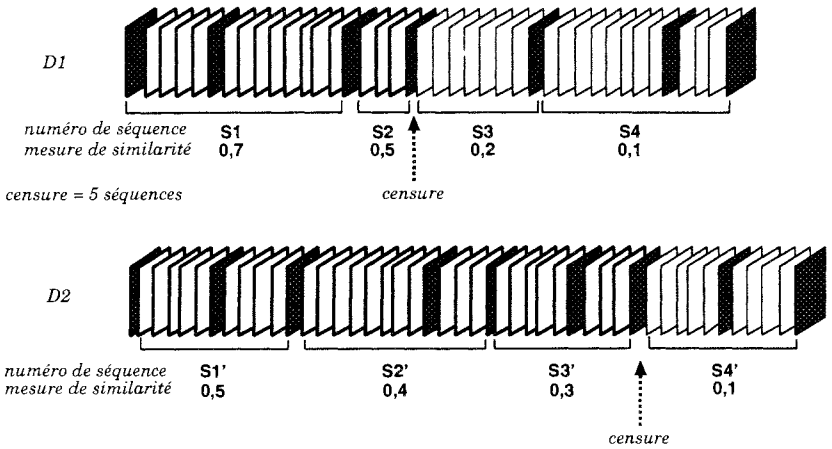


FIG. 4.6 – *Censure tenant compte du degré de similarité des séquences.*

ce classement global par séquences, on retient les mesures $mexh_2(sequence)$ et $mspe_2(sequence)$. Dans le cas où l'utilisateur veut explorer un document long, le critère de censure ne sera pas utilisé. Dans le cas d'un besoin précis, il sera nécessaire d'établir une censure pour que les séquences les plus pertinentes de différents documents puissent être présentées à un rang raisonnable.

2. Il est préférable de respecter pour la présentation des séquences d'un même document l'ordre dans lequel elles se trouvent dans ce document. Dans ce cas, il faut malgré tout indiquer à l'utilisateur quelles sont les séquences les plus pertinentes. Un moyen simple de résoudre ce problème est de proposer les séquences dans l'ordre dans lequel elles arrivent dans le document, mais de représenter par un signe visuel leur degré de pertinence, par exemple, une barre d'autant plus colorée ou foncée que la séquence est pertinente (figure 4.7).

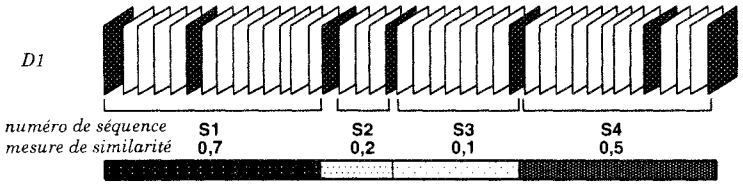


FIG. 4.7 – *Représentation de la pertinence des séquences.*

Si l'utilisateur souhaite une structuration à la demande des séquences qu'il veut visualiser, le problème est plus compliqué : il faudra définir quels sont les intervalles d'images les plus pertinents dans le document, et regrouper les intervalles contigus en fonction de leurs valeurs de similarité. Il est alors nécessaire d'établir une algèbre sur les intervalles d'images, cet algèbre prenant en compte le chevauchement possible des intervalles. Des propositions ont déjà été faites dans ce domaine, en particulier dans [Oomoto 93, Duda 95].

Nous résumons ci-après l'impact de la structuration des documents sur chaque phase du processus de recherche et les choix par défaut effectués par le système.

Phase du processus	Type de demande de l'utilisateur			
	Besoin exploratoire		Besoin thématique	Besoin précis
	d'une collection	d'un document long		
présélection	séquences	1 document	séquences	séquences
mise en correspondance	$mexh_2(doc)$ $mexh_2$(scène)	$mexh_2(doc)$ $mspe_2$(scène)	$mexh_2$(scène) $mspe_2$(scène)	$mexh_2$(scène) $mspe_2$(scène)
critère de censure	n séquences	∞	n séquences	n séquences
présentation	1 séquence par document	toutes les séquences pertinentes du document	1 séquence par document	séquences les plus pertinentes par document

4.5 Résumé

Le modèle paramétré de processus de recherche interactive résulte d'une analyse profonde d'un processus de RI et des différentes phases qu'il comporte. En nous appuyant sur ce modèle, nous avons montré comment ses primitives pouvaient être interprétées dans le formalisme des graphes conceptuels. Ceci nous a amené à définir une primitive de reformulation pour ce formalisme, proposition tout à fait nouvelle à notre connaissance dans le domaine de la RI. Nous avons également tenu compte du nouveau contexte distribué qui s'impose de plus en plus pour les systèmes de recherche d'information.

Toutefois, en essayant d'appliquer ce modèle à la vidéo, nous nous sommes trouvée confrontée à un certain nombre de difficultés, liées principalement à la nature composite de ces documents complexes. Nous avons donc dû adapter ce modèle pour la recherche dans des collections de documents structurés. La motivation principale de la RI dans des documents structurés est l'allégement de la charge cognitive de l'utilisateur : un SRI doit lui proposer en priorité la partie de document qui l'intéresse et non un document global et volumineux en réponse à une requête. Cette approche est très pertinente pour des documents multimédias, notamment pour les images animées. D'ailleurs, les documents textuels peuvent être considérés comme des séquences d'images au sens large, lorsque le texte est segmenté en régions par des algorithmes de reconnaissance de formes [Rus 94]. Les documents de la Bibliothèque Nationale de France n'ont-ils pas été numérisés sous forme d'images ?

Cette analyse de divers travaux prenant en compte la structure logique et physique des documents fait ressortir plusieurs points principaux :

- la structuration ne doit pas être imposée à l'utilisateur qui doit rester libre de l'ignorer s'il le souhaite ; c'est pourquoi une structuration *par défaut* doit être disponible : si aucun critère dans la requête de l'utilisateur ne permet de déterminer le niveau de structure pertinent, le système doit pouvoir en proposer un ;
- la recherche d'extraits pertinents se fait en général par une stratégie de recherche globale/locale. La recherche globale présélectionne les documents susceptibles d'être pertinents parmi la collection entière, alors que la recherche locale permet de trouver le passage ou l'extrait précis des documents pertinent pour la requête. Pour l'indexation, la stratégie inverse est utilisée : les parties les plus fines de la structure sont d'abord indexées, et les résultats sont « remontés » récursivement aux niveaux supérieurs de la structure.

Nous avons également montré que le traitement des documents par facette pouvait faciliter la prise en compte des documents structurés.

Ces propositions rendent le modèle de processus très complexe. Intuitivement, cela conforte notre conviction que la notion de système multi-agents est opportune pour modéliser un tel comportement. Nous allons donc étudier ce que cette notion peut apporter à la modélisation de l'architecture d'un SRI.

Chapitre 5

Les agents d'un Système de Recherche d'Information

Les SRI peuvent être réalisés sous forme de systèmes experts particuliers. Or, les systèmes experts des années 80 ont rapidement atteint leurs limites : lourds à mettre en place, ils étaient peu adaptables à de nouveaux contextes. Nous avons pu vérifier la difficulté que représente l'adaptation d'un système expert à un nouveau contexte au début de ce travail de thèse, lorsque nous avons dû implanter une nouvelle application pour le prototype RIVAGE de recherche d'images. La nouvelle application concernait du télé-achat, et devait permettre à un utilisateur de consulter un catalogue de photographies de produits, après qu'il ait formulé une requête sous forme de mots-clés choisis dans un thésaurus[16]*. Même si le prototype RIVAGE avait été développé dans un environnement orienté-objet, la prise en compte de nouveaux types de données n'était pas aisée : le comportement adopté pour traiter une collection d'œuvres de photographes célèbres du début du XXième siècle n'était pas adapté à l'application plus pragmatique que nous devions réaliser. Par ailleurs, les relations entre les différentes parties du processus étaient figées et implicites.*

Dans les chapitres précédents, nous avons fait apparaître la diversité des connaissances que peut exploiter un système de recherche d'information (SRI), ainsi que les différents aspects du comportement qu'il peut adopter. Intuitivement, vu la complexité du problème à traiter, nous pensons que le concept d'agent et de système multi-agent peut nous aider d'une part à définir une architecture flexible et adaptative pour le SRI, et d'autre part à étudier les relations qui existent entre les différentes entités au sein du système.

Les premiers chercheurs en intelligence artificielle distribuée sont partis de l'observation des systèmes naturels : face à leurs limites individuelles, ces systèmes ont tendance à s'unir pour réaliser collectivement leurs buts. Cette observation a fait naître l'idée d'éclater les systèmes complexes en une collection d'experts distribués. La problématique de l'intelligence artificielle distribuée consiste à définir initialement une tâche globale, et à concevoir des entités distribuées qui permettent de la réaliser [Demazeau 89]. L'étude de la distribution et de la résolution collaborative est donc au cœur de cette problématique, ce qui concorde avec notre approche

16. Cette application a été réalisée dans le cadre d'un contrat avec le CCETT.

d'un système de recherche d'information.

La conception orientée objet a mis l'accent sur la modularité et l'encapsulation des données, mais elle ne permet pas d'avoir facilement une idée globale du comportement dynamique d'un système, ni des dépendances qui peuvent exister entre les modules. Par exemple, OMT (Object Modeling Technique) [Rumbaugh 91] comporte

- *un modèle objet qui décrit la structure statique des différents objets du système et les relations entre les classes d'objets,*
- *un modèle dynamique qui décrit les aspects du système qui évoluent avec le temps sous la forme d'un diagramme d'états représentant les transitions provoquées par des événements, et*
- *un modèle fonctionnel qui décrit les transformations des données dans le système sous forme de diagrammes de flux de données entre les différents processus.*

Toutefois, les paradigmes liés à la dynamique globale du système ne sont que peu ou pas développés, et n'ont qu'un faible pouvoir d'expression. La «conception orientée agent», même si cette appellation est encore ambitieuse du fait de l'absence d'une vraie méthodologie de conception dans ce domaine, permet de doter chaque objet d'une certaine autonomie d'action afin qu'il ait un comportement moins rigide et puisse évoluer au sein d'un système.

Dans ce chapitre, nous allons présenter les principaux concepts des systèmes multi-agents (SMA) et nous appuyer sur ces concepts pour définir les agents d'un SRI.

5.1 Les Agents : principaux concepts

5.1.1 Qu'est ce qu'un agent ?

Minsky a défini le concept d'**agent** de manière informelle et très vague :

> *Toute partie ou tout processus de l'esprit assez simple à comprendre – même si les interactions entre des groupes de ces agents peuvent produire des phénomènes beaucoup plus difficiles à comprendre* [Minsky 88].

Cette définition ne nous permet pas de spécifier réellement ce qu'est un agent, et ce qui ne l'est pas dans un système. Marvin Minsky rapproche le concept d'agent de celui de «boîte noire» : il permet de faire abstraction d'un certain nombre de détails liés au fonctionnement interne, comme une machine qui accomplit une tâche sans que l'on ait besoin de savoir comment. Pour nous, le concept de «boîte noire» peut déjà être appliqué au concept d'objet, qui permet d'encapsuler données et comportement ; il n'est donc pas propre au concept d'agent. Par ailleurs, il est souvent nécessaire d'avoir un peu plus d'information sur ce que fait un agent, ne serait-ce que pour décider qu'il intervienne ou non à un moment de la résolution de problème. Par contre,

la notion d'interaction présente dans la définition de Minsky est au cœur de la problématique multi-agents.

En fait, le concept d'agent est encore assez mal défini, et en donner une définition relève du pari. Une des premières définitions sur laquelle notre travail s'est appuyé est celle de Jacques Ferber :

> un agent est «*une entité physique ou abstraite qui est capable d'agir sur elle-même et son environnement, qui dispose d'une représentation partielle de cet environnement, qui, dans un univers multi-agents, peut communiquer avec d'autres agents, et dont le comportement est la conséquence de ses observations, de sa connaissance et des interactions avec les autres agents.*» [Ferber 88]

Cette définition fait apparaître les principales propriétés d'un agent :

- l'action,
- la réflexivité,
- la partition des connaissances,
- des capacités de perception,
- la communication avec l'environnement et les autres agents,
- l'interactivité.

Pour M. Wooldridge [Wooldridge 95], il existe une notion faible et une notion forte d'agent. La notion faible recouvre l'autonomie, l'aptitude sociale, la réactivité par rapport aux événements qui se produisent dans l'environnement, et la pro-activité, c'est-à-dire l'aptitude à prendre des initiatives. Cette catégorie d'agents comprend par exemple les *softbots*. Un softbot (software robot) [Genesereth 94] est un agent qui interagit avec un environnement logiciel en émettant des commandes destinées à modifier l'état de cet environnement ; il peut s'agir de commandes de déplacement de fichier, de copie, etc. Les capteurs du softbot sont des commandes qui donnent de l'information, par exemple la liste des fichiers d'un répertoire ou le nom du répertoire courant [Etzioni 94a, Etzioni 94b]. La notion forte d'agent de M. Wooldridge recouvre quelques propriétés supplémentaires : un agent a des connaissances, des croyances, des intentions et des obligations ; il peut même avoir des émotions. Enfin, quelquefois, un agent peut être mobile, comme les agents de Telescript (General Magic [White 94]) ; la mobilité de l'agent peut d'ailleurs être vue comme une forme d'autonomie : l'agent n'est pas lié à un lieu particulier, et il conserve ses capacités d'action lorsqu'il est déplacé.

Un agent est toujours supposé **rationnel**, c'est-à-dire qu'il est censé agir pour réaliser son but et non pour l'empêcher de se réaliser.

5.1.2 Propriétés d'un agent

L'action

En général, l'action est dirigée par des buts. Les actions d'un agent sont déterminées par son **plan**, c'est-à-dire la suite des comportements que l'agent adopte pour tenter de se rapprocher de ses buts. Le comportement d'un agent peut être complexe s'il consiste en une suite structurée de comportements élémentaires. La *planification* est l'activité d'un agent entre l'apparition du but à atteindre et l'exécution du plan. La conduite des actions ou *contrôle* consiste à exécuter le plan. Les actions peuvent également être ordonnées de façon dynamique en fonction d'un contexte, selon certaines priorités. L'agent peut alors réviser ses plans. Un agent peut s'adapter à son environnement en réagissant aux événements - c'est le domaine de la planification réactive. Il n'est pas seulement capable de réagir mais peut aussi prendre des initiatives, anticiper pour satisfaire son but ; enfin, il doit être capable de *rétro-agir*, c'est-à-dire de modifier son comportement en fonction des retours qu'il perçoit dans son environnement.

L'autonomie

Une définition «faible» du concept d'autonomie recouvre la capacité à se régir d'après ses propres lois. Une définition plus forte indique la capacité à se doter de ses propres lois [17].

Pour un agent individuel, l'autonomie implique qu'il puisse disposer de connaissances et de modes d'inférence qui lui sont propres : son existence n'est pas justifiée par celle d'autres agents. Toutefois, un agent est toujours situé dans un système, il ne peut ignorer le monde qui l'entoure : un agent autonome n'est pas pour autant un agent autiste. Cette notion faible d'autonomie recouvre en fait seulement l'auto-suffisance. La véritable autonomie implique qu'il puisse prendre des décisions en fonction de l'évolution du contexte dans lequel il se trouve. Castelfranchi [Castelfranchi 95] distingue l'**autonomie de moyens** et l'**autonomie de buts**. L'autonomie de moyens correspond par exemple à celle d'un robot; elle implique des capacités de décisions, mais uniquement pour choisir de mettre en œuvre un type de moyen plutôt qu'un autre. L'autonomie de buts d'un agent ne l'empêche pas de recevoir de l'extérieur des ordres, des demandes, des normes et ces stimulations reçues de l'extérieur peuvent modifier son comportement, mais elles ne doivent pas le déterminer : un agent peut refuser d'effectuer l'action qu'on lui demande de faire si elle ne concorde pas avec son propre but, ou il peut décider de son propre chef de faire quelquechose qui ne lui a pas été demandé. Son comportement n'est pas un ensemble de réflexes rigides et déterministes à des stimuli externes.

Nous avons dit précédemment que la mobilité d'un agent pouvait être considérée comme une manifestation d'autonomie. En effet, un agent autonome ne doit pas dépendre exclusivement d'un environnement donné pour pouvoir exister. Il s'agit toutefois d'une autonomie du point de vue de l'environnement physique, pas forcément d'une autonomie sociale : un agent peut être contraint à se déplacer à la demande d'un autre agent.

Pour un système, l'autonomie recouvre sa capacité à s'auto-organiser en fonction d'un

17. *Encyclopedia Universalis*, 1989.

contexte. Se situant dans l'approche systémique, K.E. Boulding et J.L. Le Moigne ont proposé d'évaluer l'autonomie selon plusieurs degrés incrémentaux [Lapierre 92] :

- L'*autoconservation passive* est le degré le plus faible d'autonomie, car un système fermé est autonome par rapport à son environnement ; il ne peut être désorganisé par les contraintes ou perturbations extérieures qu'il ne laisse pas entrer.
- L'*autoconservation active* concerne les systèmes qui répètent les mêmes mouvements tant qu'une énergie est fournie, une entité extérieure permettant éventuellement le réglage. Un exemple de ce type de système est le chauffage central régulé par un thermostat.
- L'*autorégulation* permet à un système de corriger ses erreurs. Ceci nécessite d'une part une phase d'évaluation à partir d'un retour de l'environnement sur les actions du système, d'autre part la capacité de réagir en modifiant son comportement et éventuellement ses buts. Ces deux conditions constituent une *boucle de rétroaction* (feedback), qui peut provoquer des réactions en chaîne. La rétroaction est *amplifiante* si la variable qui revient en entrée varie continuellement dans le même sens : c'est une *variation cumulative* si elle est positive, ou *récessive* si elle est négative. La rétroaction est *compensatrice* si la réaction de l'environnement fait passer la variable d'une valeur à sa valeur contraire.
- l'*autoreproduction* est considérée comme le degré supérieur de l'autonomie : les entités peuvent choisir de se reproduire, en se modifiant ou non. En informatique, on pense bien sûr aux algorithmes génétiques, mais aussi aux travaux de Gasser dont nous reparlerons au §5.7.5.

Les connaissances d'un agent

Un agent possède un certain nombre de connaissances sur lui-même, sur les autres agents avec qui il collabore, et sur son environnement, mais ses connaissances sont partielles. Il peut disposer de capteurs qui lui transmettent dynamiquement des informations sur l'évolution du monde qui l'entoure. Si tous les agents ont le même modèle, un certain nombre de connaissances sont implicites. Les connaissances dont dispose un agent sur lui-même lui permettent de raisonner sur ses propres capacités : en s'appliquant à lui-même ses capacités de raisonnement, l'agent devient réflexif[18].

Les connaissances d'un agent qui sont établies sont appelées **faits**. Pour un système de recherche d'information, les descriptions de documents qui concernent la signalétique, le contenu, la morphologie du document sont considérées comme des faits. Les faits connus du système peuvent être contradictoires. Par exemple, les descriptions du contenu de l'œuvre de René Magritte (*«Ceci n'est pas une pomme»*) seraient pour le média image `pomme`, et pour le média texte `NON pomme` (en supposant que nous pouvons avoir des descriptions négatives pour les documents). Les connaissances enregistrées dans le thésaurus peuvent également être considérées comme des faits, tout au moins pour les liens *sorte-de*, *partie-tout* et *employé-pour* entre les termes.

18. La réflexivité peut être souhaitable pour un système en ce qu'elle permet d'éviter de devoir chercher un nouveau mode de raisonnement pour les méta-connaissances du système [Pitrat 90].

Par contre, les descriptions relatives à la connotation, lorsqu'elles sont constituées de mots-clés qui reflètent l'impression produite sur le lecteur, peuvent être considérées comme des croyances car elles sont hautement subjectives. La dénomination de **croyances** est réservée aux connaissances qui sont révisables, car liées à des hypothèses [Shoham 1989]. Ces descriptions seraient donc de la forme : $Believe(Indexeur\ \ p)$ (c'est-à-dire : l'indexeur croit p) où p serait de la forme [Document] $\rightarrow$ (décrit par) $\rightarrow$ [CONNOTATION : terme-connotatif]. Pour des collections d'œuvres d'art, si des critiques sont enregistrées sous forme de documents particuliers, des concepts connotatifs pourraient être extraits de ces critiques pour indexer le document; ces descriptions seraient de la forme $Believe(Auteur\ \ p)$.

Deux stratégies permettent de les remettre en cause [Galliers 89] :

- soit les justifications sur lesquelles reposent les croyances changent,
- soit les croyances persistent jusqu'à ce qu'elles soient récusées (c'est la théorie de la cohérence).

Les croyances peuvent être de premier ordre si elles s'appliquent à des faits, ou de second ordre si elles s'appliquent aux croyances propres de l'agent et à celles des autres [Wooldridge 95]. Ainsi, la logique du 1er ordre ne convient pas pour modéliser les croyances, car la valeur sémantique d'une expression de croyance ne dépend pas seulement de la valeur de vérité des sous-expressions qui la composent. Si p est une proposition, la valeur de «*Dupond croit* p» ne dépend pas seulement de la valeur de p. En général, on utilise la logique modale [Kripke 63] et la sémantique des mondes possibles pour modéliser les croyances [Hintikka 62]. Diverses représentations alternatives ont été proposées, comme dans la théorie de la croyance et de la conscience de Levesque [Levesque 84] qui emprunte à la *théorie des situations* [Barwise 83], ou encore le modèle de déduction de Konolige [Konolige 86] où les croyances sont représentées symboliquement sous forme de règles, de schémas (frames), de réseaux sémantiques ou de formules logiques. La description de la connotation d'un document par mot-clé reflète les croyances de l'indexeur au moment où il décrit les documents. Si la connotation du document est enregistrée sous forme de critiques ou de commentaires de personnes diverses sur les documents, les croyances seront très variées, et éventuellement peu cohérentes.

La **partition des connaissances** est une caractéristique des systèmes multi-agents : chaque agent ne dispose que d'une vue partielle sur les connaissances du système. Ceci est vrai pour les connaissances de notre SRI : chaque facette de description des documents peut être traitée par un agent de recherche indépendant. C'est également vrai pour le savoir-faire du système : différents agents de résolution de problème ont chacun des capacités de résolution partielles et doivent coopérer pour améliorer la résolution.

Diversité des modes de raisonnement d'un agent

Chaque agent au sein d'un système peut avoir son propre mode raisonnement. Dans le cas qui nous occupe, la recherche d'information multimédia, le système devrait pouvoir mettre en

œuvre :

- le raisonnement spatial, pour analyser les images et répondre à des requêtes contenant des critères spatiaux ; ces critères peuvent être absolus (par exemple, chercher les images dont le centre est occupé par un personnage), ou relatifs (par exemple, chercher des images où l'on voit une personne sur un pont) ;
- le raisonnement temporel, en particulier pour la vidéo ; par exemple, un utilisateur peut chercher des documents qui commencent avec un gros plan sur un personnage, suivi d'un zoom arrière ;
- le raisonnement structurel, c'est-à-dire sur les parties qui composent un objet ; par exemple l'utilisateur demande des séquences vidéo contenant un plan fixe ;
- le raisonnement fonctionnel qui repose sur la manière de fonctionner d'un objet, notamment pour le calcul de la similarité entre requêtes et documents,
- le raisonnement causal qui permet d'inférer des actions et provoque un changement d'état des objets suite à certains événements ; le raisonnement causal pourrait aider à expliquer aux utilisateurs les résultats du système d'une part, et notamment la reformulation de la requête suite aux choix et rejets de documents faits par l'utilisateur ;
- l'extraction de connaissances a priori d'un ensemble de faits et de règles qui génèrent des explications sur une situation,
- le raisonnement par cas où un nouveau problème est résolu par analogie avec des solutions à des problèmes déjà rencontrés, comme l'a proposé Malika Smaïl [Smaïl 93a].

Un système multi-agent peut faire cohabiter ces diverses formes de raisonnement, et les utiliser de façon opportune en fonction de la situation.

5.2 Systèmes multi-agents

5.2.1 Notion de système

La théorie des systèmes multi-agents s'inspire beaucoup de la *systémique*, qui étudie les systèmes pour eux-mêmes et non seulement en tant que collections d'individus ou d'entités individuelles. En effet, le système a une certaine structure sociale au sein de laquelle chaque agent prend sa place. Considéré comme faisant partie d'un système au sein duquel il évolue, un agent peut appartenir à plusieurs groupes sociaux, avec des rôles différents dans chacun des groupes : c'est le concept de «polynèmes» de Minsky, qui fait référence aux différentes activités qu'un agent peut mener en parallèle. Ce concept est anthropomorphique : un humain peut être à la fois salarié d'une entreprise, père ou mère de famille, posséder et conduire un véhicule, jouer au tennis durant ses loisirs, etc.

Un système multi-agents peut être défini comme «*un réseau faiblement couplé de modules de résolution de problème qui travaillent ensemble pour résoudre des problèmes qui sont au-*

delà de leurs capacités individuelles » [Durfee 89]. La distribution peut être une caractéristique «naturelle» de certaines applications, qu'elle soit physique et/ou fonctionnelle.

5.2.2 L'organisation du système

En général, les agents s'organisent autour d'un but commun à la réalisation duquel chacun va participer selon ses capacités. Il n'existe pas de structure optimale d'organisation : l'organisation des agents peut être statique ou dynamique. Toutefois, il semble exister une corrélation positive entre complexité croissante du problème à résoudre et dynamicité de l'organisation [Chevrier 93].

Un système multi-agents est cohérent si les actions de chaque agent ont un sens vis-à-vis des buts communs du groupe d'agents tout entier. La cohérence de l'organisation peut être réalisée par un certain nombre de moyens, en particulier par l'affectation de rôles aux agents, la planification pour ajuster l'activité des agents, la gestion des ressources, l'utilisation de méta-informations, et la gestion des communications [Aitken 94] Le concept de communication est particulièrement important pour les SMA, car c'est en particulier grâce à la communication que les agents peuvent acquérir de la connaissance et se concerter pour améliorer la résolution du problème qui les occupe.

5.3 La communication

5.3.1 Moyens de communication

Les deux principaux paradigmes utilisés pour la communication sont la *mémoire partagée* et le *partage de messages*, ces deux paradigmes pouvant être utilisés de façon complémentaire au sein d'un système.

Tableaux noirs

Lorsque la mémoire est partagée, le modèle le plus courant est celui des tableaux noirs. Un **tableau noir** est une mémoire globale à laquelle différents experts ou *sources de connaissances* peuvent accéder, soit pour y enregistrer des données ou des résultats partiels, soit pour y rechercher de l'information [Hayes-Roth 85, Nii 86]. L'accès des sources de connaissance au tableau noir est contrôlé par un ordonnanceur qui sélectionne dans l'agenda la prochaine source à exécuter.

L'utilisation d'un tableau noir global est particulièrement appropriée lorsque les agents qui échangent de l'information sont mal identifiés : cela permet d'éviter de se poser la question de l'identification de l'émetteur et du récepteur de l'information. Toutefois, par son aspect centralisateur, il constitue un goulot d'étranglement dans les communications du système. C'est pourquoi un tableau noir est généralement structuré en plusieurs niveaux d'abstraction, chaque

niveau synthétisant les données du niveau inférieur. Cette structuration permet d'éviter qu'il ne soit parcouru par tous les agents à la recherche d'une information, ce qui n'est pas très efficace lorsque les récepteurs potentiels d'une information sont bien identifiés. Des zones peuvent être réservées à certains experts : les tableaux noirs peuvent ainsi représenter explicitement la partition de la connaissance du domaine. Dans certains systèmes, chaque agent a son propre tableau noir ; c'est le cas par exemple de DVMT [Lesser 83] où tous les agents ont la même architecture, ce qui leur permet de communiquer : chaque agent a le même modèle global de résolution de problème et connaît donc le modèle des autres.

SRI construits sur le modèle des tableaux noirs

La plupart de SRI multi-agents ont été construits sur le modèle des tableaux noirs. C'est le cas d'I^3R (Intelligent Interface for Information Retrieval) [Thompson 89], conçu pour rechercher des informations dans la collection des revues *Communications of the ACM*. Différents experts, chacun traitant un aspect différent du problème, peuvent accéder à un tableau noir global mais structuré ; ces experts ne se connaissent pas mutuellement, sauf le contrôleur et l'ordonnanceur. Chaque expert d'I^3R est implantée par un système à base de règles, ces règles étant de la forme <condition, action>.

Le système CODER (COmposite Document Expert/Extended/Effective Retrieval) [Fox 87, Fox 92, Belkin 87], qui traite d'une collection de messages électroniques (AIList digest) est construit, lui, autour de deux tableaux noirs : l'un est réservé aux fonctions de recherche et l'autre à l'analyse des documents. Chaque tableau noir est divisé en deux parties :

1° un ensemble de zones prioritaires où sont stockées les hypothèses en attente, les questions et les réponses, et auxquelles tous les experts peuvent accéder,

2° un ensemble de domaines dont chacun est réservé à un expert particulier : pour la recherche, il s'agit du modèle de l'utilisateur, des termes utiles, de la requête formulée par l'utilisateur, et des sorties planifiées; pour l'analyse des documents, il s'agit des différents niveaux de structure des documents (messages, blocs, attributs, mots-clés) et de références entre les documents.

Pour notre SRI, la requête, ses différentes évolutions, et les ensembles successifs de documents résultats peuvent être stockés dans un tableau noir structuré par facettes. Le problème que pose ce type de solution est qu'elle est peu adaptée à la distribution physique, sauf si le tableau noir est lié à l'interface utilisateur.

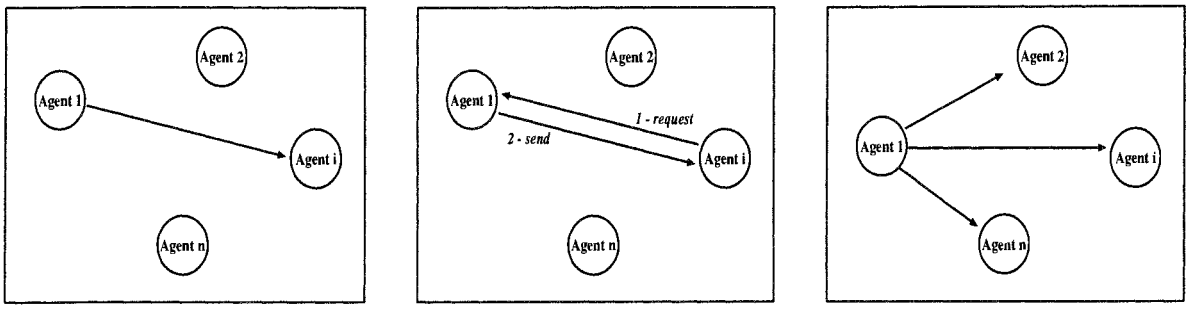
Communication par messages

L' **envoi de messages** provient du paradigme d'acteur. Un *acteur* [Hewitt 77, Agha 90] peut être vu comme un agent de calcul qui effectue des actions (décrites dans un script) en réponse à une sollicitation : la réception d'une communication provoque chez lui un événement auquel il réagit ou non, et qu'il peut propager vers d'autres acteurs. Le modèle acteur permet de concevoir des systèmes ouverts au sens de Hewitt, puisque les acteurs sont indépendants, concurrents et

asynchrones.

Le passage de messages est un moyen de communication simple et abstrait qui rend les agents plus indépendants : leur existence et leur fonctionnement n'est pas lié à une structure de données externe. Ce paradigme de communication nécessite un protocole [Cammarata 83] dont les principaux types sont

- la communication sélective : les messages sont adressés à un récepteur spécifique (figure 5.1.a).
- la communication à la demande : les messages ne sont envoyés qu'à la demande d'un agent (figure 5.1.b).
- la communication diffusée (broadcast) : les messages peuvent être lus par tous les agents. Ce type de communication est non sollicité (figure 5.1.c).



(a) Communication sélective

(b) Communication à la demande

(c) Communication diffusée (broadcast)

FIG. 5.1 – *Les différents types de communication*

Un des protocoles les plus connus est celui des *réseaux de contrats* (contract net) [Davis 83, Smith 80, Smith 88]. Particulièrement adapté aux agents hétérogènes, il modélise l'appel de procédures «dirigé par le contenu» : un manager décrit la tâche à résoudre en la décomposant en sous-tâches. Un certain nombre d'offrants choisissent de répondre à l'appel d'offre en expliquant pourquoi ils pensent pouvoir remplir le contrat. Le manager évalue chacune des offres et choisit le(s) contractant(s) qui a (ont) fait la meilleure. Un dialogue «privé» peut alors s'engager entre chacun des offrants et le manager pour régler les détails de la résolution. Ce type de protocole s'avère très souple, il peut en effet être dirigé par les buts ou par la disponibilité des agents : les agents déjà occupés ne participent pas à la négociation.

Pour l'anecdote, nous rappelons que certains systèmes ont été élaborés sans communication. C'est ce que Schelling et Tubbs appellent le *contrat tacite*, qui nécessite que les agents aient de bons modèles les uns des autres et n'aient pas de buts conflictuels.

La communication entre agents est le plus souvent asynchrone : un agent peut continuer à effectuer certains traitements en attendant d'avoir des réponses aux messages qu'il a émis.

Système mixant tableau noir et passage de message : MINDS

Le système MINDS (Multiple Intelligent Node Document Servers)[Huhns 87] est un des premiers SMA pour l'organisation et la recherche de documents distribués entre des stations de travail en réseau local.

Dans MINDS, chaque serveur de documents possède des descriptions des documents qu'il gère, et son propre tableau noir où sont enregistrées des «méta-connaissances» sur les documents qu'il stocke et sur ceux qui sont stockés sur les autres serveurs (chaque serveur est géré par un utilisateur). Le rôle de ces méta-connaissances est d'ordonner les recherches d'information sur les différents serveurs. Un élément de méta-connaissance, exprimé sous la forme :

(Smith, Jones, compilateur, 0.8)

représente que l'expérience passée de Smith suggère que la possibilité de trouver des documents à propos de compilateurs parmi ceux possédés par Jones est élevée (8 sur une échelle de 10). Ces méta-connaissances sont mises à jour dynamiquement grâce à un ensemble de règles, qui permettent aux serveurs de document de tirer parti d'une recherche de document. L'ajout d'un nouveau document sur un serveur provoque l'insertion d'une nouvelle méta-connaissance dans le tableau noir du serveur. C'est une recherche d'information lancée par un utilisateur qui provoque la mise à jour du tableau noir contenant les méta-connaissances. Par exemple, une recherche d'un utilisateur *Dupond* sur le mot-clé *organisation* chez l'utilisateur U_2, si elle aboutit, augmente le facteur de certitude chez *Dupond* sur les connaissances de U_2 à propos de *organisation*; qu'elle aboutisse ou non, cette recherche augmente aussi le facteur de certitude chez U_2 sur les connaissances de *Dupond* pour ce mot-clé (l'hypothèse sous-jacente est que si *Dupond* fait une recherche sur ce sujet, il sera un peu plus savant après qu'il ne l'était avant). Même si dans MINDS, les techniques de RI utilisées sont assez peu sophistiquées (modèle booléen, généralisation à l'indexation), et si la révision des «méta-connaissances» repose essentiellement sur des heuristiques, ce prototype est intéressant parce que construit sur le modèle de la circulation des informations dans un environnement de bureau : lorsqu'une personne cherche des informations, elle s'adresse en priorité à ses voisins de bureau si elle a de bonnes raisons de croire qu'ils peuvent l'aider dans sa recherche.

Contenu des communications

Le contenu des communications peut être caractérisé par trois propriétés inter-dépendantes [Durfee 87] :

- la **pertinence** qui mesure la consistance d'un message par rapport à ce qui est déjà su ; la non pertinence des messages peut entraîner un gaspillage des ressources ;
- l'**opportunité** ou l'**à-propos** des messages transmis, c'est-à-dire ce que le message apporte à l'activité du récepteur,
- la **complétude** qui mesure la proportion de solution qu'un message communique ; ce critère permet d'éliminer les messages partiellement ou totalement redondants.

L'information est opportune dans plusieurs situations :

- Lorsqu'elle permet d'éviter des résolutions redondantes de problèmes, par exemple lorsque les agents se communiquent des solutions partielles ; c'est le cas des documents présélectionnés pour une recherche d'information.
- Lorsque l'information est pertinente, soit parce que l'agent à qui elle est adressée ne la connaissait pas, soit parce qu'il la connaissait mais ne l'appliquait pas au problème courant ; c'est le cas d'un agent qui remémorerait à un autre un fait passé. Les documents choisis par l'utilisateur constituent une information pertinente pour les agents de reformulation de la requête : ils peuvent ainsi examiner de nouveaux concepts à l'intérieur de la facette dont ils s'occupent, ou analyser les concepts de nouvelles facettes qui avaient été exclues par l'utilisateur dans sa formulation. Un agent de raisonnement à base de cas peut également apporter de l'information pertinente en exploitant une mémoire de sessions de recherche pour y retrouver une situation de recherche analogue [Smaïl 93b].
- Lorsqu'elle permet d'élaguer l'espace de recherche, en diffusant des solutions partielles ou des échecs antérieurs, de façon à ce que les agents évitent les solutions menant à ces échecs. C'est le cas des documents rejetés par l'utilisateur, et des sessions de recherche qui ont échoué dans la mémoire des sessions antérieures. Un agent peut aussi indiquer sa stratégie de résolution de problème en suggérant explicitement une approche au problème ou en impliquant les autres agents dans une discussion sur un sous-but. Ainsi, l'agent qui traite la facette CONNOTATION devra échanger des informations avec les agents qui traitent d'autres facettes afin de faire converger la recherche sur des critères plus «objectifs» : il peut éviter à ces autres agents un échec dans leur tentative de discrimination des caractéristiques du besoin, échec dû au fait que les connaissances dont ils disposent sont insuffisantes pour mener un raisonnement convergent. Par exemple, si un utilisateur cherche des portraits d'un personnage, et que parmi ces portraits il choisit ceux qui sont plutôt gais et rejette ceux qui sont plutôt sérieux ou tristes, le système ne doit pas en déduire que l'utilisateur ne cherche pas vraiment des portraits de ce personnage (ou pire encore qu'il ne sait pas ce qu'il veut) mais plutôt que la formulation de la requête doit être complétée dans la facette CONNOTATION ; la facette CONNOTATION pourra ainsi guider la facette CONTENU vers une analyse des caractéristiques du personnage (sourire, froncement de sourcils, etc), éventuellement en s'appuyant sur des connaissances acquises par observation.

Un problème important dans l'élaboration d'un SMA est de sélectionner l'information qui sera échangée dans les dialogues entre agents. E. Turner [Turner 93a] élabore des heuristiques de sélection de l'information qui reposent sur la structure de la connaissance qui contient l'élément d'information. Cette structure indique le degré de possibilité que le destinataire connaisse déjà l'information, le type et la difficulté du raisonnement utilisé pour atteindre ou inférer cette information, et l'effet de l'information sur l'accomplissement de la tâche : l'information peut indiquer un échec à éviter ou une contrainte qui permet à la résolution de problème de se focaliser. L'ensemble initial d'heuristiques retenues comprend les éléments suivants :

- la longueur de la chaîne de raisonnement depuis la connaissance partagée jusqu'à l'information considérée doit être directement proportionnelle au degré d'utilité de l'informa-

tion ;

- plus nombreux sont les choix qui ont été faits pour trouver une solution partielle, plus fort est son degré d'utilité ;
- plus un élément d'information a été utilisé pour la résolution de problème, plus son degré d'utilité croît ;
- une information retrouvée à partir d'un cas obtient un degré d'utilité élevé, car ce cas n'est connu que des agents qui l'ont expérimenté ou en ont été informés.

Dans d'autres approches, un agent, lorsqu'il détecte un manque de connaissances chez d'autres agents, génère un but qui consiste à leur communiquer l'information nécessaire. Un but consistant à communiquer l'information est produit à chaque fois que le degré d'utilité de l'information dépasse un certain seuil. Ce schéma a été testé dans le domaine de véhicules sous-marins autonomes coopérants (AUV, autonomous underwater vehicles) qui explorent le fond des océans et y effectuent des tâches scientifiques.

Interactions

Lorsque la communication qui s'établit entre les différents agents d'un système est susceptible de modifier leurs états internes, on parle d'interaction.

> «*L'idée d'interaction n'est pas une notion de sens commun ; elle représente l'aboutissement d'une réflexion longuement développée par les êtres humains, dans leur inlassable effort pour résoudre l'antique paradoxe de l'unité dans la diversité, de l'un et du multiple, pour trouver loi et ordre dans le chaos apparent des changements physiques et des événements sociaux - et pour découvrir ainsi des explications au comportement de l'univers, de la société et de l'homme.*» [Park et Burgess 1907, "Introduction à la science de la sociologie"].

Les interactions qui se produisent entre les agents d'un système peuvent être analysées en termes de coopération et de contrôle.

5.3.2 La coopération

La **coopération**, souvent codée de façon implicite dans les systèmes informatiques, suppose que les différents agents aient un **but commun** et qu'**aucun d'entre eux ne puisse accomplir seul la tâche sous-jacente de façon suffisamment efficace**. Le but commun peut être décidé au départ, ou élaboré en cours de résolution. Il arrive que la coordination soit accidentelle, lorsque deux agents ont un sous-but commun pour réaliser chacun un but global différent. Un cas de coopération accidentelle dans le SRI se produit entre l'agent chargé d'évaluer les résultats a priori pour anticiper un échec et l'agent chargé de sélectionner a priori un sous-ensemble de documents : si l'évaluation porte sur le nombre de documents susceptibles d'être pertinents pour

la requête, ce nombre peut coïncider avec la taille du sous-ensemble de documents présélectionnés. Toutefois, les buts de chacun des agents sont liés mais différents: l'agent de présélection veut uniquement améliorer les performances du système en limitant le volume d'informations à traiter lors de la mise en correspondance, alors que l'agent d'évaluation a priori veut éviter de lancer un cycle de recherche voué à l'échec.

Selon [Durfee 89], la coopération dans un groupe d'agents peut avoir quatre buts génériques :

- augmenter la vitesse d'exécution d'une tâche par le parallélisme,
- augmenter le nombre de tâches réalisables grâce au partage des ressources (information, expertise, unités physiques, etc.),
- augmenter la fiabilité de réalisation des tâches en entreprenant des tâches dupliquées, éventuellement avec des méthodes différentes de réalisation,
- et diminuer l'interférence entre les tâches en évitant les interactions «dangereuses».

Ces buts semblent évidemment conflictuels, parce que l'augmentation du nombre de tâches réalisables peut augmenter les interactions entre elles, y compris les interactions «dangereuses» ; par ailleurs, la duplication de la réalisation des tâches, si elle améliore la fiabilité, diminue les ressources disponibles et donc le nombre d'autres tâches réalisables.

Les styles de coopération des agents vont de la coopération totale à l'égoïsme complet des agents [Werner 89]. L'hypothèse de l'**agent bénévole** [Genesereth 86] entraîne une situation de coopération complète : l'agent bénévole est prêt à adopter les buts d'un autre agent. Cette hypothèse est très simplificatrice, mais elle permet d'éviter de devoir gérer les conflits. Ces agents n'ont pas d'intérêts propres mais sont des entités qui opèrent sous la contrainte de divers axiomes de rationalité, ces axiomes restreignant leurs choix dans leurs interactions. Un agent bénévole n'est pas autonome : il fait tout ce qui lui est demandé. En fait, un agent n'est vraiment autonome que s'il a des buts qui lui sont propres. En l'absence de buts autonomes, un agent n'est capable que de répondre à des sollicitations : il s'agit en fait d'un *agent de service* qui n'est là que remplir de menues fonctions sans avoir de pouvoir décisionnel. De tels agents peuvent bien sûr être utiles au sein d'une architecture, mais ce modèle d'agent très réducteur est à notre sens insuffisant pour composer une société d'agents. Une hypothèse un peu plus réaliste consiste donc à doter les agents d'intérêts propres. Lorsque les agents ont des intérêts propres, ces intérêts peuvent être secondaires par rapport au but global du système (entraide) ou prioritaires, auquel cas les agents coopèrent seulement pour réaliser leurs propres buts (coopération accidentelle ou opportuniste). Cette graduation des hypothèses sur les intentions et préférences des agents permet au système d'avoir un fonctionnement concurrentiel ou au contraire totalement coopératif [Durfee 89, Durfee 92].

Pour qu'il y ait réellement coopération au sein d'un système, les agents doivent donc avoir des buts autonomes, mais être également susceptibles d'adopter les buts des autres s'ils ne sont pas contradictoires avec les leurs.

Zhang a identifié quatre catégories de coopération.

La coopération horizontale : chaque expert peut obtenir des solutions aux problèmes sans dépendre d'autres experts, mais la coopération entre les experts qui utilisent des données et des expertises différentes augmente la confiance en leurs solutions. C'est, par exemple, la coopération de médecins pour un diagnostic médical. C'est aussi le cas des facettes d'information de notre architecture qui peuvent, pour la mise en correspondance, traiter seules la partie d'information qui les concerne. Si des facettes différentes convergent vers les mêmes documents, la probabilité de pertinence de ceux-ci en est renforcée.

La coopération arborescente : un expert «senior» dépend d'experts de plus bas niveau pour obtenir une solution aux problèmes. Par exemple, les décisions d'un ingénieur en chef dépendent souvent du travail des ingénieurs «juniors». Ce type de coopération est à rapprocher de la notion de délégation.

La coopération récursive : les experts dépendent mutuellement les uns des autres pour trouver une solution aux problèmes. Par exemple, des experts en géologie en géophysique et en géochimie dépendent souvent les uns des autres de façon récursive : le géophycisien peut demander au géologue d'effectuer une sous-tâche qui, à son tour, demande que le géophysicien effectue une sous-tâche. La coopération arborescente est un cas particulier de coopération récursive. Un cas de coopération récursive entre les facettes du SRI peut se produire lorsque les liens d'association entre les termes sont exploités. Un agent-facette peut alors signaler à d'autres agents-facettes les termes qui relèvent de leurs domaines sémantiques et qui sont liés à un terme pertinent de son propre domaine. Ceci peut provoquer une réaction en chaîne où les agents-facettes vont se communiquer des informations.

La coopération hybride : les experts utilisent une coopération horizontale jusqu'à un certain niveau de l'arbre des buts et une coopération récursive pour d'autres niveaux. Par exemple, c'est le cas lorsque plusieurs opinions peuvent être obtenues pour améliorer la qualité d'un résultat final. Pour le SRI, la coopération qui s'établit lors de la mise en correspondance entre les différentes facettes de description des documents est hybride : elle est horizontale a priori pour la mise en correspondance, mais peut devenir hybride pour raffiner un résultat.

Zhang a étudié formellement ces schémas de coopération dans le cas d'agents raisonnant de façon incertaine [Zhang 92]. Il propose un modèle de synthèse de solutions différentes qui repose sur les valeurs moyennes et l'uniformité des valeurs incertaines.

La **collaboration** est une notion légèrement différente de la coopération : les agents qui collaborent sont capables de résoudre seuls une tâche et le problème de la communauté est alors d'élire celui qui effectuera la tâche. Le fonctionnement du système est concurrentiel. Un système conçu selon ce schéma est en général robuste grâce à la redondance des aptitudes des agents [Demazeau 89]. Si nous organisons le SRI en fonction de la partition des connaissances, les agents-facettes coopèrent plus qu'ils ne collaborent : aucun d'entre eux n'a une vue globale du problème à résoudre. Par contre, des agents implantant différents modes de mise en correspondance peuvent être vus comme des collaborateurs s'ils travaillent en parallèle sur toutes les descriptions de documents.

5.3.3 Les différents types de contrôle possibles

Les chercheurs en intelligence artificielle distribuée ont exploré de nombreuses solutions pour la définition du contrôle au sein du système.

Gestion du contrôle au méta-niveau du système

Lorsque le contrôle est géré au méta-niveau du système, il peut être [Haton 91]

- procédural : les sources de connaissances (SC) candidates sont détectées par un moniteur qui les place dans un agenda. Un ordonnanceur sélectionne dans l'agenda la prochaine source de connaissance à exécuter.
- hiérarchique : une hiérarchie englobe les sources de connaissances à la fois du domaine et de contrôle. Le niveau le plus bas est constitué des SC du domaine alors que les niveaux supérieurs sont constitués de SC de contrôle. Le contrôle est ainsi propagé de haut en bas de la hiérarchie. En fonction du niveau de la hiérarchie auquel on se situe, la vision que l'on a du système change ; en effet :

  > «*À un niveau donné d'une hiérarchie, un système particulier peut être vu comme extérieur aux systèmes qui sont en dessous et comme intérieur aux systèmes qui sont au dessus ; ainsi le statut (c'est-à-dire la marque de distinction) d'un système donné change quand on traverse son niveau, que ce soit en se dirigeant vers le bas ou vers le haut. Le choix de considérer le niveau du dessus ou du dessous correspond au choix de traiter un système donné comme autonome ou contrôlé (contraint).* » (traduit de Goguen et Varela, 1979, cité dans [Esch 95, p247]).

  Une hiérarchie de contrôle permet de faire abstraction à chaque niveau de détails gérés aux niveaux inférieurs. Toutefois, la hiérarchisation peut introduire des tâches supplémentaires et une certaine redondance dans le traitement, surtout si les agents ne communiquent que par l'intermédiaire de l'entité qui leur est supérieure dans la hiérarchie.
- à base de tableau noir : les sources de connaissance de contrôle sont concurrentes dans un agenda unique. Un ordonnanceur les traite par ordre de priorité. Ce type de contrôle a été mis en œuvre dans BB-1 [Hayes-Roth 85].

La gestion du contrôle au méta-niveau simplifie la modélisation des agents : le concepteur d'un agent spécifie explicitement l'information qu'un agent peut fournir et quelles sont les informations qu'on doit lui fournir. Les agents peuvent être construits de façon indépendante tant qu'ils se conforment à une interface standard. L'agent de planification centrale peut établir le plan global lui-même, mais pour cela, il doit avoir une connaissance globale du processus de résolution de problème.

Le contrôle, dans le système I^3R, est complètement centralisé : un expert de contrôle dirige en effet le déroulement d'une session de recherche (voir figure 5.2). Le schéma de contrôle a été inspiré du modèle de dialogue de Belkin et Daniels [Daniels 85].

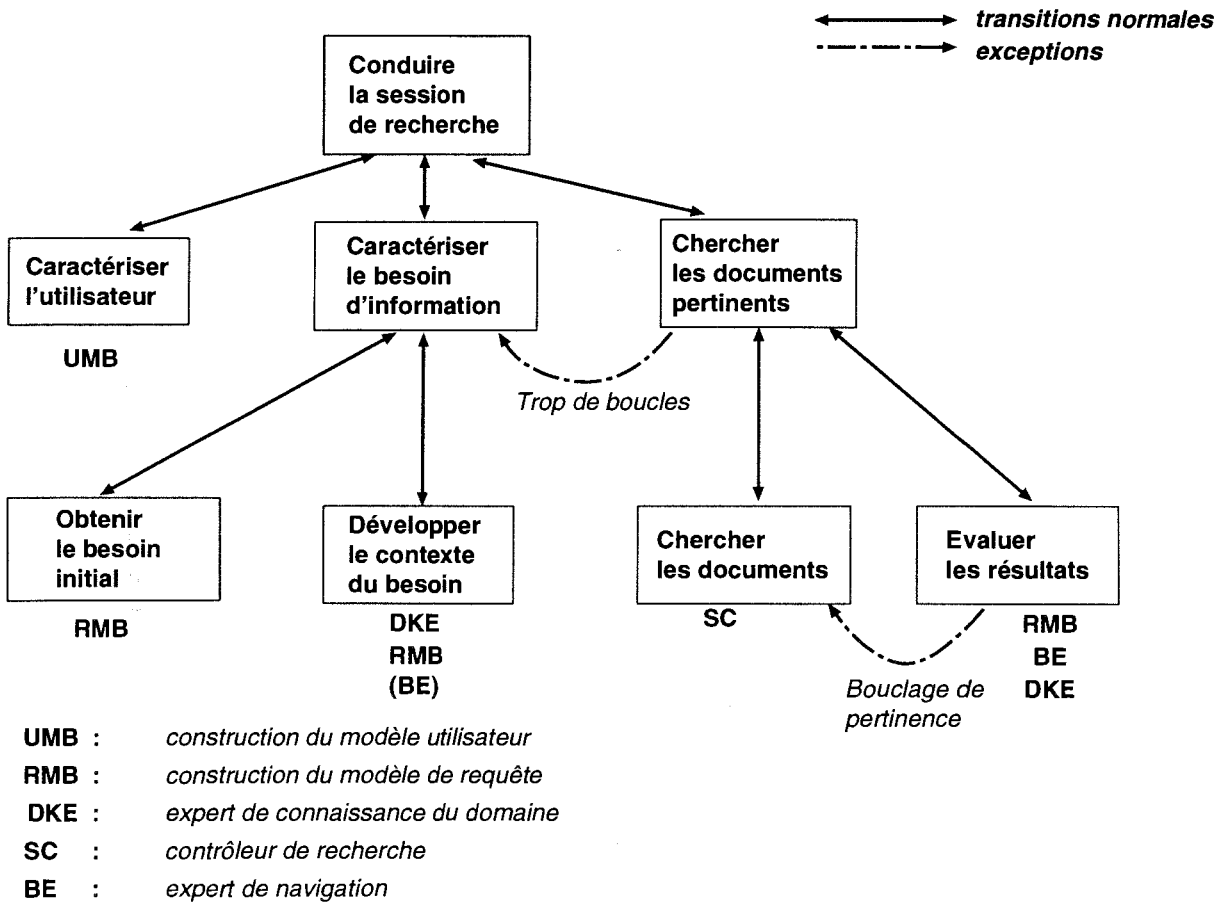


FIG. 5.2 – *Le contrôle dans I^3R*

De même dans CODER [Fox 87], un stratège à cinq composants permet l'identification des experts et des priorités liées au traitement d'une nouvelle requête, il réveille et contrôle les autres experts.

Alternatives au contrôle centralisé

Nous avons dit que le contrôle centralisé simplifie la conception des agents. De plus, la hiérarchisation de la conduite du processus permet de faire abstraction, aux niveaux supérieurs, de détails inutiles pour les décisions à prendre. Toutefois, il s'instaure alors un pouvoir social entre le contrôleur et les agents de résolution, ce pouvoir pouvant être source de conflits (nous reviendrons sur ce point au §5.9). De nombreux chercheurs ont tenté de trouver des solutions alternatives au contrôle hiérarchique. Le contrôle peut rester centralisé comme par exemple dans le système de [Cammarata 83] où les différents agents choisissent un coordinateur qui établit le plan que devront suivre les autres et résout les conflits. C'est le cas également de l'architecture de [Georgeff 88] (cité dans [Moulin 94]) qui rassemble les plans locaux des agents, les analyse et essaie de résoudre les conflits en modifiant les plans locaux et en les synchronisant.

Le contrôle est hybride lorsque chaque sous-problème a un type de contrôle particulier. C'est le cas d'ATOME (figure 5.3) où les tâches contrôlent un ensemble de spécialistes et où la stratégie oriente la résolution à un niveau plus abstrait en sélectionnant les tâches à activer [Lâasri 89].

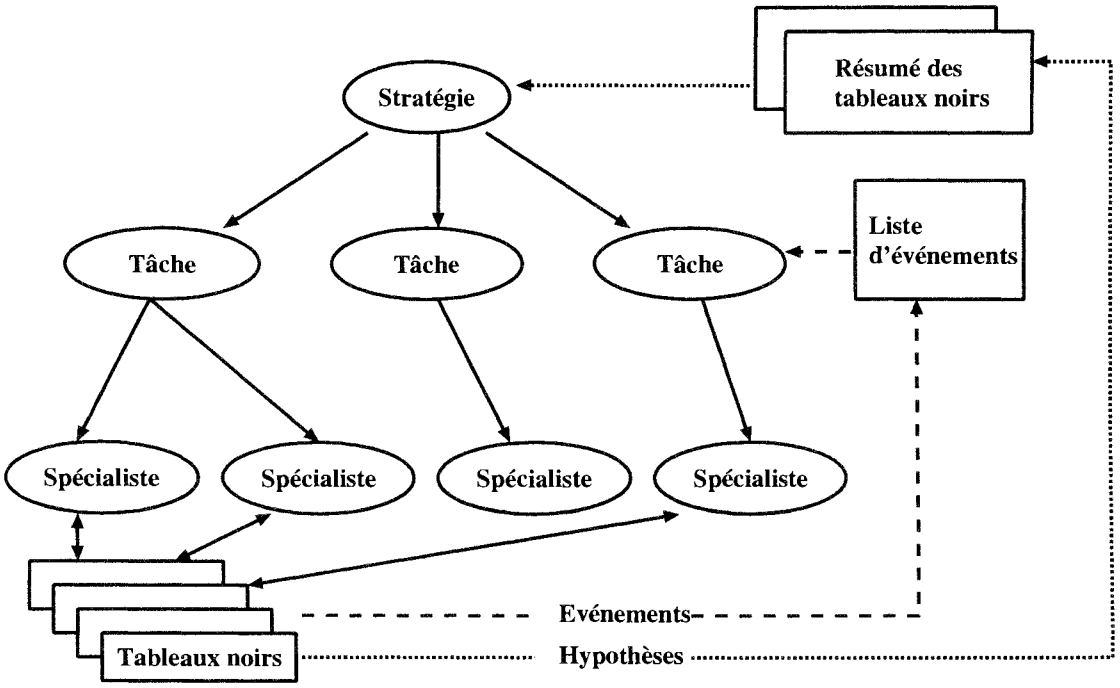


FIG. 5.3 – *Le contrôle hybride multi-phase d'ATOME*

Dans la Planification Globale Partielle [Gasser 91], le contrôle repose sur l'ordonnancement partiel des étapes du plan de façon à identifier les possibilités d'amélioration de la coopération.

Les agents peuvent établir un plan, passer à l'action et modifier ou achever l'élaboration de leurs plans en alternance.

Structuration organisationnelle

La **structuration organisationnelle** [Corkill 83] repose sur des agents dont les intérêts sont spécifiés : ils indiquent quelles sont les informations qu'ils ont besoin de recevoir et quels sont les buts qu'ils veulent réaliser. Lorsqu'un agent génère une nouvelle information, il consulte la structure organisationnelle pour décider à qui «parler». Ce modèle établit un compromis entre une structuration hiérarchique descendante et une coopération fonctionnelle ascendante. Il permet également d'attribuer des priorités et des crédibilités différentes aux agents, de façon à ce que le récepteur d'un message puisse décider de le traiter rapidement ou de l'ignorer. Il a été implanté dans le système générique MACE (Multi-Agent Computing Environment) [Gasser 88], un environnement qui intègre un langage de définition et de manipulation d'agents. C'est également l'approche utilisée dans DVMT (the Distributed Vehicle Monitoring Testbed) [Lesser 83].

5.4 Les buts du système de recherche d'information

Au chapitre 4, nous avons analysé un processus de recherche interactive d'information dont se dégage un ensemble de primitives, qui correspondent aux fonctions d'un SRI. Il nous semble intéressant de refaire cette analyse en terme de buts, puisque ce concept détermine en grande partie les actions des agents. Avant de définir l'architecture que nous proposons, nous allons donc faire un inventaire des buts du système.

Le but principal d'un SRI est la satisfaction du besoin d'information de l'utilisateur. Toutefois, ce but général recouvre en fait plusieurs sous-buts différents. Un critère particulier qui permet de distinguer ces différents sous-buts est le type de demande de l'utilisateur. Nous nous appuyons à nouveau sur la typologie de demandes élaborée par Muriel Cluzeau-Ciry [Cluzeau-Ciry 88] dont nous rappelons les principaux éléments.

Une **demande exploratoire** est une demande de type culturel : consultation d'une collection particulière composée par exemple des œuvres d'un même auteur, ou d'un même mouvement artistique. L'utilisateur peut savoir ce qu'il cherche, mais il ne sait pas s'il peut le trouver dans la collection à laquelle il s'adresse. Face à ce type de demande, les informations qui sont utiles à l'utilisateur sont

- un aperçu du contenu, sous la forme par exemple de la liste des concepts ayant une forte fréquence dans la collection, ou d'un ensemble de documents représentatifs de clusters,
- le modèle de description (liste des facettes et des attributs de description) qui indique quelles sont les informations considérées sur les documents,
- des statistiques sur la stabilité de la collection, de façon à ce que l'utilisateur sache, lorsqu'il a déjà consulté cette collection et échoué dans sa recherche, s'il vaut la peine de

réitérer sa demande.

Une **demande précise** est clairement définie, c'est-à-dire délimitée par plusieurs paramètres qu'on ne peut changer : le nom d'une personne ou d'un lieu, une date, un événement. L'utilisateur sait ce qu'il cherche. Ce type de requête exige en général une réponse rapide. Le but du SRI est alors de *maximiser la précision.* Toutefois, la précision peut n'être atteinte qu'indirectement, car les descriptions de document sont par essence incomplètes, et ne comportent pas forcément le terme utilisé par le chercheur d'information. Par exemple, une photographie montrant un groupe de personnes sera décrite par les noms des personnes représentées qui sont célèbres au moment où l'indexation est faite. Si une des personnes du groupe ne devient célèbre que quelques années plus tard, cette photographie ne sera pas directement retrouvée en réponse à des requêtes précises donnant le nom de cette personne.

Une **demande thématique** est large : l'utilisateur veut obtenir une grande quantité d'images pour avoir beaucoup de possibilités de choix. La demande est souvent formulée de façon imprécise et comprend beaucoup de non-dit. Elle n'est pas définie dans tous ses paramètres (espace, temps, sujet, objet, contexte). Le but du SRI est alors de *maximiser le rappel* en tenant compte de la largeur des critères de recherche. Cette largeur peut être évaluée de deux façons : par la portée de chaque terme dans la collection, c'est-à-dire le nombre de documents que le terme rapporte, ou par son degré de généralité, c'est-à-dire le nombre de spécifiques que l'on trouve sous ce terme dans le thésaurus.

Une **demande connotative** ne concerne pas forcément un élément précis de l'image ni son sujet, mais l'impression qui s'en dégage. Ce type de demande est particulièrement difficile à traiter car il demande de prévoir le sens qu'aura l'image pour le lecteur au moment où il confrontera l'image proposée avec ses images mentales et son système de valeurs. Le but du SRI est alors d'*obtenir des critères objectifs* qui lui permettront de satisfaire la demande.

Pour une recherche interactive, le temps de réponse et la bande passante [19] disponible sont des critères plus pragmatiques, mais également importants pour la satisfaction de l'utilisateur. Ces deux critères ne pouvant pas être systématiquement liés au type de besoin, nous les considérons comme des critères séparés.

Nous retenons donc ces différents sous-buts :

- **obtenir une formulation appropriée de la demande**, en fonction du type de besoin de l'utilisateur ; en particulier, pour un besoin exploratoire, cela peut consister à trouver des concepts représentatifs de la collection, pour les proposer à l'utilisateur ; pour un besoin connotatif, trouver des termes objectifs qui expriment la connotation recherchée;

- **maximiser le rappel**, c'est-à-dire la proportion de documents retrouvés parmi les documents de la collection qui sont pertinents pour la requête,

19. La bande passante correspond à la gamme des fréquences qu'un dispositif peut transmettre sans affaiblissement du signal. La largeur de bande s'exprime en un multiple de hertz, en général en kilo ou megahertz. Par exemple, la bande téléphonique va de 300 à 3400 Hz; la largeur de bande de transmission de sons haute fidélité est de 20kHz, celle des télévisions est de 5,5 Mhz.

- **maximiser la précision**, c'est-à-dire la proportion de documents pertinents parmi les documents retrouvés en réponse à une requête,
- **améliorer les temps de réponse**, en évitant de faire intervenir dans la résolution des agents peu utiles,
- si les documents doivent transiter sur un réseau, **diminuer la bande passante** occupée.

Daniels, Brooks et Belkin [Daniels 85] ont travaillé dans les années 80 à la modélisation de la structure d'un problème de RI pour conduire le dialogue entre le système et l'utilisateur. Dans leur approche, un SRI doit simuler le comportement d'un documentaliste auquel l'utilisateur s'adresserait pour une recherche documentaire. Les buts qu'ils identifient pour une session de recherche d'information sont dans l'ordre (1) que l'utilisateur quitte le système (but trivial pour tout système informatique), (2) que l'utilisateur soit satisfait, (3) que l'utilisateur ait obtenu une réponse appropriée à sa demande et (4) que la formulation de la demande soit appropriée. Pour ces auteurs, une réponse appropriée non seulement satisfait le besoin d'information de l'utilisateur, mais lui est adaptée : ils s'appuient pour cela sur une modélisation de l'utilisateur. Ils décomposent à nouveau le but (4) en plusieurs sous-buts :

1° déterminer si la demande correspond aux capacités du système,

2° déterminer un modèle de l'utilisateur,

3° déterminer l'état du problème : les auteurs font ici référence aux travaux de Belkin sur l'état d'anomalie de connaissance (en anglais *anomalous state of knowledge*, abrégé en ASK) qui est à l'origine de toute démarche de recherche d'information [Belkin 86] : un chercheur d'information ne s'adresse à un SRI que s'il prend conscience d'une anomalie dans l'état de ses connaissances (ceci exclut le butinage ou la navigation ludique dans un ensemble d'informations). L'état du problème peut être déterminé par une étude des travaux de l'utilisateur dans le domaine d'application considéré ; par exemple, pour la recherche de références bibliographiques, cela consiste à déterminer quelles sont les activités de l'utilisateur liées au problème : est-il lui-même auteur d'articles, et si oui, dans la discipline qui concerne sa recherche ?

4° obtenir une description du problème ; ceci consiste à déterminer non seulement le thème général et le sujet de la recherche, mais aussi la finalité de la recherche d'information de l'utilisateur, c'est-à-dire essentiellement l'usage qu'il veut faire des informations cherchées ;

5° élaborer une stratégie de recherche, c'est-à-dire sélectionner des termes représentant la demande, formuler une requête, mettre en œuvre une stratégie de recherche et choisir la collection de données à laquelle la demande sera adressée ;

6° générer une réponse

7° expliquer.

La modélisation de l'utilisateur, bien que déjà intégrée dans certains SRI (IOTA, I^3R), est encore un problème ouvert et controversé : comment éviter d'enfermer un utilisateur dans des

stéréotypes ? Comment généraliser les connaissances acquises sur lui par le système ? Ces questions étant encore en suspens, nous préférons appuyer notre analyse sur la typologie de demandes de Muriel Cluzeau-Ciry plutôt que sur un modèle de l'utilisateur.

5.5 Les principaux agents qui gèrent les connaissances dans un SRI

Nous avons choisi un certain nombre d'agents qui nous semblent représentatifs des activités qui se tiennent dans un SRI. Nous présentons ici ces différents agents : il ne s'agit pas d'une liste exhaustive, mais d'un noyau qui constitue le coeur d'une société d'agents pour la recherche d'information.

Une première distinction peut être faite entre ces agents en fonction du type de connaissances qu'ils gèrent. En particulier, nous individualisons certaines entités du SRI qui peuvent être considérées comme autonomes, parce que leur comportement ne dépend pas du processus de recherche. C'est le cas du thésaurus et de la base descriptive.

5.5.1 L'application de recherche d'information

L'application en elle-même est une entité nécessaire pour stocker les méta-connaissances du système : facettes de connaissances prises en compte pour la description des documents, et formalismes de description possibles : vecteurs de termes ou graphes conceptuels. C'est elle aussi qui répertorie les différents agents de résolution de problème qui sont disponibles.

5.5.2 Le thésaurus

Le thésaurus peut être vu comme un agent autonome : son existence ne dépend pas de celle du SRI. Par ailleurs, les connaissances dont il dispose, même si elles dépendent en partie d'un domaine d'application, doivent pouvoir être réutilisables. En effet, la construction d'un thésaurus est une tâche trop lourde pour être liée uniquement à une application particulière.

Le thésaurus a deux types de comportement : celui qui est nécessaire à la gestion des connaissances, à leur acquisition et au maintien de leur intégrité, et d'autre part celui qui permet l'exploitation de ces connaissances par des agents de recherche d'information.

Étant donné la partition des connaissances que nous avons considérée depuis le début de ce travail, faut-il considérer un ou plusieurs thésaurus pour le système ? Logiquement, nous considérons des *agents-facettes du thésaurus*. Chaque agent-facette du thésaurus gère un ensemble de champs sémantiquement connexes, ce qui correspond à une connaissance partielle du domaine.

Les agents-facettes du thésaurus peuvent néanmoins avoir des connaissances les uns sur les autres ; c'est indispensable pour la gestion des liens d'association d'idées entre les termes, qui

relient souvent des termes appartenant à des facettes différentes. Ces agents particuliers peuvent ne pas être les détenteurs exclusifs de la connaissance du domaine, qui peut être répertoriée sous d'autres formes (lexiques, dictionnaires, etc.).

5.5.3 La base descriptive

La base descriptive regroupe les descriptions de documents. La base descriptive est autonome si le langage d'indexation est libre. Sinon, elle a d'étroites relations avec l'agent qui gère le vocabulaire d'indexation, c'est-à-dire en général un thésaurus (ou, dans notre cas, un ensemble d'agents-facettes du thésaurus). L'autonomie de la base descriptive par rapport au thésaurus dépend aussi des hypothèses sur lesquelles repose l'indexation. Par exemple, la suppression d'un terme du thésaurus peut être interdite si ce terme est utilisé effectivement dans les descriptions. L'ajout d'un nouveau terme dans le thésaurus n'a pas de conséquence immédiate pour la base descriptive, sauf dans deux cas précis :

- si les termes descripteurs des documents sont exclusivement choisis parmi les termes les plus spécifiques du thésaurus (hypothèse d'indexation aux feuilles), la spécialisation d'un terme-feuille du thésaurus par ajout de spécifiques doit être répercutée sur la base descriptive, afin de mettre à jour l'indexation des documents concernés.

- si l'implantation de la représentation des connaissances utilise les niveaux du thésaurus pour représenter les concepts, toute mise à jour du thésaurus devra être répercutée sur les descriptions de documents. Ce cas correspond par exemple à l'implantation par signatures de restriction choisie dans le prototype ELEN : la représentation d'un concept utilise sa hauteur dans le thésaurus (voir §3.4.3 page 92).

Si nous considérons l'organisation en facettes des connaissances du SRI, la base descriptive est un ensemble d'agents de description, un par facette considérée. Nous nommons ces agents les *agents facettes de description*. Chacun de ces agents connaît le formalisme de représentation des descriptions qui le concernent, sait si son vocabulaire est contrôlé et, si c'est le cas, connaît l'agent qui gère ce vocabulaire. Ces agents ne seront véritablement autonomes que lorsque des méthodes d'indexation automatique des documents existeront : la description d'un nouveau document pourra alors être intégrée à la base descriptive sans intervention humaine. Toutefois, pour les documents vidéo, cette forme d'autonomie n'est pas prête d'être effective, même si de nombreux travaux de recherche tendent vers ce but.

Lorsque les documents sont structurés, chaque agent-facette de description doit connaître le niveau de structure du document auquel il s'applique, ce niveau pouvant être unique ou multiple.

5.5.4 La base de connaissances sur les individus et les événements

Nous avons vu que, souvent, des noms propres apparaissent dans les descriptions de documents, qu'il s'agisse de noms de personnes, de lieux ou d'événements. Ces noms propres ont un

comportement particulier : en effet, ils ne peuvent être intégrés à une hiérarchie unique car ils ramènent à un ensemble de concepts. Ainsi, Mistinguett peut être vue comme un `être humain`, ou plus spécifiquement comme une `femme` (importance du genre) ou comme `actrice` (importance de la fonction). De même la `Tour Eiffel` peut être vue comme une `tour` c'est à dire une sorte de `monument`, ou plus spécifiquement comme un `monument parisien` (importance du lieu), ou encore comme un emblème de la France (importance de la connotation). Les connaissances relatives à ces individus (au sens large) peuvent être stockées dans une base descriptive séparée, ou intégrées au thésaurus.

Un ensemble de liens partant de chaque nom propre mène aux différents concepts du thésaurus qui lui correspondent. Tout ou partie de ces concepts pourraient être activés en fonction de la situation de recherche. Pour un besoin précis ou thématique, les concepts des facettes incluses dans la requête seront choisis en priorité. Pour un besoin exploratoire ou connotatif, les concepts décrivant toutes les facettes de l'individu pourront être introduits dans la requête.

5.5.5 Le serveur de documents

Il n'est pas nécessaire de stocker les documents avec la base descriptive. On peut d'ailleurs imaginer plusieurs bases descriptives pour une même collection de documents, ou à l'inverse une base descriptive commune pour une base de documents répartie : le projet VidéoMuseum[20], par exemple, qui vise à établir le recensement systématique et permanent de toutes les oeuvres d'art du XXe siècle de tous les musées et collections publiques, en atteste. Un serveur de documents est une entité autonome dont le rôle est de délivrer des documents à la demande. La gestion du stockage des documents est relativement indépendante du reste de l'architecture. Toutefois, on peut imaginer que les sessions de recherche aient une certaine influence sur le stockage des documents. Par exemple, les documents souvent consultés peuvent être dupliqués. Dans un cadre distribué, les documents peuvent être dupliqués sur un site plus proche de celui où ils sont souvent demandés [Bandon 94].

Le serveur de documents doit délivrer les documents sous la forme qui lui est demandée par le SRI : de manière synchrone ou asynchrone, et dans un temps qui peut être limité ou non.

Pour éviter un couplage trop étroit entre le(s) serveur(s) de documents et la base descriptive – il faut bien savoir à quel document se rapporte une description – il est possible d'avoir recours à une sorte d'annuaire qui établit la correspondance entre le document physique (son nom, sa localisation) et son identité logique connue de l'application.

5.5.6 L'utilisateur

L'utilisateur pourrait être considéré comme un agent à part entière dans la recherche d'information : lui seul connaît le besoin d'information que le système va essayer de satisfaire. Lors d'une session de recherche d'information, une interactivité profonde peut s'établir entre l'utilisateur et le système : non seulement les réactions de l'utilisateur peuvent modifier l'état

20. VidéoMuseum est sur W3 à l'URL : http://www.videomuseum.fr/

de connaissance du système sur le besoin d'information sous-jacent, mais de plus les réponses du SRI peuvent modifier profondément le besoin. Par exemple, l'utilisateur peut découvrir des informations dont il ne soupçonnait pas l'existence et avoir soudain envie d'explorer un nouveau pan de la collection de documents. Toutefois, ses réactions étant filtrées et communiquées par l'interface, c'est cette dernière que nous considérons comme un agent du système, tout en gardant à l'esprit l'utilisateur qui se cache derrière elle.

5.5.7 L'interface utilisateur

L'interface utilisateur est un agent particulier du SRI. C'est elle qui permet d'une part l'acquisition de la requête et des connaissances sur le besoin d'information de l'utilisateur, et d'autre part la transmission des résultats obtenus lors d'une recherche. Nous avons déjà dit que, dans les nouveaux contextes où se trouvent les SRI, les interfaces doivent être de moins en moins couplées au système lui-même. C'est pourquoi nous choisissons de considérer l'interface comme un agent autonome. Le comportement de l'interface est guidé par les actions de l'utilisateur dont chacune provoque un événement, et par les réponses du SRI qui en provoquent d'autres. Un protocole de communication entre le système et l'interface est nécessaire pour définir la façon dont ces deux entités échangent des informations. L'interface, si elle n'est pas dédiée au système, est malgré tout très dépendante du format d'échange utilisé.

5.6 Les principaux agents qui détiennent le savoir-faire du SRI

Outre les agents qui détiennent une partie de la connaissance exploitée lors de la résolution de problème, un certain nombre d'agents possèdent une partie du *savoir-faire* du système. Le découpage que nous effectuons ici est un découpage fonctionnel.

5.6.1 Les agents d'indexation

Les agents d'indexation dépendent du modèle de représentation choisi pour les informations. Si le vocabulaire est contrôlé par un thésaurus, ils dépendent de ce dernier pour obtenir la liste des termes qu'ils doivent reconnaître. Ils peuvent également utiliser d'autres connaissances que le thésaurus ; par exemple, des agents d'indexation du langage naturel utilisent des règles syntaxiques, des agents d'indexation par algorithmes de reconnaissances de formes s'appuient sur des descriptions des formes à reconnaître [Goble 92]. Les agents d'indexation ont un rôle essentiel pour l'autonomie du système : c'est eux qui permettent au SRI de pouvoir traiter une collection de documents. C'est pourquoi ils doivent être définis de façon autonome par rapport au reste du système. Ils doivent pouvoir se déplacer dans un environnement qu'ils ne connaissent pas, demander quel est le vocabulaire d'indexation à retenir et mettre en œuvre leur comportement sur ce vocabulaire. Cela nécessite d'élaborer des **conventions sociales**.

Deux types de conventions peuvent exister pour l'indexation des documents. La première concerne les mots qui seront pris en compte par l'agent d'indexation : certains mots, considérés comme peu porteurs d'information, peuvent être ignorés par ces agents ; les listes de «mots-vides», couramment employées pour exclure certains mots du vocabulaire d'indexation, peuvent être considérées comme une sorte de convention sociale. Ces listes peuvent être réduites aux articles (le, la, les, un, des, ...) ou étendues aux prépositions. Elles reflètent le degré auquel les relations entre les concepts sont prises en compte pour l'indexation. Un autre exemple de convention sociale concerne les *robots* de recherche, (appelés aussi *wanderers* ou *spiders*) qui partent sur World Wide Web (W3) rechercher des informations à la demande sur des sites distants ; ce sont des agents d'indexation particuliers, qui utilisent les liens des hyperdocuments pour explorer récursivement W3. Une sorte de convention sociale permet de restreindre l'activité de ces robots sur un site : un fichier portant un nom conventionnel peut spécifier qu'un robot particulier (ou que tous les robots) n'aura pas accès à des fichiers dont les identifiants sont donnés. Cette convention est destinée d'une part à ménager la confidentialité de données, et d'autre part à éviter aux robots d'explorer récursivement des listes de liens circulaires.

Lorsque l'indexation est manuelle, elle peut malgré tout être assistée par des agents qui contrôlent la cohérence des informations saisies, ou segmentent de façon semi-automatique les documents longs (par exemple, TextTiling [Hearst 93a] pour les textes, ou VideoScribe[21] pour la segmentation des vidéos à l'INA).

5.6.2 Les agents statistiques

Des agents «statistiques» peuvent avoir un certain nombre de rôles différents dans un SRI. Ils peuvent par exemple déterminer la fréquence documentaire des concepts dans la collection.

La fréquence documentaire d'un concept t est calculée par le nombre de documents indexés par ce concept $|D_t|$ par rapport au nombre total de documents de la collection |D|:

$$f(t) = \frac{|D_t|}{|D|}.$$

Pour des documents composites, tous les descripteurs peuvent être ramenés au niveau du document global. La mesure de la fréquence du terme par rapport à la fréquence documentaire inverse, $tf.idf$ (§2.1.2.0), peut alors être utilisée. Cette information permet de donner un aperçu des principaux sujets traités dans une collection.

Des agents statistiques surveillent l'arrivée d'un nouveau document, ou en sont avertis par l'agent d'indexation des documents. Ils s'intéressent plus particulièrement à la date d'arrivée des documents dans la collection, de façon à pouvoir indiquer si la collection est statique ou dynamique. La dynamicité de la collection est mesurée par le nombre de nouveaux documents arrivés dans la collection pour une période donnée par rapport au nombre total de nouveaux documents :

$$Dyn = \frac{|D_p|}{|D|}$$

21. ©INA

où $|D_p|$ représente le nombre de documents arrivés pendant la période p. Pour un SRI personnalisé, la période considérée pourrait être celle qui court depuis la dernière visite de l'utilisateur à la collection. En l'absence de personnalisation, la période pourra être fixée par défaut à un mois, un trimestre, une année ...

Enfin, ces agents peuvent s'intéresser également aux performances du système dans différentes situations (temps de réponse, précision, rappel), et avoir un rôle consultatif important pour l'évaluation du système.

5.6.3 Les agents de recherche d'information

Les agents de résolution de problème sont ceux qui mettent en œuvre une méthode participant à la recherche des documents pertinents. Parmi ceux-ci, nous pouvons distinguer :

- les agents qui permettent à l'utilisateur d'appréhender une nouvelle collection,
- les agents qui donnent une interprétation sémantique de la requête,
- les agents qui sélectionnent un sous-ensemble de documents susceptibles d'être pertinents,
- les agents qui évaluent à priori et anticipent le résultat d'une recherche,
- les agents qui permettent de mesurer une similarité entre documents et requête,
- les agents qui permettent de classer les documents retrouvés pour présenter à l'utilisateur les plus pertinents d'abord,
- les agents de reformulation de la requête à partir des jugements de pertinence de l'utilisateur.

Ces différents agents peuvent être distingués en fonction de leur capacités à atteindre un des buts énoncés au §5.4 : capacité à donner un aperçu de la collection à l'utilisateur, capacité à améliorer le rappel (la proportion de documents retrouvés dans les documents pertinents de la collection), la précision (la proportion de documents pertinents dans les documents retrouvés par le système), le temps de traitement ou, dans le cas d'une distribution physique sur un réseau, la bande passante.

Agents qui peuvent donner un aperçu de la collection

Ces agents sont chargés de répondre à une demande de consultation en particulier dans les cas suivants :

- aucune requête n'est formulée (l'utilisateur ne sait pas formuler sa requête car il ne sait pas ce qu'il peut demander) ou est formulée de façon très vague (un concept très général et souvent présent dans la collection),

- une requête est formulée par la négative : l'utilisateur ne sait pas ce qu'il veut, il sait ce qu'il ne veut pas.

Plusieurs tactiques peuvent être utilisées pour répondre à cette situation, par exemple la présentation de la structure des descriptions du système, du thésaurus ou la construction de clusters de documents.

Agents qui améliorent le rappel

Les agents qui améliorent le rappel sont ceux qui

- traduisent les synonymes : ils permettent en effet d'étendre le vocabulaire de requête au-delà du vocabulaire d'indexation ;
- généralisent les termes de la requête : en passant au générique d'un terme, ils permettent de corriger une sur-spécification de la demande de l'utilisateur ;
- généralisent l'expression formelle de la requête : en opérant une disjonction des sous-requêtes plutôt qu'une conjonction pour le modèle booléen ; en éclatant les graphes en projections partielles pour le formalisme des graphes conceptuels ;
- effectuent une mise en correspondance vectorielle entre requête et documents, la mise en correspondance vectorielle peut rapporter davantage de documents susceptibles d'être pertinents qu'une mise en correspondance de graphes conceptuels par exemple ;
- reformulent la requête en tenant compte uniquement des concepts et non des relations qui existent entre ces concepts ; une reformulation qui repose uniquement sur les concepts permet en effet de découvrir de nouveaux concepts et de nouvelles facettes de description du besoin.

Agents qui améliorent la précision

Les agents qui améliorent la précision sont ceux qui

- permettent de spécifier plus finement la requête, en demandant à l'utilisateur de choisir parmi les spécifiques d'un terme trop général qu'il aurait utilisé, ou parmi les contextes sémantiques dans lesquels ces termes se trouvent dans les descriptions,
- donnent une expression formelle plus stricte de la requête, en spécialisant les rôles et les relations, (par exemple en opérant des conjonctions entre les sous-requêtes),
- effectuent une mise en correspondance en exploitant les graphes conceptuels de description de la requête et des documents,
- reformulent la requête en tenant compte non seulement des concepts, mais des relations qui existent entre eux, et donnent une forte importance aux concepts initiaux de la requête.

Agents qui améliorent le temps de traitement

Pour améliorer les temps de traitement, deux types d'agents peuvent intervenir dans la résolution du problème :

- l'agent qui sélectionne un sous-ensemble de documents sur lequel se déroulera ensuite la mise en correspondance,
- l'agent qui évalue a priori les résultats, ce qui évite au système de lancer un cycle de recherche voué à l'échec.

Ces agents ont des buts distincts et ne doivent pas être confondus.

Agents qui permettent d'économiser de la bande passante

Les agents qui permettent d'économiser de la bande passante sont ceux qui privilégient les échanges les moins volumineux. Par exemple, demander à l'utilisateur d'affiner la description de son besoin en lui fournissant des aides, lui fournir en priorité des descriptions textuelles ou iconiques plutôt que de lui demander de visualiser directement les documents vidéo, lui présenter des résumés ou des photogrammes plutôt que des documents vidéo entiers.

5.6.4 Les outils de visualisation des documents

Pour visualiser les documents, des outils spécifiques sont nécessaires, en particulier pour la vidéo. Ces outils peuvent être indépendants du SRI ou lui être intégrés. L'intégration des outils de visualisation permet de développer des fonctions d'affichage et de manipulation spécifiques à la RI ; par exemple, la mosaïque d'images intégrée à RIVAGE [Chaffiol 93] permet de faire des piles d'images, de les juxtaposer pour les consulter simultanément, etc. Pour la vidéo, nous préférons utiliser des outils de visualisation indépendants. Ceci permet à l'utilisateur de pouvoir utiliser son outil de visualisation préféré. Par ailleurs, les travaux de recherches autour de la vidéo numérique évoluent rapidement. L'indépendance de l'outil de visualisation lui permet d'évoluer au gré des avancées technologiques dans les standards de compression par exemple.

5.7 Stratégie et organisation dynamique des agents

5.7.1 Introduction

La définition d'une stratégie de recherche consiste à choisir quels sont les agents qui seront activés au cours de la résolution de problème. Par rapport à l'approche distribuée de résolution que nous avons adoptée, ces choix doivent porter sur les différents savoir-faire qui vont se succéder au cours d'une session de recherche d'une part, et d'autre part les différentes facettes de description des documents qui interviendront pour la recherche.

Le modèle de processus que nous avons présenté au chapitre 4 correspond assez bien au comportement d'un système multi-agent, une société d'assistants pour la recherche d'information. Le système met en œuvre une stratégie de recherche itérative dont les principales phases sont

- la proaction où il essaie d'anticiper les échecs,
- la résolution du problème proprement dit, qui aboutit au classement des documents selon une certaine mesure et à leur présentation à l'utilisateur,
- et une phase de rétroaction où il remet en cause ses résultats et éventuellement sa stratégie en exploitant les retours de l'utilisateur.

Il est possible de décider de laisser l'utilisateur choisir lui-même la stratégie qui sera adoptée. Toutefois, cela nécessite qu'il dispose d'une bonne connaissance du système, de façon à pouvoir mesurer les conséquences de ses choix. Dans le contexte des systèmes de recherche en ligne, où les utilisateurs ne sont pas forcément formés à la recherche d'information, cette solution semble peu réaliste à mettre en œuvre, malgré ses vertus pédagogiques évidentes. Elle peut toutefois être choisie lorsque les utilisateurs sont des experts en documentation, en particulier pour leur permettre de réaliser des tests grâce aux fonctions de recherche. Nous voyons plutôt le SRI comme un assistant de recherche, ce qui impose qu'il puisse déterminer quelle est la meilleure stratégie à suivre en fonction de la situation afin de la conseiller à l'utilisateur. Le choix qui a été fait dans le système MARS (Multi-Agent Retrieval System) [Nie 94] est de laisser l'utilisateur établir directement la stratégie en choisissant un agent de recherche dans un menu (agent vectoriel, booléen strict, d'analyse du langage naturel, de consultation du thésaurus, de navigation dans un treillis de concepts); l'utilisateur a aussi la possibilité d'établir indirectement la stratégie en donnant des valeurs souhaitées pour le rappel et la précision, ces valeurs déterminant les experts qui participent à la résolution. Toutefois, la collaboration entre les agents est limitée à la combinaison de leurs scores respectifs.

5.7.2 Variations et dynamicité de la stratégie

Dans le processus EXPRIM, la stratégie de recherche était figée. Le modèle paramétré (présenté au § 4.2 page 100) l'a rendue plus flexible grâce à la définition de paramètres dont la valeur influence le comportement global du système. Des stratégies par défaut sont élaborées pour répondre à l'utilisateur en fonction du type de sa demande : précise, exploratoire, thématique ou connotative. Le SRI est doté de capacités d'apprentissage à long terme, qui utilisent le raisonnement à base de cas ; une mémoire des sessions de recherche peut permettre de retrouver un cas similaire à celui que le SRI doit traiter, et d'adapter la stratégie qui avait été utilisée au nouveau cas. Bien que nous proposions des stratégies légèrement différentes de celles de ce modèle, nous nous appuyons sur lui pour nos propositions. Deux limitations existent en effet dans ce modèle. D'une part, la résolution est systématiquement distribuée par facette, ce qui n'est pas forcément souhaitable dans toutes les situations. D'autre part, le système est fermé : le contrôle est procédural, et l'apparition par exemple d'une nouvelle méthode de calcul de similarité entre

requête et documents peut remettre en cause en grande partie le système existant. Il n'est pas possible d'intégrer le nouvel agent qui arrive sans bousculer les autres ...

Nous souhaitons mettre en place une architecture plus ouverte : non seulement la conduite du processus ne doit pas être figée, mais il doit être possible d'intégrer rapidement un nouvel agent sans remettre en cause toute l'architecture. Pour ce faire, il nous semble opportun de concevoir une stratégie qui s'établit de façon dynamique, et, pour cela, de définir les contraintes d'intervention des différents modules en fonction des buts à atteindre et de la situation de recherche.

5.7.3 Auto-conception d'organisation

Une flexibilité maximale peut être obtenue si les agents du système s'auto-organisent pour réaliser leurs buts. L'auto-conception de l'organisation repose sur deux primitives de réorganisation : la composition et la décomposition. Ces primitives permettent de faire varier la macro-architecture du système de façon dynamique. Les demandes de changement d'organisation peuvent être

- des demandes pour changer de niveau de performance organisationnel, qui se manifestent par de nouveaux niveaux de qualités ou par l'exigence de temps de résolution plus courts ou plus longs,
- des changements par type de solution, les solutions adoptées par défaut ne convenant pas à un type de problème particulier,
- des changements dans le niveau de demande de ressources que l'organisation partage avec d'autres dans l'environnement. En particulier pour la vidéo numérique, le problème des ressources est crucial car ce média est gourmand en place mémoire et en temps de transit sur le réseau.

5.7.4 Composition/décomposition en agents-facettes

Dans l'organisation de notre SRI, la décomposition peut être associée au traitement par facettes de la recherche d'information. En effet, le comportement lié aux phases de présélection, de mise en correspondance et de reformulation est un comportement générique qui peut être mis en œuvre sur tout ou partie des connaissances du système. Ces comportements correspondent plutôt à ceux d'*acteurs* auxquels on communique (ou qui vont chercher) les informations qu'ils doivent traiter. En effet, leur comportement peut intervenir à différents moments de la résolution, pour jouer un rôle différent : de même que la mise en correspondance booléenne peut permettre la présélection de documents dans une collection volumineuse, la reformulation peut être mise en œuvre avant qu'une requête n'ait été formulée par l'utilisateur, à partir de choix sur un échantillon aléatoire par exemple.

Nous avons avancé plusieurs avantages à la décomposition de la recherche d'information par facette : la mise en évidence de corrélations partielles entre les documents, la mise en œuvre

de fonctions différentes pour mesurer la similarité entre différents aspects de la requête et des documents en sont deux exemples. Toutefois, cette décomposition peut présenter aussi des inconvénients. Par exemple, pour la présélection des documents, est-il utile d'opérer une décomposition par facette ? Cela ajoute un niveau de traitement supplémentaire qui doit être justifié. Les éléments de décision, pour effectuer ce choix, sont de deux ordres. D'une part, l'implication de différentes facettes dans la requête initiale et le type de besoin d'information de l'utilisateur déterminent si plusieurs facettes vont intervenir dans la résolution : si la requête de l'utilisateur ne porte que sur une seule facette, et que sa demande est précise, la résolution globale peut être mise en œuvre sur une seule facette. D'autre part, si l'utilisateur souhaite une vision globale des résultats, et si plusieurs facettes sont impliquées dans sa requête, une étape supplémentaire de synthèse est nécessaire, ce qui n'améliore pas les temps de réponse du système. La décomposition du traitement par facette sera donc fonction de cet ensemble de critères. Les facettes peuvent ainsi être vues comme des *organisations de résolution de problème* au sens de [Gasser 91].

La primitive de *décomposition* sépare la correspondance entre une collection de règles de résolution de problème et leur interpréteur : pour certaines règles, un second interpréteur peut être créé et un protocole spécialisé aide à lever les inter-blocages en imposant une synchronisation des actions. Les ressources supplémentaires augmentent le parallélisme au sein du problème et améliorent les performances, mais alors la charge due à la coordination augmente aussi. La *composition* combine deux agents dont les actions sont interdépendantes en un seul : cela crée une relation plus forte dans les groupes de connaissance de résolution de problème, et libère des ressources de calcul et de communication en supprimant des agents et des messages entre agents. Une décomposition maximum ne conduit pas forcément au meilleur temps de réponse ou de production, car la coordination et la communication ajoutent leurs propres charges au fonctionnement du système. La composition peut donc améliorer les performances quand la charge de coordination est élevée. Ces deux primitives indépendantes, décomposition et composition, peuvent être invoquées simultanément sur des parties différentes de l'organisation : la résolution de problème et l'auto-organisation sont traitées comme des processus décentralisés.

Par rapport au problème qui nous occupe, l'auto-conception de l'organisation est particulièrement opportune lors de la phase de mise en correspondance. En effet, la décomposition par facette n'est pas forcément adéquate. Par exemple, si les différentes facettes utilisent une méthode de mise en correspondance vectorielle, et que toutes les facettes donnent les mêmes types de résultats, il peut être souhaitable d'éviter la décomposition, ou de composer les facettes «homogènes». De plus, en présence de documents structurés, il peut être souhaitable de traiter ensemble les facettes de description attachées à un même niveau de la structure des documents.

5.7.5 La connaissance organisationnelle

La connaissance organisationnelle comporte deux aspects différents : d'une part les **relations de dépendance et d'interférence entre les agents**, et d'autre part **les règles de réorganisation** qui tiennent compte de la qualité souhaitée pour la solution et de la disponibilité des agents. La qualité de solution souhaitée pour une session de RI dépend en partie du type de besoin d'information de l'utilisateur, comme l'a mis en évidence Muriel Cluzeau-Ciry dans sa typologie de demandes (voir §1.3.4.0) [Cluzeau-Ciry 88].

Une architecture où une telle connaissance est explicitement représentée permet de créer et de détruire des agents, ainsi que de transférer la connaissance organisationnelle et de résolution de problème entre les agents : les agents se construisent et se reconstruisent eux-mêmes de façon active et flexible. Seule la connaissance globale de résolution de problème est fixée a priori, la définition des agents n'est pas figée. Ce type d'architecture a été mis en œuvre par [Gasser 91].

La connaissance organisationnelle peut être globale ou partielle. Une architecture alternative à celle proposée par Gasser repose essentiellement sur l'échange périodique d'informations de modélisation entre les agents, ces informations pouvant concerner les plans en cours, les buts et les agendas : on parle alors de planification distribuée [Durfee 92]. Les agents utilisent ces informations pour réviser leurs activités et prendre des décisions de communication. La recherche de coordination entre agents repose sur des conventions ou lois sociales qui promeuvent les interactions souhaitables entre agents et évitent celles qui sont indésirables.

5.8 Relations de dépendance entre les agents du SRI

Les relations entre agents peuvent être analysées en termes de théorie des dépendances [Castelfranchi 92]. La dépendance entre agents peut être ou non sociale. Un exemple de dépendance non sociale est celle qui existe entre un agent et un événement. Une dépendance est sociale si un agent dépend d'un autre agent pour réaliser un de ses buts. Elle est bilatérale lorsque deux agents dépendent l'un de l'autre. Une dépendance bilatérale n'est pas symétrique car elle ne recouvre pas forcément la même action et le même but dans les deux sens de la relation. Il s'agit d'une **dépendance mutuelle** lorsque deux agents veulent réaliser un but commun, chacun pouvant réaliser une partie du plan (il y a alors coopération), ou d'une **dépendance réciproque** si les buts des deux agents sont différents (il y a alors échange social).

Les dépendances sont graduées quantitativement en plusieurs degrés. Des critères de pertinence pour ces relations permettent de choisir les relations les plus pertinentes pour prédire les différentes formes d'interactions sociales : un graphe de dépendances entre agents leur permet de s'auto-organiser. Ce graphe peut être un arbre ET/OU.

La dépendance OU recouvre deux situations distinctes :

- une même action utile pour réaliser un but peut être faite par un certain nombre d'agents indépendants les uns des autres. Il suffit dans ce cas que l'un des agents effectue l'action.
- des actions alternatives permettent de réaliser un but, et chaque action peut être effectuée par un agent différent. Il faut alors choisir l'action à effectuer.

La dépendance ET peut être

- multi-partite lorsque plusieurs actions sont nécessaires pour réaliser un but et que les actions sont réalisées par des agents différents, ou
- multi-buts lorsque les actions sont réalisées par un même agent pour réaliser un certain nombre de buts interconnectés.

Si nous analysons les dépendances sociales entre les agents du SRI, elles sont de plusieurs ordres : celles liées aux choix de représentation des connaissances du système (vocabulaire contrôlé ou non contrôlé, contrainte d'indexation aux feuilles du thésaurus, formalisme de représentation des descriptions), celles liées à l'interface (interface dédiée ou non dédiée) et celles liées aux contraintes de traitement (temps, ressources disponibles). Nous allons examiner à nouveau chaque phase d'une session de recherche pour y déterminer les dépendances.

5.8.1 Consultation de la collection

Si l'utilisateur veut un aperçu de la collection à laquelle il s'adresse, il peut être dans l'incapacité de formuler une requête. Dans ce cas, il doit pouvoir consulter des méta-informations sur la collection : nombre de documents qu'elle contient, origine de ces documents ; dynamicité de la collection, c'est-à-dire nombre de nouveaux documents arrivant par période de temps ; type d'informations prises en compte sur ces documents (modèle de description). Ces informations

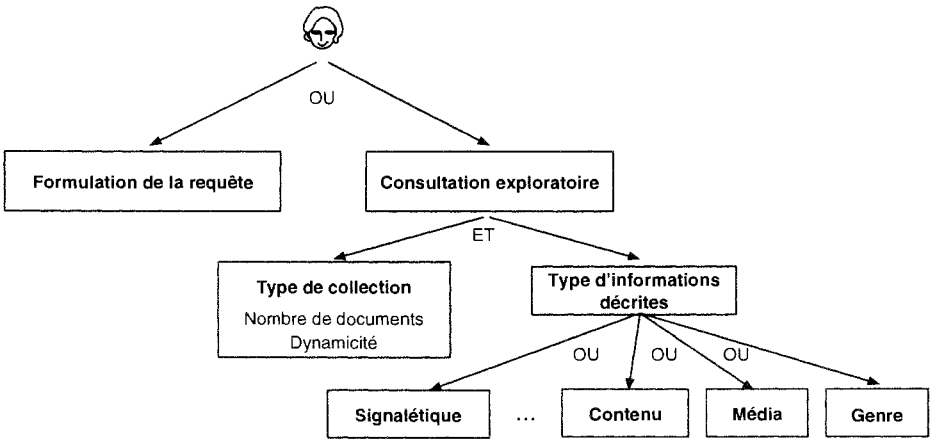


FIG. 5.4 – *Entrée dans un Système de Recherche d'Information*

lui donnent une première idée de ce qu'il peut demander au système. Des fonctions plus sophistiquées peuvent chercher à construire des clusters, groupes de documents dont les centroïdes seraient présentés [Findler 92, Dunlop 93]. La figure 5.4 montre les choix qui devraient être proposés à un utilisateur lorsqu'il entre dans un SRI. Lors de cette phase, le SRI exploite surtout ses propres méta-connaissances.

Dans cette situation, le système doit présenter à l'utilisateur les principaux sujets abordés dans les documents de la collection. Ces sujets peuvent être obtenus à partir des concepts décrivant le genre des documents (actualités télévisées, reportages, documentaires, films scientifiques, etc) et ceux qui décrivent le contenu des documents tout ayant une fréquence documentaire importante. Dans l'application que nous réalisons, où les documents sont des extraits d'actualités télévisées, la période et les lieux de production des documents sont des indices importants pour le chercheur d'information.

D'autres éléments peuvent aider l'utilisateur à se faire une idée de la collection qu'il consulte : le modèle de description de l'application, c'est-à-dire la répartition en facettes et en attributs,

donne une idée du type d'information qui est enregistré pour la collection courante. Enfin, la stabilité de la collection est un autre indice important : s'il s'agit d'une collection statique, l'utilisateur n'aura pas tendance à la consulter à nouveau ultérieurement après un premier échec. S'il s'agit d'une collection dynamique, à laquelle de nouveaux documents sont ajoutés régulièrement, il sera tenté de rechercher si de nouveaux documents sont susceptibles de l'intéresser. Ce genre d'indice peut être obtenu grâce à des statistiques sur les mises à jour de la collection.

5.8.2 Formulation de la requête

La figure 5.5 montre en détail la phase de formulation de la requête, et montre bien le dialogue qui s'instaure entre les agents du système à cette phase. En fonction de son besoin, l'utilisateur entre des termes, et leur affecte éventuellement un poids pour indiquer leur importance relative pour le besoin d'information courant. Le niveau d'importance d'un terme est en effet une indication utile pour le SRI et peut permettre d'améliorer la recherche [Croft 90a].

Il peut demander à consulter les facettes de description, le thésaurus tout entier ou entrer ses termes librement. La dépendance OU qui apparaît entre ces différents modes de formulation reflète les alternatives entre lesquelles l'utilisateur peut choisir. Toutefois, ces alternatives ne correspondent qu'à des alternatives de comportement de l'utilisateur. En effet, les traitements qui suivent ces choix sont étroitement imbriqués.

Lors de cette phase, l'application de RI et le thésaurus se comportent comme des serveurs de données pour transmettre respectivement le modèle de description des documents, ou des concepts représentatifs de la collection.

Le résultat de cette phase peut être : une liste de concepts pondérés ou une liste d'attributs valués par des concepts pondérés. Nous nommons **concept** un terme situé dans le thésaurus, c'est-à-dire non ambigu car attaché à un domaine sémantique.

5.8.3 Interprétation sémantique de la requête

L'interprétation sémantique de la requête n'est nécessaire que si l'utilisateur a entré librement des termes. Elle nécessite l'intervention de deux agents : le thésaurus d'une part, et éventuellement un agent qui connaît le modèle de description de l'application. Ce dernier intervient seulement si les concepts peuvent être attachés à plusieurs attributs, c'est-à-dire lorsque l'intersection entre les domaines de valeurs des attributs n'est pas vide et qu'il existe une ambiguïté sur le rôle d'un terme saisi pour la requête.

L'interprétation peut être synchrone ou asynchrone : soit les concepts sont vérifiés et interprétés au fur et à mesure de leur saisie par le système comme dans la figure 5.5, soit l'interface collecte l'ensemble des termes pondérés et les envoie en bloc à l'application qui les contrôle ensuite.

Si l'interprétation est synchrone, une dépendance ET s'instaure entre l'interface et le thésaurus ; les communications entre ces deux agents sont proportionnelles au nombre de concepts

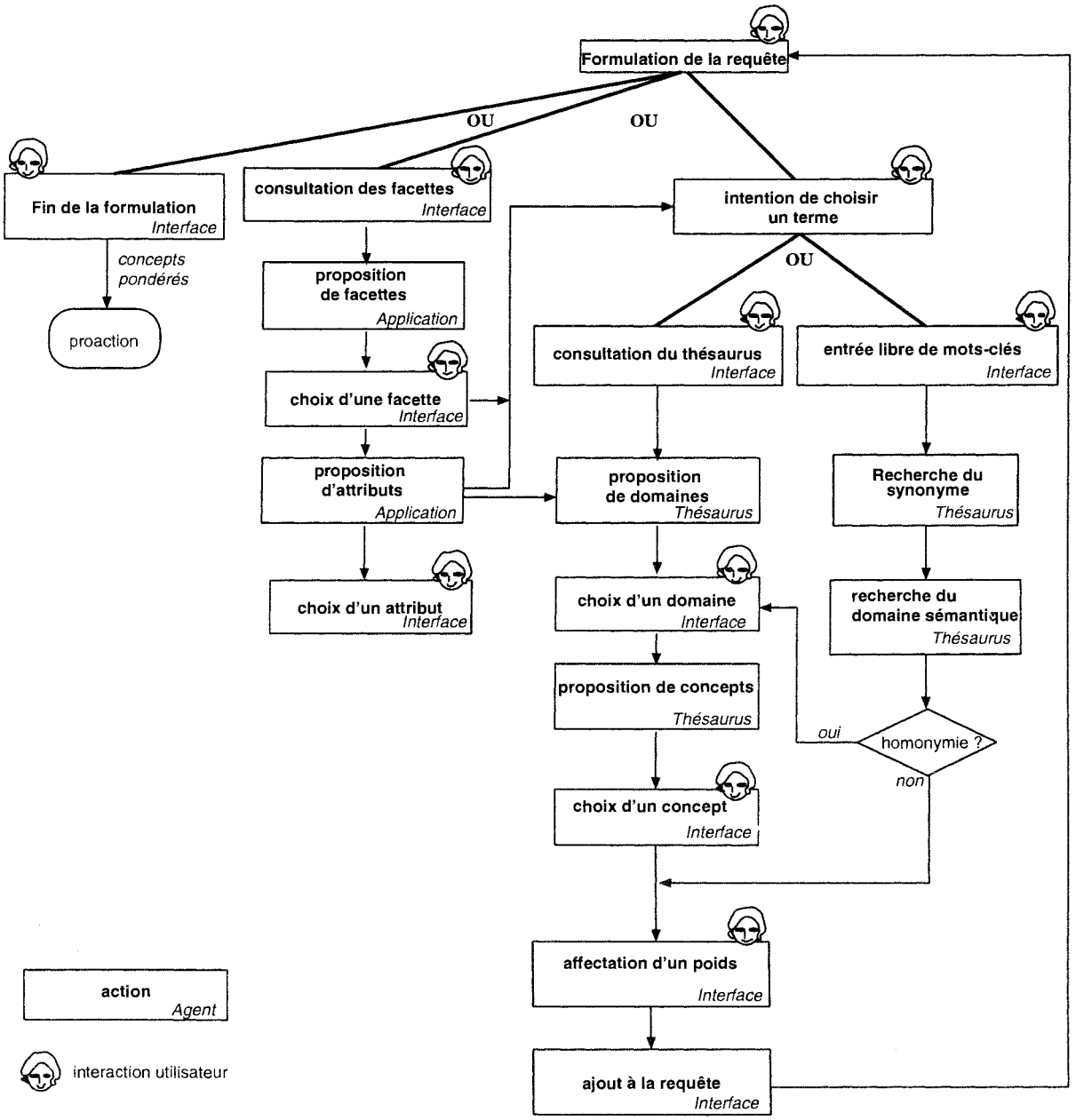


FIG. 5.5 – *Formulation d'une requête sous forme de mots-clés*

entrés pas l'utilisateur. Si elle est asynchrone, ces deux agents sont plus indépendants, et les communications moins nombreuses. L'échec d'une interprétation se traduit alors par une seule communication supplémentaire entre l'agent d'interprétation et l'interface.

L'agent d'interprétation sémantique dépend étroitement du thésaurus et, dans une certaine mesure, de l'agent qui gère le modèle de description de l'application.

Si l'utilisateur n'a pas indiqué les attributs auxquels les concepts sont rattachés, toutes les occurrences possibles d'attributs valués par ces concepts sont incluses dans la requête.

5.8.4 Proaction

Lors de cette phase, le système tente d'anticiper un échec. Les causes de l'échec peuvent être l'absence de documents décrits par les concepts de l'utilisateur dans la collection ou au contraire le fait que les concepts de la requête rapportent trop de documents. Une autre cause d'échec peut aussi être un temps de recherche des documents trop long par rapport au temps dont dispose l'utilisateur.

La figure 5.6 montre en détail le déroulement de cette phase.

Un agent particulier est spécialisé dans le traitement des noms propres. Si la requête doit être généralisée, il peut choisir une facette de description du personnage pour l'ajouter à la requête. Par exemple, si un utilisateur cherche des documents où Mistinguett apparaît, et qu'un seul document de la collection est concerné par ce terme, le concept `actrice` et la période où elle a exercé pourront être ajoutés à la requête si elle doit être généralisée, après accord de l'utilisateur. Cet agent peut être extérieur au thésaurus ou lui être intégré, mais il gagne à connaître le thésaurus : les concepts auxquels il renvoie pourront être mieux exploités s'ils font partie de la base de connaissance du thésaurus.

Présélection des documents

L'agent de présélection des documents exploite des listes inverses, structures d'accès très efficaces qui permettent d'obtenir la liste des documents indexés par un terme. Cette phase du processus doit-elle exploiter la répartition des connaissances par facettes ? Si oui, nous devons instaurer un ensemble de listes inverses par facette de description. Si non, les listes inverses seront globales. Bien sûr la répartition par facette pourrait permettre de paralléliser cette partie du processus. Cependant, lorsque les mots-clés sont entrés librement par l'utilisateur, cette répartition oblige à un traitement supplémentaire pour retrouver la facette correspondant à un concept, et ce traitement supplémentaire peut faire reperdre le temps que la parallélisation par facette peut faire gagner par ailleurs.

Si la présélection repose sur l'interprétation ensembliste d'une requête booléenne, les interprétations larges (union des ensembles de documents rapportés par tous les concepts) et strictes (intersection des ensembles de documents rapportés par les concepts) n'ont pas besoin de faire

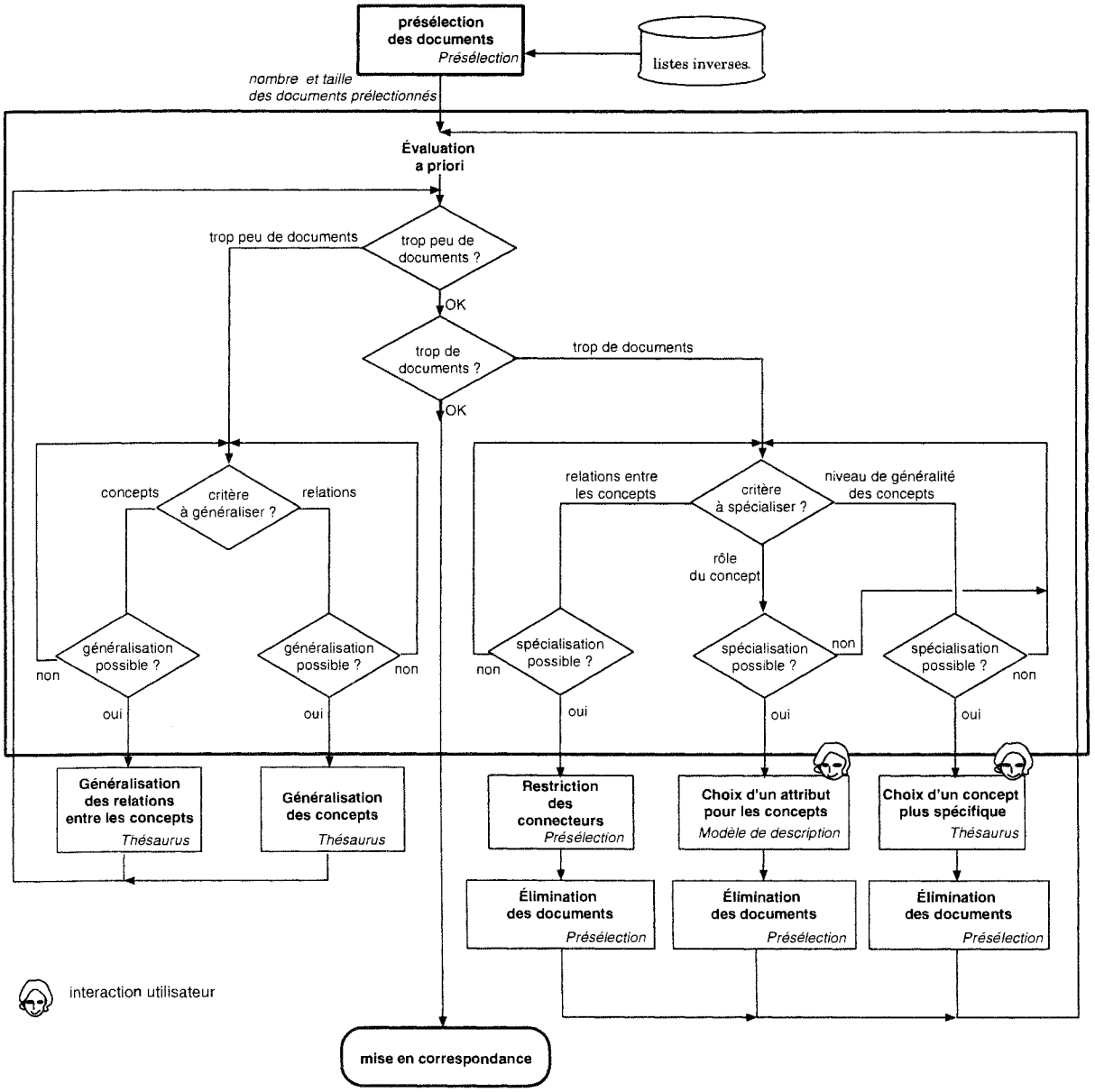


FIG. 5.6 – *Proaction*

intervenir ce type de connaissances. Pour une interprétation médiane, il est possible d'utiliser

1° la connaissance sur l'appartenance des termes à un même domaine : les termes d'un même domaine sont connectés par des OU alors que les expressions disjonctives ainsi constituées sont connectées par des ET ou

2° la connaissance sur la structure du document décrit : pour limiter le nombre de documents présélectionnés, il faut faire l'intersection entre les ensembles de documents rapportés par les concepts liés à la description globale des documents et ceux décrivant des niveaux plus fins de la structure; ceci donne un rôle de censure aux concepts liés au document global : des séquences dont le contenu est a priori pertinent mais qui n'ont pas la signalétique demandée ne seront pas sélectionnées.

La répartition par facette peut bien sûr être également exploitée, soit seule, soit conjointement avec la connaissance sur les niveaux de structure auxquels elles sont attachées dans un document.

D'autres techniques d'implantation peuvent être retenues pour améliorer les temps de traitement, comme l'accès aux concepts par des tables de hachage par exemple. Il n'est donc pas utile de construire les listes inverses en fonction du modèle de description, c'est-à-dire en exploitant la répartition par facettes des attributs de description.

Chaque concept retourne la liste des documents qu'il indexe. La présélection est opérée par une interprétation ensembliste sur ces listes de la requête mise sous forme d'expression booléenne.

La stratégie par défaut mise en œuvre par le SRI dépend du type de besoin, mais aussi du temps dont le chercheur d'information dispose.

Pour une demande thématique, la présélection s'appuie sur une forme large de requête booléenne. Pour une demande précise, la présélection s'appuie sur une forme standard.

L'agent de présélection des documents transmet à l'agent d'évaluation a priori le nombre total de documents retenus et la taille globale qu'ils représentent.

Évaluation a priori des résultats

L'agent qui évalue les résultats a priori s'appuie sur deux types de critères : le **nombre** et la **taille des documents** pré-sélectionnés d'une part, et le **temps de traitement** qui va être nécessaire pour la mise en correspondance d'autre part. Il doit donc connaître la stratégie par défaut que le système va utiliser pour cette mise en correspondance, et peut éventuellement provoquer sa remise en cause si l'agent de présélection ne peut plus agir.

L'agent d'évaluation peut obtenir des indications quant aux valeurs moyennes de ces critères en ayant recours à l'agent-mémoire du système, qui exploite le raisonnement à base de cas sur une mémoire de sessions où sont enregistrées les performances réalisées par le système dans différentes situations de recherche.

Il peut également déclencher une déformation de la requête, qui sera propagée vers l'agent de présélection pour qu'il modifie en conséquence l'ensemble de documents présélectionnés.

Généralisation de la requête

La généralisation est déclenchée par l'agent d'évaluation a priori si le nombre des documents présélectionnés est trop insignifiant. Nous donnons ici des exemples de comportements des agents de généralisation en fonction du type de besoin.

Pour satisfaire un besoin précis, la généralisation procède dans l'ordre :

1° à la relaxation des éventuels connecteurs logiques introduits entre les concepts de la requête (agent de formulation booléenne de la requête), puis

2° en passant au générique du terme pondéré le plus faiblement dans la requête (thésaurus et requête). Si tous les termes ont le même poids dans la requête, le choix du terme à généraliser se fait en fonction de son niveau de généralité dans le thésaurus d'une part, et de sa portée – c'est-à-dire le nombre de documents qu'il rapporte – d'autre part, ce qui nécessite une consultation de la base descriptive si cette information n'est pas disponible dans le thésaurus : on généralisera le terme le plus spécifique qui rapporte le moins de documents. Si le générique du terme déformé rapporte un nombre de documents pléthorique, on proposera à l'utilisateur de choisir parmi ses frères, c'est-à-dire les concepts de même niveau dans le thésaurus, plutôt que de les inclure tous dans la requête.

Si le besoin est thématique, le SRI doit exploiter les liens d'association entre les termes du thésaurus, de préférence dans la même facette, pour suggérer de nouveaux concepts de recherche ayant un rapport avec ceux entrés initialement par l'utilisateur. C'est donc l'agent-facette du thésaurus correspondant qui intervient dans ce cas.

Pour un besoin connotatif, les liens d'association du thésaurus qui partent des concepts connotatifs vers des concepts d'un autre type seront exploités. L'agent-facette du thésaurus qui gère les domaines liés à la connotation intervient.

Spécialisation de l'interprétation

La spécialisation de l'interprétation intervient lorsque le nombre de documents susceptibles d'être pertinents est pléthorique et lorsque le temps de recherche doit être limité. Les critères de spécialisation dépendent aussi du type du besoin d'information courant.

Dans le cas d'un besoin précis, il est possible de passer à un type d'interprétation strict de la requête formalisée afin d'éliminer certains document (agent de formalisation booléenne de la requête). Il est également possible d'analyser avec profit le niveau de généralité des concepts utilisés dans la requête (thésaurus), et de faire spécifier plus finement par l'utilisateur les concepts jugés trop généraux (interface). Si le domaine sémantique d'un concept peut correspondre à plusieurs attributs, l'utilisateur devra choisir l'attribut auquel il attache le concept, de façon à rendre le rôle de ce concept moins ambigu. Par exemple, si la requête contient le terme "Jean Vigo", l'utilisateur devra choisir entre les attributs AUTEUR de la facette SIGNALÉTIQUE et PERSONNE de la facette CONTENU. La dernière tentative de spécialisation consistera à faire choisir à l'utilisateur une relation entre les concepts de sa requête, parmi les relations dans

lesquelles le concept peut entrer (base canonique des graphes conceptuels).

Pour un besoin thématique, le système procédera par restrictions successives de l'interprétation booléenne (formulation booléenne de la requête), puis pourra demander à l'utilisateur de choisir parmi les spécifiques du terme qui rapporte le plus de documents (facette du thésaurus, base descriptive éventuellement pour la portée des termes, interface utilisateur). Enfin, il pourra demander à l'utilisateur d'ajouter des concepts à sa requête. Ces critères supplémentaires, dans le cadre d'une interprétation stricte, réduiront le nombre de documents susceptibles d'être pertinents.

Dépendance entre les agents de proaction

Il existe une dépendance bilatérale entre l'agent de présélection des documents et l'agent d'évaluation des résultats a priori : l'agent d'évaluation ne peut opérer qu'après que l'agent de présélection ait obtenu un premier sous-ensemble de documents et ne lui ait transmis les données nécessaires à l'évaluation (nombre et taille globale des documents présélectionnés). En retour, l'agent d'évaluation demande à l'agent de présélection de modifier son comportement (élargir/restreindre sa sélection) ou lui donne le feu vert pour transmettre ses résultats aux agents de mise en correspondance. Pour l'implantation, l'agent de présélection ne pourra libérer sa mémoire qu'après avoir obtenu le feu vert de l'agent d'évaluation a priori. S'il existe une mémoire de session, l'opération de libération de mémoire correspond à l'enregistrement des données dans la mémoire de session. Le but des deux agents (évaluation/présélection) n'est pas le même : pour l'agent de présélection des documents, il s'agit avant tout d'optimiser les temps de traitement alors que pour l'agent d'évaluation, il s'agit d'anticiper un échec de la recherche. L'agent d'évaluation exploite les résultats de l'agent de présélection pour réaliser son propre but ; il s'agit d'un échange social au sens de Castelfranchi (voir page 179). L'agent de présélection peut très bien opérer sans l'agent d'évaluation, mais l'agent d'évaluation l'aide à réaliser son propre but en lui indiquant les valeurs standards qu'il peut tirer de la mémoire du système d'une part, et en adaptant d'autre part cette évaluation au type de besoin de l'utilisateur en suggérant une solution.

5.8.5 Mise en correspondance entre requête et documents

Nous disposons de deux types d'agents pour la mise en correspondance : **l'agent vectoriel** et **l'agent exploitant les graphes conceptuels**. Ces deux modes de mise en correspondance peuvent être alternatifs (un résultat peut être obtenu si l'un ou l'autre est mis en œuvre) ou complémentaires. Lorsque les deux types d'agents interviennent sur les mêmes facettes, ils peuvent travailler en complément l'un de l'autre : par exemple, l'un va diviser les classes de pertinence de documents que l'autre a constituées si elles sont trop étendues (et si le temps imparti au traitement le permet). L'agent de graphes conceptuels peut ainsi intervenir pour raffiner une partie du classement. Une dépendance mutuelle s'instaure alors entre les deux agents : ils ont un but commun, classer les documents, et coopèrent pour le réaliser. L'agent vectoriel se comporte dans cette situation comme un agent de présélection pour l'agent de mise en correspondance de

graphes conceptuels.

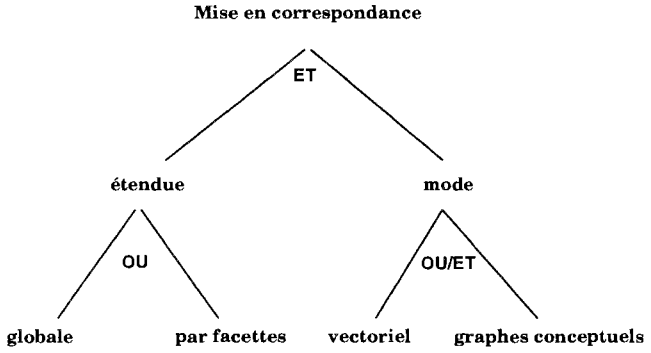


FIG. 5.7 – *Choix pour la mise en correspondance*

Pour un besoin précis, c'est le modèle de graphes conceptuels qui sera choisi pour calculer la similarité entre la requête et le document, alors que pour des besoins exploratoires ou thématiques, le modèle vectoriel suffit.

Pour un besoin connotatif, la stratégie choisie dépend des concepts utilisés dans la requête. Si les termes relatifs à la connotation concernent un document ou une des ses parties (par exemple, l'utilisateur cherche une séquence gaie), le modèle vectoriel peut suffire. Si par contre ces termes qualifient un autre concept de la requête (par exemple, l'utilisateur cherche des séquences où Françoise Sagan est gaie), le modèle de graphes conceptuels devra intervenir.

Pour pouvoir réaliser leur but, les agents de mise en correspondance ont besoin d'une requête exprimée dans leur formalisme. Si l'interface n'est pas dédiée à un modèle de requête, la mise en œuvre de ces agents nécessite de réaliser un but intermédiaire : transformer la requête dans le formalisme nécessaire à la mise en correspondance. Pour chaque type de modèle, il devra donc exister un agent de formalisation correspondant. La dépendance entre un agent de mise en correspondance et celui de formalisation est une dépendance multi-partite (dépendance ET) : ces deux types d'agents ne peuvent en effet intervenir l'un sans l'autre.

5.8.6 Synthèse des résultats : élection des documents pertinents

Si l'utilisateur souhaite une forme synthétique des résultats de sa requête, le système doit opérer cette synthèse en fonction de la situation. Toutefois, il est souhaitable que l'agent qui effectue la synthèse des résultats soit indépendant du nombre d'agents qui entrent dans la mise en correspondance : il doit être possible d'ajouter un nouvel agent sans devoir modifier le reste de l'architecture.

Cette synthèse peut s'effectuer sous la forme d'un vote : chaque agent-facette dispose d'un nombre de voix proportionnel au nombre de documents que l'utilisateur souhaite retrouver. Il affecte un nombre de voix aux documents qu'il a retrouvés. Plusieurs modes de calcul sont possibles pour le nombre de voix affecté à chaque document :

1. Les voix sont distribuées proportionnellement au rang obtenu par le document, le clas-

sement étant effectué par mesure de similarité décroissante. Si l'utilisateur souhaite voir n documents, si le $n$ième document est classé au rang $i+1$, et si le nombre cumulé des documents classés jusqu'au rang $i+1$ est supérieur à n, seuls les i premiers rangs sont considérés. Le nombre de voix attribuées correspond au rang du document dans le classement par ordre de pertinence croissante :

$$nv_k = rang_k.$$

Les documents classés au rang $i+1$ se partagent les voix restantes. Pour l'exemple considéré, nous obtenons le nombre de voix indiqué en colonne ① du tableau 5.8. Ce type de calcul fait abstraction des mesures de similarité obtenues, seul le classement est pris en compte.

2. Le nombre de voix attribué à chaque document est proportionnel à la mesure de similarité obtenue :

$$nv_i = \frac{nv}{\sum_{i=1..nbcl}(msim_i \times nbdoc_i)} \times msim_i$$

où nv_i est le nombre de voix attribué à la classe de pertinence i
nv est le nombre total de voix dont dispose la facette,
$msim_i$ est la mesure de similarité correspondant à la classe i,
$nbcl$ est le nombre de classes de pertinence obtenues pour les n premiers documents, n étant le critère de censure,
$nbdoc_i$ est le nombre de documents dans la même classe de pertinence.
Pour l'exemple considéré, le nombre de voix attribué apparaît en colonne ② du tableau 5.8.

Par exemple, si l'agent de mise en correspondance sur la facette CONTENU dispose de vingt voix, ces deux modes de scrutin donnent les résultats donnés figure 5.8. D'autres variantes de scrutin peuvent bien sûr être utilisées.

Mesure de similarité	Rang	Document	Nombre de voix	
			①	②
0,72	1	D_{14}	4	4,12
0,54	2	D_{27}, D_5	3	2,94
0,43	3	D_{10}, D_{36}, D_{21}	2	2,53
0,10	4	D_{33}, D_2, D_{11}, D_3	1	0,58

FIG. 5.8 – *Élection de documents pertinents : table des votes*

La synthèse consiste simplement ensuite à comptabiliser le nombre de voix recueillies par chaque document, et à classer les documents par nombre décroissant de voix recueillies. Ce mode de calcul ne varie pas en fonction du nombre ou du type d'agents-facettes qui interviennent dans la résolution.

Au fur et à mesure des cycles de résolution, le nombre de voix dont disposent les agents de mise en correspondance peut être proportionnel à leur utilité (voir §5.11.2).

5.8.7 Préparation de la visualisation

L'agent qui prépare la visualisation dépend du type d'interface du système (dédiée / non dédiée, hypermédia ou non). Si l'interface est hypermédia, cet agent peut être complexe et exploiter les résultats de la mise en correspondance pour composer des documents hypermédias qui permettront à l'utilisateur d'explorer les résultats par navigation, comme dans Microcosm [Davis 92] ou Hyper-RIME [Kheirbek 95] par exemple.

Des agents différents devront intervenir en fonction du standard d'échange éventuellement utilisé : HTML (HyperText Markup Language), MHEG (Multimedia Hypermedia Exchange Group [MHEG 93], OMFI (Open Media Framework Interchange [Buford 93]), etc.

5.8.8 Visualisation

L'interface reçoit les informations à afficher à l'utilisateur, et en particulier la liste des documents pertinents. En fonction du format du document, elle fait appel à un outil de visualisation dédié.

Pour chaque document, elle doit permettre de recueillir les choix des utilisateurs. Ces choix peuvent porter sur les descriptions de documents (tout ou partie), ou sur d'autres critères. Elle retourne au SRI les éléments choisis ou rejetés par l'utilisateur : identification du document, granularité de description choisie, valeur du choix.

Cette phase, si elle est limitée à cette définition, est forcément dépendante de la formulation de la requête. Or, certains systèmes ont déjà été conçus qui ne nécessitent pas cette formulation préliminaire. En particulier, il est possible de concevoir deux fonctionnements alternatifs :

- celui des systèmes hypermédias où il existe un graphe de navigation par défaut, ce graphe pouvant correspondre à un parcours prédéfini de découverte de la collection ; par exemple, tout document résultat aurait des liens vers des documents biographiques sur les personnages qu'il montre, des liens vers des représentations géographiques du lieu où se situe le document ;

- celui des systèmes d'analyse du contenu des images par reconnaissances de forme où des images sont tirées au hasard pour fournir des points d'entrée dans la collection ; l'utilisateur en choisit une dont les caractéristiques (couleurs, textures, formes) sont analysées et servent de requête.

5.8.9 Analyse des choix et reformulation

La mise en œuvre de la reformulation dépend essentiellement du temps dont l'utilisateur dispose pour sa recherche. L'utilisateur doit également s'être prononcé sur des documents présentés par le système. Si nous voulons donner une indépendance maximale à cette phase du processus, il faut supposer qu'elle peut intervenir sans qu'une requête initiale ait été formulée.

La reformulation se fait facette par facette. Une facette examine les descriptions de documents choisis et rejetés par l'utilisateur pour l'aspect qui la concerne.

Le but de la reformulation diffère en fonction de la situation de recherche. La stratégie par défaut dépend du temps de traitement dont on dispose. Si ce temps n'est pas limité, toutes les facettes sont mises à contribution pour la reformulation. Pour un besoin précis, la reformulation devra tendre à la convergence de la recherche : l'introduction de termes liés à de nouvelles facettes par rapport au besoin exprimé initialement ne devra se faire que s'ils améliorent effectivement la convergence. Les agents-facettes de description n'interviennent qu'au prorata de leur utilité pour la convergence de la recherche. Nous reviendrons sur ce point au §5.11.2. La reformulation portera, si possible, sur des graphes conceptuels.

Pour un besoin exploratoire, une reformulation vectorielle est suffisante. La convergence de la recherche ne sera pas encouragée. Ceci correspond à une faible prise en compte de la requête précédente (paramètre α faible dans la formule 4.3 page 127). Par contre, les choix et les rejets auront une forte importance.

Pour un besoin thématique, la reformulation vectorielle sera également utilisée par défaut. L'ancienne requête, les choix et les rejets auront la même importance.

Pour un besoin connotatif, le type de reformulation choisi dépendra du niveau auquel sont attachés les termes de connotation. L'exploitation des liens d'associations d'idées entre les termes du thésaurus est particulièrement appropriée dans cette situation : ces liens peuvent permettre de «traduire» les termes abstraits de connotation en termes concrets. Si les termes connotatifs sont attachés au niveau du document, une reformulation vectorielle peut suffire. Les facettes de description du genre et de la morphologie du document pourront être exploitées avec davantage de soin ; en effet, particulièrement pour la vidéo, les impressions du spectateur sont suscitées par des cadrages, des éclairages, ou des mouvements particuliers de la caméra [Colin 92]. Si les termes connotatifs caractérisent un objet contenu dans le document, la reformulation par graphes conceptuels sera utilisée, car elle seule permet d'exploiter réellement le contexte de l'objet représenté dans le document, comme ses caractéristiques : par exemple, le terme `gai` dans la facette CONNOTATION pourra être traduit par la caractéristique `sourire` dans la facette CONTENU. La facette MORPHOLOGIE peut également jouer un rôle dans ce type de situation : une prise de vue de bas en haut donne de l'importance à un personnage par exemple, alors que l'inverse donne une impression d'écrasement.

Dans l'avenir, d'autres agents de reformulation pourront intervenir, qui s'appuieront non sur les descriptions textuelles de documents, mais sur la recherche de critères visuels discriminants : couleur, texture, etc [Picard 95].

5.9 Les conflits

Les conflits se produisent essentiellement en présence de buts contradictoires. Ils peuvent naître au sein même d'un agent lorsqu'il doit par exemple choisir entre plusieurs buts, mais le plus souvent au sein d'une société d'agents lorsque plusieurs agents ont des buts opposés ou

incompatibles.

5.9.1 Utilité des conflits

En psychologie sociale, les conflits sont considérés comme une force essentielle et positive pour la coopération et la stabilité d'une société [Deustch 71] (cité dans [Galliers 89]). Ils ne sont pas forcément destructeurs mais comportent de nombreuses fonctions positives : ils empêchent la stagnation, stimulent l'intérêt et la curiosité, permettent de circonscrire les problèmes et de trouver des solutions. Le conflit exerce une pression pour l'innovation et la créativité dans les systèmes sociaux : il permet un décalage entre les normes et les attentes et revitalise les normes existantes. C'est un mécanisme de réajustement de normes à de nouvelles conditions. Il clarifie les objectifs et permet l'accord plutôt que la subordination. L'acceptation des conflits est donc une force positive pour la maintenance et l'évolution d'un système.

5.9.2 Types de conflits au sein d'un SRI

Les conflits portent sur des valeurs mutuellement exclusives ou incompatibles (en général des croyances ou des buts), et résultent d'une limitation des ressources (à cause de leur rareté ou parce qu'elles ne peuvent être disponibles à plusieurs endroits simultanément).

Les conflits qui peuvent exister au sein du SRI peuvent être liés :

- à l'incompatibilité entre le temps de traitement nécessaire pour obtenir la qualité de résultat souhaitée et le temps dont dispose l'utilisateur,
- à l'incompatibilité entre les ressources nécessaires pour répondre à une demande et les ressources effectivement disponibles (par exemple, vouloir répondre à une demande d'information exploratoire d'un utilisateur alors que le système où la demande s'exécute est très chargé),
- au fait que la mise en correspondance ne puisse pas départager de façon suffisamment fine une classe de pertinence des documents pour respecter le critère de censure fixé, c'est-à-dire le nombre maximum de documents qui seront présentés à l'utilisateur,
- à la recherche de l'amélioration conjointe de la précision et du rappel, ces deux critères d'évaluation variant souvent en sens inverse l'un de l'autre.

Résolution des conflits

La résolution des conflits dans les SMA peut être obtenue par diverses tactiques :

- soit par un *superviseur* [Georgeff 88] qui examine les plans des différents agents, détecte les conflits et a la capacité de modifier leurs plans locaux, de les synchroniser pour améliorer le fonctionnement global ; il s'agit alors d'une résolution centralisée des conflits ;

- soit par un médiateur [Syraca 88] qui aide à l'élaboration d'un compromis en négociant parallèlement avec les deux partis en conflit,
- soit par négociation directe entre les agents impliqués ; la résolution est alors totalement distribuée.

Castelfranchi [Castelfranchi 89] relie les conflits au sein d'un système aux notions de

pouvoir social entre les agents : un agent peut avoir le pouvoir d'en aider un autre à atteindre son but ou au contraire de l'en empêcher ; c'est le cas en particulier lorsque l'organisation est centralisée, et que l'intervention d'un agent est décidée de façon hiérarchique ;

dépendance : un agent dépend d'un autre agent pour réaliser son propre but ; par exemple, l'agent de mise en correspondance a besoin de l'intervention de l'agent de formalisation correspondant pour «traduire» de manière adéquate la requête de l'utilisateur ;

pouvoir d'influence : un agent peut modifier les hypothèses d'un autre agent ; un agent-facette peut ainsi indiquer à un autre agent-facette des termes associés relevant de son domaine ; un agent de pré-sélection influence l'agent de mise en correspondance qui ne classera que le sous-ensemble présélectionné ;

possession : un agent dispose de ressources qui sont nécessaires à un autre agent. Par exemple, l'agent thésaurus dispose de la connaissance sur les relations entre les termes qui seront nécessaires à différentes phases du processus de recherche.

Les conflits que nous venons d'évoquer peuvent être résolus de différentes façons. S'il existe une incompatibilité entre le temps de traitement nécessaire et le temps souhaité par l'utilisateur, il est possible soit de dégrader la qualité de la solution, on privilégie alors le temps de traitement, soit de demander à l'utilisateur s'il accepte le temps de traitement nécessaire à la qualité de la solution qu'il demande. Si le conflit concerne les ressources disponibles, il est possible de mettre en attente la demande jusqu'à ce que des ressources se libèrent ou de diminuer les ressources nécessaires en éliminant certains agents de la résolution parce qu'ils consomment des ressources sans améliorer de façon significative la résolution ; là encore, c'est en général l'utilisateur qui décidera entre ces alternatives. Si la mise en correspondance ne peut pas départager une classe de pertinence pour respecter le critère de censure fixé, il est possible de mettre en œuvre une méthode de mise en correspondance alternative sur cette classe de documents afin d'obtenir un classement plus fin ; dans ce cas, l'agent complémentaire jouera le rôle d'arbitre.

Dans la plupart des cas, l'apparition d'un conflit permet de signaler à l'utilisateur que les différents critères de sa demande sont incompatibles, ou que sa demande est incompatible avec les connaissances du système. Les conflits non résolus devront interpeller le concepteur de l'application pour qu'éventuellement il remette en cause certains choix.

5.10 Caractérisation de la stratégie de recherche

Muriel Cluzeau-Ciry a suggéré que le type de besoin détermine en grande partie la stratégie de recherche. Toutefois, il est difficile de trouver des critères objectifs pour déterminer de fa-

çon automatique quel est le type du besoin d'information de l'utilisateur. Parmi les critères de caractérisation du besoin avancés dans cette typologie, nous trouvons :

- la **présence de noms propres** dans la requête : la demande «oiseaux, Alfred Hitchcock» peut être interprétée comme «*Les Oiseaux*» *d'Alfred Hitchcock*, demande précise puisque le cinéaste anglais a réalisé un film portant ce titre, ou comme *les oiseaux chez Alfred Hitchcock*, demande thématique où l'utilisateur cherche toutes les apparitions d'oiseaux dans ses films. La première interprétation fait intervenir deux «noms propres» (le titre étant considéré comme un nom propre) dans la même facette SIGNALÉTIQUE. La seconde interprétation fait intervenir un nom propre et un nom commun dans deux facettes différentes. Plus la requête comporte de noms propres, plus le besoin semble donc précis.

- le **niveau de généralité des concepts** de la requête : plus les concepts sont généraux, moins la demande est précise.

- le fait que **peu d'attributs différents d'une même facette** soient impliqués dans la requête initiale peut permettre de déceler une demande thématique : seul le sujet, l'objet, le lieu ou la période de description du contenu est renseignée par exemple.

- la **présence de termes connotatifs** dans la requête ; la présence de termes liés aux impressions ou aux émotions (ou plus généralement à une interprétation subjective) dans la requête permettent de déceler une demande connotative.

Ces critères ne permettent pas de déterminer une stratégie de recherche complète pour chaque type de besoin ; il s'agit plutôt d'hypothèses de travail qui pourront être analysées lors des utilisations du système. D'autres chercheurs ont déjà échoué à la mise en évidence de critères pour la sélection automatique de stratégies de recherche d'information [Croft 84, McCall 86], mais les critères qui ont été analysés étaient moins liés à la sémantique que ceux que nous proposons d'étudier ; il s'agissait par exemple du nombre de termes dans la requête ou de la fréquence d'occurrence des termes dans la collection. Le tableau suivant résume les critères qui peuvent être analysés pour définir le type de la demande :

Type de besoin	Critère à analyser
Besoin précis	Niveau de spécificité des concepts Fréquence faible des termes de la requête dans la collection Grand nombre d'attributs valués par facette Grand nombre de noms propres dans la requête Grand nombre de termes dans la requête
Besoin thématique	Niveau intermédiaire de généralité des concepts, Peu d'attributs différents valués dans la facette CONTENU,
Besoin connotatif	Présence de termes dans la facette CONNOTATION

Si le critère observé ne correspond pas au critère attendu, le système pourra suggérer à l'utilisateur de modifier la formulation de sa requête, ou remettre en cause ses propres critères. Un

comportement est préconisé dans chaque situation :

Type de besoin	Comportement préconisé
Besoin précis	Éviter de généraliser les termes Favoriser une présélection stricte Favoriser la convergence du bouclage de pertinence : - importance forte de l'ancienne requête, - rendre plus difficile l'introduction de nouvelles facettes, en élevant le seuil de pertinence minimum pour une facette, ou en distribuant très peu de voix à de nouvelles facettes.
Besoin thématique	Favoriser une présélection intermédiaire Trouver de nouveaux indices dans la même facette grâce au bouclage de pertinence Exploiter les liens d'association au sein de la facette
Besoin exploratoire	Exploiter le modèle de description (facettes, attributs) pour suggérer des pistes Favoriser une présélection large Générer un hypergraphe pour les résultats pour permettre la navigation Permettre à l'utilisateur de revenir sur ses traces en organisant des points de reprises dans son historique. Trouver de nouveaux indices grâce au bouclage de pertinence : - importance faible de l'ancienne requête, - exploitation des liens d'association entre termes.
Besoin connotatif	Exploiter le bouclage de pertinence pour trouver des termes pertinents dans les facettes autres que CONNOTATION Exploiter les liens d'association menant des termes de la facette CONNOTATION à d'autres facettes

Ce tableau montre que la répartition par facette des connaissances sur les documents peut être utilement exploitée pour définir le comportement du système.

Nous pouvons observer qu'une demande précise est opposée à une demande exploratoire pour le nombre de documents qu'elles doivent rapporter ; une demande thématique est opposée à une demande précise par le nombre de facettes et d'attributs au sein d'une facette qui sont mis en œuvre simultanément ; mais une demande connotative peut être précise, thématique ou exploratoire, selon que le concept subjectif se rapporte au document global ou à un objet représenté dans le document. Par exemple, un utilisateur peut rechercher des séquences vidéo gaies (demande exploratoire), des documents où les personnes sont joyeuses (demande thématique), ou des documents où Jacques Chirac est joyeux (demande précise).

Par rapport à la stratégie mise en œuvre dans chacune de ces situations, nous choisissons de considérer plusieurs critères distincts :

1. le type de besoin de l'utilisateur : précis, thématique, ou exploratoire ; un besoin thématique correspond en effet à un traitement intermédiaire entre le traitement d'un besoin précis et celui d'un besoin exploratoire. C'est l'utilisateur qui indique explicitement quel est le type de son besoin.

2. le nombre de documents que souhaite retrouver l'utilisateur, avec un nombre fixé par défaut par type de média ;

3. le temps que l'utilisateur accepte de passer à sa recherche, une valeur par défaut lui étant suggérée (par exemple, la moyenne des temps de recherche de sessions passées pour le même type de besoin). L'utilisateur n'indiquera ce critère que s'il est crucial dans sa situation.

La demande sera considérée comme connotative si un des termes de la requête appartient à la facette CONNOTATION du thésaurus.

Les tactiques suggérées pour un besoin précis et un besoin connotatif sont contradictoires si elles sont appliquées uniformément à toutes les facettes ; en fait, la tactique mise en œuvre pour un besoin connotatif ne concerne que la facette CONNOTATION. Les autres facettes conservent la stratégie préconisée par défaut selon le type de besoin.

Une autre question se pose : faut-il considérer que le type du besoin est constant au cours d'une même session de recherche ? Une réponse positive simplifie le choix de comportement du système, mais ne correspond pas forcément à la réalité : en effet, un utilisateur peut commencer par explorer la collection, puis émettre un besoin précis sur un sous-ensemble de cette collection. Il peut au contraire avoir un besoin précis, et trouver des documents qui l'incitent à une exploration. Nous proposons donc que l'utilisateur puisse indiquer un changement dans le type de son besoin au cours d'une session. Ceci provoquera le marquage de la trace par un signet particulier, et réinitialisera le nombre de voix affecté à chaque agent de mise en correspondance pour son vote. Diverses stratégies pourront ainsi être mises en œuvres au cours d'une même session, l'utilisateur partant d'un sous-ensemble de documents qu'il aura présélectionnés par ses actions précédentes.

5.11 Conduite du processus de recherche

5.11.1 Les cycles du processus de recherche

Dans un processus de recherche interactive tel que nous l'avons analysé, nous pouvons identifier deux types principaux de cycles d'actions pour le comportement du système :

1. interrogation initiale par requête
   (a) avec prise en compte de l'ensemble de la collection
   (b) avec reprise d'un résultat obtenu lors d'une session précédente (réutilisation d'un sous-ensemble de documents résultats obtenu lors d'une session précédente)

Dans ces deux cas, la conduite du processus est la suivante :

- formulation de la requête

- interprétation sémantique
- proaction
- mise en correspondance
- visualisation et choix
- reformulation sur les concepts
- retour à la formulation.

2. interrogation initiale par exploration

   (a) les documents proposés sont obtenus par tirage aléatoire ou on utilise un sous-ensemble représentatif de concepts puis de documents
   - visualisation et choix
   - analyse des descriptions textuelles et reformulation sur les concepts
   - contrôle éventuel par l'utilisateur de la requête reformulée
   - proaction
   - mise en correspondance
   - retour à la visualisation

   (b) les documents proposés sont obtenus par tirage aléatoire ou on utilise un sous-ensemble représentatif de documents
   - visualisation et choix
   - analyse du (ou des) document(s) choisi(s) par algorithmes de reconnaissances de formes, d'analyse de couleurs ou de textures
   - recherche de documents ayant les mêmes caractéristiques
   - retour à la visualisation

Les alternatives possibles dans le contrôle sont déterminées par la collection initiale de documents considérée : collection globale pour une RI classique, collection résultat obtenue lors d'une session précédente pour l'affinement d'une recherche, tirage aléatoire ou utilisation d'un sous-ensemble représentatif de documents pour une exploration. L'ordre d'intervention des différentes phases dépend ensuite uniquement de la présence ou de l'absence d'une requête initiale, et du type de reformulation choisie par l'utilisateur (exploitation des concepts et/ou des relations entre eux, ou des caractéristiques physiques des images : couleur, formes, texture). Ce qui peut être modifié, c'est par exemple le nombre moyen de cycles de recherches (nombre d'étapes dans une session) au bout duquel le système intervient pour signaler à l'utilisateur l'inefficacité de sa recherche, et lui proposer de modifier manuellement sa requête.

Par contre, la flexibilité de la résolution provient du fait que des agents de résolution différents peuvent intervenir dans diverses situations de recherche. Si l'utilisateur est pressé, le type de traitement le plus rapide sera privilégié. S'il ne l'est pas, une mise en correspondance plus détaillée sera mise en œuvre.

Une flexibilité qui peut être introduite dans la résolution de problème est l'intervention des agents-facettes au cours de la résolution. La présence de concepts appartenant à des facettes données déclenche l'intervention des agents-facettes correspondant pour la résolution, mais comment décider si d'autres doivent intervenir ?

5.11.2 Critères sociaux de choix d'un agent pour la résolution

Pour déterminer quels sont les agents qui vont intervenir dans la résolution du problème courant, le coordinateur doit définir des critères d'évaluation. Les stratégies suggérées par la typologie des demandes permettent de définir quels sont les agents détenant un savoir-faire particulier qui vont intervenir. Pour affiner ces stratégies, il est possible de définir les conditions dans lesquelles les agents-facettes, outre ceux impliqués dans la requête initiale, vont entrer dans la résolution à un moment donné du processus de recherche.

Nous proposons de doter chaque agent-facette d'un intérêt propre : chaque agent dispose, au départ, d'un *capital de crédibilité* vis-à-vis du système. Le but propre d'un agent est de maximiser son capital de crédibilité. Lorsque l'utilisateur émet des jugements de pertinence sur les documents, le capital de crédibilité de chaque agent-facette augmente au prorata des documents qu'il a proposés et qui ont été jugés pertinents par l'utilisateur. Ce capital diminue au prorata des documents qu'il a proposés et qui soit n'ont pas été retenus par l'utilisateur, soit ont été rejetés. Ces trois règles permettent d'éviter une pléthore d'offres inopportunes tout en stimulant les propositions.

Lorsque l'utilisateur s'est prononcé sur la pertinence des documents que le système lui a proposés, chaque agent-facette analyse ses propositions par rapport aux documents pertinents et de documents non pertinents. Les agents de mise en correspondance qui ont proposé des documents jugés pertinents sont récompensés : leur capital augmente de façon proportionnelle au nombre de documents pertinents qu'ils ont suggérés. Les agents qui ont proposé des documents jugés non pertinents par l'utilisateur sont pénalisés : ils paient une amende pour chaque document non pertinent qu'ils ont proposé. Pour les documents indifférents, il est difficile d'établir s'il s'agit de documents sur lesquels l'utilisateur hésite à se prononcer ou s'il s'agit simplement d'un comportement «paresseux» de l'utilisateur, qui veut économiser ses gestes et ses analyses. Ici, tout dépend de la façon dont les documents sont présentés. S'ils sont présentés grâce à un outil de visualisation particulier et de façon indépendante, il peut être intéressant d'enregistrer l'identification des documents effectivement regardés par l'utilisateur, et seuls ceux-ci participeront à la diminution du capital d'une facette donnée. Lors de l'étape suivante de résolution de problème, les agents les plus crédibles interviennent en priorité, par exemple en ayant davantage de voix à attribuer aux documents lors de la mise en correspondance.

Avec ce système, chaque agent-facette est doté d'un but propre : intervenir de façon la plus opportune possible pour maximiser son capital individuel.

Comment gérer la coopération grâce à ce principe ? Il doit être possible à certains agents-facette de faire des coalitions : ils mettent ensemble une partie ou la totalité de leur capital et partagent équitablement les gains et les pertes. Ainsi, les agents sont ammenés à coopérer lorsque cette coopération est réellement utile, et dans le but de maximiser leurs propres gains. À l'occasion de ces coopérations, les agents-facettes peuvent chercher à renforcer leurs connaissances sur les autres agents : avec quel agent est-il plus efficace de s'allier dans telle situation ? Ceci est une forme d'apprentissage social qui peut être étudiée sur un tel système.

De nombreuses autres configurations peuvent être envisagées.

5.11.3 Architecture globale du système

En résumé, l'architecture du système de recherche d'information que nous proposons comprend les composants suivants :

- les agents qui gèrent la représentation de la connaissance : agents de gestion du thésaurus, agents d'indexation, agents de formalisation de la requête ;
- les agents qui disposent d'un savoir-faire pour la recherche d'information ; il s'agit des agents de présélection des documents, de mise en correspondance et de reformulation. Le rôle de ces agents varie en fonction de la situation de recherche ;
- l'agent qui gère la trace d'une session de recherche : enregistrement de résultats partiels, positionnement de signets et de points de reprise, transmission d'informations sur la stratégie utilisée aux étapes précédentes;
- l'agent qui gère les votes pour la mise en correspondance : attribution d'un nombre de voix aux facettes, rassemblement et synthèse des votes;
- les agents qui permettent l'évaluation au sein du système : évaluation de la satisfaction de l'utilisateur, des performances de la recherche, de l'adéquation de la stratégie choisie à la situation ;
- les agents de présentation des résultats (liste ordonnée de documents retrouvés, descriptions de ces documents, requête reformulée, graphe hypermédia) ;
- l'interface, qui gère les communications entre l'utilisateur et le SRI ;
- les outils de visualisation des documents ;
- l'application, qui répertorie les facettes de description et les agents disponibles (booléens, vectoriels, graphes conceptuels). Elle joue le rôle de conteneur de méta-connaissances pour les différents agents de recherche.

Cette architecture peut être complétée par des agents qui permettent au SRI d'améliorer à long terme son fonctionnement. L'apprentissage à long terme peut permettre de déterminer les stratégies les plus adaptées en fonction d'une situation de recherche donnée. Il donne également la possibilité d'évaluer l'indexation qui a été faite des documents : le niveau de granularité des descriptions est-il suffisant ? les concepts d'indexation ont-ils été bien choisis ? Enfin, il donne un retour au concepteur du système : les performances des agents sont-elles acceptables, en qualité et en temps ? Pour cela, nous pourrions adapter les propositions de Malika Smaïl [Smaïl 93b] d'utiliser le raisonnement à base de cas sur une mémoire de sessions de recherche.

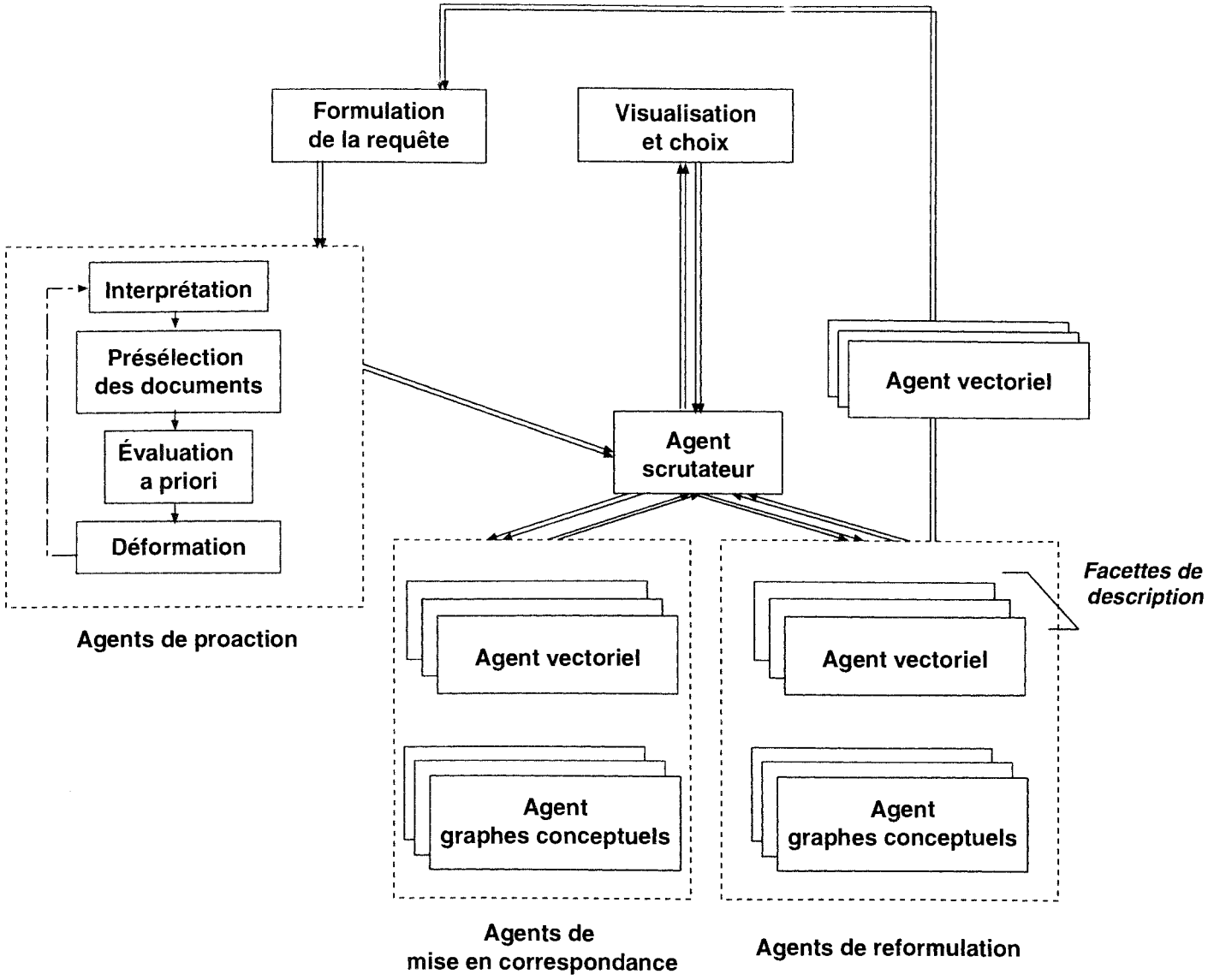


FIG. 5.9 – *Architecture générale du SRI*

5.12 Alternatives possibles pour l'organisation des agents

Les concepts développés autour des systèmes multi-agents ne permettent pas de déterminer une architecture unique et optimale. Ainsi, nous avons envisagé différentes solutions avant d'opérer un choix. Nous disons quelques mots des autres solutions envisagées.

5.12.1 Interactions plus fortes entre les agents de mise en correspondance

Les agents de mise en correspondance travaillant sur des facettes différentes auraient pu communiquer directement entre eux des résultats partiels : lors du premier cycle de recherche, les facettes pondérées le plus fortement dirigent la recherche ; elles consultent les facettes pondérées le plus faiblement en cas de conflit. Ceci supprime la phase de synthèse globale ... mais oblige chaque agent de mise en correspondance à disposer d'une fonction de synthèse et multiplie les communications pendant la mise en correspondance. Ceci introduit également des dépendances supplémentaires entre les agents-facettes.

5.12.2 Granularité des agents

Avant d'obtenir le découpage fonctionnel que nous venons de présenter, nous nous sommes interrogée sur le bon niveau de granularité à retenir pour les agents. Nous avons réfléchi à une solution de très fine granularité pour les agents : les concepts. La requête pouvait alors être vue comme un ensemble d'agents envoyés au SRI pour chercher des documents pertinents. Ce modèle devenait très proche d'un modèle connexionniste, sans en avoir forcément le fonctionnement. Ce niveau de granularité nous a paru trop fin, car il maximise les communications dans le système ; il est néanmoins intéressant intellectuellement. Il permet en effet d'analyser les relations entre les concepts sous la forme d'émergence de relations sociales entre agents-concepts : relations entre les concepts qui indexent les mêmes documents, entre ceux qui sont simultanément choisis ou rejetés, etc.

En fait, trouver le bon niveau de granularité consiste à déterminer à quel niveau il est préférable de s'arrêter de diviser un agent en société d'agents. Si l'on souhaite une société d'agents *cognitifs*, ils doivent conserver une certaine complexité, disposer d'un ensemble non négligeable de connaissances et donc être plutôt «gros grain». Si l'on souhaite au contraire un système purement réactif, où les actions ne sont que des réactions à des événements, chacun des agents peuvent avoir un comportement très simple et des connaissances très limitées : ce sont des agents plutôt «grain fin».

5.13 Résumé

Nous avons adopté un double découpage des compétences dans le SRI multi-agent : le premier s'appuie sur la distribution logique des connaissances – c'est la notion de facette (figure

5.10) – et le second, fonctionnel, s'appuie sur le savoir-faire du système (figure 5.11). Nous

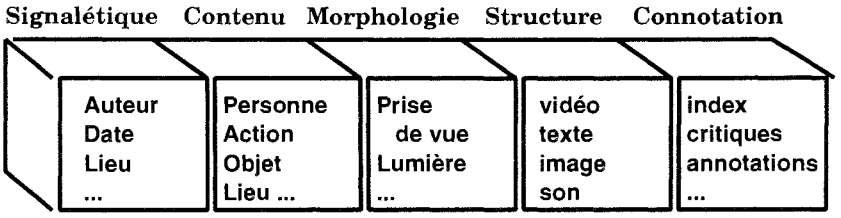


FIG. 5.10 – *La partition des connaissances du SRI*

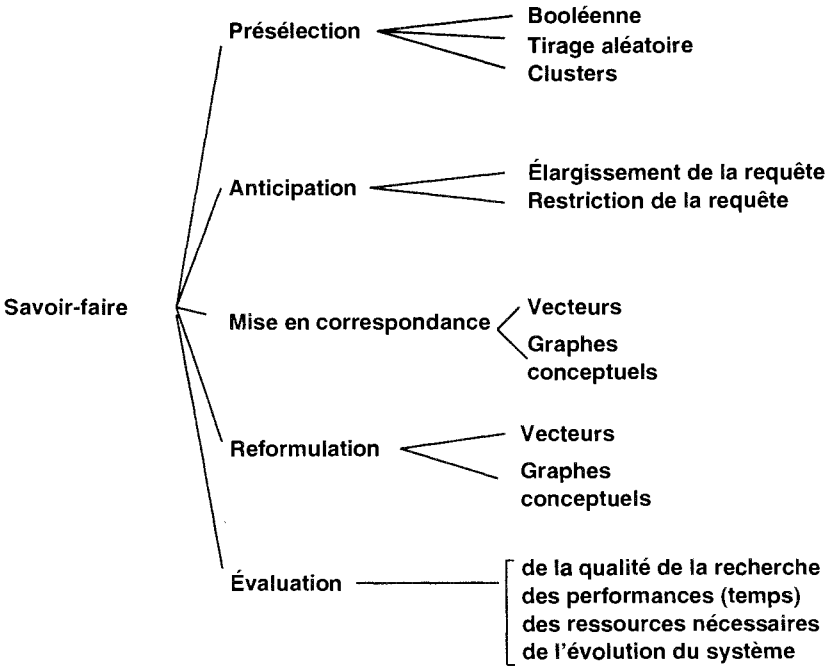


FIG. 5.11 – *La diversité des savoir-faire du SRI*

avons défini les principaux agents qui composent la société de recherche d'information. Nous avons mis en évidence leurs buts et les dépendances qui existent entre eux. La conduite d'une recherche n'est plus un processus totalement prédéfini, mais une coopération à laquelle participent différents agents de façon opportuniste. Nous allons développer dans le chapitre 6 la façon dont l'intervention des agents peut varier au cours d'une session de recherche.

L'amélioration apportée par des représentations multiples d'une requête ou d'une combinaison de méthodes de recherche a déjà été constatée par exemple dans [Belkin 93]. L'originalité de notre proposition est de ne pas imposer au chercheur d'information une formulation de sa requête sous des formes multiples. Par ailleurs, la combinaison de différentes méthodes de recherche semble également améliorer la qualité de la recherche [Belkin 95].

Il est clair que le fonctionnement optimal d'un tel système ne peut être complètement défini à l'avance. S'il est possible de déterminer intuitivement l'apport spécifique de chaque agent pour

une recherche d'information, cet aspect mérite d'être étudié sur des cas réels. Doter les agents de capacités d'apprentissage à long terme pourrait permettre d'améliorer le fonctionnement du système. Les travaux de Malika Smaïl, intégrant le raisonnement à base de cas à un processus de RI, nous semblent donc particulièrement applicables à ce nouveau contexte de système multi-agents pour que chaque agent apprenne quelles sont les situations où il peut intervenir avec profit.

Chapitre 6

Le modèle de système multi-agent retenu et sa mise en œuvre

Dans le chapitre précédent, nous avons analysé les principaux concepts qui émergent des travaux de l'intelligence artificielle distribuée. Nous avons fait des propositions pour la modélisation multi-agents d'un SRI. Nous avons en particulier détaillé les fonctions des principaux agents et les connaissances qu'ils utilisent. Le fonctionnement de la société d'agents repose sur le principe de la composition/décomposition d'agents qui participent à l'élection des documents susceptibles d'être pertinents. Dans ce chapitre, nous allons définir plus précisément le modèle d'agent que nous retenons. Nous distinguons les agents cognitifs *des* agents de service *: les premiers sont amenés à prendre des décisions, à proagir et à établir une stratégie alors que les seconds répondent simplement à des sollicitations de façon réactive. Nous énonçons les critères retenus pour le choix des agents qui participent à une session de recherche d'information. Nous présentons ensuite le prototype réalisé qui met en œuvre les idées développées, son architecture et un scénario standard pour une session de recherche d'information. Enfin, nous émettons des propositions pour son évaluation.*

6.1 Typologie des agents du SRI

Le modèle de SRI multi-agents que nous proposons comporte trois types principaux d'agents cognitifs au sens de [Dieng 94] : les agents de recherche de documents, les agents de reformulation de la requête, les agents qui permettent à l'organisation de fonctionner en assurant des tâches de coordination. Il comporte également un certain nombre d'*agents de service*, qui réagissent simplement à des demandes.

6.1.1 Les agents de recherche de documents

Un agent de recherche de document est un agent composite au sens où il doit réaliser plusieurs types de tâches pour accomplir son but. Il peut avoir deux rôles distincts dans le système :

présélectionner simplement un ensemble de documents susceptibles d'être pertinents ou classer les documents retrouvés. Le rôle qui lui est affecté détermine son plan d'action.

Il a recours à plusieurs *agents de services* pour formaliser la requête (c'est-à-dire la représenter sous la forme qu'il souhaite traiter), calculer une mesure de similarité entre des documents et la requête, préparer les résultats de la mise en correspondance en classant ou non les documents, et en leur affectant un nombre de voix, et évaluer les résultats en fonction des jugements de l'utilisateur.

Ce type d'agent dispose

- de connaissances sur le modèle de description des documents qu'il peut exploiter,
- de connaissances sur le besoin d'information de l'utilisateur sous forme d'une requête interprétée qu'il va formaliser, de façon à la faire correspondre au modèle dont il a besoin pour effectuer ses calculs.

Il a un pouvoir de décision sur le type de représentation des documents sur lequel il va travailler. Il obtient des connaissances sur les domaines sémantiques auxquels il s'intéresse, en communiquant avec le thésaurus qui lui transmet des informations sur les concepts qu'il traite. La stratégie qu'il met en œuvre dépend essentiellement du type de besoin de l'utilisateur, et du domaine sémantique des concepts auxquels il s'intéresse dans la requête. Nous détaillerons plus loin les caractéristiques qui sont liées aux domaines sémantiques.

Pour résumer, un agent de recherche est décrit par

- le **rôle** qu'il peut adopter (présélectionner, classer) dans une session de recherche,
- une **portée**, qui représente la proportion de requête qu'il traite : mono-facette ou multi-facettes, nombre de concepts de recherche pris en compte,
- un **plan d'action**, qu'il construit en fonction du rôle qu'il doit jouer : obtenir la requête dans un format adapté au modèle de représentation choisi, obtenir des connaissances supplémentaires sur les concepts traités (en consultant le thésaurus par exemple), calculer la similarité entre la (sous-)requête et les descriptions de documents – exhaustivité, spécificité, similarité, prise en compte de la co-taxonomie des termes –, classer les documents par ordre de similarité décroissante avant d'effectuer son vote, et évaluer sa performance en confrontant ses résultats avec les jugements de l'utilisateur. Ce plan représente l'imbrication des actions des agents de service qui sont sollicités pour réaliser chacun des sous-buts nécessaires. Ce plan est élaboré selon une stratégie qui dépend en grande partie du type de besoin d'information de l'utilisateur.
- ses accointances, c'est à dire la liste des agents de services auxquels il peut sous-traiter une partie des opérations, ou qu'il peut solliciter pour obtenir des connaissances qui l'aideront à résoudre le problème.
- la liste de décisions qu'il prend au fur et à mesure de la session de recherche, et des raisons qui l'ont poussé à faire ces choix.

Selon ce schéma, les agents de recherche n'agissent pas directement les uns sur les autres pendant la recherche. Il sont mutuellement indépendants pour constituer leur présélection ou leur classement. Cependant, à partir de la deuxième étape de recherche, ils possèdent des sortes de croyances sur leur utilité dans la recherche (ces croyances sont quantitatives dans le prototype implanté). Ils sont en effet ammenés à s'auto-évaluer et donc à décider ou non de participer à la suite de la session. Ce faisant, s'ils abandonnent leur recherche, ils doivent le signaler de façon à ce que les concepts corespondants soient malgré tout traités sous une autre forme. Ainsi, un agent de recherche travaillant sur une seule facette, et qui abandonne sa recherche, devra le signaler au scrutateur (agent de coordination que nous présenterons au § 6.1.4) de façon à ce que celui-ci premièrement n'attende pas son vote et ensuite essaie de composer un autre agent de recherche pour traiter des concepts ainsi délaissés.

6.1.2 Les agents de reformulation de la requête

Le but d'un agent de reformulation de la requête est de trouver une meilleure expression du besoin d'information de l'utilisateur, à partir des jugements émis par le chercheur d'information sur les documents que lui présente le système à chaque étape, et de connaissances sémantiques sur les concepts de la requête. Il communique avec le thésaurus pour obtenir des connaissances sur les concepts du domaine.

Un agent de reformulation de la requête a deux composants essentiels : l'agent qui recherche une meilleure expression de la requête et l'agent qui évalue l'apport du bouclage de pertinence. Son **rôle** est de trouver une expression de la requête dans un formalisme particulier (vecteur pondéré de termes, ensemble de graphes conceptuels pondérés). Il communique avec le scrutateur pour obtenir les jugements de l'utilisateur, et en fonction du type de reformulation mise en œuvre, avec le thésaurus.

6.1.3 Les agents de service

Les agents de service sont ceux qui effectuent des tâches simples à la demande des autres agents de l'architecture. Leur comportement est invariant, et ils n'agissent qu'à la demande d'autres agents.

Nous classons dans cette catégorie, notamment :

- les agents d'interprétation sémantique de la requête; dans notre application, ces agents utilisent le thésaurus : recherche du domaine d'un terme, d'un terme générique, des spécifiques d'un terme, etc.

- les agents qui recherchent à quelle facette un concept peut être attaché.

- ceux qui calculent une mesure de similarité entre documents et requête, et ceux qui calculent le nouveau poids d'un concept (simple ou complexe).

- les agents de tri : tri des documents par ordre de similarité décroissante avec la requête, tri des descriptions par concepts, etc.
- les agents qui permettent d'accéder aux listes inverses et autres structures de données chargées de rendre la recherche plus rapide.

6.1.4 Les agents de coordination de l'organisation

Ces agents n'ont aucune connaissance sémantique sur la tâche à accomplir. Ce sont des agents qui assurent une certaine complétude de la solution, en vérifiant que tous les aspects de la requête ont été étudiés par au moins un agent, et qui recueillent les résultats des agents coopérant pour la recherche et les synthétisent. Ainsi, un agent *scrutateur* connaît les différents concepts de la requête à chaque étape, et la liste des agents qui s'intéressent à chacun d'entre eux. Il s'assure que les différents agents de recherche ont fourni leur vote avant de synthétiser leurs résultats. C'est également lui qui analyse les résultats des élections de documents, choisit le mode de scrutin (dont nous reparlerons au §6.5.2) et transmet les résultats à l'agent de préparation de la visualisation.

La solution que nous avons adoptée, qui consiste à coordonner explicitement les résultats grâce à un agent *scrutateur*, simplifie l'organisation du système. Une organisation alternative aurait consisté à doter chaque agent de recherche de capacité de communication avec ses pairs.

6.2 Scénario général d'une session de recherche d'information

L'architecture du système pour la recherche proprement dite des documents est représentée en figure 6.1.

Afin de fixer les idées du lecteur, nous donnons ci-après le scénario général d'une session de recherche d'information. Lorsqu'une requête formulée par un utilisateur arrive au SRI, elle est analysée par un agent d'accueil qui la transforme en un ensemble de sous-requêtes, chaque sous-requête correspondant à un concept et étant susceptible de concerner un ou plusieurs agents de recherche. L'agent d'accueil sollicite l'agent de présélection de documents qui recherche les documents susceptibles d'être pertinents pour la requête.

La composition de l'équipe des agents de recherche dépend à la fois des concepts de la requête et du type de besoin. Si plusieurs concepts de la requête sont relatifs à une même facette de description, un agent de recherche est créé pour la facette. Si la requête ne comporte qu'un concept par facette, un agent de recherche inter-facette est mis en œuvre.

L'agent de présélection contacte l'agent d'*évaluation a priori* pour lui transmettre ses résultats. Ce dernier prend l'initiative de demander ou non une déformation de la requête. Lorsque l'ensemble de documents présélectionnés a une taille satisfaisante, il est mis à la disposition des agents de recherche. Soit un seul ensemble de documents est constitué, qui va être traité par

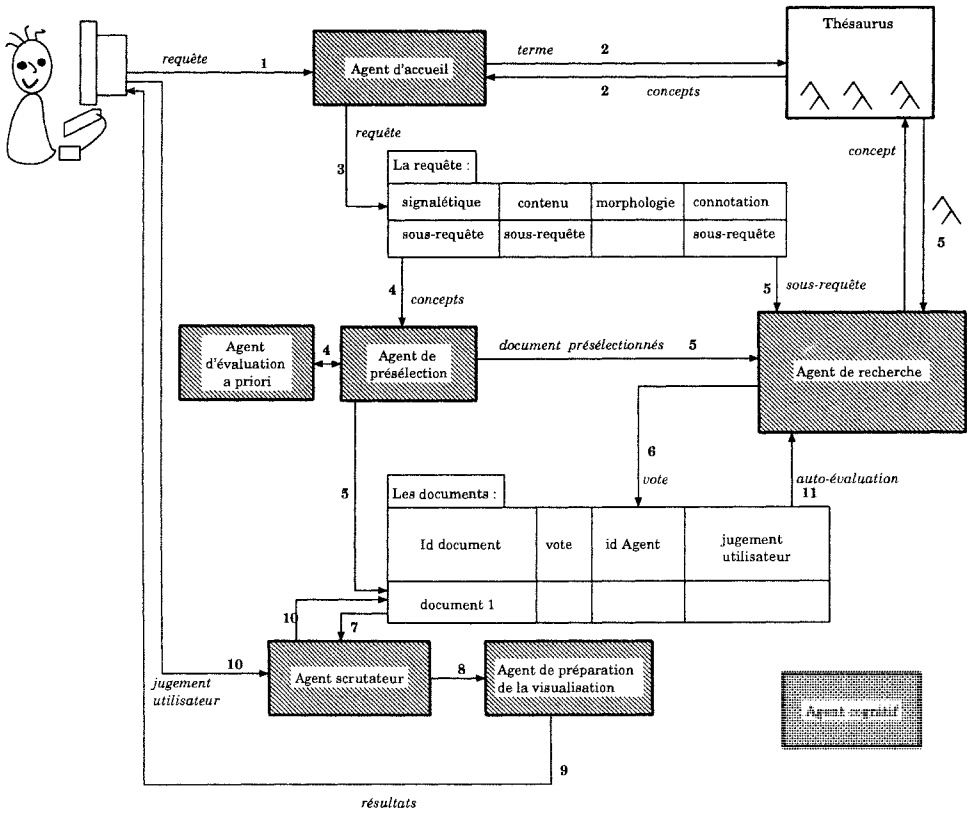


FIG. 6.1 – *Architecture du SRI multi-agents pour la recherche des documents.*

tous les agents de recherche; cette situation se produit lorsque les documents cibles sont structurés et que les concepts sélectionnés par l'utilisateur sont attachés à des niveaux de structure différents des documents. Soit plusieurs ensembles sont constitués : chaque concept rapporte un ensemble de documents et les agents de recherche constituent eux-mêmes les sous-ensembles de documents sur lesquels ils vont travailler; cette solution est adoptée lorsque les concepts de la requête sont liés à un même niveau de structure du document, mais pour des facettes différentes.

Lorsque les documents sur lesquels porte la recherche ne sont pas structurés, les critères de choix des ensembles de documents présélectionnés pour chaque agent de recherche sont légèrement différents. Dans le cas d'un besoin précis, les agents de recherche travaillent sur tous les documents rapportés par tous les concepts. Pour une demande exploratoire, les agents de recherche sont constitués par facette, et travaillent sur les seuls documents rapportés par les concepts de la facette. Si la demande est thématique, les agents de recherche sont également constitués par facette et travaillent sur un ensemble de documents communs avec le ou les agents avec lesquels ils ont le plus d'affinités.

Comment déterminer les affinités entre agents de recherche ? Une heuristique consiste à les mesurer à travers le nombre de liens *termes associés* entre les concepts des domaines du thésaurus. Le nombre de ces liens, qui sont en général très dépendants du domaine d'application, reflètent en effet dans une certaine mesure le degré de proximité entre les domaines sémantiques. Une autre façon de mesurer ces affinités serait d'exploiter une mémoire des sessions de recherche pour y étudier les facettes de descriptions qui travaillent le plus fréquemment conjointement dans une résolution.

L'évaluation a priori porte, pour le nombre minimum et maximum de documents, sur la taille du plus grand ensemble de documents ainsi constitué.

Chaque agent de recherche choisit un mode de mise en correspondance entre la requête et les documents. Il obtient, grâce aux agent de service, le calcul d'une mesure de similarité pour chaque document de l'ensemble présélectionné, et un nombre de voix qu'il affecte à chaque document.

Un agent *scrutateur* recueille les votes, et classe les documents en fonction du nombre de voix qu'ils ont obtenus, et du type de scrutin choisi. Le scrutin peut être au choix de type :

- majorité absolue – seuls les documents jugés pertinents par au moins la moitié des agents de recherche sont retenus –,
- majorité relative (appelée aussi majorité simple) où sont retenus les documents qui obtiennent le plus grand nombre de voix,
- proportionnel, où les documents les mieux classés par chaque agent sont présentés à l'utilisateur; ce type de scrutin permet de présenter un document retrouvé par un agent isolé et non par la majorité des agents.

Nous reviendrons au § 6.5.2 sur le choix d'un mode de scrutin.

Pour les documents structurés, les agents de recherche adaptent leurs résultats au niveau de structure retenu pour la présentation : niveau par défaut ou niveau spécifié par l'utilisateur. Le

scrutateur transmet le résultat du scrutin à l'agent chargé de présenter les résultats.

L'interface se charge de la présentation des documents et de leurs descriptions, et recueille les jugements de l'utilisateur qu'elle transmet à la société d'agents qui ont participé à la recherche. L'agent scrutateur, qui sait quels sont les agents qui ont participé à la recherche, leur communique les évaluations de l'utilisateur. Les agents de recherche analysent les jugements portés par l'utilisateur pour s'auto-évaluer (voir § 6.4.1).

La figure 6.2 montre l'architecture nécessaire à la reformulation de la requête. Parallèlement

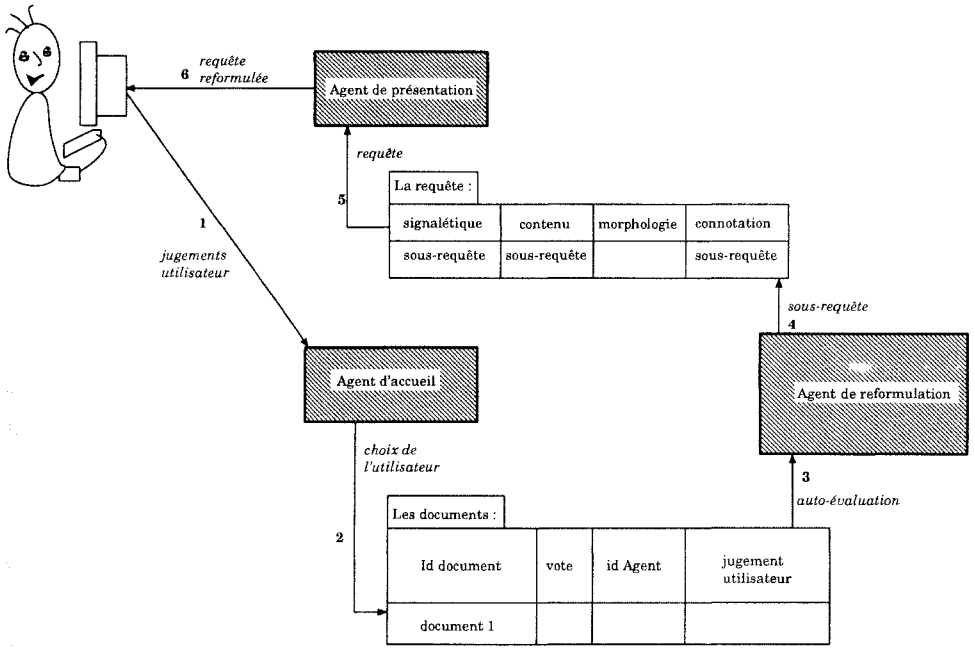


FIG. 6.2 – *Architecture du SRI multi-agents pour la reformulation.*

à l'auto-évaluation des agents de recherche, les agents de reformulation de la requête exploitent les descriptions des documents choisis ou rejetés par l'utilisateur pour essayer d'obtenir une meilleure expression de la requête. Ils exploitent également ces jugements portés par le chercheur d'information pour évaluer leur capacité à reformuler la requête de façon appropriée (voir § 6.4.2). Dans le cas où la requête reformulée est présentée à l'utilisateur pour contrôle et modification éventuelle, les modifications qu'il apporte devraient pouvoir également être analysées par ces agents. Ceci n'a pas été réalisé dans le système implanté, mais est envisageable.

6.3 Critères de décision pour la composition/décomposition des agents

Les critères de décision de composition/décomposition des agents de recherche pour la recherche des documents sont de trois ordres. La **répartition des concepts de la requête parmi les facettes** détermine les agents de mise en correspondance qui seront créés. Si une facette

comporte plus de deux concepts de la requête, un agent de mise en correspondance est mis en œuvre pour cette facette. En complément, un agent vectoriel est créé pour traiter ensemble les concepts de la requête qui sont isolés dans une facette.

Le **type de besoin** détermine les paramètres stratégiques du processus et les méthodes de mise en correspondance qui seront utilisées : modèle vectoriel pour une demande exploratoire ou thématique, modèle de graphes conceptuels pour une demande précise. En fonction du type de besoin et de ses résultats, un agent de recherche d'information est plus ou moins complexe. En effet, si les résultats obtenus ne sont pas suffisamment discriminants, l'agent de recherche peut mettre en œuvre une autre méthode de calcul de similarité entre les documents retrouvés pour les départager.

Les **ressources disponibles** déterminent également la composition des agents de recherche. Il s'agit du type de descriptions qui existent pour une facette (graphes conceptuels ou simples mots-clés), des connaissances qui sont disponibles sur le domaine sémantique (thésaurus) et du temps dont l'utilisateur dispose pour sa recherche : s'il a peu de temps, les méthodes les plus rapides seront privilégiées.

Une session de recherche interactive comportant plusieurs étapes, la composition de l'équipe d'agents qui intervient à chaque étape peut changer, en fonction des résultats de l'évaluation.

6.4 Évaluation des résultats d'un agent

Lors d'une session de recherche d'information, les agents du SRI cherchent à apprendre quelle est la meilleure stratégie de recherche qu'ils doivent mettre en œuvre. Leur stratégie repose sur des fonctions d'évaluation qui leur permettent d'ajuster leur comportement à la situation.

Pour cette évaluation, les agents exploitent l'interactivité de la recherche. Nous allons utiliser (voir figure 6.3) :

- l'ensemble de documents jugés pertinents par l'utilisateur, $Pertinent_{\mathcal{U}}$,
- l'ensemble de documents jugés non pertinents par l'utilisateur, $NonPertinent_{\mathcal{U}}$,
- l'ensemble de documents trouvés par l'agent $\mathcal{A}$, $Trouvés_{\mathcal{A}}$.

Au §5.11.2 page 198, nous avons émis le principe de moduler le capital de crédibilité des agents en fonction de leur «utilité» dans la recherche. Nous détaillons ci-après la façon dont nous proposons de mettre en œuvre ces propositions.

6.4.1 Auto-Évaluation des agents de mise en correspondance

Pour évaluer les résultats d'un agent de recherche pour la mise en correspondance, nous nous appuyons sur les deux mesures traditionnelles d'évaluation en recherche d'information : le rappel et la précision (nous avons présenté ces mesures au §1.4.2 page 24).

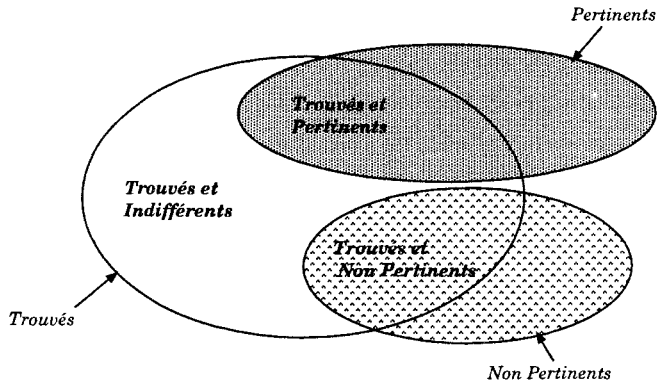


FIG. 6.3 – *Ensembles de documents après une mise en correspondance.*

Puisque nous sommes dans le cadre d'une recherche interactive d'information, nous utilisons la notion de **pertinence relative pour l'utilisateur**, qui correspond à ce que le SRI peut observer, c'est-à-dire aux jugements positifs que l'utilisateur porte sur les documents présentés par le système. Chaque agent de recherche peut évaluer sa **précision relative** par le nombre de documents qu'il a retrouvés et qui ont été jugés pertinents par l'utilisateur :

$$Précision_{\mathcal{A}} = \frac{|Pertinent_{\mathcal{U}} \cap Trouvés_{\mathcal{A}}|}{|Trouvés_{\mathcal{A}}|}$$

Pour tenir compte de la finesse du classement effectué par l'agent, il est possible non seulement de compter le nombre de documents, mais d'étudier les valeurs de similarité qui leur ont été affectées par l'agent :

$$PR_{\mathcal{A}} = \frac{\sum_{i=1}^{p_{\mathcal{A}}} msim_i}{\sum_{j=1}^{t_{\mathcal{A}}} msim_j}$$

où $msim_i$ représente la mesure de similarité affectée au document i, $p_{\mathcal{A}}$ représente le nombre de documents pertinents pour l'utilisateur parmi ceux proposés par l'agent $\mathcal{A}$, $t_{\mathcal{A}}$ représente le nombre de documents trouvés par l'agent éventuellement limité par le critère de censure.

Traditionnellement, le rappel représente la proportion de documents retrouvés parmi les documents pertinents de la collection. Cette mesure ne peut être calculée que si l'on connaît le nombre total de documents pertinents dans une collection par rapport à une requête. Or, cette information n'est pas disponible (si elle l'était, une recherche d'information serait triviale) : elle est en effet différente pour chaque utilisateur, en fonction de ce qui motive sa recherche d'information et de ses connaissances préalables. C'est pourquoi nous utilisons le rappel relatif, qui tient compte non pas de tous les documents potentiellement pertinents dans la collection, mais des documents que l'utilisateur a jugés pertinents pour sa recherche parmi ceux qui lui ont été présentés.

Chaque agent de recherche $\mathcal{A}$ calcule un **rappel relatif** qui correspond à la proportion de documents qu'il a retrouvés et qui ont été jugés pertinents par l'utilisateur par rapport au nombre global de documents qui ont été jugés pertinents :

$$RappelRel_{\mathcal{A}} = \frac{Pertinent_{\mathcal{U}} \cap Trouvés_{\mathcal{A}}}{Pertinent_{\mathcal{U}}}$$

Le rappel relatif permet d'estimer la proportion de solution qu'un agent a communiquée. De même que pour la précision, si nous voulons tenir compte des mesures attribuées à chaque document, nous pouvons utiliser le nombre de voix attribuées aux documents (les mesures de similarité attribuées par les autres agents ne sont pas connues de l'agent $\mathcal{A}$) :

$$RR_{\mathcal{A}} = \frac{\sum_{i=1}^{p_{\mathcal{A}}} nbv_i}{\sum_{j=1}^{p} nbv_j}$$

où p est le nombre total de documents jugés pertinents par l'utilisateur, nbv_i est le nombre de voix attribuées au $i$ème document, et $p_{\mathcal{A}}$ est le nombre de documents pertinents trouvés par l'agent $\mathcal{A}$.

Chaque agent évalue également ses **digressions** dans la recherche par la proportion de documents qu'il a trouvés et qui sont jugés non pertinents par l'utilisateur :

$$Digression_{\mathcal{A}} = \frac{|NonPertinent_{\mathcal{U}} \cap Trouvés_{\mathcal{A}}|}{|Trouvés_{\mathcal{A}}|}.$$

La digression d'un agent est moins grave si les documents non pertinents proposés sont en fin de classement. Pour moduler cette mesure, nous allons donc utiliser la mesure de **digression relative** suivante :

$$DR_{\mathcal{A}} = \frac{\sum_{i=1}^{np_{\mathcal{A}}} msim_i}{\sum_{j=1}^{t_{\mathcal{A}}} msim_j}$$

où $np_{\mathcal{A}}$ est le nombre de documents non pertinents retrouvés par l'agent $\mathcal{A}$ et $t_{\mathcal{A}}$ le nombre total de documents retrouvés par l'agent $\mathcal{A}$.

Un agent peut également évaluer son **silence** par la proportion de documents jugés pertinents par l'utilisateur et qu'il n'a pas retrouvés :

$$Silence_{\mathcal{A}} = 1 - RappelRel_{\mathcal{A}}$$

Un agent trop silencieux pourra être éliminé de l'équipe de recherche lors des étapes ultérieures en cas de rareté des ressources. Attention : l'élimination d'un agent de recherche correspond à l'élimination d'un mode de traitement des connaissances, et non à l'élimination des connaissances qu'il a traitées.

L'importance relative de ces mesures d'auto-évaluation d'un agent pour la recherche peut dépendre du type de besoin : pour un besoin précis, les digressions seront fortement pénalisées, alors qu'elles n'auront que peu d'importance pour un besoin thématique et pas d'importance pour un besoin exploratoire.

Ces différentes mesures peuvent être utilisées pour moduler le capital de crédibilité des agents au cours d'une session de recherche. Le capital de crédibilité CC peut ainsi être modifié d'étape en étape :

$$CC_i = CC_{i-1} * \frac{a \times PR + b \times RR - c \times DR}{a + b - c}$$

où

$$\begin{cases} i \text{ est l'étape de recherche.} \\ PR \text{ est la précision relative de l'agent,} \\ RR \text{ est son rappel relatif,} \\ DR \text{ est sa mesure de digression relative,} \\ a,\ b,\ c \text{ sont des paramètres qui varient en fonction du type de besoin,} \\ \text{et vérifient } a + b - c \neq 0. \end{cases}$$

Le tableau suivant résume l'importance des différentes mesures pour l'auto-évaluation des agents en fonction du type de besoin :

Type de Besoin	Importance de		
	a (précision relative)	b (rappel relatif)	c (digression)
Précis	forte	moyenne	forte
Thématique	moyenne	forte	moyenne
Exploratoire	faible	moyenne	faible

6.4.2 Auto-évaluation des agents de bouclage de pertinence

Pour évaluer l'efficacité du bouclage de pertinence et de la reformulation, il est nécessaire d'introduire de nouvelles mesures. En effet, les mesures proposées jusqu'ici sont valables pour une étape donnée. À partir de la deuxième étape, les agents de recherche peuvent évaluer l'amélioration des résultats apportée par la dernière étape grâce à une mesure de gain de bouclage proposée dans [Lejal 95, p29] : le **gain de bouclage** mesure la proportion de nouveaux documents jugés pertinents par l'utilisateur à l'étape i par rapport au nombre total de documents jugés pertinents par l'utilisateur au cours de la session de recherche.

$$GainDeBouclage_i = \frac{|Pertinent_{\mathcal{U},i}|}{|\bigcup_{j=1}^{i} Pertinent_{\mathcal{U},j}|}$$

où $|Pertinent_{\mathcal{U},i}|$ désigne le nombre de nouveaux documents retrouvés à l'étape i et jugés pertinents par l'utilisateur, et $|\bigcup_{j=1}^{i} Pertinent_{\mathcal{U},j}|$ désigne le nombre total de documents jugés pertinents lors de toutes les étapes précédentes.

Pour un agent donné, cette mesure sera calculée par

$$GainDeBouclage_{\mathcal{A},i} = \frac{|Pertinent_{\mathcal{U},i} \cap Trouvés_{\mathcal{A},i}|}{|\bigcup_{j=1}^{i} Pertinent_{\mathcal{U},j} \cap Trouvés_{\mathcal{A},j}|}$$

où $Trouvés_{\mathcal{A},i}$ représente l'ensemble des documents trouvés par l'agent $\mathcal{A}$ à l'étape i.

Le gain de bouclage permet d'exploiter les choix positifs de l'utilisateur. Puisque nous lui donnons la possibilité d'émettre des choix négatifs en rejetant des documents qui ne lui

conviennent pas, nous devons également exploiter ces rejets. C'est pourquoi nous proposons de calculer une mesure de **perte de bouclage**, pour mesurer le nombre de nouveaux documents trouvés à l'étape i et jugés non pertinents par l'utilisateur. Salton a proposé de mesurer l'échec (*fallout* en anglais) d'une recherche par le nombre de documents retrouvés qui ne sont pas pertinents par rapport au nombre total de tous les documents non pertinents de la collection :

$$\textit{Échec} = \frac{|\textit{NonPertinent} \cap \textit{Trouvés}|}{\textit{NonPertinent}}$$

Nous proposons d'adapter cette mesure pour calculer la perte de bouclage :

$$PerteDeBouclage_i = \frac{|\ \textit{NonPertinent}_{\mathcal{U},i} \cap \textit{Trouvés}_i\ |}{|\ \bigcup_{j=1}^{i} \textit{NonPertinent}_j \cap \textit{Trouvés}_j\ |}$$

où $|\textit{NonPertinent}_{\mathcal{U},i}|$ désigne le nombre de nouveaux documents retrouvés à l'étape i et jugés non pertinents par l'utilisateur.

Pour chaque agent de recherche $\mathcal{A}$, la perte de bouclage à l'étape i est évaluée par :

$$PerteDeBouclage_{\mathcal{A},i} = \frac{|\textit{NonPertinent}_{\mathcal{U},i} \cap \textit{Trouvés}_{\mathcal{A},i}|}{|\ \bigcup_{j=1}^{i} (\textit{NonPertinent}_{\mathcal{U},j} \cap \textit{Trouvés}_{\mathcal{A},j})\ |}$$

Cette mesure permet d'évaluer si la non satisfaction de l'utilisateur vis-à-vis des documents présentés croît ou non durant la session de recherche.

Une autre solution aurait été de comparer la proportion de de documents non pertinents par rapport à la proportion de documents pertinents retrouvés : si un agent retrouve plus de documents non pertinents que de documents pertinents, il pourrait être éliminé. Cette dernière mesure est très différente de la précédente. Elle ne permet en effet que de mesurer ponctuellement l'insatisfaction de l'utilisateur.

Un agent dont le gain de bouclage progresse au cours des étapes de la recherche prend de plus en plus d'importance pour la reformulation. Cette importance peut se traduire par un abaissement du seuil de pertinence au-delà duquel de nouveaux concepts (simples s'il s'agit de termes ou complexes s'il s'agit de graphes conceptuels) seront introduits dans la requête reformulée. Un agent dont la perte de bouclage est supérieure au gain de bouclage ne participera pas à une éventuelle prochaine reformulation dans le cas d'un besoin précis ou thématique.

L'utilité de bouclage d'un agent est donc mesurée par

$$\textit{Utilité de bouclage} = \frac{\textit{gain de bouclage}}{\textit{perte de bouclage} + \textit{gain de bouclage}}$$

Cette mesure vaut 1 si la perte de bouclage est nulle, 0 si le gain de bouclage est nul. Elle croît avec le gain de bouclage et diminue lorsque la perte de bouclage augmente.

6.5 Évaluation globale du système

6.5.1 Les mesures d'évaluation de la société des agents

Les mesures d'évaluation proposées jusqu'ici concernent principalement les agents individuels. Or, le fonctionnement global de la société d'agents doit également être évalué. C'est pourquoi les mesures de précision, de rappel relatif, de gain et de perte de bouclage seront effectuées pour le système global. C'est le nombre total de voix obtenues, c'est à dire le résultat du scrutin qui est utilisé pour le calcul des mesures de précision, rappel et digression à l'échelle du système. Pour le gain et la perte de bouclage, seule l'évolution de ces mesures est exploitée; en effet, si une recherche est particulièrement efficace et le besoin bien exprimé[22], les principaux documents pertinents seront retrouvés lors des premières étapes, et seuls quelques documents supplémentaires pourront être retrouvés par la suite.

6.5.2 Choix d'un mode de scrutin pour l'élection des documents pertinents

Le mode de scrutin est décidé en fonction de la cohérence des résultats des agents de recherche et de la quantité de résultats fournis. L'agent scrutateur analyse les meilleurs votes des différents agents de recherche. Si les résultats des différents agents sont plutôt similaires, le scrutin sera majoritaire. Par contre, si les agents de recherche donnent des résultats très différents les uns des autres, le système pourra opter pour un scrutin proportionnel, en diffusant les documents qui correspondent aux meilleurs votes de chaque agent de recherche. Les résultats sont jugés similaires si les documents les mieux classés par chaque agent obtiennent des voix des autres agents.

Il peut être particulièrement intéressant de détecter des agents marginaux utiles, c'est à dire dont les résultats fournissent de façon isolée par rapport aux autres agents des documents que l'utilisateur juge pertinents. Le scrutin proportionnel est justifié dans ce cas : ces agents risquent de voir leurs résultats « censurés » par la société d'agents si le mode de scrutin retenu est un scrutin majoritaire, puisque leur vote ne correspond pas à celui de la majorité.

6.5.3 Notion de complétude de la recherche

Le système doit assurer une certaine complétude de la recherche, c'est à dire que tous les concepts de la requête (connus du SRI) soient traités par au moins un agent. Chaque concept de la requête doit être traité par au moins un agent de recherche. C'est pourquoi nous avons dû introduire un coordinateur dont le rôle est d'assurer que chaque concept a été choisi par au moins un agent de recherche. Lorsque les agents de recherche sélectionnent un concept de la requête

22. bien exprimé signifie ici exprimé de façon adéquate par rapport au SRI, c'est-à-dire proche de l'indexation des documents.

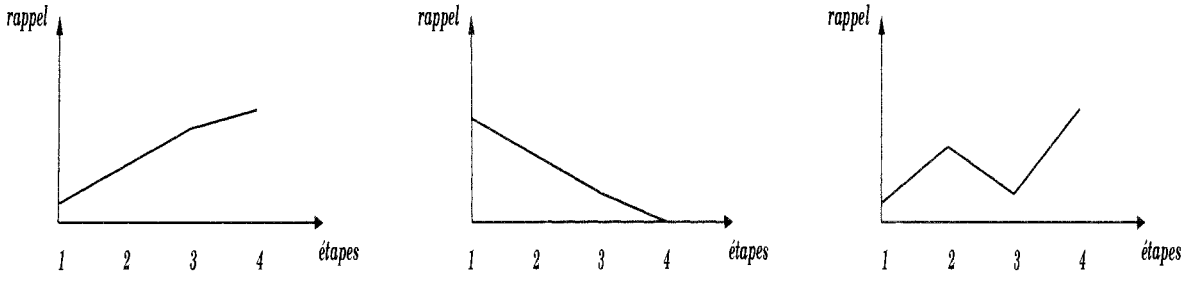
pour le traiter, ils effectuent un marquage du concept considéré dans la requête. L'ensemble des concepts non marqués est affecté à un agent de recherche vectoriel sollicité par le coordinateur.

6.5.4 Conséquences de l'auto-évaluation sur le fonctionnement du système

L'auto-évaluation dote le système d'une certaine autonomie : en fonction des résultats obtenus, son comportement est modifié de manière dynamique. Ainsi, le nombre de voix attribué à chaque agent pour l'élection des documents pertinents peut être modulé en fonction de ses résultats. Un agent qui ne dispose plus de voix à une étape donnée peut continuer, si les ressources le permettent, à effectuer son traitement mais sans communiquer ses résultats par vote. Ainsi, il pourra éventuellement regagner des voix s'il améliore ses résultats et participer à nouveau au vote lors d'étapes ultérieures.

6.5.5 Évaluation de la progression d'une session de recherche

L'évaluation d'une session de recherche porte sur la progression globale de la recherche au cours de la session. Il est possible de faire cette évaluation pour chaque agent de recherche qui est intervenu lors de la session, et globalement pour la société d'agents.



(a) Rétroaction amplifiante cumulative

(b) Rétroaction amplifiante récessive

(c) Rétroaction compensatrice

FIG. 6.4 – *Les types de rétroaction possibles pour le rappel.*

Les différents cas qui peuvent être examinés sont les suivants :

1. la rétroaction est amplifiante et cumulative : lors des différentes étapes de la recherche, la mesure considérée évolue de façon monotone croissante (figure 6.5.5.a). L'interprétation de ce type de rétroaction diffère en fonction du critère d'évaluation utilisé. Si l'on observe ce type d'évolution pour le rappel relatif et la précision, cela signifie que la stratégie de recherche est bien adaptée à la situation. Ceci peut correspondre à l'affinement progressif d'un besoin mal identifié ou mal exprimé au départ, que la session de recherche aide à

cerner. Si les digressions d'un agent augmentent de façon continue, cela signifie que cet agent n'est pas approprié pour traiter le type de besoin courant.

2. la rétroaction est amplifiante et récessive : lors des différentes étapes de la recherche, la mesure observée évolue de façon monotone décroissante (figure 6.5.5.b). Si ce type de rétroaction est observé pour le rappel et la précision relative, cela peut signifier que la stratégie utilisée est mauvaise, ou que le nombre de documents pertinents pour le besoin d'information courant est restreint. Par contre, s'il s'agit du profil d'évolution des mesures de digression, il s'agit d'un indice positif de la convergence de la recherche.

3. la rétroaction est compensatrice : d'une étape de la recherche à la suivante, la mesure étudiée évolue en sens contraire (figure 6.5.5.c). Lorsque les mesures varient en sens inverse d'une étape à l'autre, il est difficile d'en tirer des conclusions; toutefois, ce peut être l'indice d'un changement dans le type de besoin de l'utilisateur, ce dernier ne l'ayant pas signalé au SRI. Dans ce cas, le système pourra mettre en œuvre en parallèle des stratégies alternatives c'est-à-dire correspondant à d'autres types de besoin que celui signalé initialement.

Il serait intéressant d'étudier l'évolution de ces courbes sur un échantillon important de sessions de recherche afin d'essayer d'obtenir des profils « standards » d'évolution en fonction du type de besoin indiqué.

6.6 Le prototype implanté : SAFARI-VIDEO

Les idées développées dans cette thèse ont été mises en œuvre dans le prototype SAFARI-VIDEO (Société d'Agents FAcilitant la Recherche d'Images - Video). Une partie de ce prototype a été développée dans le cadre d'un contrat d'études avec le CCETT et Euroclid[23].

6.6.1 Macro-architecture du système

La figure 6.5 décrit l'architecture globale du système.

Le prototype a été développé en C++. Ce langage de programmation nous a été imposé dans le cadre du contrat qui est à l'origine de la naissance de ce prototype. Malheureusement, nous n'avons pas pu réutiliser les différents développements effectués dans l'équipe EXPRIM en Smalltalk. Nous avons pu constater que le langage C++ n'est pas aussi adapté que Smalltalk au prototypage rapide : il ne dispose pas d'un environnement de développement aussi convivial; par ailleurs la gestion de la mémoire est laissée à la charge du programmeur en C++ alors que Smalltalk la gère lui-même. Toutefois, l'obligation de rompre avec le langage Smalltalk nous a permis d'élargir la portée des réalisations effectuées autour d'EXPRIM. Le prototype développé

23. Contrat 93 ME 13 pour le développement d'une maquette d'un serveur de liens hypermédias pour services multimédias à large bande, où notre participation a consisté à proposer une architecture multi-agents et une application de recherche de séquences d'actualités télévisées.

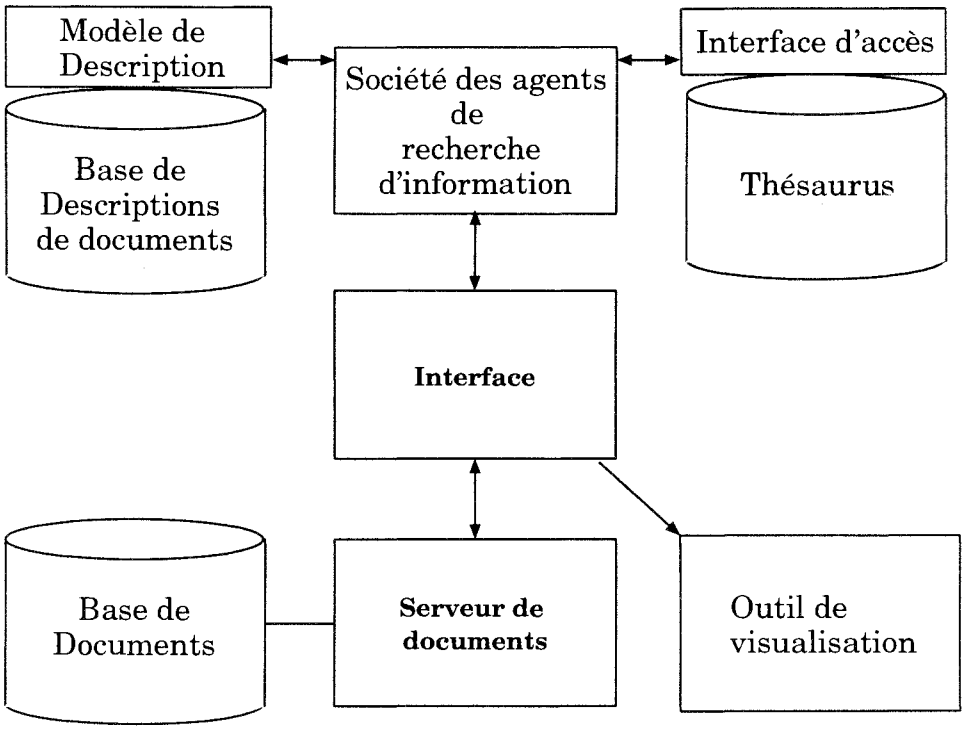


FIG. 6.5 – *Macro-architecture du système.*

devrait être le noyau d'une nouvelle plateforme plus facilement portable d'un système à l'autre, à laquelle pourront venir se greffer de nouveaux développements.

6.6.2 Le Thésaurus

Les données du thésaurus sont stockées sous forme d'un fichier ASCII. Elles sont codées dans un fichier sous forme pseudo-SGML (voir syntaxe en figure 6.6), de façon à ce que leur structure soit conservée. Elles sont chargées en mémoire au lancement de l'application.

Dans ce système, le thésaurus a un rôle consultatif lors d'une recherche d'information. Ce rôle pourrait être remis en cause, par exemple pour une application où un thésaurus n'est pas disponible.

6.6.3 La Base Descriptive

La base descriptive est un objet qui contient la structure de données et les descriptions de documents, et dont le comportement consiste à charger les données à partir d'un fichier en format SGML et à servir aux agents de recherche les descriptions qu'ils demandent.

Outre l'ensemble de descriptions qu'elle contient, elle est caractérisée par un nom et un modèle de description.

unThesaurus	::	<thesaurus> <thName> nomDuThesaurus <\thName> champSemantique+ <\thesaurus>
champSemantique	::	domaine conceptDescripteur+ synonyme*
domaine	::	<thDomain> nomDuDomaine <\thDomain>
conceptDescripteur	::	<thNode> nomConcept nomDomaine generique specifique* termeAssocie* <\thNode>
generique	::	<thNodeGen> nomDuTerme <\thNodeGen>
specifique	::	<thNodeSpe> nomDuTerme <thNodeSpe>
nomConcept	::	<thTermName> nomDuTerme <\thTermName>
nomDomaine	::	<thNodeDom> nomDuDomaine <\thNodeDom>
synonyme	::	<thNodeSyn> nomConcept conceptAEmployer <\thNodeSyn>
conceptAEmployer	::	<thNodeEmp> nomDuTerme <\thNodeEmp>
termeAssocie	::	<thNodeTa> nomConcept nomDomaine nomThesaurus <\thNodeTa>
nomThesaurus	::	<thNodeThesaurus> nomDuThesaurus <\thNodeThesaurus>
nomDuThesaurus	::	chaîne de caractères
nomDuDomaine	::	chaîne de caractères
nomDuTerme	::	chaîne de caractères

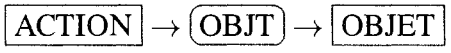
La notation *élément* + signifie que l'élément est présent une fois ou plus, *élément* * qu'il est facultatif et peut être multiple).

FIG. 6.6 – *La syntaxe de description du thésaurus.*

6.6.4 Les descriptions de documents

Les descriptions de documents sont bi-partites : chaque document est décrit par un ensemble d'attributs valués par des concepts; pour les facettes qui comportent une base canonique de graphes conceptuels, la description comporte également une signature de jointure, calculée à partir des codes uniques affectés à chaque graphe des bases canoniques. Le nombre total de graphes conceptuels des bases canoniques est peu élevé, ce qui nous permet d'attribuer à chaque graphe primitif une position binaire dans la signature de jointure. L'addition des signatures des graphes primitifs donne la signature utilisée pour décrire le document. Par exemple, si le graphe

a une signature de jointure égale à 0000001 et si, par ailleurs, le graphe

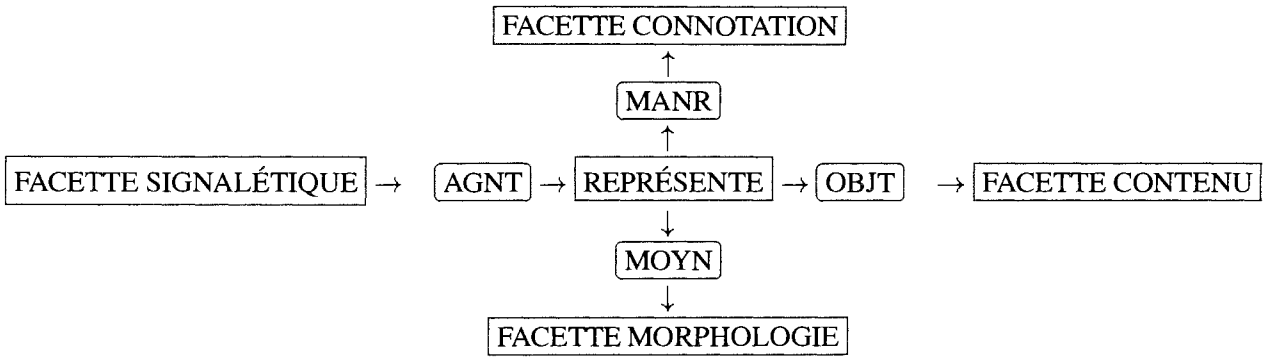


a pour signature de jointure 0000010, le graphe qui résulte de la jointure de ces deux graphes aura pour signature 0000001 + 0000010 = 0000011.

Une description par graphe conceptuel est donc représentée par une liste ordonnée de concepts pondérés (il s'agit ici uniquement des concepts non relationnels), et d'une signature qui indique le graphe conceptuel dans lequel ces concepts interviennent.

6.6.5 Le modèle de description

Le modèle de description donne la liste des facettes d'une application, et pour chaque facette le type de descriptions : simples concepts ou concepts complexes sous formes de graphes conceptuels. Une base de graphes canoniques qui indique les relations possibles entre les facettes. Pour notre application, le modèle de description global correspond au graphe conceptuel suivant :



Un document, repéré par sa signalétique (auteur, titre, date), représente un contenu (événement, personne effectuant une action, etc.) au moyen de techniques décrites dans la facette MORPHOLOGIE (ou TECH-MEDIA), de manière à donner au spectateur une impression décrite par la facette CONNOTATION.

Une facette de description peut être de type simple et dans ce cas elle est caractérisée par un ensemble d'attributs. Si elle est de type graphes conceptuels, elle comporte en plus une base de graphes canoniques.

Un objet de type *attribut* est caractérisé par un nom et son appartenance à une facette. Il sait si son vocabulaire est contrôlé et dans l'affirmative, connaît les domaines du thésaurus qui permettent de contrôler ses valeurs. Il connaît également les méthodes qui permettent de comparer ses valeurs.

La facette Contenu

Elle dispose d'une représentation des descriptions sous forme de graphes conceptuels. Nous avons donné au §3.3.4 page 85 la base canonique de la facette CONTENU. Un graphe conceptuel relie des concepts qui correspondent aux valeurs de chaque attribut. La mise en correspondance peut être vectorielle ou s'appuyer sur les graphes conceptuels.

La facette Signalétique

La facette SIGNALÉTIQUE traite les attributs AUTEUR, DATE-PRODUCTION, LIEU-PRODUCTION, GENRE, TYPE-MÉDIA. Sa base canonique contient les graphes conceptuels

suivants :

AUTEUR → AGNT → RÉALISE → OBJT → TITRE
RÉALISE → MANR → GENRE
RÉALISE → INST → TYPE-MÉDIA
RÉALISE → LIEU → LIEU-PRODUCTION
RÉALISE → TMPS → DATE-PRODUCTION

La facette Morphologie

La facette MORPHOLOGIE est spécialisée par type de média. Pour des images fixes, les attributs retenus sont : PROFONDEUR DE CHAMP, LUMIÈRE, COULEUR, TEXTURE. Pour des images animées, nous utilisons les attributs : TYPE DE PLAN (qui décrit la distance de la caméra au sujet), MOUVEMENT DE LA CAMÉRA et PRISE DE VUE (qui décrit la ligne de visée de l'objectif vers l'objet). Il n'existe pas de relations particulières entre ces attributs.

Une analyse qu'il serait intéressant d'effectuer sur cette facette concerne la suite des types de plans qui existent au sein d'une séquence. Par exemple, une séquence peut être composée dans l'ordre des plans suivants : deux plans moyens (qui montrent quelques personnages en pied), un plan rapproché (où les personnages sont coupés à la ceinture), un gros plan (qui isole un objet ou le visage d'un personnage), deux plans rapprochés et enfin un plan d'ensemble (qui découvre un décor d'assez grandes dimensions).

Si cette séquence est choisie par l'utilisateur, on pourrait rechercher des séquences ayant le même type de composition du point de vue des plans.

6.6.6 Transmission de connaissances entre agents

Dans le prototype réalisé, les agents communiquent en partageant directement des instances d'objets (requête, thésaurus, descriptions, ...).

Mémoire partagée entre les agents de recherche

Les agents de recherche ont accès à une mémoire partagée, puisqu'ils ont des instances d'objets communes, parmi lesquelles :

- la requête initiale de l'utilisateur, qui comprend la description par des termes pondérés des documents recherchés, le type de besoin (exploratoire, thématique, précis), et le nombre de documents souhaités par l'utilisateur (ou sa valeur par défaut),
- la requête interprétée en fonction des connaissances sémantiques du système,
- la liste des documents proposés à l'utilisateur à chaque étape de la recherche, avec pour chaque document le nombre de voix obtenues,

- la liste des documents jugés pertinents par l'utilisateur à chaque étape de la recherche,
- la liste des documents jugés non pertinents par l'utilisateur à chaque étape de la recherche,
- les mesures d'évaluation globales du système à chaque étape.

Mémoire propre des agents de recherche

Chaque agent mémorise à l'échelle de la session de recherche la sous-requête qui le concerne, les choix stratégiques opérés à chaque étape et à chaque phase, et les valeurs de ses mesures d'évaluation. Il a accès à une base de connaissance qui l'aide à déterminer sa stratégie, en fonction du type de besoin.

6.6.7 L'interface

L'interface a été réalisée en **Tcl/Tk** [Ousterhout 94]. Tcl (Tool command language) est un langage de script simple qui permet de contrôler et d'étendre des applications. C'est un langage interprété qui est implanté sous la forme d'une bibliothèque de procédures C. Il peut faire appel à des procédures C «utilisateurs». Un script Tcl peut être lui-même intégré dans un programme C.

Tk est une boîte à outils pour le système X-Window qui permet de construire des interfaces utilisateurs en Motif. Tk comporte une bibliothèque de procédures C qui répertorie des procédures de définition et de gestion de différents types de widgets, c'est-à-dire des types d'objets visuels particuliers : fenêtres, boutons, ascenseurs, etc. La création et l'affichage d'un widget se fait simplement grâce à une commande Tk. Par exemple, les deux lignes de commandes :

```
button .b -text "Au revoir" -commande exit
pack .b
```

permettent d'afficher une petite fenêtre de type *bouton* contenant le texte « Au revoir» :

L'interface réalisée partage avec le SRI un certain nombre d'instances d'objets (Thésaurus, Requête, ListeRésultats, ...). Une commande Tcl a été créée pour chaque fonction de l'interface, et est liée à une procédure C qui fait appel aux méthodes C++ du SRI.

Le comportement de l'interface consiste à créer un interpréteur Tcl, à l'initialiser et à lui affecter les commandes nécessaires. L'interface évalue le script Tcl avant d'entrer dans une boucle d'attente des événements. L'interpréteur est détruit lorsque l'utilisateur quitte l'application.

6.6.8 Le serveur de documents

C'est le système de gestion de fichiers Unix. Les documents vidéo sont prédécoupés en scènes, chaque scène correspondant à un fichier codé en MPEG-1[24] [ISO/IEC-CD-111172, Hung 94]. Un document vidéo correspond à un répertoire Unix qui contient toutes les scènes du document.

6.7 Déroulement d'une session de recherche

6.7.1 Vue de la session de recherche par l'utilisateur

Pour la formulation de sa requête, l'utilisateur peut naviguer dans le thésaurus, en choisissant une domaine (figure 6.7).

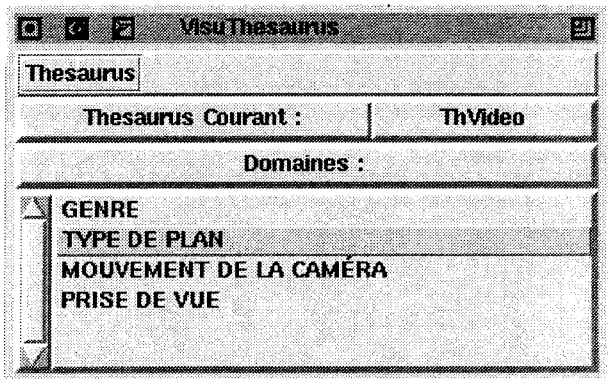


FIG. 6.7 – *Choix d'un domaine sémantique dans le thésaurus.*

Lorsqu'il a sélectionné un domaine sémantique, il peut naviguer dans les concepts de ce domaine en consultant les spécifiques d'un terme, ses synonymes et ses termes associés (figure 6.8).

Pour formuler sa requête, il lui suffit alors de sélectionner un terme et de lui affecter un poids (figure 6.9). Le poids est affecté par manipulation d'un curseur sur une réglette : l'utilisateur déplace le curseur pour indiquer la valeur du poids qu'il souhaite donner au terme sélectionné. Ceci permet à l'utilisateur d'exprimer plus finement son besoin. La valeur nulle (terme indifférent) est exclue des valeurs possibles. Les deux boutons du haut de la fenêtre, qui permettent soit de sélectionner, soit d'éviter un terme s'excluent mutuellement et permettent de déterminer le signe du poids attribué.

Par défaut, le poids affecté est de 1, c'est-à-dire que les documents indexés par ce terme devront être absolument sélectionnés.

24. MPEG = Moving Picture Experts Group, comité OSI qui définit la norme MPEG de codage de séquences d'images numérisées.

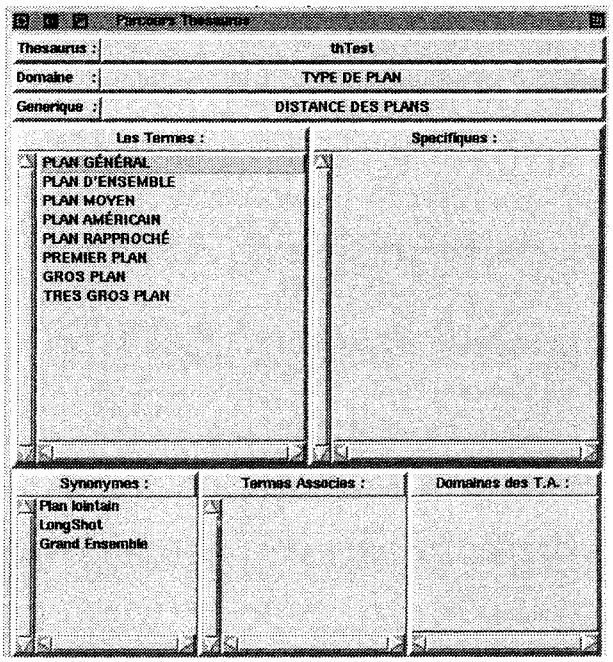


FIG. 6.8 – *Navigation dans un domaine sémantique dans le thésaurus.*

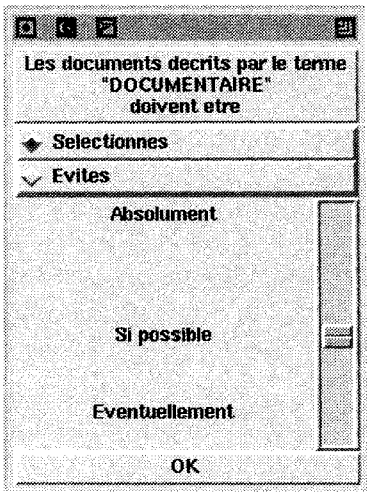


FIG. 6.9 – *Affectation d'un poids d'importance à un terme.*

L'utilisateur voit à tout moment la requête constituée (figure 6.10). Il peut la modifier en supprimant ou en ajoutant un terme, ou modifier le poids d'un des termes. Il peut aussi ajouter directement un terme à la requête sans devoir le retrouver au préalable dans le thésaurus.

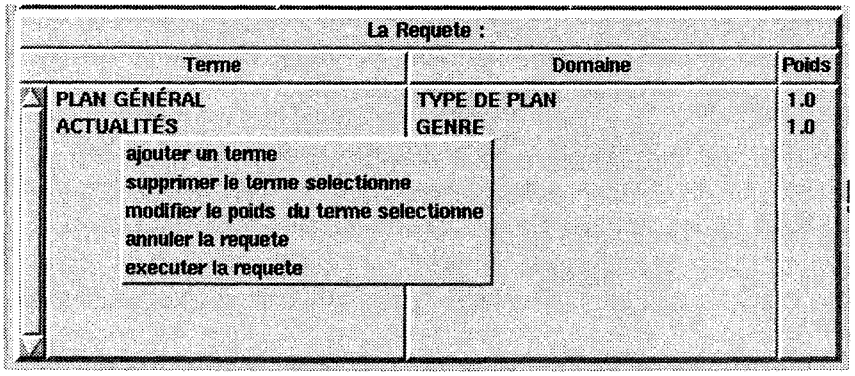


FIG. 6.10 – *Visualisation de la requête constituée.*

Lorsque l'utilisateur lance l'exécution de la requête, les résultats sont affichés sous forme d'une liste de documents (figure 6.11). Lorsque l'utilisateur sélectionne un élément de la liste, l'interface lance la visualisation du document.

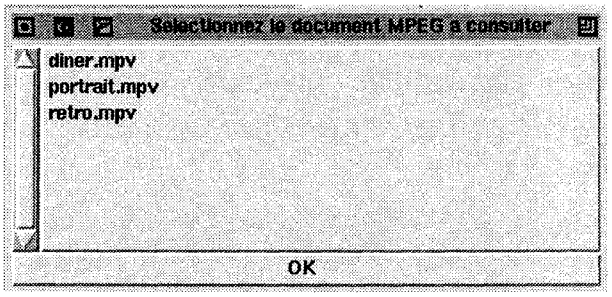


FIG. 6.11 – *Affichage de la liste des documents résultats.*

C'est l'outil *mpeg_play* qui a été utilisé pour faire les tests de visualisation dans notre prototype, mais cet outil très simple et peu interactif mériterait d'être remplacé par une interface de visualisation plus conviviale.

6.7.2 L'application de recherche de séquences d'actualités

Nous disposons de très peu de séquences d'actualités pour évaluer notre application. Le problème des droits sur les documents vidéo est en effet encore plus complexe que pour les documents textuels ou images. S'il est possible de consulter des documents vidéo après signature d'une convention avec l'INA[25] dans le cadre de recherches scientifiques, ceci n'entraîne pas le

25. L'INA (Institut National de l'Audiovisuel) est institué dépôt légal des documents audio-visuels télévisés depuis le 1er Janvier 1995. Toutefois, les chaînes de télévision restent propriétaires de leurs documents pendant trois

droit de recopier ces extraits ou de les diffuser : il faut alors acquérir les extraits à un coût non négligeable.

Les quelques séquences d'actualité que nous avons analysées nous ont permis néanmoins de réfléchir à la façon de traiter ces documents par rapport à des cas réels, et non simplement par rapport à de courts extraits d'images de synthèse ou au contenu peu riche sur lesquels les chercheurs qui utilisent la vidéo travaillent habituellement.

Par ailleurs, même si la technologie a beaucoup évolué autour de la vidéo, et en particulier les techniques de compression, le support numérique pour stocker ces documents n'est pas encore généralisable : il faudrait des disques d'une capacité énorme pour stocker une collection de ces documents. Il faut donc encore attendre des progrès dans ce domaine de recherche, ou investir dans un matériel coûteux permettant d'accéder à des unités de stockage secondaires.

6.7.3 Évaluation du prototype réalisé

Les essais que nous avons réalisé autour de ce prototype sont encourageants, mais ne peuvent servir à une véritable évaluation. Le nombre de documents dont nous disposons est en effet insuffisant.

Un aspect intéressant de ce prototype est que les mesures d'évaluation des agents peuvent être directement utilisées pour comparer les résultats d'une recherche sur une facette ou globalement sur toutes les facettes. Il pourrait être complété d'un mini-questionnaire pour déterminer la satisfaction de l'usager, et comparer ainsi les réponses obtenues avec les mesures effectuées par les agents du système.

La convivialité de l'interface, très importante pour ce type d'application, peut être encore améliorée. Ce ne sont pas les idées qui manquent pour cela, mais le temps de réalisation. En effet, il est encore plus difficile de construire un prototype convivial pour la recherche de vidéos que pour la recherche d'autres types de documents : l'utilisateur est habitué à des images d'assez bonne qualité avec la télévision, et lui restituer cette qualité est un minimum qu'il exige ...

6.8 Travaux liés : agents de RI

Un certain nombre de travaux de recherche sont en cours pour l'exploitation de systèmes multi-agents pour la recherche d'information. Toutefois, la plupart des travaux que nous connaissons s'attachent au filtrage d'information plutôt qu'à la recherche dans des corpus particuliers. La problématique du filtrage d'information est duale de celle de la recherche d'information [Belkin 92] : le besoin d'information de l'utilisateur est représenté dans une sorte de profil d'intérêt qui est plutôt statique, et les collections de documents explorées sont dynamiques soit parce

ans après leur création. Quant aux documents diffusés par les chaînes françaises mais achetés auprès de télévisions étrangères, les droits qui les concernent doivent être renégociés avec le producteur. Il faudrait un véritable système expert pour gérer convenablement tous les aspects de droits d'usage et de recopie de ces documents ...

qu'elles voient arriver de nouveaux documents très fréquemment, soit parce l'exploration peut porter sur des collections différentes d'une recherche à l'autre.

M. Balabanovic et Y. Shoham [Balabanović 95] ont développé à l'Université de Stanford en Californie un système qui aide les utilisateurs à rester au courant des informations nouvelles et intéressantes : ce système exploite World Wide Web et présente chaque jour une sélection de pages censées intéresser l'utilisateur. Celui-ci évalue chaque page présentée, et le système exploite ces retours pour adapter sa recherche et trouver de meilleures pages les jours suivants. Les agents de ce système ont, comme ceux que nous avons conçus, un comportement cyclique. Durant chaque cycle, l'agent part sur W3 à la recherche de pages HTML en utilisant une heuristique et en ayant un temps limité; les p meilleures pages sont sélectionnées et montrées à l'utilisateur qui évalue chaque page présentée. Les jugements de l'utilisateur sont pris en compte pour modifier les heuristiques de recherche et de sélection de documents. Seules les pages de textes sont analysées de façon à en extraire des mots-clés. Les poids des mots dans le texte sont calculés d'après la mesure $tf.idf$ normalisée par la longueur des documents. Les jugements de l'utilisateur sont recueillis globalement pour une page, par une mesure entière dans l'intervalle [-5, +5]. La reformulation modifie un vecteur $\vec{M}$ qui représente les centres d'intérêts de l'utilisateur par des termes pondérés. Les jugements de l'utilisateur servent de coefficient pondérateur pour prendre en compte le poids de chaque concept dans le document :

$$\vec{M} = \vec{M} + \sum_{i=1}^{p} e_i \vec{V_i}$$

Ce prototype est assez représentatif des nouveaux agents qui explorent W3 à la recherche d'information.

La *collecte coopérative d'information* (Cooperative Information Gathering) de [Oates 94, Oates 95] a pour but l'acquisition proactive d'information à partir de sources qui peuvent être hétérogènes, en réponse à une requête complexe. La constitution de l'organisation de recherche repose sur des relations détectées entre leurs buts ou des parties de leurs buts : *facilite* , *permet*, *recouvre*, *gène*, *favorise*, etc. Chaque agent de recherche est responsable d'un corpus dans lequel il va travailler. La solution globale est obtenue par composition de l'information retrouvée par les différents agents, et transformée de façon appropriée par la connaissance du domaine qu'ont les agents. Ainsi, les résultats d'un agent peuvent venir contraindre la recherche d'autres agents. Par exemple, si les agents recherchent un séjour touristique dans une région particulière à une date donnée, les conditions météo vont déterminer le type d'activités qui seront organisées (intérieur/extérieur). Les critères d'évaluation de la recherche portent essentiellement sur la qualité, le coût en ressources et les temps de réponse.

Notre approche diffère de ces travaux en ce que nos agents ne sont pas mobiles : ils sont pour le moment attachés à une application, qui peut être distribuée. La perspective à court terme qui consiste à doter nos agents d'un vrai langage de communication devrait faciliter leur déplacement et encourager leur mobilité.

6.9 Vers un langage de communication entre agents de recherche

Une perspective proche consiste à doter les agents d'un véritable langage d'échange afin qu'ils soient plus autonomes. Ce langage pourrait être par exemple KQML (Knowledge Query and Manipulation Language) [Finin 94b, Finin 94a, Labrou 94], créé à l'initiative du consortium ARPA-KSE (Knowledge Sharing Effort). Le but de KQML est de faciliter le partage et la réutilisation des bases de connaissances. Il s'agit non seulement d'un langage, mais d'un ensemble de protocoles qui permettent à des agents d'échanger de l'information avec d'autres.

KQML s'occupe d'abord de la pragmatique des échanges entre agents : il permet de définir à qui un agent va parler, comment trouver l'interlocuteur, et comment initier et maintenir un échange. Ces fonctions sont assurées par un *facilitateur*, c'est-à-dire un agent qui effectue divers services de communication (acheminement des messages, mise en correspondance de fournisseurs et de clients, services de médiation et de traduction).

KQML repose sur la théorie des actes de langage de Searle [Searle 69]. Il permet de distinguer trois niveaux logiques dans la communication : le contenu du message bien sûr (qui correspond à la *locution* dans la théorie de Searle), et la localisation de celui avec qui la communication doit être établie, mais aussi la façon de présenter le message de manière à établir clairement le but de la communication (l'*illocution* ou transmission des intentions du locuteur à celui qui l'écoute).

Dans la théorie des actes de langage, une illocution a un *contenu propositionnel* et une *force* qui correspond au type d'action qui est attendue en réponse au message transmis [Duval 94]. Cinq forces illocutoires ont été identifiées : les assertions (par exemple, «il pleut») qui provoquent simplement un changement dans l'état des connaissances du receveur mais ne provoquent pas d'action en retour, les directives (par exemple, «lisez ce passage») qui expriment qu'une action est attendue en retour, les engagements (par exemple «je lirai votre article ce soir») qui informent qu'un agent s'engage à exécuter une action, les déclarations (par exemple «je m'appelle Dupont») qui comprennent l'action en elle-même, et l'expressivité qui exprime sentiments et attitudes.

Ces classes d'illocution se retrouvent dans la notion de *performative* de KQML : une performative est un verbe qui indique clairement le type d'acte de langage que l'émetteur attache au contenu du message. Les principaux types de performatives de KQML sont :

- les requêtes (*evaluate*, *ask-if*, *ask-all*, *ask-one*, etc...) qui indiquent le type de réponse attendue par l'émetteur; il existe des requêtes simples et des requêtes qui demandent plusieurs réponses;
- les verbes assertifs qui délivrent simplement une information ou une réponse (*tell, deny, sorry, error*);
- les génératrices qui créent un état chez l'agent receveur,
- la définition de capacités où un agent informe d'autres agents de ce qu'il peut faire.

Doter les agents du SRI d'un tel langage de communication permettrait aux agents de travailler réellement en parallèle, sans partager par convention une mémoire commune (dans le cas de notre prototype des instances d'objets). Toutefois, cela nécessite également d'utiliser un langage d'échange de connaissances; les travaux en cours autour de KIF (Knowledge Interchange Format) [Genesereth 92] doivent donc également être explorés.

6.10 Résumé

Dans ce chapitre, nous avons présenté de manière plus concrète le modèle d'agent que nous avons retenu, et le prototype SAFARI-Vidéo réalisé. Notre système multi-agent a été implanté grâce à un langage orienté objet. Les perspectives de prolongement de ce travail pluridisciplinaire sont nombreuses, et nous avons déjà évoqué la plus immédiate d'entre elles : l'utilisation d'un langage d'agents qui, seul, pourra réellement donner une certaine autonomie aux différentes entités de l'architecture, en mettant l'accent sur les communications. Il nous reste à conclure sur le travail effectué et à présenter d'autres perspectives de prolongement de ces recherches.

Chapitre 7

Conclusion et Perspectives

7.1 Pour conclure ...

La conception et la construction d'un système de recherche d'information est une tâche lourde, et nous avons souhaité concevoir une architecture flexible et ouverte, susceptible de s'adapter à des applications différentes et à des besoins d'information variés. C'est pourquoi nous avons adopté une approche multi-agents pour la modélisation du système. Le SRI n'est plus un système monolithique, mais une société d'agents qui s'organisent pour assister l'utilisateur dans sa recherche d'information, l'aidant à mieux définir sa demande, et lui proposant une stratégie de recherche adaptée à sa situation. Plus que de simples assistants auxquels on délègue une tâche, les agents de recherche sont de véritables collaborateurs du chercheur d'information puisqu'ils peuvent prendre des décisions et des initiatives.

Le modèle de société d'agents que nous avons proposé constitue un noyau qui est destiné à être enrichi par des agents ayant des compétences hétérogènes. Mais d'ores et déjà, l'apport des systèmes multi-agents pour la recherche d'information est effectif et prometteur : l'analyse d'un SRI en tant que système proprement dit permet déjà de mettre en relief ce qui est dû au fait d'agents individuels, et ce qui résulte de l'exploitation collective de leurs connaissances et de leurs compétences.

L'analyse «orientée agent» d'un système n'est pas simple, et nous avons tenté de répondre à des problèmes non triviaux lors de l'étape de conception. Le premier problème auquel nous avons été confrontée est le choix de la granularité et du type des agents : Quel type d'agent est le plus adapté pour résoudre un problème ? Quelle est la granularité optimale d'un agent ? Nous avons appuyé nos choix et nos décisions sur des facteurs fonctionnels et pragmatiques : limiter le nombre de communications au sein du système, éviter un couplage trop étroit entre des agents différents mais liés nous a fait opter pour des agents de recherche plutôt «gros grain» et cognitifs : ils sont amenés à prendre des décisions en fonction du problème à résoudre et des connaissances dont ils peuvent disposer. Pour permettre la réutilisation, ces agents ont été conçus comme des unités composites, faisant appel à des agents très simples («fin grain») qui effectuent une tâche à la demande.

D'autres problèmes de conception se posent, en particulier à propos des moyens d'assurer une certaine complétude de la résolution de problème en concevant la coordination des agents. Existe-t-il un mode de coordination optimal en fonction du problème résoudre ? Enfin, le problème de l'évaluation du système global est encore ouvert. S'il est relativement courant de montrer que les résultats du système sont meilleurs que la simple somme de ceux des agents individuels, comment mesurer effectivement l'apport de chaque agent et les performances par rapport aux ressources supplémentaires nécessaires ? Les propositions que nous avons faites en ce sens ne sont qu'un embryon de solution, et méritent d'être approfondies. Ce ne sont là que quelques unes des questions qui subsistent à l'issue de ce travail.

Nous espérons avoir convaincu le lecteur que la recherche d'information est une problématique passionnante et qu'elle fournit un cas d'étude bien adapté pour réfléchir aux problèmes que nous venons d'évoquer, en particulier lorsque la recherche est interactive et qu'elle concerne des collections de documents multimédias.

La recherche d'information dans des collections de documents vidéo pose les deux principaux problèmes liés à l'accès simultané par un utilisateur à des SRI différents (problèmes bien résumés dans [Wiederhold 92]) :

- le volume de données potentiellement accessible est énorme, et l'utilisateur doit pouvoir se faire une idée de ce dont traite l'application avant de se lancer dans une recherche. Le modèle que nous proposons comporte un méta-modèle, une abstraction des différentes facettes de représentation, qui permet à l'utilisateur d'avoir une vision abstraite et globale du système et de ce qu'il peut en attendre. De plus, les descriptions textuelles peuvent être vues comme un résumé du document. Le modèle de description composite que nous avons proposé permet de représenter de façon économique les substituts sur lesquels s'effectue la recherche dans de telles collections de documents.
- les informations ont des représentations différentes (images, sons, textes) qui sont toutes importantes, et se modifient les unes les autres lorsqu'elles sont présentées ensemble. Ces informations hétérogènes nécessitent chacune des modes de traitement appropriés.

Le média vidéo est donc complexe mais très intéressant car il est à lui seul multimédia. L'association des images, du son et quelquefois du texte en fait un support riche, où les différentes modalités peuvent se renforcer mutuellement ou au contraire s'opposer pour solliciter davantage les réflexions du spectateur. La plupart des travaux, très récents, qui s'intéressent à la recherche dans des collections de documents vidéo adoptent une approche résolument «vision», en cherchant des indices invariants dans les images pour les caractériser : couleur, texture, traits caractéristiques. Nous pensons que l'approche classique de description des documents par des mots-clés ou du texte a également son importance pour traiter une recherche dans de tels documents : un petit résumé textuel et quelques images peuvent donner une idée du contenu du document filmé; le commentaire parlé d'un document vidéo peut être retranscrit afin d'être exploité pour une recherche. Ces approches sont complémentaires et leur exploitation conjointe est rendue possible par l'architecture multi-agents que nous avons proposée.

Nous avons proposé dans ce travail d'exploiter de façon incrémentale des descriptions de documents plus riches que de simples listes de mots-clés. Les agents de notre système exploitent

d'abord les possibilités du modèle vectoriel et ne mette en œuvre une recherche plus précise que sur certains aspects des documents, et pour un sous-ensemble limité de documents. De même, nous avons proposé une reformulation de la requête exploitant le mécanisme des graphes conceptuels, ce type de reformulation permettant non seulement de déterminer les termes qui représentent le mieux le besoin d'information de l'utilisateur, mais aussi le contexte dans lequel ces termes sont pertinents. L'exploitation du formalisme de représentation par graphes conceptuels ouvre de nouvelles perspectives, qui méritent d'être approfondies, comme la représentation de connaissances en apparence contradictoires. Ceci n'est donc pas une conclusion, au sens d'un arrangement définitif, mais un point d'arrêt provisoire dans un travail qui mérite de nombreux autres développements.

Face à la complexité et à la grande quantité d'information que contiennent les documents vidéo, une analyse par facettes de connaissances s'est naturellement imposée. Cette approche a pu être également adoptée avec profit pour des documents images fixes. Elle permet d'identifier les aspects des documents qui intéressent le plus l'utilisateur, et éventuellement de lui faire découvrir de nouveaux pans de connaissance. Comme le constate Philip Pacey ([Pacey 79, p417] cité dans [Turner 93b]), il existe une différence cruciale entre l'illettrisme et l'illettrisme visuel :

> « Les personnes qui ne savent pas lire savent qu'elles ne savent pas lire. Nous *pensons* tous que nous savons *lire* les images - mais très probablement nous ne le savons pas ... »

Même si la plupart des demandes d'information portent sur le contenu du document, un système de recherche d'information ne peut pas faire l'impasse sur l'analyse du langage propre à un média donné. Sans avoir la prétention de vouloir apprendre au chercheur d'information à lire les images, le SRI multi-agent doit utiliser ses connaissances expertes pour l'aider à progresser dans sa recherche.

7.2 Perspectives

La recherche d'information dans des collections de documents vidéo est une problématique récente, et nous n'avons pas du tout épuisé le sujet, loin de là. Nous avons dû construire pour ce travail un prototype complètement nouveau, tâche lourde qui a nécessité beaucoup d'efforts avant d'obtenir les premiers résultats. La contrepartie de cette implantation *ex nihilo* est que de nombreuses idées n'ont pu être réalisées et testées sur ce prototype. Les perspectives sont donc nombreuses. Tout au plus avons nous ouvert quelques portes, derrière lesquelles se trouvent de nombreuses voies que nous n'avons pas eu le temps d'explorer. Nous allons tenter de dresser une liste des points essentiels qui devraient être poursuivis dans ce travail.

7.2.1 Perspective d'évaluation d'autres critères du système

Des critères d'évaluation autres que ceux que nous avons proposés seraient intéressants à étudier dans un SRI multi-agents, parmi lesquels ceux qui traitent de la redondance entre les

agents de recherche. La redondance peut être analysée de deux façons : par les données qui sont prises en compte par les agents pour réaliser leur but, ou par les résultats qu'ils délivrent. Un agent peut en effet être considéré comme redondant s'il affecte la même proportion de voix à chaque document qu'un autre agent, tout en votant ou non pour moins de documents. Si le vote d'un agent est représenté par une liste ordonnée $\mathcal{V}$ de couples v = (document, nombre de voix), un vote $\mathcal{V}1$ est redondant vis-à-vis d'un vote $\mathcal{V}2$ si

$$\begin{gathered}\forall v1_i \in \mathcal{V}1, \exists v2_i \in \mathcal{V}2 \\ (document(v1_i) = document(v2_i)) \\ \text{et} \\ (nombreDeVoix(v1_i) = k.nombreDeVoix(v2_i))\end{gathered}$$

Comment exploiter ces connaissances sur la redondance ? Un agent redondant ne peut pas être simplement supprimé : son vote vient conforter celui d'un autre agent. Toutefois, dans le cas où l'utilisateur est pressé, il est possible de supprimer l'agent redondant pour augmenter les ressources disponibles et diminuer le temps de recherche. Dans ce cas, les voix de l'agent redondant pourraient être affectées à l'agent qui le remplace, qui servirait alors d'échantillon représentatif pour le sondage que constituerait cette recherche d'information amoindrie.

De même, les performances des différents agents pourraient être analysées de manière approfondie, et pas seulement en terme de rappel et de précision relative.

7.2.2 Apprentissage à long terme

Les agents de recherche pourraient être dotés d'une mémoire à long terme qui leur permette, dans certaines situations, d'exploiter l'expérience acquise au cours de sessions d'interrogation précédentes pour améliorer leur stratégie. Les travaux de Malika Smaïl [Smaïl 94] sur l'utilisation du raisonnement à base de cas pour la recherche d'information, que nous avons évoqués brièvement au § 5.7.2, pourraient être adaptés à notre système. Le raisonnement à base de cas repose sur l'hypothèse que les expériences passées, stockées dans une mémoire à long terme, forment autant de cas accessibles et adaptables qui aident à résoudre de nouveaux cas, par adaptation d'expériences analogues à la nouvelle situation. Pour chaque session de recherche, le système doit mémoriser des informations sur la requête de l'utilisateur et ses évolutions au cours des différentes étapes de la recherche, les valeurs des paramètres stratégiques, les documents retrouvés par le système et les jugements de l'utilisateur sur ces documents. Dans le cas du système multi-agent SAFARI-Vidéo, il serait nécessaire non seulement de doter les agents individuels de capacités de mémorisation, mais aussi le système global d'une mémoire collective qui serait exploitée, en particulier, par le scrutateur. L'accès à cette mémoire pourrait non seulement permettre de déterminer les valeurs des paramètres stratégiques «optimales» pour chaque agent, mais aussi d'analyser les interactions les plus fructueuses entre les agents en fonction de la situation. Le contrôle de la société d'agents pourrait y gagner en souplesse, chaque agent apprenant avec quels autres agents il peut coopérer de façon à améliorer les performances de la recherche (en qualité et en temps de traitement).

7.2.3 Aide à l'indexation automatique

Souvent, les documents vidéo sont accompagnés de textes qui les commentent, les décrivent, les critiquent et qui pourraient servir de support à une indexation automatique. L'indexation pourrait être obtenue en établissant une collaboration entre des agents d'indexation automatique de textes et d'autres capables de détecter des changements de scènes [Zhang 95], ou des agents d'analyse des images-clés d'un scène exploitant des techniques déjà utilisées pour l'image fixe [Gupta 91, Hirata 92, Swanberg 93, Lee 94a, Gorkani 94, Picard 93]. Toutefois, ceci nécessite d'avoir des images-clés de bonne qualité dans un document vidéo; certains chercheurs y ont déjà travaillé [Lippman 89]. Un format de compression comme MPEG, avec perte d'information, ne convient en effet pas très bien à ce type d'analyse si elle doit être précise.

Des agents de détection d'indices visuels (couleurs, textures, traits caractéristiques) pourraient participer à l'analyse des jugements de l'utilisateur pour permettre d'accéder à des documents moins bien indexés. De même la bande sonore pourrait être exploitée pour retranscrire les éléments principaux des dialogues ou des commentaires parlés. La collaboration entre des agents de nature très différente devrait leur permettre de s'enrichir mutuellement. L'architecture de notre système a été conçue de façon à pouvoir accueillir facilement de tels agents.

7.2.4 Amélioration de la gestion des descriptions par graphes conceptuels

L'implantation que nous avons réalisée autour des graphes conceptuels ne peut pas rivaliser par exemple avec la plateforme SYRIMME de M. Mechkour construite au-dessus du SGBD O_2 [Mechkour 95]. Une amélioration possible consisterait à déléguer la gestion de ces représentations à un agent spécialisé.

La représentation des documents par graphes conceptuels nous semble prometteuse pour déboucher sur de nombreux travaux de recherche. La reformulation de la requête que nous avons proposée dans ce formalisme pourrait être améliorée par l'adaptation des techniques d'apprentissage symbolique développées en intelligence artificielle, et en particulier celles de Mitchell. Une première étape de la reformulation consisterait à trouver les relations maximales entre les concepts pertinents pour décrire la requête. L'espace initial des hypothèses serait constitué à partir des graphes conceptuels de description des documents choisis, triés par généralité croissante. Cet ordre partiel ne concerne pas les concepts du thésaurus, puisque les descriptions de documents ne contiennent que les termes les plus spécifiques du vocabulaire, mais le nombre de graphes primitifs contenus. L'algorithme d'élimination des candidats de [Mitchell 82] analyserait les descriptions de documents rejetés pour élaguer les graphes de l'ensemble des hypothèses en supprimant les graphes conceptuels primitifs correspondant aux descriptions de documents rejetés. Une deuxième étape pourrait chercher à généraliser les concepts des graphes conceptuels obtenus. Pour mettre en œuvre ces techniques d'apprentissage, il faut au préalable trouver des heuristiques qui améliorent l'efficacité des algorithmes pour ces problèmes qui sont NP-complets [Haussler 93].

Enfin, il pourrait être intéressant, pour certaines applications, d'exploiter de façon correcte la négation dans les graphes conceptuels. En effet, un graphe de description de document peut

être en version positive pour un média particulier et en version négative pour un autre média. C'est le cas par exemple, de graphes qui décriraient respectivement le texte et l'image du tableau de Magritte «Ceci n'est pas une pomme» (voir §1.3 page 9). La présence de descriptions en version positive et en version négative pour le même document pourrait permettre de répondre à des requêtes générales de type «constraste». Esch et Levinson ont proposé une représentation économique des graphes négatifs en ajoutant deux positions binaires au graphe, pour indiquer si le graphe représente une affirmation, une négation, ni l'un ni l'autre ou les deux [Esch 95]. Ce type d'extension pourrait être exploité en particulier pour les collections de documents publicitaires qui utilisent très souvent ce genre de contraste pour interpeller le client potentiel. Lors de la mise en correspondance, la présence d'un graphe conceptuel en version positive pour un type de média et en version négative pour un autre devrait amener à tirer des conclusions dans un autre monde que celui des représentations traitées. Il reste à définir le langage de requête qui correspondrait à de telles représentations, puisque la négation est utilisée dans notre système comme moyen d'exclure certains documents. La sémantique des poids négatifs affectés aux termes de la requête devrait alors être modifiée en conséquence.

7.2.5 Les croyances

Nous avons proposé de modéliser la connotation des documents sous forme de croyances. Les termes donnés par l'indexeur pourraient servir de base à un processus de révision de croyances qui s'appuierait sur la mémoire à long terme du système. Quelques travaux ont déjà cherché à exploiter la révision de croyances pour la RI [Cawsey 92, Logan 94]. Les mécanismes de révision sont en général assez lourds à mettre en œuvre, et c'est pourquoi il ne nous semble pas souhaitable de généraliser cette approche à toutes les facettes de connaissances du SRI. Toutefois, pour traiter les descriptions relatives à la connotation, cette approche nous semble prometteuse.

Une autre application des théories de révision de croyances concerne le choix des actions et la planification de l'activité des agents de recherche d'information : l'auto-évaluation modifierait l'état de leurs croyances sur leur utilité et sur les performances dont ils sont capables.

7.2.6 Le raisonnement temporel

Parmi les modes de raisonnement possibles des agents, le raisonnement temporel devrait exister. En effet, nous avons simplifié la nature de la structure des documents en la considérant comme une structure de composition : en fait, il s'agit d'une structure temporelle.

Il convient de distinguer les liens temporels qui existent entre des documents et dans le document – succession des plans dans une scène, des scènes dans un document – de ceux qui existent entre les éléments représentés dans le document. Le raisonnement temporel pourrait permettre de retrouver le fil du discours dans un document, ou de comparer des documents abordant des sujets similaires à des points de temps différents : par exemple pour une application médicale, il peut être utile de retrouver des documents montrant l'évolution d'un symptôme dans le temps. Les graphes conceptuels ont déjà été utilisés pour modéliser des données temporelles

(voir par exemple [Moulin 92]). Une analyse approfondie des besoins, c'est à dire des types de requêtes temporelles que les usagers d'un SRI pourraient émettre est nécessaire avant de mettre en œuvre ce type de raisonnement.

7.2.7 Langage de requête visuel

Nous avons choisi un langage de requête minimal pour permettre l'accès au SRI. Ce langage limite les possibilités d'expression de l'utilisateur. Il semble donc souhaitable d'établir un langage de requête plus expressif et qui permette en particulier de traiter les aspects structurels et temporels de la vidéo. Ce langage devrait être aussi visuel que possible. Cette problématique a déjà été abordée dans certains travaux, par exemple [Hibino 95]. Comment exploiter un tel langage visuel dans une application distribuée, et n'y aurait-il pas une contradiction avec le souhait que nous avons formulé d'une interface d'accès unique à plusieurs SRI distribués ? C'est là qu'un format d'échange d'objets multimédias/hypermédias comme MHEG [MHEG 93] pourrait intervenir avec profit; toutefois, ce format devrait être doté d'un véritable langage de script pour que l'interface et l'application soient réellement autonomes les unes par rapport aux autres. En l'absence d'un tel langage, l'interface peut «piloter» les objets MHEG qu'elle reçoit d'une application grâce à des procédures écrites dans un langage prédéfini (par exemple C++), mais cela la rend vraiment dépendante des applications pour laquelle elle réalise les affichages.

7.2.8 Amélioration de l'interface de consultation des documents vidéo

Nous n'avons pas eu le temps de développer un outil de consultation de séquences vidéo propre au SRI, et avons donc réutilisé un outil existant, afin de pouvoir réaliser un prototype extensible. Toutefois, la présentation des documents vidéos pourrait être améliorée par un outil qui permette l'accès direct à une image donnée dans un document, à la façon des magnétoscopes professionnels. En l'absence de cette fonctionnalité, nous avons dû prédécouper les documents en séquences de taille acceptable.

Par ailleurs, il serait intéressant d'améliorer l'interface pour donner des indications visuelles sur l'activité des agents de recherche, sous forme d'icônes animées.

Dans ce travail, nous avons posé les premières pierres d'une nouvelle approche pour la conception des systèmes de recherche d'information. Ces préliminaires étaient nécessaires, mais comme l'indiquent nos nombreuses perspectives, il reste bien sûr beaucoup à faire pour les consolider et les élargir.

Table des figures

1.1 Problématique générale de la recherche d'information interactive. 6

1.2 Relations entre objet, représentant et interprétant lors de la lecture d'une image. 8

1.3 « *Ceci n'est pas une pomme* » René Magritte, 1964. 9

1.4 La vidéo vue comme un objet structuré composite. 12

1.5 Icone vidéo (zoom avant). 14

1.6 Rappel, précision, silence et bruit en Recherche d'Information. 24

2.1 Comparaison des résultats obtenus avec des mesures différentes. 33

2.2 Les différents types de relations d'à-propos. 54

2.3 Les trois interprétations possibles du principe d'incertitude. 57

3.1 Une ligne d'identité regroupant deux concepts complexes sur un concept commun : chat. 71

3.2 Graphe conceptuel pour «un chat a chassé une souris durant 13 secondes de 19:29:32 GMT à 19:29:45 GMT». 81

3.3 «Comtesse portant son fils à califourchon», *Nadar*. 83

3.4 Hiérarchie d'attributs de description des documents. 87

3.5 Treillis «sorte-de» dans le thésaurus. 89

3.6 Représentation dans RIME de «Opacité sur le côté antérieur droit de l'humerus dû à une tumeur». 92

3.7 Représentation de «pollution des rivières en France». 95

4.1 Le processus EXPRIM. 100

4.2 Le modèle paramétré de RI interactive. 101

4.3 Choix de résolution possibles. 104

4.4 Deux exemples de thésaurus. 118
4.5 Réutilisation d'une partie d'une séquence d'archive dans un document 137
4.6 Censure tenant compte du degré de similarité des séquences. 144
4.7 Représentation de la pertinence des séquences. 144

5.1 Les différents types de communication 156
5.2 Le contrôle dans I^3R 163
5.3 Le contrôle hybride multi-phase d'ATOME 164
5.4 Entrée dans un Système de Recherche d'Information 180
5.5 Formulation d'une requête sous forme de mots-clés 182
5.6 Proaction 184
5.7 Choix pour la mise en correspondance 188
5.8 Élection de documents pertinents : table des votes 189
5.9 Architecture générale du SRI 200
5.10 La partition des connaissances du SRI 202
5.11 La diversité des savoir-faire du SRI 202

6.1 Architecture du SRI multi-agents pour la recherche des documents. 209
6.2 Architecture du SRI multi-agents pour la reformulation. 211
6.3 Ensembles de documents après une mise en correspondance. 213
6.4 Les types de rétroaction possibles pour le rappel. 218
6.5 Macro-architecture du système. 220
6.6 La syntaxe de description du thésaurus. 221
6.7 Choix d'un domaine sémantique dans le thésaurus. 225
6.8 Navigation dans un domaine sémantique dans le thésaurus. 226
6.9 Affectation d'un poids d'importance à un terme. 226
6.10 Visualisation de la requête constituée. 227
6.11 Affichage de la liste des documents résultats. 227

Bibliographie

[Adiba 95] Michel Adiba. STORM: Structural and Temporal Object-oRiented Multimedia Database System. *First International Workshop on Multi-Media Database*, pages 12–19, Blue Mountain Lake, NY, 1995. IEEE Computer Society.

[Agha 90] Gul Agha. Concurrent Object-Oriented Programming. *Communications of the ACM*, 33(9):125–141, 1990.

[Aitken 94] J. Stuart Aitken, Franz Schmalhofer et Nigel Shadbolt. A Knowledge Level Characterisation of Multi-Agent Systems. *ECAI-94 Workshop on Agent Theories, Architectures, and Languages Proceedings, Lecture Notes in Artificial Intelligence 890*, pages 179–190. Springer-Verlag, 1994.

[Avrahami 93] Judith Avrahami et Yaakov Kareev. What do you expect when you ask for a "cup of coffee and a muffin or a croissant"? On the interpretation of sentences containing multiple connectives. *International Journal on Man-Machine studies*, 38:429–434, 1993.

[Balabanović 95] Marko Balabanović et Yoav Shoham. Learning Information Retrieval Agents: Experiments with Automated Web Browsing. *Working Notes of the AAAI'95 Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, pages 13–18, 1995.

[Bandon 94] D. Bandon, R. Kantz, V. Boissart, C. Debas, G. Evers, J. Duchêne et C. Wehenkel. Medical Image Storage and Retrieval Strategies. *Intelligent Multimedia Information Retrieval Systems and Management, RIAO'94*, pages 438–448, New York, NY, 1994.

[Bartell 94] Brian T. Bartell, Garrison W Cottrell et Richard K. Belew. Automatic Combination of Multiple Ranked Retrieval Systems. W. Bruce Croft et C.J. Van Rijsbergen, éditeurs, *Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 173–181, Dublin, July 1994. Springer-Verlag.

[Barwise 83] J. Barwise et J. Perry. *Situations and Attitudes*. The MIT Press, Cambrige,MA, 1983.

[Belkin 86] N. J. Belkin et B. Kwasnik. Using Structural Representations of Anomalous States of Knowledge for Choosing Document Retrieval Strategies. *ACM/SIGIR International Conference on Research and Development in Information Retrieval Proceedings*, pages 11–22, Pisa, 1986.

[Belkin 87] Nicholas J. Belkin, Christine L. Borgman, Helen M. Brooks, Tom Bylander, W. Bruce Croft, Penny Daniels, Scott Deerwester, Edward A. Fox, Peter Ingewersen, Roy Rada, Karen Sparck Jones, Roger Thompson et Donald Walker. Distributed Expert-based Information Systems : an Interdiciplinary Approach. *Information Processing & Management*, 23(5):395–409, 1987.

[Belkin 92] Nicholas J. Belkin et W. Bruce Croft. Information Filtering and Information Retrieval : Two Sides of the Same Coin ? *Communications of the ACM*, 35(12):29–38, Dec 1992.

[Belkin 93] N. J. Belkin, C. Cool et W. B. Croft qnd J. P. Callan. The Effect of Multiple Query Representations on Information Retrieval System Performances. *SIGIR'93 (International Conference on Research and Development in Information Retrieval)*, pages 339–346, 1993.

[Belkin 95] N.J. Belkin et P. Kantor. Combining the evidence of multiple query representations for information retrieval. *Information Processing & Management*, 31(3):431–448, 1995.

[Berrut 90] Catherine Berrut. Indexing Medical Reports: the RIME approach. *Information Processing & Management*, 26(1):93–109, 1990.

[Bélières 95] Bruno Bélières et Claude Trépied. Un langage visuel de bases de données associé à un modèle sémantique à objets. *XIIIe Congrès Inforsid (INFormatique des Organisations et Systèmes d'Information et de Décision)*, pages 393–412, Grenoble, juin 1995.

[Bournaud 95] Isabelle Bournaud et Jean-Gabriel Ganascia. Conceptual Clustering of Complex Objects: A Generalization Space based Approach. *Conceptual structures: Applications, Implementations and Theory, (ICCS'95), LNAI 954*, pages 173–187. Springer Verlag, 1995.

[Bruza 91] P.D. Bruza et T.P. van der Weide. The Modelling and Retrieval of Documents using Index Expressions. *SIGIR Forum*, 25(2):91–103, 1991.

[Bruza 94] P.D. Bruza et T. W. C. Huibers. Investigating Aboutness Axioms Using Information Fields. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 112–121. Springer-Verlag, 1994.

[Buford 93] John F. Buford et Chetan B. Gopal. Standardizing a Multimedia Interchange Format: A Comparison of OMFI and MHEG. *ftp://img.uml.edu/pub/MHEG*, 1993.

[Cammarata 83] S. Cammarata, D. McArthur et R. Steeb. Strategies of Cooperation in Distributed Problem Solving. *Proceedings of 8th Joint Conf. on AI*, pages 767–770, 1983.

[Castelfranchi 89] Cristiano Castelfranchi. Social Power: a Point Missed in Multi-Agent, Distributed Artificial Intelligence and Human Computer Interaction. Y. Demazeau et J.P. Müller, éditeurs, *Proceedings of the First European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, 1989.

[Castelfranchi 92] Cristiano Castelfranchi, Maria Miceli et Amedeo Cesta. Dependence Relations among Autonomous Agents. Eric Werner et Yves Demazeau, éditeurs, *Decentralized A.I. 3*, pages 215–227. Elsevier Science Publishers B.V., 1992.

[Castelfranchi 95] Cristiano Castelfranchi. Guarantees for Autonomy in Cognitive Agent Architecture. *Intelligent Agents, ECAI'94 workshop on Agent Theories, Architectures and Languages*, pages 56–70. LNAI 890, Springer-Verlag, 1995.

[Cawsey 92] Alison Cawsey, Julia Galliers, Steven Reece et Karen Sparck Jones. Automating the Librarian: Belief Revision as a Base for System Action and Communication with the User. *The Computer Journal*, 35(3):221–232, 1992.

[Chaffiol 93] Daniel Chaffiol. *RIVAGE et Imageur*. CRIN, Nancy, 1993. Rapport de stage ESIAL 2.

[Cherfaoui 95] Mourad Cherfaoui. *Indexation et consultation de documents vidéo*. Thèse de Doctorat, Université de Rennes I, 1995.

[Chevallet 92] Jean Pierre Chevallet. *Un modèle logique de Recherche d'Information appliqué au formalisme des graphes conceptuels. Le prototype ELEN et son expérimentation sur un corpus de composants logiciels*. Thèse de Doctorat, Université Joseph Fourier - Grenoble I, Grenoble, France, mai 1992.

[Chevrier 93] Vincent Chevrier. *Étude et mise en œuvre du paradigme multi-agents*. Thèse de Doctorat, Université de Nancy I, Nancy, France, 1993.

[Chiaramella 92] Yves Chiaramella et Jean Pierre Chevallet. About retrieval models and logics. *The Computer Journal*, 35(3):233–242, 1992.

[Cluzeau-Ciry 88] Muriel Cluzeau-Ciry. Typologie des utilisateurs et des utilisations d'une banque d'images. *Le Documentaliste*, 25(3):155–120, 1988.

[Colin 92] Michel Colin. *Cinéma, Télévision, Cognition.* Processus Discursifs - Langage et Cognition. Presses Universitaires de Nancy, 1992.

[Corkill 83] D. D. Corkill. *A Framework for Organizational Self-design in Distributed Problem Solving Networks.* Thèse de Doctorat, University of Massachusetts, Amherts, MA, 1983.

[Créhange 85] M. Créhange, J.M. David, O. Foucaut et B. Heulluy. Le point sur EXPRIM. Rapport no. 85-R-019, CRIN, 1985.

[Croft 84] W. Bruce Croft et Roger H. Thompson. The Use of Adaptive Mechanisms for Selection of Search Strategies in Document Retrieval Systems. *Proceedings of the third joint BCS-ACM symposium on Research and Development in information retrieval*, pages 96–110, 1984.

[Croft 87] W. Bruce Croft. Approaches to Intelligent Information Retrieval. *Information Processing & Management*, 23(4):249–254, 1987.

[Croft 90a] W. Bruce Croft et Raj Das. Experiments with Query Acquisition and Use in Document Retrieval Systems. *13th International Conference on Research and Development in Information Retrieval, ACM SIGIR'90*, pages 349–368, Sep 1990.

[Croft 90b] W.B. Croft, R. Krovetz et H. Turtle. Interactive Retrieval of Complex Documents. *Information Processing & Management*, 26(5):593–613, 1990.

[Croft 92a] W. B. Croft et H. R. Turtle. Retrieval of Complex Objects. *LNCS 580*, pages 217–229. EDBT 92, 1992.

[Croft 92b] W. Bruce Croft et Lisa A. Smith. A Loosely-coupled Integration of a Text Retrieval System and an Object-oriented Database System. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 223–232, 1992.

[Crouch 90] Carolyn J. Crouch, Donald B. Crouch et Krishna R. Nareddy. The automatic generation of extended queries. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 369–383, 1990.

[Daniels 85] P.J. Daniels, H.M. Brooks et N.J. Belkin. Using problem structures for driving human-computer dialogues. *RIAO 85 (Recherche d'Information Assistée par Ordinateur)*, pages 645–660, 1985.

[Dauzats 94] Michel Dauzats, éditeur. *Le Thesaurus de l'image - Étude des langages documentaires pour l'audiovisuel.* ABDS Éditions, 1994.

[David 90] Amos Abayomi David. *Processus EXPRIM, Image et IA pour un EIIAO individualisé : le prototype BIRDS*. Thèse de Doctorat, INPL, 1990.

[Davis 83] R. Davis et R. G. Smith. Negociation as a Metaphor for Distributed Problem Solving. *Artificial Intelligence*, 20:63- 109, 1983.

[Davis 92] Hugh Davis, Wendy Hall, Ian Heath, Gary Hill et Rob Wilkins. Towards An Integrated Information Environment With Open Hypermedia Systems. *Proceedings of the ACM Conference on Hypertext, ECHT'92*, pages 181–190, Milano, oct 1992.

[Davis 93] Hugh Davis, Wendy Hall et Ian Heath. Media Integration Issues within Open Hypermedia Systems. Rapport, The Image and Media Laboratory, Department of Electronics and Computer Science, University of Southampton, UK, 9293.

[Defude 86] Bruno Defude. *Étude et réalisation d'un système intelligent de recherche d'informations : le prototype IOTA*. Thèse de Doctorat, Université Joseph Fourier, Grenoble, Juillet 1986.

[Demazeau 89] Yves Demazeau et Jean-Pierre Müller. Decentralized Artificial Intelligence. Yves Demazeau et Jean-Pierre Müller, éditeurs, *Proceedings of the First European Workshop on Modelling Autonomous Agents in a Multi-Agent World*. North Holland, 1989.

[Deustch 71] M. Deustch. *Conflict Resolution: Contributions of the Behavioural Sciences*, chapitre "Conflict and its Resolution". C. G. Smith, University of Notre Dame Press, London, 1971.

[Dieng 94] Rose Dieng, Olivier CORBY et Sofiane LABIDI. Agent-based knowledge acquisition. *European Knowledge Acquisition Workshop'94*, 1994.

[Duda 95] Andrzej Duda. Structured temporal composition of multimedia data. *First International Workshop on Multi-Media Database*, pages 136–142, Blue Mountain Lake, NY, 1995. IEEE Computer Society.

[Dunlop 93] M.D. Dunlop et C.J. Van Rijsbergen. Hypermedia and Free Text Retrieval. *Information Processing & Management*, 29(3):287–298, 1993.

[Durfee 87] E. H. Durfee, V. R. Lesser et D. D. Corkill. Coherent cooperation among communicating problem solvers. *IEEE Transactions on Computers*, 36(11):1275–1291, 1987.

[Durfee 89] E. H. Durfee, V. R. Lesser et D. D. Corkill. Trends in cooperative distributed problem solving. *IEEE Transactions on Knowledge and Data Engineering*, 1(1):63–83, March 1989.

[Durfee 92] Edmund H. Durfee, Daniel Darmouth, Marcus Huber, Thomas A. Montgomery et Sandip Sen. *The Search for Coordination: Knowledge-Guided Abstraction and Search in a Hierarchical Behavior Space*, volume 830, série *LNAI*, pages 187–206. Springer-Verlag, 1992.

[Duval 94] France Duval. Mémoire et Intention collectives pour une stratégie d'actions coordonnées. Communication lors d'un séminaire de l'équipe RFIA - CRIN, Juin 1994.

[Duval 95] France Duval. Technologie et dynamisme des sémantiques collectives. Communication lors d'un séminaire de l'équipe EXPRIM - CRIN, 1995.

[Esch 95] John Esch et Robert Levinson. An Implementation Model for Contexts and Negation in Conceptual Graphs. *Conceptual structures: Applications, Implementations and Theory, (ICCS'95), LNAI 954*, pages 247–262. Springer Verlag, 1995.

[Etzioni 94a] O. Etzioni, N. Lesh et R. Segal. Building softbots for UNIX. *Software Agents - Papers from the AAAI 1994 Spring Symposium*, pages 9–16. AAAI Press, 1994.

[Etzioni 94b] Oren Etzioni et Daniel Weld. A Softbot-based Interface to Internet. *Communications of the ACM*, 37(7):72–76, July 1994.

[Ferber 88] Jacques Ferber et Malik Ghallab. Problématique des univers multi-agents intelligents. *Deuxièmes Journées GRECO-PRC IA*, 1988.

[Fillmore 68] C.J. Fillmore. *Universals in Linguistic Theory*, pages 1–88. E. Bach and L.G. Harns eds, 1968.

[Findler 92] Nicolas V. Findler, Sarita Maini et Albert F. M. Yuen. SHRIF, a general-purpose system for heuristic retrieval of information and facts, applied to medical knowledge processing. *Information Processing & Management*, 28(2):219–240, 1992.

[Finin 94a] Tim Finin, Rich Fritzson, Don McKay et Robin McEntire. KQML - A Language and Protocole for Knowledge and Information Exchange. Rapport no. CS-94-02, CompputerScience Department, University of Maryland, UMBC, 1994.

[Finin 94b] Tim Finin, Richard Fritzson, Don McKay et Robin McEntire. KQML as an Agent Communication Language. *The Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94)*, Nov 1994.

[Fischer 89] Gerhard Fischer et Helga Nieper-Lemke. HELGON: Extending the Retrieval by Reformulation Paradigm. *Conference on Human Factors in Computing Systems (CHI'89) Proceedings*, pages 357–362, Austin, TX, 1989. SIGCHI Bulletin.

[Fox 87] Edward A. Fox. Development of the CODER System: a Testbed for Artificial Intelligence Methods in Information Retrieval. *Information Processing & Management*, 23(4):341–366, 1987.

[Fox 92] Edward A. Fox. *Tutorial on Knowledge-Based Information Retrieval.* SIGIR'92 (International Conference on Research and Development in Information Retrieval), 1992.

[Fuhr 92] Norbert Fuhr. Probabilistic Models in Information Retrieval. *The Computer Journal*, 35(3):243–255, 1992.

[Galliers 89] Julia Rose Galliers. The Positive Role of Conflict in Cooperative Multi-Agent Systems. Y. Demazeau et J.P. Müller, éditeurs, *Decentralized Artificial Intelligence - Proceedings of the First European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, 1989.

[Gasser 88] L. Gasser, C. Braganza et N. Herman. *Readings in Distributed Artificial Intelligence*, chapitre "Implementing Distributed AI systems using MACE", pages 445–450. Morgan Kaufmann, 1988.

[Gasser 91] Les Gasser et Toru Ishida. A Dynamic Organizational Architecture for Adaptive Problem Solving. *Proceedings of the Ninth National Conference on AI*, volume 1, pages 185–190. AAAI Press, The MIT Press, July 1991.

[Genesereth 86] M.R. Genesereth, M.L. Ginsberg et J.S. Rosenchein. Cooperation without Communication. *Proceedings of AAAI'86*, pages 51–57, 1986.

[Genesereth 92] M. R. Genesereth et R. E. Fikes. Knowledge Interchange Format. Reference manuel, version 3.0, report logic-92-1, Computer Science Department, Stanford University, 1992.

[Genesereth 94] Michael R. Genesereth et Steven P. Ketchpel. Software Agents. *Communications of the ACM*, 37(7):48–53, 147, July 1994.

[Georgeff 88] M. Georgeff. *Readings in Distributed Artificial Intelligence*, chapitre "Communication and Interaction in Multiagent Planning", pages 200–204. Morgan Kaufmann, 1988.

[Goble 92] C. Goble, M. O'Docherty, P.Crowther, M. Ireton, J. Oakley et C. Xydeas. The Manchester Multimedia Information System. *LNCS 580*, pages 39–55. (Proceedings of EDBT 92), 1992.

[Gorkani 94] Monika M. Gorkani et Rosalind W.Picard. Texture orientation for sorting photos "at a glance". *ICPR'94 proceedings*, pages 459–464. IEEE, 1994.

[Gupta 91] Amarnath Gupta, Terry Weymouth et Ramesh Jain. Semantic Queries with Pictures : the VIMSYS Model. G.M. Lohman, A. Sernadas et R. Camps, éditeurs, *Proceedings of the 17th International Conference on Very Large Date Bases*, pages 69–79, sept 1991.

[Hahn 90] Udo Hahn. Topic parsing: Accounting for text macro structures in full-text analysis. *Information Processing & Management*, 26(1):135–170, 1990.

[Halin 89] Gilles Halin. *Apprentissage pour la recherche interactive et progressive d'images : processus EXPRIM et prototype RIVAGE.* Thèse de Doctorat, Université de Nancy I, oct 1989.

[Halin 90] Gilles Halin, Marion Créhange et Pierre Kerekes. Machine learning and vectorial matching for an image retrieval model : EXPRIM and the system RIVAGE. *Proceedings of the Annual International ACM SIGIR Conference*, Bruxelles, 1990.

[Hamers 89] Lieve Hamers, Yves Hemeryck, Guido Herweyers, Marc Janssen, Hans Keters, Ronald Rousseau et André Vanhoutte. Similarity measures in scientometric research: The Jaccard index versus Salton's cosine formula. *Information Processing & Management*, 25(3):315–318, 1989.

[Harman 92] Donna Harman. Relevance Feedback Revisited. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 1–10, Copenhagen, 1992.

[Harman 93] Donna Harman. Overview of the First TREC Conference. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 36–47, Pittsburgh,PA, 1993.

[Hart 91] Paul J. Hart et Ronald E. Rice. Using Information from External Databases : Contextual Relationships of Use, Access Method, Task, Database Type, Organizational Differences, and Outcomes. *Information Processing & Management*, 27(2/3):461–479, 1991.

[Haton 91] Jean-Paul Haton, Nadjet Bouzid, François Charpillet et Marie-Christine Haton. *Le Raisonnement en Intelligence Artificielle.* Inter-Editions, 1991.

[Haussler 93] David Haussler. *Apprentissage Symbolique : une approche de l'Intelligence Artificielle*, volume 2, chapitre 20 - "Application de l'approche de Valiant à l'apprentissage de concepts", pages 563–589. Cépaduès-Éditions, 1993.

[Hayes-Roth 85] Barbara Hayes-Roth. A Blackboard Architecture for Control. *Artificial Intelligence*, 26:251–321, July 1985.

[Hearst 93a] Marti A. Hearst. TextTiling: a quantitative approach to discourse segmentation. Sequoia 2000 no. 93/24, University of California, Berkeley, 1993.

[Hearst 93b] Marti A. Hearst et Christian Plaunt. Subtopic structuring for full-lenght document access. *Proc. of the Sixteenth ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 59–68, Pittsburgh,PA, 1993.

[Hewitt 77] Carl E. Hewitt. Viewing Control Structures as Patterns of Passing Messages. *Artificial Intelligence*, 8:323–364, 1977.

[Hibino 95] Stacie Hibino et Elke A. Rundensteiner. A Visual Query Language for Identifying Temporal Trends in Video Data. IEEE Computer Society, éditeur, *First International Workshop on Multi-Media Database*, pages 74–81, 1995.

[Hintikka 62] J. Hintikka. *Knowledge and Belief.* Cornell University Press, Ithaca, NY, 1962.

[Hirata 92] K. Hirata et T. Kato. Query by Visual Example. *LNCS 580, Proc. of EDBT 92*, pages 56–71, 1992.

[Hjelsvold 95] R. Hjelsvold, R. Midtstraum et O. Sandstå. Searching and browsing a shared video database. *First International Workshop on Multi-Media Database*, pages 90–98. IEEE Computer Society, 1995.

[Hudrisier 83] Henri Hudrisier. *L'iconothèque.* La Documentation française, 1983.

[Huhns 87] Michael N. Huhns, Uttam Mukhopadhyay, Larry M Stephens et Ronald D. Bonnell. DAI for Document Retrieval : The MINDS projects. Michael N. Huhns, éditeur, *Distributed Artificial Intelligence*, pages 249–283, London, 1987. Pitman Morgan Kaufmann.

[Hung 94] Andy C. Hung. PVRG-MPEG CODEC 1.1. Rapport, Portable Video Research Group, Stanford University, 1994.

[Iivonen 95] Mirja Iivonen. Consistency in the selection of search concepts and search terms. *Information Processing & Management*, 31(2):173–190, 1995.

[Ingwersen 92a] Peter Ingwersen. *Information Retrieval Interaction.* Taylor Graham, 1992.

[Ingwersen 92b] Peter Ingwersen et Irene Wormell. Ranganathan in the Perspective of Advanced Information Retrieval. *Libri*, 42(3):184–201, 1992.

[ISO-8879 86] ISO-8879. Information processing - Text and Office Systems - Standard Generalized Markup Language (SGML). Rapport, ISO, 1986.

[ISO/DIS8613-1 88] ISO/DIS8613-1. Information processing - text and office systems - office document architecture (ODA) and interchange format. Rapport, ISO, 1988.

[ISO/IEC-CD-111172] ISO/IEC-CD-111172. Information Technology - Coding of Moving Pictures and Associated Audio - For Digital Storage. Rapport, ISO.

[Jones 72] Karen Sparck Jones. A Statistical Interpretation of Term Specificity and its Application in Retrieval. *Journal of Documentation*, 28:11–21, 1972.

[Jones 73] Karen Sparck Jones et M. Kay. *Linguistic and Information Science.* Academic Press, New York, 1973.

[Kheirbek 95] Ammar Kheirbek. *Modèle d'intégration d'un Système de Recherche d'Informations et d'un Système Hypermédia basé sur le formalisme des Graphes Conceptuels. Application au système RIME.* Thèse de Doctorat, Université Joseph Fourier, Grenoble, 29 mai 1995.

[Kim 90] Y. W. Kim et J. H. Kim. A Model of Knowledge Based Information Retrieval with Hierarchical Concept Graph. *Journal of Documentation*, 46(2):113–136, 1990.

[Konolige 86] K. Konolige. *A Deduction Model of Belief.* Pitmann Publishing, London and Morgan Kaufmann, San Mateo, CA., 1986.

[Kripke 63] S. Kripke. Semantical Analysis of Modal Logic. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

[Lâasri 89] Hassan Lâasri et Brigitte Maître. *Coopération dans un univers multi-agent basée sur le modèle du blackboard: Études et réalisations.* Thèse de Doctorat, Université de Nancy I, Fev 1989.

[Labrou 94] Yannis Labrou et Tim Finin. A Semantics Approach for KQML - a General Purpose Communication Language for Software Agents. *The Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94)*, Nov 1994.

[Lapierre 92] Jean-William Lapierre. *L'Analyse des Systèmes - L'application aux sciences sociales.* Syros, 1992.

[Lee 94a] Denis Lee, Ron Barber, Wayne Niblack, Myton Flickner, Jim Hafner et Dragutin Petkovic. Indexing for complex queries on a query-by-content image database. *ICPR'94 proceedings*, pages 142–149. IEEE, 1994.

[Lee 94b] Joon Ho Lee. Properties of Extended Boolean Models in Information Retrieval. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 182–190, 1994.

[Lee 94c] Joon Ho Lee, Myoung Ho Kim et Yoon Joon Lee. Ranking Documents in Thesaurus-based Boolean Retrieval Systems. *Information Processing & Management*, 30(1):79–91, 1994.

[Lejal 95] Frédéric Lejal. Évaluation de systèmes interactifs et évolutifs de recherche d'information multimédia. Mémoire de DEA no. 95-R-308, Université Henri Poincaré - CRIN, Nancy, 1995.

[Lesser 83] V. R. Lesser et D. D. Corkill. The Distributed Vehicle Monitoring Testbed. *AI Magazine*, 4:63–109, 1983.

[Levesque 84] Hector J. Levesque. A Logic of Implicit and Explicit Belief. *Proceedings of AAAI'84*, Austin,TX, 1984.

[Li 92] Zhuoxun Li, Hugh Davis et Wendy Hall. Hypermedia Links and Information Retrieval. *Proceedings of the 14th British Computer Society Research Colloquium on Information Retrieval*, Lancaster University, 1992.

[Lippman 89] Andrew Lippman et William Buetra. Coding Image Sequences for Interactive Retrieval. *Communications of the ACM*, 32(7):852–860, juillet 1989.

[Logan 94] Brian Logan, Steven Reece et Karen Sparck Jones. Modelling information retrieval agents with belief revision. *SIGIR'94 (International Conference on Research and Development in Information Retrieval)*, pages 91–100, 1994.

[Lu 90] Xin Lu. Document Retrieval: a Structural Approach. *Information Processing & Management*, 26(2):209–218, 1990.

[Macleod 90] Ian A. Macleod. Storage and Retrieval of Structured Documents. *Information Processing & Management*, 26(2):197–208, 1990.

[McCall 86] F. M. McCall et P. Willett. Criteria for the selection of search strategies in best match document retrieval systems. *International Journal of Man-Machine Studies*, 25:317–326, 1986.

[McGill 76] M. J. McGill, L.Smith, S. Davidson et T. Noreault. Syracuse Information Retrieval Experiment (SIRE): Design of an On-Line Bibliographic Retrieval System. *SIGIR Forum*, 10(4):37–44, 1976.

[McMath 89] Charles F. McMath, Robert S. Tamaru et Roy Rada. A Graphical Thesaurus-Based Information Retrieval System. *International Journal of Man-Machine Studies*, 31:121–147, 1989.

[Mechkour 95] Mourad Mechkour. *EMIR*^{2}. *Un modèle étendu de représentation et de correspondance d'images pour la recherche d'informations. Application à un corpus d'images historiques.* Thèse de Doctorat, Université Joseph Fourier, Grenoble, 1995.

[Meghini 91] C. Meghini, F. Rabitti et C. Thanos. Conceptual Modeling of Multimedia Documents. *Computer*, pages 23–30, Octobre 1991.

[MHEG 93] MHEG. Information technology – Coded representation of Multimedia and Hypermedia Information Objects. Rapport no. CD 13522-1, ISO/IEC, 1993.

[Michalski 83] Ryszard Stanislaw Michalski, Jaime Guillermo Carbonell et Tom Michael Mitchell, éditeurs. *Machine learning : an artificial intelligence approach.* Tioga, Palo Alto, 1983.

[Minsky 88] Marvin Minsky. *La Société de l'Esprit.* Inter-Editions, Paris, 1988.

[Mitchell 82] Tom M. Mitchell. Generalization as search. *Artificial Intelligence*, 18:203–226, 1982.

[Motro 86] A. Motro. Constructing queries from tokens. *Proceedings of the ACM/SIGMOD Conference*, pages 120–131, 1986. Sigmod Record Vol.15, num 2, 1986.

[Moulin 92] Bernard Moulin. Conceptual-graph approach for the representation of temporal information in discourse. *Knowledge-Based Systems*, 5(3):183–192, septembre 1992.

[Moulin 94] Bernard Moulin. *Soft Computing: Fuzzy Logic, Neural Networks and Distributed Artificial Intelligence*, chapitre 12- "Collaborative work based on multiagent architectures: a methodological perspective", pages 261–296. PTR Prentice Hall, 1994.

[Napoli 92] Amedeo Napoli. Représentations à objets et raisonnement par classification en intelligence artificielle. Thèse de Doctorat d'État ès Sciences Mathématiques, spécialité Informatique, 1992.

[Nie 88] Jian-Yun Nie. An Outline of a General Model for Information Retrieval Systems. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 495–506, 1988.

[Nie 89] Jian-Yun Nie. An information retrieval model based on modal logic. *Information Processing & Management*, 25(5):477–491, 1989.

[Nie 94] Jian-Yun Nie, Jean Vaucher et Nicolas Anquetil. A Flexible Architecture for Collaborative Information Retrieval. *Intelligent Multimedia Information Retrieval Systems and Management - RIAO 94 conference proceedings.*, pages 600–611, New York, oct 1994.

[Nii 86] H. P. Nii. Blackboards Systems: the Blackboard Model of Problem-solving and the Evolution of Blackboard Architectures. *AI Magazine*, 7(3):39–53, 1986.

[Oates 94] Tim Oates, M.V. Nagendra Prasad et Victor R. Lesser. Cooperative Information Gathering : A Distributed Problem Solving Approach. Rapport, Dept of Computer Science, University of Massachusetts, 1994.

[Oates 95] Tim Oates, M.V. Nagendra Prassad, Victor R. Lesser et Keith Decker. A Distributed Problem Solving Approach to Cooperative Information Gathering. *Working Notes of the AAAI'95 Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments*, pages 133–137, 1995.

[Oliver 95] Ron Oliver. Interactive Information Systems: Information Access and Retrieval. *The Electronic Library*, 13(3):187–193, June 1995.

[Oomoto 93] Eitetsy Oomoto et Katsumi Tanaka. OVID : Design and Implementation of a Video-Object Database System. *IEEE Transactions on Knowledge and Data Engineering*, 5(4):629–643, August 1993.

[Ousterhout 94] John K. Ousterhout. *TCL and the TK Toolkit*. Addison Wesley, 1994.

[Pacey 79] Philip Pacey. Information technology and the universal availability of images. *The American Archivist*, 42(4), 1979.

[Paice 93] Chris D. Paice et Paul A. Jones. The identification of important concepts in highly structured technical papers. *Proc. of the Sixteenth ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 69–78, Pittsburgh,PA, 1993.

[Pair 88] Claude Pair, René Schott et Roger Mohr. *Construire les algorithmes : les améliorer, les connaître, les évaluer*. Dunod informatique, 1988.

[Picard 93] Rosalind W. Picard et T. Kabir. Finding Similar Patterns in Large Images Databases. *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 161–164, Minneapolis, MI, 1993.

[Picard 95] R. W. Picard et T. P. Minka. Vision texture for annotation. *Multimedia System*, 3:3–14, 1995.

[Pitrat 90] Jacques Pitrat. *Métaconnaissance : futur de l'intelligence artificielle*. Hermès, Paris, 1990.

[Quint 94] Vincent Quint. *Le Traitement Électronique du Document*, chapitre 1 - "Édition de documents structurés", pages 11–47. ADBS Éditions, 1994.

[Rabitti 91] F. Rabitti et P. Savino. Automatic Image Indexation and Retrieval. *RIAO 91*, pages 864–884, 1991.

[Rijsbergen 86a] C. J. Van Rijsbergen. A new theoretical framework for information retrieval. *ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 194–200, Pisa, 1986.

[Rijsbergen 86b] C. J. Van Rijsbergen. A non-classical logic for information retrieval. *The Computer Journal*, 29(6):481–485, 1986.

[Rijsbergen 92] C. J. Van Rijsbergen. Probabilistic Retrieval Revisited. *The Computer Journal*, 35(3):291–298, 1992.

[Robertson 82] S.E. Robertson, M.E. Maron et W.S.Cooper. Probability of relevance : A unification of two competing models for document retrieval. *Information Technology : Research and Development*, 1(1):1–21, 1982.

[Rocchio 71] J. J. Jr Rocchio. *Relevance Feedback in Information Retrieval*, chapitre 14. Prentice-Hall, Englewood Cliffs, NJ, 1971. (première édition en 1965).

[Rumbaugh 91] James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy et William Lorensen. *Object-Oriented Modeling and Design.* Prentice Hall, 1991.

[Rus 94] D. Rus et D. Subramanian. Designing structure based information agents. *Working Notes of the AAAI Spring Symposium on Software Agents*, pages 79–86, 1994.

[Salton 65] G. Salton et M.E. Lesk. The SMART Automatic Document Retrieval System - An Illustration. *Communications of the ACM*, 8(6):613–620, 1965.

[Salton 68] G. Salton. *Automatic Information Organization and Retrieval.* McGraw-Hill, New York, 1968.

[Salton 83a] G. Salton et M. McGill. *Introduction to Modern Information Retrieval.* McGraw Hill International Company, New York, 1983.

[Salton 83b] Gerard Salton, Edward A. Fox et Harry Wu. Extended Boolean Information Retrieval. *Communications of the ACM*, 26(12):1022–1036, decembre 1983.

[Salton 85] G. Salton. A Note on Information Retrieval Models and Theories. *Proceedings of RIAO 85 (Recherche d'Information Assistée par Ordinateur)*, 1985.

[Salton 90] G. Salton et C. Buckley. Improving Retrieval Performance by Relevance Feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.

[Salton 93] G. Salton, J. Allan et C. Buckley. Approaches to passage retrieval in full text information systems. *Proc. of the Sixteenth ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 49–58, 1993.

[Salton 94] Gerard Salton, James Allan et Chris Buckley. Automatic structuring and retrieval of large text files. *Communications of the ACM*, 37(2):97–108, feb 1994.

[Schiel 89] Ulrich Schiel. Abstractions in Semantic Networks: Axiom Schemata for Generalization, Aggregation and Grouping. *SIGART Newsletter*, 107:25–26, 1989.

[Searle 69] J.R. Searle. *Speech Acts: An Essay in the Philosophy of Language.* Cambridge University Press, Cambridge - England, 1969.

[Shafer 76] G. Shafer. *A Mathematical Theory of Evidence*. Princetown University Press, Princetown, 1976.

[Simon 73] Herbert A. Simon. The Structure of Ill-Structured Problems. *Artificial Intelligence*, 4:181–201, 1973.

[Simonnot 94] Brigitte Simonnot et Malika Smaïl. Flexible model for multimedia document retrieval. Rapport no. 94-R-229, CRIN, Nancy, France, 1994.

[Simonnot 95a] B. Simonnot. A cooperation model for video document retrieval. *Proceedings of Storage and Retrieval for Image and Video Databases III*, volume 2420, pages 307–317, San Jose, CA, Feb 1995. Electronic Imaging Symposium.

[Simonnot 95b] B. Simonnot et M. Smaïl. Modèle flexible pour la recherche interactive de documents multimédias. *XIIIe Congrès Inforsid (INFormatique des Organisations et Systèmes d'Information et de Décision)*, pages 165–178, Grenoble, juin 1995.

[Smaïl 93a] M. Smaïl. Case-Based Information Retrieval. *Proceedings First European Workshop on Case-Based Reasoning*, pages 71–75, Kaiserslautern (Germany), novembre 1993. Springer Verlag.

[Smaïl 93b] M. Smaïl et M. Créhange. Adaptation par cas des stratégies de Recherche d'Information. *Actes Journées Bases de Données Avancées 93*, pages 155–174, Toulouse (France), septembre 1993.

[Smaïl 94] M. Smaïl. Raisonnement à base de cas pour une recherche évolutive d'information; Prototype Cabri-n. Vers la définition d'un cadre d'acquisition de connaissances. Thèse de Doctorat de l'Université Henri Poincaré, Nancy I, octobre 1994.

[Smith 80] Randall G. Smith. The Contract Net Protocol : High Level Communication and Control in a Distributed Problem Solver. *IEEE transactions on Computer*, C-29(12):1104–1113, dec. 1980.

[Smith 88] R. Smith. *Readings in Distributed Artificial Intelligence*, chapitre "The Contract Net Protocole: High-level Communication and Control". Morgan Kaufmann, 1988.

[Sowa 84] John F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, Reading, MA, 1984.

[Sowa 91] John F. Sowa, éditeur. *Principles of Semantic Networks*. Morgan Kaufmann Publishers Inc., 1991.

[Sowa 95] John F. Sowa. Syntax, Semantics, and Pragmatics of Contexts. *Conceptual structures: Applications, Implementations and Theory, (ICCS'95), LNAI 954*, pages 1–15. Springer Verlag, 1995.

[Subtil 95] Pascal Subtil. *Amélioration de la flexibilité dans les bases d'objets flous. Expérimentation dans le cadre du logiciel FIRMS*. Thèse de Doctorat, Université Henri Poincaré-Nancy I, 1995.

[Swanberg 92] Deborah Swanberg, Chiao Fe Shu et Ramesh Jain. System for Content-Based Retrieval. P.Venkat Rangan, éditeur, *Lecture Notes in Computer Science*, volume 712. Springer-Verlag, nov 1992.

[Swanberg 93] Deborah Swanberg, Chiao-Fe Shu et Ramesh Jain. Knowledge Guided Parsing in Video Databases. Wayne Niblack, éditeur, *Proceedings of Storage and Retrieval for Image and Video Databases*, volume 1908, pages 13–24. SPIE, feb 1993.

[Syraca 88] K. S. Syraca. Resolving Goal Conflicts via Negociation. *Proceedings of AAAI*, pages 245–250, 1988.

[Thompson 89] R. H. Thompson et W. B. Croft. Support for Browsing in Intelligent Text Retrieval System. *Int. J. Man Machine Studies*, 30:639–668, 1989.

[Thompson 90] Paul Thompson. A combination of expert opinion approach to probabilistic information retrieval, part 1: The conceptual model. *Information Processing & Management*, 26(3):371–382, march 1990.

[Turner 93a] Elise H. Turner. Selecting Information to Communicate. Rapport, Dept of Computer Science, Univ. of New Hampshire, Kingsbury Hall, Durham, MH, 1993.

[Turner 93b] James M. Turner. Subject access to pictures: Considerations in the surrogation and indexing of visual documents for storage and retrieval. *Visual Resources*, IX:241–271, 1993.

[Turner 94] James M. Turner. Le choix spontané de termes pour l'indexation des images : résultats de recherche. *L'industrie de l'Information en Transition - 22ème Conf. de l'Association canadienne pour les sciences de l'information*, pages 376–393, 1994.

[Turtle 92] Howard R. Turtle et W. Bruce Croft. A Comparison of Text Retrieval Models. *The Computer Journal*, 35(3):279–290, 1992.

[Wermelinger 95] Michel Wermelinger. Conceptual Graphs and First-Order Logic. *Conceptual structures: Applications, Implementations and Theory, (ICCS'95), LNAI 954*, pages 323–337. Springer Verlag, 1995.

[Werner 89] Eric Werner. Distributed Cooperation Algorithms. Y. Demazeau et J.P. Müller, éditeurs, *Proceedings of the First European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, pages 17–31, 1989.

[White 94] James E. White. Telescript Technology: The Foundation for the Electronic Marketplace. General Magic White Paper (http://www.genmagic.com), 1994.

[Wiederhold 92] Gio Wiederhold. Mediators in the Architecture of Future Information Systems. *IEEE Computer*, pages 38–48, mars 1992.

[Winston 87] M.E. Winston, R. Chaffin et D. Herrmann. A Taxonomy of Part-Whole Relations. *Cognitive Science*, 11:417–444, 1987.

[Wooldridge 95] Michael Wooldridge et Nicholas R. Jennings. Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.

[Zhang 92] Chengqui Zhang. Cooperation Under Uncertainty in Distributed Expert Systems. *Artificial Intelligence*, 56:21–69, 1992.

[Zhang 95] H. Zhang, S. W. Smoliar et J. H. Wu. Content-based Video Browsing Tools. *Multimedia Computing and Networking, Vol 2417-35*. IS & T - SPIE, 1995.

[Zipf 49] George Kingsley Zipf. *Human Behavior and the Principle of Least Effort: an Introduction to Human Ecology.* Addison-Wesley, 1949.

BIBLIOTHEQUE UNIVERSITAIRE NANCY I SCIENCES

Nom : SIMONNOT

Prénom : Brigitte

DOCTORAT de l'UNIVERSITE HENRI POINCARE, NANCY-I

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER

Nancy, le 29 JAN. 1996 DHP

Le Président de l'Université

J.P. FINANCE

3C-F9

UNIVERSITÉ HENRI POINCARÉ - NANCY 1
24-30, rue Lionnois B.P.3069 54013 Nancy cedex
Tél : 83 32 81 81 Fax : 83 32 95 90
Adresse électronique : _____ @presn1.u-nancy.fr

Modélisation Multi-agents d'un système de recherche d'information multimédia à forte composante vidéo

Résumé

Les techniques mises en œuvre pour retrouver et présenter des informations multimédias font appel à des connaissances et des savoir-faire hétérogènes : celles du documentaliste, celles des spécialistes de chaque média impliqué, celles des spécialistes du domaine de l'application, etc.) dont la plupart évoluent rapidement. Nous nous intéressons plus particulièrement à la recherche d'information dans des collections de documents vidéo, qui sont complexes et riches en information.

L'approche documentaire traditionnelle exploite des descriptions des documents pour effectuer la recherche. Dans le modèle de description que nous proposons, les différents aspects de connaissance qui peuvent être exploités sont modélisés par facettes, chaque facette correspondant au point de vue d'un expert sur les documents. Nous avons retenu le formalisme des graphes conceptuels pour représenter ces descriptions : plus riches que des descriptions par mots-clés, les graphes conceptuels permettent de représenter des nuances et des concepts complexes. Toutefois, le système n'impose pas à l'utilisateur de formuler sa requête dans le formalisme que connaît l'application, mais sous la forme d'une liste de mots-clés, qu'il peut éventuellement pondérer pour indiquer leurs importances relatives.

Nous avons conçu une société d'agents qui se met au service de l'utilisateur pour l'aider dans sa recherche d'information : les agents de recherche établissent leur stratégie de manière dynamique, en fonction de la situation. Ils s'appuient sur les jugements de l'utilisateur à propos des documents proposés non seulement pour améliorer leur recherche, mais pour évaluer leurs performances. Cette «société» d'agents doit être capable d'évoluer. Définir les agents d'un tel système et leur organisation dynamique, étudier les modes d'interactions et de coopération qui peuvent s'établir entre eux durant une session de recherche, tel a été l'objet principal de notre travail. Les idées que nous avons développées ont été implantées dans le prototype multi-agents SAFARI-Vidéo.

Mots-clés : recherche d'information interactive, modèle de représentation des connaissances par facettes, documents multimédias et vidéos, architecture multi-agents, recherche d'information coopérative.

MultiAgent Modelling of a Information Retrieval System for Images and Videos Collections

Abstract

Techniques to retrieve and present multimedia documents require heterogeneous knowledge and know-how (librarians, media specialists, application domain specialists, etc). Most of these techniques are quickly evolving, as video processing is an example. We are particularly interested in video retrieval.

Librarians searching for documents traditionally use some surrogates of them, represented by lists of keywords. We adopt the conceptual graph theory of Sowa to represent documents descriptions. In our model, the descriptions are modelized by facets, each facet matching the viewpoint of an expert on the collection. During the retrieval process, documents descriptions can be exploited by a vectorial or a conceptual graph matching, depending on the user's need.

Our approach consists in conceiving a set of agents to assist users in their information retrieval. This "society of agents" must be able to adjust its behaviour to various information needs. We define the main agents of such a system and their organisation, studying their interactions to improve the retrieval. This model has been implemented a multiagent prototype, Video-SAFARI.

Keywords: interactive information retrieval, facets of knowledge, multimedia and video documents, multiagent architecture, cooperative information retrieval.