



HAL
open science

Un modèle général de recherche d'information : Application à la recherche de documents techniques par des professionnels

Leila Kefi-Khelif

► **To cite this version:**

Leila Kefi-Khelif. Un modèle général de recherche d'information : Application à la recherche de documents techniques par des professionnels. Autre [cs.OH]. Université Joseph-Fourier - Grenoble I, 2006. Français. NNT : . tel-00300459

HAL Id: tel-00300459

<https://theses.hal.science/tel-00300459>

Submitted on 18 Jul 2008

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.

UNIVERSITE JOSEPH FOURIER-GRENOBLE I
U.F.R. INFORMATIQUE ET MATHEMATIQUES APPLIQUEES

THESE

Pour obtenir le titre de

DOCTEUR DE L'UNIVERSITE JOSEPH FOURIER – GRENOBLE I

Discipline : Informatique

Présentée et soutenue publiquement le 27 octobre 2006 par

Leïla Kefi-Khelif

TITRE

**Un modèle général de recherche d'information :
Application à la recherche de documents
techniques par des professionnels**

Directeurs de thèse : Catherine Berrut et Eric Gaussier

Composition du jury :

Président : Mme Dominique RIEU
Rapporteurs : M. Patrick BOSC
: M. Jean-Marie PINON
Examineurs : Mme Catherine BERRUT
: Mr Eric GAUSSIER

Thèse préparée au sein de l'équipe MRIM du laboratoire CLIPS-IMAG
(Communication Langagière et Interaction Personne-Système)
Université Joseph Fourier – Grenoble I

*« Now this is not the end. It is not even the beginning of the end.
But it is, perhaps, the end of the beginning »
Winston Churchill*

*A mes parents
A Khaled
A mon petit Wissem*

Remerciements

C'est un grand plaisir pour moi de remercier toutes les personnes qui ont permis à ce travail d'être ce qu'il est.

Je remercie tout d'abord Mme Dominique RIEU qui m'a fait l'honneur de présider le jury de cette thèse.

Je tiens ensuite à remercier M. Jean-Marie PINON ainsi que M. Patrick BOSCH pour avoir accepté de rapporter ce travail, ainsi que pour l'intérêt qu'ils ont manifesté à l'égard de ce travail de thèse.

Je voudrais également remercier Mme Catherine Berrut et M. Eric Gaussier pour avoir co-encadré mon travail. Leur compétence, leur patience et leurs encouragements m'ont été des plus précieux durant ce travail et ce depuis le début de mon DEA.

Je remercie aussi vivement tous les membres du laboratoire CLIPS pour leur accueil ainsi que tous les membres de l'équipe MRIM pour leurs remarques et leurs questions pertinentes lors des réunions, ainsi que tou(te)s les participant(e)s à l'évaluation pour le temps qu'il lui ont généreusement consacré.

Merci également à tous les thésards de l'équipe MRIM et du laboratoire CLIPS : Caro, Delphine, Helga, Loïc, Te, Hung, Stéphane, Dima, Olfa ainsi que nos prédécesseurs Lizbeth, Moh, Mbarek, Jean, Quoc et les autres, pour les bons moments passés au labo et pour les pauses très professionnelles à la cafétéria.

Sans oublier Ouafa qui a partagé mes doutes et mes périodes de stress et Cécile qui me les a fait oublier...

Enfin, je voudrais remercier toute ma famille pour leurs encouragements constants: karim, Hinda, Nizar, Aïcha, Hend, Rym, Emilie et mes beaux parents. Une mention spéciale à mes parents pour tellement de choses que je ne peux en faire la liste. Mes plus sincères remerciements à Khaled pour sa patience, sa compréhension et son soutien indéfectible et inconditionnel...

Et je n'oublie pas de remercier mon petit Wissem qui a attendu patiemment que sa maman termine la rédaction de ce manuscrit avant de se décider à pointer le bout de son nez et à embellir ses journées...

Je remercie enfin toutes les personnes que j'ai oublié de remercier ici.

Table des matières

INTRODUCTION	1
1. PROBLEMATIQUE	3
2. OBJECTIFS	4
3. APPROCHE	5
3.1. Proposition d'un modèle général de recherche d'information	5
3.2. Application à la recherche d'information technique par des professionnels	6
4. PLAN DU MANUSCRIT	7
PARTIE 1. ÉTAT DE L'ART	11
CHAPITRE I. LA RECHERCHE D'INFORMATION : MODELISATION ET QUALITE	15
1. SYSTEME DE RECHERCHE D'INFORMATION : ARCHITECTURE GENERALE	15
2. L'INDEXATION	17
2.1. Langages simples	18
2.2. Langages complexes	19
2.3. Bilan	22
3. L'INTERROGATION	22
3.1. Importance des unités de la requête	23
3.2. Incertitude dans l'expression de la requête	24
4. QUALITE EN RECHERCHE D'INFORMATION : VALIDATION DE L'INDEXATION EN AMONT	25
4.1. Evaluation globale du système	26
4.2. Evaluation de l'indexation	26
4.3. Synthèse des mesures pour la validation d'indexation	29
5. CONCLUSION	30
CHAPITRE II. DOCUMENTS TECHNIQUES ET USAGES: UN ACCES PAR LE GRAPHIQUE	31
1. DESCRIPTION DU CONTEXTE : DOCUMENTS ET UTILISATEURS	31
1.1. Le document technique : définition et caractéristiques	31
1.2. Les utilisateurs professionnels : caractéristiques	35
1.3. Bilan	37
2. LE GRAPHIQUE : UN POINT D'ACCES AU DOCUMENT TECHNIQUE	37
2.1. Un exemple d'application dans la recherche des documents techniques	38
2.2. Graphiques et mémoire	39
2.3. Indexation des graphiques et des images : un tour d'horizon	42
3. CONCLUSION	49
PARTIE 2. PROPOSITION D'UN MODELE DE RECHERCHE D'INFORMATION FONDE SUR DES CRITERES D'OBLIGATION ET DE CERTITUDE	51
CHAPITRE III. PRELIMINAIRES : DEFINITIONS ET NOTATIONS	55
1. CORPUS DE DOCUMENTS	55
2. VOCABULAIRE	56
3. INDEXATION DES DOCUMENTS	57

4. FORMULATION DE LA REQUETE	58
5. CORRESPONDANCE ENTRE LA REQUETE ET LES DOCUMENTS	60
6. RECAPITULATIF	60
7. CONCLUSION	61
CHAPITRE IV. UN MODELE DE RECHERCHE D'INFORMATION POUR DES LANGAGES COMPLEXES	63
1. SPECIFICITES DU MODELE	63
2. VUE D'ENSEMBLE DU MODELE	65
2.1. Vocabulaire	65
2.2. Vocabulaire d'indexation	66
2.3. Vocabulaire d'interrogation	67
2.4. Fonctions utiles	69
2.5. La fonction de correspondance	69
3. PRESENTATION DETAILLEE DU MODELE	70
3.1. Vocabulaire	70
3.2. Indexation des documents	73
3.3. Formulation de la requête	76
3.4. Correspondance entre la requête et les documents	83
4. UNE APPROCHE POUR LA REFORMULATION DE LA REQUÊTE	88
4.1. Définition des classes de pertinence	89
4.2. Distributions possibles des documents jugés pertinents dans les classes	93
4.3. Description des actions pour la reformulation de la requête	95
5. CONCLUSION	97
<u>PARTIE 3. INSTANCIATION DU MODELE</u>	99
CHAPITRE V. VERS UNE DESCRIPTION EXHAUSTIVE DES GRAPHIQUES	103
1. ETUDE DU CORPUS : QUELLE DESCRIPTION DU GRAPHIQUE RETENIR ?	103
1.1. Données intrinsèques du graphique	103
1.2. Données sémantiques enrichissant le graphique	107
1.3. Impact de la mémorisation visuelle des graphiques	109
1.4. Récapitulatif	110
2. UNE REPRESENTATION MULTI-VUES DU GRAPHIQUE TECHNIQUE	111
2.1. Vue physique	111
2.2. Vue structurelle	111
2.3. Vue symbolique	113
2.4. Vue opératoire	114
2.5. Vue mémoire visuelle	116
3. CONCLUSION	117
CHAPITRE VI. MODELE DE RECHERCHE DE GRAPHIQUES TECHNIQUES	119
1. VOCABULAIRE	119
1.1. Vue structurelle	120
1.2. Vues symbolique et opératoire	121
1.3. Vue mémoire visuelle	123
1.4. Vocabulaire global	126
2. INDEXATION	126
3. FORMULATION DE LA REQUETE : 3 LANGAGES	128
3.1. Langage textuel	128
3.2. Langage graphique	131
3.3. Langage mixte	133
4. CORRESPONDANCE ENTRE LA REQUETE ET LES DOCUMENTS	136
5. CONCLUSION	137

PARTIE 4. MISE EN OEUVRE ET VALIDATIONS **139**

CHAPITRE VII. INDEXATION TEXTUELLE ET VISUELLE DES GRAPHIQUES **143**

1. INDEXATION TEXTUELLE : PROCESSUS ET VALIDATION	143
1.1. Caractérisation du texte commentant un graphique	144
1.2. Repérage des termes d'indexation et construction de l'index	147
1.3. Processus automatique d'indexation	153
1.4. Validation qualitative de l'indexation par le texte	156
2. INDEXATION VISUELLE : PROPOSITION D'UN PROTOCOLE D'EVALUATION	167
2.1. Le processus d'indexation manuel	167
2.2. Description de l'évaluation	168
2.3. Exploitation des résultats de l'évaluation	170
2.4. Conclusion sur l'indexation visuelle	173
3. CONCLUSION	173

CHAPITRE VIII. FORMULATION DE LA REQUETE ET CORRESPONDANCE : VALIDATION DE L'AJOUT DES CRITERES D'OBLIGATION ET DE CERTITUDE **175**

1. MODÈLE OPÉRATIONNEL CONSIDÉRÉ	175
2. DESCRIPTION DES EXPÉRIENCES	178
2.1. Collection de test	178
2.2. Expériences réalisées	179
3. RÉSULTATS DES EXPÉRIENCES	181
3.1. Comparaison interne	181
3.2. Comparaison avec les modèles classiques de recherche d'information	183
4. CONCLUSION	188

CONCLUSION ET PERSPECTIVES **189**

1. SYNTHÈSE ET CONTRIBUTIONS	191
2. PERSPECTIVES	193

ANNEXES **195**

BIBLIOGRAPHIE **207**

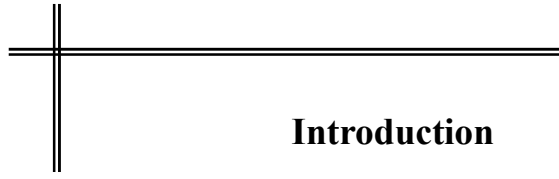
Table des figures

Figure 1 Schéma récapitulatif de l'approche	5
Figure 2 Architecture d'un SRI	17
Figure 3 Représentation de l'information : « opacité tissulaire au niveau du poumon »	19
Figure 4 Représentation du concept complexe « appendicite aiguë »	20
Figure 5 Un exemple d'image indexée selon EMIR ²	21
Figure 6 Exemple d'un document et de sa représentation sous forme d'un graphe conceptuel	21
Figure 7 Exactitude et consistance de l'indexation	27
Figure 8 Les deux points de vue de l'indexation	27
Figure 9. Exemple de propagation de l'information dans la hiérarchie d'un document technique	33
Figure 10. Exemple de dépendance entre partie textuelle et partie graphique	33
Figure 11 Structure d'un document intégrant un élément non textuel	39
Figure 12 Passage de l'information dans le système mnésique	40
Figure 13 Les segments extraits d'un graphique représentant une pince.	43
Figure 14 Exemple d'image et de requête	44
Figure 15 Exemples de symboles graphiques et leur F-signature dans $[0, \Pi]$	45
Figure 16 PICTION : correspondance entre les noms et les visages dans l'image	47
Figure 17 Liens entre des zones de la carte géographique et des fragments de son commentaire	48
Figure 18 Le corpus C de documents D_i	55
Figure 19 Exemple de corpus de documents C	56
Figure 20 Le vocabulaire \mathcal{V}	56
Figure 21 Le vocabulaire d'indexation \mathcal{V}_D rattaché au vocabulaire \mathcal{V}	57
Figure 22 Le corpus indexé CI	58
Figure 23 Le vocabulaire d'interrogation \mathcal{V}_Q rattaché au vocabulaire \mathcal{V}	59
Figure 24 La requête q	59
Figure 25 Schéma récapitulatif	61
Figure 26 Le vocabulaire \mathcal{V} est formé par les ensembles d'entités \mathcal{T} et de relation \mathcal{R}	66
Figure 27 L'ensemble Δ des identifiants rattachés aux entités de \mathcal{T}	66
Figure 28 Le vocabulaire d'indexation \mathcal{V}_D	67
Figure 29 Les ensembles d'identifiants d'interrogations Δ_Q et de relations d'interrogation \mathcal{R}_Q	68
Figure 30 Le vocabulaire d'interrogations \mathcal{V}_Q	69
Figure 31 Les fonctions : de l'unité d'interrogation à l'entité et la relation	69
Figure 32 Exemple d'association deux à deux des entités de DI_i et de q	70
Figure 33 Le document D_5 et le document indexé DI_5	75
Figure 34 Exemple de corpus indexé	75
Figure 35 Classification des entités d'une requête q selon leurs critères	81
Figure 36 Répartition des unités d'une requête q selon les critères de leurs relations	82
Figure 37 Vérification de l'appartenance des entités de q à un document DI	85
Figure 38 Documents indexés jugés pertinents par la correspondance basée sur l'appartenance	85
Figure 39 Schématisation du contenu obligatoire d'un document pertinent pour q	86
Figure 40 Le document D_7 serait considéré comme pertinent pour q si la correspondance s'appuyait sur l'appartenance des unités de q au document indexé	86
Figure 41 Correspondance un à un entre identifiants d'entité de DI_i et de q	87
Figure 42 Exemple de répartition des documents jugés pertinents par l'utilisateur (en gris) dans les classes de pertinence	93
Figure 43 Le graphique est un objet structuré	104

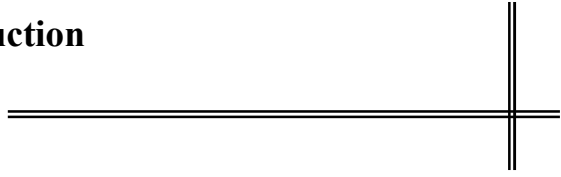
Figure 44 Exemple d'association de forme à un objet du graphique _____	105
Figure 45 Exemple de représentation spatiale des faces d'un objet graphique multi-faces ____	105
Figure 46 Un exemple de graphique et de sa représentation géométrique _____	106
Figure 47 Un exemple de graphique avec illustratifs (zoom, flèche et énumération) _____	106
Figure 48 Exemple illustrant l'aspect descriptif du graphique _____	107
Figure 49 Exemple illustrant l'aspect opératoire du graphique _____	109
Figure 50 Exemple d'encodage possible d'un graphique visualisé _____	110
Figure 51 Exemple de vue structurelle d'un graphique _____	112
Figure 52 Exemple de vue descriptive d'un graphique _____	114
Figure 53 Exemple de vue opératoire d'un graphique _____	116
Figure 54 Exemple de vue mémoire visuelle d'un graphique _____	117
Figure 55 Le vocabulaire V_{GRIM} _____	120
Figure 56 Schématisation du langage GRIM _____	127
Figure 57 Exemple d'interface pour la saisie d'une requête textuelle _____	129
Figure 58 Exemple de requête graphique _____	131
Figure 59 Blocs de textes correspondant à un graphique _____	145
Figure 60 Exemple d'utilisation de l'interface graphique de GATE (repérage des termes respectant les patrons COM et ACT) _____	154
Figure 61 Architecture de notre processus d'indexation _____	155
Figure 62 Exemple de document du corpus _____	160
Figure 63 Simplification par suppression de la vue structurelle _____	161
Figure 64 Schématisation du langage $GRIM_{Texte}$ _____	162
Figure 65 Exemple d'interface de validation de l'indexation _____	164
Figure 66 Les trois étapes de l'expérience _____	169
Figure 67 Vue globale du modèle considéré dans l'évaluation _____	176
Figure 68 Courbe de rappel/précision pour la comparaison interne _____	182
Figure 69 Courbe de rappel/précision pour les modèles de base _____	183
Figure 70 Courbe de rappel/précision pour la première évolution _____	185
Figure 71 Courbe de rappel/précision pour la deuxième évolution _____	186
Figure 72 Schéma récapitulatif de l'approche _____	191
Figure 73 Représentation vectorielle de deux documents d_1 et d_2 et une requête q dans l'espace (t_1, t_2, t_3) _____	198
Figure 74 Schématisation du modèle EMIR ² _____	201
Figure 75 Inutilité d'une vue perceptive _____	202
Figure 76 Prise en compte d'une vue opératoire _____	204
Figure 77 Impact de la mémoire visuelle sur la représentation spatiale d'un graphique _____	205
Figure 78 Prise en compte de la vue mémoire visuelle _____	205
Figure 79 Une synthèse de notre modèle _____	206

Liste des tableaux

Tableau 1 Tableaux récapitulatifs des mesures de qualité d'indexation existantes _____	28
Tableau 2 Tableau comparatif des méthodes présentées _____	49
Tableau 3 Récapitulatif des actions sur la requête en fonction de la répartition des documents jugés pertinents dans les classes _____	95
Tableau 4 Les critères par défaut, selon l'étiquette de l'entité, dans le cas du langage textuel _____	130
Tableau 5 Les critères par défaut, selon l'étiquette de l'entité, dans le cas du langage mixte _____	134
Tableau 6 Les critères par défaut, selon le type de relation, dans le cas du langage mixte _____	135
Tableau 7 Tableaux récapitulatifs des mesures (anciennes et ajoutées) de qualité d'indexation _____	158
Tableau 8 Mesures qualitatives d'indexation à langage complexe _____	160
Tableau 9 Résultats globaux de l'évaluation _____	164
Tableau 10 Valeurs de l'exactitude de la construction des entités de $\mathcal{T}_{\text{Texte}}$ par type d'étiquette _____	166
Tableau 11 Valeurs de la justesse de la construction des unités de $\mathcal{U}_{\text{Texte}}$ par type de relation _____	166
Tableau 12. Tableau récapitulatif des expériences réalisées _____	179
Tableau 13. Comparaison interne du modèle proposé _____	181
Tableau 14. Précision moyenne et à 5 et 10 documents avec les modèles de base _____	183
Tableau 15. Précision moyenne et à 5 et 10 documents avec les modèles existants étendus afin de prendre en compte le critère de certitude/incertitude _____	184
Tableau 16. Précision moyenne et à 5 et 10 documents avec les modèles existants étendus afin de prendre en compte les deux critères conjointement _____	186



Introduction



La recherche d'information est « l'opération qui permet à partir d'une expression des besoins en information d'un utilisateur de retrouver l'ensemble des documents contenant l'information recherchée » [Salton,83]. Cela passe par une définition de modèles de recherche d'information intégrant une représentation des documents, une représentation du besoin (ou requête) et un appariement entre le document et la requête. Plusieurs modèles ont été proposés; leur principale différence réside dans la façon de représenter les documents et la requête et de les mettre en correspondances, et ce généralement en fonction du contexte de la recherche (les connaissances que l'utilisateur a du domaine et des documents, sa façon d'exprimer son besoin, ses exigences, etc.) Mais malgré leurs différences, le but de ces modèles reste commun : satisfaire au mieux les besoins de l'utilisateur.

1. Problématique

« Le besoin d'information correspond à un manque de connaissance d'un individu dans une situation » [Tricot,04]. Afin de combler ce manque de connaissance, l'individu recherche l'information manquante dans une mémoire externe composée en grande partie des documents. En général, l'individu est considéré comme ne connaissant pas le contenu de ces documents susceptibles de satisfaire son besoin, pourtant il peut arriver qu'il ait déjà consulté ces documents auparavant et qu'il ait donc une idée plus ou moins vague de leur contenu.

Imaginons, par exemple, un individu qui visite un musée et qui a souvenir d'un tableau qu'il aimerait retrouver. Cet individu pose son besoin ainsi:

«Je recherche un tableau qui représente une barque...enfin, je crois que c'est une barque, et sur cette barque il y a une femme debout et une autre assise, ça c'est sûr... et peut être qu'il y a aussi un homme. »

Dans cet exemple, l'individu connaît le document, ou plus exactement le tableau, qui l'intéresse. Dans la formulation de son besoin, il décrit le tableau qu'il aimerait retrouver tel qu'il le perçoit. Il imagine ce tableau tel qu'il pense qu'il est et utilise cette description pour formuler son besoin. Ce type de recherche d'information se situe dans un contexte bien particulier : d'abord la formulation du besoin est une description du document "idéal" que l'utilisateur aimerait retrouver et ensuite l'utilisateur a une mémoire des documents, mémoire qu'il met à profit pour les retrouver.

Nous remarquons que la description que l'utilisateur fait du tableau qu'il recherche n'est pas très précise (utilisation des expressions *je crois* et *peut être*). Cela est assez compréhensible, vu que l'utilisateur reconstruit mentalement un document déjà vu, reconstruction qui n'est pas forcément précise, mais qui comporte souvent des doutes. Ceci pose certains problèmes pour la modélisation du système de recherche, où il n'est pas aisé de raisonner avec de vagues assertions ou des affirmations qui comportent des incertitudes. Pourtant, le système de recherche d'information devrait permettre à l'utilisateur d'exprimer son besoin en donnant une description

du document tel qu'il pense qu'il est, avec tout ce que cela comprend comme doutes et incertitudes.

Ces doutes au niveau des affirmations de l'utilisateur, ont fait l'objet de certains travaux, notamment dans le domaine des systèmes à base de connaissances [Akdag,94] [Khoukhi,96], ou des bases de données [Bosc,94] [Rocacher,03], mais en recherche d'information, peu de travaux prennent en compte les doutes des utilisateurs en situation de recherche, l'incertitude est le plus souvent considérée du point de vue document : incertitude du contenu des documents.

Ce contexte particulier de formulation de la requête (exprimer le besoin en information en décrivant le document 'idéal' et en ayant en mémoire ce document) est particulièrement observable dans les milieux professionnels. En effet, les professionnels ont des connaissances du domaine ainsi que des documents (comptes rendus, documentation technique, textes de loi, etc.) qu'ils consultent régulièrement. Ainsi lorsqu'ils recherchent des documents, c'est généralement pour compléter une information qu'ils ont déjà mais qui est insuffisante et/ou incertaine. Ainsi, un médecin voulant retrouver certains comptes rendus (qu'il a probablement déjà consulté auparavant) devrait, s'il le désire, pouvoir exprimer son besoin en ces termes : « Ils concernent une tumeur située au niveau du poumon. On y mentionne aussi le stade de la maladie... je crois que c'est le stade 3B. Et on y parle peut être du traitement qui devrait être, si je me souviens bien, les Interférons». Prendre en compte, dans le processus de recherche, les doutes et les incertitudes de l'utilisateur, quand à la représentation qu'il se fait des documents susceptibles de l'intéresser, devrait permettre de retrouver des documents en meilleure adéquation avec son réel besoin, ce qui est un critère important pour ce genre d'utilisateurs qui s'attendent à trouver une réponse précise et de qualité à leur requête leur permettant ainsi de réaliser leur tâche professionnelle (établir un diagnostic, se documenter, réparer une machine, etc.)

2. Objectifs

L'objectif de notre travail est de définir un modèle de recherche d'information qui soit en adéquation avec le contexte particulier dans lequel nous nous situons:

- la formulation de la requête est une description du document 'idéal' recherché par l'utilisateur. L'utilisateur décrit donc le document qu'il souhaite retrouver en précisant ce qui est important (critère obligatoire) ou moins important (critère optionnel) que ce document contienne.
- les utilisateurs connaissent les documents et ils en ont donc un souvenir (une mémoire) plus ou moins fiable. Lors de la formulation du besoin, la description du document idéal peut alors, dans certaines situations, être la description du souvenir que l'utilisateur a du document recherché, avec tout ce que cela comporte comme doutes (critères de certitude ou d'incertitude).
- les utilisateurs sont des professionnels avec une forte connaissance du domaine qui doit être représentée au sein du système. En considérant leurs exigences en terme de précision du système, le langage sur lequel sera fondé le modèle doit être expressif (complexe).

3. Approche

Dans ce travail, nous nous intéressons à une formulation particulière du besoin de l'utilisateur qui est une description, pouvant être imprécise ou incertaine, du document 'idéal' qu'il recherche. Nous nous intéressons plus particulièrement aux applications dédiées à des professionnels qui nécessitent une représentation spécifique des documents.

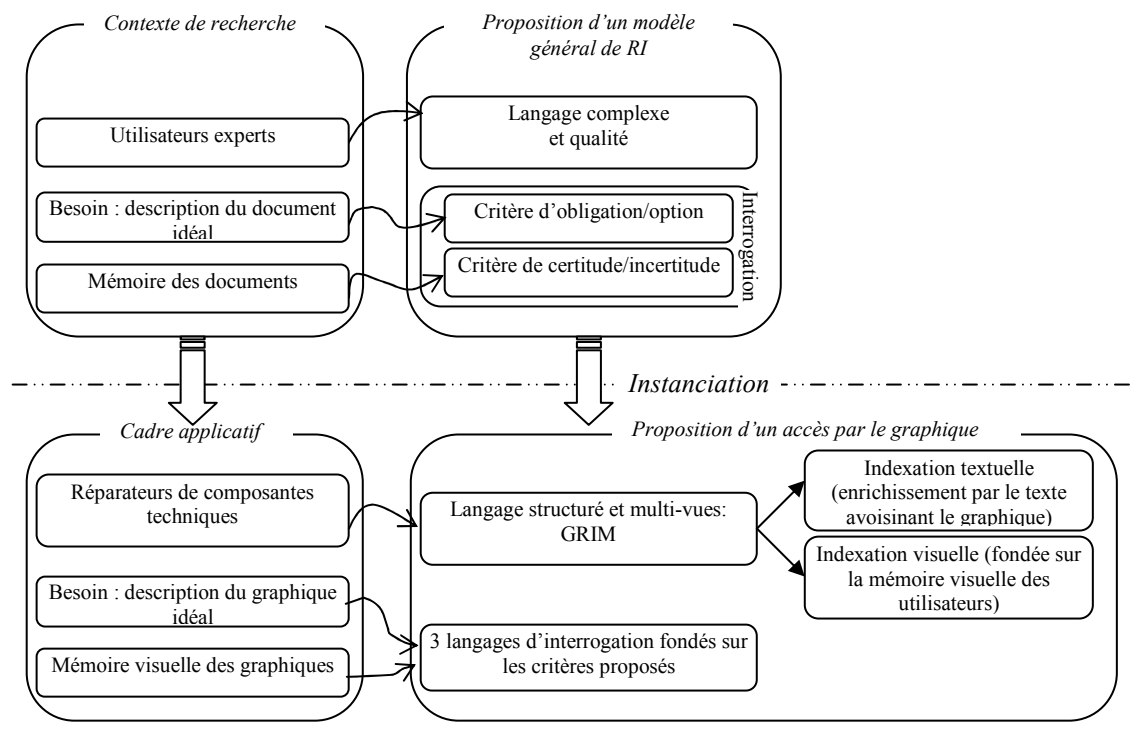


Figure 1 Schéma récapitulatif de l'approche

Notre travail s'articule donc autour de deux points (voir Figure 1) :

3.1. Proposition d'un modèle général de recherche d'information

Le modèle de recherche que nous proposons doit:

- Fonder son indexation sur un langage complexe (par opposition à sac de mots clés), dans le but de satisfaire aux exigences des utilisateurs experts du domaine. La qualité d'une telle indexation pour de tels utilisateurs ne doit pas être négligée.
- intégrer, dans la formulation de la requête, les jugements d'importance (les critères d'obligation et d'option) et les doutes (les critères de certitude et d'incertitude) de l'utilisateur quant au contenu des documents recherchés. À partir des caractéristiques des documents renvoyés (en relation avec les critères posés), nous proposons une approche pour la reformulation de la requête.

L'indexation : complexe et de qualité

Les systèmes de recherches d'information dédiées à des professionnels sont des systèmes orientés précision. Ceci implique une représentation rigoureuse de l'indexation et donc une *indexation fondée sur un langage complexe* (par opposition à langage à base de mots clés) des documents afin de restituer fidèlement leur contenu.

Les indexations à base de langages complexes étant jugées peu fiables qualitativement, un intérêt est donné à la validation de la qualité de l'indexation en proposant une adaptation de *mesures de qualité* existantes pour des indexations à base de langages simples (liste de mots clés) afin de permettre la vérification d'indexations fondées sur des langages complexes.

Formulation du besoin : description du document idéal et doutes de l'utilisateur

Les utilisateurs recherchant une information ne savent pas exactement ce que contiennent les documents susceptibles de les intéresser. Afin de les retrouver, ils tentent de décrire ce qu'ils pensent être le document idéal répondant à leur besoin, en précisant ce qui est important ou moins important que ce document contienne. Nous prenons en compte cette importance donnée par l'utilisateur aux éléments de la requête, en leur associant des *critères d'obligation et d'option*. Ainsi, les documents susceptibles d'intéresser l'utilisateur devront contenir tous les éléments marqués « obligatoires » et éventuellement les éléments marqués « optionnels ».

D'un autre côté, lorsqu'ils ont déjà consulté les documents qu'ils recherchent, comme c'est souvent le cas dans les milieux professionnels, les utilisateurs utilisent leurs souvenirs (qui ne sont pas toujours fiables ni clairs) pour construire la requête. Il utilisent alors des tournures telles que « je crois que », « je suis sûr que », « peut être que », etc. pour différencier ce dont ils sont certains, dans les documents susceptibles de les intéresser, de ce dont ils sont incertain. Nous prenons en compte ces doutes de l'utilisateur, en enrichissant la formulation de la requête par des *critères de certitude et d'incertitude*. Ainsi, les documents susceptibles d'intéresser l'utilisateur devront contenir les éléments marqués « certains » tels qu'ils apparaissent dans la requête, et les éléments marqués « incertains » tels qu'ils apparaissent dans la requête ou sous une forme qui leur est « proche » ou « similaire ».

Cette première formulation vague permet à l'utilisateur d'avoir un premier échantillon de documents classés en fonction des critères qu'ils vérifient en commun (obligation/option ou certitude/incertitude) : les documents contenant le plus grand nombre d'éléments optionnels (ou d'éléments incertains sous la forme mentionnée dans la requête) seront rangés dans la classe de plus haut degré de pertinence (présentée en premier à l'utilisateur). Cet échantillon classé lui permet alors de préciser sa requête, soit en la reformulant lui-même car il a une meilleure idée de ce qu'il veut retrouver, soit en laissant au système le choix de la reformulation en fonction des caractéristiques des documents qu'il juge pertinents.

3.2. Application à la recherche d'information technique par des professionnels

Une fois le modèle général de recherche d'information proposé, nous travaillons dans le cadre applicatif de la recherche d'information technique par des professionnels (réparateurs de dispositifs). La prise en compte du contexte de cette application, nous amène à proposer une recherche par le graphique. Le modèle de recherche proposé dans cette partie est une *instance* du modèle général proposé précédemment.

Le graphique comme point d'accès au document

La lecture des documents techniques est rarement linéaire : l'utilisateur accède directement au passage qui l'intéresse pour l'accomplissement de sa tâche. Ce passage est souvent formé par un couple graphique-texte, et ce dans le but de faciliter sa compréhension. En outre, la consultation fréquente des documents techniques par les utilisateurs professionnels a forcément un impact sur leur mémoire à long terme et notamment sur leur *mémoire visuelle* (surtout au niveau du graphique).

En partant de là, nous pensons qu'il est intéressant de permettre aux utilisateurs d'accéder à la documentation technique via le média *graphique*, ou plus exactement via ce dont ils s'en souviennent. Une fois le graphique retrouvé, les utilisateurs pourront naviguer dans le reste de la documentation technique, qui est structurée.

L'indexation mixte des graphiques

Les illustrations ont toujours été présentes et ont sans cesse occupé une place importante dans les modes de communication. Dans les documents techniques, la coexistence du texte et des illustrations est omniprésente. En effet, l'information véhiculée dans ces documents est présente dans les deux modes d'expressions que sont le texte et le graphique et notamment entre le graphique et son commentaire. Une complémentarité informationnelle entre ces deux médias existe mutuellement dans ces documents. On peut cependant noter que généralement la plupart des systèmes délaissent les graphiques au profit du texte. En effet, de nombreux travaux existent sur l'indexation d'images et de graphiques, mais peu se sont concentrés sur le complément informatif graphique-texte.

Notre objectif est d'exploiter cette complémentarité informationnelle dans les documents techniques, afin d'aboutir à un modèle de représentation des graphiques tenant compte de leurs *propriétés visuelles* ainsi que du *contexte textuel* dans lequel ils apparaissent. Nous souhaitons ainsi parvenir à un processus d'indexation semi-automatique des graphiques contenus dans les documents techniques qui permet (i) la représentation structurée du contenu visuel des graphiques et (ii) l'extraction automatique des termes appropriés à partir du texte les décrivant. Le langage d'interrogation fondé sur la mémoire de l'utilisateur professionnel (notamment sa mémoire visuelle) permettra la description du graphique idéal qu'il souhaiterait retrouver en prenant en compte ses doutes selon le modèle de RI proposé précédemment.

4. Plan du manuscrit

La suite du manuscrit se présente en quatre parties. La première présente un état de l'art, la deuxième la proposition d'un modèle de recherche d'information dans un cadre général, la troisième décrit une instanciation de ce modèle et la quatrième partie une mise en œuvre ainsi que des validations relatives à cette instance. Nous terminons, ce manuscrit par une conclusion et des perspectives.

Première partie : Etat de l'art

L'état de l'art s'articule autour de deux chapitres.

Nous nous intéressons, dans un premier chapitre, à la modélisation des systèmes de recherche d'information et particulièrement :

- au langage d'indexation qui permet de restituer le contenu des documents avec plus ou moins de fidélité selon le type de langage employé: nous distinguons ici les deux grands types de langages existants, soient les langages simples et les langages complexes et démontrons l'apport des seconds par rapport aux premiers.
- au langage d'interrogation qui permet de décrire avec plus ou moins de rigueur le besoin en information des utilisateurs : nous nous intéressons, en particulier, à des langages qui prennent en compte l'importance donnée par l'utilisateur aux éléments de la requête ainsi que les doutes de l'utilisateur quant aux documents susceptibles de l'intéresser.
- à l'évaluation de la qualité des systèmes de recherche: nous nous intéressons en particulier, à la qualité de l'indexation, qui étant un processus souvent coûteux et irréversible, doit être validée avant d'être intégrée dans le système global.

Dans un deuxième chapitre, nous nous intéressons à la recherche d'information dans les documents techniques par des utilisateurs professionnels. Dans une telle application, les utilisateurs sont des experts du domaine et ils ont une mémoire des documents qu'ils recherchent, ce qui coïncide avec le contexte de recherche dans lequel nous situons notre travail.

Nous commençons donc par caractériser le contexte de la recherche dans documents techniques (documents, utilisateurs et usages) et nous nous intéressons par la suite au média graphique, omniprésent dans ce type de documents, et à la possibilité de le considérer comme un point d'accès au document. Un tour d'horizon des modèles d'indexation d'un tel média est alors proposée en fin de ce chapitre.

Deuxième partie : Proposition d'un modèle de RI fondé sur des langages complexes et sur des critères d'obligation/option et de certitude/incertitude

Cette proposition comprend deux chapitres. Après un chapitre préalable (chapitre III) décrivant les grandes lignes d'un modèle général de RI, et proposant des notations pour ce modèle, nous proposons, dans le quatrième chapitre, un modèle de recherche d'information fondé sur un langage complexe et qui a la particularité de (i) proposer une représentation fidèle du document et du besoin, et ce en permettant l'utilisation multiple d'une même entité dans l'index du document ou dans la requête (il s'agit, par exemple, d'indexer un document représentant deux femmes, par deux termes correspondant toutes les deux à « femme » et non pas par un unique terme « femme » ayant éventuellement un poids plus important) et (ii) proposer une formulation riche de la requête permettant ainsi à l'utilisateur d'exprimer son besoin en décrivant ce qu'il pense être le document idéal répondant à son besoin (en fonction de ce qu'il veut exactement retrouver dans les documents). Cet enrichissement est effectué en utilisant des critères d'obligation/option et de certitude/incertitude associés aux éléments de la requête. La correspondance entre le document et la requête doit alors prendre en compte les contraintes engendrées par ces deux points.

Les propriétés des documents retournés par le système (par rapport aux critères associés aux éléments de la requête), permettent une nouvelle formulation de cette requête en fonction des jugements de pertinence de l'utilisateur. Nous proposons, à la fin de ce chapitre, une approche pour cette reformulation.

Troisième partie : Instanciation du modèle

Dans la troisième partie, nous abordons notre approche dans le cadre concret de la recherche de documents techniques, via le média graphique. Nous proposons donc une indexation riche de ce média (d'où le besoin d'un langage complexe), prenant en compte ses données intrinsèques ainsi que son contexte (afin d'enrichir son indexation), et une recherche fondée sur les critères d'obligation et de certitude précédemment mentionnés. Cette application est d'autant plus intéressante, que la recherche est fondée sur la mémoire visuelle de l'utilisateur professionnel, qui a une connaissance vague des graphiques qu'il veut retrouver.

Cette partie se décline en deux chapitres : un chapitre V, qui a pour objectif d'identifier les informations à retenir, nécessaires à la description exhaustive des graphiques, et de proposer une façon de les représenter, et un chapitre VI décrivant le modèle de recherche de graphique qui prend en compte les informations dégagées dans le chapitre V, et qui est une instance du modèle général de RI proposé dans la deuxième partie.

Quatrième partie : Mises en œuvre et validations

Toujours dans le cadre de la recherche de documents techniques via le graphique, la dernière partie, présentée en deux chapitres, est consacrée à la mise en œuvre du processus d'indexation et à une validation de la recherche:

- la mise en œuvre du processus d'indexation des graphiques (chapitre VII). Ce processus englobe une première partie automatique correspondant à l'extraction des termes appropriés à partir du texte commentant le graphique et une seconde partie manuelle correspondant à l'indexation visuelle du graphique (sa « géométrie », en fonction de ce qui devrait être retenu dans la mémoire visuelle de l'utilisateur). La validation de l'indexation automatique est faite à l'aide de mesures qualitatives d'indexation, que nous proposons en adaptant les mesures de qualité existant pour des langages simples, et un protocole expérimental est proposé pour la validation de l'indexation manuelle du graphique (son indexation visuelle).
- la validation de l'ajout des critères d'obligation et de certitude dans la formulation de la requête (chapitre VIII) en mettant en œuvre une correspondance entre les documents indexés et des requêtes faisant intervenir uniquement une partie de l'index comprenant les termes d'indexation simples (pas de relations). Cela nous permet de montrer que (i) une requête sans prise en compte des critères proposés ne donne pas les mêmes résultats et (ii) il n'est pas possible de rendre compte, parfaitement, de ces critères avec les modèles classiques (booléen, vectoriel, de langue et probabiliste).

Partie 1 : Etat de l'art

Chapitre I La recherche d'information : modélisation et qualité

Chapitre II Documents techniques et usages : un accès par le graphique

Partie 2: Proposition d'un modèle de RI

Chapitre III Préliminaires : Définitions et notations

Chapitre IV Modèle de RI pour langages complexes intégrant obligation et certitude

Partie 3 : Instanciation du modèle

Chapitre V Vers une représentation exhaustive des graphiques

Chapitre VI

- Indexation : le graphique un objet multi-vues
- Formulation de la requête : trois langages
- Correspondance

Partie 4 : Mise en œuvre et validations

Chapitre VII Indexation textuelle et visuelle du graphique

Chapitre IIX Formulation de la requête et correspondance :
validation de l'ajout des critères



Partie 1. État de l'art

Il existe un grand nombre de modèles de recherche d'information. Ces modèles diffèrent principalement sur la façon dont les informations disponibles sont représentées ainsi que sur la façon d'interroger le corpus de documents. Le choix de la représentation des informations (document et requête) et de la fonction de correspondance constitue le principal problème en recherche d'information car de ce choix dépendent la qualité des résultats et la satisfaction des utilisateurs. Le premier chapitre de l'état de l'art s'intéresse à différents choix dans la modélisation d'un système de recherche d'information que ce soit au niveau du langage d'indexation ou au niveau du langage d'interrogation et à l'impact de ces choix sur le comportement du système. Il s'intéresse aussi, à l'évaluation de la qualité des systèmes d'information et en particulier à l'évaluation de la qualité de leur indexation qui souvent coûteuse et irréversible, devrait être validée avant d'être intégrée dans le système global.

Rappelons que, dans le contexte de notre travail, nous souhaitons que le modèle de recherche permette une représentation fidèle du contenu des documents (car les utilisateurs sont des experts du domaine) ainsi qu'une formulation du besoin permettant la description d'un document 'idéal' avec ce qu'il contient d'important et de moins important et la prise en compte des doutes des utilisateurs résultant du souvenir qu'ils ont des documents recherchés et qu'ils ont en mémoire, car déjà consultés.

Afin de concrétiser un tel contexte, nous avons choisi la recherche d'information dans les documents techniques (de type manuels d'utilisation/réparation de composants techniques) par des utilisateurs professionnels. Le deuxième chapitre s'intéresse à une telle application : une première partie concerne la description des caractéristiques de ce contexte particulier telles que le type des documents recherchés et les besoins et habitudes de travail des utilisateurs, et une seconde partie s'intéresse aux graphiques qui sont omniprésents dans la documentation technique, et à la possibilité que ce média présente un point d'accès au document puisqu'il s'agit d'une information visuelle aisément mémorisable par l'utilisateur. Un tour d'horizon des modèles d'indexation d'un tel média est alors proposée.

Chapitre I. La recherche d'information : modélisation et qualité

1. Système de recherche d'information : architecture générale

Les systèmes de recherche d'information (SRI) ont pour rôle de permettre l'accès aux documents par leur contenu. Ainsi, l'utilisateur exprime son besoin d'information en indiquant le contenu qu'il souhaite observer dans les documents retrouvés, et le système lui renvoie les documents qui correspondent à ce contenu.

L'ensemble des documents sur lequel porte la recherche est appelé le corpus, qui est en général relatif à un thème particulier ou à un ensemble de thèmes. La notion de document est prise au sens large en recherche d'information : elle comprend les documents textuels, les images, les graphiques, les sons ou toute combinaison de ces médias.

Comme le montre la Figure 2, classiquement, un système de recherche d'information se décompose en deux phases:

a. Une phase d'indexation

Elle a pour objectif de définir pour chaque document ses unités d'indexation représentant son contenu sémantique. La notion d'unité d'indexation est à prendre ici au sens large : nous entendons par unité d'indexation toute forme produite par l'indexation d'un document, quelque soit sa complexité.

L'indexation produit ainsi la représentation de chaque document, conformément à un langage d'indexation, définissant les unités d'indexation utilisables. Elle comprend deux points bien distincts.

- D'une part, il s'agit de définir le *langage d'indexation*, permettant la représentation des concepts des documents. Le choix de ce langage conditionne la qualité (en terme de précision) des réponses du système qui l'implante. En effet, plus le langage est expressif et représentatif du contenu du document, plus le système de recherche a des chances d'être précis.
- D'autre part, il s'agit de définir un *processus d'indexation* permettant l'extraction, à partir des documents, d'une représentation de leur contenu, conformément au langage d'indexation. Lorsque le langage est complexe, le processus est souvent source d'erreurs d'indexation altérant ainsi la qualité du système.

b. Une phase d'interrogation

Elle permet à l'utilisateur de formuler une requête conformément à un langage prédéfini appelé *langage d'interrogation* et ainsi d'interroger le corpus. Par analogie au langage d'indexation, le langage d'interrogation définit les unités d'interrogation permettant de représenter le besoin en

information de l'utilisateur. Une *fonction de correspondance* compare la requête aux documents, représentés respectivement selon les langages d'interrogation et d'indexation : les documents répondant à la requête sont ainsi donnés en réponse, généralement dans une liste ordonnée (les documents les plus pertinents pour le système sont classés en haut de la liste).

Le langage d'interrogation ne permet pas toujours à l'utilisateur d'exprimer convenablement son besoin (comme il le perçoit en réalité). Par conséquent, sa satisfaction concernant les documents retournés par le système, n'est pas toujours acquise. Certains systèmes permettent aux utilisateurs de marquer parmi les documents résultats ceux qu'ils jugent pertinents ou non pertinents. Ces jugements sont alors pris en compte pour définir une nouvelle requête : il s'agit là du processus de *reformulation*. Ce processus n'est pas toujours automatique : une stratégie classique d'utilisation des systèmes de recherche d'information consiste à reformuler manuellement la requête en tenant compte des documents pertinents et non pertinents obtenus. Lorsqu'elle est automatique la reformulation est dite bouclage de pertinence (ou relevance feedback). Cette technique a d'abord été introduite par [Rocchio,71] dans le système SMART [Salton,71]. Elle consiste à modifier la requête en fonction des termes d'indexation présents dans les documents jugés pertinents.

Une *modélisation de la connaissance* peut être incluse dans le système de recherche. Elle sert de référence aux processus d'indexation et d'interrogation. Cette représentation des connaissances peut inclure, par exemple, un thésaurus composé des mots apparaissant dans l'ensemble des documents, reliés entre eux par des liens de généralité/spécificité ou de synonymie. L'utilisation d'un thésaurus permet d'augmenter le nombre de réponses du système, en y incluant des documents contenant des mots reliés aux mots de la requête.

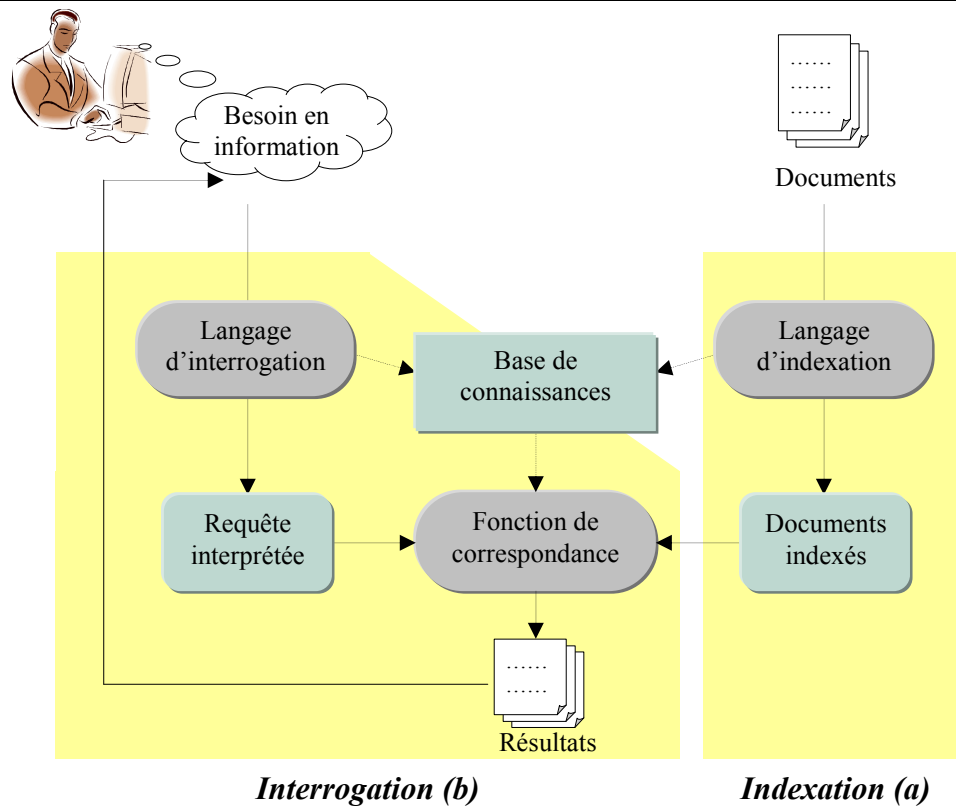


Figure 2 Architecture d'un SRI

Pour résumer, un système de recherche d'information remplit deux rôles : d'une part, il crée une représentation pour les documents conformément à un langage d'indexation, et une interprétation du besoin d'information de l'utilisateur conformément à un langage d'interrogation, et d'autre part, il compare, à travers une fonction de correspondance, la représentation du document et la représentation de la requête afin de déterminer leur degré de similarité. Du choix des langages d'indexation et d'interrogation et de la fonction de correspondance dépend la qualité du SRI et donc la satisfaction de l'utilisateur.

2. L'indexation

L'indexation passe par la définition d'un langage d'indexation.

Le langage d'indexation exprime le contenu sémantique des documents. Il doit offrir un compromis entre la compacité de la représentation, pour qu'elle puisse être traitée efficacement par un système informatique, et l'expressivité afin d'exprimer aussi fidèlement que possible le contenu des documents. Le choix de ce langage est fondamental, car il détermine la qualité de la représentation interne des documents, et donc la qualité de la recherche et des résultats.

Nous distinguons deux types de langages d'indexation. Le premier est le plus basique : le langage simple, et le second plus expressif est le langage complexe : il intègre d'autres éléments afin de décrire avec plus de fidélité le contenu des documents.

2.1. Langages simples

La description la plus basique des documents est fondée sur l'utilisation d'un ensemble de mots clés (l'unité d'indexation, ici, est aussi appelée mot clé). Il s'agit d'un ensemble de syntagmes¹ qui représentent le contenu du document. Ainsi, un document représenté par le mot clé « sport » est un document *à propos du sport*, ce document est alors pertinent pour une requête qui porte sur le sport.

Les approches classiques de recherche d'information textuelles sont fondées sur des langages simples : le contenu d'un document textuel est ainsi exprimé sous la forme d'un ensemble de mots clés jugés représentatifs de ce contenu. Ces mots clés sont généralement un sous-ensemble des syntagmes apparaissant dans les documents. Les mots clés d'un document peuvent tous être considérés comme ayant une même importance : ils sont considérés comme étant aussi représentatif les uns que les autres de la sémantique du document. Il s'agit, dans ce cas, d'une pondération booléenne [Baeza-Yates,99] où seule la présence ou l'absence d'un mot clé dans le document est importante. Cependant, en réalité, ces mots clés n'ont pas tous la même importance dans les documents : pour refléter leur importance et affiner ainsi l'indexation, ils peuvent être pondérés : plus la pondération est grande, plus il sont représentatif du contenu des documents.

Le grand avantage des langages d'indexation simples réside dans leur simplicité. En effet, les approches fondées sur ces langages peuvent facilement être automatisées et elles sont applicables à tout type de document, à tout domaine et à des corpus de grandes tailles. Cependant, une telle représentation est trop pauvre, et doit donc être exclue pour les systèmes où la précision est importante (les systèmes orientés précision). Cela est le cas des systèmes dédiés à des utilisateurs experts d'un domaine dont l'objectif est de ne retrouver que des documents pertinents, mais pas forcément tous. Disposer d'un langage plus expressif et plus représentatif du contenu devient donc une nécessité dans de tels systèmes.

Exemple

Dans le domaine médical où la précision est fondamentale pour les médecins, prenons l'exemple d'un compte rendu médical indexé par la liste des mots clés : *tumeur, poumon, gauche*. Cette représentation porte à confusion : est-ce que le compte rendu concerne un patient ayant une tumeur située au *niveau gauche des poumons*, ou est-ce que cette tumeur se situe au *niveau du poumon gauche* ?

Nous remarquons, ici, que l'intégration des relations (dans l'exemple, la relation « au niveau de ») dans le langage d'indexation a un impact majeur sur la description des documents et donc, théoriquement, sur la précision des systèmes.

Selon [Blair,90][Lalmas,96][Ounis,97], la pauvreté de la représentation des documents est la raison principale de l'échec relatif des approches simples en recherche d'information. Pour pallier cette représentation trop pauvre, une solution possible consiste à utiliser des unités d'indexation plus riches grâce à la prise en compte de relations dans le langage d'indexation. Nous parlons dans ce cas de langages complexes.

¹ Groupe de mots en séquence formant une unité à l'intérieur de la phrase.

2.2. Langages complexes

La nécessité des langages complexes est accrue dans certains systèmes dédiés à des utilisateurs spécialistes d'un domaine : les langages complexes sont le seul moyen d'atteindre le niveau de précision recherché.

Nous citons, dans cette partie, différents travaux qui ont fondé leur indexation sur des langages complexes: Certains, en s'aidant de bases de connaissances structurées du domaine, extraient, à partir des documents, les syntagmes représentatifs de leurs contenu et les relations qui les unissent. Ces informations sont ensuite décrites à l'aide d'un formalisme de représentation de connaissances tels que les graphes conceptuels [Sowa,84], les arborescences sémantiques, etc.

RIME (Recherche d'Information MEdicale) [Berrut,88] [Berrut,89] [Berrut,90]

RIME est un SRI destiné à des spécialistes compétents du domaine de la radiologie. Le corpus traité est constitué d'images et des comptes rendus médicaux qui leurs sont associés.

La définition du langage d'indexation dans RIME est inspirée des dépendances conceptuelles de Schank [Schank,80] [Schank,81]. Le contenu d'un document est décrit sous forme d'une arborescence (voir Figure 3) dans laquelle :

- les nœuds feuilles sont des *concepts primitifs* du domaine, c'est-à-dire, des syntagmes médicaux ou techniques tels que « opacité » ou « poumon ». Ils représentent le niveau le plus bas de représentation des documents et des requêtes,
- les nœuds non terminaux sont des *opérateurs sémantiques* explicitant la relation entre deux concepts de plus bas niveaux, représentés par les deux sous arbres (par exemple, la relation « a-pour-valeur »). Ces noeuds dénotent des relations sémantiques et permettent la construction de nœuds complexes.

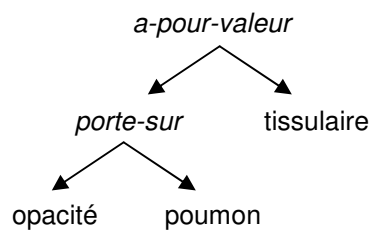


Figure 3 Représentation de l'information : « opacité tissulaire au niveau du poumon »

MENELAS [Zweigenbaum,94]

Ce projet, dédié lui aussi au domaine médical, propose une indexation fondée sur un langage complexe, des comptes rendus d'hospitalisation.

Le processus automatique d'indexation analyse le résumé médical (par le biais d'une grammaire contextuelle et d'une analyse sémantique/pragmatique) et modélise l'information dans le formalisme des graphes conceptuels. Le langage est fondé sur des ensembles structurés de concepts (simples ou complexes) et de relations sémantiques. Un concept complexe est

représenté par un petit réseau de concepts simples et de relations. Par exemple, le concept « appendicite aiguë » est décomposé selon le schéma suivant :

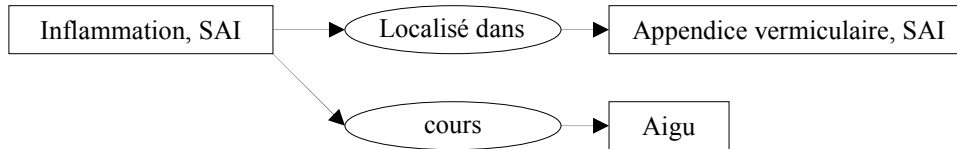


Figure 4 Représentation du concept complexe « appendicite aiguë »

EMIR² (Extended Model for Information Retrieval) [Mechkour,95]

EMIR² permet la représentation du contenu des images selon un langage complexe représenté dans le formalisme des *graphes conceptuels*. Son objectif est de modéliser le contenu des images en intégrant plusieurs niveaux d'interprétation. L'image est ainsi considérée comme un objet complexe pouvant être interprété comme :

- un ensemble d'objets, appelés *objets images* (notés oi), jugés pertinents par l'indexeur et reliés par des *relations de composition* (vue structurelle),
- un ensemble d'*objets géométriques* (notés o_sp) associés à leurs contours, et reliés entre eux par des *relations spatiales* (vue spatiale),
- un ensemble de descriptions symboliques (*objets* notés o_sy et *relations symboliques*) correspondant à des interprétations sémantiques de son contenu (vue symbolique),
- un ensemble de *descriptions visuelles* (couleur, texture, brillance) des objets (notés o_pe) qu'elle contient (vue perceptive).

Les vues spatiale, symbolique et visuelle sont des facettes du même objet image décrit par la vue structurelle. La description d'une image est une abstraction qui regroupe toutes ces vues partielles (voir la Figure 5 sur laquelle se retrouvent ces 4 vues)

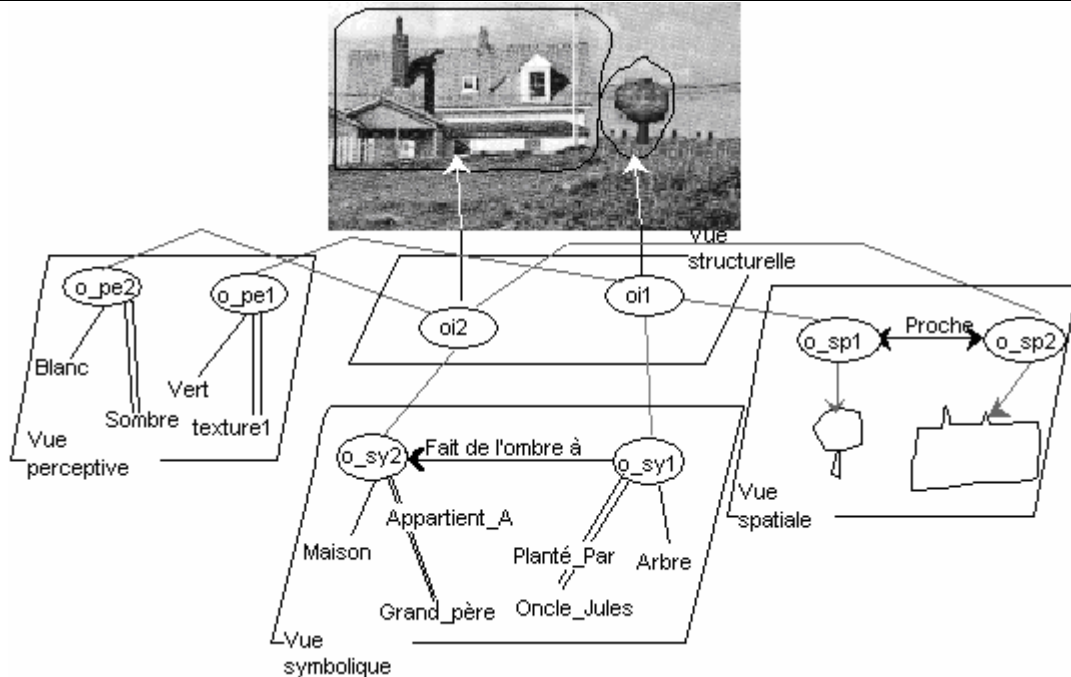


Figure 5 Un exemple d'image indexée selon EMIR²

OntoSeek [Guarino,99]

Il s'agit d'un système de recherche documentaire développé pour des documents de type catalogue (de structures proches). Les documents sont décrits par des concepts inter reliés et représentés sous forme de graphes conceptuels.

Un exemple de document et sa représentation sont donnés dans la Figure 6.

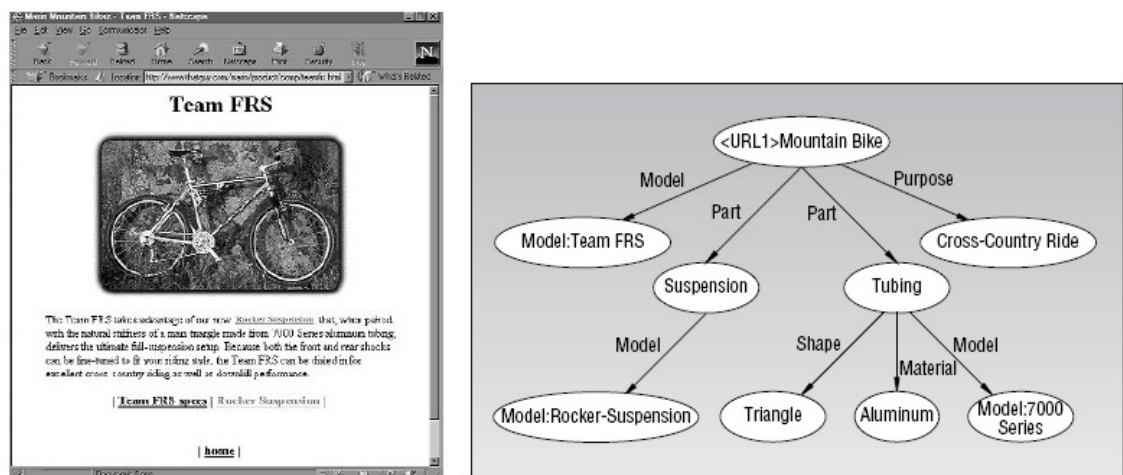


Figure 6 Exemple d'un document et de sa représentation sous forme d'un graphe conceptuel

Dans les travaux présentés, la représentation du contenu des documents repose sur des langages complexes dits conceptuels [Croft,86] : il s'agit d'une représentation sémantique pouvant se limiter aux concepts, mais comprenant généralement des liens ou relations entre ces concepts. La représentation de ces concepts dans la littérature prend diverses formes dont les graphes conceptuels, les arborescences, les formules logiques, etc.

2.3. Bilan

Les langages complexes peuvent représenter une solution à certains problèmes (la polysémie par exemple) en permettant une indexation, non plus à base de syntagmes (comme c'est le cas pour les langages simples) mais fondée sur la notion de concepts et sur les relations entre ces concepts. Ils permettent ainsi une description plus fine des documents, caractéristique nécessaire dans certains domaines, notamment ceux dédiés à des experts d'un domaine, tel que le domaine médical (RIME et MENELAS). Nous pouvons également mentionner d'autres travaux [Mihalcea,00] [Schutze,95], à mi chemin entre les langages simples et les langages complexes, qui dans le but d'affiner la représentation des documents, l'enrichissent en rattachant aux syntagmes extraits leurs définitions, ou d'autres syntagmes qui leur sont sémantiquement proches (désambiguïsation sémantique).

Même si les langages complexes multiplient les possibilités d'expression par rapport aux langages simples, ils sont souvent jugés comme difficilement manipulables : les processus d'indexation fondés sur ce type de langage, nécessitent, dans le cas de processus manuels, des indexeurs experts maîtrisant le langage d'indexation, et, dans le cas de processus automatiques, ils nécessitent souvent des ressources considérables (les bases de connaissances du domaine considéré) et des algorithmes plus ou moins complexes, autant au niveau du processus d'indexation qu'au niveau de la fonction de correspondance.

En plus, les langages complexes sont considérés comme peu fiables car sources de nombreuses erreurs d'indexation. Pourtant, dans les systèmes de recherche orientés précision, il est nécessaire d'avoir recours à un langage qui restitue fidèlement le contenu des documents, ce qui devrait, théoriquement augmenter la précision. Une validation préalable de la qualité de l'indexation paraît alors nécessaire dans le cas de l'utilisation de langages complexes, afin de vérifier qu'il n'y a pas (ou qu'il y a peu) d'erreurs d'indexation. Nous nous y intéressons dans le §4 de ce chapitre.

3. L'interrogation

L'interrogation passe par la définition d'un langage d'indexation et d'une fonction de correspondance :

- Le langage d'interrogation exprime le contenu sémantique du besoin d'information de l'utilisateur et détermine la précision de la définition de ce besoin. Il peut être identique au langage d'indexation : c'est le cas pour le modèle EMIR² [Mechkour,95] ainsi que le système OntoSeek [Guarino,99], ou bien, il peut se présenter sous une forme différente : c'est le cas, par exemple, dans Menelas [Zweigenbaum94], où les requêtes ne sont pas interprétées sous forme de graphes conceptuels (concepts inter reliés) mais sous forme de conjonctions de disjonctions de concepts (pas de prise en compte de relations entre les concepts dans le langage d'interrogation).

- La fonction de correspondance formalise le degré de similarité entre la requête et un document. Elle évalue cette similarité afin de déterminer la pertinence des documents pour cette requête.

Afin de formuler une requête, l'utilisateur doit spécifier les unités d'interrogations (descripteurs de son besoin) pouvant être comparés ou associés au contenu des documents de la collection. La représentation la plus basique associée à un besoin d'information est un ensemble de syntagmes, c'est ce que l'on voit le plus souvent dans les moteurs de recherche sur le web. Cependant, afin de représenter au mieux le besoin d'information de l'utilisateur, on associe souvent une importance aux unités d'interrogation (voir § 3.1), et certains travaux tentent de prendre en compte l'incertitude dans la formulation du besoin (voir § 3.2).

3.1. Importance des unités de la requête

Dans la formulation d'un besoin en information, les poids associés aux unités de la requête peuvent avoir différentes interprétations, selon ce que désire l'utilisateur. Nous citons, ici, trois interprétations qui sont les plus utilisées et les plus étudiées [Kraft,94]:

- l'importance [Waller,79] : dans ce cas, l'utilisateur désire retrouver les documents dont le contenu représente plus le concept associé à l'unité d'interrogation de poids le plus important, par rapport, à ceux de poids moins importants, autrement dit, les documents dominés par l'unité de la requête de poids le plus élevé;
- le seuil [Buell,81][Kraft, 83]: dans ce cas, l'utilisateur veut retrouver les documents qui sont suffisamment représentés par les concepts associés à chaque unité de la requête : autrement dit, les documents dont les poids des unités d'indexation dépassent ceux des unités correspondantes dans la requête;
- la perfection [Bordogna,91][Cater,87] : les poids sont considérés comme des descriptions des documents parfaits ou idéaux que l'utilisateur souhaite retrouver. Dans ce cas, l'utilisateur veut retrouver les documents dont les poids des termes d'indexation sont égaux ou à la limite proches des poids dans la requête, autrement dit, les documents dont le contenu satisfait ou est plus ou moins proche du document idéal décrit dans le besoin d'information. Cela implique que l'utilisateur doit être capable de spécifier avec précision les caractéristiques des documents considérés comme « parfaits » dans une forme en adéquation avec la représentation des documents.

Chaque pondération a donc une interprétation propre. Généralement, les modèles de RI proposés utilisent l'une ou l'autre de ces interprétations: rares sont les travaux qui en combinent deux en même temps [Herrera,01].

Dans d'autres travaux, au lieu d'associer des poids, souvent mal interprétés par l'utilisateur, aux unités de la requête, une distinction est faite entre les unités importantes et celles qui le sont moins. C'est le cas pour le moteur de recherche Altavista¹, par exemple. Dans ce système, le langage d'interrogation est un ensemble de syntagmes auxquels peut être rattaché ou non un préfixe "+". Ce préfixe, lorsqu'il est rattaché à un syntagme, signifie que ce syntagme est obligatoire, autrement dit, que l'utilisateur désire absolument le voir apparaître dans les documents renvoyés par le système. Son absence signifie que le syntagme en question est optionnel. La prise en compte d'un tel préfixe dans le langage d'interrogation a pour but de

¹ <http://www.altavista.com/>

fournir une syntaxe simple et intuitive de la requête et surtout faciliter l'expression de la requête par rapport à l'utilisation des opérateurs booléens, autrement dit, de résoudre la difficulté rencontrée par les utilisateurs pour exprimer des requêtes booléennes.

Exemple

La requête : « +barque homme » exprime que les documents pertinents sont ceux qui contiennent absolument une « barque » et éventuellement un « homme ».

Nous pouvons voir une autre utilisation de cette notion dans [Denos,97a] [Denos,97b] qui a proposé d'associer aux termes de la requête un critère d'obligation/option. La requête peut être vue, dans ce cas, comme un ensemble de couples formés chacun par un terme et le critère d'obligation qui lui est associé. Les documents retournés par le système devaient tous contenir les termes marqués comme obligatoires dans la requête. Ces documents étaient alors présentés à l'utilisateur dans des classes : chaque classe regroupant les documents contenant les mêmes termes optionnels. Cette approche permettait à l'utilisateur de mieux comprendre la relation entre sa requête et les documents retrouvés ce qui aidait à reformuler correctement la requête.

3.2. Incertitude dans l'expression de la requête

Toutes les informations contenues dans notre représentation mentale de la réalité sont « des idées que l'on juge être vraies ou probablement vraies » : Le cerveau humain peut raisonner avec de vagues assertions ou des affirmations qui comportent des doutes.

Ces doutes ont fait l'objet de plusieurs travaux qui se sont surtout souciés de l'incertitude concernant le contenu des documents (quantifier la façon dont un terme peut décrire le contenu d'un document), dans un contexte général, ainsi que celle concernant leur structure (évaluer la capacité que possède un élément de la structure du document à fournir une information pertinente à l'utilisateur), dans le cas de la recherche d'information structurée (les documents XML, par exemple) [Lalmas,96] [Lalmas,04]. Pourtant cette notion existe aussi au niveau des requêtes, et ce pour des raisons différentes. D'abord, lorsque la requête est exprimée en langage naturel, certaines parties peuvent être interprétées de différentes manières, des connaissances contextuelles peuvent alors aider à choisir l'interprétation la plus plausible. Ensuite, l'utilisateur peut aussi avoir des doutes concernant ce qu'il recherche, auquel cas, la requête doit exprimer ses doutes ou ses imprécisions.

Cette incertitude ou imprécision, au niveau des affirmations de l'utilisateur, a fait l'objet de certains travaux, notamment dans le domaine des systèmes à base de connaissances [Akdag,94] [Khoukhi,96], afin de permettre la modélisation et l'interprétation d'énoncés tels que « Il est *très vraisemblable* que la neige soit *assez blanche* ».

Nous pouvons aussi trouver une prise en compte de l'incertitude dans le domaine des bases de données [Bosc,94] [Rocacher,03], dont l'objectif est de permettre la formulation et la gestion, de requêtes graduelles incluant des préférences, telles que : « trouver les restaurants chinois proches des Halles avec menu aux environs de 20 € » où "chinois" peut se comprendre comme "de préférence chinois, éventuellement japonais ou vietnamien, à la rigueur indien" et la condition sur le lieu et le prix signifient une valeur idéale dont on peut quelque peu s'écarter avec une pénalisation sur la satisfaction obtenue. La prise en compte de l'incertitude est gérée, ici, en utilisant la théorie des ensembles flous, qui se trouve être adaptée à l'expression de préférences dans les requêtes.

En recherche d'information, la prise en compte de l'incertitude apparaît de manière implicite. Ainsi l'utilisateur peut exprimer ses préférences en affectant des poids plus ou moins fort aux unités d'interrogation, ou en utilisant la disjonction entre les unités sur lesquels portent ses doutes (comme par exemple, "chinois" OU "japonais" OU "vietnamien" OU "indien"). Pourtant, dans certaines applications, telles que celles où la recherche concerne des données déjà consultées par les utilisateurs et donc mémorisées, l'incertitude engendrée par le souvenir devrait être prise en compte de façon plus explicite.

4. Qualité en recherche d'information : validation de l'indexation en amont

Depuis longtemps, des spécialistes se sont employés à indexer des documents, dans les centres documentaires, les bibliothèques, etc. Ces spécialistes jouent un rôle pivot fondamental puisque, d'une part, ils permettent la représentation du contenu des documents, et d'autre part, c'est grâce à cette même représentation que les utilisateurs retrouvent les documents en formulant leurs besoins au travers de requêtes.

Avec la multiplication des sources d'informations numérisées, la nécessité d'automatiser tout ou partie du processus d'indexation est devenue évidente : « La mécanisation de l'indexation (...) a un intérêt d'abord pratique : des dizaines de milliers de spécialistes consacrent une part notable de leur temps à exprimer le contenu de documents scientifiques toujours plus nombreux, en vue de faciliter les recherches rétrospectives ultérieures ; le recrutement et la formation d'analystes compétents, pour cette tâche, sont de plus en plus difficiles, et il est naturel que l'on cherche à contourner l'obstacle, par la mécanisation de celle-ci. » [Gardin,70].

Dans tous les cas, que le processus d'indexation soit manuel, semi-automatique ou automatique, son principe paraît simple : identifier un document par un ensemble de termes représentant son contenu, porteurs de l'information le caractérisant et permettant son interrogation. Pourtant la mise en œuvre d'un tel processus n'en demeure pas moins fort délicate et compliquée, entraînant généralement des problèmes d'exactitude par l'omission ou l'excédent de termes d'indexation, ainsi que des problèmes de consistance lors de désaccords, que ce soit sur un même document ou sur des documents sémantiquement équivalents.

On oublie souvent que la qualité finale de tout système de recherche d'information s'appuie fortement sur la qualité de son indexation [Soergel,94]. En effet, bon nombre de travaux fondent l'évaluation du système d'indexation sur sa performance pour la recherche de documents : examiner les résultats d'interrogations menées par des utilisateurs différents sur le même ensemble de textes à partir de l'index produit, et mesurer le rappel et la précision afin de décider de la qualité de tout le système. L'évaluation de la qualité de l'indexation est alors masquée par l'évaluation globale du système de recherche d'information.

Pourtant, coûteux et souvent irréversible, le processus d'indexation doit, de notre point de vue, être validé en amont, avant d'être intégré dans le système de recherche d'information. Certains travaux [Soergel,94] soutiennent ce point de vue, en insistant sur l'impact qualitatif global de l'indexation sur le système de recherche d'information et en proposant différentes mesures de qualité de l'indexation.

4.1. Evaluation globale du système

Les méthodes d'évaluation généralement proposées examinent les résultats de recherches effectuées par des utilisateurs sur un corpus indexé, et mesurent le rappel et la précision afin d'évaluer le système.

Le rappel mesure la capacité du système de RI à trouver, pour une requête, *tous* les documents pertinents. Le rappel peut donc se définir comme la probabilité pour un document d'être retrouvé, sachant qu'il est pertinent.

$$\text{Rappel} = \frac{\text{nb documents pertinents et retrouvés}}{\text{nb documents pertinents}}$$

La précision mesure la capacité du système de RI à trouver, pour une requête, *uniquement* les documents pertinents. La précision est une mesure très intéressante pour mesurer la qualité des réponses du point de vue de l'utilisateur.

$$\text{Précision} = \frac{\text{nb documents pertinents et retrouvés}}{\text{nb documents retrouvés}}$$

C'est principalement sur ce type d'indicateurs que se fondent les méthodes d'évaluation mises en pratique lors des conférences TREC¹, qui ont mis en place un protocole pour évaluer les systèmes automatisés de recherche d'information dans de gros volumes de données.

D'autres mesures telles que l'élimination et l'hallucination (qui mesurent la capacité du système à éliminer tous les documents pertinents), et le bruit et le silence (complémentaires respectifs de la précision et du rappel) sont moins souvent utilisées. Le choix de l'utilisation du couple rappel/précision vient du fait qu'il donne une mesure orientée utilisateur des performances du système.

Ce type d'évaluation, largement utilisé, qualifie le processus de recherche en entier, englobant l'indexation, l'interprétation du besoin et la fonction de correspondance. Pourtant, une mauvaise indexation influera sur la qualité globale du système surtout si l'indexation est fondée sur un langage complexe. Qui plus est, l'indexation est un processus souvent coûteux et irréversible, nous nous intéressons donc à des méthodes qui ciblent le processus d'indexation. Ces méthodes, qui n'impliquent ni la fonction de correspondance, ni la formulation des requêtes, confrontent l'indexation d'un corpus à d'autres indexations de ce même corpus afin de mesurer sa qualité.

4.2. Evaluation de l'indexation

Nous nous intéressons ici à des méthodes, qui n'impliquent ni la fonction de correspondance, ni la formulation des requêtes dans la vérification qualitative du système, mais qui confrontent l'indexation d'un corpus à d'autres indexations de ce même corpus afin de vérifier sa qualité, autrement dit son *exactitude* et sa *consistance* (§4.2.1). Des mesures qui permettent de mesurer cette qualité de l'indexation ont été proposées. Nous les présentons dans §4.2.2.

¹ <http://trec.nist.gov>

4.2.1. Qualité de l'indexation : Exactitude et consistance

Une indexation doit être exacte et donc contenir *tous les et uniquement les* termes d'indexation corrects du document. Le processus d'indexation doit, par conséquent, éviter deux types d'erreurs : les erreurs d'omission (oubli d'un terme d'indexation) et les erreurs d'excédent (indexation par un terme incorrect).

Par ailleurs, l'indexation doit être consistante, autrement dit, le processus d'indexation doit assurer qu'un même document ou que des documents similaires aient toujours la même forme indexée. Lorsque le processus est manuel, le problème est posé sous la forme : est-ce que deux indexeurs différents indexent de la même façon le même document (consistance inter-annotateurs) ? Lorsqu'il est automatique, le problème est posé sous une autre forme : est-ce que deux documents de même contenu sémantique, mais d'expressions différentes, sont indexés de la même façon (consistance intra-annotateurs) ?

La Figure 7 résume l'exactitude et la consistance de l'indexation.

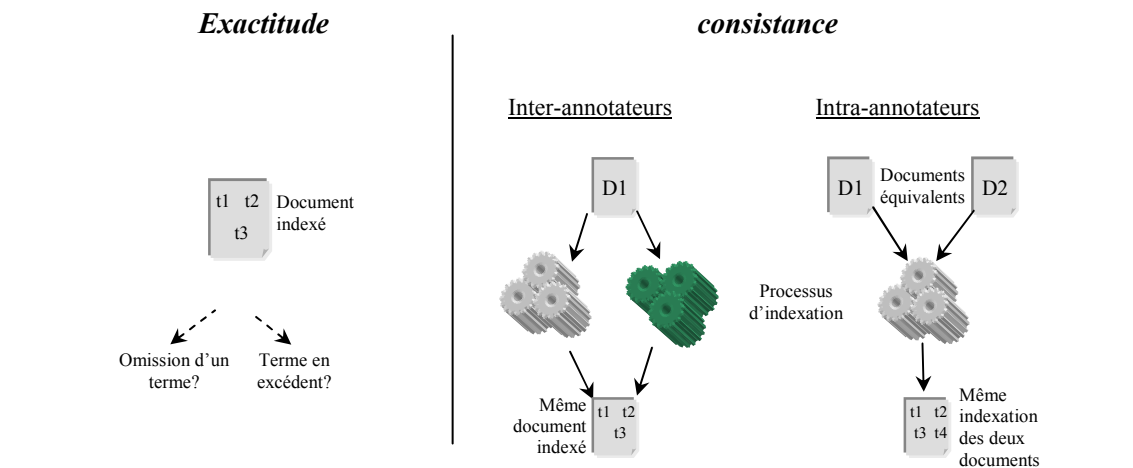


Figure 7 Exactitude et consistance de l'indexation

4.2.2. Mesures pour la qualité de l'indexation

Les mesures de qualité de l'indexation sont perçues selon deux points de vue :

- (a) Un point de vue document pour vérifier que chaque document est correctement indexé
- (b) Un point de vue terme pour s'assurer que chaque terme est correctement associé aux documents qu'il indexe.

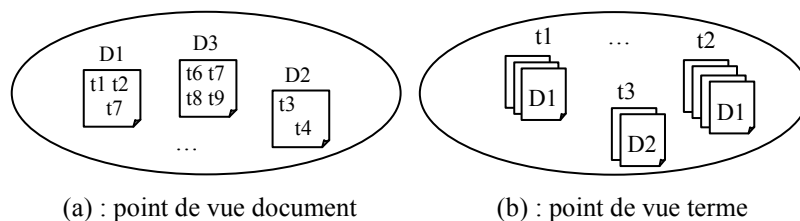


Figure 8 Les deux points de vue de l'indexation

Dans ces deux cas, la qualité de l'indexation est fonction de son exactitude, en terme de complétude et de pureté, et de sa consistance [Soergel,94]. Une bonne indexation doit assurer des mesures de complétude, de pureté et de consistance proches de 1.

Ces mesures données sous forme de formules dans le Tableau 1 sont explicitées ci-après.

		Point de vue document D	Point de vue terme t
Exactitude	Complétude	$\frac{\text{nb termes correctement affectés à D}}{\text{nb termes qui devraient être affectés à D}}$	$\frac{\text{nb documents correctement indexés par t}}{\text{nb documents qui devraient être indexés par t}}$
	Pureté	$\frac{\text{nb termes correctement rejetés pour D}}{\text{nb termes qui devraient être rejetés pour D}}$	$\frac{\text{nb documents correctement non indexés par t}}{\text{nb documents qui ne devraient pas être indexés par t}}$
Consistance		$\frac{\text{nb termes affectés à D par les indexeurs A et B}}{\text{nb termes affectés à D par les indexeurs A ou B}}$	$\frac{\text{nb documents indexés par t par les indexeurs A et B}}{\text{nb documents indexés par t par les indexeurs A ou B}}$

Tableau 1 Tableaux récapitulatifs des mesures de qualité d'indexation existantes

a. Exactitude

L'indexation est exacte lorsqu'il n'y a aucune erreur d'omission ou d'excédent de termes d'indexation, mais une telle indexation est difficile à générer. Aussi existe-il des mesures permettant de caractériser son exactitude.

Complétude

La complétude de l'indexation est liée à la présence des descripteurs corrects. D'un point de vue document, elle mesure le taux de termes affectés au document par le processus d'indexation par rapport à ceux qui devraient l'être. D'un point de vue terme, elle mesure le taux de documents indexés par chaque terme par rapport à ceux qui devraient l'être.

Plus précisément,

$$\text{Complétude}(D) = \frac{\text{nb termes correctement affectés à D}}{\text{nb termes qui devraient être affectés à D}}$$

$$\text{Complétude}(t) = \frac{\text{nb documents correctement indexés par t}}{\text{nb documents qui devraient être indexés par t}}$$

Cette mesure correspondant au *rappel* (le rappel calcule le rapport entre les documents retrouvés et pertinents et ceux retrouvés) est très intéressante pour prédéterminer les performances du système de recherche.

Pureté

La pureté de l'indexation est liée à l'absence de descripteurs erronés affectés au document. Elle est considérée par rapport à un document pris individuellement, ou par rapport à un terme et son occurrence (ou non) dans les documents du corpus.

Plus précisément,

$$\text{Pureté}(D) = \frac{\text{nb termes correcteme nt rejetés pour } D}{\text{nb termes qui devraient être rejetés pour } D}$$

$$\text{Pureté}(t) = \frac{\text{nb documents correcteme nt non indexés par } t}{\text{nb documents qui ne devraient pas être indexés par } t}$$

Cette mesure est intéressante pour prédéterminer les performances du système de recherche puisqu'elle correspond à *l'élimination* (l'élimination calcule le rapport entre les documents non pertinents et non retrouvés et les documents non pertinents).

b. Consistance

Assurer que des documents similaires aient toujours la même forme indexée quels que soient les indexeurs, est un problème délicat aussi bien dans le cas d'une indexation manuelle qu'automatique. Il est à noter que même si la consistance est une condition nécessaire au bon fonctionnement d'un système, celle-ci ne garantit pas son exactitude [Cooper,69]. Dans tous les cas, que l'on raisonne d'un point de vue document ou terme, la consistance peut être mesurée en comparant les réponses entre deux indexeurs différents A et B (inter-indexeurs), ou entre deux indexations A et B différentes d'un même indexeur sur le même document lors de sessions différentes A et B (intra-indexeur), ou entre deux indexations A et B différentes d'un même indexeur sur deux documents à sémantique équivalente (intra-indexeur).

Plus précisément,

$$\text{Consistance}(D) = \frac{\text{nb termes affectés à } D \text{ par les indexeurs A et B}}{\text{nb termes affectés à } D \text{ par les indexeurs A ou B}}$$

$$\text{Consistance}(t) = \frac{\text{nb documents indexés par } t \text{ par les indexeurs A et B}}{\text{nb documents indexés par } t \text{ par les indexeurs A ou B}}$$

4.3. Synthèse des mesures pour la validation d'indexation

La qualité d'un système de recherche d'information repose avant tout sur une bonne indexation. Cette dernière étant un processus coûteux et souvent irréversible, nous insistons sur l'importance de sa validation en amont, avant de l'englober dans un système de recherche.

Afin de mesurer la qualité d'une indexation, des mesures établies pour des langages d'indexation à base de mots clés ont été proposées [Soergel,94]:

- l'exactitude : elle est mesurée en terme de complétude et de pureté. Ces deux mesures calculent des taux par rapport à une indexation de référence : les termes qui devrait être affectés ou rejetés par le processus d'indexation. Des mesures qui calculeraient des taux par rapport aux termes réellement affectés par ce processus

n'apparaissent pas. Pourtant, dans le cas où le processus d'indexation n'utilise pas de vocabulaire prédéfini, il est difficile de calculer la pureté d'une telle indexation. Estimer l'absence d'erreurs d'indexation par des termes incorrects peut difficilement être mesuré par la pureté. Il manque une mesure liée à l'absence de descripteurs erronés affectés au document qui serait définie par rapport à ce qui a effectivement été indexé.

- la consistance : elle est mesurée en terme d'accord entre indexeur(s). Nous pensons toutefois, que les mesures proposées peuvent être biaisées par des accords aléatoires entre indexeurs.

Avec l'évolution des langages d'indexation et le développement de langages complexes (par opposition à langages à base de mots clés), les mesures de qualité proposées pour des langages à mots clés ne conviennent plus. D'autres mesures ou une adaptation de ces dernières doivent alors être fournies dans le but de permettre l'évaluation de processus d'indexation fondée sur des langages complexes.

5. Conclusion

Nous avons présenté ici différents choix possibles pour la modélisation des systèmes de recherche d'information. Les choix effectués (jusqu'à quel point le langage d'indexation est expressif, que permet d'exprimer le langage d'interrogation...) influent sur le comportement du système et donc sur la satisfaction des utilisateurs. Ainsi, pour des systèmes orientés précision, il est nécessaire d'avoir une certaine rigueur dans la description des documents, et dans ce cas, le choix d'un langage d'indexation complexe s'impose. De même, le besoin de l'utilisateur et la vision qu'il en a (qu'est ce qui est important, certain, tolérable, etc. que les documents contiennent) doivent être gérés de façon adéquate par le langage d'interrogation et la fonction de correspondance pour de meilleures performance de la recherche et donc une meilleure satisfaction des utilisateurs. C'est pour cela que le choix de la représentation des informations (document et requête) et de la fonction de correspondance, est si important en recherche d'information. Dans la partie 2 de ce manuscrit, nous effectuons ces choix, en proposant un modèle de recherche d'information qui repose sur un langage complexe et une formulation du besoin exprimant l'importance que donne l'utilisateur aux unités de la requête ainsi que ses incertitudes quand à son besoin.

Les choix de modélisation ayant pour but la satisfaction des utilisateurs, nous nous intéressons à la qualité des systèmes de recherche, et plus précisément, à la qualité de l'indexation des documents. Les mesures qualitatives que nous avons décrites ici, seront adaptées à des indexations fondées sur des langages complexes, et utilisées pour valider la qualité de notre indexation dans la dernière partie de ce manuscrit (partie 4).

Chapitre II. Documents techniques et usages: Un accès par le graphique

Comme cela a été mentionné dans nos objectifs, nous nous situons dans un contexte où les utilisateurs sont des experts du domaine qui ont une mémoire des documents qu'ils manipulent, et où ils expriment leur besoin en information en décrivant le document qu'ils souhaiteraient retrouver et dont ils ont souvenir. Afin de concrétiser un tel contexte, nous avons choisi la recherche d'information dans les documents techniques par des utilisateurs professionnels (les réparateurs). En effet, dans un tel contexte, les utilisateurs sont des experts du domaine qui consultent régulièrement la documentation technique. Donc, lorsqu'ils ont besoin d'une information, ils savent vaguement où la trouver, et peuvent donc décrire avec plus ou moins de précision le document susceptible des satisfaire. Plus particulièrement, dans la documentation technique, le média graphique est omniprésent, et représente une information souvent consultée par les réparateurs techniques. Le fait que cette information soit visuelle renforce l'idée de mémorisation des documents.

Dans ce chapitre, nous nous intéressons à ce cadre applicatif. La première partie est une description des caractéristiques de ce contexte particulier : caractéristiques des documents techniques et besoins et habitudes de travail des utilisateurs. La seconde partie s'intéresse aux graphiques qui sont omniprésents dans la documentation technique, et à la prise en compte de ce média dans la recherche d'information technique. Un tour d'horizon des modèles d'indexation d'un tel média est alors proposée.

1. Description du contexte : documents et utilisateurs

Cette description du cadre applicatif a pour but de faire ressortir les caractéristiques propres à ce domaine, et de dégager la problématique de la recherche dans les documents techniques à usage professionnel.

1.1. Le document technique : définition et caractéristiques

La notion de document technique recouvre, au sens large, des réalités extrêmement diverses : en ce sens très général, un texte de loi est un document technique de même qu'un rapport financier relatif à une entreprise. Celui qui nous intéresse ici est le document de type manuel d'utilisation, d'entretien, de réparation, etc. de dispositifs techniques. Ce document, fortement structuré, véhicule des savoirs et des savoir-faire propres à un champ technique particulier. Une définition de ces documents est donnée par [Montmollin,96] : il s'agit d'un «type de document visant à guider, par une liste d'instructions organisées, un utilisateur dans la réalisation d'une tâche. »

Le fait que ces documents constituent la principale, voire l'unique, source d'information disponible en terme d'aide à la réalisation de tâches et/ou à la résolution de difficultés

[Allwood,97], nécessite qu'ils soient facilement lisibles et compréhensibles par l'utilisateur, autrement dit facilement utilisables et correspondant aux besoins des utilisateurs en situation de travail (utilisation, entretien, réparation, etc. d'un dispositif). Pour répondre à cette contrainte, les psychologues du travail et les ergonomes se sont intéressés à la bonne compréhension de ces documents par leurs destinataires lorsqu'ils sont en situation de travail, ce qui a abouti à l'élaboration d'un ensemble de recommandations portées à l'attention des rédacteurs, recommandations qui sont publiées sous forme de recueils par certains organismes publics ou privés ([CEP,83][ISO/CEI,95]). Les éléments sur lesquels portent ces recommandations sont, entre autres, la structure du document, la syntaxe du texte (style clair, vocabulaire technique, etc.), l'articulation entre différents formats de présentation de l'information (par exemple, la combinaison d'énoncés verbaux et de schémas techniques induit généralement des traitements plus efficaces de l'information) [Ganier,03][Ganier,02].

1.1.1. Structure du document

Il n'existe pas une unique structure mais au moins deux types de structure pour décrire un document structuré, à savoir, la structure logique et la structure physique [Fourel,97] : la première définit une organisation hiérarchique des données du document (un document est composé d'un titre et de sections qui sont composées à leur tour de titres et de sous-sections, etc.) alors que la seconde définit une organisation externe des données du document (un document est composé de pages qui sont composées à leur tour de blocs, etc.) Lorsque nous parlons de structuration, nous entendons la structure logique. Celle-ci représente un aspect important pour le lecteur. Par exemple, les titres sont d'un grand secours au lecteur puisqu'ils lui fournissent une structure explicite qui leur permet d'intégrer plus facilement l'information au cours de la lecture. Ils lui permettent également d'ajuster ses objectifs de lecture à ses besoins. Les documents techniques respectent cette organisation hiérarchique des données : ce sont des documents fortement structurés formant une arborescence de blocs, éventuellement un graphe.

Indépendance entre blocs de la structure : exploitables partie par partie

Contrairement à d'autres catégories de documents, les manuels techniques sont constitués de parties souvent indépendantes, chacune formant un tout : ils sont conçus pour une lecture non linéaire, et sont donc exploitables partie par partie [Badjo,00]. Cette indépendance justifie les approches actuelles des systèmes de recherche d'information structurée qui clament que l'un des blocs (ou un ensemble de blocs) formant la hiérarchie du document, peut représenter une réponse plus pertinente à la requête de l'utilisateur que le document en entier.

Dépendance entre blocs de la structure : Propagation de l'information

Loin d'être indépendants, certains blocs de la hiérarchie des documents structurés sont reliés entre eux par des relations qui permettent de donner au document une intégrité sémantique. Dans ce cas, il existe une propagation de l'information entre ces blocs. Ces blocs peuvent être, soient des blocs de la hiérarchie reliés par une relation de composition (voir Figure 9), et dans ce cas, il y a une propagation descendante de l'information, soient des blocs d'un même niveau de la hiérarchie exprimant une même information mais s'enrichissant mutuellement : c'est le cas entre un graphique technique et son commentaire textuel, par exemple (voir Figure 10). Cette dépendance texte et graphique dans la documentation technique est revue en détail dans le § 1.1.3.

Certaines recherches ont tenu compte de cette dépendance entre blocs de la hiérarchie du document dans le but d'enrichir l'indexation d'un bloc par l'ajout des index des blocs avoisinants ou supérieurs hiérarchiquement (la notion de portée, par exemple). Plus particulièrement, dans les documents structurés multimédia, cette utilisation du contenu sémantique des blocs avoisinants pour enrichir l'indexation d'un bloc non textuel est nécessaire. En effet, certains blocs, comme les images et les graphiques, manquent d'un langage permettant d'en exprimer la sémantique, et même si ces blocs contiennent en eux-mêmes des informations permettant leur description, celle-ci reste pauvre sémantiquement et ne permet pas de représenter les données en question de façon suffisante et par conséquent de les retrouver efficacement.

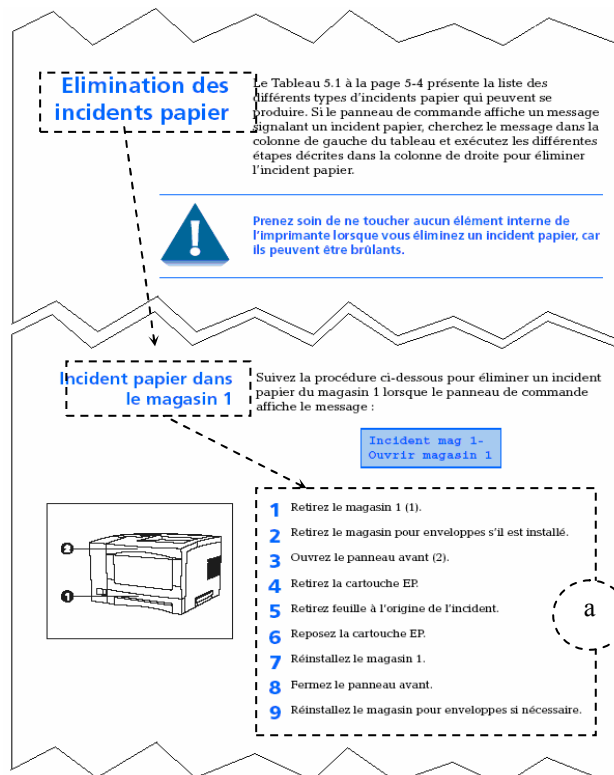


Figure 9. Exemple de propagation de l'information dans la hiérarchie d'un document technique

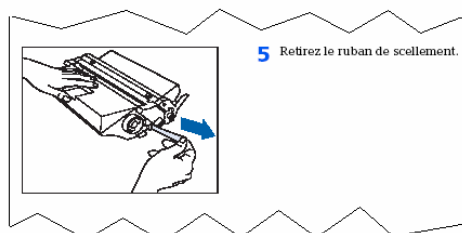


Figure 10. Exemple de dépendance entre partie textuelle et partie graphique

1.1.2. Syntaxe du texte procédural

Le style du texte dans les documents technique est précis et direct. En effet, les sujets particulièrement complexes sont plus facilement compris lorsqu'ils sont présentés de manière synthétique (par opposition à la prose). Par exemple, les actions à effectuer en vue de réaliser une opération, apparaissent souvent dans une liste numérotée et sous forme de phrases impératives courtes afin d'être facilement assimilées par le lecteur.

Exemple

- 1 Retirez le magasin 1.
- 2 Retirez le magasin pour enveloppes s'il est installé.
- 3 Ouvrez le panneau avant.
- 4 Retirez la cartouche EP.

D'un autre côté, les phrases complexes, composées d'une imbrication de propositions, ou de multiples propositions subordonnées, apparaissent rarement si ce n'est presque jamais dans un texte technique. Ceci est dû au fait que ce type de phrases peut s'avérer parfaitement incompréhensible [Wright,77].

Toujours dans un souci de faciliter la lecture et la compréhension du texte technique, celui ci se spécifie par la répétition des noms et notions techniques qui se traduit le plus souvent par la rareté volontaire des anaphores, une faible utilisation de synonymes ainsi qu'une faible utilisation de pronoms.

1.1.3. Articulation de différents formats : cas du texte et du graphique

Le texte et le graphique forment un couple omniprésent dans les documents techniques. Le texte accompagne toujours le graphique, et le texte est quasiment toujours accompagné d'un graphique. Graphique et texte sont complémentaires : le texte exprime ce qui est difficile à montrer dans un graphique et réciproquement, le graphique montre ce qui est difficile à exprimer verbalement. Il y a donc une forme de relais entre les deux média puisque l'un aide l'autre dans le message qu'il souhaite communiquer :

- D'un côté, les graphiques des documents techniques contiennent des données spatiales et de nombreuses expériences ont montré que lorsqu'une personne est confrontée à un énoncé de "spatialisation" organisant des objets les uns par rapport aux autres, la personne soumise à cet énoncé construit mentalement une représentation de la scène [Michel,97]. Le graphique décharge donc le lecteur d'un effort supplémentaire de modélisation de l'information géométrique.
- D'un autre côté, les propriétés (descriptions, fonctions, etc.) des objets représentés dans le graphiques ne sont pas toujours explicites dans ce graphique mais sont mentionnées dans le texte qui lui correspond, ce qui va reconstituer l'interprétation complète du graphique : selon [Joly,93] «les mots vont compléter l'image. »

Donc, l'association du texte et du graphique permet une compréhension rapide et complète de leur contenu : lors de la lecture du texte, les données qui y sont évoquées sont facilement localisables dans le graphique associé. L'effort de modélisation est ainsi très largement diminué. De même, lors de la visualisation du graphique, la description des éléments qui y sont représentés est aisément retrouvée dans le texte. L'interprétation du graphique est ainsi

complète. Le graphique et le texte présentent ainsi une même information exprimée de deux façons différentes mais complémentaires.

1.2. Les utilisateurs professionnels : caractéristiques

Le document technique s'adresse à un groupe socialement et professionnellement homogène d'utilisateurs qui effectuent une activité opératoire et partagent des connaissances communes. Ces utilisateurs professionnels ont des habitudes de travail communes qui permettent de caractériser leurs interactions avec les documents. Nous nous intéressons aux trois points suivants : les modes de consultation, le type d'information recherchée et leurs connaissances du contenu des documents.

1.2.1. Modes de consultation des documents

[Wright,99] considère qu'il existe principalement deux modes de consultation des documents techniques. Le premier mode se traduit par une lecture linéaire des documents préalablement à la manipulation de l'équipement : ce mode concernerait principalement les utilisateurs débutants ou « précautionneux ». Dans ce cas, la consultation du document guide les interactions de l'utilisateur avec l'équipement. Le second mode se traduit par une consultation ponctuelle ou sélective des documents, par exemple lorsqu'un problème survient alors que l'utilisateur manipule l'équipement ou en cas de doute afin de confirmer une action : ce mode concernerait principalement les utilisateurs expérimentés ou préférant manipuler d'abord l'équipement (du fait d'un usage préalable d'un équipement similaire, par exemple). Dans ce deuxième cas, la consultation du document est guidée par la tâche à réaliser. Un exemple de ce type de consultation est observé lors d'une étude réalisée dans un environnement réel de production [Brinkman,01]. Cette étude montre que des techniciens devant remplacer une caméra de surveillance d'une machine n'utilisaient le manuel que lorsqu'ils rencontraient une difficulté. Même s'il n'existe pas réellement de données prouvant cette tendance, la consultation est généralement sélective.

Dans ce travail, nous nous intéressons à la consultation sélective des documents qui résulte d'un besoin en information déclanchant ainsi une recherche de l'information utile dans la documentation technique.

1.2.2. Type d'information recherchée

La recherche d'information dans les documents techniques est essentiellement à visée opératoire, pour réaliser une tâche ou une action [Vigner,76]. Lorsqu'il manipule un équipement, l'utilisateur consulte les documents pour trouver rapidement des informations, et ce lorsqu'il hésite quant à la démarche à adopter (il lui manque un détail, par exemple), ou dans le cas où le résultat de manipulation de l'équipement ne correspond pas au résultat attendu. Autrement dit, l'utilisateur désire trouver une information nécessaire à la prise d'une décision ou à la résolution d'une difficulté technique rencontrée en situation de travail. La consultation est dans ce cas sélective : il s'agit d'une recherche d'information dans le but de savoir pour faire.

a. Descriptions vs tâches

Les documents techniques visent à informer les utilisateurs de la réalité du *fonctionnement* et des *propriétés* d'un dispositif technique. Deux types d'informations semblent alors susceptibles d'être sollicitées par les utilisateurs : des descriptions concernant un équipement et des

instructions à suivre pour réaliser une tâche. Une observation d'experts techniques de différents domaines [Paganelli,97] a permis de valider cette hypothèse en collectant un corpus de requêtes servant à obtenir une information technique. Ce corpus a permis de distinguer deux types de requêtes :

- Des requêtes portant sur un objet, du type « *Qu'est ce que 'x' ?* » : dans ce cas, l'utilisateur a besoin d'une description de l'objet « *x* » et de ses composants ainsi qu'une description de leurs propriétés et fonctions.
- Des requêtes portant sur la réalisation d'une tâche, du type « *Comment faire pour 'x' ?* » : L'utilisateur a dans ce cas besoin d'informations indiquant quelles actions accomplir afin de réaliser la tâche en question.

Cette distinction concernant le type de la requête posée par les utilisateurs de la documentation technique a été prise en compte dans le système de consultation de documentation technique Sysrit proposé par [Paganelli,02a] [Clavier,97].

b. Informations précises

La recherche d'information dans les documents techniques est une recherche précise. Elle ne porte pas sur un sujet général mais plutôt sur un aspect de ce sujet. L'utilisateur aura besoin de savoir précisément « comment résoudre un incident papier *dans le magasin1* de l'imprimante Docuprint N17? » et non pas tout savoir sur « l'imprimante Docuprint N17 » ou même sur « la résolution d'incidents papier dans l'imprimante Docuprint N17. »

Dans les documents structurés, en général, et selon le besoin de l'utilisateur, l'un des blocs formant la hiérarchie du document peut représenter une réponse plus pertinente à la requête de l'utilisateur que le document en entier. C'est le cas pour la documentation technique. Des expériences rapportées dans [Paganelli,97] [Paganelli,99] ont permis de dégager les caractéristiques des réponses jugées pertinentes par les utilisateurs experts : les réponses satisfaisantes apparaissent comme étant des passages de texte repérés dans une ou plusieurs unités hiérarchique du document (unités repérées par un titre dans le sommaire) et selon [Paganelli,02b] « Les sujets ont tendance à choisir des unités d'information plus petites et plus précises que les unités hiérarchiques. »

Certains travaux en recherche d'information se sont intéressés à la segmentation des textes avec pour préoccupation, en premier lieu, de répondre au souci de ne pas noyer l'utilisateur sous une masse trop importante de documents en réponse à une requête lorsque les documents sont de taille importante et, en second lieu, d'aboutir à une plus grande efficacité de la recherche elle même. Parmi ces travaux, nous pouvons citer ceux de [Ouerfelli,00] dont l'objectif est la segmentation du texte technique en unités documentaires fines (reflétant l'articulation logique des différents aspects qui y sont traités) pour un accès plus précis et plus localisé à l'information à l'intérieur de la structure globale du texte.

1.2.3. Connaissances du contenu des documents

Les manuels d'utilisation des composants matériels sont des documents utilisés fréquemment par les professionnels. Ces utilisateurs connaissent globalement le contenu des documents qu'ils manipulent et lorsqu'ils recherchent une information, il s'agit souvent d'un manque de détail et non pas d'une ignorance totale de ce qui est recherché. En effet, ils souhaitent *retrouver* (et non pas trouver) un fragment précis d'information qu'ils ont déjà rencontrés lors de précédentes consultations: ils savent a priori ce qu'ils recherchent et désirent retrouver une information dont

ils se souviennent vaguement. La description qu'ils font de l'information recherchée peut alors être imprécise et/ou incomplète.

De plus, en considérant l'omniprésence des graphiques dans ce type de documents, il paraît logique de penser que ces graphiques, fréquemment consultés, représentent une information facilement mémorisable par l'utilisateur. En effet, de nombreuses expériences ont montré que lorsqu'une personne est confrontée à un énoncé de "spatialisation" organisant des objets les uns par rapport aux autres, elle construit mentalement une représentation de la scène [Michel,97]. Cette représentation mentale augmentée d'une compréhension du graphique par l'utilisateur entraîne une mémorisation visuelle du graphique (voir §2.2.2) qu'il serait intéressant d'exploiter pour accéder à l'information utile dans le document. Ainsi, un utilisateur pourrait, par exemple, vouloir retrouver le graphique décrivant le chargement d'incidents papiers dans le magasin d'alimentation manuelle et qui, dans son souvenir, contient un zoom en haut à droite et une flèche descendante du côté gauche.

1.3. Bilan

Dans le contexte de la recherche d'information dans la documentation technique, les informations manipulées et les utilisateurs concernés permettent d'envisager une nouvelle approche pour l'accès à l'information technique utile dans une situation de recherche particulière « chercher une information pour réaliser une tâche » :

- La structuration des documents et le mode de lecture « sélectif » adopté par les professionnels permettent d'envisager qu'une unité du document est une réponse plus pertinente à une requête que le document en entier.
- Le graphique contenu dans les documents, dont la forte présence a pour but d'aider l'utilisateur à comprendre l'information textuelle, est une information pertinente que l'utilisateur désire retrouver. Le graphique doit donc faire partie des blocs à renvoyer comme réponses à l'utilisateur.
- La consultation répétitive des documents par les professionnels ainsi que leurs connaissances du domaine entraînent une mémorisation du contenu consulté. Cette information mémorisée est incomplète et imprécise pour satisfaire l'utilisateur. Par contre, elle peut être considérée comme une requête possible pour retrouver dans le document l'information exacte et utile.

Le graphique étant un média facilement mémorisable visuellement, nous envisageons de le considérer comme un point d'accès aux documents techniques, autrement dit, comme une requête probable pour accéder à une zone (unité) précise du document représentant le graphique mémorisé, avant de naviguer, éventuellement, dans le reste du(des) document(s). Ce qui coïncide parfaitement avec le contexte de recherche dans lequel nous nous positionnons, dans notre travail.

2. Le graphique : un point d'accès au document technique

Comme cela a déjà été mentionné, l'association du texte et du graphique permet une compréhension rapide et complète de leur contenu : lors de la lecture du texte, les données qui y sont évoquées sont facilement localisables dans le graphique associé. L'effort de modélisation est ainsi très largement diminué. De même, lors de la visualisation du graphique, la description

des éléments qui y sont représentés est aisément retrouvée dans le texte. L'interprétation du graphique est ainsi complète.

Habituellement, la recherche d'information dans la documentation technique se fait à travers le texte : à partir d'une requête textuelle telle que « *comment faire pour dégager un incident papier dans le bac supérieur des imprimantes z ?* », les blocs textuels répondant à cette requête sont retournés à l'utilisateur, accompagnés éventuellement des graphiques illustrant ce texte. Pourtant, comme nous l'avons montré précédemment, la consultation fréquente des documents et la sémantique rattachée aux graphiques favorise leur mémorisation. Il s'agit donc d'une information pouvant représenter un bon moyen d'accéder à la partie qui intéresse l'utilisateur. Le graphique n'est donc plus considéré comme un média d'accompagnement ne servant qu'à aider à la compréhension du texte par le lecteur, mais comme un bloc riche en information qui peut être considéré comme une information clé utile à l'utilisateur.

Un travail de recherche est allé dans ce sens en considérant le graphique comme une information ayant la même importance que le texte, et qui doit faire partie des réponses à renvoyer à l'utilisateur lors de sa recherche. Ce travail est présenté dans ce qui suit.

2.1. Un exemple d'application dans la recherche des documents techniques

[Badjo,00] ont proposé une méthode pour l'indexation des documents techniques qui traite de façon aussi similaire que possible les éléments textuels et ceux non textuels au niveau de la RI. Il s'agit d'une extension d'une méthode d'indexation de documents techniques structurés qui ne considérerait que les éléments textuels dans le modèle du document, modèle représenté sous forme hiérarchique selon la structure logique du texte (chapitres, sections, etc.).

Pour prendre en compte les éléments non textuels de leurs corpus, [Badjo,00] ont inséré dans la structure du document un nouvel élément textuel *DescElTextuel* correspondant à l'élément graphique et qui apparaît alors dans la représentation structurelle du corpus au même titre que les éléments textuels. Le graphique pouvant être structuré, le schéma de la structure est dans ce cas étendu en associant des descriptifs textuels à chaque niveau hiérarchique de la structure du graphique. La Figure 11 montre un exemple de modèle d'un document qui contient un élément graphique structuré. Ce graphique représentant « un panneau d'alimentation générale » est constitué de deux sous éléments, soient « un disjoncteur principal » et « un disjoncteur courant continu ».

La méthode proposée par [Badjo,00] permet d'intégrer dans l'index d'interrogation les éléments qui permettent de retrouver les unités graphiques contenues dans le document technique. Cependant, le graphique est considéré comme une unité textuelle, élément de la structure hiérarchique du document, et non pas comme un média particulier, ayant un contenu sémantique et des données propres. Ainsi, les informations visuelles telles que la disposition des composantes du graphique et leurs formes ne sont pas considérées dans ce modèle et l'interprétation sémantique, comme le fonctionnement décrit par le graphique, est également ignorée.

1 :	Titre=	Documentation de maintenance système Bull GCOS-7
2 :	Section1=	Imprimante ligne PRU 1115 et 1515
3 :	Sous-section 1.1=	...
...		
7 :	Sous-section 2.2=	...
8 :	Paragraphe 2.2.5=	...
9 :	Texte=	...
10 :	Texte=	...
...		
12 :	EltNonTexStruct=	Fig. 2-5 Panneau d'alimentation électrique
13 :	DescElnonText=	Disjoncteur principal
14 :	DescElnonText=	Disjoncteur courant continu

Figure 11 Structure d'un document intégrant un élément non textuel

2.2. Graphiques et mémoire

Les propriétés visuelles du graphique nous semblent d'une grande importance dans le cadre de la recherche d'information dans les documents techniques. En effet, comme nous l'avons mentionné précédemment, les graphiques sont souvent visualisés, en raison de la consultation fréquente du document technique et de leur utilité pour la compréhension des tâches à accomplir. Ceci a pour effet que l'utilisateur construit une représentation mentale du graphique pouvant être utilisée comme requête pour retrouver l'information utile.

Afin de confirmer cette hypothèse de mémorisation visuelle du graphique par les utilisateurs professionnels, nous nous intéressons au fonctionnement de la mémoire en général et de la mémoire visuelle en particulier d'un point de vue cognitif, afin de vérifier qu'il y a bien mémorisation visuelle des graphiques et que ce qui a été mémorisé peut servir pour retrouver les graphiques réels.

2.2.1. La mémoire humaine: généralités

La mémoire humaine est un système permettant le stockage et la récupération de l'information, laquelle est naturellement transmise par nos sens. Lorsque nous voyons, entendons, ou sentons quelque chose, cela ne peut qu'influer nos souvenirs et nous stockons alors ce que nous percevons. La trace de ce que nous stockons peut persister quelques secondes ou plusieurs jours, ce qui laisse à penser que l'enregistrement des percepts est réalisé en différentes étapes de traitement et de mémorisation et que nous ne possédons pas une mémoire unitaire.

A la fin des années soixante, un certain nombre de modèles représentant la mémoire virent le jour. Le plus caractéristique et le plus influent fut proposé par [Atkinson,68] : il s'agit du modèle modal. Ce modèle modal postule que l'information en provenance du monde extérieur est tout d'abord traité par le *registre sensoriel* qui transmet ensuite l'information à un *registre à court terme* de capacité limitée qui, à son tour, transmet l'information à un *registre à long terme*. Outre le stockage de l'information, le registre à court terme était également censé

effectuer un certain nombre de traitements appelés processus de contrôle contenant entre autres la *répétition* et le *codage sémantique* de l'information.

La notion de mémoire à court terme a ensuite été profondément renouvelée par le concept de mémoire de travail. Le modèle de la mémoire s'est alors vu divisé en quatre sous-systèmes principaux résumés dans la Figure 12:

- La **mémoire sensorielle** qui est une mémoire automatique, fruit de nos capacités perceptives, s'évanouissant généralement en moins d'une seconde.
- La **mémoire à court terme** qui dépend de l'attention portée aux éléments de la mémoire sensorielle. Elle permet de garder en mémoire une information pendant moins d'une minute environ et de pouvoir la restituer pendant ce délai. En général, nos facultés nous permettent de retenir entre 5 et 9 éléments.
- La **mémoire de travail** qui est une extension plus récente au concept de mémoire à court terme. Elle permet d'effectuer des traitements cognitifs sur les éléments qui y sont temporairement stockés.
- La **mémoire à long terme** qui comprend la mémoire des faits récents, où les souvenirs sont encore fragiles, et la mémoire des faits anciens, où les souvenirs ont été consolidés. Elle peut être schématisée comme la succession dans le temps de 3 grands processus de base : l'encodage, le stockage et la restitution des informations.

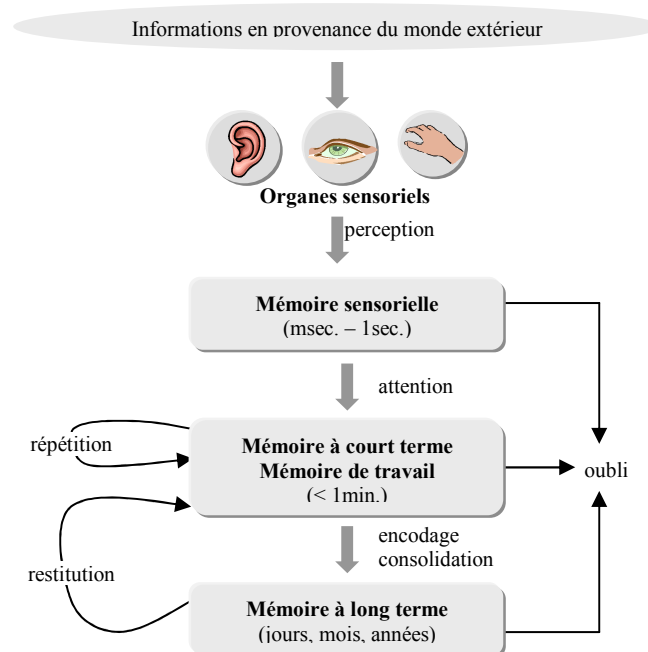


Figure 12 Passage de l'information dans le système mnésique

2.2.2. Critères pour une mémorisation à long terme de l'information

Les travaux de [Craik,72] ont établi deux généralisations à propos de la mémorisation :

- lorsque l'*encodage*, qui vise à donner un sens à la chose à remémorer, est élaboré et profond, il aboutit généralement à une meilleure mémorisation. Par exemple, le mot "citron" peut être encodé de la manière suivante : fruit, rond, jaune. Si ce mot n'est pas spontanément restitué, l'évocation d'un indice issu de l'encodage (par exemple, fruit) permettra de le retrouver.
- la *répétition* permet soit la maintenance de l'information sur une courte période, ce qui renforce peu la mémorisation, soit l'intégration d'information nouvelle à l'ancienne ce qui dans ce cas renforce d'avantage la mémorisation.

Pour résumer, nous énonçons que :

- de la profondeur de l'encodage, donc de l'organisation et de la structuration des données, dépendra l'efficacité de la mémorisation, et
- plus une information sera répétée, plus elle aura de chances d'être mémorisée.

2.2.3. La mémoire visuelle

« Nous sommes capables de nous rappeler l'image d'un coucher de soleil, nous pourrions probablement reconnaître une photographie d'Albert Einstein ou de Joseph Staline ... Tout cela révèle l'existence d'un certain stockage sensoriel à long terme. » Cette citation de [Baddeley,95] concerne la mémoire visuelle.

Cette mémoire peut persister bien au-delà de quelques secondes. [Rock,59] ont étudié la mémorisation d'une unique forme sans signification sur des périodes dépassant un mois : bien que la capacité à dessiner correctement la forme déclinait, leurs sujets étaient toujours capables de la reconnaître presque parfaitement parmi d'autres formes similaires quatre semaines plus tard.

Nous trouvons aussi une démonstration de la mémoire visuelle chez [Standing,70]. Ces auteurs présentèrent 2560 diapositives en couleurs (images) pendant 10 secondes chacune à un ensemble d'utilisateurs. La performance était ensuite évaluée en présentant des paires d'images, une ancienne et une nouvelle, et en demandant aux sujets laquelle des deux avait déjà été présentée. En dépit du nombre très important d'images, la performance se situait encore aux environs de 90% de réponses correctes après plusieurs jours.

Bien entendu, le fort taux de remémoration ne signifie pas que les sujets ont stocké continuellement une quantité colossale d'informations visuelles, mais il leur a suffi de stocker une quantité minimum d'informations permettant à une des images de paraître plus familière que l'autre. Cela signifie que :

quelque chose dans l'image est stocké et tout n'est pas récupéré.

Différents travaux ont été menés sur la mémoire visuelle à long terme et tous montrent que « notre mémoire n'est en aucune façon miraculeuse, mais le niveau de performance en reconnaissance est tout à fait impressionnant. » [Baddeley,95]

2.2.4. Mémoire visuelle et graphiques techniques

Les utilisateurs professionnels de la documentation technique ont deux spécificités :

- ils consultent régulièrement la documentation, d'où la *répétition* dans la visualisation des graphiques contenus dans cette documentation,

- ils ont des connaissances du domaine technique et lorsqu'ils consultent un graphique, ils ne voient pas uniquement des traits, mais ils comprennent le graphique et la sémantique qu'il contient. Les connaissances qu'ils ont autour de ce graphique permettent un *meilleur encodage* de cette information.

Ceci nous permet d'affirmer qu'il y a une mémorisation visuelle à long terme des graphiques consultés par les utilisateurs professionnels de la documentation technique. Et comme nous l'avons mentionné précédemment, « tout n'est pas récupéré » par la mémoire. Ainsi, nous pouvons dire que :

- tous les graphiques ne sont pas mémorisés,
- dans un graphique, tout n'est pas mémorisé,
- ce qui est mémorisé dans un graphique n'est pas précis (est approximatif).

2.2.5. Bilan

Le graphique dans les documents techniques est un média facilement mémorisable par les utilisateurs professionnels en raison de leurs habitudes de travail (consultation répétitive des documents) et de leurs compétences (compréhension des graphiques et de la sémantique qui les entoure). Et même si la mémorisation est visuelle, ceci n'empêche pas qu'une sémantique y est rattachée favorisant son maintien en mémoire. Le graphique est alors considéré comme un objet structuré qui offre des informations visuelles, intrinsèques et intéressantes (traits, flèches, positions des objets, etc.) et auquel est rattachée une sémantique (quelle procédure y est décrite, quels objets y sont représentés, etc.)

2.3. Indexation des graphiques et des images : un tour d'horizon

Historiquement, la recherche d'information s'est concentrée sur la façon d'extraire l'information à partir du texte. Bien que loin de la perfection, il existe des techniques bien développées pour analyser les concepts présents dans un texte et pour prévoir la pertinence d'un document par rapport à une requête. Un avantage important pour la recherche de textes est que les moyens mis en œuvre pour interroger une collection des textes sont ceux servant à décrire les documents eux-mêmes (en employant un ensemble de termes). La recherche des graphiques ou des illustrations en général, est plus complexe: il est plus difficile d'extraire la notion du contenu de l'illustration en raison de la variété des manières permettant de la décrire et, en plus, du manque d'un langage permettant d'en exprimer la sémantique; car même si les illustrations, contiennent en eux-mêmes des informations permettant leur description, celles-ci restent pauvres sémantiquement et ne permettent pas de représenter les données en question de façon suffisante et par conséquent de les retrouver efficacement. Le contexte dans lequel apparaît l'illustration est, dans ce cas, utile pour enrichir son indexation. Nous pouvons donc distinguer deux approches pour l'indexation des illustrations. La première se base sur une analyse de l'illustration, et dans ce cas, seules entrent en jeu ses données intrinsèques: il s'agit de l'approche syntaxique. La deuxième se base sur l'exploitation du contexte dans lequel elles apparaissent, et dans ce cas, il y a enrichissement de l'indexation de l'illustration par son contexte: il s'agit de l'approche sémantique.

Des exemples de travaux relatifs à chacune de ces deux approches concernant les graphiques mais aussi les images en général sont donnés dans ce qui suit.

2.3.1. Approche syntaxique

L'approche syntaxique consiste à réduire l'illustration (image, graphique, etc.) à un ensemble de paramètres physiques qui peuvent varier d'une approche à l'autre. Certains systèmes utilisent la couleur, d'autres les contours, d'autres des caractéristiques propres au signal. Cette approche syntaxique présente l'avantage de pouvoir être entièrement automatisable via des algorithmes de traitement d'images.

Nous nous intéressons ici au média graphique qui contient des données intrinsèques riches : formes, positions, tailles, etc. Ces données ont été exploitées afin de permettre la description de son contenu, autrement dit son indexation. Nous décrivons, dans ce qui suit, trois approches d'indexation et de recherche de graphiques s'appuyant sur leur traitement analytique.

L'approche de Lorenz et Monagan [Lorenz,94] [Lorenz,95]

Il s'agit d'une approche d'indexation automatique de graphiques basée sur leur contenu textuel et « graphique ». [Lorenz,95] considèrent le graphique comme une couche textuelle et une couche graphique. Ils distinguent alors deux classes de dispositifs d'indexation : graphique et textuelle. En partant d'une image matricielle, on identifie les différentes zones graphiques et textuelles et on extrait les « termes » d'indexation de ces deux zones. Au niveau du graphique, on passe par deux phases : d'abord, une phase de prétraitement et d'extraction des primitives. Cela consiste à amincir et vectoriser les composants de la zone graphique. On obtient alors un graphe non orienté dont les nœuds représentent les points extrémités et les arcs les lignes. La Figure 13 illustre un tel graphe. Vient ensuite une phase de génération des « termes » d'indexation. Dans cette phase, les segments de lignes (primitives) sont les entrées d'un système de reconnaissance d'objets qui regroupera ces segments en une jonction de lignes adjacentes, paire de lignes parallèles, etc. Les coefficients de l'approximation de Fourier correspondants à ces jonctions sont pris comme descripteurs.

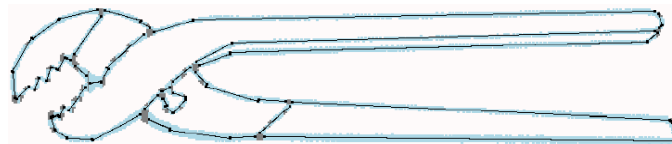


Figure 13 Les segments extraits d'un graphique représentant une pince.

Cette approche permet de retrouver les graphiques en donnant comme requête un graphique scannérisé. La description du document est alors faite non pas par un ensemble de termes textuels mais par un ensemble de termes « graphiques » correspondants aux caractéristiques de ses primitives graphiques. Par conséquent, cette méthode ne permet pas de représenter le contenu sémantique du graphique. Même si le texte présent dans le graphique (annotations, légende, etc.) sert lui aussi pour l'indexation, cela ne permet toujours pas de donner une interprétation complète de son contenu sémantique.

L'approche de Rabitti et Savino [Rabitti,90] [Rabitti,91]

Une analyse de l'image basée sur la donnée du domaine d'application tente de reconnaître les objets contenus dans les images qu'ils soient simples ou complexes (composés d'objets simples et complexes) et d'extraire différentes interprétations, le degré de reconnaissance et la position des objets dans l'image. Le résultat de cette analyse est un ISR-DB (Image Symbolic Representation at Database Level selon le format décrit dans [Rabitti,90]) pour chaque domaine d'application donné. Le langage de requête proposé par [Rabitti,91] permet de restreindre la requête à un ou plusieurs domaines d'application et à exprimer des combinaisons booléennes des conditions sur les objets à trouver (quantificateurs (at least), contraintes de position, etc.) La Figure 14 donne un exemple de requête et une image répondant à cette requête. Les zones en pointillés ont été rajoutées pour mettre en évidence les différentes zones mentionnées dans la requête.

La requête est exécutée par une procédure *SearchImage* dont les étapes sont les suivantes :

- Générer un arbre d'analyse correspondant à la requête Q
- Trouver l'ensemble D' de tous les domaines spécifiés dans Q
- Trouver dans D' tous les domaines D'' contenant les objets contenus dans Q
- Trouver tous les RSI-DB appartenant au domaine D''
- Sélectionner parmi ces RSI-DB, ceux satisfaisant Q

Pour l'accès rapide aux images de la base, une structure d'accès est proposée : il s'agit d'une méthode basée sur la technique de signature qui consiste à extraire et comprimer les propriétés des objets et les stocker dans des fichiers séparés.

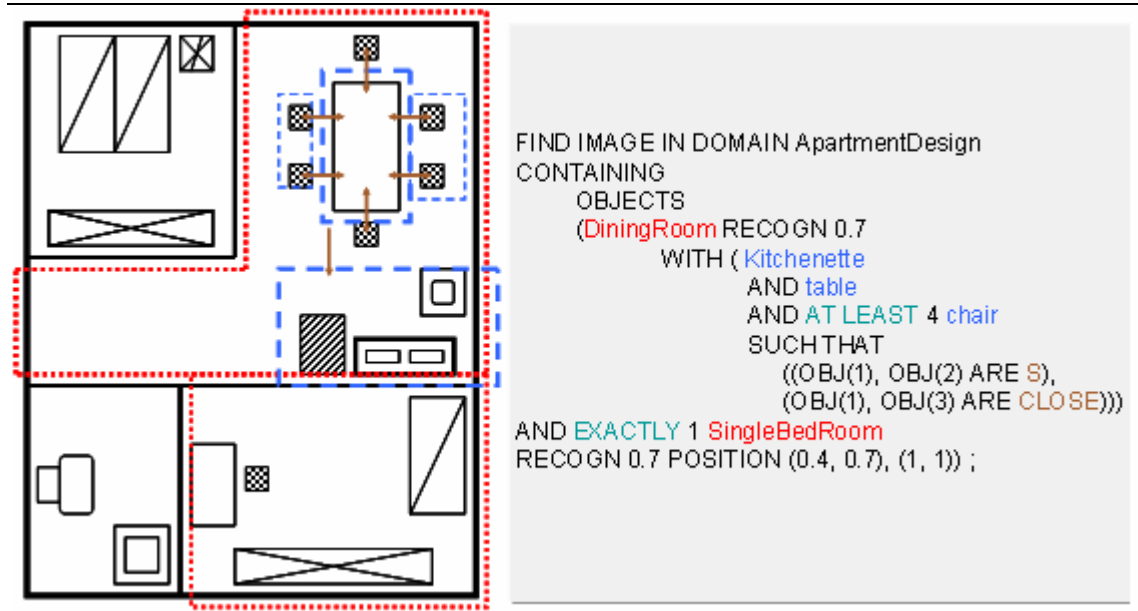


Figure 14 Exemple d'image et de requête

Une caractéristique de cette approche est la prise en compte du contexte de l'application dans le processus d'analyse du graphique. Une seconde caractéristique est la prise en compte d'une structure au niveau des objets contenus dans le graphique et de leurs positions relatives. Ainsi,

un objet complexe du graphique est composé d'autres objets complexes ou d'objets simples, positionnés les uns par rapport aux autres d'une manière précise.

Utilisation des F-Signature [Tabbone,01][Tabbone,03]

Les auteurs proposent une méthode pour indexer des graphiques représentant des plans architecturaux. Il s'agit de reconnaître, dans ces graphiques, des indices, ou symboles, graphiques prédéfinis. Cette reconnaissance est basée sur l'utilisation d'un histogramme de forces : la F-signature. Il s'agit d'une représentation des forces d'attraction exercées entre les parties d'un objet dans différentes directions.

Cet histogramme de forces est intéressant car il est invariable par rapport aux propriétés géométriques concernant la taille, la translation, la symétrie et la rotation.

La Figure 15 montre cinq indices graphiques choisis par les auteurs ainsi que leur F-signature associées.

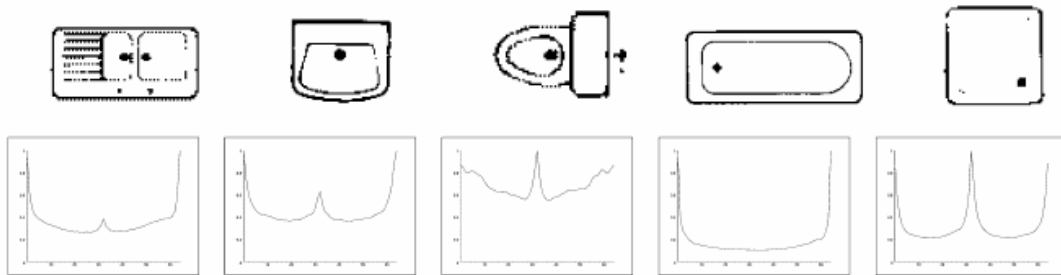


Figure 15 Exemples de symboles graphiques et leur F-signature dans [0, II]

2.3.2. Approches sémantiques

Les approches sémantiques consistent à réduire l'illustration à un ensemble d'objets sémantiques, autrement dit, d'objets identifiés par leur signification dans le monde réel. Ces objets peuvent éventuellement être reliés par des relations sémantiques. Ces approches nécessitent généralement soit l'intervention d'un expert, soit une prise en compte du contexte de ces illustrations.

Nous nous intéressons, dans cette partie, aux images en général, ce qui est tout aussi intéressant que si nous avons considéré les graphiques, car ce qui nous intéresse ici est la sémantique rattachée à l'illustration, et non pas ses propriétés internes. Nous présentons donc certains travaux qui ont utilisé le contexte dans lequel apparaît une image (que ce soit son commentaire ou une information connue par un individu) pour enrichir son indexation.

Le projet MARIE [Rowe,96]

Ce projet a pour but la recherche et l'indexation d'images avec légende dans des bases documentaires. MARIE exploite l'idée qu'il est beaucoup plus facile de comprendre une image grâce à son contexte, et plus particulièrement grâce au texte descriptif que l'on trouve à

proximité. L'objectif de MARIE est de créer automatiquement des liens entre une image et sa légende. Le système est testé sur un corpus militaire ou les photos commentées par une légende contiennent des personnes, des véhicules, des armes...

MARIE comporte quatre étapes. La première consiste à extraire les images et leur légende. Ensuite viennent les étapes d'extraction et d'interprétation des expressions linguistiques, d'une part, et d'extraction et de compréhension de l'image et des objets qu'elle contient d'une autre part. Pour l'extraction et la compréhension dans le texte, il s'agit d'extraire les syntagmes nominaux de la légende de l'image et de leur associer quelques propriétés en utilisant une base de données lexicale, WordNet, associée à un réseau sémantique (par exemple, au terme « missile » est attribué, entre autres, la propriété « est un projectile »). Et pour ce qui est de l'extraction et de la compréhension de l'image, il s'agit de délimiter des zones pouvant représenter des objets. Ces zones sont alors caractérisés par 17 propriétés (texture, couleur, taille, etc.) et classifiées par un réseau de neurones dans des catégories d'objets (avion, homme, etc.) La dernière étape cherche à mettre en relation les extractions des deux médias. Il s'agit alors de rechercher dans le texte des objets présents dans l'image (« the picture shows... », etc.) et de sélectionner dans l'image les objets « importants » (grands, proches du centre, etc.). Les objets du texte et ceux de l'image qui se recoupent le mieux sont alors associés.

Le projet MARIE prend en compte l'existence d'une légende associée à l'image, détecte cette légende et effectue une extraction dans le texte des objets dont il est question dans l'image. D'un autre côté, il détecte les zones dans l'image et en déduit à quel type d'objets ces zones peuvent correspondre. Puis pour terminer, il effectue une mise en relation des objets extraits de l'image et les termes extraits du texte. Ce projet détecte automatiquement les zones dans l'image et utilise le contexte textuel dans lequel elle se trouve (la légende) pour les caractériser. La délimitation des zones ne permet cependant pas de détecter les compositions d'objets. Ainsi, on détectera une « arme » mais pas la « gâchette », le « chargeur » et le « canon ».

Le projet PICTION[Rohini,94] [Rohini,95]

Ce projet développé au CEDAR¹ entre dans le cadre de l'indexation d'images en utilisant leur contexte (légende) lors de leur interprétation. Le système traite des images représentant des personnes ainsi que leur légende afin de mettre en relation les noms des personnes (dans le commentaire) avec les parties de l'image correspondant à leur visage (voir Figure 16). Il procède en trois phases :

- La première phase consiste à extraire, à partir de la légende, les noms des personnes et leurs propriétés en utilisant une sémantique visuelle correspondant à 4 catégories de propriétés : spatiales (géométrie, topologie...), de position (par rapport à l'image), propriétés des objets (visages, couleur des cheveux...) et contextuelles (dans une maison, à l'extérieur...). À partir de ces informations, un graphe de contraintes permettant d'organiser les personnes telles qu'elles devraient être dans l'image est construit.
- La deuxième phase consiste à utiliser un module de détection de visages en s'aidant des informations extraites précédemment (nombre de visages...) et utilise un réseau de neurones pour extraire certaines propriétés à partir des visages détectés (couleur de cheveux, sexe...)

¹ Center for Document Analysis and Recognition

- La dernière phase consiste à mettre en relation les deux extractions (dans la légende et dans l'image) et ce en résolvant le système de contraintes du graphe construit à partir du texte.

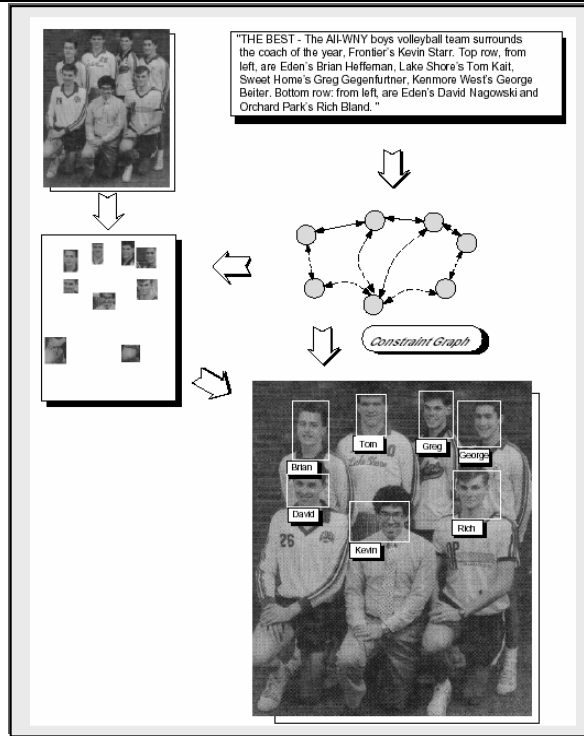


Figure 16 PICTION : correspondance entre les noms et les visages dans l'image

L'association texte-zone dans les cartes géographiques [Malandain,00]

Ce projet entre dans le cadre de la recherche d'information géographique.

Le corpus, géographique, est composé de cartes et de textes. Le but de relier le texte et l'image afin que celle-ci soit prise en compte au même titre que le texte dans les différents traitements que peut subir un document, notamment dans son indexation.

Ce projet propose un système de liens permettant de structurer la relation entre le texte et l'image. Ce système de liens est composé de liens globaux reliant l'image et son commentaire dans leur globalité (unités d'informations globales) et de liens minimaux reliant des parties du commentaire et de l'image (unités d'informations minimales).

La création du lien global est basée sur une recherche dans le texte du paragraphe répondant au mieux à une requête formée d'unités extraites du titre de l'image.

La création d'un lien minimal repose sur une modélisation de l'information géographique qui a pour but de s'abstraire des représentations graphique et textuelle. Au niveau du texte un système d'extraction et d'interprétation des unités d'informations délimite les zones auxquelles elles réfèrent. Et au niveau de la carte, une étude basée sur les variations de teintes dans la carte

permet d'extraire les unités d'informations graphiques. Des liens sont tissés entre les unités d'informations graphiques et textuelles ayant des modélisations qui correspondent.

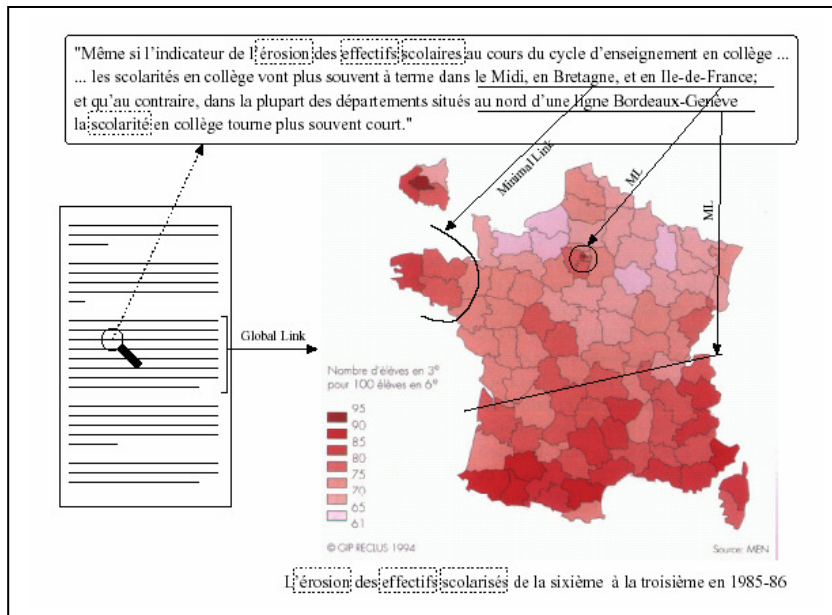


Figure 17 Liens entre des zones de la carte géographique et des fragments de son commentaire

Le modèle EMIR² [Mechkour,95]

Nous avons déjà introduit ce modèle précédemment (langage d'indexation complexe). Pour rappel, il permet de représenter une image à travers un modèle organisé à travers des vues. Dans cet outil, l'extraction d'informations dans l'image est totalement manuelle: l'homme intervient, via une interface, pour délimiter et décrire les objets représentés dans les images.

L'originalité de EMIR² réside essentiellement dans l'intégration d'un ensemble complet de représentations d'images hétérogènes dans une seule structure. Ce modèle combine donc un ensemble de descriptions spécifiques permettant de capter et de combiner un maximum d'informations relatives au contenu de l'image, informations apportées par un intervenant humain.

2.3.3. Bilan des systèmes présentés

L'approche de Lorenz et Monagan ainsi que celle basée sur les F-signatures permettent de retrouver un graphique à partir d'une requête représentée par un graphique scanné. Elles sont toutes deux basées sur un traitement automatique du graphique. Quant à celle de Rabitti, elle permet de retrouver un graphique à partir d'un langage requête ressemblant de prêt au langage SQL. Ces trois méthodes ne prennent en compte que l'aspect spatial du graphique.

Les projets MARIE, PICTION et le système géographique prennent en compte l'existence d'une légende associée à l'image. MARIE et le système géographique se demandent où se situe cette légende en étudiant la structure du document (MARIE) ou en effectuant une recherche du titre du graphique dans le texte commentaire (système géographique) et effectuent une extraction dans le texte des objets dont il est question dans l'image. PICTION nécessite une affectation manuelle de la légende. Quant à EMIR², il n'y a pas de prise en compte d'un texte

associé à l'image mais de son contexte connu par l'indexeur. Une description est faite à travers des vues dans lesquelles des objets pointent vers des objets de l'image.

Les projets MARIE, PICTION et le système géographique détectent automatiquement des zones dans l'image et ensuite, par les propriétés de ces zones, ils en déduisent à quels types d'objets ces zones peuvent correspondre. Par contre EMIR2 et son interface n'ont rien d'automatique : l'homme doit délimiter les objets dans l'image et décrire celle-ci à l'aide d'un système de vues et de relations entre ces objets.

Le Tableau 2 permet de montrer les différences nous paraissant intéressantes entre ces différents systèmes.

	Structuration	Formes	Sémantique	Contexte	automatique	Requête
[Lorenz,95]	+	++	-	-	+	graphique
[Rabitti,90]	++	++	-	-	+	Langage requête
F-signatures	-	+	-	-	+	graphique
MARIE	-	-	+	++	+	texte
PICTION	-	-	+	++	+	texte
[Malandain,00]	+	-	+	++	+/-	-
EMIR ²	++	++	++	+	-	texte

Tableau 2 *Tableau comparatif des méthodes présentées*

3. Conclusion

La recherche d'information technique à usage professionnel constitue un cadre applicatif intéressant permettant, en tenant compte du contexte d'une telle application (les propriétés des documents consultés, tels que la structuration, ainsi que les habitudes de travail et compétences des utilisateurs), d'envisager une approche de recherche d'information dans ces documents via le média graphique. Cette approche est d'autant plus intéressante pour deux raisons :

D'un côté, le graphique est un média qui contient des informations visuelles et auquel est aussi rattachée sémantique qui, même si elle n'apparaît pas dans le graphique en lui-même, peut être déduite de son contexte textuel. Cette richesse du média graphique, tant au niveau de ses données intrinsèques, qu'au niveau de la sémantique qui lui est rattachée, couplée à la nature des utilisateurs (experts du domaine), en font un média nécessitant une représentation complexe afin de rendre compte, le plus fidèlement possible, de son contenu.

D'un autre côté, la recherche est basée sur la description du graphique tel que l'utilisateur pense qu'il devrait être, description qui prend sa source dans la mémoire de l'utilisateur. Ceci nécessite que l'utilisateur puisse préciser, en formulant son besoin, ce qui est important dans le

graphique et ce qui l'est moins (description du graphique idéal), ainsi que ce dont il est certain et ce dont il doute (mémoire du graphique recherché). Ainsi, une requête pourrait être formulée ainsi : «Le graphique contient *peut être* un zoom, une imprimante *peut être* à droite, ainsi qu'un objet en bas à gauche qui *pourrait être* ovale. »

Ce cadre applicatif fait l'objet des parties 3 et 4 de ce manuscrit :

Une étude du corpus et des graphiques qu'il comporte permet de dégager quelle description des graphiques techniques est à retenir (Partie 3-chapitre V.1), et de proposer une représentation possible de ces graphiques (Partie 3-chapitre V.2). À partir de là, nous proposons une description détaillée du modèle de recherche des graphiques (Partie 3-chapitre VI) selon le modèle général que nous proposons dans la partie 2 de ce manuscrit.

La partie 4 se focalise sur l'indexation des graphiques (Partie 4-chapitre VII) ainsi que sur une évaluation de la recherche (Partie 4-chapitre VIII).

**Partie 2. Proposition d'un modèle de recherche
d'information fondé sur des critères
d'obligation et de certitude**

Reprenons ici l'exemple de besoin d'information : «je recherche une image qui représente...une barque...enfin, je crois que c'est une barque, et sur cette barque il y a une femme debout et une autre assise... et peut être qu'il y a aussi un homme. »

Dans ce besoin, l'utilisateur décrit le souvenir qu'il a de l'image (image souvenir) qu'il aimerait retrouver. Dans cette description, trois besoins apparaissent :

(i) L'image souvenir représente des *entités inter reliées*. Il s'agit des entités barque, femme et homme telles que : une première femme est « sur » barque, une seconde femme est « sur » barque et un homme est « sur » barque.

(ii) L'image souvenir représente *deux* femmes. Généralement, dans les systèmes existants, la requête est représentée par une liste d'entités {femme, barque, homme} sans considérer le fait que l'image recherchée doit en contenir deux. Une image contenant une unique femme en plus d'une barque et d'un homme serait alors considérée comme étant pertinente par le système, alors qu'elle ne l'est pas pour l'utilisateur. Dans notre modèle, nous décrivons l'image souvenir (la requête) ainsi que les documents par une même entité autant de fois que nécessaire. L'image souvenir est alors représentée par : une fois le terme «barque», deux fois le terme « femme» et une fois le terme « homme ». Afin de rendre compte d'une telle représentation et d'éviter les conflits entre les mêmes entités (surtout avec la prise en compte des relations), nous proposons de représenter l'image souvenir par quatre *identifiants*: le premier est rattaché au terme «barque», le deuxième au terme « homme » et les deux derniers sont rattachés chacun à une «femme ».

(iii) Dans la description de l'image souvenir, l'utilisateur exprime, à travers les expressions *je crois* et *peut être*, l'idée que ce dont il se rappelle de l'image recherchée est plutôt vague. Ainsi dans l'image souvenir, l'utilisateur n'est pas sûr que l'image qu'il recherche contienne bien un homme, cette entité est alors considérée comme *optionnelle*, par contre il est sûr qu'elle contient une barque et deux femmes considérées, alors, toutes les trois comme étant *obligatoires*. De la même façon, l'utilisateur n'est pas sûr de l'embarcation représentée dans l'image, il pense qu'il s'agit d'une barque mais il n'en est pas convaincu : il peut s'agir de tout autre embarcation, telle qu'un voilier, par exemple. L'entité barque est alors considérée comme étant *incertaine*. Par contre, l'utilisateur étant sûr des entités homme et femme (les deux), celles-ci sont considérées comme étant *certaines*. Ces doutes de l'utilisateur sont reflétés, dans la formulation du besoin que nous proposons dans notre modèle, par l'association, à chaque entité et relation de la requête, de deux critères *obligatoire/optionnel* et *certain/incertain*. Le premier critère reflète ce qui est important que l'image recherchée contienne (une entité ou relation optionnelle peut ou non apparaître dans les documents), et le second reflète la certitude de l'utilisateur quand à la forme sous laquelle apparaît l'entité ou la relation dans ce document (une entité incertaine peut apparaître dans les documents sous une forme similaire ou proche de celle mentionnée dans la requête).

La prise en compte de ces critères dans l'expression de la requête permet d'organiser les documents renvoyés par le système dans des classes : chaque classe contient les documents vérifiant une certaine combinaison des critères, les classes vérifiant le plus grand nombre de critères étant classées en premier. Ainsi, sans considérer les relations entre entité, et en

considérant une classification selon le critère d'obligation, nous obtenons deux classes pour le besoin donné en exemple, la première classe regroupant les documents contenant, en plus des entités barque et deux fois femme, l'entité homme et la deuxième regroupant les documents contenant, uniquement les entités barque et deux fois femme (et ne contenant pas d'entité homme). De la même façon, sans considérer les relations entre entité, et en considérant une classification selon le critère de certitude, nous obtenons deux classes pour le besoin donné en exemple, la première classe regroupant les documents contenant l'embarcation barque en plus des deux fois femme et éventuellement l'homme et la deuxième regroupant les documents contenant, une autre embarcation que barque (voilier par exemple) en plus des deux fois femme et éventuellement l'homme. Cette classification (selon le critère d'obligation ou de certitude), permet à l'utilisateur de mieux voir la relation entre sa requête et les documents renvoyés et de distinguer plus facilement les documents qu'il juge pertinents. Ainsi, en visualisant les documents classés selon le critère de certitude, l'utilisateur peut s'apercevoir que les images qui l'intéressent ne contiennent pas une barque mais plutôt un voilier (ils sont contenus dans la deuxième classe). La distribution, dans les différentes classes, des documents jugés pertinents par l'utilisateur, permet de proposer une nouvelle requête plus précise et proche de l'image qu'il désire retrouver.

Cette partie du manuscrit, est une description du modèle que nous proposons. Dans ce modèle, (i) la description des documents est fondée sur un langage complexe faisant intervenir des entités inter reliées et l'utilisation multiple d'une même entité, (ii) la description de la requête est fondée sur ce même langage enrichi par les deux critères d'obligation/option et de certitude/incertitude, et (iii) la fonction de correspondance respecte les contraintes liées à l'ajout des critères et à l'utilisation multiple d'une même entité. Les documents retenus par cette fonction de correspondance sont rangés dans des classes, selon le critère d'obligation/option ou le critère de certitude/incertitude. Une approche pour la reformulation de la requête, fondée sur les jugements de pertinence de l'utilisateur et les caractéristiques communes des documents retenus (par rapport à la satisfaction des critères), est aussi proposée.

Mais avant d'entrer dans les détails de notre modèle, nous commençons par rappeler les composantes qui entrent en jeu dans la modélisation d'un système de recherche d'information (de façon générale) en proposant une notation pour ces éléments, notation qui sera aussi utilisée dans la description de notre modèle.

Chapitre III

Préliminaires : Définitions et notations

Chapitre IV

Un modèle de RI pour langages complexes :

- Spécificités du modèle
- Le vocabulaire
- L'indexation
- La formulation de la requête
- La correspondance
- Une approche pour la reformulation

Chapitre III. Préliminaires : Définitions et notations

En général, un modèle pour la recherche d'information est défini par :

- un corpus de documents,
- un vocabulaire, sur lequel se base habituellement l'indexation et la formulation de la requête,
- une indexation de documents qui fournit un ensemble de documents indexés,
- une formulation de la requête,
- une correspondance entre la requête et les documents.

Nous reprenons ces cinq parties une à une et proposons de les définir de manière générale.

1. Corpus de documents

Nous disposons d'un corpus de documents D_i .

D_i est un document qui peut être un article de journal, un compte rendu médical, une photographie, une vidéo, etc.

L'ensemble des documents D_i est le corpus noté C . Nous notons \mathcal{N}_C sa cardinalité.

$$C = \{D_i, 1 < i < \mathcal{N}_C\}$$

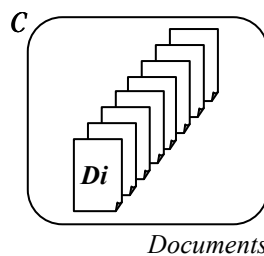


Figure 18 Le corpus C de documents D_i

Exemple

Soit le corpus d'images $C = \{D_1, D_2, D_3, D_4, D_5, D_6, D_7, D_8\}$

$\mathcal{N}_C = 8$.

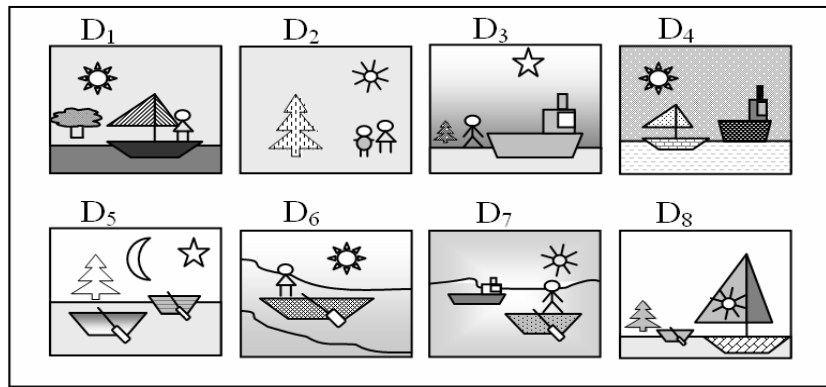


Figure 19 Exemple de corpus de documents C

2. Vocabulaire

Nous disposons d'un vocabulaire, formé d'un ensemble d'unités v , que nous notons \mathcal{V} .

Il peut s'agir d'un ensemble de termes, de concepts, de caractérisation de couleurs (système RGB), de formes géométriques, etc.

Nous disposons d'une relation de proximité, notée *Proche*, entre les unités du vocabulaire \mathcal{V} définie de façon générale¹ :

$$\textit{Proche} : \mathcal{V} \rightarrow \mathcal{P}(\mathcal{V})$$

$$v \rightarrow \textit{Proche}(v)$$

Cette fonction peut se présenter sous différentes formes. On peut citer la relation de synonymie comme celle utilisée dans WordNet², la relation spécifique/générique utilisée dans les thésaurus, etc.

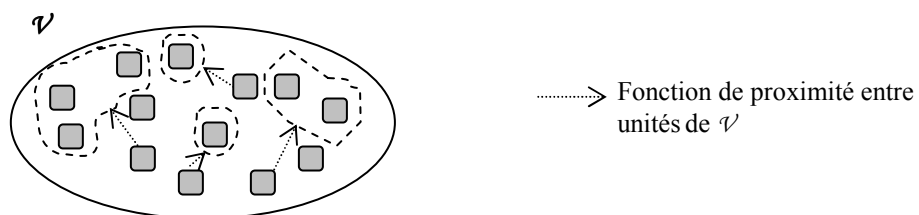


Figure 20 Le vocabulaire \mathcal{V}

¹ Soit un ensemble X , $\mathcal{P}(X)$ est l'ensemble des parties de X

² <http://wordnet.princeton.edu>

3. Indexation des documents

Le vocabulaire d'indexation

Le vocabulaire d'indexation est un ensemble qui sert de base pour l'indexation des documents du corpus C . Il peut être égal au vocabulaire \mathcal{V} , comme c'est le cas pour une indexation simple à base de mots clés, ou avoir une relation plus complexe avec le vocabulaire.

Nous notons \mathcal{V}_D le vocabulaire d'indexation et $\mathcal{N}_{\mathcal{V}_D}$ sa cardinalité.

Nous appelons les éléments du vocabulaire d'indexation unités d'indexation et nous les notons \mathbf{u}_{Dj} .

$$\mathcal{V}_D = \{\mathbf{u}_{D1}, \mathbf{u}_{D2}, \mathbf{u}_{D3}, \dots, \mathbf{u}_{Dj}, \dots, \mathbf{u}_{D\mathcal{N}_{\mathcal{V}_D}}\}.$$

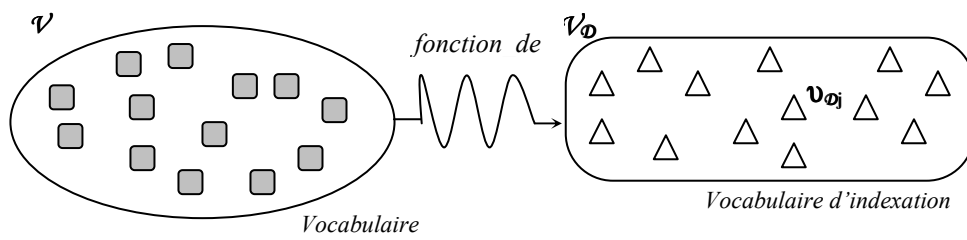


Figure 21 Le vocabulaire d'indexation \mathcal{V}_D rattaché au vocabulaire \mathcal{V}

Le corpus de documents indexés

A chaque document Di du corpus C correspond un document Dli qui est une représentation du contenu du document Di .

Il s'agit d'un sous-ensemble du vocabulaire d'indexation \mathcal{V}_D .

$$Dli \in \mathcal{P}(\mathcal{V}_D)$$

Nous notons \mathcal{N}_{Di} la cardinalité d'un document Dli et $\mathbf{u}_{Di,j}$ sa $j^{\text{ème}}$ unité d'indexation.

Dli s'écrit alors :

$$Dli = \{\mathbf{u}_{Di,1}, \mathbf{u}_{Di,2}, \dots, \mathbf{u}_{Di,j}, \dots, \mathbf{u}_{Di,\mathcal{N}_{Di}}\}.$$

L'ensemble des documents indexés Dli constitue le corpus indexé CI .

CI est donc de cardinalité \mathcal{N}_C .

$$CI = \{Dli, 1 < i < \mathcal{N}_C\}$$

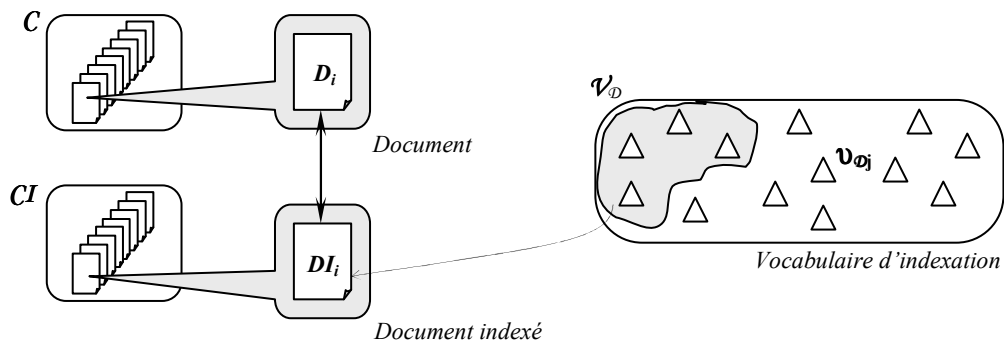


Figure 22 Le corpus indexé CI

4. Formulation de la requête

Le vocabulaire d'interrogation

Le vocabulaire d'interrogation sert à la formulation de la requête.

Il peut être égal au vocabulaire \mathcal{V} , comme c'est le cas pour une formulation simple à base de mots clés, ou être égal à un ensemble de couples formés chacun par une unité du vocabulaire \mathcal{V} et sa pondération reflétant l'importance de cette unité pour l'utilisateur. Il peut aussi être égal à un ensemble de couples représentant chacun une unité du vocabulaire \mathcal{V} enrichie par sa caractérisation en terme d'obligation/option comme c'est le cas dans le modèle proposé par [Denos,97a] [Denos,97b]...

Dans un cas général, nous définissons le vocabulaire d'interrogation comme étant le résultat d'une combinaison du vocabulaire \mathcal{V} et d'un ensemble de critères \mathcal{C}_α permettant à l'utilisateur de préciser ses besoins.

Nous notons \mathcal{V}_Q cet ensemble et v_{ϕ_j} ses éléments que nous appelons unités d'interrogation.

Exemples

- $\mathcal{C}_\alpha = \emptyset$ et $\mathcal{V}_Q = \mathcal{V}$ dans le cas d'une formulation simple à base de mots clés,
- $\mathcal{C}_\alpha = \{\text{obligatoire}, \text{optionnel}\}$ et $\mathcal{V}_Q = \mathcal{V} \times \mathcal{C}_\alpha$ dans le modèle de [Denos97a].

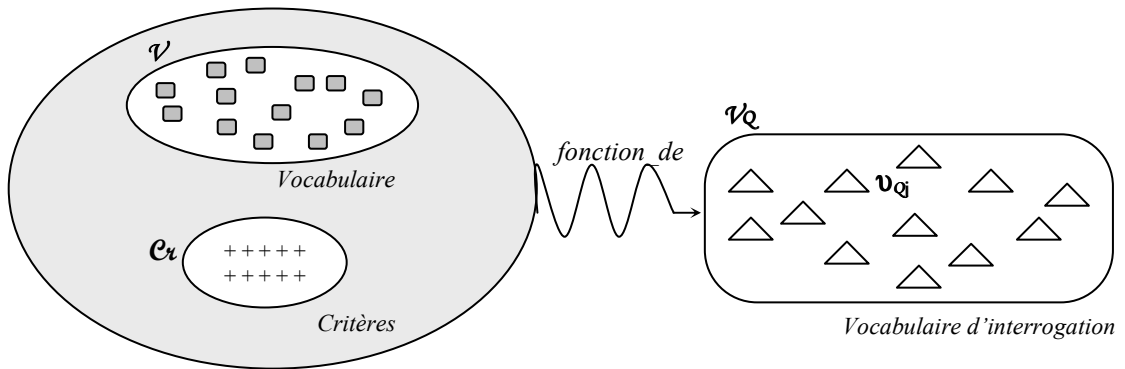


Figure 23 Le vocabulaire d'interrogation \mathcal{V}_Q rattaché au vocabulaire \mathcal{V}

La requête

Une requête q est un sous-ensemble du vocabulaire d'interrogation \mathcal{V}_Q . Elle est définie par \mathcal{N}_Q unités d'interrogation qu'on notera $u_{q,k}$.

$$q \in \mathcal{P}(\mathcal{V}_Q)$$

Une requête q s'écrit :

$$q = \{u_{q,1}, u_{q,2}, \dots, u_{q,k}, \dots, u_{q,\mathcal{N}_Q}\}.$$

Nous noterons \mathcal{CQ} l'ensemble des requêtes possibles.

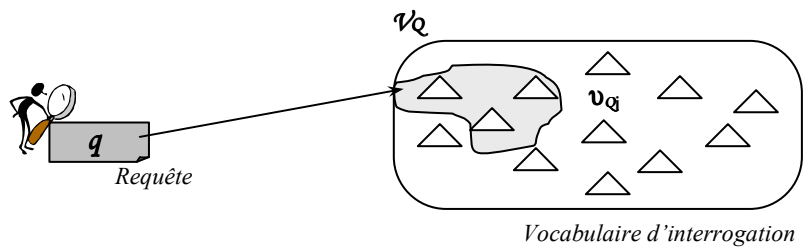


Figure 24 La requête q

5. Correspondance entre la requête et les documents

La fonction de correspondance

Nous définissons la fonction *Corresp* qui permet de vérifier si un document indexé correspond à une requête :

$$\begin{aligned} \text{Corresp} : CQ \times CI &\rightarrow \text{booléen} \\ (q, DIi) &\rightarrow \text{vrai si } DIi \text{ correspond à } q \text{ et faux sinon.} \end{aligned}$$

Ainsi, un document Di est pertinent pour q , si et seulement si $\text{Corresp}(q, DIi)$ est *vrai*.

Remarque

Nous nous limitons, dans cette description du modèle, à une correspondance booléenne (vrai ou faux). Une fonction \mathcal{S} , qui se base sur la fonction *Corresp*, peut être utilisée afin de pondérer les documents retournés.

Exemples

- Dans le cas du modèle booléen, $\text{Corresp}(q, DIi)$ est *vrai* si pour chaque unité d'interrogation $v_{q,k}$ de q , il existe une unité d'indexation $v_{Di,j}$ de DIi telle que $v_{q,k} = v_{Di,j}$.
- Dans le cas du modèle proposé par [Deno,97a] [Deno,97b] où la requête $q \in \mathcal{P}(\mathcal{V} \setminus \{\text{obligatoire, optionnel}\})$, $\text{Corresp}(q, DIi)$ est *vrai* si pour chaque unité d'interrogation $v_{q,k} = (v_j, \text{obligatoire})$ de q , il existe une unité d'indexation $v_{Di,j}$ de DIi telle que $v_j = v_{Di,j}$.

L'ensemble des documents pertinents

Nous définissons la fonction \mathcal{D} qui associe à une requête l'ensemble des documents qui y répondent :

$$\begin{aligned} \mathcal{D} : CQ &\rightarrow \mathcal{P}(C) \\ q &\rightarrow \mathcal{D}(q) = \{Di \in C, \text{Corresp}(q, DIi) = \text{vrai}\} \end{aligned}$$

6. Récapitulatif

Nous disposons d'un corpus de documents C et d'un vocabulaire \mathcal{V} .

Un vocabulaire d'indexation \mathcal{V}_D est rattaché au vocabulaire \mathcal{V} et un vocabulaire d'interrogation \mathcal{V}_Q est rattaché au vocabulaire \mathcal{V} augmenté des critères de l'ensemble \mathcal{C}_α .

A chaque document Di de C correspond un document indexé DIi qui est un sous-ensemble du vocabulaire d'indexation \mathcal{V}_D et une requête q est un sous-ensemble du vocabulaire d'interrogation \mathcal{V}_Q .

Nous disposons enfin d'une fonction de correspondance *Corresp* entre un document indexé et une requête.

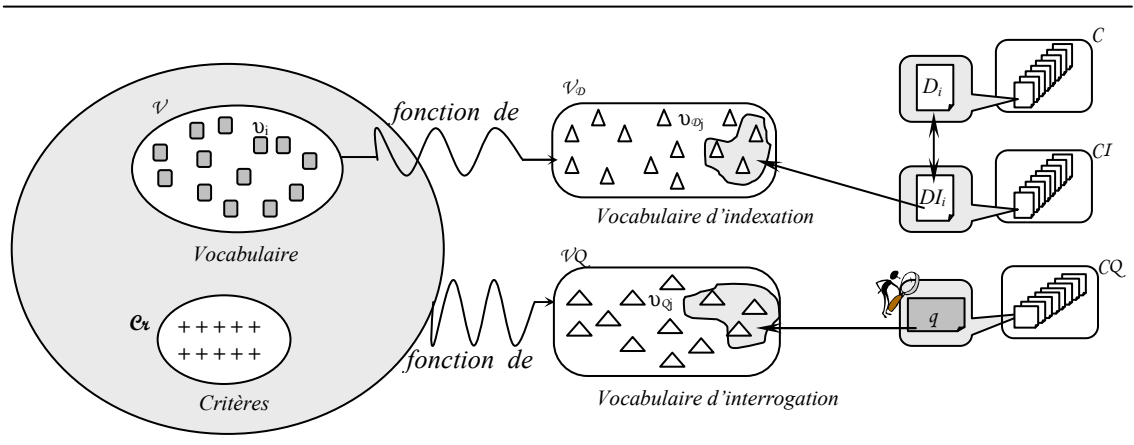


Figure 25 Schéma récapitulatif

7. Conclusion

Dans ce chapitre, nous avons défini de façon générale les différentes parties qui entrent en jeu dans un modèle de recherche d'information. Dans le chapitre suivant, en suivant ce même schéma, nous définissons le modèle que nous proposons, modèle qui permet la description d'un document en terme d'entités reliées entre elles par des relations, et qui permet la formulation d'une requête en termes d'entités inter-reliées et augmentées chacune de critères reflétant l'obligation ou non de l'apparition d'une entité (respectivement relation) dans les documents, et la certitude de l'utilisateur quand à cette entité (respectivement relation).

Chapitre IV. Un modèle de recherche d'information pour des langages complexes

Ce chapitre fournit une description (à partir des notations introduites dans le chapitre précédent) de notre modèle de recherche d'information pour des langages à base d'entités structurées et fondé sur des critères d'obligation/option et de certitude/incertitude.

Dans notre modèle :

- le vocabulaire est constitué d'un ensemble d'*entités* (que ce soit des termes, des concepts, des objets image, etc.) et d'un ensemble de *relations*.
- le contenu d'un document est décrit par ces entités mises en relations les unes avec les autres en permettant l'*utilisation multiple* d'une même entité dans un même document.
- la requête est représentée par ces entités et leurs interrelations augmentées toutes deux de *critères* d'obligation/option et certitude/incertitude.
- La correspondance entre la requête et les documents est réalisée en respectant les contraintes liées à l'utilisation multiple des entités ainsi qu'aux critères associés aux éléments de la requête.

La formulation proposée du besoin, permet de classer les documents retrouvés en fonction des critères qu'ils ont en commun. Cette classification nous permet de proposer une approche de reformulation de la requête, en fonction des jugements de pertinence de l'utilisateur et des caractéristiques des documents marqués pertinents.

Dans cette partie, nous justifions tout d'abord (§1) les spécificités de notre modèle (les relations, l'utilisation multiple d'entités et les critères dans l'interrogation). Nous décrivons par la suite les grandes lignes de notre modèle dans le § 2 que nous présentons en détails dans le § -. Le § 4 propose une approche pour la reformulation de la requête.

1. Spécificités du modèle

Utilisation des relations entre entités

Il est généralement admis que l'approche classique décrivant les documents par des représentations « plates » à base de mots clés est insuffisante à mesure que les corpus subissent une augmentation en information, aussi bien en quantité qu'en richesse. Ces approches ne pouvant pas engendrer des systèmes orientés vers la précision des réponses, il est nécessaire de s'orienter vers des structures complexes impliquant entités et relations.

Les relations entre entités peuvent se présenter sous différentes formes ; il peut s'agir de relations spatiales décrivant les positions relatives des objets d'une image comme celles utilisées dans le modèle EMIR² [Mechkour,95], il peut s'agir aussi de relations permettant d'apporter une précision sur une entité ambiguë du document, comme la relation « est_un », ou de relations sémantiques comme la relation « parle_à », etc.

Dans tous les cas, l'utilisation des relations entre entités multiplie les possibilités d'expression, aussi bien au niveau du document, qu'au niveau de la requête.

Dans notre modèle les relations entre les entités sont toutes des relations binaires et orientées.

Utilisation multiple d'entités

Généralement, dans les approches classiques, un élément contenu dans un document plusieurs fois n'est représenté qu'une seule fois dans le document indexé. Ainsi, une première image représentant un bateau et une seconde en représentant deux, seront toutes deux indexées par l'entité « bateau ». Une pondération peut être ajoutée à ces entités afin d'en privilégier certaines par rapport à d'autres, ainsi on pourra privilégier l'entité « bateau » car elle apparaît deux fois dans la requête : l'existence multiple d'une même entité, dans ces systèmes, n'est prise en compte qu'au niveau de la pondération. Pourtant, dans certains contextes, il est plus pertinent d'utiliser l'entité autant de fois que nécessaire pour une meilleure expressivité du langage d'indexation (de même pour la requête).

Afin de rendre compte d'une telle représentation et d'éviter les conflits entre les mêmes entités, nous proposons d'introduire un ensemble intermédiaire entre le vocabulaire d'une part et les vocabulaires d'indexation et d'interrogation d'autre part : Il s'agit d'un ensemble d'*identifiants* rattachés aux entités décrivant les documents ou la requête. Ainsi, notre image sera représentée par trois identifiants δ_1 , δ_2 et δ_3 : δ_1 étant rattaché à la première « barque », δ_2 à la deuxième et δ_3 à « homme ». Qui plus est, aucune ambiguïté n'apparaîtra dans la description des relations et on obtiendra les relations: (δ_3 , δ_1 , sur), (δ_1 , δ_2 , à_droite).

Utilisation des critères d'obligation/option et certitude/incertitude

Nous nous intéressons ici à une formulation particulière du besoin : l'utilisateur connaît les documents (ils les a donc en mémoire), et il formule son besoin en décrivant ce qui représenterait le document idéal à ses yeux, en fonction du souvenir qu'il en a. Ceci implique, d'une part, qu'il est important ou moins important que certains éléments soient contenus dans les documents recherchés, et d'autre part, que l'utilisateur a des doutes quant au contenu des documents qu'il recherche (est-ce bien exactement cet élément qui est contenu dans le document ?).

Dans le premier cas, l'utilisateur peut imposer qu'une entité (respectivement une relation entre deux entités) apparaisse dans les documents qui l'intéressent, ou lever cette obligation s'il juge que l'entité (respectivement, la relation) n'est pas très importante.

Dans le deuxième cas, l'utilisateur peut avoir une certitude quand à l'entité (respectivement la relation entre deux entités) qu'il désire voir apparaître dans les documents qu'il recherche ou avoir un doute sur cette entité (respectivement, cette relation), auquel cas, les documents contenant une entité (respectivement, une relation) « proche » (selon une fonction de proximité rattachée au vocabulaire) de celle qu'il a mentionnée pourraient aussi l'intéresser.

Afin de permettre l'expression de ces critères, chaque entité et relation de la requête est augmentée par deux critères, l'un exprimant l'obligation ou non de son apparition dans les documents et l'autre exprimant la certitude de l'utilisateur quant à sa cette entité ou relation.

Exemple

Un médecin veut retrouver certains comptes rendus :

« Ils parlent de « tumeur » située « au niveau du » « poumon ». On y mentionne aussi le « stade » de la maladie, qui est peut être le « 3B ». Et on y parle peut être du « traitement » qui est, je crois, les « Interférons ».

Dans cette requête, nous pouvons distinguer quatre entités et trois relations :

- « tumeur » et « poumon » sont deux entités obligatoires et certaines,
- « 3B » est une entité obligatoire et incertaine,
- « Interféron » est une entité optionnelle et incertaine,
- « Être situé(e) au niveau de » entre « tumeur » et « poumon » et « Être au stade » entre « tumeur » et « 3B » sont deux relations obligatoires et certaines,
- « Avoir pour traitement » entre « tumeur » et « Interférons » est une relation optionnelle et certaine.

2. Vue d'ensemble du modèle

Nous désirons décrire un document D_i et une requête q par des entités reliées entre elles par des relations, tout en permettant l'utilisation multiple des entités.

Dans la requête, nous désirons que chaque entité et chaque relation soit marquée par deux critères, l'un indiquant si elle est obligatoire ou optionnelle et l'autre indiquant si elle est certaine ou incertaine.

La correspondance doit prendre en compte les critères sur les entités et les relations de la requête sans oublier qu'une même entité peut être utilisée plusieurs fois dans le document indexé ou dans la requête, ce qui pose une contrainte supplémentaire à respecter dans la fonction de correspondance.

2.1. Vocabulaire

Nous disposons de deux ensembles :

- un ensemble \mathcal{T} contenant $\mathcal{N}_{\mathcal{T}}$ entités τ_k ,
- un ensemble \mathcal{R} contenant $\mathcal{N}_{\mathcal{R}}$ Relations ρ_j .

Ces deux ensembles \mathcal{T} et \mathcal{R} constituent le Vocabulaire \mathcal{V} .

Il existe deux éléments neutres τ_0 et ρ_0 ne correspondent pas à des éléments existants dans le document. Ils modélisent une entité qui n'est en relation avec aucune autre.

A chacun des ensembles \mathcal{T} et \mathcal{R} est associée une relation de proximité entre les éléments :

- $Proche_{\mathcal{T}}$ entre certaines entités τ_k ,
- $Proche_{\mathcal{R}}$ entre certaines relations ρ_j .

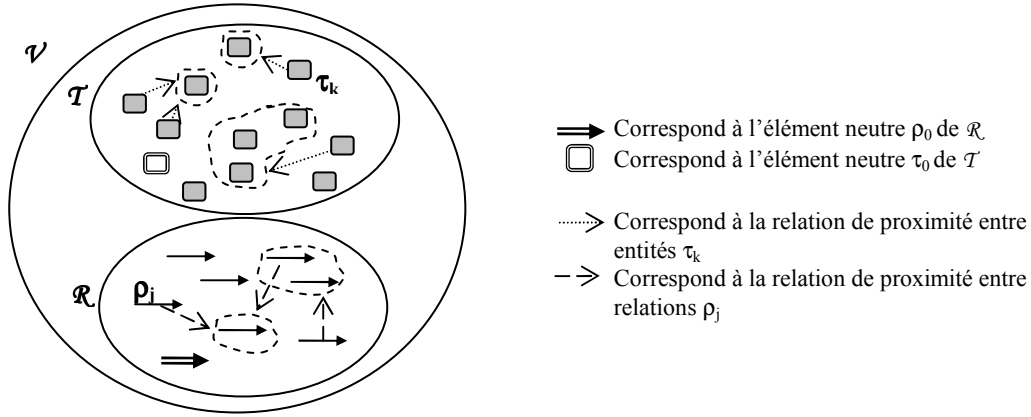


Figure 26 Le vocabulaire \mathcal{V} est formé par les ensembles d'entités \mathcal{T} et de relation \mathcal{R}

2.2. Vocabulaire d'indexation

Nous définissons un ensemble d'identifiants d'entités. Nous notons Δ cet ensemble et δ_j ses éléments.

Chaque élément de Δ est associé à un élément de \mathcal{T} par une fonction notée $F_{\delta\tau}$.

$F_{\delta\tau}$ permet ainsi l'utilisation multiple d'entités dans les documents.

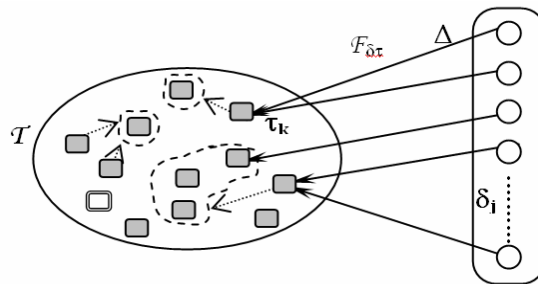


Figure 27 L'ensemble Δ des identifiants rattachés aux entités de \mathcal{T}

Nous définissons le vocabulaire d'indexation \mathcal{V}_d comme étant un ensemble de triplets définis chacun par deux identifiants δ_{j_1} et δ_{j_2} de Δ et une relation ρ_j de \mathcal{R} .

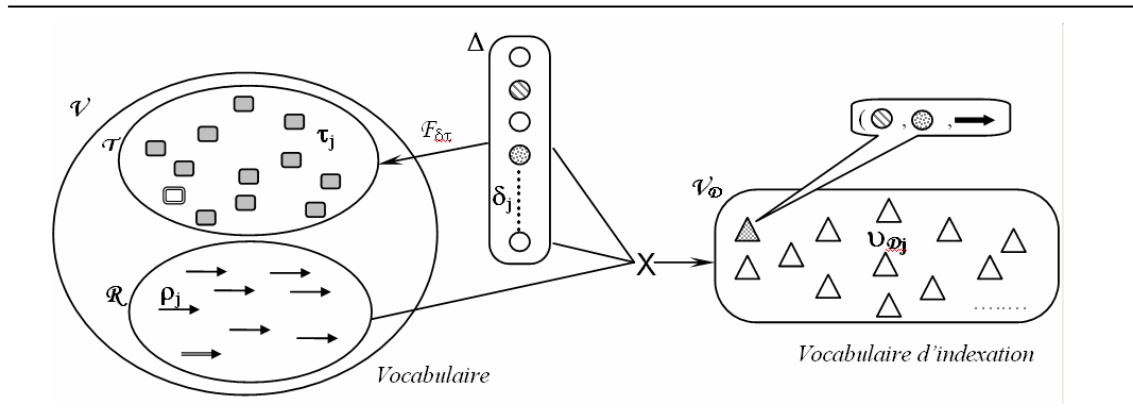


Figure 28 Le vocabulaire d'indexation \mathcal{V}_D

2.3. Vocabulaire d'interrogation

Nous associons aux entités et aux relations des critères d'obligation/option et de certitude/incertitude. Ainsi une entité, ou une relation, peut être obligatoire ou optionnelle. Elle peut également être certaine ou incertaine.

Définitions des critères (selon le Robert 97)

- *Obligatoire* : Qui a la force d'une obligation. *Fam.* Inévitable nécessaire.
- *Optionnel* : Qui donne lieu à un choix, qu'on peut acquérir facultativement.
- *Certain* : Qui est effectif, sans aucun doute : assuré, incontestable, indubitable.
- *Incertain* : Qui n'est pas connu avec certitude. Dont la forme, la nature n'est pas nette : confus, imprécis, vague.

Sens des critères dans la requête

- Nous disons qu'une unité de la requête est *obligatoire* lorsqu'elle doit obligatoirement être mentionnée dans les documents renvoyés par le système. Par opposition, nous disons qu'une unité est *optionnelle* lorsqu'elle peut ou non être mentionnée dans les documents renvoyés par le système.
- Nous disons qu'une unité de la requête est *certaine* lorsque, si elle est mentionnée dans les documents renvoyés par le système, elle l'est exactement sous la forme donnée dans la requête. Par opposition, nous disons qu'une unité de la requête est *incertaine* lorsque, si elle est mentionnée dans les documents renvoyés par le système, elle l'est sous la forme donnée dans la requête ou sous une forme qui lui est proche (selon la relation de proximité associée au vocabulaire d'indexation).

Par exemple, si l'on considère la relation de généralité/spécificité pour définir la relation *Proche* :

- la requête « je veux retrouver des documents contenant le mot *garçon*, mais je ne suis pas sûr de ce terme » renverra aussi bien les documents contenant *garçon* que les documents contenant *enfant* sachant que $enfant \in Proche(garçon)$.
- la requête « je veux retrouver des documents contenant le mot *garçon*, et je suis sûr de ce terme » ne renverra que les documents contenant exactement le mot *garçon*.

Afin de prendre en compte ces critères, nous définissons deux ensembles \mathcal{O} et \mathcal{J} exprimant respectivement le critère d'obligation/option et le critère de certitude/incertitude

Comme nous désirons permettre l'utilisation multiple des entités lors de la formulation d'un besoin, nous associons les critères aux identifiants des entités et non pas aux entités elles-mêmes.

Nous définissons deux ensembles intermédiaires (voir Figure 29):

- Un ensemble d'identifiants d'interrogation noté Δ_Q . Chaque identifiant d'interrogation, noté δ_{Q_i} est un triplet défini par un identifiant de l'ensemble Δ auquel sont associés un élément de l'ensemble \mathcal{O} et un élément de l'ensemble \mathcal{J}
- Un ensemble de relations d'interrogation noté \mathcal{R}_Q . Chaque relation d'interrogation, noté ρ_{Q_i} est un triplet défini par une relation de l'ensemble \mathcal{R} à laquelle sont associés un élément de l'ensemble \mathcal{O} et un élément de l'ensemble \mathcal{J} .

Ainsi, nous considérons le vocabulaire d'interrogation \mathcal{V}_Q comme un ensemble de triplets : Chaque triplet est formé de deux identifiants d'interrogation et d'une relation d'interrogation (Voir Figure 30).

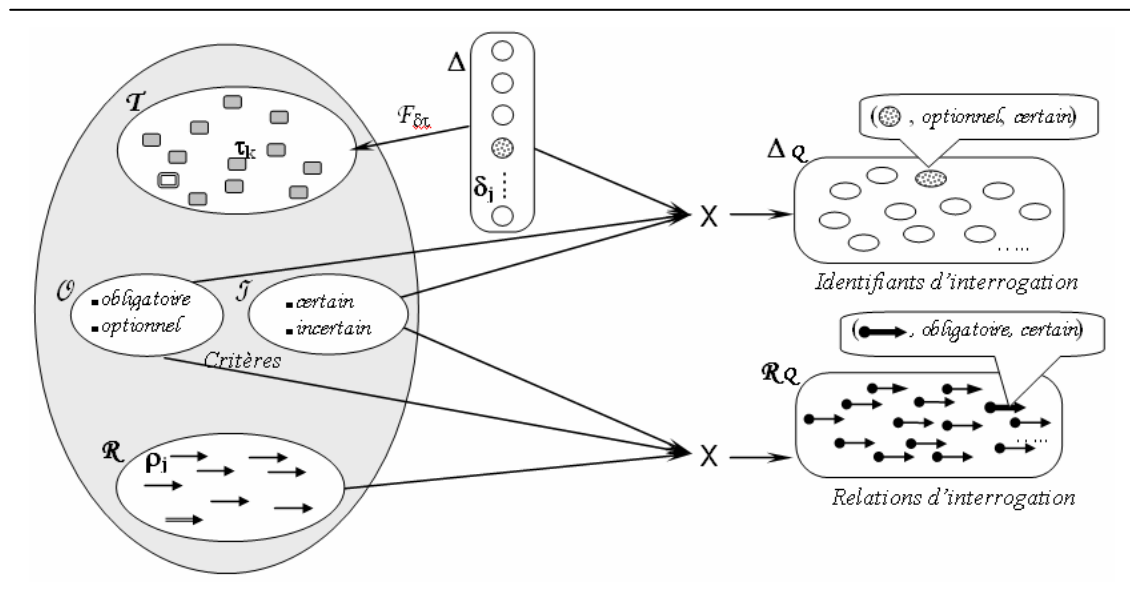


Figure 29 Les ensembles d'identifiants d'interrogations Δ_Q et de relations d'interrogation \mathcal{R}_Q

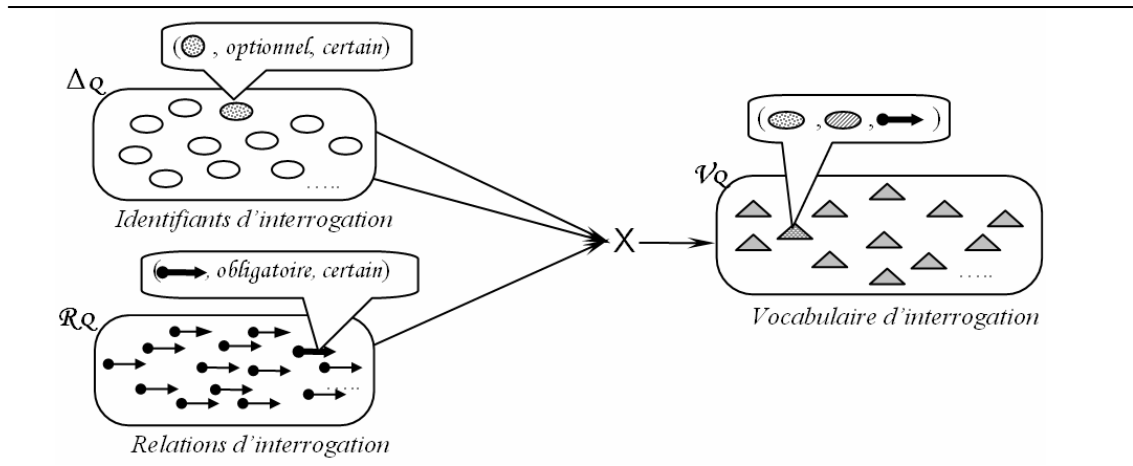


Figure 30 Le vocabulaire d'interrogations \mathcal{V}_Q

2.4. Fonctions utiles

Des fonctions $F_{\nu\delta_1}$, $F_{\nu\delta_2}$, $F_{\nu\rho}$, F_δ , F_ρ , F_{ob} , F_{cer} et $F_{\delta\tau}$ permettent d'associer entre eux les éléments des différents ensembles définissant notre modèle. La **Figure 31** schématise ces associations:

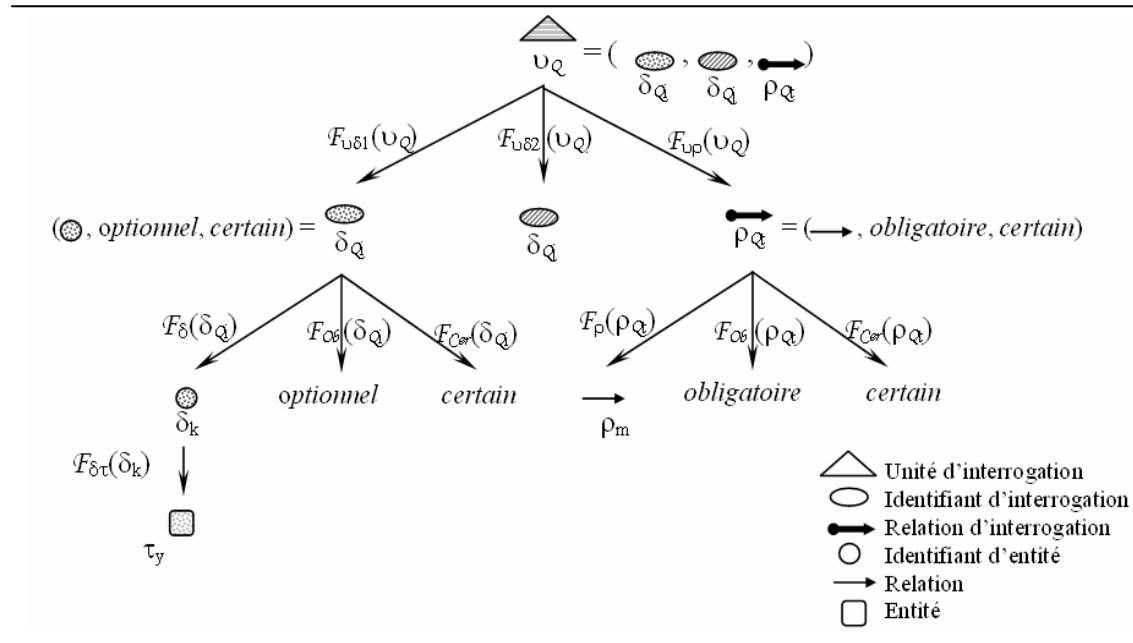


Figure 31 Les fonctions : de l'unité d'interrogation à l'entité et la relation

2.5. La fonction de correspondance

La fonction de correspondance doit respecter deux contraintes.

D'une part, elle doit respecter les contraintes sur les critères de la requête :

- une entité ou relation obligatoire doit être contenue dans les documents retrouvés,
- une entité ou relation optionnelle peut ou non être contenue dans ces documents,
- une entité ou relation certaine doit être contenue telle qu'elle dans ces documents,
- une entité ou relation incertaine doit être contenue telle qu'elle ou sous une forme « proche » (selon la fonction de proximité définie dans le vocabulaire) dans ces documents.

D'autre part, elle doit respecter les contraintes liées à l'utilisation multiple d'une entité. Elle doit donc associer deux à deux, d'un côté les entités de la requête ayant un correspondant dans le document indexé (selon les critères qui leurs sont associées), et d'un autre côté ceux du document indexé qui lui correspondent. Nous basons donc notre fonction de correspondance sur une injection (détaillée dans la suite de ce chapitre).

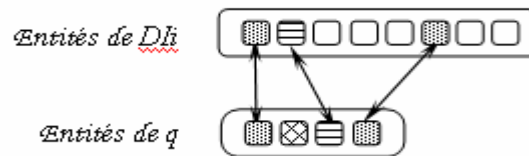


Figure 32 Exemple d'association deux à deux des entités de D_i et de q

3. Présentation détaillée du modèle

Dans cette partie, nous reprenons avec détail les différents ensembles et fonctions introduits.

3.1. Vocabulaire

Le vocabulaire \mathcal{V} est formé par l'ensemble des entités et l'ensemble des relations. Un ensemble d'identifiants rattaché à l'ensemble des entités est introduit afin de permettre l'utilisation multiple d'une même entité dans l'index du document et la requête.

3.1.1. Les entités

Nous désignons par entité tout élément pouvant être considéré comme unité d'indexation d'un document. Habituellement, on parle de terme d'indexation. Nous généralisons cette notion à de nouveaux types d'objets, comme les graphiques.

Nous définissons en plus l'entité vide ou neutre que nous notons τ_0 (son utilité apparaîtra lorsque nous définirons les unités d'indexation et d'interrogation).

Exemple

$$\tau_1 = \text{bateau}, \tau_2 = \text{soleil}, \tau_3 = \text{mer}, \tau_4 = \text{lac} \dots$$

Nous notons \mathcal{T} l'ensemble des entités d'indexation τ_i et $\mathcal{N}_{\mathcal{T}}$ sa cardinalité.

$$\mathcal{T} = \{\tau_1, \tau_2, \tau_3, \dots, \tau_i, \dots, \tau_{\mathcal{N}_{\mathcal{T}}}\}.$$

Exemple

Soit l'ensemble:

$$\mathcal{T} = \{\tau_0, \text{🧑}, \text{🧑}, \text{🧑}, \text{🚤}, \text{🚤}, \text{🚤}, \text{🌳}, \text{🌲}, \text{☀️}, \text{☀️}, \text{★}, \text{lac}, \text{mer}, \text{rivière}, \text{pêcheur}, \text{promeneur}, \text{marin}\}$$

$$\mathcal{N}_{\mathcal{T}} = 18.$$

Nous supposons de plus qu'existe une *relation de proximité* entre entités de l'ensemble \mathcal{T} définie de façon générale :

$$\begin{aligned} \text{Proche}_{\mathcal{T}}: \mathcal{T} &\rightarrow \mathcal{P}(\mathcal{T}) \\ \tau &\rightarrow \text{Proche}_{\mathcal{T}}(\tau) \text{ avec } \text{Proche}_{\mathcal{T}}(\tau_0) = \mathcal{T} \end{aligned}$$

Cette fonction est utile pour gérer les entités incertaines dans la fonction de correspondance.

Exemple

Soit la fonction $\text{Proche}_{\mathcal{T}}$, rattachée à l'ensemble \mathcal{T} donné en exemple, telle que :

$$\begin{aligned} \text{Proche}_{\mathcal{T}}(\text{🧑}) &= \{\text{🧑}, \text{🧑}, \text{🧑}\}, \\ \text{Proche}_{\mathcal{T}}(\text{🚤}) &= \{\text{🚤}, \text{🚤}, \text{🚤}\}, \\ \text{Proche}_{\mathcal{T}}(\text{🌳}) &= \{\text{🌳}, \text{🌳}\}, \\ \text{Proche}_{\mathcal{T}}(\text{☀️}) &= \{\text{☀️}, \text{☀️}\}, \\ \text{Proche}_{\mathcal{T}}(\text{lac}) &= \{\text{lac}, \text{mer}, \text{rivière}\}, \\ \text{Proche}_{\mathcal{T}}(\text{pêcheur}) &= \{\text{pêcheur}, \text{marin}\}, \\ \text{Proche}_{\mathcal{T}}(\text{★}) &= \{\text{★}\}, \\ \text{Proche}_{\mathcal{T}}(\tau_0) &= \mathcal{T}. \end{aligned}$$

3.1.2. Les relations

Nous désignons par relation, tout lien pouvant exister entre deux entités d'indexation. Il s'agit d'une relation orientée.

Nous définissons aussi une relation vide ou neutre que nous notons ρ_0 . (Son utilité apparaîtra lorsque nous définirons les unités d'indexation et d'interrogation).

Exemple

$$\rho_1 = \text{est_un}, \rho_2 = \text{au_dessus_de}, \rho_3 = \text{a_droite_de} \dots$$

Nous appelons \mathcal{R} l'ensemble des relations ρ_i pouvant exister entre les entités décrivant les documents du corpus.

Nous notons $\mathcal{N}_{\mathcal{R}}$ sa cardinalité.

$$\mathcal{R} = \{\rho_1, \rho_2, \rho_3, \dots, \rho_i, \dots, \rho_{\mathcal{N}_{\mathcal{R}}}\}.$$

Exemple

$$\mathcal{R} = \{\rho_0, \text{proche_de}, \text{a_droite_de}, \text{au_dessus_de}, \text{en_dessous_de}, \text{sur}, \text{sous}, \text{est_un}, \text{decore}\}$$

$$\mathcal{N}_{\mathcal{R}} = 9.$$

Nous supposons de plus qu'il existe une relation de proximité entre relations de l'ensemble \mathcal{R} définie de façon générale :

$$\begin{aligned} \text{Proche}_{\mathcal{R}}: \mathcal{R} &\rightarrow \mathcal{P}(\mathcal{R}) \\ \rho &\rightarrow \text{Proche}_{\mathcal{R}}(\rho) \text{ avec } \text{Proche}_{\mathcal{R}}(\rho_0) = \mathcal{R} \end{aligned}$$

Cette fonction est utile pour gérer les relations incertaines dans la fonction de correspondance.

Exemple

Soit la fonction $\text{Proche}_{\mathcal{R}}$ rattachée à l'ensemble \mathcal{R} donné en exemple:

$$\begin{aligned} \text{Proche}_{\mathcal{R}}(\text{au_dessus_de}) &= \{\text{au_dessus_de}, \text{sur}, \text{proche_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{en_dessous_de}) &= \{\text{en_dessous_de}, \text{sous}, \text{proche_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{sur}) &= \{\text{sur}, \text{au_dessus_de}, \text{proche_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{sous}) &= \{\text{sous}, \text{en_dessous_de}, \text{proche_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{a_droite_de}) &= \{\text{a_droite_de}, \text{proche_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{proche_de}) &= \{\text{proche_de}, \text{en_dessous_de}, \text{sur}, \text{sous}, \text{au_dessus_de}, \text{a_droite_de}\}, \\ \text{Proche}_{\mathcal{R}}(\text{est_un}) &= \{\text{est_un}\}, \\ \text{Proche}_{\mathcal{R}}(\text{décore}) &= \{\text{décore}\}, \\ \text{Proche}_{\mathcal{R}}(\rho_0) &= \mathcal{R} \end{aligned}$$

3.1.3. Les identifiants d'entités

Afin de permettre l'utilisation multiple d'une entité dans un document indexé, nous définissons un ensemble d'identifiants rattaché à l'ensemble \mathcal{T} . Nous notons Δ cet ensemble, et δ_j ses éléments. Nous notons \mathcal{N}_{Δ} la cardinalité de Δ .

$$\Delta = \{\delta_1, \delta_2, \delta_3, \dots, \delta_j, \dots, \delta_{\mathcal{N}_{\Delta}}\}$$

À chaque identifiant δ_j de Δ correspond une entité de l'ensemble \mathcal{T} . Cette correspondance est définie par la fonction $\mathcal{F}_{\delta\tau}$:

$$\mathcal{F}_{\delta\tau}: \Delta \rightarrow \mathcal{T}$$

$$\delta_j \rightarrow \tau_k = \mathcal{F}_{\delta\tau}(\delta_k)$$

Nous notons δ_0 l'identifiant référençant l'entité vide: $\mathcal{F}_{\delta\tau}(\delta_0) = \tau_0$

Dans les exemples, et par souci de clarté, nous noterons un identifiant d'entité δ_j :

$$\delta_j : \mathcal{F}_{\delta\tau}(\delta_j)$$

Nous noterons, par exemple, l'identifiant δ_1 qui référence l'entité ⋈ ainsi :

$$\delta_1 : \text{⋈}$$

3.2. Indexation des documents

Nous définissons ici le vocabulaire d'indexation qui va servir de base à l'indexation et décrivons les documents indexés.

3.2.1. Le vocabulaire d'indexation

Nous désignons par *unité d'indexation*, tout triplet ayant un sens (logique), et formé par deux identifiants d'entités δ_i et δ_j de Δ et une relation ρ_t de \mathcal{R} .

Nous notons $\nu_{\mathcal{D}k}$ une unité d'indexation.

$$\nu_{\mathcal{D}k} = (\delta_i, \delta_j, \rho_t)$$

Exemple

$\nu_{\mathcal{D}1} = (\delta_1 : \text{⋈}, \delta_2 : \text{⋈}, \text{proche_de})$, $\nu_{\mathcal{D}2} = (\delta_1 : \text{⋈}, \delta_3 : \text{⋈}, \text{sur})$ et $\nu_{\mathcal{D}3} = (\delta_4 : \text{⋈}, \delta_5 : \text{pêcheur}, \text{est_un})$ sont des unités d'indexation.

$\nu = (\delta_1 : \text{☆}, \delta_2 : \text{⋈}, \text{est_un})$ n'est pas considérée comme une unité d'indexation car elle n'a pas de sens (dans un cadre applicatif qui reflète la réalité)

Notons que l'unité d'indexation $(\delta_i, \delta_0, \rho_0)$ telle que $\mathcal{F}_{\delta\tau}(\delta_0) = \tau_0$ représente, dans un document indexé, l'identifiant d'entité δ_i lorsque ce dernier n'est en relation avec aucun autre identifiant d'entité. On remarquera que l'indexation standard à base de mots simples, par exemple, sera fondée uniquement sur des unités d'indexation de la forme $(\delta_i, \delta_0, \rho_0)$.

L'ensemble des unités d'indexation $\nu_{\mathcal{D}k}$ forme le vocabulaire d'indexation $\mathcal{V}_{\mathcal{D}}$.

Rappelons que $\mathcal{V}_{\mathcal{D}}$ est de cardinalité $\mathcal{N}_{\mathcal{V}_{\mathcal{D}}}$.

$$\mathcal{V}_{\mathcal{D}} \in \mathcal{P}(\Delta \times \Delta \times \mathcal{R})$$

$$\mathcal{V}_{\mathcal{D}} = \{\nu_{\mathcal{D}1}, \nu_{\mathcal{D}2}, \nu_{\mathcal{D}3}, \dots, \nu_{\mathcal{D}j}, \dots, \nu_{\mathcal{D}\mathcal{N}_{\mathcal{V}_{\mathcal{D}}}}\}.$$

3.2.2. Le corpus indexé

Rappelons qu'un document indexé DI_i est une représentation du contenu du document D_i du corpus \mathcal{C} basée sur le vocabulaire \mathcal{V} . Il s'agit d'un sous-ensemble de l'ensemble du vocabulaire d'indexation de cardinalité \mathcal{N}_{DI_i} .

$$DI_i \in \mathcal{P}(\mathcal{V}_{\mathcal{D}})$$

DI_i s'écrit :

$$DI_i = \{\upsilon_{DI_i.1}, \upsilon_{DI_i.2}, \upsilon_{DI_i.3}, \dots, \delta_{DI_i.N_{DI_i}}\}$$

Un document indexé DI_i étant défini à partir d'identifiants d'entités référencées et de relations, nous définissons la fonction $\mathcal{F}_{\Delta D}$ qui pour un document indexé renvoi l'ensemble des identifiants d'entités décrivant le document :

$$\begin{aligned} \mathcal{F}_{\Delta D} : CI &\rightarrow \mathcal{P}(\Delta) \\ DI_i &\rightarrow \mathcal{F}_{\Delta D} (DI_i) \end{aligned}$$

Nous notons $\delta_{DI_i.j}$ le $j^{\text{ième}}$ identifiant d'entité du document VD et N_{DI_i} le nombre total de ses identifiants :

$$\mathcal{F}_{\Delta D} (DI_i) = \{\delta_{DI_i.1}, \delta_{DI_i.2}, \dots, \delta_{DI_i.N_{DI_i}}\}$$

Exemple

Prenons l'exemple du document D_5 rappelé à la Figure 33.

L'indexation de ce document donne :

$$DI_5 = \{\upsilon_{D_5.1}, \upsilon_{D_5.2}, \upsilon_{D_5.3}, \upsilon_{D_5.4}, \upsilon_{D_5.5}\}$$

tels que :

- $\upsilon_{D_5.1} = (\delta_{D_5.1: \text{arbre}}, \delta_0, \rho_0)$,
- $\upsilon_{D_5.2} = (\delta_{D_5.2: \text{étoile}}, \delta_0, \rho_0)$,
- $\upsilon_{D_5.3} = (\delta_{D_5.3: \text{mer}}, \delta_{D_5.4: \text{mer}}, \text{proche_de})$,
- $\upsilon_{D_5.4} = (\delta_{D_5.3: \text{mer}}, \delta_{D_5.5: \text{mer}}, \text{sur})$ et
- $\upsilon_{D_5.5} = (\delta_{D_5.4: \text{mer}}, \delta_{D_5.5: \text{mer}}, \text{sur})$

L'ensemble des identifiants de ce document indexé est :

$$\mathcal{F}_{\Delta D} (DI_5) = \{ \delta_{D_5.1: \text{arbre}}, \delta_{D_5.2: \text{étoile}}, \delta_{D_5.3: \text{mer}}, \delta_{D_5.4: \text{mer}}, \delta_{D_5.5: \text{mer}} \}$$

Nous écrivons,

$$DI_5 = \{ (\delta_{D_5.1: \text{arbre}}, \delta_0, \rho_0), (\delta_{D_5.2: \text{étoile}}, \delta_0, \rho_0), (\delta_{D_5.3: \text{mer}}, \delta_{D_5.4: \text{mer}}, \text{proche_de}), (\delta_{D_5.3: \text{mer}}, \delta_{D_5.5: \text{mer}}, \text{sur}), (\delta_{D_5.4: \text{mer}}, \delta_{D_5.5: \text{mer}}, \text{sur}) \}$$

que nous présentons aussi graphiquement dans la Figure 33.

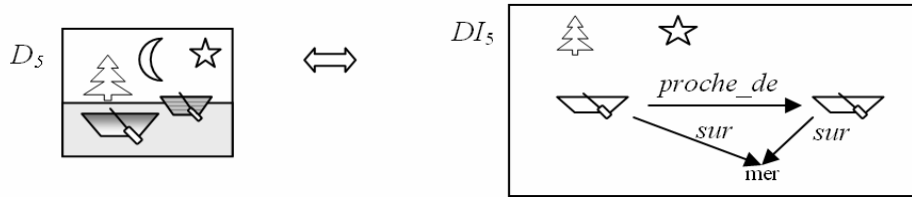


Figure 33 Le document D_5 et le document indexé DI_5

Nous rappelons que le corpus indexé CI est :

$$CI = \{VD, i \in [1, \mathcal{N}_c]\}$$

Exemple

L'indexation structurée de l'ensemble C donné dans l'exemple de la Figure 19 (page56) donne le corpus indexé:

$$CI = \{DI_1, DI_2, DI_3, DI_4, DI_5, DI_6, DI_7, DI_8\}$$

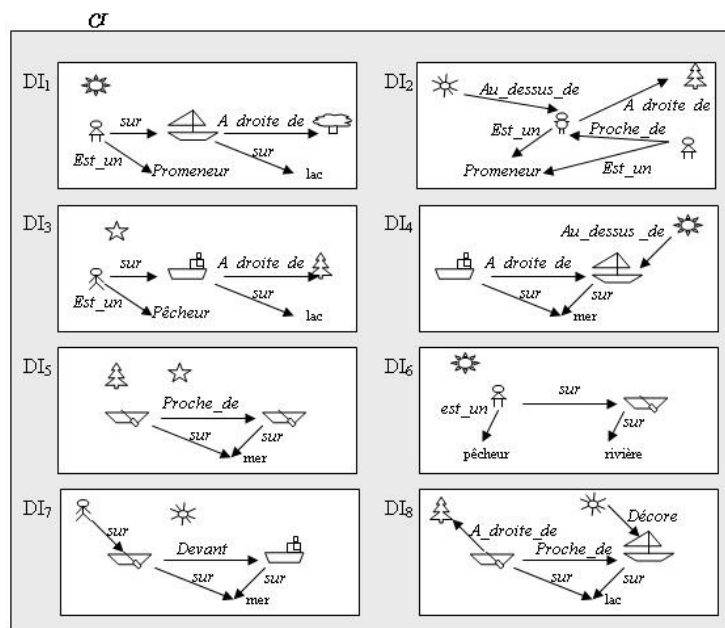


Figure 34 Exemple de corpus indexé

3.3. Formulation de la requête

Nous décrivons ici l'ensemble des critères formé par les ensembles \mathcal{O} (obligation/option) et \mathcal{J} (certitude/incertitude) que nous associons au vocabulaire \mathcal{V} . Cette association nous donne les ensembles d'identifiants d'interrogation Δ_Q et de relations d'interrogation \mathcal{R}_Q dont la combinaison permet de définir le vocabulaire d'interrogation \mathcal{V}_Q .

La requête q est un sous-ensemble de ce vocabulaire d'interrogation.

3.3.1. Les critères Obligation/Option et Certitude/Incertitude

Nous notons \mathcal{O} l'ensemble permettant de définir les critères d'obligation et d'option,

$$\mathcal{O} = \{\text{obligatoire, optionnel}\}$$

et \mathcal{J} l'ensemble permettant de définir les critères de certitude et d'incertitude.

$$\mathcal{J} = \{\text{certain, incertain}\}$$

Nous définissons en général, les fonctions associées aux ensembles \mathcal{O} et \mathcal{J} :

- F_{Ob} la fonction qui définit pour chaque triplet de $(\Delta U\mathcal{R}) \times \mathcal{O} \times \mathcal{J}$, la valeur de son critère \mathbf{a} d'obligation/option :

$$F_{Ob}: (\Delta U\mathcal{R}) \times \mathcal{O} \times \mathcal{J} \rightarrow \mathcal{O}$$

$$(x, \mathbf{a}, \mathbf{c}) \rightarrow F_{Ob}(x) = \mathbf{a}$$

- F_{Cer} la fonction qui définit pour chaque triplet de $(\Delta U\mathcal{R}) \times \mathcal{O} \times \mathcal{J}$, la valeur de son critère \mathbf{c} de certitude/incertitude:

$$F_{Cer}: (\Delta U\mathcal{R}) \times \mathcal{O} \times \mathcal{J} \rightarrow \mathcal{J}$$

$$(x, \mathbf{a}, \mathbf{c}) \rightarrow F_{Cer}(x) = \mathbf{c}$$

3.3.2. Les identifiants d'interrogation

Une entité τ_i peut être obligatoire ou optionnelle et certaine ou incertaine.

Comme nous désirons permettre l'utilisation multiple des entités, nous augmentons chaque identifiant d'entité de deux critères \mathbf{a}_k et \mathbf{c}_k : le premier exprime son obligation/option (représenté dans l'ensemble \mathcal{O}) et le second exprime sa certitude/incertitude (représenté dans l'ensemble \mathcal{J}).

Nous définissons alors un *identifiant d'interrogation*, noté δ_{Qk} comme un triplet:

$$\delta_{Qk} = (\delta_j, \mathbf{a}_k, \mathbf{c}_k)$$

Les triplets ainsi formés constituent l'ensemble des entités d'interrogation Δ_Q .

$$\Delta_Q = \Delta \times \mathcal{O} \times \mathcal{J}$$

Nous associons par défaut à l'élément δ_0 les critères *optionnel* et *incertain* et noterons δ_{Q0} l'entité d'interrogation ($\delta_0, \text{optionnel}, \text{incertain}$).

Nous définissons la fonction \mathcal{F}_δ qui, à chaque identifiant d'interrogation associe son identifiant d'entité:

$$\begin{aligned} \mathcal{F}_\delta: \quad \Delta_Q &\rightarrow \Delta \\ \delta_Q = (\delta, \mathbf{a}, \mathbf{c}) &\rightarrow \delta \end{aligned}$$

Rappelons que les fonctions \mathcal{F}_{Ob} et \mathcal{F}_{Cer} définissent respectivement l'obligation/option et la certitude/incertitude de chaque triplet de $(\Delta U \mathcal{R}) \times \mathcal{C} \times \mathcal{J}$. Nous pouvons donc appliquer ces fonctions à l'ensemble Δ_Q .

Ainsi,

$$\begin{aligned} \mathcal{F}_{Ob}(\delta_Q) &= \text{obligatoire} \text{ si l'entité identifiée par } \mathcal{F}_T(\delta_Q) \text{ est obligatoire et } \text{optionnel} \text{ sinon, et} \\ \mathcal{F}_{Cer}(\delta_Q) &= \text{certain} \text{ si l'entité identifiée par } \mathcal{F}_T(\delta_Q) \text{ est certaine et } \text{incertain} \text{ sinon} \end{aligned}$$

Nous pouvons donc écrire:

$$\delta_Q = (\mathcal{F}_\delta(\delta_Q), \mathcal{F}_{Ob}(\delta_Q), \mathcal{F}_{Cer}(\delta_Q))$$

Dans les exemples, nous écrirons un identifiant d'interrogation :

- $(\delta_i : \mathcal{F}_{\delta_i}, ob, cer)$ signifiant que l'entité identifiée par δ_i est obligatoire et certaine,
- $(\delta_i : \mathcal{F}_{\delta_i}, ob, inc)$ signifiant que l'entité identifiée par δ_i est obligatoire et incertaine,
- $(\delta_i : \mathcal{F}_{\delta_i}, op, cer)$ signifiant que l'entité identifiée par δ_i est optionnelle et certaine,
- $(\delta_i : \mathcal{F}_{\delta_i}, op, inc)$ signifiant que l'entité identifiée par δ_i est optionnelle et incertaine.

3.3.3. Les relations d'interrogation

On appelle *relation d'interrogation* un triplet formé par une relation de l'ensemble \mathcal{R} à laquelle sont associés un élément de l'ensemble \mathcal{C} (exprimant le critère d'obligation/option) et un élément de l'ensemble \mathcal{J} (exprimant le critère de certitude/incertitude).

Une relation d'interrogation est notée ρ_{Qk} :

$$\rho_{Qk} = (\rho_j, \mathbf{a}_k, \mathbf{c}_k)$$

L'ensemble des triplets possibles est l'ensemble des relations d'interrogation noté \mathcal{R}_Q .

$$\mathcal{R}_Q = \mathcal{R} \times \mathcal{C} \times \mathcal{J}$$

Nous associons par défaut à la relation ρ_0 les critères *optionnel* et *incertain* et noterons ρ_{Q0} la relation d'interrogation ($\rho_0, \text{optionnel}, \text{incertain}$).

Nous définissons la fonction \mathcal{F}_p qui, à chaque relation d'interrogation, associe la relation de \mathcal{R} qui lui correspond:

$$\begin{aligned} \mathcal{F}_p: \quad \mathcal{R}_Q &\rightarrow \mathcal{R} \\ \rho_Q = (\rho_i, \mathbf{a}, \mathbf{c}) &\rightarrow \mathcal{F}_p(\rho_Q) = \rho_i \end{aligned}$$

Rappelons que les fonctions \mathcal{F}_{Ob} et \mathcal{F}_{Cer} définissent respectivement l'obligation/option et la certitude/incertitude de triplet de $(\Delta \cup \mathcal{R}) \times \mathcal{C} \times \mathcal{I}$. Nous pouvons donc appliquer ces fonctions à l'ensemble \mathcal{R}_Q .

Ainsi,

$$\begin{aligned} \mathcal{F}_{Ob}(\rho_Q) &= \textit{obligatoire} \text{ si la relation } \mathcal{F}_p(\rho_Q) \text{ est obligatoire et } \textit{optionnel} \text{ sinon, et} \\ \mathcal{F}_{Cer}(\rho_Q) &= \textit{certain} \text{ si la relation } \mathcal{F}_p(\rho_Q) \text{ est certaine et } \textit{incertain} \text{ sinon.} \end{aligned}$$

Nous pouvons donc écrire:

$$\rho_Q = (\mathcal{F}_p(\rho_Q), \mathcal{F}_{Ob}(\rho_Q), \mathcal{F}_{Cer}(\rho_Q))$$

Dans les exemples, nous écrirons une relation d'interrogation sous une des formes suivantes:

- (ρ_i, ob, cer) signifie que la relation ρ_i est obligatoire et certaine,
- (ρ_i, ob, inc) signifie que la relation ρ_i est obligatoire et incertaine,
- (ρ_i, op, cer) signifie que la relation ρ_i est optionnelle et certaine,
- (ρ_i, op, inc) signifie que la relation ρ_i est optionnelle et incertaine.

3.3.4. Le vocabulaire d'interrogation

Nous définissons le *vocabulaire d'interrogation* \mathcal{V}_Q :

$$\mathcal{V}_Q \in \mathcal{P}(\Delta_Q \times \Delta_Q \times \mathcal{R}_Q)$$

Ainsi, une *unité d'interrogation*, notée v_{Qk} , est un triplet formé de deux entités d'interrogation de Δ_Q et d'une relation d'interrogation de \mathcal{R}_Q .

$$v_{Qk} = (\delta_{Qi}, \delta_{Qj}, \rho_{Qk})$$

L'unité d'interrogation $(\delta_{Qi}, \delta_{Qj}, \rho_{Qk})$ sert à représenter, dans une requête, l'identifiant d'interrogation δ_{Qi} lorsque ce dernier n'est en relation avec aucun autre identifiant d'interrogation.

Nous définissons :

- $\mathcal{F}_{v\delta_1}$ la fonction qui, à chaque unité d'interrogation (\mathbf{v}_{Qk}), associe son premier (1er) identifiant d'interrogation (δ_{Qj}) :

$$\begin{aligned} \mathcal{F}_{v\delta_1}: \quad \mathcal{V}_Q &\rightarrow \Delta_Q \\ \mathbf{v}_{Qk} = (\delta_{Qj}, \delta_{Qj}, \rho_{Qt}) &\rightarrow \mathcal{F}_{v\delta_1}(\mathbf{v}_{Qk}) = \delta_{Qj} \end{aligned}$$

- $\mathcal{F}_{v\delta_2}$ la fonction qui, à chaque unité d'interrogation (\mathbf{v}_{Qk}), associe son deuxième (2ème) identifiant d'interrogation (δ_{Qj}) :

$$\begin{aligned} \mathcal{F}_{v\delta_2}: \quad \mathcal{V}_Q &\rightarrow \Delta_Q \\ \mathbf{v}_{Qk} = (\delta_{Qj}, \delta_{Qj}, \rho_{Qt}) &\rightarrow \mathcal{F}_{v\delta_2}(\mathbf{v}_{Qk}) = \delta_{Qj} \end{aligned}$$

- $\mathcal{F}_{v\rho}$ la fonction qui, à chaque unité d'interrogation (\mathbf{v}_{Qk}), associe sa relation d'interrogation (ρ_{Qt}) :

$$\begin{aligned} \mathcal{F}_{v\rho}: \quad \mathcal{V}_Q &\rightarrow \mathcal{R}_Q \\ \mathbf{v}_{Qk} = (\tau_{Qi}, \tau_{Qj}, \rho_{Qt}) &\rightarrow \mathcal{F}_{v\rho}(\mathbf{v}_{Qk}) = \rho_{Qt} \end{aligned}$$

Ainsi, une unité d'interrogation \mathbf{v}_Q peut s'écrire :

$$\mathbf{v}_Q = (\mathcal{F}_{v\delta_1}(\mathbf{v}_Q), \mathcal{F}_{v\delta_2}(\mathbf{v}_Q), \mathcal{F}_{v\rho}(\mathbf{v}_Q)) = (\delta_{Qj}, \delta_{Qj}, \rho_{Qt})$$

3.3.5. La requête

Comme nous l'avons défini dans le modèle général, une requête q est un élément de $\mathcal{P}(\mathcal{V}_Q)$.

Elle est définie par \mathcal{N}_q unités d'interrogation. La $k^{\text{ième}}$ unité d'interrogation de q est notée \mathbf{v}_{qk} .

Une requête q s'écrit alors :

$$q = \{\mathbf{v}_{q.1}, \mathbf{v}_{q.2}, \dots, \mathbf{v}_{q.k}, \dots, \mathbf{v}_{q.\mathcal{N}_q}\}$$

Chaque unité d'interrogation de q est formée par deux identifiants d'interrogation et une relation d'interrogation. Un identifiant d'interrogation étant un identifiant d'entité augmenté de deux critères : obligatoire/optionnel et certain/incertain, et une relation d'interrogation étant une relation augmentée aussi de deux critères : obligatoire/optionnel et certain/incertain.

$$\begin{array}{c} \mathbf{v}_{qk} = ((\mathcal{F}_\delta(\delta_{Qj}), \mathcal{F}_{O\delta}(\delta_{Qj}), \mathcal{F}_{Cer}(\delta_{Qj})), (\mathcal{F}_\delta(\delta_{Qm}), \mathcal{F}_{O\delta}(\delta_{Qm}), \mathcal{F}_{Cer}(\delta_{Qm})), (\mathcal{F}_\rho(\rho_{Qt}), \mathcal{F}_{O\rho}(\rho_{Qt}), \mathcal{F}_{Cer}(\rho_{Qt}))) \\ \underbrace{\hspace{10em}} \qquad \underbrace{\hspace{10em}} \qquad \underbrace{\hspace{10em}} \\ \delta_{Qj} \qquad \qquad \delta_{Qm} \qquad \qquad \rho_{Qt} \\ \parallel \qquad \qquad \parallel \qquad \qquad \parallel \\ \mathcal{F}_{v\delta_1}(\mathbf{v}_{qk}) \qquad \mathcal{F}_{v\delta_2}(\mathbf{v}_{qk}) \qquad \mathcal{F}_{v\rho}(\mathbf{v}_{qk}) \end{array}$$

Exemples de requêtes








Soient les requêtes q_1 et q_2 :

$$- q_1 = \{((\delta_1: \text{bateau}, ob, cer), \delta_{Q0}, \rho_{Q0})\}$$

q_1 signifie que l'on désire retrouver tous les documents contenant un .

$$- q_2 = \{((\delta_1: \text{bateau}, ob, inc), (\delta_2: \text{soleil}, op, cer), (sous, ob, cer)), ((\delta_3: \text{personne}, ob, cer), (\delta_1: \text{bateau}, ob, inc), (sur, ob, inc))\}$$

q_2 signifie que l'on désire retrouver tous les documents:

- contenant un  ou toute autre entité qui lui est proche,
- contenant ou pas un ,
- contenant un ,
- et tels que :
 - ✓  ou l'un de ses proches est situé « sous »  lorsque ce dernier est présent,
 - ✓  est situé « sur »- ou toute autre relation qui lui est proche-  ou toute autre entité qui lui est proche

De façon à accéder aux différents éléments de la requête en fonction des critères qui leurs sont associés, nous définissons deux classifications : la première concerne les identifiants d'entités et la seconde concerne les unités d'interrogation. Ces deux classifications sont utiles pour la définition de la fonction de correspondance que nous développons dans le § 3.4.

a. Classification des identifiants d'entités de la requête

L'ensemble des *identifiant d'entités* d'une requête est déterminé grâce à la fonction F_Δ que nous définissons ainsi :

$$F_\Delta: CQ \rightarrow \mathcal{P}(\Delta)$$

$$q \rightarrow F_\Delta(q) = \bigcup_{v_{q,k} \in q} F_\delta(F_{\nu\delta_1}(v_{q,k})) \cup F_\delta(F_{\nu\delta_2}(v_{q,k}))$$

Nous notons :

- $\mathcal{N}_{\Delta,q}$ la cardinalité de $F_\Delta(q)$.
- $F_\Delta(q) = \{\delta_{q,1}, \delta_{q,2}, \dots, \delta_{q,j}, \dots, \delta_{q,\mathcal{N}_{\Delta,q}}\}$ où $\delta_{q,j}$ est le $j^{\text{ème}}$ identifiant d'entité de q .

De façon à accéder aux différents identifiants d'entités de la requête en fonction des critères qui leurs sont associés, nous définissons :

- Δ_{ObCer} la fonction qui définit pour une requête l'ensemble de ses identifiants d'entités **O**bligatoires et **C**ertaines:

$$\begin{aligned} \Delta_{ObCer}: CQ &\rightarrow \mathcal{P}(\Delta) \\ q &\rightarrow \{\delta_j \in \mathcal{F}_\Delta(q) / (\delta_j, \text{obligatoire, certain}) \in \mathcal{F}_{\nu\delta_1}(\nu_{q,k}) \cup \mathcal{F}_{\nu\delta_2}(\nu_{q,k}) \} \end{aligned}$$

- Δ_{ObInc} la fonction qui définit pour une requête l'ensemble de ses identifiants d'entités **O**bligatoires et **I**ncertaines:

$$\begin{aligned} \Delta_{ObInc}: CQ &\rightarrow \mathcal{P}(\Delta) \\ q &\rightarrow \{\delta_j \in \mathcal{F}_\Delta(q) / (\delta_j, \text{obligatoire, incertain}) \in \mathcal{F}_{\nu\delta_1}(\nu_{q,k}) \cup \mathcal{F}_{\nu\delta_2}(\nu_{q,k}) \} \end{aligned}$$

- Δ_{OpCer} la fonction qui définit pour une requête l'ensemble de ses identifiants d'entités **O**ptionnelles et **C**ertaines:

$$\begin{aligned} \Delta_{OpCer}: CQ &\rightarrow \mathcal{P}(\Delta) \\ q &\rightarrow \{\delta_j \in \mathcal{F}_\Delta(q) / (\delta_j, \text{optionnel, certain}) \in \mathcal{F}_{\nu\delta_1}(\nu_{q,k}) \cup \mathcal{F}_{\nu\delta_2}(\nu_{q,k}) \} \end{aligned}$$

- Δ_{OpInc} la fonction qui définit pour une requête l'ensemble de ses identifiants d'entités **O**ptionnelles et **I**ncertaines:

$$\begin{aligned} \Delta_{OpInc}: CQ &\rightarrow \mathcal{P}(\Delta) \\ q &\rightarrow \{\delta_j \in \mathcal{F}_\Delta(q) / (\delta_j, \text{optionnel, incertain}) \in \mathcal{F}_{\nu\delta_1}(\nu_{q,k}) \cup \mathcal{F}_{\nu\delta_2}(\nu_{q,k}) \} \end{aligned}$$

L'ensemble des identifiants d'entités de q peut alors s'écrire :

$$\mathcal{F}_\Delta(q) = \Delta_{ObCer}(q) \cup \Delta_{ObInc}(q) \cup \Delta_{OpCer}(q) \cup \Delta_{OpInc}(q)$$

Remarque : nous plaçons toujours les identifiants d'entités obligatoires et certaines en tête de l'ensemble $\mathcal{F}_\Delta(q)$, suivies par les identifiants d'entités obligatoires et incertaines. Nous plaçons ensuite les identifiants d'entités optionnelles et certaines et terminons par les identifiants d'entités optionnelles et incertaines.

La Figure 35 schématise la répartition des identifiants d'entités d'une requête selon leurs critères associés : l'obligation/option et la certitude/incertitude.

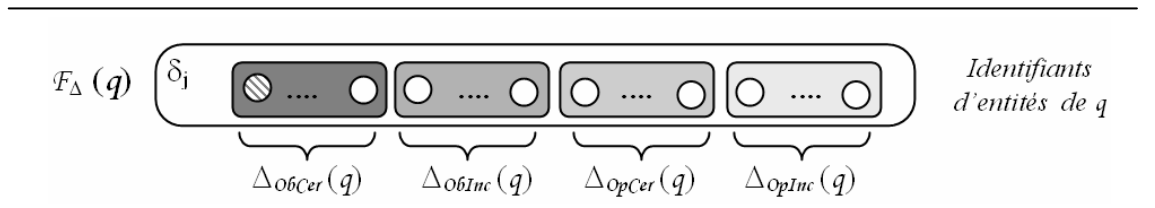


Figure 35 Classification des entités d'une requête q selon leurs critères

b. Classification des unités d'interrogation de la requête

De manière similaire à la classification des identifiants d'entité d'une requête, nous classifions les unités d'interrogation de la requête en fonction des critères associés à leurs relations. Nous définissons donc:

- \mathcal{V}_{ObCer} la fonction qui définit pour une requête l'ensemble de ses unités d'interrogation dont la relation est **Obligatoire et Certaine**:

$$\mathcal{V}_{ObCer}: CQ \rightarrow \mathcal{P}(\mathcal{V}Q)$$

$$q \rightarrow \{ \mathbf{v}_{q,k} \in q / \mathcal{F}_{Ob}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{obligatoire} \text{ et } \mathcal{F}_{Cer}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{certain} \}$$

- \mathcal{V}_{ObInc} la fonction qui définit pour une requête l'ensemble de ses unités d'interrogation dont la relation est **Obligatoire et Incertaine**:

$$\mathcal{V}_{ObInc}: CQ \rightarrow \mathcal{P}(\mathcal{V}Q)$$

$$q \rightarrow \{ \mathbf{v}_{q,k} \in q / \mathcal{F}_{Ob}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{obligatoire} \text{ et } \mathcal{F}_{Cer}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{incertain} \}$$

- \mathcal{V}_{OpCer} la fonction qui définit pour une requête l'ensemble de ses unités d'interrogation dont la relation est **Optionnelle et Certaine**:

$$\mathcal{V}_{OpCer}: CQ \rightarrow \mathcal{P}(\mathcal{V}Q)$$

$$q \rightarrow \{ \mathbf{v}_{q,k} \in q / \mathcal{F}_{Ob}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{optionnel} \text{ et } \mathcal{F}_{Cer}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{certain} \}$$

- \mathcal{V}_{OpInc} la fonction qui définit pour une requête l'ensemble de ses unités d'interrogation dont la relation est **Optionnelle et Incertaine**:

$$\mathcal{V}_{OpInc}: CQ \rightarrow \mathcal{P}(\mathcal{V}Q)$$

$$q \rightarrow \{ \mathbf{v}_{q,k} \in q / \mathcal{F}_{Ob}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{optionnel} \text{ et } \mathcal{F}_{Cer}(\mathcal{F}_{vp}(\mathbf{v}_{q,k})) = \textit{incertain} \}$$

La requête q peut alors s'écrire :

$$q = \mathcal{V}_{ObCer}(q) \cup \mathcal{V}_{ObInc}(q) \cup \mathcal{V}_{OpCer}(q) \cup \mathcal{V}_{OpInc}(q)$$

La Figure 36 schématise la répartition des unités d'interrogations d'une requête selon les critères de leurs relations (l'obligation/option et la certitude/incertitude):

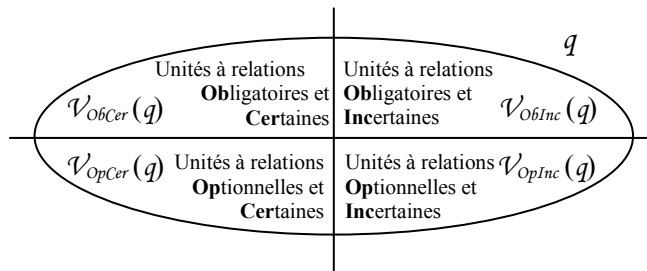


Figure 36 Répartition des unités d'une requête q selon les critères de leurs relations

3.4. Correspondance entre la requête et les documents

Soit la requête :

$$q = \{v_{q,1}, v_{q,2}, \dots, v_{q,k}, \dots, v_{q,n_q}\} \text{ où}$$

$$v_{q,k} = (F_{v\delta_1}(v_{q,k}), F_{v\delta_2}(v_{q,k}), F_{v\rho}(v_{q,k})) = (\delta_{q,j}, \delta_{q,m}, \rho_{q,t})$$

Et soit le document indexé VD représentant le document D_i :

$$VD = \{v_{D_i,1}, v_{D_i,2}, \dots, v_{D_i,j}, \dots, v_{D_i,n_{D_i}}\}$$

Nous rappelons qu'un document D_i est pertinent pour une requête q , si et seulement si $Corresp(q, VD)$ est vrai.

Cette fonction de correspondance $Corresp$ doit tenir compte des contraintes engendrées par les choix effectués dans la représentation des documents et des requêtes. Nous nous intéressons dans le paragraphe suivant à ces contraintes et proposons par la suite (§ 3.4.2) la fonction $Corresp$ qui respecte ces contraintes.

3.4.1. Contraintes pour la correspondance entre la requête et les documents indexés

La représentation des documents et de la requête dans le modèle que nous proposons sont caractérisés par (a) l'association de deux critères, obligation et certitude, aux éléments de la requête, ainsi que (b) l'utilisation d'un ensemble intermédiaire : les identifiants, permettant l'utilisation multiple d'une même entité. Ceci implique certaines contraintes à respecter lors de la correspondance requête/document.

a. Contraintes liées aux critères :

Un document D_i répond à une requête q s'il vérifie les critères de ses identifiants d'interrogation ainsi que ceux de ses relations d'interrogation.

Dire que « D_i vérifie les critères des identifiants d'interrogation de q » signifie que :

- à chaque identifiant d'entité obligatoire et certaine de q doit correspondre un identifiant d'entité de DI_i référant exactement la même entité;
- à chaque identifiant d'entité obligatoire et incertaine de q doit correspondre un identifiant d'entité de DI_i référant une entité qui lui est proche;
- à chaque identifiant d'entité optionnelle et certaine de q doit correspondre un ou zéro identifiant d'entité de DI_i référant exactement la même entité
- à chaque identifiant d'entité optionnelle et incertaine de q doit correspondre un ou zéro identifiant d'entité de DI_i référant une entité qui lui est proche.

Dire que « D_i vérifie les critères des relations d'interrogation de q » signifie que :

Pour chaque unité $(\delta_{Q_1}, \delta_{Q_2}, \rho)$ de q dont les deux identifiants d'entités ont chacun un correspondant, qu'on notera δ_{D_1} et δ_{D_2} , dans le document VD :

- Si sa relation ρ est obligatoire et certaine, le document DI_i doit contenir une unité d'indexation ayant pour identifiants δ_{D1} et δ_{D2} (dans l'ordre) et ayant pour relation exactement la relation ρ .
- Si sa relation ρ est obligatoire et incertaine, le document DI_i doit contenir une unité d'indexation ayant pour identifiants δ_{D1} et δ_{D2} (dans l'ordre) et ayant pour relation une relation proche de ρ .
- Si sa relation ρ est optionnelle et certaine, le document DI_i peut ou non contenir une unité d'indexation ayant pour identifiants δ_{D1} et δ_{D2} (dans l'ordre) et ayant pour relation exactement la relation ρ .
- Si sa relation ρ est optionnelle et incertaine, le document DI_i peut ou non contenir une unité d'indexation ayant pour identifiants δ_{D1} et δ_{D2} (dans l'ordre) et ayant pour relation une relation proche de ρ .

b. Contraintes liées à l'utilisation multiple d'entités:

L'utilisation multiple d'entités impose qu'à un identifiant d'entité du document corresponde au maximum un identifiant d'entité de la requête :

Dans notre modèle, la correspondance entre un document et une requête nécessite de relier à chaque identifiant d'entité de la requête son correspondant unique dans le document (s'il existe), en respectant les contraintes liées aux critères qui leur sont associés. Cette correspondance « un à un » entre les identifiants d'entités de la requête et ceux du document indexé est nécessaire en raison de l'utilisation multiple d'entité. En effet, il n'est pas possible de se limiter à la vérification d'appartenance des entités de la requête au document indexé car cette méthode engendre deux types d'incohérence. D'abord, les documents contenant un unique identifiant d'entité référençant une entité τ , seraient renvoyés comme réponse à une requête contenant plus d'un identifiant d'entité obligatoires référençant cette même entité τ . Ensuite, il peut y avoir un risque d'ambiguïté au niveau des relations, ainsi, une relation de la requête peut être vérifiée dans un document indexé considéré alors comme pertinent, seulement cette relation ne relie pas les identifiants d'entité adéquats.

Afin de mieux comprendre ces deux incohérences, qui pourraient apparaître en utilisant une vérification de l'appartenance simple des entités de la requête aux documents indexés, nous proposons de présenter un exemple relatif à chacun d'eux.


Premier cas :

Soit la requête :

$$q = \{((\delta_1: \text{☒}, ob, cer), \delta_{Q0}, \rho_{Q0}), ((\delta_2: \text{☒}, ob, inc), \delta_{Q0}, \rho_{Q0})\}$$

Il s'agit de retrouver les documents contenant obligatoirement deux entités :

- un ☒ (l'entité est obligatoire et certaine), et
- un ☒ ou ☒ ou ☒ (puisque l'entité ☒ est obligatoire et incertaine)

En supposant que la fonction de correspondance s'appuie sur la vérification de l'appartenance des entités de la requête au document indexé (en respectant les critères), tous les documents contenant au moins un  seraient considérés comme répondant à la requête (voir Figure 37).

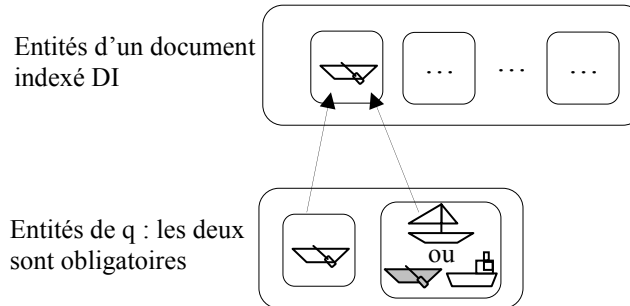


Figure 37 Vérification de l'appartenance des entités de q à un document DI

Ainsi, les documents DI_5 , DI_6 , DI_7 et DI_8 (voir Figure 38) seraient tous considérés comme répondant à la requête alors qu'il est bien évident que le document DI_6 ne devrait pas l'être.

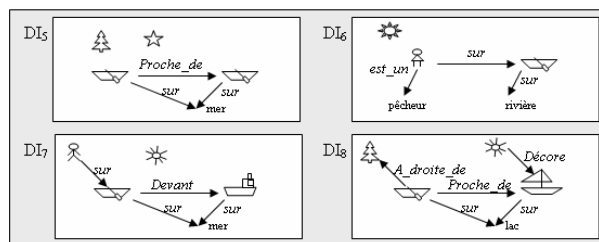





Figure 38 Documents indexés jugés pertinents par la correspondance basée sur l'appartenance

Deuxième cas :

Soit la requête : «Retrouver les documents représentant un  (ou une entité proche) qui se trouve sur une  (ou une entité proche) qui se trouve elle-même derrière une  (ou une entité proche)» représentée ainsi :

$$q = \{ ((\delta_1: \text{person icon}, ob, cer), (\delta_2: \text{boat icon}, ob, inc), (sur, ob, cer)), \\ ((\delta_3: \text{boat icon}, ob, cer), (\delta_2: \text{boat icon}, ob, inc), ((Devant, ob, cer))) \}$$

Les documents répondant à cette requête devraient présenter, dans leur contenu, les éléments tels que représentés dans la Figure 40:

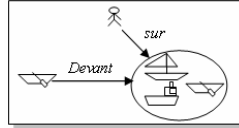




Figure 39 Schématisation du contenu obligatoire d'un document pertinent pour q






Nous pouvons constater que le document DI_7 ne satisfait pas une telle contrainte, pourtant si notre correspondance s'appuie sur l'appartenance des entités de la requête au document indexé, ce document serait considéré comme répondant à la requête. En effet (voir Figure 40) :

L'entité incertaine référencée par δ_2 dans la requête a été considérée dans le DI_7 comme :

-  pour la première unité d'interrogation de q ,
-  pour la deuxième unité d'interrogation de q ,

Ce qui ne correspond pas à la requête qui veut que ces deux entités soient identiques (c'est la même barque qui porte l'homme et qui se trouve derrière l'autre barque.)

L'entité δ_1 n'est pas en relation avec l'entité adéquate. En effet :

- Dans q , δ_1 doit se trouver sur le (, , ) qui est « devant » .
- Dans DI_7 , δ_1 se trouve sur le  qui est « derrière ».

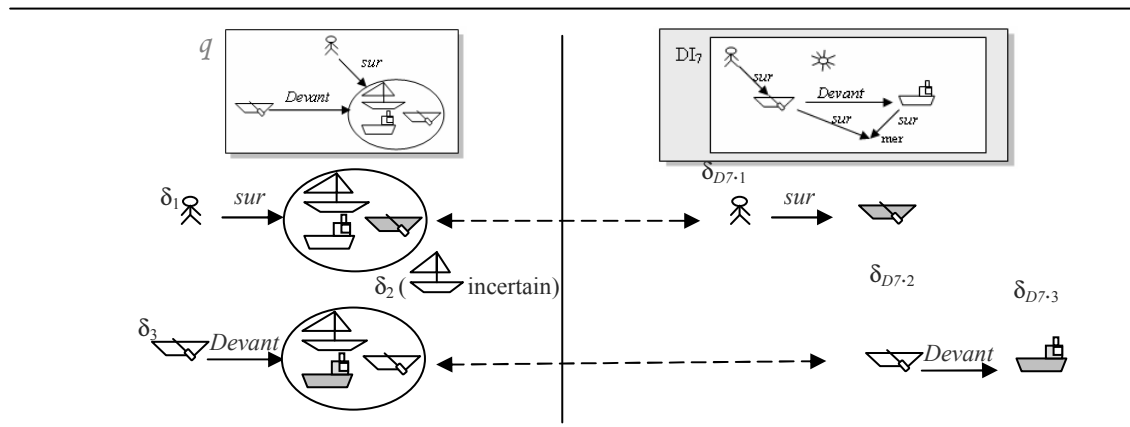


Figure 40 Le document D_7 serait considéré comme pertinent pour q si la correspondance s'appuyait sur l'appartenance des unités de q au document indexé

Afin d'éviter ces incohérences, autrement dit, afin de tenir compte de la contrainte relative à l'utilisation multiple d'une même entité, nous nous basons sur une mise en correspondance « un à un » de certains identifiants d'entités de la requête (selon les critères qui leurs sont associés) et des identifiants d'entités adéquats des documents indexés.

Cette correspondance est développée dans ce qui suit.

3.4.2. La fonction de correspondance *Corresp*

Dans une requête q , certains identifiants d'entités réfèrent des entités obligatoires et d'autres réfèrent des entités optionnelles. Chaque document répondant à la requête va donc :

- contenir des identifiants correspondant à tous les identifiants de q référant des entités obligatoires qu'elles soient certaines ou incertaines.
- contenir des identifiants correspondant à certains identifiants de q référant des entités optionnelles (l'autre partie n'apparaissant pas dans le document). Dans ce cas, nous considérons :
 - ✓ le sous-ensemble de Δ_{OpCer} qui contient les identifiants d'entités optionnelles et certaines ayant un correspondant dans le document. Nous notons cet ensemble Δ'_{OpCer} .
 - ✓ le sous-ensemble de Δ_{OpInc} qui contient les identifiants d'entités optionnelles et incertaines ayant un correspondant dans le document. Nous notons cet ensemble Δ'_{OpInc} .

De manière similaire, un document va contenir des identifiants d'entités ayant un correspondant dans la requête et des identifiants n'ayant pas de correspondant dans cette même requête. Nous considérons donc le sous-ensemble de VD qui contient les identifiants d'entités correspondant à des identifiants de q . Nous notons cet ensemble VD

La correspondance se fait donc entre les ensembles $\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc}$ d'un côté et DI' de l'autre. Cette correspondance est caractérisée par une fonction injective que nous notons $F_{Corresp}$ et qui à chaque identifiant δ de $\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc}$ associe un unique identifiant δ' de VD qui vérifie :

-Si δ est certaine, alors l'entité référencée par δ' doit être identique à celle référencée par δ :

$$\text{Si } \delta \in \Delta_{ObCer} \cup \Delta'_{OpCer}, F_{\delta\tau}(\delta') = F_{\delta\tau}(\delta)$$

-Si δ est incertaine, alors l'entité référencée par δ' doit être identique à celle référencée par δ ou être l'un de ses proches (selon la fonction de proximité entre entités) :

$$\text{Si } \delta \in \Delta_{OpInc} \cup \Delta'_{OpInc}, F_{\delta\tau}(\delta') \in \mathcal{V}D(F_{\delta\tau}(\delta))$$

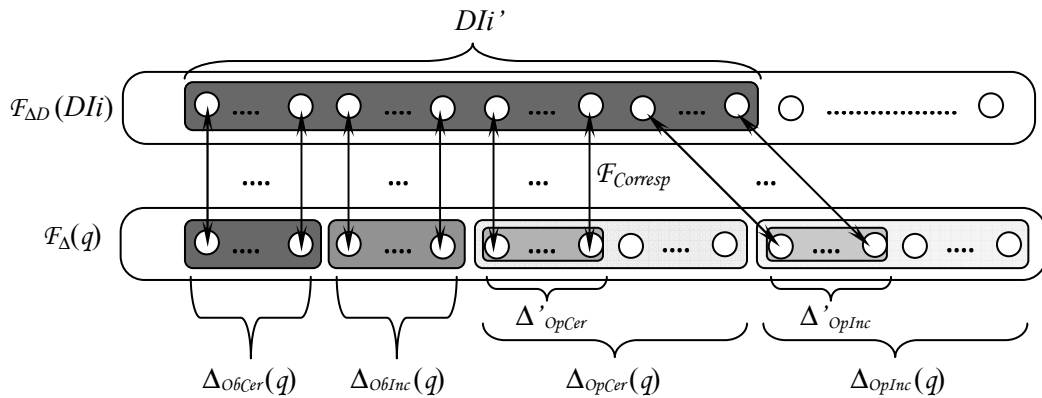


Figure 41 Correspondance un à un entre identifiants d'entité de DI' et de q

Nous définissons donc la fonction *Corresp* ainsi :

Corresp(q, VD) est vrai ssi :

$\exists \Delta'_{OpCer} \subseteq \Delta_{OpCer}(q), \Delta'_{OpInc} \subseteq \Delta_{OpInc}(q)$ et $DI' \subseteq \mathcal{F}_{\Delta D}(DI)$, et

\exists une fonction injective $\mathcal{F}_{Corresp}$ définie de $\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc}$ dans DI' , tels que :

(i) si $\delta \in \Delta_{ObCer} \cup \Delta'_{OpCer}, \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta)) = \mathcal{F}_{\delta\tau}(\delta)$,

et si $\delta \in \Delta_{ObInc} \cup \Delta'_{OpInc}, \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta)) \in \mathcal{Proche}_{\mathcal{T}}(\mathcal{F}_{\delta\tau}(\delta))$

(ii) $\forall v_{q,k} = (\delta_{Q_i}, \delta_{Q_j}, \rho_{Q_t}) \in q$ tel que $(\delta_{Q_i}, \delta_{Q_j}) \in (\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc})^2$, $\delta_1 = \mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\delta_{Q_i}))$, $\delta_2 = \mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\delta_{Q_j}))$ et $\rho = \mathcal{F}_{\rho}(\rho_{Q_t})$,

si $v_{q,k} \in \mathcal{V}_{ObCer}, (\delta_1, \delta_2, \rho) \in DI_i$,

et si $v_{q,k} \in \mathcal{V}_{ObInc}, \exists \rho' \in \mathcal{Proche}_{\mathcal{R}}(\rho)$ tel que $(\delta_1, \delta_2, \rho') \in DI_i$

La fonction de correspondance, telle que nous l'avons définie, permet uniquement de savoir si un document est pertinent par rapport à la requête ou non, et ne permet pas d'affecter un score aux documents pertinents. Néanmoins, il est possible de classer les documents retrouvés selon certaines caractéristiques qu'ils partagent, caractéristiques identifiées par un ensemble d'entités et de relations présentes dans la requête (par exemple, tous les documents contenant les mêmes éléments *optionnels* de la requête seront rangés dans une même classe). Cette classification des documents retournés par le système est explicitée dans le paragraphe suivant.

Il est aussi possible de définir une fonction, que l'on notera S , et qui permet d'affecter un score aux documents (en supposant que des pondérations sont affectées aux identifiants d'entités et/ou aux unités des documents indexés). Cette fonction S , pouvant se présenter sous différentes formes, sera appliquée aux documents VD tels que *Corresp*(q, VD) est vrai¹.

4. Une approche pour la reformulation de la requête

La proposition faite du modèle permet, entre autres, à l'utilisateur de formuler son besoin, en décrivant ce qu'il pense être le document 'idéal' répondant à ce besoin, sachant qu'il connaît a priori les documents et donc qu'il utilise son souvenir de ce document idéal pour formuler sa requête. Ceci implique, comme nous l'avons mentionné précédemment, qu'il sera plus ou moins important que certains éléments soient contenus dans les documents susceptibles d'intéresser l'utilisateur, et que parmi ces éléments, il y en a dont l'utilisateur sera certain et d'autre dont il sera moins sûr. En fonction de ces critères, un ensemble de documents va être renvoyé à l'utilisateur. Nous proposons, en s'inspirant d'une proposition de [Denos,97a], que les documents retournés par le système soient repartis dans des classes dites « de pertinence ». Chaque classe regroupe les documents partageant des caractéristiques communes relatives au contenu de la requête. Ces classes sont définies dans le § 4.1.

¹ Un exemple de fonction S est utilisé dans la partie validation (chapitre VIII. 1)

La visualisation de ce premier échantillon de documents renvoyés par le système devrait permettre à l'utilisateur d'avoir une meilleure idée des documents qu'il recherche. En marquant les documents qu'il juge pertinents, l'utilisateur a une idée plus claire du document 'idéal' répondant à son besoin, il peut alors reformuler son besoin, ou interagir avec le système afin d'obtenir une nouvelle requête, résultat de l'application de certaines actions sur l'ancienne, actions que nous définissons dans le § 4.3. Ces actions sont appliquées en fonction de la répartition des documents jugés pertinents par l'utilisateur dans les classes de pertinence (voir §4.2). Le but est que tous les documents jugés pertinents se retrouvent au sein d'une même classe.

4.1. Définition des classes de pertinence

Une classe de pertinence est un sous-ensemble des documents retrouvés qui partagent certaines caractéristiques identifiées par un ensemble d'entités et de relations présentes dans la requête. Les identifiants et unités d'interrogation dont les critères sont obligatoires et certains définissent des caractéristiques de pertinence communes à tous les documents retrouvés. Ceux dont les critères sont optionnels ou incertains définissent des caractéristiques qui servent à organiser les documents retrouvés selon des classes représentatives des particularités de leur relation de pertinence avec la requête.

Notre formulation de la requête étant basée sur deux critères, nous proposons deux types de classification des documents (l'utilisateur peut en choisir une, ou passer de l'une à l'autre) :

- dans la première classification, qui organise les documents en des Classes Options, on retrouve dans une même classe les documents contenant tous les mêmes éléments optionnels de la requête (voir §4.1.1),
- dans la deuxième classification qui organise les documents en des Classes Incertitude, on retrouve dans une même classe les documents contenant tous les mêmes éléments *incertains* apparaissant tels que mentionnés dans la requête (voir §4.1.2).

L'ordre dans lequel les classes sont affichées n'est pas lié à une valeur numérique issue d'un calcul abstrait de pertinence mais est défini par rapport aux éléments optionnels ou incertains contenus dans les classes(voir §4.1.3).

Rappelons qu'une requête q s'écrit :

$$q = \{v_{q,1}, v_{q,2}, \dots, v_{q,k}, \dots, v_{q,n_q}\} \text{ où}$$

$$v_{q,k} = (F_{v\delta_1}(v_{q,k}), F_{v\delta_2}(v_{q,k}), F_{v\rho}(v_{q,k})) = (\delta_{q,j}, \delta_{q,m}, \rho_{q,t})$$

Exemple

En considérant le vocabulaire donné en exemple à la page 65, nous considérons la requête :

$$q = \{((\delta_1: \text{⌘}, ob\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_2: \text{⌘}, op\ c), \delta_{Q0}, \rho_{Q0}), ((\delta_3: \text{⌘}, op\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_4: \text{⌘}, ob, c), \delta_{Q0}, \rho_{Q0})\}$$

Cette requête nous servira d'exemple tout au long de ce paragraphe.

4.1.1. Les Classes Options (\mathcal{CO})

Nous considérons ici les identifiants d'entités et les unités *optionnelles*. Chaque classe correspond à une façon de combiner ces éléments optionnels de la requête.

Rappelons que :

- Δ'_{OpCer} est le sous-ensemble de $\Delta_{OpCer}(q)$ contenant les identifiants d'entités optionnelles et certaines de la requête ayant un correspondant dans le document.
- Δ'_{OpInc} est le sous-ensemble de $\Delta_{OpInc}(q)$ contenant les identifiants d'entités optionnelles et incertaines de la requête ayant un correspondant dans le document.
- $\mathcal{V}_{OpCer}(q)$ est l'ensemble des unités optionnelles et certaines de la requête (relations optionnelles et certaines).
- $\mathcal{V}_{OpInc}(q)$ est l'ensemble des unités optionnelles et incertaines de la requête (relations optionnelles et incertaines).

Soit \mathcal{A}_{Op} l'ensemble des combinaisons possibles des identifiants d'entités optionnels δ_{Opj} de $\Delta'_{OpCer} \cup \Delta'_{OpInc}$. Chaque combinaison de \mathcal{A}_{Op} est dénotée par un nombre binaire $s\delta_{Op}$ de longueur $\mathcal{N}\Delta_{Op}$. Un 1 en $j^{\text{ème}}$ place de $s\delta_{Op}$ indique que le $j^{\text{ème}}$ identifiant d'entité optionnel δ_{Opj} de $\Delta'_{OpCer} \cup \Delta'_{OpInc}$ a un correspondant dans le document (ie. $F_{Corresp}(\delta_{Opj})$ existe) et un 0 indique le contraire.

De même, soit $\mathcal{S}\mathcal{V}_{Op}$ l'ensemble des combinaisons possibles des unités d'interrogation optionnelles ν_{Opj} de $\mathcal{V}_{OpCer}(q) \cup \mathcal{V}_{OpInc}(q)$. Chaque combinaison de $\mathcal{S}\mathcal{V}_{Op}$ est dénotée par un nombre binaire $s\nu_{Op}$ de longueur $\mathcal{N}\mathcal{V}_{Op}$. Un 1 en $j^{\text{ème}}$ place de $s\nu_{Op}$ indique que la $j^{\text{ème}}$ unité d'interrogation optionnelle ν_{Opj} de $\mathcal{V}_{OpCer}(q) \cup \mathcal{V}_{OpInc}(q)$ a un correspondant dans le document et un 0 indique le contraire.

Une Classe Option $CO_{s\delta_{Op}^{s\nu_{Op}}}$, où $s\delta_{Op} \in \mathcal{A}_{Op}$ et $s\nu_{Op} \in \mathcal{S}\mathcal{V}_{Op}$, est définie par :

$CO_{s\delta_{Op}^{s\nu_{Op}}} = \{D \in \mathcal{D}(q) \text{ tels que :}$

(i) $\forall j \in [1, \mathcal{N}\Delta_{Op}]$ si $s\delta_{Op}(j)=1$ alors $\exists \delta \in D / F_{Corresp}(\delta_{Opj}) = \delta$ et

(ii) $\forall j \in [1, \mathcal{N}\mathcal{V}_{Op}]$ si $s\nu_{Op}(j)=1$ alors $\exists \nu = (\delta_1, \delta_2, \rho) \in D /$

$F_{Corresp}(F_{\delta}(F_{\nu\delta_1}(\nu_{Opj}))) = \delta_1, F_{Corresp}(F_{\delta}(F_{\nu\delta_2}(\nu_{Opj}))) = \delta_2$ et

$\rho = F_{\rho}(F_{\nu\rho}(\nu_{Opj}))$ si $F_{cer}(F_{\nu\rho}(\nu_{Opj})) = \text{certain}$ ou $\rho \in \text{Proche}_{\mathcal{R}}(F_{\rho}(F_{\nu\rho}(\nu_{Opj})))$ si $F_{cer}(F_{\nu\rho}(\nu_{Opj})) = \text{incertain}$

Exemple

Rappelons la requête :

$q = \{((\delta_1: \text{⌘}, ob\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_2: \text{⌘}, op\ c), \delta_{Q0}, \rho_{Q0}), ((\delta_3: \text{⌘}, op\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_4: \text{⌘}, ob, c), \delta_{Q0}, \rho_{Q0})\}$

Dans cette requête, deux entités sont optionnelles : il s'agit de $\delta_2: \text{⌘}$ et $\delta_3: \text{⌘}$. Et il n'y a pas de relations entre les entités.

A l'issus de la phase de recherche, nous obtenons quatre classes options, toutes contenant les deux entités obligatoires $\delta_1: \text{⌘}$ et $\delta_4: \text{⌘}$. Il s'agit des classes $CO_{00}, CO_{10}, CO_{01}$ et CO_{11} :

CO_{00}^- contient les documents ne contenant ni $\hat{\otimes}$, ni ($\hat{\oplus}$ ou $\hat{\boxplus}$ ou $\hat{\boxminus}$).	
CO_{10}^- contient les documents contenant $\hat{\otimes}$ et ne contenant pas ($\hat{\oplus}$ ou $\hat{\boxplus}$ ou $\hat{\boxminus}$).	
CO_{01}^- contient les documents ne contenant pas $\hat{\otimes}$ et contenant ($\hat{\oplus}$ ou $\hat{\boxplus}$ ou $\hat{\boxminus}$).	
CO_{11}^- contient les documents contenant $\hat{\otimes}$ et contenant ($\hat{\oplus}$ ou $\hat{\boxplus}$ ou $\hat{\boxminus}$).	

4.1.2. Les Classes Incertitude (\mathcal{CI})

Nous considérons ici les identifiant d'entités et unité *incertaines*. Chaque classe correspond à une façon de combiner ces éléments incertains de la requête.

Rappelons que :

- $\Delta_{ObInc}(q)$ est l'ensemble des identifiants d'entités obligatoires et incertains de la requête (ayant tous un correspondant dans le document).
- $\Delta_{OpInc}(q)$ est l'ensemble des identifiants d'entités optionnels et incertains de la requête.
- $\mathcal{V}_{ObInc}(q)$ est l'ensemble des unités obligatoires et incertaines de la requête (relations obligatoires et incertaines).
- $\mathcal{V}_{OpInc}(q)$ est l'ensemble des unités optionnelles et incertaines de la requête (relations optionnelles et incertaines).

En suivant la même démarche que précédemment :

Soit $\mathcal{S}\Delta_{Inc}$ l'ensemble des combinaisons possibles des identifiants d'entités incertains δ_{Incj} de $\Delta_{ObInc}(q) \cup \Delta_{OpInc}(q)$. Chaque combinaison de $\mathcal{S}\Delta_{Inc}$ est dénotée par un nombre binaire $s\delta_{Inc}$ de longueur $\mathcal{N}\Delta_{Inc}$. Un 1 (resp. 0) en $j^{\text{ème}}$ place de $s\delta_{Inc}$ indique que le $j^{\text{ème}}$ identifiant d'entité incertain δ_{Incj} de $\Delta_{ObInc}(q) \cup \Delta_{OpInc}(q)$ a un correspondant dans le document référant la même entité. Un 0 indique le contraire.

De même, soit $\mathcal{S}\mathcal{V}_{Inc}$ l'ensemble des combinaisons possibles des unités d'interrogations incertaines v_{Incj} de $\mathcal{V}_{ObInc}(q) \cup \mathcal{V}_{OpInc}(q)$. Chaque combinaison de $\mathcal{S}\mathcal{V}_{Inc}$ est dénotée par un nombre binaire sv_{Inc} de longueur $\mathcal{N}\mathcal{V}_{Inc}$. Un 1 en $j^{\text{ème}}$ place de sv_{Op} indique que la $j^{\text{ème}}$ unité

d'interrogation optionnelle v_{Incj} de $\mathcal{V}_{ObInc}(q) \cup \mathcal{V}_{OpInc}(q)$ a un correspondant ayant exactement la même relation dans le document. Un 0 indique le contraire.

Une Classe Incertitude $CI_{s\delta_{Inc}}^{sv_{Inc}}$, où $s\delta_{Inc} \in \mathcal{S}\Delta_{Inc}$ et $sv_{Inc} \in \mathcal{S}\mathcal{V}_{Inc}$, est définie par :

$CI_{s\delta_{Inc}}^{sv_{Inc}} = \{ D \in \mathcal{D}(q) \text{ tels que :}$

(iii) $\forall j \in [1.. \mathcal{N}\Delta_{Inc}]$ si $s\delta_{Inc}(j)=1$ alors $\exists \delta \in D / \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta_{Incj})) = \mathcal{F}_{\delta\tau}(\delta)$ et

(iv) $\forall j \in [1.. \mathcal{N}\mathcal{V}_{Inc}]$ si $sv_{Inc}(j)=1$ alors $\exists v = (\delta_1, \delta_2, \rho) \in D /$

$\mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\mathcal{F}_{v\delta_1}(v_{Opj}))) = \delta_1, \mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\mathcal{F}_{v\delta_2}(v_{Opj}))) = \delta_2$ et $\rho = \mathcal{F}_{\rho}(\mathcal{F}_{v\rho}(v_{Opj}))$

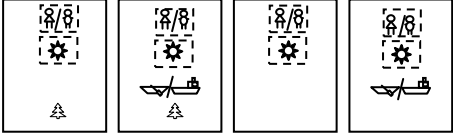
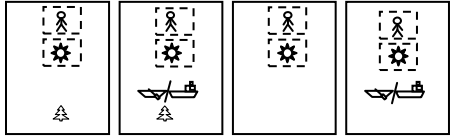
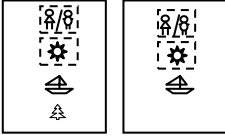
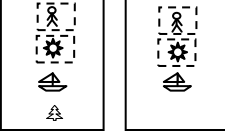
Exemple

Rappelons la requête :

$q = \{ ((\delta_1: \text{⌘}, ob\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_2: \text{⌘}, op\ c), \delta_{Q0}, \rho_{Q0}), ((\delta_3: \text{⌘}, op\ i), \delta_{Q0}, \rho_{Q0}), ((\delta_4: \text{⌘}, ob, c), \delta_{Q0}, \rho_{Q0}) \}$

Dans cette requête, deux entités sont incertaines : il s'agit de $\delta_1: \text{⌘}$ et $\delta_3: \text{⌘}$. Et il n'y a pas de relations entre les entités.

A l'issus de la phase de recherche, nous obtenons quatre classes incertitude. Il s'agit des classes $CI_{00}^-, CI_{10}^-, CI_{01}^-$ et CI_{11}^- :

CI_{00}^-	contient les documents ne contenant ni ⌘ , ni ⌘ .	
CI_{10}^-	contient les documents contenant exactement ⌘ et ne contenant pas ⌘ .	
CI_{01}^-	contient les documents ne contenant pas ⌘ et contenant exactement ⌘ .	
CI_{11}^-	contient les documents contenant exactement ⌘ et contenant exactement ⌘ .	

4.1.3. Degrés de pertinence

Le degré de pertinence d'une classe est fonction du nombre d'éléments (identifiants d'entités et unités) optionnels ou du nombre d'éléments incertains (selon le classement considéré) pris en compte dans cette classe.

Classes Option

Le degré de pertinence d'une Classe Option est :

$$d^{\circ}(CO_{s\delta_{op}}^{sv_{op}}) = card(\{ j \in [1..N\Delta_{Op}], s\delta_{op}(j) = 1 \}) + card(\{ j \in [1..NV_{Op}], sv_{op}(j) = 1 \})$$

Les classes option de degré le plus élevé sont celles qui vérifient le plus d'éléments optionnels de la requête. Les classes vérifiant le même nombre d'éléments optionnels de la requête sont de même degré de pertinence.

Classes Incertitude

Le degré de pertinence d'une classe incertitude est :

$$d^{\circ}(CI_{s\delta_{inc}}^{sv_{inc}}) = card(\{ j \in [1..N\Delta_{Inc}], s\delta_{inc}(j) = 1 \}) + card(\{ j \in [1..NV_{Inc}], sv_{inc}(j) = 1 \})$$

Les classes certitude de degré le plus élevé sont celles qui vérifient le plus d'éléments incertains de la requête. Les classes vérifiant le même nombre d'éléments incertains de la requête sont de même degré de pertinence.

4.2. Distributions possibles des documents jugés pertinents dans les classes

Nous considérons ici la répartition des documents jugés pertinents par l'utilisateur, parmi les documents retournés par le système, dans les classes de pertinence. Etant donné que la classification des documents retournés par le système peut se faire selon le critère d'option ou selon le critère de certitude, nous obtenons donc deux distributions différentes des documents jugés pertinents par l'utilisateur dans les classes de pertinence, selon le critère de classification considéré (voir l'exemple de la Figure 42).

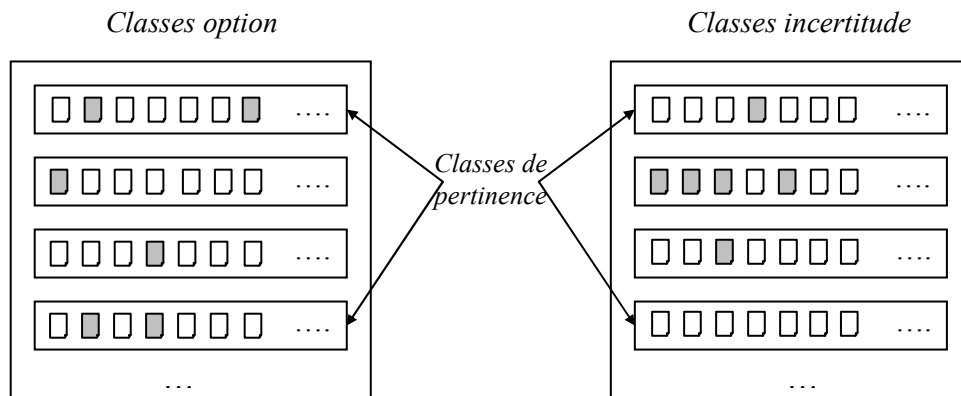


Figure 42 Exemple de répartition des documents jugés pertinents par l'utilisateur (en gris) dans les classes de pertinence

Le choix du classement à retenir pour la reformulation est fonction de la distribution des documents jugés pertinents par l'utilisateur dans les classes de pertinence. Ainsi nous favoriserons le classement dont les documents pertinents sont concentrés sur un nombre limité de classes (dans l'exemple donné, nous favoriserons le classement selon le critère d'incertitude).

Il est aussi possible de combiner les actions à effectuer sur la requête initiale, en considérant de façon transparente, les deux classifications.

En considérant la classification optimale, nous pouvons distinguer différentes distributions des documents jugés pertinents. Trois situations, inspirées de [Deno,97a], peuvent être considérées : nous les expliquons dans ce qui suit, et dégageons pour chaque situation les actions à entreprendre pour reformuler la requête. Ces actions sont définies par la suite.

S1 : Pas de documents jugés pertinents par l'utilisateur

Les documents retournés par le système ne correspondent pas à son besoin d'information. Soit, le corpus ne contient pas d'excellentes réponses à la requête d'origine, soit l'utilisateur s'est mal exprimé ou a posé des contraintes trop strictes, la requête telle qu'il l'a posée n'est pas en adéquation avec ce qu'il recherche : Les caractéristiques des documents retournés devraient lui permettre de mieux identifier son réel besoin. Il doit relâcher les contraintes ou remplacer et/ou supprimer certaines entités et relations pour atteindre d'autres documents.

S2 : Répartition des documents jugés pertinents dans un petit nombre de classes

Si la classification optimale est une classification selon le critère d'obligation/option, alors :

- les éléments *optionnels* apparaissant dans les documents de la classe Option, contenant des documents jugés pertinents, se trouvent être « *obligatoires* » pour l'utilisateur. Il faut donc resserrer les contraintes sur ces éléments
- les éléments *optionnels*, n'apparaissant dans aucun document de cette classe, doivent être supprimés.

Et si la classification optimale est une classification selon le critère de certitude/incertitude, alors :

- les éléments *incertains*, apparaissant tels que mentionnés dans la requête, dans les documents de la classe Incertitude, contenant des documents jugés pertinents, se trouvent être « *certaines* » pour l'utilisateur. Il faut donc resserrer les contraintes sur ces éléments,
- les éléments *incertains* apparaissant dans les documents jugés pertinents de cette classe sous certaines formes *proches* de celle mentionnée dans la requête sont les seuls intéressants pour l'utilisateur. Il faut donc mettre à jour la fonction de proximité.

Dans les deux cas, si en plus, tous ces documents contiennent un élément commun non mentionné dans la requête, il faut l'y ajouter.

S3 : Documents jugés pertinents éparpillés dans différentes classes

Dans ce cas, la requête n'est pas assez précise. L'utilisateur doit préciser sa requête en y ajoutant des éléments. Si tous les documents jugés pertinents contiennent un élément commun non mentionné dans la requête, il faut l'y ajouter automatiquement.

D'un autre côté, les contraintes doivent être resserrées en essayant de supprimer les classes qui n'ont aucun document jugé pertinent.

Récapitulatif

Les situations de distributions des documents marqués pertinents ainsi que les actions qui s'y rapportent ne prétendent pas à l'exhaustivité. Cependant, cela donne une base sur laquelle il est possible de s'appuyer pour aider l'utilisateur à interagir avec le système. Ceci a pour objectif d'aboutir à une requête décrivant avec le plus de précision possible le document 'idéal' (en réduisant les doutes de l'utilisateur).

Nous reprenons dans le tableau suivant, les trois situations répertoriées ainsi que les actions qui se rapportent à chacune d'elles. Ces dernières sont définies dans le paragraphe qui suit.

Situations	Actions
S1	Relâcher Remplacer Supprimer
S2	Resserrer Ajouter Supprimer (\mathcal{C}) Mettre à jour (\mathcal{J})
S3	Resserrer Ajouter

Tableau 3 Récapitulatif des actions sur la requête en fonction de la répartition des documents jugés pertinents dans les classes

4.3. Description des actions pour la reformulation de la requête

Nous avons détecté six actions permettant de modifier la requête initiale selon la situation rencontrée (S1, S2 ou S3) et l'intention de l'utilisateur. Il s'agit d'actions qui s'appliquent soit aux contraintes (Relâcher et Resserrer), soit aux entités et aux relations (Ajouter, Supprimer et Remplacer), soit aux fonctions de proximité (Mettre à jour)

Nous reprenons, ici, ces actions, les spécialisons (par exemple, Relâcher se spécialise en deux actions : RelâcherOb et RelâcherCer) et les spécifions.

a. Relâcher

Deux types d'opérations ont pour but de relâcher les contraintes imposées dans la requête : *RelâcherOb* permet de marquer des éléments (entités/rerelations) initialement obligatoires comme optionnels et *RelâcherCer* permet de marquer des éléments initialement certains comme incertains. Ces deux fonctions peuvent être globales ou paramétrées. Dans le premier cas,

l'action est appliquée à tous les éléments de la requête ($RelâcherOb()$ et $RelâcherCer()$) et dans le deuxième cas, elle est appliquée uniquement à un élément χ spécifié de la requête initiale ($RelâcherOb(\chi)$ et $RelâcherCer(\chi)$).

b. Resserrer

Nous définissons cette opération comme une action inverse de l'action *Relâcher*.

Deux types d'opérations ont pour but de resserrer les contraintes imposées dans la requête : *ResserrerOb* permet de marquer des éléments (entités/rerelations) initialement optionnels comme obligatoires et *ResserrerCer* permet de marquer des éléments initialement incertains comme certains. Ces deux fonctions, comme celles relatives à la relaxe des contraintes, peuvent être globales ou paramétrées. Dans le premier cas, l'action est appliquée à tous les éléments de la requête ($ResserrerOb()$ et $ResserrerCer()$) et dans le deuxième cas, elle est appliquée uniquement à un élément χ spécifié de la requête initiale ($ResserrerOb(\chi)$ et $ResserrerCer(\chi)$).

c. Supprimer / Ajouter / Remplacer

Ces trois opérations sont paramétrées, c'est-à-dire qu'elles s'appliquent à un élément χ spécifié de la requête initiale.

- *Supprimer* (χ) supprime l'élément χ de la requête. S'il s'agit d'une entité (plus précisément d'un identifiant d'entité), toute relation en contact avec cette entité sera aussi supprimée. La requête sera réécrite en fonction de cela.
- *Ajouter* (δ) ajoute l'identifiant δ à la requête, avec les critères *optionnel* et *incertain*.
- *Ajouter* (δ_Q) ajoute l'identifiant δ à la requête, avec les critères spécifiés dans ρ_Q .
- *Ajouter* ($\rho, \delta_{Q1}, \delta_{Q2}$) ajoute l'unité ($\delta_{Q1}, \delta_{Q2}, \rho_Q$) à la requête, avec les critères *optionnel* et *incertain* appliqués à ρ (ρ_Q).
- *Ajouter* ($\rho_Q, \delta_{Q1}, \delta_{Q2}$) ajoute l'unité ($\delta_{Q1}, \delta_{Q2}, \rho_Q$) à la requête, avec les critères spécifiés dans ρ_Q .
- *Remplacer* (χ_1, χ_2) remplace l'élément χ_1 de la requête par l'élément χ_2 . Les critères associés à χ_2 restent les mêmes que ceux associés à χ_1 .

d. Mettre à jour

Cette opération est particulière car elle repose sur une hypothèse non défini dans le langage proposé mais que nous pensons être intéressante, d'un point de vue personnalisation : nous supposons ici, que les fonctions de proximités associées au vocabulaire sont différentes pour chaque utilisateur. Ainsi ces fonctions ne sont pas propres à un vocabulaire donné, mais plutôt à un vocabulaire et à un utilisateur donnés. En effet, nous pouvons supposer que deux utilisateurs s'ils ont des doutes sur un objet, l'étendue de leur doute n'est pas forcément le même : prenons l'exemple de deux utilisateurs qui doutent sur l'entité ⊕ , nous pouvons penser que pour le premier les entités ⊕ et ⊖ font toutes les deux parties des entités qui peuvent remplacer ⊕ , et que pour le deuxième, s'il n'est pas certain que c'est ⊕ , cela peut être ⊖ , mais sûrement pas ⊕ , car il s'y connaît plus, par exemple. Nous pouvons voir ici, que la fonction de proximité n'est pas la même pour les deux utilisateur.

L'opération *MettreAJour* permet, dans ce cas, de modifier l'ensemble des entités ou relations proches à un élément donné.

Les opérations décrites, ci-dessus, peuvent être automatiques ou nécessiter l'intervention de l'utilisateur pour sélectionner précisément la partie de la requête à modifier ou les éléments à ajouter à cette requête. Lorsqu'elles sont automatiques, les opérations sont proposées en fonction d'une analyse du système des documents marqués pertinents et de leur classe d'appartenance (lorsque cela est possible) et lorsqu'elles sont manuelles, cela signifie que l'utilisateur a désormais une meilleure idée de son besoin qui s'est précisé en voyant un premier échantillon des documents du corpus.

5. Conclusion

Nous avons présenté dans ce chapitre notre modèle de recherche d'information, adapté au contexte de recherche dans lequel nous nous situons dans ce travail. Ce contexte rappelons-le, concerne la recherche d'information par des utilisateurs experts du domaine, ayant une mémoire des documents et formulant leur requête en décrivant le document qui répondrait de façon idéale à leur besoin.

Ce contexte particulier impose certaines contraintes que notre modèle prend en compte, ce qui fait toute son originalité. En effet, (i) il est fondé sur un langage complexe, permettant aussi l'utilisation multiple d'une même entité, ce qui répond au besoin de rigueur et d'exhaustivité dans la représentation des documents imposé par l'expertise des utilisateurs, (ii) il enrichi les éléments de la requête par un critère de certitude/incertitude, critère qui reflète les doutes de l'utilisateur engendrés par le souvenir qu'il a des documents qu'il recherche, et (iii) il enrichi les éléments de la requête par un critère d'obligation/option qui traduit l'importance, plus ou moins grande, pour l'utilisateur, que les éléments qu'il utilise pour décrire son document idéal (ie formuler sa requête) soient contenu dans les documents retrouvés.

En plus, les critères enrichissant la requête et l'organisation des documents renvoyés par le système dans des classes, favorisent une approche fondée sur la reformulation de la requête, dont nous avons donné un aperçu en fin de ce chapitre.

Dans la suite de ce manuscrit, et afin de donner vie à notre modèle, nous nous situons dans le cadre applicatif de la recherche de graphiques techniques par des professionnels. Nous proposons, donc, une instance du modèle, que nous venons de décrire, correspondant à la recherche de ce type de graphiques (partie 3-Chapitre VI). Cette instance est définie suite à une identification de éléments les décrivant et à un choix de représentation de ces éléments (partie 3-chapitre V). Nous nous intéressons, par la suite, au processus d'indexation de ces graphiques (partie 4-Chapitre VII) et à une évaluation de notre système (partie 4-Chapitre VIII).



Partie 3. Instanciation du modèle

Dans la deuxième partie de ce manuscrit, nous avons décrit le modèle de recherche que nous proposons. Ce modèle a la particularité de se baser sur un langage complexe permettant de représenter fidèlement le contenu des documents et des requêtes, en introduisant en plus la notion d'utilisation multiple des entités, ce qui est très utile voir primordial dans un contexte où les utilisateurs sont des experts du domaine. Ce modèle a aussi la particularité de fonder l'expression du besoin sur des critères d'obligation et de certitude, traduisant ce qu'il est important ou moins important que les documents contiennent et sur la certitude de l'utilisateur quand à ce contenu. Cela rend bien compte des du contexte de recherche dans lequel nous nous positionnons et qui est, rappelons le, la formulation du besoin en décrivant le document 'idéal' qui répondrait au besoin de l'utilisateur, utilisateur qui a une mémoire des documents susceptibles de l'intéresser.

Mais afin de concrétiser cette proposition, nous l'abordons dans le cadre applicatif de la recherche de graphiques dans les documents techniques à usage professionnel. Comme nous l'avons expliqué en début de ce manuscrit, cette application se prête bien au contexte de recherche qui nous intéresse ici, puisqu'il s'agit d'un côté d'une application dédiée à des utilisateurs experts (les réparateurs de composantes techniques), qui connaissent a priori les documents (mémoire des graphiques qu'ils ont consultés) et à qui l'on propose un accès au document technique via le média graphique, et ce en décrivant le graphique 'idéal' auquel ils aimeraient accéder.

Dans cette partie du manuscrit, nous nous intéressons donc au graphique comme point d'accès à la documentation technique. Afin de modéliser ce type de média, nous empruntons une approche sémantique, qui se base sur l'exploitation du contexte dans lequel apparaît le graphique afin d'enrichir sa sémantique.

Cette partie se déroule en deux chapitres.

Le premier consiste en (i) une étude de la documentation technique ainsi que de ses graphiques, dont le but est de dégager les informations utiles les décrivant de façon exhaustive (Chapitre VI.1) et (ii) une présentation du type de modélisation retenu pour ces graphiques (Chapitre VI.2).

Le deuxième est une description détaillée du modèle de recherche de graphiques, qui est une instance du modèle général proposé dans la partie II et qui se base sur les informations retenues dans le chapitre VI.1. Ce modèle est fondé sur une indexation exhaustive des graphiques (dont le processus et une évaluation qualitative sont proposés dans la partie 4) et sur trois façons possibles d'interrogation (une description visuelle ou textuelle ou combinant les deux modes du graphique 'idéal').

— Chapitre V —

Vers une description exhaustive des graphiques

- ✓ Descriptions à retenir
- ✓ Choix d'une représentation

— Chapitre VI —

Un modèle de recherche de graphiques techniques

- ✓ Indexation : le modèle GRIM
- ✓ Formulation de la requête : 3 langages
- ✓ Correspondance

Chapitre V. Vers une description exhaustive des graphiques

L'information véhiculée par les documents techniques est présente dans deux modes d'expressions qui sont le texte et le graphique : l'un aide l'autre dans le message qu'il veut transmettre au lecteur. Ainsi, ce qui est représenté dans le graphique (traits, zooms, etc.) est complété par le texte qui l'entoure dans le document technique. L'interprétation du graphique peut alors être considérée comme complète lorsqu'elle est couplée au texte avoisinant. Ceci nous a poussé à nous orienter vers une approche sémantique pour représenter les graphiques techniques, en exploitant le contexte dans lequel apparaît ce média, en plus de la prise en compte de ses données intrinsèques.

Dans la première partie de ce chapitre, nous tentons donc, à partir du graphique mais aussi de son contexte textuel, d'identifier les informations qui sont véhiculées par ce média, autrement dit, qu'est ce qui dans ce graphique où dans le texte qui l'accompagne, permet de décrire de manière assez complète le graphique ? Une étude d'un corpus technique nous permet de dégager ces informations (voir aussi [kefi,02]).

Dans la deuxième partie de ce chapitre, nous reprenons ces informations récoltées et les réorganisons afin de définir une description structurée du contenu des graphiques techniques, description sur laquelle nous nous basons dans la définition du modèle de recherche, développé dans le chapitre suivant.

1. Etude du corpus : quelle description du graphique retenir ?

Notre corpus d'étude est constitué de manuels utilisateurs d'imprimantes Xerox. Nous cherchons à identifier les informations véhiculées par les graphiques qu'il contient. Nous nous intéressons donc aux informations intrinsèques du graphique qui concernent ses caractéristiques visuelles (§1.1) ainsi qu'aux informations sémantiques que l'on peut extraire de son contexte textuel et qui concernent le sens qu'il véhicule (§1.2). Dans cette identification des informations à retenir pour la description du contenu des graphiques, nous prenons aussi en compte le contexte dans lequel nous situons notre travail : les utilisateurs ont une mémoire des documents (§1.3).

1.1. Données intrinsèques du graphique

Le graphique est un média qui offre une multitude d'informations intrinsèques permettant une réflexion intéressante sur ce qui peut améliorer son indexation. L'étude des graphiques de notre corpus, nous a permis de dégager trois caractéristiques de ce média particulier, à savoir :

- le graphique est un objet structuré,


- le graphique est un ensemble de formes géométriques, plus ou moins grandes, disposées d'une manière particulière,
- le graphique contient des symboles particuliers utiles à sa compréhension (le zoom, les flèches, etc.) que nous appelons les « illustratifs ».

Nous reprenons ces trois points dans ce qui suit.

1.1.1. Structure

Le dictionnaire Hachette définit le terme structure comme étant : « l'organisation complexe considérée sous l'angle de ses principaux éléments constitutifs. »

D'après cette définition, il est possible de considérer le graphique comme un objet structuré, puisqu'il représente une ou plusieurs composantes matérielles ainsi que leur composition en d'autres composantes matérielles.

Cette décomposition est souvent perçue dans le graphique par un symbole d'énumération tel que .

Dans le graphique, représenté dans la Figure 43, nous distinguons clairement une telle décomposition. En effet, le graphique représente une imprimante composée de dix éléments. L'objet « Imprimante N17 » et les dix objets qui la composent (par exemple « MBF ») sont reliés par une relation composant-composé.

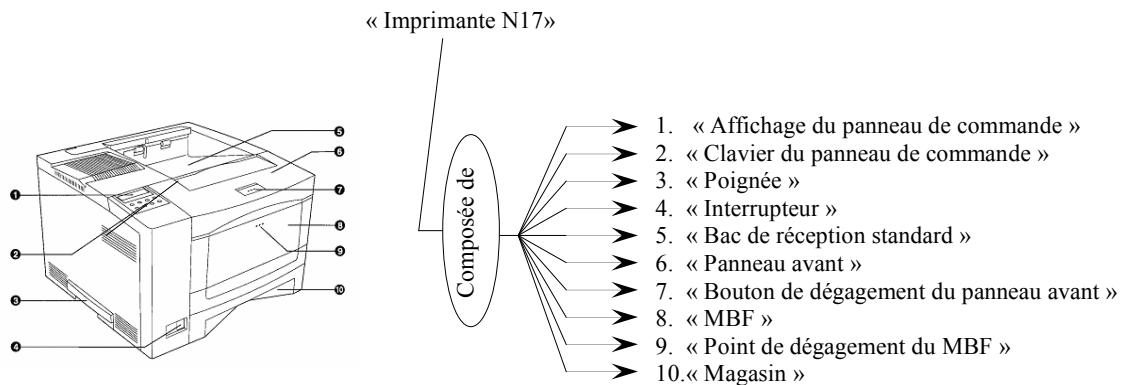


Figure 43 Le graphique est un objet structuré

1.1.2. Formes, tailles et dispositions

Le graphique est un ensemble de lignes, courbes et polygones et les objets qui le composent sont représentés par ces primitives.

Les différents objets contenus dans le graphique peuvent donc être perçus comme un ensemble de formes géométriques associés à leurs contours, ayant des tailles plus ou moins importantes, et une disposition particulière les unes par rapport aux autres. Ainsi, en prenant l'exemple de la

Figure 44, un *petit parallélogramme* peut être associé au panneau de commande. Cet objet se trouve à *gauche* du bac de réception.

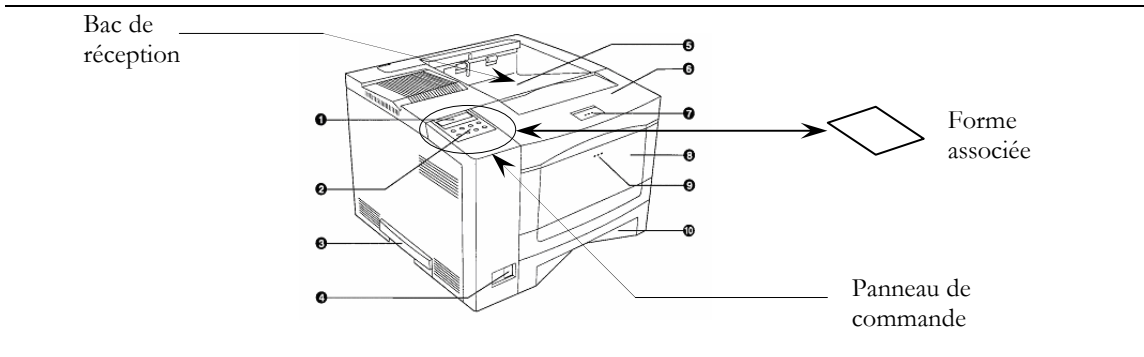


Figure 44 Exemple d'association de forme à un objet du graphique

Nous pouvons aller au-delà de cette description, en prenant en compte le fait que certains objets sont représentés en relief (un cube, par exemple). En effet, les graphiques de notre corpus sont souvent des représentations tridimensionnelles de composantes matérielles ce qui permet de distinguer trois de leurs faces. Ces objets multi-faces peuvent alors être représentés spatialement par trois polygones représentant chacune de leurs faces visibles, et non pas uniquement par un polygone délimitant leur contour.

L'« imprimante » de la Figure 45, sera ainsi représentée spatialement par un polygone représentant sa face avant, un deuxième représentant sa face de dessus, et un troisième représentant sa face gauche.

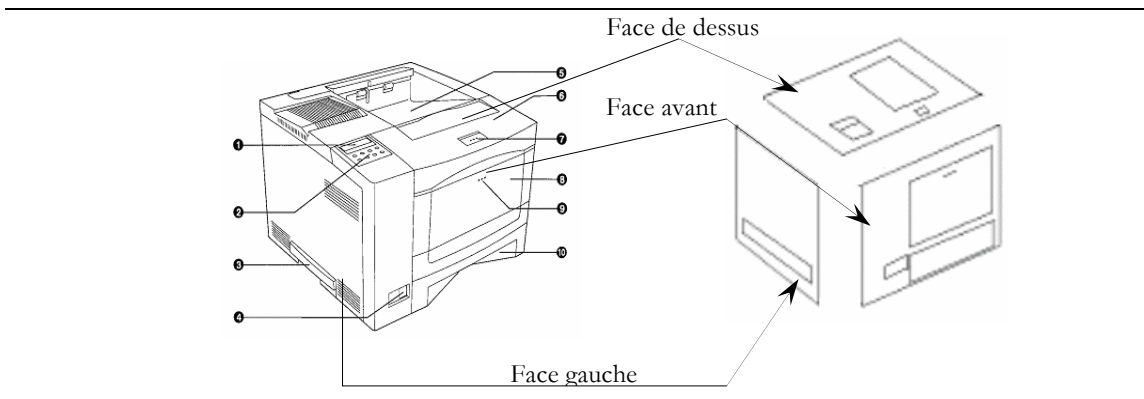


Figure 45 Exemple de représentation spatiale des faces d'un objet graphique multi-faces

En considérant cette représentation tridimensionnelle d'un objet, il nous est possible de considérer que les formes des objets qu'il contient ne sont pas simplement contenues dans son contour, mais plus précisément dans le contour de l'une de ses faces.

Ainsi, dans l'exemple qui suit (Figure 46), l'objet géométrique (a) représentant le bac de réception, n'est pas seulement contenu dans l'objet (c) représentant l'imprimante, mais plus exactement dans la face de dessus de cet objet, autrement dit dans le contour de l'objet (b).

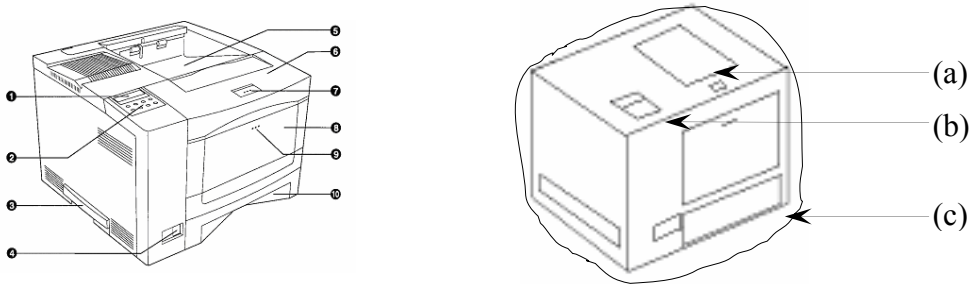


Figure 46 Un exemple de graphique et de sa représentation géométrique

1.1.3. Objets illustratifs

Un aspect intéressant des graphiques techniques est la présence d'objets utilisés par le rédacteur afin de mettre l'accent sur une partie du graphique. A partir de l'étude menée, nous avons distingué trois sortes d'objets illustratifs :

- les zooms : ils permettent de reproduire une composante technique important dans le graphique en question afin de la détailler,
- les flèches (ou mains) : ils permettant de représenter une action sur une composante technique telle que « tirer », « pousser »,
- les énumérations : elles permettant de pointer une ou différentes composantes techniques afin de la (les) nommer et/ou la (les) décrire.

Parmi ces objets illustratifs, le zoom a la particularité de dédoubler un composant déjà représenté dans le graphique. Une relation d'équivalence existe alors entre les deux représentations de ce composant (forme normale et forme zoomée).

La Figure 47 illustre un graphique contenant un zoom, une flèche ainsi qu'une énumération.

Ces objets illustratifs nous semblent représenter une information important dans le graphique puisqu'il s'agit d'une information facilement mémorisable visuellement.

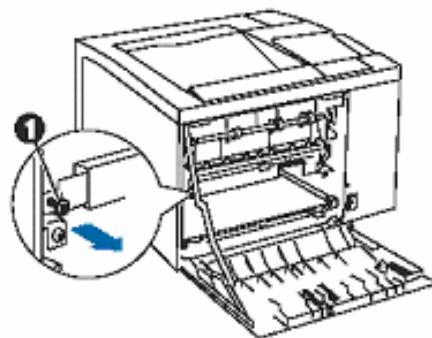


Figure 47 Un exemple de graphique avec illustratifs (zoom, flèche et énumération)

1.2. Données sémantiques enrichissant le graphique

Même si le graphique est considéré comme un média pauvre sémantiquement, une sémantique lui est néanmoins rattachée. Et même si cette sémantique n'apparaît pas explicitement dans le graphique, elle peut être déduite du texte qui l'avoisine dans le document technique.

Ainsi, nous distinguons deux aspects sémantiques rattachés à nos graphiques. Il s'agit d'une part de la description des composantes techniques qu'ils contiennent et d'autre part de la description des procédures à suivre pour accomplir une tâche technique.

a. Description des composantes techniques : aspect descriptif du graphique

Certains graphiques ont pour but d'apporter une description des composantes techniques schématisées. Au niveau du graphique, cela peut être distingué par la présence de l'objet illustratif représentant une énumération. En recherchant cette description dans le texte du document faisant référence au graphique, nous distinguons deux types d'informations descriptives de son contenu :

- la liste des composantes qui y sont représentées,
- la description des propriétés de certaines de ces composantes (ou toutes).

Prenons l'exemple du graphique présenté dans son contexte textuel dans la Figure 48 : on y retrouve les informations citées ci-dessus permettant de décrire « le panneau de commande ». Ces informations sont :

- l'énumération des composantes contenues dans le « panneau de commande » :
 1. Ecran d'affichage
 2. Voyants
 3. Touches de commande

- la description des propriétés de certaines de ces composantes :

Propriété de « Ecran d'affichage » : « deux lignes de 16 caractères ».

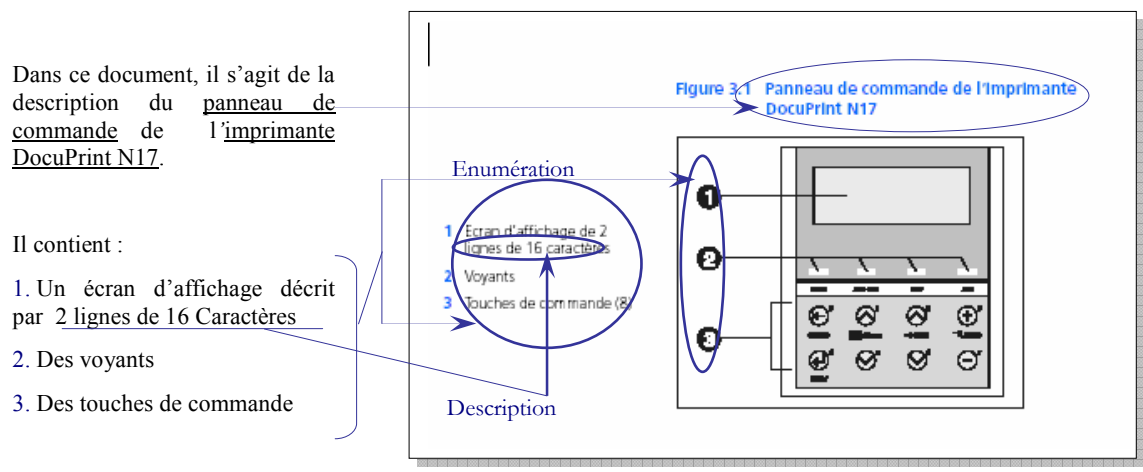


Figure 48 Exemple illustrant l'aspect descriptif du graphique

b. Description des procédures à suivre: aspect opératoire du graphique

De la même façon, certains graphiques ont pour but d'apporter une description des procédures à suivre et des actions à effectuer sur les composantes techniques. Au niveau du graphique, cela peut être distingué par la présence de l'objet illustratif représentant une flèche (ou une main). En recherchant cette description dans le texte du document faisant référence au graphique, nous distinguons des informations indiquant :

- La procédure générale décrite par le graphique,
- le cas d'application de cette procédure,
- son étape d'exécution,
- les actions intermédiaires à appliquer sur les composantes techniques,
- l'ordre d'exécution de ces actions.

Ce côté opératoire, représenté dans les graphiques techniques, est illustré dans l'exemple de la Figure 49. On y retrouve les informations citées ci-dessus, soient:

- la procédure générale :
 - « *Élimination des incidents papiers* »
- Le cas d'application de cette procédure:
 - « *incident papier dans le module four* »
- Les actions intermédiaires sur les composantes techniques:
 - *action sur le « panneau arrière » : « ouvrir »*
 - *action sur le « module recto verso » : « retirer »*
 - *etc.*
- L'ordre d'exécution de ces actions :
 - *ordre d'exécution de l'action « ouvrir sur le « panneau arrière » : « 2 »*
 - *ordre d'exécution de l'action « retirer » sur le « module recto verso » : « 3 »*
 - *etc.*

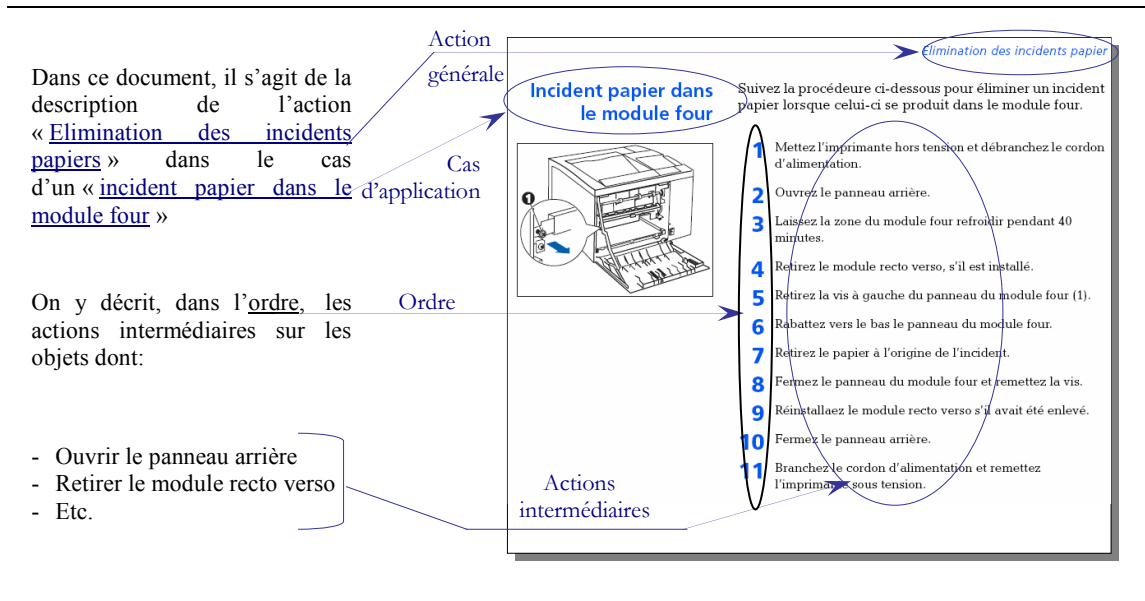


Figure 49 Exemple illustrant l'aspect opératoire du graphique

1.3. Impact de la mémorisation visuelle des graphiques

Comme cela a été mentionné, vue leur visualisation répétitive et la sémantique qui leur est rattachée, dans le contexte particulier de l'accès à la documentation techniques à usage professionnel, les graphiques subissent un encodage dont le résultat (qui est une approximation du graphique original) est stocké dans la mémoire visuelle de l'utilisateur. Nous nous intéressons à savoir quelle est la profondeur de cet encodage, c'est-à-dire, qu'est ce qui est encodé dans ce type de graphiques et quelle est son approximation. Cette information est intéressante dans le sens où elle permet de vérifier et/ou préciser, les aspects descriptifs du contenu visuel du graphique retenus précédemment (§1.1), sachant que notre contexte de recherche se base sur la mémoire des documents.

En effet, puisque la mémoire visuelle ne permet de stocker qu'une partie des informations définissant le graphique visualisé et ce de manière approximative, une partie des informations ne sera pas encodée (voir Figure 50), Ainsi, tous les objets contenus dans un graphique ne sont pas à retenir dans sa description et les formes/tailles/positions utilisées pour les caractériser ne sont pas précises, mais juste approximatives. Cette restriction des objets à retenir dans la description visuelle d'un graphique est développée ci-après.

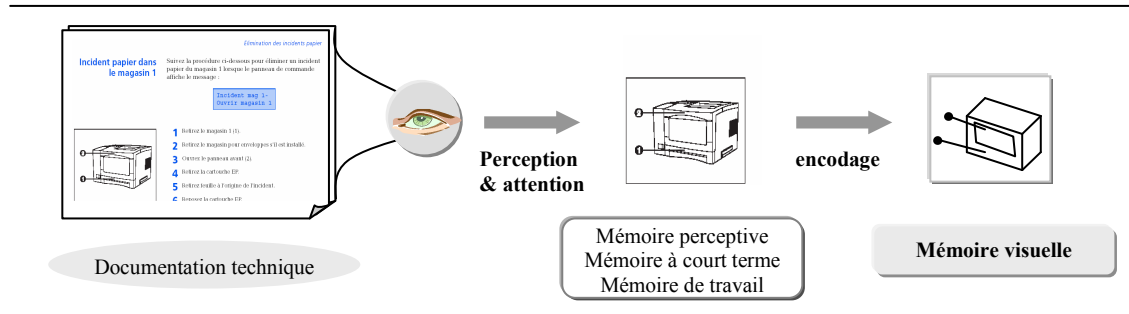


Figure 50 Exemple d'encodage possible d'un graphique visualisé

Objets du graphique retenus lors de l'encodage (i.e. mémorisés)

D'après le fonctionnement de la mémoire humaine, lorsqu'un graphique est mémorisé, cela ne veut pas dire que tout est stocké dans la mémoire mais que quelque chose dans ce graphique l'est, permettant ainsi au graphique de paraître familier. Ainsi certains éléments d'un graphique laissent une trace sur la mémoire visuelle d'un utilisateur, autrement dit, sont encodés. Nous posons l'hypothèse que ces objets retenus sont les objets illustratifs, les objets de grandes tailles et les objets importants mentionnés dans le texte.

- **Les objets illustratifs** tels que les zooms, les flèches, etc. En effet, ces objets sont utilisés par les rédacteurs techniques afin de mettre l'accent sur un objet du graphique ou sur une action à effectuer. Ils attirent l'attention de l'utilisateur et sont par conséquent facilement mémorisés.
- **Les objets de grandes tailles.** En effet, [Martinet,04] s'est intéressé à la facilité, pour un observateur, de percevoir les différentes régions d'une image. A l'issue d'évaluations confrontant des personnes et des images [Martinet,04] a vérifié que : *Un objet de grande taille est plus représentatif du contenu d'une image qu'un objet de petite taille.* Cela laisse à penser que l'utilisateur sera plus attentif aux objets de grandes tailles, ce qui induit le fait qu'un objet de grande taille est plus facilement mémorisé qu'un objet de petite taille.
- **Les objets importants mentionnés dans le texte.** Dans ce cas, ce n'est pas la géométrie de l'objet qui a un impact sur la mémoire de l'utilisateur, mais plutôt l'importance de cet objet dans le message que le graphique veut transmettre à l'utilisateur. Il s'agit, dans la plupart des cas, des objets qui entrent en jeu dans la réalisation de la tâche décrite par le graphique.

1.4. Récapitulatif

L'étude des graphiques techniques, dans leur contexte d'apparition et d'utilisation, nous a permis d'identifier les différents aspects à prendre en compte pour les décrire de façon riche. Nous en avons déduit différents niveaux de description :

- un niveau structurel : le graphique est un objet structuré dont les objets intéressants représentent soit des composantes techniques, soit des objets illustratifs,

- un niveau descriptif : le graphique est une description des objets qu'il contient ainsi que de leurs propriétés
- un niveau opératoire : le graphique est une description des procédures et actions à effectuer afin de réaliser une tâche technique,
- un niveau visuel : le graphique est perçu comme un ensemble de formes géométriques disposées d'une manière particulière, représentant leur encodage en mémoire visuelle de l'utilisateur.

Cette description multi-niveaux des graphiques techniques, à laquelle nous avons abouti, est caractéristique du modèle EMIR² [Mechkour,95]. Nous nous sommes donc inspirés de ce modèle, qui a déjà fait ses preuves dans le domaine, afin de représenter exhaustivement les graphiques techniques. La description de l'aboutissement, à partir de ce modèle, à notre modèle de représentation est donnée en annexe B et est décrit dans [Kefi,02]. Dans ce qui suit, nous décrivons, de façon informelle, le modèle de représentation obtenu, spécifique à notre cadre applicatif (voir aussi [kefi,03]).

2. Une représentation multi-vues du graphique technique

Nous considérons le graphique comme une entité structurée pouvant être considéré selon différentes interprétations ou vues :

- une vue **physique** décrivant les caractéristiques générales du graphique,
- une vue **structurelle** définissant les objets graphiques (OG) contenus dans le graphique,
- une vue **symbolique** définissant les OG (imprimante, panneau avant, écran d'affichage, etc.) et leurs propriétés,
- une vue **opératoire** représentant les procédures et actions à appliquer sur les OG.
- une vue **mémoire visuelle** décrivant les formes géométriques, tailles et positions relatives des OG ayant un impact sur la mémoire visuelle de l'utilisateur professionnel.

Nous reprenons, ici, ces différentes vues et les décrivons plus en détail.

2.1. Vue physique

C'est la vue la plus élémentaire d'un graphique. Elle est décrite par la donnée des caractéristiques générales du graphique, telles que ses dimensions, la résolution, la matrice de pixels. Le graphique peut être visualisé sur écran, sauvé dans un fichier et peut subir des transformations à l'aide de fonctions de traitement d'images tels que le zoom, la rotation, etc.

2.2. Vue structurelle

Elle représente la décomposition d'un graphique en objets graphiques (OG). Elle permet d'exprimer la décomposition d'objets graphiques en objets graphiques composants. A cette vue correspond alors une première relation : la relation de composition « *contient* ».

D'autres relations de cette vue, relient des objets graphiques illustratifs et des objets graphiques représentant une composante technique :

- La relation « *équivalent* » relie un illustratif représentant un zoom et l'objet zoomé.
- La relation « *numérote* » relie un illustratif représentant une numération et l'objet numéroté.
- La relation « *action_sur* » relie un illustratif représentant une flèche ou une main et l'objet qu'il vise.

La vue structurelle est, de ce fait, représentée par :

- un ensemble d'objets graphiques. Ces objets sont soit représentatifs d'une composante matérielle (tels qu'un panneau avant ou une imprimante), soit des illustratifs (tels que le zoom ou la flèche),
- un ensemble de relations entre ces objets graphiques : il s'agit de relations de composition (*contient*), d'équivalence (*équivalent*), d'énumération (*numérote*) et d'action (*action_sur*).

Exemple

Dans la Figure 51, le graphique est composé des objets graphiques OG1 et OG2. Le premier est lui-même composé des objets graphiques OG11 et OG12, alors que le deuxième contient les objets OG21 et OG22.

Une relation d'équivalence relie l'objet OG21 à l'objet OG11, et une relation d'action relie l'objet OG21 à l'objet OG22.

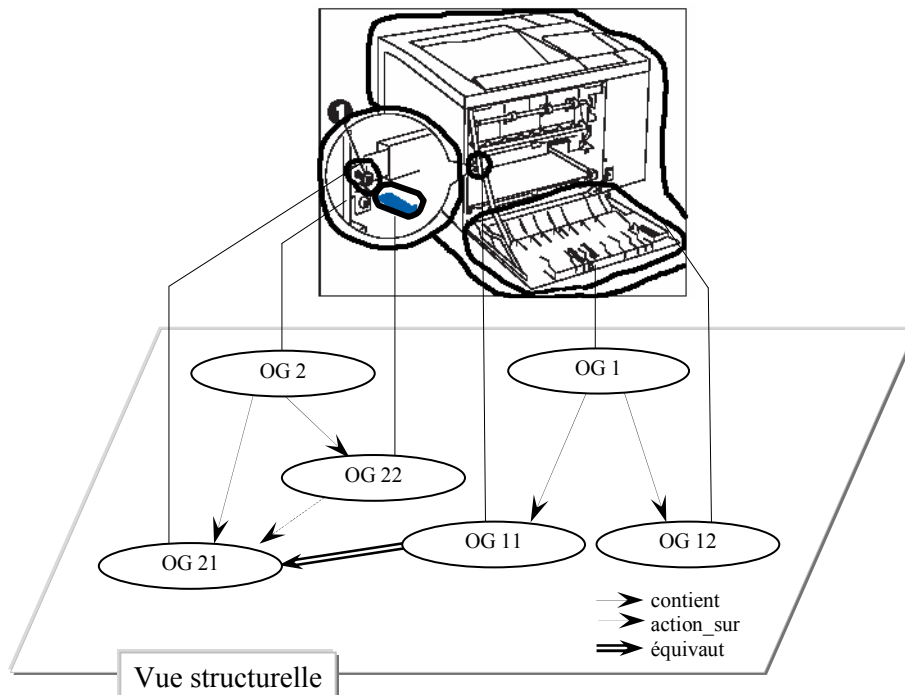


Figure 51 Exemple de vue structurelle d'un graphique

2.3. Vue symbolique

Cette vue correspond à la représentation du contenu descriptif d'un graphique.

Dans cette vue, sont représentées les propriétés du graphique et des objets qu'il contient : à chaque objet graphique (OG) est associé un objet symbolique (composant ou illustratif) défini dans cette vue, auquel peuvent correspondre des caractéristiques le décrivant.

Dans l'exemple de la Figure 52, à l'objet graphique OG11 de la vue structurelle est associé un objet symbolique : il s'agit d'un « *Composant* » décrit par le terme « Ecran d'affichage » et auquel est rattaché une « *Description* » : « 2 lignes de 16 caractères ».

Nous considérons deux niveaux de description dans la vue symbolique :

- des descriptions relatives au graphique dans sa globalité,
- des descriptions relatives aux objets pertinents du graphique, autrement dit, aux OG.

Nous considérons, dans le premier niveau les informations suivantes :

- le type de la *Machine* représentée ou dont une composante est décrite: exp. imprimante, scanner...
- son *Constructeur* : exp. HP, Epson...
- son *Modèle* : exp. N17, Stylus 500...

Dans le deuxième niveau, nous faisons la distinction entre deux types d'objets. D'un côté, nous avons les *composants* (représentant un objet réel tel qu'une cartouche) et d'un autre côté nous avons les *illustratifs* (représentant un zoom, une énumération, etc.)

- chaque *Composant* est définie par :

- son nom ou sa désignation : exp. câble, panneau, cartouche...
- éventuellement, sa *Description* : exp. ancien, rouge...

- et chaque *Illustratif* est défini par :

- son type : exp. zoom, flèche...
- éventuellement un *Caractère* (lettre ou numéro) qui désigne une composante dans le graphique.

Des relations symboliques peuvent relier deux objets symboliques.

Remarques

A un composant peuvent être associées plusieurs valeurs d'une même propriété. Il s'agit du cas d'objets auxquels correspondraient plusieurs descriptions.

Différents objets graphiques peuvent avoir un seul objet symbolique qui leur correspond. Il s'agit du cas d'objets graphiques équivalents auxquels nous associons un même objet symbolique afin d'éviter la redondance (composant zoomé par exemple).

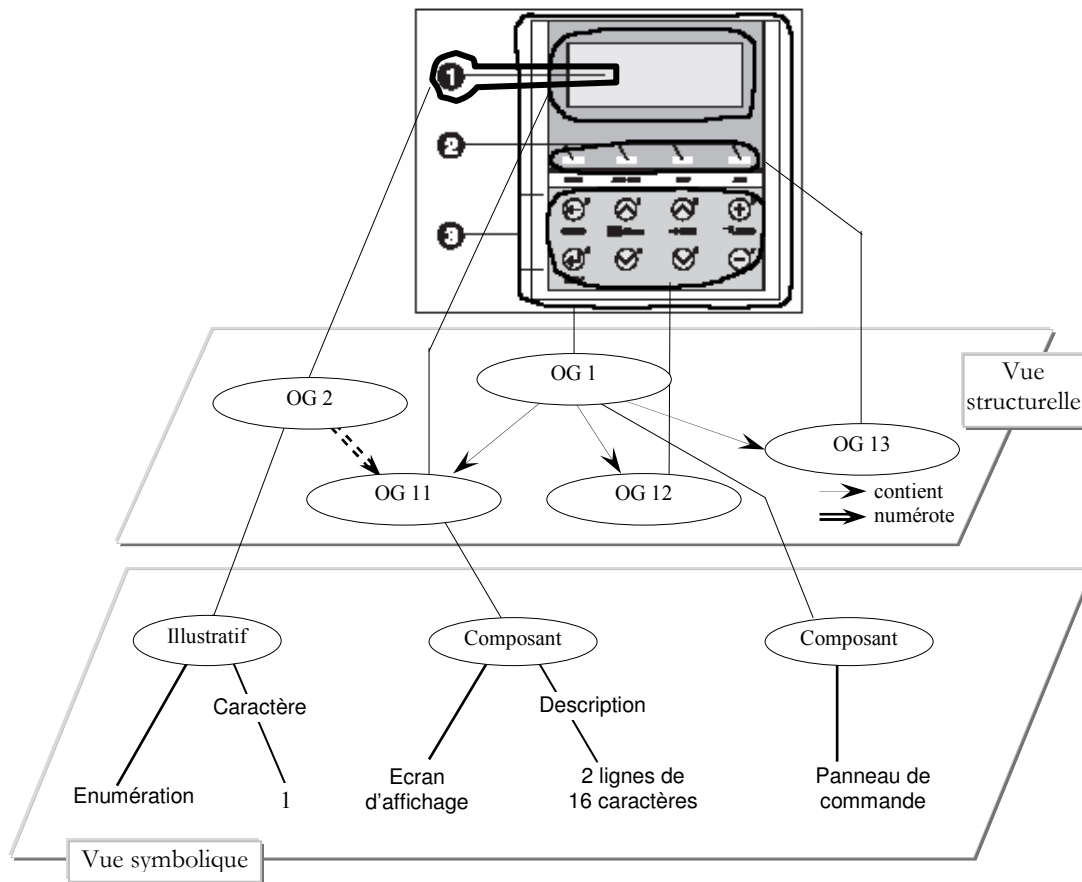


Figure 52 Exemple de vue descriptive d'un graphique

2.4. Vue opératoire

La perception du contenu des graphiques par un utilisateur spécialisé est fonction non seulement de la description des objets qu'il contient, mais aussi de la description de son aspect opératoire. Ainsi, un graphique contenant un objet correspondant à un « magasin », est incomplètement décrit si on ignore dans sa représentation l'action qu'on applique sur le « magasin », soit le « chargement de papier », par exemple.

C'est la vue opératoire qui correspond à la représentation et la description des actions à appliquer sur les objets graphiques. Elle est définie par des objets opératoires associés aux objets graphiques de la vue structurelle, mis à part les objets illustratifs autres que la flèche et la main, ainsi que des relations entre objets opératoires.

À un objet graphique peuvent être associés plusieurs objets opératoires. Ceci est vrai lorsque plusieurs actions doivent être exécutées sur ce même objet. Par contre, dans le cas de l'existence d'un objet illustratif représentant une flèche ou une main dans la vue structurelle, cet objet et celui qu'il vise sont tous deux représentés par un seul objet opératoire dans la vue

correspondante pour éviter la redondance. L'exemple de la Figure 53 illustre ce cas : aux objets graphiques OG2 et OG12 correspond le même objet opératoire relatif à l'action « Tirer ».

Comme pour la vue symbolique, nous considérons deux niveaux informatifs dans la vue opératoire :

- des informations descriptions relatives au graphique dans sa globalité,
- des informations relatives aux objets pertinents du graphique.

Nous considérons, dans le premier niveau les informations suivantes :

- la **Procédure** décrite entre autre par le graphique: exp. « Résolution de problèmes papier »
- le **Cas** particulier d'application de cette procédure : exp. « Bourrage papier dans le bac avant »
- éventuellement, l'**Etape** d'exécution de la liste des actions décrites par le graphique.

Dans le deuxième niveau, nous considérons les actions à appliquer aux composantes du graphique. Chaque **Action** est définie par :

- son nom : il s'agit du verbe correspondant à l'action à appliquer sur la composante adéquate. Exp. pousser, visser...
- éventuellement, l'**Etape** d'exécution de cette action

Les relations liant deux objets opératoires sont des relations d'ordre. Il s'agit des relations « avant », « après » et « simultané ». Ainsi, dans l'exemple de la Figure 53, l'action « faire glisser » doit être exécutée avant l'action « tirer », les objets opératoires relatifs à ces deux actions sont alors liés par la relation « avant ».

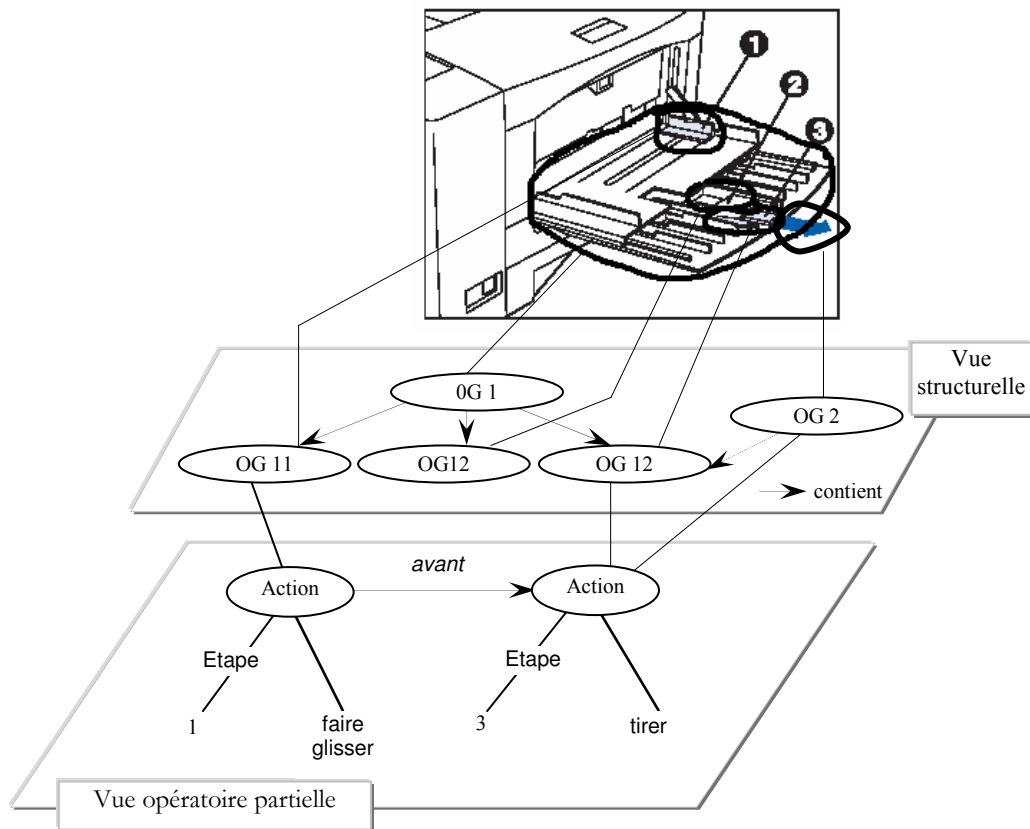


Figure 53 Exemple de vue opératoire d'un graphique

2.5. Vue mémoire visuelle

Cette vue permet de représenter l'encodage du graphique se produisant lors de sa mémorisation visuelle. Autrement dit, de représenter approximativement les formes géométriques des objets graphiques susceptibles d'avoir été mémorisés par l'utilisateur, leurs tailles et leurs dispositions les unes par rapport aux autres.

Elle comporte donc des informations géométriques décrivant les objets visuels (OV) et des relations visuelles liant ces objets. Ces relations peuvent être :

- des relations décrivant les positions relatives des objets, comme par exemple la relation « *touche* »,
- une relation permettant de décrire la décomposition des objets tridimensionnels en objets représentant leurs différentes faces : il s'agit de la relation « *face* » (voir Figure 54).

La forme des objets visuels est représentée par une combinaison d'éléments géométriques de base.

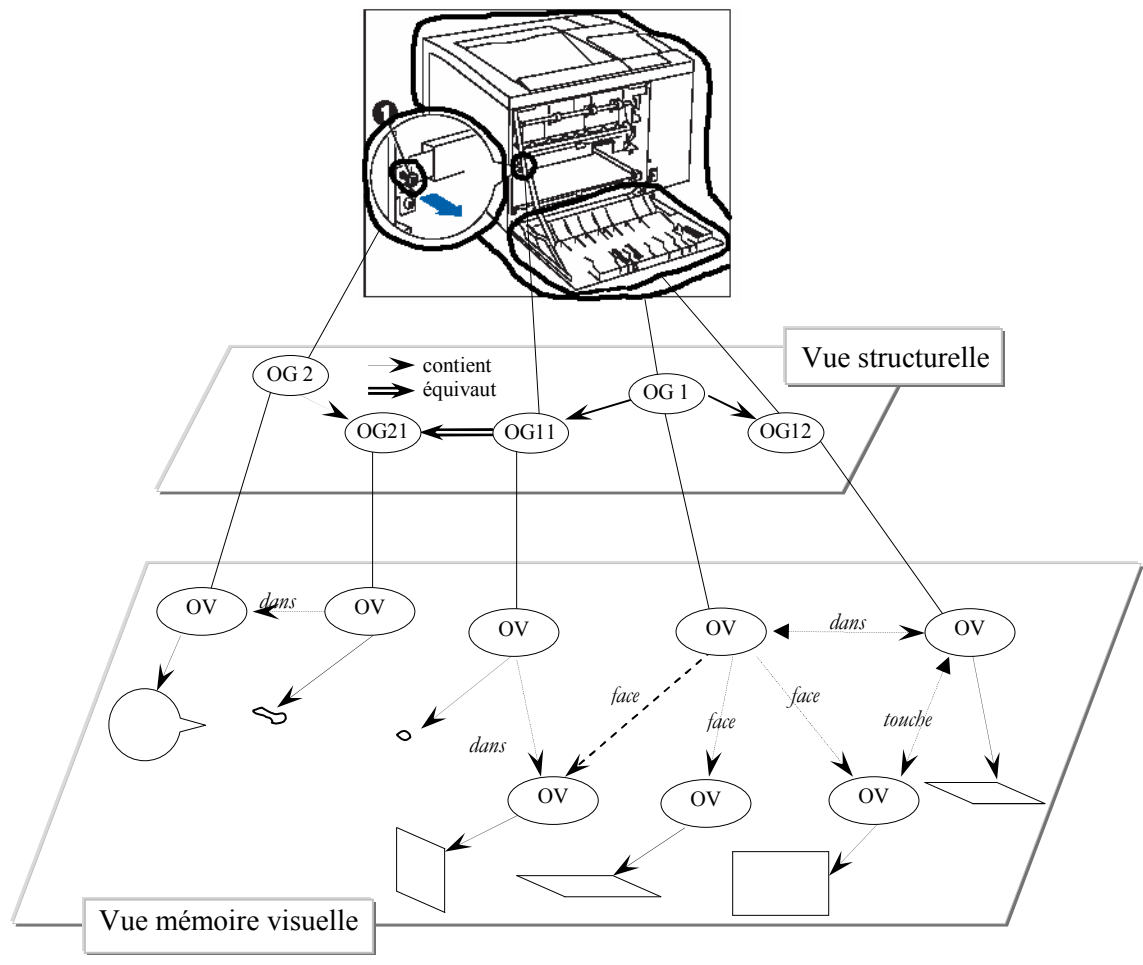


Figure 54 Exemple de vue mémoire visuelle d'un graphique

3. Conclusion

Dans ce chapitre, nous avons identifié les différentes informations nécessaires à la description exhaustive des graphiques contenus dans la documentation technique, informations qui provient soit du graphique en lui-même, soit du texte qui lui est associé dans le document. Nous avons proposé de représenter ces informations, à travers un ensemble de vues, comme cela a été fait dans le modèle EMIR², afin de représenter de manière structurée un maximum d'informations relatives au contenu des graphiques, en organisant ces informations selon différents points de vues.

Dans le chapitre suivant (chapitre VI), nous proposons une description détaillée d'un modèle de recherche de graphiques techniques qui se base sur la représentation du contenu des graphiques proposée ci-dessus (un langage complexe) et sur des critères d'obligation/option et certitude/incertitude au niveau de la formulation du besoin.

Chapitre VI. Modèle de recherche de graphiques techniques

A partir de la représentation multi-vues du contenu des graphiques proposée dans le chapitre précédent, nous proposons, dans ce chapitre, un modèle de recherche de ces graphiques qui se base sur cette représentation et qui instancie le modèle général proposé dans la partie 2 de ce manuscrit.

Rappelons que ce modèle :

- est fondé sur un langage complexe, ce qui est en adéquation avec le type de représentation des graphiques que nous avons proposée dans le chapitre précédent.
- permet l'utilisation multiple d'une même entité dans les documents indexés et les requêtes. Cette caractéristique est utile dans le cas de la représentation des graphiques, surtout en raison de leur nature visuelle. Par exemple, il est nécessaire de pouvoir décrire correctement un graphique représenté par trois cubes et deux zooms (trois fois l'entité cube et deux fois l'entité zoom)
- est fondé sur des critères d'obligation/option et de certitude/incertitude qui sont utiles dans le cadre d'une recherche fondée sur la description du graphique 'idéal', graphique dont l'utilisateur se souvient plus ou moins vaguement.

Nous détaillons donc, ici, le modèle de recherche de graphiques techniques, en définissant :

- le *vocabulaire* \mathcal{V}_{GRIM} . (GRIM pour **G**raphics **I**ndexing **M**odel) : nous le définissons pour les différentes vues (structurelle, symbolique, opératoire et mémoire visuelle). Cela inclut la redéfinition des entités, des relations et des identifiants d'entités entrant en jeu dans chacune de ces vues,
- l'*indexation des graphiques*: elle est fondée sur le vocabulaire. Nous en décrivons la structure globale,
- la *formulation de la requête* : nous décrivons ici trois types de langages d'indexation permettant de choisir un mode de représentation du besoin: purement textuel, purement graphique ou mixte,
- la *fonction de correspondance* : elle doit être en adéquation avec les représentations définies et les différents modes de recherche.

1. Vocabulaire

Nous définissons ici le vocabulaire \mathcal{V}_{GRIM} permettant la description des graphiques techniques. Rappelons que le vocabulaire, tel que nous l'avons défini dans notre modèle, est formé par un ensemble d'entités que nous appelons, ici, \mathcal{T}_{GRIM} et d'un ensemble de relations, que nous

appelons, ici, \mathcal{R}_{GRIM} . Rappelons aussi qu'un ensemble d'identifiants, que nous notons Δ_{GRIM} , est rattaché à l'ensemble des entités.

Comme cela a été défini précédemment, la description des graphiques techniques repose sur différentes vues. Nous définissons donc pour chacune de ces vues le vocabulaire qui lui est associé. L'ensemble de ces vocabulaires constitue \mathcal{V}_{GRIM} .

Dans la suite, nous regroupons les vocabulaires des vues symbolique et opératoire en un seul vocabulaire, car ils sont définis de façon similaire.

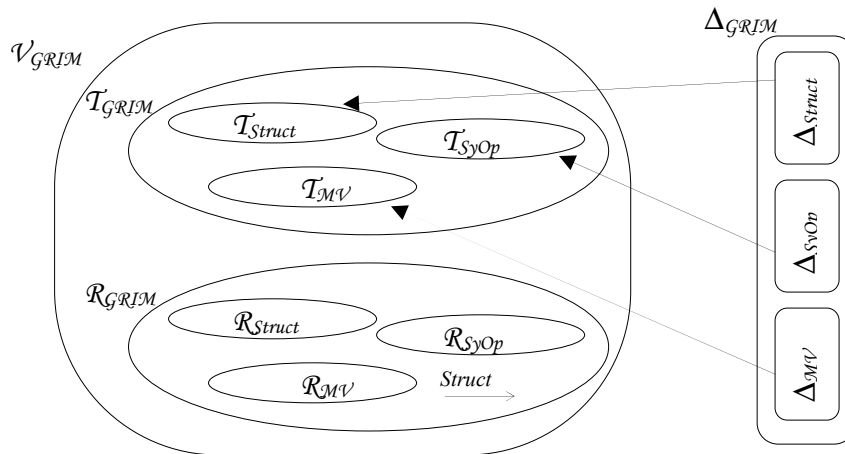


Figure 55 Le vocabulaire \mathcal{V}_{GRIM}

1.1. Vue structurelle

Elle représente la décomposition d'un graphique en objets graphiques ainsi que leurs inter-relations.

a. L'ensemble des entités

Une entité de la vue structurelle est un objet graphique, que nous notons OG. Nous définissons donc l'ensemble des entités associées à la vue structurelle:

$$\mathcal{T}_{Struct} = \{OG\} \cup \{\tau_0\}$$

b. L'ensemble des relations

Les relations possibles entre OG sont des relations de composition (*contient*), d'équivalence (*équivalent*), d'énumération (*numérote*) ou d'action (*action_sur*). L'ensemble des relations associées à la vue structurelle est donc:

$$\mathcal{R}_{Struct} = \{\textit{contient}, \textit{équivalent}, \textit{numérote}, \textit{action_sur}\} \cup \{\rho_0\}$$

c. L'ensemble des identifiants

Un ensemble d'identifiant est rattaché à $\mathcal{T}_{\text{Struct}}$. Nous le définissons ainsi :

$$\Delta_{\text{Struct}} = \{\delta_{\text{Struct}1}, \delta_{\text{Struct}2}, \dots\}$$

d. Les fonctions de proximité

Aucune fonction de proximité n'est rattachée à la vue structurelle.

1.2. Vues symbolique et opératoire

Les vues symbolique et opératoire sont définies par un ensemble d'entités inter-relies.

a. L'ensemble des entités

Une entité de la vue symbolique ou de la vue opératoire est un terme étiqueté. Autrement dit, il s'agit d'un terme auquel est associée une étiquette permettant de le qualifier. Par exemple, le terme 'imprimante' peut être étiqueté par :

- l'étiquette 'Machine', afin de désigner l'appareil auquel sont rattachées les composantes schématisées dans le graphique (il se peut, par exemple, que seul le panneau d'affichage de l'imprimante soit représenté dans le graphique)
- l'étiquette 'Composante', afin de désigner la composante matérielle imprimante lorsqu'elle est schématisée dans le graphique.

Nous notons $\mathcal{E}_{\text{SyOp}}$ l'ensemble des étiquettes e_i définies dans ces deux vues et $\mathcal{T}_{\text{SyOp}}$ l'ensemble des termes t_i de ces vues. Le premier est un ensemble prédéfini de cardinalité $\mathcal{N}_{\mathcal{E}_{\text{SyOp}}}=12$, alors que le deuxième, de cardinalité $\mathcal{N}_{\mathcal{T}_{\text{SyOp}}}$, correspond à une combinaison d'un ensemble de termes prédéfinis et d'un ensemble de termes extraits du commentaire du graphique :

- $\mathcal{E}_{\text{SyOp}} = \{Machine, Constructeur, Modèle, Composant, Illustratif, Description, Caractère, Procédure, Cas, Action, Etape, e_0\}$ (e_0 est une étiquette neutre)
- $\mathcal{T}_{\text{SyOp}} = \mathcal{T}_{\text{Illustr}} \cup \mathcal{T}_{\text{Machines}} \cup \mathcal{T}_{\text{Constructeurs}} \cup \mathcal{T}_{\text{Modèles}} \cup \mathcal{T}_{\text{Procédures}} \cup \mathcal{T}_{\text{mots}}$

avec

- ✓ $\mathcal{T}_{\text{Illustr}} = \{\text{zoom, énumération, flèche, main}\}$
- ✓ $\mathcal{T}_{\text{Machines}}$: liste des machines concernées par les documents,
- ✓ $\mathcal{T}_{\text{Constructeurs}}$: liste des constructeurs répertoriés,
- ✓ $\mathcal{T}_{\text{Modèles}}$: liste des modèles de machines répertoriés,
- ✓ $\mathcal{T}_{\text{Procédures}}$: liste des procédures prédéfinies,
- ✓ $\mathcal{T}_{\text{mots}}$: {chaînes de caractères définissant les noms des composantes matérielles, leurs descriptions possibles, les cas d'application des procédures à suivre, les actions à effectuer et leurs étapes}

L'ensemble $\mathcal{T}_{\text{SyOp}}$ des entités des vues symbolique et opératoire est un ensemble de couples étiquette-terme que nous définissons ainsi:

$$\begin{aligned} \mathcal{T}_{\text{Sym}} = & \{ \textit{Illustratif} \} \times \mathcal{T}_{\text{Illustr}} \\ & \cup \{ \textit{Machine} \} \times \mathcal{T}_{\text{Machines}} \\ & \cup \{ \textit{Constructeur} \} \times \mathcal{T}_{\text{Constructeurs}} \\ & \cup \{ \textit{Modèle} \} \times \mathcal{T}_{\text{Modèles}} \\ & \cup \{ \textit{Procédure} \} \times \mathcal{T}_{\text{Procédures}} \\ & \cup \{ \textit{Composant, Description, Caractère, Cas, Action, Etape} \} \times \mathcal{T}_{\text{mots}} \\ & \cup \{ \tau_0 \} \end{aligned}$$

Nous définissons, ici, les fonctions $F_{t_{\text{SymOp}}}$ et $F_{e_{\text{SymOp}}}$ qui, pour chaque entité symbolique, spécifient respectivement le terme et l'étiquette qui la définissent.

$$\begin{aligned} F_{t_{\text{SymOp}}} : \mathcal{T}_{\text{SymOp}} &\rightarrow \mathcal{T}_{\text{SymOp}} & F_{e_{\text{SymOp}}} : \mathcal{T}_{\text{SymOp}} &\rightarrow \mathcal{E}_{\text{SymOp}} \\ \tau=(e, t) &\rightarrow t & \tau=(e, t) &\rightarrow e \end{aligned}$$

b. L'ensemble des relations

Les relations entre les entités des vue symbolique et opératoire sont, soit des relations de structuration (ou d'association), soit des relations d'ordre entre les actions spécifiques à la vue opératoire.

Nous choisissons de nommer les relations de structuration afin de les différencier :

- *ComDesc* : relation permettant de rattacher une entité dont l'étiquette est *Description* à l'entité correspondante et dont l'étiquette est *Composant*.
- *IllusDesc*: relation permettant de rattacher une entité dont l'étiquette est *Caractère* à l'entité correspondante et dont l'étiquette est *Illustratif*.
- *ActEtap*: relation permettant de rattacher une entité dont l'étiquette est *Etape* à l'entité correspondante et dont l'étiquette est *Action*.

Et nous notons les relations d'ordre : *Avant, Après et Simultané*. Ces relations permettent de spécifier l'ordre selon lequel deux actions décrites par le graphique doivent être effectuées (soit avant, soit après soit en même temps).

$$\mathcal{R}_{\text{SymOp}} = \{ \textit{ComDesc, IllusDesc, ActEtap, Avant, Après, Simultané} \} \cup \{ \rho_0 \}$$

c. L'ensemble des identifiants

Un ensemble d'identifiant est rattaché à $\mathcal{T}_{\text{SymOp}}$. Nous le définissons ainsi :

$$\Delta_{\text{SymOp}} = \{ \delta_{\text{SymOp}1}, \delta_{\text{SymOp}2}, \dots \}$$

d. Les fonctions de proximité

Rappelons que dans le modèle que nous avons proposé, une relation de proximité \mathcal{VD} existe entre les entités du vocabulaire. Dans le cadre des vues symbolique et opératoire, nous la notons $\textit{Proche}_{\text{OpSy}}$.

$$\text{Proche}_{\mathcal{T}_{\text{SyOp}}}: \mathcal{T}_{\text{SyOp}} \rightarrow \mathcal{P}(\mathcal{T}_{\text{SyOp}})$$

$$\tau_i \rightarrow \{\tau_k \text{ tels que (i) } F_{e_{\text{SyOp}}}(\tau_k) = F_{e_{\text{SyOp}}}(\tau_i) \text{ ou } F_{e_{\text{SyOp}}}(\tau_k) = \epsilon_0 \text{ et}$$

$$(ii) F_{t_{\text{SyOp}}}(\tau_k) \in \text{Proche}_{t_{\text{SyOp}}}(F_{t_{\text{SyOp}}}(\tau_i)) \}$$

avec $\text{Proche}_{t_{\text{SyOp}}}$ définit, pour chaque terme, l'ensemble de ses proches :

$$\text{Proche}_{t_{\text{SyOp}}}: \mathcal{T}_{\text{SyOp}} \rightarrow \mathcal{P}(\mathcal{T}_{\text{SyOp}})$$

$$t \rightarrow \text{Proche}_{t_{\text{SyOp}}}(t)$$

Cela signifie qu'une entité τ_k est proche d'une entité τ_i si et seulement si ces deux entités ont la même étiquette et que le terme associé à l'entité τ_k est proche de celui associé à τ_i .

Par exemple, l'entité (*Composante*, bac récepteur) est considérée comme proche de (*Composante*, bac de sortie) si le terme 'bac de sortie' est considéré comme proche de 'bac récepteur'.

D'un autre côté, il existe une relation de proximité entre les relations de $\mathcal{R}_{\text{SyOp}}$, notée $\text{Proche}_{\mathcal{R}_{\text{SyOp}}}$, et définie ainsi :

- $\text{Proche}_{\mathcal{R}_{\text{SyOp}}}(\text{Avant}) = \{\text{Simultané}\}$
- $\text{Proche}_{\mathcal{R}_{\text{SyOp}}}(\text{Après}) = \{\text{Simultané}\}$
- $\text{Proche}_{\mathcal{R}_{\text{SyOp}}}(\text{Simultané}) = \{\text{Avant}, \text{Après}\}$

1.3. Vue mémoire visuelle

La vue mémoire visuelle est définie par un ensemble d'objets visuels inter-reliés par des relations de position ou de composition.

a. L'ensemble des entités

Une entité de la vue mémoire visuelle est un objet visuel caractérisé par une forme géométrique, une position spatiale, une taille et une direction. Ces caractéristiques sont définies par les ensembles :

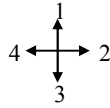
$$- \text{Formes} = \{ \text{cylindre}, \text{cube}, \text{carré}, \text{losange}, \text{triangle}, \text{cercle}, \text{flèche}, \text{double flèche}, \text{rectangle}, \text{cercle avec croix}, \text{cercle avec x}, \text{cylindre empilé}, \text{cylindre empilé}, \text{cylindre empilé} \}$$

$$- \text{Tailles} = \{\text{très_petit}, \text{petit}, \text{moyen}, \text{grand}, \text{très_grand}\}$$

- $\text{Positions} = \mathcal{P}(\{1, 2, 3, 4, 5, 6, 7, 8, 9\})$ correspondant à la zone du graphique où la forme se situe, sachant que le graphique se inscrit dans le rectangle quadrillé :

1	2	3
4	5	6
7	8	9

- $\text{Directions} = \{1, 2, 3, 4\}$ correspondant aux quatre directions :



Une entité de la vue mémoire visuelle est donc un ensemble de quatre termes respectivement étiquetés par les étiquettes « *Forme* », « *Taille* », « *Position* » et « *Direction* ».

Nous notons \mathcal{E}_{MV} l'ensemble des étiquettes e_i définies dans la vue Mémoire Visuelle et \mathcal{T}_{MV} l'ensemble des termes t_i définis dans cette vue. Le premier est un ensemble prédéfini de cardinalité $\mathcal{N}_{\mathcal{E}_{MV}}=4$, et le deuxième, de cardinalité $\mathcal{N}_{\mathcal{T}_{MV}}$, correspond à un ensemble de termes prédéfinis correspondant aux termes des ensembles *Formes*, *Tailles*, *Positions* et *Directions*.

- $\mathcal{E}_{MV} = \{Forme, Taille, Position, Direction\}$
- $\mathcal{T}_{MV} = Formes \cup Tailles \cup Positions \cup Directions$

L'ensemble \mathcal{T}_{MV} des entités de la vue Mémoire Visuelle est un ensemble de quatre couples étiquette-terme que nous définissons ainsi:

$$\mathcal{T}_{MV} = \{ \{Forme\} \times Formes, \{Taille\} \times Tailles, \{Position\} \times \mathcal{P}(Positions), \{Direction\} \times Directions \} \cup \{\tau_0\}$$

Nous définissons, ici, la fonction $\mathcal{F}_{t_{MV}}$ qui pour chaque entité visuelle et étiquette (ou caractéristique) spécifie le terme correspondant à cette étiquette :

$$\begin{aligned} \mathcal{F}_{t_{MV}} : \mathcal{T}_{MV} \times \mathcal{E}_{MV} &\rightarrow \mathcal{T}_{MV} \\ (\tau, e) &\rightarrow \mathcal{F}_{t_{MV}}(\tau, e) \end{aligned}$$

b. L'ensemble des relations

Les relations entre les entités de la vue mémoire visuelle sont des relations de positionnement notés : *Touche*, *Coupe*, *Dans*, *A_dte*, *A_gche*, *Dessus*, *Dessous* et *Face*.

$$\mathcal{R}_{MV} = \{Touche, Coupe, Dans, A_dte, A_gche, Dessus, Dessous, Face\} \cup \{\rho_0\}$$

c. L'ensemble des identifiants

Un ensemble d'identifiant est rattaché à \mathcal{T}_{MV} . Nous le définissons ainsi :

$$\Delta_{\text{SyOp}} = \{\delta_{MV1}, \delta_{MV2}, \dots\}$$

d. Les fonctions de proximité

Comme nous l'avons fait dans les vues symbolique et opératoire, nous proposons de noter $\mathcal{Proche}_{\mathcal{T}_{MV}}$ la relation de proximité existant entre les entités et nous la spécifions.

Nous définissons, tout d'abord, la fonction symétrique $\mathcal{Proche}_{\mathcal{T}_{MV}}$ qui pour chaque terme, définit l'ensemble de ses proches :

$$\begin{aligned} \mathcal{Proche}_{\mathcal{T}_{MV}} : \mathcal{T}_{MV} &\rightarrow \mathcal{P}(\mathcal{T}_{MV}) \\ t &\rightarrow \mathcal{Proche}_{\mathcal{T}_{MV}}(t) \text{ tel que :} \end{aligned}$$

$$\begin{aligned}
 \text{Prochet}_{MV}(\text{carré}) &= \{\text{cylindre}\} \\
 \text{Prochet}_{MV}(\text{cylindre}) &= \{\text{carré}\} \\
 \text{Prochet}_{MV}(\text{carré}) &= \{\text{carré}\} \\
 \text{Prochet}_{MV}(\text{carré}) &= \{\text{carré}, \text{cercle}, \text{triangle}\} \\
 \text{Prochet}_{MV}(\text{cercle}) &= \{\text{cercle}\} \\
 \text{Prochet}_{MV}(\text{triangle}) &= \{\text{triangle}\} \\
 \text{Prochet}_{MV}(\text{carré}) &= \emptyset \\
 \text{Prochet}_{MV}(\text{très_petit}) &= \{\text{petit}, \text{moyen}\} \\
 \text{Prochet}_{MV}(\text{très_grand}) &= \{\text{grand}, \text{moyen}\} \\
 \forall \tau_k \in \text{Directions}, \text{Prochet}_{MV}(\tau_k) &\in \text{Directions} \\
 \forall \tau_k \in \text{Directions}, \text{Prochet}_{MV}(\tau_k) &\in \text{Directions}
 \end{aligned}$$

Ensuite, nous définissons pour chaque terme étiqueté caractérisant une entité visuelle, l'ensemble de ses proches, de la même façon que nous l'avons fait pour les entités des vues symbolique et opératoire :

$$\begin{aligned}
 \text{Proche}_{eMV}: \mathcal{E}_{MV} \times \mathcal{T}_{MV} &\rightarrow \mathcal{P}(\mathcal{E}_{MV} \times \mathcal{T}_{MV}) \\
 (e, t_i) &\rightarrow \{(e, t_k) \text{ tels que } t_k \in \text{Prochet}_{MV}(t_i)\}
 \end{aligned}$$

A partir de cette fonction, il nous est possible de définir la fonction $\text{Proche}_{T_{MV}}$ entre les entités visuelles, ainsi :

$$\begin{aligned}
 \text{Proche}_{T_{MV}}: \mathcal{T}_{MV} &\rightarrow \mathcal{P}(\mathcal{T}_{MV}) \\
 \tau_i &\rightarrow \{\tau_k \text{ tels que } \forall e \in \mathcal{E}_{MV}, (e, F_{t_{MV}}(\tau_k, e)) \in \text{Proche}_{eMV}(e, F_{t_{MV}}(\tau_i, e)) \text{ ou} \\
 &\quad (e, F_{t_{MV}}(\tau_k, e)) = (e, F_{t_{MV}}(\tau_i, e)) \}
 \end{aligned}$$

Cela signifie que si chacune des caractéristiques (termes étiquetés) d'une entité visuelle τ_k est proche ou égale à la même caractéristique de l'entité τ_i alors la première entité τ_k est considérée comme étant proche de la seconde τ_i .

Prenons, par exemple, le cas de deux entités visuelles ayant la même *forme*, la même *position* et la même *direction*, mais deux tailles différentes : supposons que la taille de la première est 'très petit' et que la taille de la deuxième est 'petit', alors nous pouvons dire que la deuxième entité visuelle est proche de la première, puisque la caractéristique (terme étiqueté) :

$$(Taille, \text{petit}) \in \text{Proche}_{eMV}(Taille, \text{très petit}).$$

D'un autre côté, il existe une relation de proximité symétrique entre les relations de \mathcal{R}_{MV} , notée $\text{Proche}_{\mathcal{R}_{MV}}$, et définie ainsi :

$$\begin{aligned}
 - \text{Proche}_{\mathcal{R}_{MV}}(\text{touche}) &= \{\text{coupe}, \text{dans}\} \\
 - \text{Proche}_{\mathcal{R}_{MV}}(\text{coupe}) &= \{\text{touche}, \text{dans}\} \\
 - \text{Proche}_{\mathcal{R}_{MV}}(\text{dans}) &= \{\text{coupe}, \text{touche}\}
 \end{aligned}$$

1.4. Vocabulaire global

Nous avons défini pour chacune des vues (structurale, symbolique et opératoire, mémoire visuelle) le vocabulaire qui lui est associé. L'ensemble de ces vocabulaires constitue le vocabulaire global \mathcal{V}_{GRIM} qui permet la description des graphiques techniques.

\mathcal{V}_{GRIM} est formé par un ensemble d'entités \mathcal{T}_{GRIM} et d'un ensemble de relations \mathcal{R}_{GRIM} . Un ensemble d'identifiants, Δ_{GRIM} , est rattaché à l'ensemble des entités.

Outre les éléments définis pour chaque vue, une relation d'association qui à chaque objet graphique (entité structurale) associe l'entité correspondante dans les vues symbolique, opératoire et mémoire visuelle, est nécessaire. Nous notons *Struct* cette relation.

Nous avons donc :

$$\begin{aligned}\mathcal{T}_{GRIM} &= \mathcal{T}_{Struct} \cup \mathcal{T}_{SyOp} \cup \mathcal{T}_{MV} \\ \Delta_{GRIM} &= \Delta_{Struct} \cup \Delta_{SyOp} \cup \Delta_{MV} \\ \mathcal{R}_{GRIM} &= \mathcal{R}_{Struct} \cup \mathcal{R}_{SyOp} \cup \mathcal{R}_{MV} \cup \{Struct\}\end{aligned}$$

Nous redéfinissons globalement la fonction qui permet de connaître quelle entité référence un identifiant, ainsi :

$$\begin{aligned}\mathcal{F}_{\delta\tau} : \Delta_{GRIM} &\rightarrow \mathcal{T}_{GRIM} \\ \delta_i &\rightarrow \tau_i = \mathcal{F}_{\delta\tau}(\delta_i)\end{aligned}$$

2. Indexation

Nous redéfinissons ici les vocabulaires d'indexation (ensemble des unités d'indexation) relatifs à chacune des vues selon GRIM, ainsi que le vocabulaire d'indexation global \mathcal{V}_{DGRIM} . Nous obtenons :

- l'ensemble des unités d'indexation de la vue structurale est:

$$\mathcal{V}_{DStruct} = \mathcal{P}(\Delta_{Struct} \times (\Delta_{Struct} \cup \{\delta_0\}) \times \mathcal{R}_{Struct})$$

- l'ensemble des unités d'indexation des vues symbolique et opératoire est:

$$\mathcal{V}_{DSyOp} = \mathcal{P}(\Delta_{SyOp} \times (\Delta_{SyOp} \cup \{\delta_0\}) \times \mathcal{R}_{SyOp})$$

- l'ensemble des unités d'indexation de la vue mémoire visuelle est:

$$\mathcal{V}_{DMV} = \mathcal{P}(\Delta_{MV} \times (\Delta_{MV} \cup \{\delta_0\}) \times \mathcal{R}_{MV})$$

L'ensemble des unités d'indexation de GRIM est alors:

$$\mathcal{V}_{DGRIM} = \mathcal{V}_{DStruct} \cup \mathcal{V}_{DSyOp} \cup \mathcal{V}_{DMV} \cup \mathcal{P}(\Delta_{Struct} \times (\Delta_{SyOp} \cup \Delta_{MV}) \times \{Struct\})$$

Remarques :

1. Il existe des contraintes sur les triplets formés dans les vocabulaires définis ci-dessus¹. Nous ne les avons pas définies explicitement, afin de ne pas compliquer la notation. Mais ils sont schématisés dans la Figure 56. Ainsi, dans la vue symbolique, par exemple, une unité est un triplet qui ne peut être formé que par :

- un identifiant référençant un terme étiqueté par 'Composante', un identifiant référençant un terme étiqueté par 'Description' et la relation 'ComDesc', ou
- un identifiant référençant un terme étiqueté par 'Illustratif', un identifiant référençant un terme étiqueté par 'Description' et la relation 'IllusDesc', ou
- un identifiant référençant un terme étiqueté par 'Illustratif' ou par 'Caractère', l'identifiant neutre δ_0 et la relation neutre ρ_0 .

2. Dans la Figure 56, les identifiants d'entités ne sont pas représentés. L'affectation de ces identifiants est implicite (à chaque entité correspond un identifiant unique).

3. Le symbole schématisé dans la Figure 56 signifie que l'entité représentée peut apparaître plusieurs fois (avec des identifiants différents).

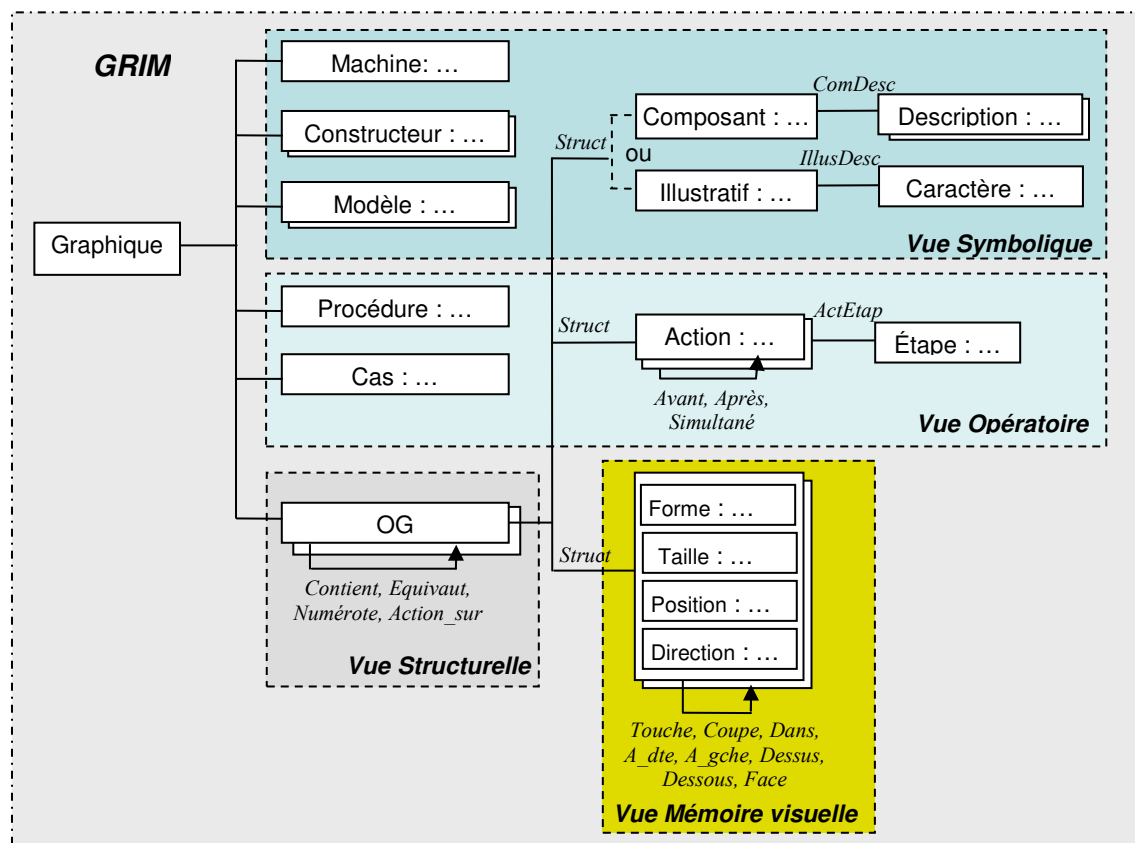


Figure 56 Schématisation du langage GRIM

¹ C'est la raison pour laquelle nous avons utilisé la fonction \mathcal{P} (sous-ensemble de)

Une fois les graphiques indexés, la question se pose de savoir comment les retrouver. Nous proposons ici trois langages de requêtes possibles.

3. Formulation de la requête : 3 langages

Comme nous l'avons mentionné précédemment, la proposition d'un accès au document par le graphique est motivée par le fait que l'utilisateur se rappelle vaguement du graphique qu'il désire retrouver (ou qui se situe dans une zone du document à laquelle il désire accéder). L'imprécision et/ou confusion concernant l'information recherchée est primordiale et doit absolument être considérée lors de l'interrogation. Les langages d'interrogation doivent donc permettre d'exprimer les doutes de l'utilisateur : « *Le graphique représentait une imprimante, la N17 je crois...* », « *Le graphique contenait un gros cube, un écran probablement rectangulaire et peut être un zoom.* »... Afin de permettre une telle expressivité dans la requête nous nous basons sur le modèle de recherche proposé dans la partie 2 de ce manuscrit afin de définir trois langages d'interrogations offrant ainsi à l'utilisateur différentes façons de mettre à plat son besoin. Ces trois langages s'articulent tous autour d'une description du graphique que l'utilisateur désire retrouver : le premier langage s'appuie uniquement sur le texte, le deuxième uniquement sur le graphique et le troisième sur un mélange des deux, que nous appelons « mixte ».

Dans tous les cas, le vocabulaire d'interrogation est formé par un ensemble d'identifiants d'interrogation $\Delta_{Q_{GRIM}}$ et d'un ensemble de relations d'interrogation $\mathcal{R}_{Q_{GRIM}}$. L'ensemble des unités d'interrogation servant de base à la requête est noté $\mathcal{U}_{Q_{GRIM}}$. Il s'agit de l'ensemble des triplets formés par deux identifiants d'interrogation et d'une relation d'interrogation.

Nous redéfinissons, pour chacun des trois langages textuel, graphique et mixte, les ensembles d'identifiants d'interrogation, de relations d'interrogation et d'unités d'interrogation, soient :

- $\Delta_{Q_{Texte}}$, $\mathcal{R}_{Q_{Texte}}$ et $\mathcal{U}_{Q_{Texte}}$ pour le langage textuel,
- $\Delta_{Q_{Graph}}$, $\mathcal{R}_{Q_{Graph}}$ et $\mathcal{U}_{Q_{Graph}}$ pour le langage graphique,
- $\Delta_{Q_{Mixte}}$, $\mathcal{R}_{Q_{Mixte}}$ et $\mathcal{U}_{Q_{Mixte}}$ pour le langage mixte.

3.1. Langage textuel

Il s'agit, grâce à ce langage de permettre à l'utilisateur de décrire le contenu du graphique, qui, selon lui, répondrait le mieux à son besoin, dans un langage textuel en utilisant des termes prédéfinis ou en en proposant d'autres. Ici, l'attention est donnée aux vues symbolique et opératoire. La vue mémoire visuelle, quand à elle, n'est pas considérée puisqu'elle permet de décrire les propriétés visuelles (formes, dispositions, etc.) des objets contenus dans le graphique, propriétés que l'utilisateur exprime en utilisant une palette de dessin et non pas du texte.

Exemple

Soit la requête, saisie dans l'interface de la Figure 57: « Je désire retrouver les graphiques expliquant la procédure à suivre pour résoudre un problème papier, pour les imprimantes xerox. Ce graphique représente un *panneau avant* et éventuellement un objet qui pourrait être un *clavier* ».

Dans la zone de texte relative à « mots clés », les termes sont séparés par des virgules. Le symbole « + » précédant un terme signifie que ce terme est obligatoire (comme cela a été proposé dans certains moteurs de recherche) et le symbole « ? » suivant un terme signifie que ce terme est incertain. Ainsi, dans l'exemple, « panneau avant » est obligatoire et certain et « clavier » est optionnel et incertain.

Figure 57 Exemple d'interface pour la saisie d'une requête textuelle

3.1.1. Le vocabulaire d'interrogation

Dans ce type d'interrogation ne sont considérés que les entités relatives aux ensembles : $\mathcal{T}_{Machines}$, $\mathcal{T}_{Constructeurs}$, $\mathcal{T}_{Modèles}$, $\mathcal{T}_{Procédures}$ et \mathcal{T}_{mots} . Ceci implique que les identifiants d'entités concernés par ce langage sont uniquement ceux référant une entité de ce sous-ensemble.

a. Les identifiants d'interrogation

Les identifiants d'entités concernés par le langage textuel sont ceux référant toute entité des vues symbolique et opératoire mises à part les entités relatives à des objets illustratifs (tels que les zooms par exemple). Il s'agit de l'ensemble défini ainsi :

$$\{\delta_{SyOp} \in \Delta_{SyOp} \text{ tels que } \mathcal{F}_{\delta t}(\delta_{SyOp}) \in \mathcal{T}_{SyOp} \setminus (\{\text{Illustratif}\} \times \mathcal{T}_{Illustr})\}$$

D'un autre côté, selon le langage général défini, chaque entité peut être marquée par des critères *obligatoire/optionnel* et *certain/incertain*. Néanmoins, dans le langage textuel, nous fixons par défaut certains de ces critères selon le type de l'entité considérée (plus exactement, selon son étiquette), car ils nous paraissent évidents (voir Tableau 4).

<i>Etiquette</i>	<i>Obligatoire/optionnel</i>	<i>Certain/ incertain</i>
<i>Procédure</i>	<i>obligatoire</i>	<i>certain</i>
<i>Constructeur</i>	<i>obligatoire</i>	-
<i>Machine</i>	<i>obligatoire</i>	-
<i>Modèle</i>	<i>obligatoire</i>	-
<i>Composant</i>	<i>obligatoire</i>	<i>certain</i>
e_0	-	-

Tableau 4 Les critères par défaut, selon l'étiquette de l'entité, dans le cas du langage textuel

L'ensemble des identifiants d'interrogation est alors :

$$\begin{aligned}
 \Delta_{Q_{\text{Texte}}} = & \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ \textit{Procédure} \} \times \mathcal{T}_{\textit{Procédure}} \} \times \{ \textit{obligatoire} \} \times \{ \textit{certain} \} \\
 & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ \textit{Constructeur} \} \times \mathcal{T}_{\textit{Constructeur}} \} \times \{ \textit{obligatoire} \} \times \mathcal{J} \\
 & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ \textit{Machine} \} \times \mathcal{T}_{\textit{Machine}} \} \times \{ \textit{obligatoire} \} \times \mathcal{J} \\
 & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ \textit{Modèle} \} \times \mathcal{T}_{\textit{Modèle}} \} \times \{ \textit{obligatoire} \} \times \mathcal{J} \\
 & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ \textit{Composant} \} \times \mathcal{T}_{\textit{mots}} \} \times \{ \textit{obligatoire} \} \times \{ \textit{certain} \} \\
 & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, F_{\delta\tau}(\delta_{\text{SyOp}}) \in \{ e_0 \} \times \mathcal{T}_{\textit{SyOp}} \} \times \mathcal{O} \times \mathcal{J} \\
 & \cup \{ \delta_{\emptyset} \}
 \end{aligned}$$

b. Les relations d'interrogation

Il s'agit de la relation d'interrogation neutre (qui permet de définir des unités d'interrogations formées d'un identifiant d'interrogation qui n'est en relation avec aucun autre):

$$\mathcal{R}_{Q_{\text{Texte}}} = \{ \rho_{\emptyset\emptyset} \}$$

Ce choix de ne pas tenir compte des relations dans le langage textuel est fait afin de simplifier la formulation de la requête textuelle. En effet, nous proposons ce langage, pour permettre à l'utilisateur de formuler rapidement son besoin sans l'encombrer de détails à préciser tels que l'ordre des actions, ou la composition des objets contenus dans le graphique, etc.

c. Les unités d'interrogation

Les unités d'interrogation sont, comme définies dans le modèle général, des triplets formés par deux identifiants d'interrogation et une relation d'interrogation. Dans le langage d'interrogation textuel, les identifiants d'interrogation sont indépendants, ils ne sont en relation avec aucun autre identifiant. Nous définissons donc :

$$\mathcal{U}_{Q_{\text{Texte}}} = \Delta_{Q_{\text{Texte}}} \times \{ \delta_{\emptyset\emptyset} \} \times \{ \rho_{\emptyset\emptyset} \}$$

Nous remarquons, ici, que le langage peut être vu, simplement, comme un ensemble d'identifiants augmentés de critères d'obligation et de certitude. Il n'y a aucune relation entre ces identifiants.

3.1.2. La requête

La requête est définie par un sous ensemble d'unités d'interrogation :

$$q \in \mathcal{P}(\mathcal{U}_{Q_{\text{Texte}}})$$

Il s'agit, en réalité, d'identifiants relatifs à des termes étiquetés, auxquels sont rattachés des critères d'obligation et de certitude.

3.1.3. Conclusion sur le langage textuel

Le langage textuel tel que nous l'avons défini, offre à l'utilisateur un moyen rapide et simple d'exprimer son besoin, dans le cas où il n'a pas forcément un souvenir visuel du graphique, mais plutôt une mémoire de sa sémantique ou de ce qu'il représente.

Dans ce langage, les critères relatifs à certaines entités ont été fixés par défaut en raison de leur évidence, et les relations entre les entités ont été mises de côté afin de ne pas encombrer l'utilisateur de détails à expliciter, ce qui ralentirait sa formulation du besoin, dans un cas de recherche que nous souhaitons simple et rapide.

3.2. Langage graphique

Dans le langage graphique, seules les données visuelles sont prises en compte. Nous nous intéressons uniquement à la vue mémoire visuelle de GRIM.

D'un point de vue pratique, l'utilisateur dessine ce dont il se souvient du graphique recherché : une palette de dessin lui est fournie pour dessiner sa requête et, pour chaque objet dessiné, il peut spécifier ses doutes quant à l'existence de l'objet dans le graphique (obligatoire/optionnel) ainsi que ses doutes quant à ses caractéristiques (position, forme, taille et sens) (certain/incertain).

Dans ce qui suit nous redéfinissons le vocabulaire d'interrogation ainsi que la requête, dans ce cas particulier d'interrogation purement graphique.

Exemple

Soit la requête, représentée dans la Figure 58. Selon la figure, l'utilisateur veut que le graphique contienne trois objets graphiques dont *obligatoirement* un zoom dont il n'est *pas certain* de la taille.

L'objet sélectionné (en rouge) est :

Obligatoire Optionnel

Etes-vous certain de :

- sa forme ?	<input checked="" type="radio"/> Oui	<input type="radio"/> Non
- sa taille ?	<input type="radio"/> Oui	<input checked="" type="radio"/> Non
- son sens ?	<input checked="" type="radio"/> Oui	<input type="radio"/> Non
- sa position ?	<input checked="" type="radio"/> Oui	<input type="radio"/> Non

—————	Obligatoire certain
-----	Optionnel certain
.....	Obligatoire incertain
.....	Optionnel incertain

Figure 58 Exemple de requête graphique

3.2.1. Le vocabulaire d'interrogation

Dans ce type d'interrogation ne sont considérés que les éléments de la vue mémoire visuelle.

a. Les identifiants d'interrogation

Les identifiants d'entités concernés par le langage graphique sont ceux définis dans la vue mémoire visuelle. Il s'agit de l'ensemble Δ_{MV} .

Selon le langage général défini, chaque entité peut être marquée par des critères *obligatoire/optionnel* et *certain/incertain*. Dans le langage graphique, la certitude ne concerne pas uniquement l'entité comme un objet fermé, mais s'applique à ses caractéristiques une à une (forme, taille, position et direction). Ainsi, un objet graphique est considéré comme certain si toutes ses caractéristiques sont certaines, et il est considéré comme incertain si au moins une de ses caractéristiques est incertaine : dans l'exemple de la Figure 58, l'entité sélectionnée est considérée comme incertaine, car sa caractéristique *taille* l'est aussi.

Ainsi, le critère d'obligation s'applique à l'entité (comme cela a été défini dans le modèle général), et le critère de certitude, rattaché à cette entité est considéré comme la combinaison de l'application de quatre autres critères de certitude à ses quatre caractéristiques, soient dans l'ordre: à sa forme, à sa taille, à sa position et à sa direction. Ainsi l'utilisateur peut mentionner exactement sur quelle caractéristique de l'entité visuelle porte son incertitude. Ainsi, concernant une entité visuelle dessinée par l'utilisateur, par exemple, il pourra exprimer que : il est important que cette entité apparaisse dans les documents recherchés (*entité obligatoire*), il est sûr de sa forme rectangulaire (*forme certaine*), de sa petite taille (*taille certaine*), de sa direction (*direction certaine*), mais pas de sa position (*position incertaine*).

Nous redéfinissons alors l'ensemble des identifiants d'interrogation ainsi :

$$\Delta_{QGraphi} = \Delta_{MV} \times \mathcal{C} \times \mathcal{J}^4 \cup \{\delta_Q\}$$

Cette redéfinition de l'ensemble des identifiants d'interrogation implique la redéfinition des fonctions qui y sont rattachées. Cela passe par :

- la redéfinition de la fonction \mathcal{F}_δ (qui pour un identifiant d'interrogation δ_Q renvoi l'identifiant δ correspondant) :

$$\begin{aligned} \mathcal{F}_\delta: \quad \Delta_{QGraphi} &\rightarrow \Delta_{MV} \\ \delta_Q = (\delta, \mathbf{a}, \mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3, \mathbf{i}_4) &\rightarrow \delta \end{aligned}$$

- ainsi que la définition d'une nouvelle fonction permettant de connaître pour un identifiant d'entité donné et pour chacune de ses caractéristique, le critère qui lui est attribué.

Cette nouvelle fonction, que nous notons $\mathcal{F}_{CerGraphi}$, est définie ainsi:

Soit un identifiant d'interrogation $\delta_Q = (\delta, \mathbf{a}, \mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3, \mathbf{i}_4)$,

$$\begin{aligned} \mathcal{F}_{CerGraphi}: \Delta_{QGraphi} \times \mathcal{E}_{MV} &\rightarrow \mathcal{J} \\ (\delta_Q, e) &\rightarrow \mathbf{i}_1 \text{ si } e=\text{Forme}, \mathbf{i}_2 \text{ si } e=\text{Taille}, \mathbf{i}_3 \text{ si } e=\text{Position}, \mathbf{i}_4 \text{ si } e=\text{Direction}. \end{aligned}$$

Nous redéfinissons donc pour un identifiant d'interrogation dans le langage graphique :

$$\mathcal{F}_{Cer}(\delta_Q) = \mathcal{F}_{CerGraphi}(\delta_Q, \text{Forme}) \wedge \mathcal{F}_{CerGraphi}(\delta_Q, \text{Taille}) \wedge \mathcal{F}_{CerGraphi}(\delta_Q, \text{Position}) \wedge \mathcal{F}_{CerGraphi}(\delta_Q, \text{Direction})$$

b. Les relations d'interrogation

Il s'agit de la relation d'interrogation neutre (qui permet de définir des unités d'interrogations formées d'un identifiant d'interrogation qui n'est en relation avec aucun autre):

$$\mathcal{R}_{QGraph} = \{\rho_{\emptyset}\}$$

c. Les unités d'interrogation

Les unités d'interrogation sont, comme définies dans le modèle général, des triplets formés par deux identifiants d'interrogation et une relation d'interrogation. Dans le langage d'interrogation graphique, comme pour le langage textuel, les identifiants d'interrogation sont indépendants, ils ne sont en relation avec aucun autre identifiant. Nous définissons donc :

$$\mathcal{U}_{QGraph} = \Delta_{QGraph} \times \{\delta_{\emptyset}\} \times \{\rho_{\emptyset}\}$$

Nous remarquons que comme pour le langage textuel, ce langage peut être vu, simplement, comme un ensemble d'identifiants augmentés de critères d'obligation et de certitude. Il n'y a aucune relation entre ces identifiants.

3.2.2. La requête

La requête est définie par un sous ensemble d'unités d'interrogation :

$$q \in \mathcal{P}(\mathcal{U}_{QGraph})$$

Il s'agit, en réalité, d'identifiants relatifs à des objets graphiques, auxquels sont rattachés des critères d'obligation et de certitude.


3.2.3. Conclusion sur le langage graphique

Tout comme le langage textuel, le langage graphique, tel que nous l'avons défini, offre à l'utilisateur un moyen rapide et simple d'exprimer son besoin. Nous avons proposé ce langage graphique, pour le cas où l'utilisateur a un souvenir plutôt visuel du graphique. Il peut ainsi dessiner rapidement le graphique qu'il juge répondre idéalement à son besoin, en désignant ce qui est important et moins important que ce graphique contienne et ce dont il est certain ou ce dont il doute quant aux caractéristiques des objets dessinés. Ces précisions n'encombrent pas l'utilisateur qui a uniquement besoin de cocher les cases adéquates, quand c'est nécessaire.





3.3. Langage mixte

Dans le langage mixte, les données textuelles et les données visuelles sont prises en compte. Nous nous intéressons, ici, aux différentes vues définissant GRIM.

D'un point de vue pratique, l'utilisateur peut, comme pour le langage graphique:

- dessiner ce dont il se souvient du graphique recherché : une palette de dessin, contenant des formes prédéfinies, lui est fournie pour dessiner sa requête. A certaines formes sont rattachés des termes par défaut (par exemple, le terme « imprimante » est rattaché systématiquement à la forme ,
- spécifier, pour chaque objet dessiné, ses doutes quant à l'existence de l'objet dans le graphique (obligatoire/optionnel) ainsi que ses doutes quant à ses caractéristiques (position, forme, taille et sens) (certain/incertain).

En plus de ce niveau graphique, l'utilisateur peut :

- associer des termes aux formes (leur correspondant symbolique et opératoire), termes auxquels sont associés des critères d'obligation et de certitude,
- ajouter des termes dans la requête sans qu'aucune forme ne leur soit associée,
- affecter des relations entre les entités (l'entité  s'applique à  où à l' « écran d'affichage », par exemple),
- définir des entités composées (par exemple, l'entité visuelle correspondant à « imprimante » est composée des entité  et  »

Dans ce qui suit nous redéfinissons le vocabulaire d'interrogation ainsi que la requête, dans ce cas particulier d'interrogation mixte.

3.3.1. Le vocabulaire d'interrogation

Dans ce type d'interrogation tous les éléments définis dans le langage GRIM sont considérés.

a. Les identifiants d'interrogation

Les identifiants d'entités concernés par le langage mixte sont ceux définis dans GRIM. Il s'agit de l'ensemble Δ_{GRIM} .

Selon le langage général défini, chaque entité peut être marquée par des critères *obligatoire/optionnel* et *certain/incertain*. Dans le langage mixte, certains critères sont fixés par défaut (voir Tableau 5) comme cela a été fait au niveau du langage textuel, et au niveau des objets graphiques, la certitude est détaillée pour les quatre caractéristiques de chaque objet, comme c'est le cas pour le langage graphique.

<i>Etiquette</i>	<i>Obligatoire/ optionnel</i>	<i>Certain/ incertain</i>
<i>Procédure</i>	<i>obligatoire</i>	<i>certain</i>
<i>Constructeur</i>	<i>obligatoire</i>	-
<i>Machine</i>	<i>obligatoire</i>	-
<i>Modèle</i>	<i>obligatoire</i>	-
<i>Composant, Illustratif, Description, Caractère, Action, Etape, Cas</i>	-	-

Tableau 5 Les critères par défaut, selon l'étiquette de l'entité, dans le cas du langage mixte

L'ensemble des identifiants d'interrogation est alors :

$$\Delta_{Q_{Mixte}} = \{\delta_{SyOp} \in \Delta_{SyOp}, F_{\delta\tau}(\delta_{SyOp}) \in \{Procédure\} \times \mathcal{T}_{Procédure}\} \times \{obligatoire\} \times \{certain\}$$

$$\cup \{\delta_{SyOp} \in \Delta_{SyOp}, F_{\delta\tau}(\delta_{SyOp}) \in \{Constructeur\} \times \mathcal{T}_{Constructeur}\} \times \{obligatoire\} \times \mathcal{J}$$

$$\cup \{\delta_{SyOp} \in \Delta_{SyOp}, F_{\delta\tau}(\delta_{SyOp}) \in \{Machine\} \times \mathcal{T}_{Machine}\} \times \{obligatoire\} \times \mathcal{J}$$

$$\cup \{\delta_{SyOp} \in \Delta_{SyOp}, F_{\delta\tau}(\delta_{SyOp}) \in \{Modèle\} \times \mathcal{T}_{Modèle}\} \times \{obligatoire\} \times \mathcal{J}$$

$$\cup \{\delta_{SyOp} \in \Delta_{SyOp}, F_{\delta\tau}(\delta_{SyOp}) \in \{Composant, Illustratif, Description, Caractère, Action, Etape, Cas\} \times \mathcal{T}_{mots}\} \times \mathcal{O} \times \mathcal{J}$$

$$\begin{aligned} & \cup \{ \delta_{\text{SyOp}} \in \Delta_{\text{SyOp}}, \mathcal{F}_{\delta_{\text{t}}}(\delta_{\text{SyOp}}) \in \{ \text{Illustratif} \} \times \mathcal{T}_{\text{illust}} \} \times \mathcal{O} \times \mathcal{J} \\ & \cup \Delta_{\text{MV}} \times \mathcal{O} \times \mathcal{J}^4 \\ & \cup \Delta_{\text{Struct}} \times \mathcal{O} \times \mathcal{J} \cup \{ \delta_{\text{Q}} \} \end{aligned}$$

b. Les relations d'interrogation

Contrairement aux langages textuel et graphique qui ne permettent pas la prise en compte de relations entre les entités, le langage mixte prend en compte toutes les relations définies dans le vocabulaire $\mathcal{V}_{\text{GRFM}}$.

Comme pour les entités, certains critères sont fixés par défaut selon la nature de la relation. Ceci est le cas, par exemple, de la relation *struct* qui met en relation un objet graphique et ses correspondants dans les vues symbolique, opératoire et mémoire visuelle. Cette relation est toujours obligatoire et certaine (elle est construite automatiquement lors de la traduction de la requête). D'autres critères sont posés par défaut, ils sont présentés dans le Tableau 6.

<i>Relation</i>	<i>Obligatoire/ optionnel</i>	<i>Certain/ incertain</i>
<i>Struct, Face, ComDesc, IllusDesc, ActEtap</i>	<i>obligatoire</i>	<i>certain</i>
<i>contient, équivaut, numérote, action sur</i>	-	<i>certain</i>
<i>Touche, Coupe, Dans, A_dte, A_gche, Dessus, Dessous, Avant, Après, Simultané</i>	-	-

Tableau 6 Les critères par défaut, selon le type de relation, dans le cas du langage mixte

L'ensemble des relations d'interrogation est alors :

$$\begin{aligned} \mathcal{R}_{\text{QMixte}} &= \{ \text{Struct, Face, ComDesc, IllusDesc, ActEtap} \} \times \{ \text{obligatoire} \} \times \{ \text{certain} \} \\ & \cup \{ \text{contient, équivaut, numérote, action_sur} \} \times \mathcal{O} \times \{ \text{certain} \} \\ & \cup \{ \text{Touche, Coupe, Dans, A_dte, A_gche, Dessus, Dessous, Avant, Après, Simultané} \} \times \mathcal{O} \times \mathcal{J} \end{aligned}$$

c. Les unités d'interrogation

Les unités d'interrogation sont, comme définies dans le modèle général, des triplets formés par deux identifiants d'interrogation et une relation d'interrogation. Dans le langage d'interrogation mixte, contrairement aux langages textuel et graphique, les identifiants d'interrogation ne sont pas indépendants, ils sont en relation les uns avec les autres. Nous définissons donc :

$$\mathcal{U}_{\text{QMixte}} \in \mathcal{P}(\Delta_{\text{QMixte}} \times \Delta_{\text{QMixte}} \times \mathcal{R}_{\text{QMixte}})$$

3.3.2. La requête

La requête est définie par un sous ensemble d'unités d'interrogation :

$$q \in \mathcal{P}(\mathcal{U}_{\text{QMixte}})$$

3.3.3. Conclusion sur le langage mixte

Le langage mixte est le plus complet puisqu'il permet à l'utilisateur d'exprimer son besoin en dessinant sa requête, mais aussi en rattachant une sémantique à son dessin. L'utilité de ce

langage apparaît lorsque l'utilisateur se souvient visuellement du graphique (ou d'une partie du graphique) et qu'il a aussi une mémoire de sa sémantique ou de ce qu'il représente.

A l'aide de ce langage, l'utilisateur peut décrire avec détail le graphique qui répondrait idéalement à son besoin, en ayant la possibilité de caractériser les objets dessinés, d'exprimer des relations entre ces objets, d'ajouter des objets sans les représenter graphiquement, etc.

4. Correspondance entre la requête et les documents

La correspondance entre la requête et les graphiques techniques est une instance particulière de la correspondance définie dans le modèle général et que nous rappelons ici. Nous gardons les notations du modèle général (qu'ils aient été redéfinis ou non) car nous l'appliquons aux trois langages d'interrogations.

$Corresp(q, DI_i)$ est vrai ssi :

$\exists \Delta'_{OpCer} \subseteq \Delta_{OpCer}(q), \Delta'_{OpInc} \subseteq \Delta_{OpInc}(q)$ et $DI'_i \subseteq \mathcal{F}_{\Delta D}(DI_i)$, et
 \exists une fonction *injective* $\mathcal{F}_{Corresp}$ définie de $\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc}$ dans DI'_i ,
tels que :

- (i) si $\delta \in \Delta_{ObCer} \cup \Delta'_{OpCer}, \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta)) = \mathcal{F}_{\delta\tau}(\delta)$,
et si $\delta \in \Delta_{ObInc} \cup \Delta'_{OpInc}, \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta)) \in \mathcal{Proche}_{\mathcal{T}}(\mathcal{F}_{\delta\tau}(\delta))$
- (ii) $\forall v_{q,k} = (\delta_{Q_i}, \delta_{Q_j}, \rho_{Q_t}) \in q$ tel que $(\delta_{Q_i}, \delta_{Q_j}) \in (\Delta_{ObCer} \cup \Delta_{OpInc} \cup \Delta'_{OpCer} \cup \Delta'_{OpInc})^2$,
 $\delta_1 = \mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\delta_{Q_i})), \delta_2 = \mathcal{F}_{Corresp}(\mathcal{F}_{\delta}(\delta_{Q_j}))$ et $\rho = \mathcal{F}_{\rho}(\rho_{Q_t})$,
si $v_{q,k} \in \mathcal{V}_{ObCer}, (\delta_1, \delta_2, \rho) \in DI_i$,
et si $v_{q,k} \in \mathcal{V}_{ObInc}, \exists \rho' \in \mathcal{Proche}_{\mathcal{R}}(\rho)$ tel que $(\delta_1, \delta_2, \rho') \in DI_i$

Pour les langages textuel et graphique, une simplification peut être faite, dans cette fonction, en ne tenant pas compte de la condition (ii). En effet, les relations entre entités ne sont pas prises en compte dans les requêtes textuelles et graphiques, d'où l'inutilité de vérifier le respect des contraintes sur les relations.

D'un autre côté, au niveau des langages graphique et mixte une particularité concernant l'entité 'objet visuel' doit être considérée. En effet, pour une telle entité, la certitude porte sur ses quatre caractéristiques. Ceci n'influe pas sur la fonction de correspondance lorsque les quatre caractéristiques sont marquées comme certaines, mais doit être considéré lorsqu'au moins une caractéristique est marquée comme incertaine. Dans ce dernier cas, la condition si $\delta \in \Delta_{ObInc} \cup \Delta'_{OpInc}, \mathcal{F}_{\delta\tau}(\mathcal{F}_{Corresp}(\delta)) \in \mathcal{Proche}_{\mathcal{T}}(\mathcal{F}_{\delta\tau}(\delta))$ doit être reconsidérée ou plus exactement spécifiée. Le problème, ici, est de savoir quel objet visuel du document indexé *proche* de celui dont il est question dans la requête est adéquat. En effet, selon que l'incertitude porte sur la *forme* de l'objet ou sa *taille*, par exemple, les objets visuels considérés comme *proches* ne sont pas les mêmes.

Rappelons d'abord que :

- un identifiant d'entité visuelle s'écrit $\delta_Q = (\delta, \alpha, i_1, i_2, i_3, i_4) \in \Delta_{MV} \times \mathcal{O} \times \mathcal{I}^4$,

- pour connaître le critère de certitude associé à une caractéristique e , il suffit d'appliquer la fonction $F_{CerGraphi}(\delta_Q, e)$ qui renvoi i_1 si $e=Forme$, i_2 si $e=Taille$, i_3 si $e=Position$ et i_4 si $e=Direction$,
- pour connaître l'identifiant d'entité δ correspondant à un identifiant d'interrogation δ_Q , il suffit d'appliquer la fonction $F_{\delta}(\delta_Q)$,
- pour connaître le terme associé à l'étiquette e , d'une entité visuelle τ , il suffit d'appliquer la fonction $F_{tMV}(\tau, e)$.

Nous modifions, alors, la contrainte :

$$' \text{ si } \delta \in \Delta_{ObInc} \cup \Delta'_{OpInc}, F_{\delta\tau}(F_{Corresp}(\delta)) \in \text{Proche}_{\tau}(F_{\delta\tau}(\delta))'$$

qui devient :

$$\text{si } \delta \in \Delta_{ObInc} \cup \Delta'_{OpInc}, F_{\delta\tau}(F_{Corresp}(\delta)) \in \text{Proche}_{\tau}(F_{\delta\tau}(\delta)), \text{ et}$$

$$\text{si, de plus, } \delta \in \Delta_{MV}, \forall e \in E_{MV}, \text{ si } F_{CerGraphi}(F_{\delta}^{-1}(\delta), e) = \text{certain} \text{ alors } F_{tMV}(F_{\delta\tau}(\delta), e) = F_{tMV}(F_{\delta\tau}(F_{Corresp}(\delta)), e)$$

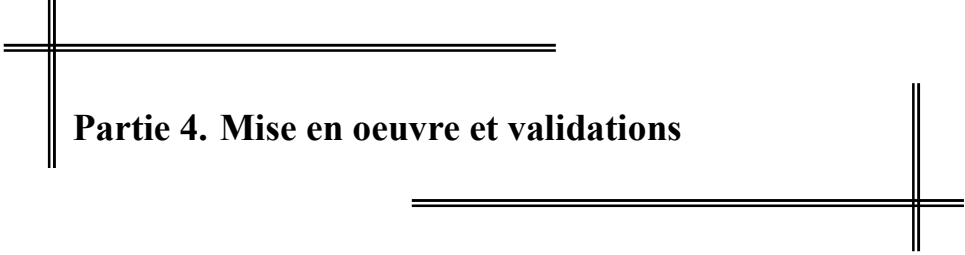
Remarque : nous pouvons garder la fonction $\text{Proche}_{\tau}(\tau)$ car nous l'avons redéfinie lors de l'instanciation du vocabulaire dans la vue mémoire visuelle ($\text{Proche}_{tMV}(\tau)$).

5. Conclusion

Nous avons présenté, dans ce chapitre, un modèle de recherche de graphiques des documents techniques à usage professionnel, qui est une instance du modèle général défini dans la partie 2 de ce manuscrit. Ce modèle combine, de façon structurée, un maximum de descriptions relatives au contenu des graphique, et propose trois langages d'interrogation permettant de formuler la requête de différentes façons.

Cette modélisation, propre à un cadre applicatif particulier, permet de ressentir, concrètement, l'utilité des différents points spécifiques au modèle général proposé, soient, l'ajout des critères d'obligation et de certitude, l'utilisation multiple d'entités (en introduisant les identifiants d'entité) et l'utilisation des relations entre entités.

Dans la partie suivante, nous nous intéressons à l'indexation et à l'interrogation des graphiques d'un point de vue pratique : (i) l'indexation en proposant un processus automatique d'indexation (permettant l'extraction des index textuels) dont nous évaluons la qualité, et (ii) l'interrogation en validant l'ajout des critères d'obligation et de certitude (en comparant avec les modèles booléen, vectoriel, probabiliste et de langue)



Partie 4. Mise en oeuvre et validations

Nous avons défini dans la partie précédente un modèle de recherche de graphiques techniques par des professionnels. Ce modèle se base, entre autres, sur une indexation fondée sur un langage complexe, et sur une interrogation fondée sur des critères d'obligation/option et de certitude/incertitude. Le but de cette partie est de mettre en œuvre ces deux processus (l'indexation et l'interrogation) et les valider.

Dans le chapitre VII, nous nous intéressons à l'indexation. Celle-ci est caractérisée par une indexation automatique correspondant à la représentation du contenu sémantique du graphique pris en compte dans les vues symbolique et opératoire (indexation textuelle), et une indexation manuelle correspondant à la représentation du contenu visuel du graphique pris en compte dans la vue mémoire visuelle (indexation visuelle). Nous décrivons donc, dans ce chapitre, (i) le processus automatique que nous avons mis en place pour la construction de l'index textuel et la validation de cette indexation en utilisant une adaptation des mesures de qualités existantes (qui sont uniquement appropriées pour des langages simples), ainsi que (ii) le protocole d'évaluation que nous proposons pour valider notre indexation visuelle.

Dans le chapitre VIII, nous nous intéressons à l'interrogation. Celle-ci est caractérisée par un enrichissement des éléments de la requête par les critères d'obligation/option et de certitude/incertitude et une fonction de correspondance prenant en compte les contraintes imposées par la formulation de la requête. Dans ce chapitre, nous décrivons des expériences dont l'objectif est de montrer le bien fondé de ces deux points, autrement dit, que sans l'ajout des critères dans la requête le résultat obtenu est moins satisfaisant, et que la fonction de correspondance, telle que nous l'avons définie, est nécessaire pour rendre compte, correctement, du comportement que nous attendons du système.

Chapitre VII. Indexation textuelle et visuelle des graphiques

L'indexation telle que nous l'avons définie englobe deux niveaux : une indexation textuelle (§1) qui tire ses informations du texte qui accompagne le graphique dans le document technique, et une indexation visuelle (§2) qui est fondée sur les caractéristiques intrinsèques de ce graphique.

Au niveau de l'indexation textuelle, il s'agit de repérer, dans le texte accompagnant le graphiques, les termes utiles à son indexation, selon le modèle proposé, GRIM. Ces termes sont ensuite organisés dans la structure de l'index selon le modèle proposé. Notons que ce niveau d'indexation se limite aux vues symbolique et opératoire de GRIM, et à une partie de la vue structurelle. L'indexation est fondée ici sur des techniques de traitement automatiques de la langue : il s'agit d'extraire des termes candidats à l'aide de patrons morphosyntaxiques et de filtrer ces termes en fonction de leur localisation dans le texte du commentaire du graphique, afin de construire un index structuré.

Cette indexation textuelle étant un processus automatique, et l'index du graphique étant fondé sur un langage complexe, la validation d'une telle indexation nous paraît nécessaire. Nous proposons donc d'adapter les mesures de qualité d'indexations fondées sur des langages simples (présentées dans l'état de l'art, Partie I-ChapitreI) afin qu'elles conviennent à des indexations fondées sur des langages complexes. Nous nous basons sur ces mesures pour valider la qualité de notre indexation textuelle.

Au niveau de l'indexation visuelle, il s'agit d'une indexation manuelle, qui se base sur la mémoire visuelle de l'utilisateur. Notons que ce niveau d'indexation se limite aux vues mémoire visuelle et structurelle de GRIM. Notre but, ici, est la validation de cette indexation, autrement dit, la validation des approximations et hypothèses que nous avons posées au niveau du langage GRIM. Cette partie présente uniquement une description du protocole expérimental que nous projetons d'appliquer.

1. Indexation textuelle : processus et validation

Dans un document structuré, les blocs formant sa hiérarchie, loin d'être indépendants, sont reliés entre eux par des relations qui permettent de donner au document une intégrité sémantique. Certaines recherches ont tenu compte de cette dépendance dans le but d'enrichir l'indexation d'un bloc par l'ajout des index des blocs avoisinants ou supérieurs hiérarchiquement (la notion de portée, par exemple). Dans les approches actuelles, l'index d'un bloc sans sémantique est considéré comme étant fonction de l'ensemble des descripteurs des blocs avoisinants. Nous désirons aller au-delà de cette approche et considérer l'indexation du bloc sans sémantique comme étant fonction de certains fragments extraits des blocs alentours, l'extraction dépendant du bloc sans sémantique. C'est dans ce but que nous avons mené une

étude concrète visant à détecter dans le texte avoisinant un graphique, les fragments nécessaires à son indexation sémantique. Notre but est de définir un processus d'indexation automatique des vues symbolique et opératoire des graphiques, qui soit de qualité, et ce malgré un langage d'indexation complexe et un processus automatique.

Donc, en exploitant la régularité des documents techniques, dont la rédaction obéit à des règles strictes, et suite à l'étude d'un document technique, nous tentons de repérer, tout d'abord, le texte relatif au graphique et ensuite, les fragments textuels correspondant à son index dans les vues symbolique et opératoire (voir aussi [Kefi,02] [Kefi,03]). Le repérage de ces fragments et la construction de l'index structuré sont le résultat de l'application:

- de patrons morphosyntaxiques permettant l'extraction, à partir d'un commentaire, des candidats termes représentant, éventuellement, le contenu du graphique concerné.
- de règles permettant de filtrer les candidats termes et de construire l'index structuré du graphique.

Le processus automatique permettant une telle indexation est décrit en §1.3.2.

Afin de valider l'extraction des termes et la construction de l'index structuré, nous proposons une évaluation qualitative de notre indexation, qui valide notre approche et notre processus automatique d'indexation. Cette évaluation repose sur des mesures qualitatives pour indexation complexes que nous proposons en adaptant les mesures précédemment décrites dans l'état de l'art.

1.1. Caractérisation du texte commentant un graphique

Le processus d'indexation du graphique par le texte est le résultat d'une étude d'un manuel technique en langue française avec une forte régularité dans le texte. Ce manuel qui s'intitule «*Les imprimantes laser réseau Xerox DocuPrint N17 et N17b - Manuel utilisateur* » fournit une documentation concernant les deux imprimantes et leur fonctionnement. Il comporte 8 chapitres. Dans notre étude nous en retenons cinq, ceux contenant des graphiques pertinents qui sont au nombre de 62.

Même si notre étude n'a porté que sur un seul document, les expériences décrites dans le § 1.4 de ce chapitre, ont porté sur plusieurs manuels de différents constructeurs et nous ont permis de valider les patrons/règles définis après l'étude de ce seul document.

Remarque : notre étude permet d'aboutir à des critères spécifiques à notre corpus technique (manuels pour imprimantes, scanners, photocopieuses, etc.) Mais, même si elle ne nous permet pas de généraliser les règles d'extraction d'index à d'autres types de corpus (documentation techniques des avions par exemple), elle est une preuve de la faisabilité de cette extraction, et dès lors une étude similaire pourrait être effectuée dans d'autres domaines.

1.1.1. Localisation des blocs textuels relatifs à un graphique

Le document technique est un document structuré. Celui que nous étudions a la structure, classique, telle que celle représentée dans la Figure 59. Ce qui nous intéresse, ici, c'est de localiser les blocs textuels de la structure du document qui décrivent textuellement ce qui est schématisé dans un graphique.

Nous avons distingué deux types d'informations : (i) des informations donnant une idée générale sur ce qui est représenté dans le graphique et (ii) des informations plus précises décrivant en détail l'information qu'il véhicule.

Il est connu que dans tout document structuré, chaque noeud feuille de son arborescence est en relation, au niveau du sens qu'il véhicule, avec ses nœuds « ascendants ». C'est le cas de notre document : chaque graphique est relié à la section, au sous-chapitre et au chapitre qui le contiennent. Cette relation se situe plus exactement entre le graphique et les titres respectifs des section, sous-chapitre et chapitre concernés. Ces nœuds ascendant, que nous appelons *blocs titres* donnent une description globale de ce dont parle le graphique.

Une description détaillée de ce qui est schématisé dans le graphique apparaît dans des blocs textuels reliés au graphique : Il s'agit des *blocs commentaires*. La relation entre le commentaire et le graphique est faite par :

- soit par des liens explicites: c'est le cas lorsque le texte référence directement le graphique (voir figure x, la figure x illustre...),
- soit par des liens implicites: c'est le cas lorsque l'auteur n'utilise pas d'expression linguistique spécifique pour référencer le graphique.

Dans le cas d'une référence explicite, le fait qu'un graphique soit éloigné de son commentaire n'est pas gênant puisque le lecteur peut pointer le graphique à l'aide de la référence. Par contre, dans le cas d'une référence implicite si le graphique est trop éloigné de son commentaire alors l'information se perd, au fur et à mesure de la lecture, empêchant ainsi le lecteur d'effectuer la relation. Nous avons donc supposé qu'un lecteur ne pouvant pas retrouver des liens implicites trop éloignés, l'auteur en a forcément tenu compte lors de la rédaction (ou mise en page). Le texte commentant le graphique est, dans ce cas, celui qui lui est le plus proche.

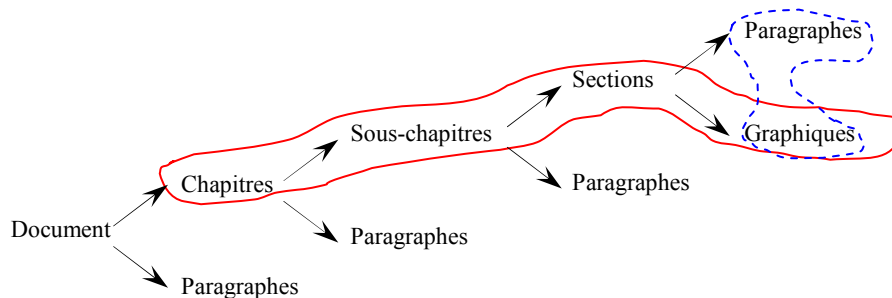


Figure 59 Blocs de textes correspondant à un graphique

1.1.2. Les types de blocs textuels relatifs à un graphique

Les blocs textuels décrivant ce qui est représenté dans un graphique sont des paragraphes. Il s'agit de *paragraphes titres* dans le cas où ils donnent une idée générale sur ce qui est représenté dans le graphique (les blocs titres) et de *paragraphes listes ou classiques* lorsqu'ils décrivent en détail l'information véhiculée par le graphique (les blocs commentaires).

a. Le paragraphe titre

Il s'agit d'un paragraphe particulier. Il se distingue des autres types de paragraphes par sa forme typographique (mise en relief par des caractères gras, par exemple) et sa fonction comme indice de continuité avec au moins le paragraphe qui lui succède directement.

Nous nous intéressons aux titres qui figurent dans le sommaire. Ils indiquent une rupture avec l'unité textuelle qui précède. Ils représentent les différents chapitres, sections, etc. du document. Dans l'exemple suivant, nous pouvons voir le titre du sous-chapitre « *Remplacement de la cartouche EP* » :

3. Remplacement de la cartouche EP
Ne pas exposer...

b. Le paragraphe liste

La liste est un objet textuel que nous considérons comme un bloc cohésif qui vise à transmettre une information d'une façon bien précise et ordonnée. Les listes sont des objets textuels formés d'un ensemble d'éléments (items). Elles peuvent être ordonnées dans un ordre numérique (1, 2, 3...) ou alphabétique (A, B, C,...) ou être marquées par des puces (•). Il peut y avoir une imbrication de listes comme dans l'exemple suivant :

- 1.** Ajustez les guides d'extrémité et de largeur...
 - A** Soulevez le guide d'extrémité...
 - B** Tirez sur le magasin...
 - C** Appuyez sur les deux côtés du guide...
 - D** Faites glisser le guide...
- 2.** Mettez en place le papier.

c. Le paragraphe classique

Il s'agit de tout paragraphe ne correspondant pas aux deux types évoqués ci-dessus. Le paragraphe classique se compose d'un nombre variable de phrases qui sont souvent autonomes du point de vue syntaxique et sémantique. Le critère formel caractérisant ce paragraphe est son ouverture par une majuscule et sa fermeture par un signe de ponctuation souvent forte.

1.1.3. Bilan

Nous avons repéré les blocs relatifs au graphique : il s'agit d'un bloc commentaire et d'au moins un bloc titre. Le bloc commentaire peut se présenter sous deux formes, soit une liste, soit un paragraphe classique.

Reste maintenant à repérer, dans ces blocs, les termes d'indexation adéquats et à construire l'index (symbolique et opératoire) du graphique.

1.2. Repérage des termes d'indexation et construction de l'index

Le critère de qualité majeur d'un document technique est basé sur son efficacité et sa facilité d'utilisation. Pour répondre à cette contrainte, un certain nombre de règles est respecté par les rédacteurs techniques sur le plan de la forme textuelle (voir Partie1-Chapitre II. 1.1.2). Ils doivent se montrer particulièrement vigilants sur la pertinence de leurs phrases, l'homogénéité du texte et les styles de phrases utilisées. Les auteurs suivent ainsi des règles bien précises lors de la rédaction d'un texte technique. Cela se traduit par l'existence d'une certaine régularité au niveau du document.

C'est en exploitant cette régularité que nous définissons des patrons morphosyntaxiques qui, combinés avec la structure du commentaire (liste, titre, etc.), permettent de définir des règles d'extraction et de construction des index des graphiques.

1.2.1. Analyse morphosyntaxique des textes

La manipulation automatique de textes écrits en langue naturelle, quelle que soit la langue utilisée, nécessite souvent une première analyse des éléments formant ces textes. L'objectif de cette étape est de récupérer toute information permettant de caractériser le comportement d'un mot dans son contexte d'énonciation.

Dans notre cas, cette analyse comprend :

- le découpage du texte en phrases : cette étape permet de repérer les frontières de chaque phrase dans le texte afin de pouvoir la séparer pour un éventuel traitement spécifique,
- le repérage des éléments linguistiques de base (segmentation), à savoir les mots. Cette étape permet aussi de distinguer la morphologie de ces éléments (ponctuation, nombres, etc.) et de retrouver pour chacun son lemme (lemmatisation),
- l'étiquetage grammatical du texte : associer à chaque mot ou *token* une catégorie d'ordre grammatical (Nom, Verbe, Adjectif, etc.) en tenant compte du contexte de leur occurrence.

Cette phase fournit des informations de base sur les textes qui sont nécessaires pour les phases suivantes.

1.2.2. Repérage des termes d'indexation potentiels ou candidats termes

a. Approches existantes pour l'extraction de candidats termes

Différentes approches permettent l'extraction de candidat termes à partir d'un texte ayant préalablement subi une analyse morpho-syntaxique. Parmi ces approches, nous citons les approches statistiques, les approches syntaxiques et les approches mixtes.

Les approches statistiques utilisent seulement les co-occurrences de mots. Le principe est que si deux mots co-occurrent souvent dans un certain type de contexte, alors ils peuvent être regroupés dans un terme (voir, par exemple, [Ouesleti,96]).

Les approches syntaxiques utilisent certaines informations syntaxiques dans le choix des termes. Parmi ces approches, nous citons deux familles :

- l'utilisation de patrons morpho-syntaxiques : il s'agit de l'une des techniques les plus utilisées pour l'extraction de termes. Les systèmes basés sur cette technique,

tels que NOMINO [David,90] et LEXTER [Bourigault,96] [Bourigault,05], supposent que les termes à extraire obéissent à des régularités syntaxiques stables. Ces systèmes prennent en entrée un ensemble de patrons constitués d'une suite de catégories grammaticales et qui peuvent être par exemple : NOM NOM / ADJQ NOM / NOM PREP NOM ...¹, et toutes les occurrences de mots correspondant à ces patrons sont extraites comme des candidats-termes potentiels,

- l'utilisation des règles de transformation : ces méthodes permettent d'extraire des termes complexes à partir de connaissances extérieures servant de référence. Généralement, elles identifient des variantes de termes fournis par un thésaurus ou un vocabulaire contrôlé. Nous pouvons citer ici FASTR [Jacquemin,97].

Les approches mixtes combinent des méthodes à orientation statistique et des méthodes à orientation syntaxique. Elles utilisent généralement des calculs statistiques afin d'affiner leurs méthodes d'extraction linguistique. [Daille,94], par exemple, utilise une méthode mixte dans son système ACABIT qui repère, dans le corpus, des candidats-termes à partir de schémas syntaxiques puis les filtre à l'aide de méthodes statistiques.

b. Extraction des candidats termes en utilisant les patrons morpho-syntaxiques

L'étude du corpus nous ayant permis de vérifier que les informations qui nous intéressent obéissent à des régularités syntaxiques stables (ce qui est une des caractéristiques des textes techniques), nous utilisons les patrons syntaxiques. Dans notre approche, un patron syntaxique permet :

- le repérage d'un type isolé de candidat terme : par exemple, le patron COM permet le repérage d'une *composante de dispositif* telle que « Panneau avant », ou
- le repérage de plusieurs types de candidats termes : par exemple, le patron ACT permet le repérage d'une *composante de dispositif* telle que « Panneau avant » et de l'*action qui s'y rapporte* soit « ouvrir ».

Dans le deuxième cas, on utilise une imbrication de patrons.

Une étude des blocs commentaires (les blocs titres ayant un traitement à part) nous a permis de définir des patrons morphosyntaxiques, permettant le repérage de candidats termes pour l'indexation des graphiques.

Un bloc commentaire peut être un paragraphe liste et, dans ce cas, chaque item du bloc commentaire est une phrase (ou à la limite deux phrases) ou il peut être un paragraphe classique, et, dans ce cas, le bloc commentaire est une succession de phrases. Nous cherchons dans chacune des phrases à détecter un patron qui nous permette d'extraire, éventuellement, une ou plusieurs informations relatives:

- aux *_composantes* schématisées dans le graphique et leur *description*,
- aux *_actions* qui s'y rapportent, et à leurs *étapes*.

¹ ADJQ : adjectif qualificatif ; PREP : préposition

Les informations relatives au type de *machine* représenté, son *constructeur*, son modèle ainsi que la *procédure* décrite par le graphique sont extraites des blocs titres.

Dans ce qui suit, nous présentons deux des principaux patrons morphosyntaxiques définis.

Le patron COM:

Les composantes matérielles (plus précisément leurs dénomination) contenues dans le graphique peuvent avoir différentes structures, que l'on peut résumer par le patron morphosyntaxique que nous appelons COM :

COM : SUBC (ABR)* ((PREP)* SUBC)*((PREP)* (SUBC))* (ADJ)* (ADV)* (NUM)*

Exemples

- « Poignée » : SUBC
- « Cartouche EP » : SUBC ABR
- « Carte réseau » : SUBC SUBC
- « Panneau avant » : SUBC ADV
- « Port parallèle » : SUBC ADJ
- « Bac 3 » : SUBC NUM
- « Panneau de commande » : SUBC PREP SUBC
- « Tablette de départ manuel » : SUBC PREP SUBC ADJ
- « Port de carte réseau » : SUBC PREP SUBC SUBC

Le patron COM permet d'extraire un candidat terme pouvant éventuellement représenter l'index « composant » contenu dans le graphique. Le filtrage des candidats termes (voir §1.2.3) permettra de décider s'il peut être ou non considéré comme un index à extraire. Par exemple, la suite de mots « longueur désirée » respecte le patron SUBC ADJ, et sera extraite, comme candidat terme, cependant, elle ne représente pas un composant contenu dans le graphique.

Le patron ACT-COM

Les manuels techniques étant à visée opératoire, un grand nombre de commentaires sont des descriptions d'actions. Les phrases énoncent, dans ce cas, les actions à effectuer sur la (ou les) composante(s) matérielle(s) contenue(s) dans le graphique.

Nous définissons d'abord le patron ACT qui décrit une action, en général :

ACT=VRB (VRB)* (PREP)*

Exemples

- « Soulevez », « Faites glisser », « appuyez sur », etc.

Le patron ACT-COM permet de détecter les parties de phrases décrivant une action et la composante matérielle à la quelle elle est associée.

$$ACT-COM=ACT (ADV)* ART COM$$

Exemples

- « **Soulevez** le guide d'extrémité (1) »
- « **Secouez** délicatement la cartouche EP»
- « **Ajustez** le guide latéral»

Il s'agit dans ce cas d'exécuter l'action étiquetée par ACT sur la composante étiquetée par COM.

Après avoir défini notre approche pour l'extraction des candidats termes, nous définissons des règles permettant de les filtrer, afin de ne retenir que les termes d'indexation adéquats, et construire l'index du graphique.

1.2.3. Règles de filtrage des candidats termes et construction de l'index

Nous nous intéressons ici au type structurel du bloc textuel considéré, afin d'identifier les index du graphique :

- dans le cas où le bloc textuel est un bloc commentaire, des patrons morphosyntaxiques ont été appliquées au bloc et une liste de candidat termes a été extraite. Nous définissons donc des filtres, qui selon la localisation du candidat terme dans le texte, permet de le retenir comme terme d'indexation du graphique ou de le rejeter.
- dans le cas où le bloc textuel est un bloc titre, un traitement différent est effectué afin d'extraire les index relatifs au type de machine décrite par le graphique, son modèle, la procédure à suivre, etc.

a. Le texte est un bloc commentaire

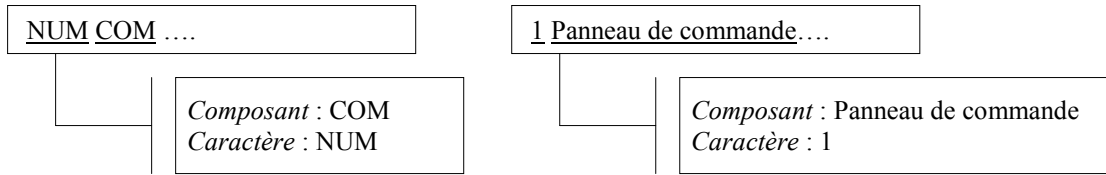
Nous présentons, ici, trois des filtres définis, et qui selon la localisation du patron trouvé, dans le commentaire, permet de valider les termes candidats repérés par le patron en question.

Le patron COM est au début des items d'une liste

Dans ce cas, le commentaire permet de lister les composantes matérielles figurant dans le graphique. Chaque syntagme étiqueté par COM est alors validé comme étant une composante matérielle.

Si la liste est numérotée, les caractères (numéros ou lettres) en début de liste caractérisent le caractère relatif à l'énumération rattachée à la composante.

Exemple

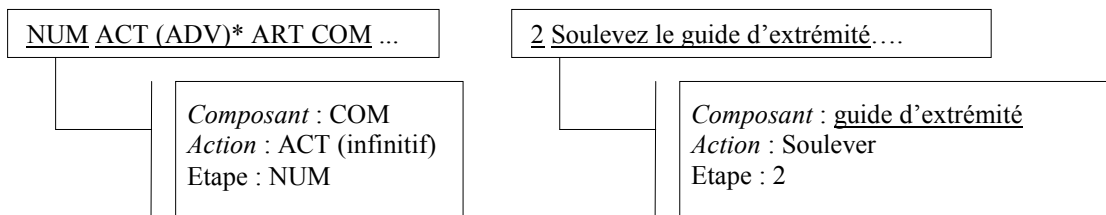


Le patron ACT-COM est au début des items d'une liste

Dans ce cas, le commentaire permet de lister les actions à réaliser sur les composantes matérielles figurant dans le graphique. Chaque syntagme étiqueté par COM et contenu dans un patron ACT-COM est alors validé comme étant une composante matérielle, et le syntagme étiqueté par ACT est l'action à effectuer sur cette composante.

Si la liste est numérotée, les caractères (numéros ou lettres) en début de liste caractérisent l'étape de l'action concernée.

Exemple



Le patron ACT-COM est au début d'une phrase dans un paragraphe classique

Comme pour le cas précédent, chaque syntagme étiqueté par COM et contenu dans un patron ACT-COM est alors validé comme étant une composante matérielle, et le syntagme étiqueté par ACT est l'action à effectuer sur cette composante. L'étape de l'action n'est pas mentionnée dans ce cas.

b. Le texte est un bloc titre

Les graphiques sont rattachés aux titres des sections auxquelles ils appartiennent (apparaissant dans le sommaire). Dans le document que nous étudions, il suffit de considérer le titre du manuel et les titres rencontrés en remontant de deux niveaux dans la hiérarchie, par rapport au graphique, pour retrouver les titres significatifs et spécifiques au graphique à indexer.

Le texte du titre du manuel

Le titre du manuel contient les informations relatives à la machine décrite (que ce soit globalement, ou partiellement) par le graphique, au modèle de la machine et à son constructeur.

Une liste prédéfinie des noms de machines (imprimante, copieur, etc.) concernées par notre cadre applicatif, des constructeurs (xerox, HP, etc.) et modèles existants (N17, stylyus 500, etc.) permet de récupérer ces index à partir du titre du manuel.

Le texte des titres de niveau 1 et 2 en remontant dans la hiérarchie

Nous distinguons ici deux cas selon que le graphique est à visée uniquement descriptive ou opératoire :

Dans le premier cas, le graphique que nous désirons indexer est descriptif. Il décrit uniquement les composantes et leurs propriétés, sans citer d'actions à exécuter sur ces composantes : si des informations intéressantes sont contenues dans les titres, nous n'en tenons pas compte, car ces informations sont aussi mentionnées dans le bloc commentaire (d'où elles seront extraites).

Dans le deuxième cas, le graphique que nous désirons indexer décrit une ou plusieurs actions (des patrons ACT ont été détectés dans le commentaire liste ou paragraphe classique). Les titres rencontrés en remontant dans la hiérarchie sont porteurs d'information pertinente:

- le deuxième titre rencontré (nous le notons TITRE2) correspond généralement à la *procédure* décrite par le graphique concerné,

Exemple : « Chargement du papier »

- le premier titre rencontré (nous le notons TITRE1) correspond généralement au *cas* d'exécution de la procédure.

Exemple : « Utilisation du magasin 1 »

Remarque : si un seul titre est rencontré en remontant dans la hiérarchie (en dehors du titre du manuel), il est considéré comme étant le TITRE2.

Dans différents manuels techniques, une même procédure ne sera pas, forcément, mentionnée de la même façon. Ainsi, dans un document, on parlera de « résolution de problèmes papiers » et, dans un second, on parlera de « bourrages papier », pour désigner une même procédure. Afin d'harmoniser l'index définissant la *procédure* à suivre, nous avons établie une liste générique des différentes procédures que l'on peut rencontrer dans la documentation technique relative à notre cadre applicatif. Ainsi, au lieu d'extraire le texte du titre correspondant à la *procédure* décrite par le graphique, nous cherchons dans la liste établie, le terme adéquat correspondant à cette même procédure et l'utilisons pour indexer le graphique : la correspondance est faite en fonction des mots contenus dans le texte du titre.

1.2.4. Bilan

Afin d'indexer les vues symbolique et opératoire du graphique (index textuel), nous avons proposé une approche qui se base sur l'application de patrons morphosyntaxiques pour l'extraction de candidats termes à partir des blocs commentaires, ainsi qu'un filtrage de ces candidats fondé sur leur localisation dans le bloc et une extraction d'autres termes à partir des blocs titres.

Nous décrivons, dans ce qui suit, le processus qui implémente cette approche.

1.3. Processus automatique d'indexation

1.3.1. Outils de TALN utilisés

Avant de décrire notre processus automatique d'indexation, nous présentons ici les outils de traitement automatique de la langue naturelle que nous avons utilisés.

a. GATE

GATE est une plate-forme d'ingénierie linguistique [Cunningam,02] qui repose sur l'application successive de transducteurs¹ aux textes. Conformément aux termes employés par ses concepteurs, nous parlons ici de ressources de traitement (Processing Resources : PR). Ces ressources de traitement utilisent le texte² modifié par les ressources précédemment appliquées pour ajouter de la structure au texte. Les ressources de traitement les plus courantes sont les segmenteurs (Tokenizers), les analyseurs morpho-syntaxiques (Part Of Speech ou POS Taggers), les dictionnaires (Gazetteers), les transducteurs (JAPE transducers), et les patrons d'extraction (Templates). Ils sont appliqués au texte au sein d'une cascade (chaîne de traitement ou pipeline).

GATE peut être utilisé de deux façons différentes : environnement de développement ou bibliothèque.

L'utilisation la plus simple est comme environnement de développement au travers des ressources développées par ses concepteurs. L'environnement de développement est constitué d'une interface graphique permettant aux utilisateurs de créer de nouvelles ressources ou paramétrer des ressources disponibles et de les appliquer aux textes au sein d'une chaîne de traitement. Nous avons utilisé cette interface pour tester nos modules d'extraction de termes et pour visualiser les résultats (voir Figure 60).

Le second niveau d'utilisation consiste à tirer parti des ressources disponibles dans le système. Nous pouvons les utiliser au sein de l'environnement de développement ou de façon embarquée dans des applications autonomes. De cette manière, il est possible de se passer de l'interface graphique de GATE et de traiter un texte dans un programme autonome hors de l'environnement de développement. C'est l'utilisation pour laquelle nous avons opté, nous avons ainsi bénéficié de l'architecture et de la bibliothèque de GATE afin d'intégrer nos différents modules et afin de les appliquer en chaîne sur les textes.

¹ Un transducteur est un automate à états finis qui, pour chaque état parcouru, produit une ou plusieurs informations

² Le texte est transformé en un arbre XML afin de permettre le partage d'informations entre les différents modules

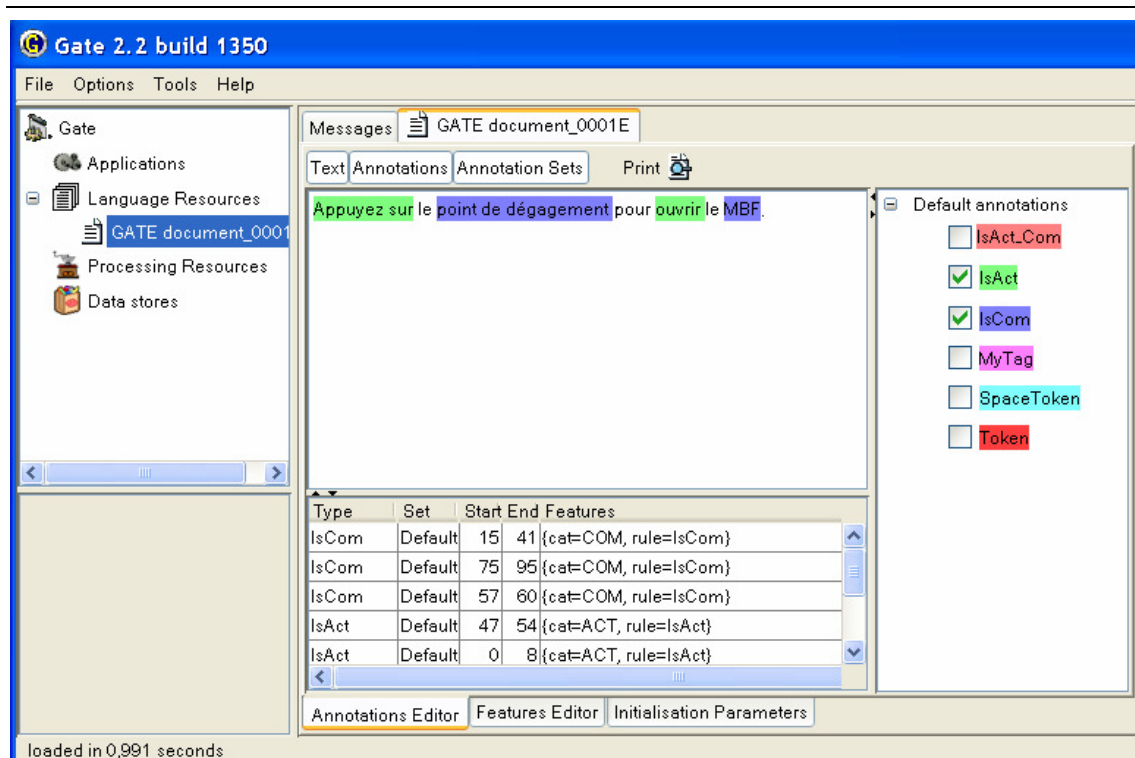


Figure 60 Exemple d'utilisation de l'interface graphique de GATE (repérage des termes respectant les patrons COM et ACT)

b. TreeTagger

TreeTagger est un outil développé au sein de l'institut de linguistique computationnelle de l'université de Stuttgart [Schmid,94]. Cet outil, utilisé pour l'étiquetage des textes en français et en anglais, spécifie pour chaque mot sa catégorie syntaxique et indique son lemme. L'estimation de la catégorie grammaticale d'un mot se base sur la construction récursive d'arbres de décisions binaires et un calcul de probabilité.

Nous avons intégré cet outil dans la plate-forme GATE par le biais d'un traducteur (wrapper) que nous avons développé, et nous l'avons utilisé pour l'étiquetage grammatical et la lemmatisation des textes. Les informations concernant chaque mot sont intégrées dans la structure XML du document texte.

Notons ici que GATE propose son propre étiqueteur grammatical mais ce dernier ne calcule pas la forme lemmatisée des mots, information indispensable dans notre processus.

c. JAPE : un langage d'expression de grammaires pour le TALN

Le langage JAPE (*Java Annotation Patterns Engine*), une variante du standard CPSL adapté au langage de programmation Java [Cunningham,02], a été proposé pour écrire des grammaires qui, une fois appliquées sur le texte, permettent d'y ajouter des informations en forme d'annotations.

Une grammaire de JAPE comporte un ensemble de phases, chaque phase étant un ensemble de règles sous forme de patron et action. Dans une règle, si les patrons sont satisfaits, alors une action (ex. attribution d'une étiquette d'annotation) pourra être déclenchée. La partie droite

d'une règle qui contient des patrons doit être écrite en JAPE mais la partie gauche, qui contient les actions, peut être écrite en JAPE ou en JAVA.

Voici, ci-dessous, un exemple général d'une grammaire JAPE permettant de repérer les adresses IP dans le texte.

```
Rule: IPAddress
( {Token.kind == number} {Token.string == "."}
  {Token.kind == number} {Token.string == "."}
  {Token.kind == number} {Token.string == "."}
  {Token.kind == number} )
: ipAddress --> :ipAddress.Address = {kind = "ipAddress"}
```

Etant donné que GATE dispose d'un transducteur permettant l'application des grammaires JAPE sur le texte, nous avons utilisé ce langage pour écrire nos grammaires de détection de candidats termes.

1.3.2. Processus

Le processus d'indexation automatique que nous avons construit, et dont l'architecture est schématisée dans la Figure 61, utilise des règles pour générer l'index des graphiques à partir des commentaires textuels les accompagnant.

(1) L'étiqueteur grammatical TreeTagger effectue une analyse linguistique sur la collection de commentaires pour générer une collection étiquetée.

(2) L'outil de TAL GATE récupère cette collection. En appliquant les patrons syntaxiques définis précédemment (a), on extrait un ensemble de termes d'indexation candidats.

(3) Le générateur d'index récupère ces termes candidats afin de générer l'index du document selon le modèle GRIM. Les règles (b) de composition et de localisation des termes récupérés que nous avons définies (exemple : en début de phrase...) permettent de sélectionner les termes à retenir et de les structurer dans l'index.

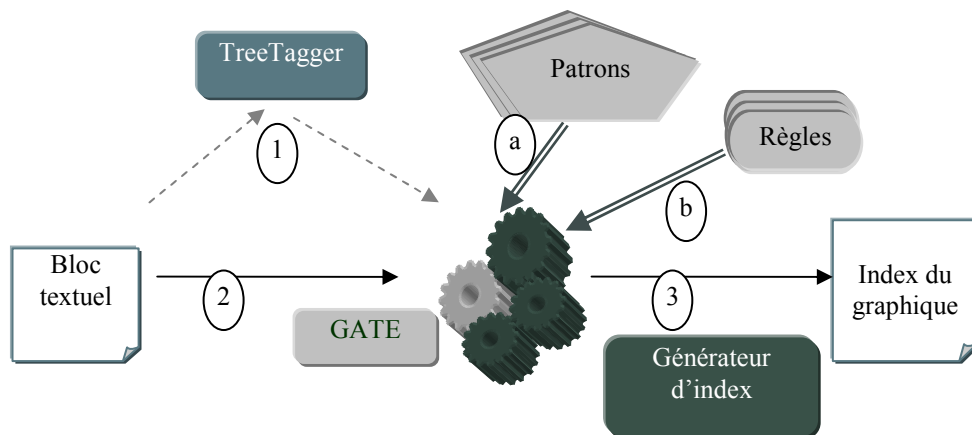


Figure 61 Architecture de notre processus d'indexation

1.4. Validation qualitative de l'indexation par le texte

1.4.1. Proposition de mesures qualitatives pour indexations complexes

Nous avons mentionné dans la première partie de ce manuscrit, des mesures de qualité d'indexation proposées pour des langages d'indexation à base de mots clés [Soergel,94]. Il s'agit de mesures concernant:

- l'exactitude : elle est mesurée en terme de complétude et de pureté. Ces deux mesures calculent des taux par rapport à une indexation de référence : les termes qui devrait être affectés ou rejetés par le processus d'indexation. Des mesures qui calculeraient des taux par rapport aux termes réellement affectés par ce processus n'apparaissent pas. Pourtant, dans le cas où le processus d'indexation n'utilise pas de vocabulaire prédéfini, il est difficile de calculer la pureté d'une telle indexation. Estimer l'absence d'erreurs d'indexation par des termes incorrects peut difficilement être mesuré par la pureté. Il manque une mesure liée à l'absence de descripteurs erronés affectés au document qui serait définie par rapport à ce qui a effectivement été indexé.
- la consistance : elle est mesurée en terme d'accord entre indexeur(s). Nous pensons toutefois, que les mesures proposées peuvent être biaisées par des accords aléatoires entre indexeurs.

Ces mesures qui sont adaptées à des langages simples ne le sont pas pour des langages structurés comme celui que nous avons proposé. D'autres mesures ou une adaptation de ces dernières doivent alors être fournies dans le but de permettre l'évaluation de processus d'indexation fondée sur des langages structurés. Nous présentons ici (i) deux autres mesures que nous proposons d'ajouter aux mesures existantes et qui nous semblent plus adaptées, et (ii) une adaptation de ces mesures afin qu'elles conviennent à des langages structurés. Nous nous intéressons au cas où l'entité serait un terme étiqueté (voir aussi [kefi,05]).

a. Proposition de nouvelles mesures qualitatives

Les mesures d'exactitude proposées sont toutes deux calculés par rapport à une indexation de référence : les termes qui devrait être affectés au document ou rejetés. Dans le cas où le processus d'indexation n'utilise pas de vocabulaire prédéfini, il est difficile de calculer la pureté d'une telle indexation. Estimer l'absence d'erreurs d'indexation par des termes incorrects peut difficilement être mesuré par la pureté. Nous proposons donc une mesure liée à l'absence de descripteurs erronés affectés au document qui serait définie par rapport à ce qui a effectivement été indexé. Nous proposons une telle mesure que nous désignons par le terme *justesse*.

D'un autre côté, pour mesurer la consistance, nous proposons d'utiliser une autre mesure qui permettrait de chiffrer l'accord entre deux ou plusieurs indexeurs et qui tiendrait aussi compte de la concordance aléatoire entre les jugements. Nous avons utilisé le test non paramétrique Kappa proposé par [Cohen,60].

- La justesse

La justesse est liée à l'absence d'erreur d'excédent. Considérée du point de vue document, la justesse renvoie à « parmi les termes affectés au document, combien sont corrects? » et du point

de vue terme, elle renvoie à « Parmi tous les documents indexés par le terme, combien le sont correctement ? ». Nous mesurons :

$$\text{Justesse}(D) = \frac{\text{nb termes correctement affectés à } D}{\text{nb total de termes affectés à } D}$$

$$\text{Justesse}(t) = \frac{\text{nb documents correctement indexés par } t}{\text{nb total de documents indexés par } t}$$

Cette mesure, qui correspond à la *précision*, est intéressante pour prédéterminer les performances du système (la précision mesure le rapport entre les documents pertinents et retrouvés et les documents retrouvés).

- Le test non paramétrique Kappa pour la consistance

Ce test généralement utilisé dans les études de reproductibilité dans le domaine biomédical permet d'estimer, en prenant en compte la concordance aléatoire, l'accord entre des jugements catégoriels appliqués aux mêmes objets, fournis par deux ou plusieurs observateurs ou techniques dans le but de déceler et de quantifier les désaccords pour les corriger [Cohen,60]. Ce test permet de chiffrer l'accord entre deux ou plusieurs observateurs lorsque les jugements sont qualitatifs, contrairement au coefficient de Kendall [Kendall,71] par exemple, qui évalue le degré d'accord entre des jugements quantitatifs. Appliqué à l'évaluation des processus d'indexation, ce test permet d'estimer l'accord entre deux indexeurs (humains ou informatiques) et donc de mesurer la consistance inter-annotateurs. Il permet aussi d'estimer la concordance entre l'indexation, par un même indexeur, d'un même document lors de sessions différentes ou de plusieurs documents équivalents, et donc de mesurer la consistance intra-annotateur.

Notons que le coefficient de Kappa a le même but que la mesure de consistance définie précédemment. Elle est néanmoins plus précise, puisqu'elle corrige l'accord observé par les effets dus au hasard.

Sans entrer dans les détails, le coefficient Kappa s'écrit $K = \frac{P_0 - P_e}{1 - P_e}$ avec :

- P_0 : proportion d'accord observée pour les termes affectés à D, d'un point de vue document D, ou pour les documents indexés par t, d'un point de vue terme t.
- P_e : proportion d'accord aléatoire pour les termes affectés à D, d'un point de vue document D, ou pour les documents indexés par t, d'un point de vue terme t.

De façon générale, l'accord entre indexeurs est considéré comme bon lorsque le coefficient de Kappa dépasse 0,61 et comme excellent lorsqu'il dépasse 0,81.

Récapitulatif des mesures

Le Tableau 7 reprennent les mesures existantes et celles que nous proposons pour mesurer la qualité d'une indexation à base de mots clés, du point de vue document. Dans la suite, nous nous intéressons aux mesures de complétude, et de justesse pour l'exactitude, ainsi qu'au test de kappa pour la consistance. Nous nous limitons au point de vue document, le point de vue terme pouvant être déduit par analogie.

Mesures de l'exactitude		Mesures de la consistance	
Complétude	nb termes correctement affectés au document	Consistance inter-indexeur	nb termes affectés au document par les indexeurs A et B
	nb termes qui devraient être affectés au document		nb termes affectés au document par les indexeurs A ou B
Pureté	nb termes correctement rejetés pour le document	Kappa	$\frac{ P_0 - P_e }{ 1 - P_e }$
	nb termes qui devraient être rejetés pour le document		
Justesse	nb termes correctement affectés au document		
	nb total de termes affectés au document		

Tableau 7 Tableaux récapitulatifs des mesures (anciennes et ajoutées) de qualité d'indexation

b. Adaptation des mesures aux indexations à langage complexe

Contrairement à un langage à base de mots clés représenté par un ensemble de termes, un langage d'indexation complexe est un ensemble d'entités structurées. Nous nous intéressons plus particulièrement au cas où une entité est un terme étiqueté, puisque c'est à ce type d'entité que nous nous intéressons dans notre instance du modèle général.

Dans ce cas, et de façon simplifiée, nous redéfinissons un langage complexe \mathcal{L} comme étant un 5-uple :

$$\mathcal{L} = \{\mathcal{T}r, \mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{U}\}$$

où, rappelons-le :

- $\mathcal{T}r$ représente l'ensemble de termes du langage,
- \mathcal{E} est l'ensemble des étiquettes,
- \mathcal{R} est l'ensemble des relations,
- \mathcal{T} est l'ensemble des entités (ou termes étiquetés) de $\mathcal{E} \times \mathcal{T}r$, et
- \mathcal{U} est l'ensemble des unités (ou entités en relation) de $\mathcal{T} \times \mathcal{T} \times \mathcal{R}$,

Autrement dit, il s'agit :

- d'un ensemble de termes,
- d'une affectation d'étiquettes aux termes: on parle alors d'entités, et
- d'une affectation de relations entre termes étiquetés : on parle alors d'unités.

Remarque : nous ne parlons pas ici de l'ensemble des identifiants d'entités afin de ne pas compliquer cette partie par des références identifiants-entités. Néanmoins, nous considérons que ce référencement est implicite dans les mesures proposées.

Utiliser les mesures précédemment définies telles quelles pour déduire la qualité d'une indexation complexe reviendrait à ignorer les étiquettes des termes ainsi que les relations entre eux pour finalement considérer l'indexation comme une liste de mots clés. Or il est important de juger aussi la répartition des termes dans la structure de l'index, car même si les termes d'indexation sont corrects, il se peut que les étiquettes qui leur ont été affectées dans la structure ne le soient pas (Les entités sont alors incorrectes). De même, il se peut que les relations posées entre deux termes étiquetés soient erronées (Les unités sont alors fausses).

Etant donné qu'une indexation complexe consiste en des termes, une affectation des termes aux étiquettes (les entités) et d'affectation de relations entre entités (les unités), il est nécessaire de faire porter l'évaluation sur ces trois niveaux :

- Vérifier la qualité d'extraction des termes de $\mathcal{T}r$: il s'agit ici de mesurer l'exactitude des termes indexant le document D, ainsi que la consistance de l'indexation au niveau de ces termes. *Par exemple, « imprimante », « panneau avant », « Appuyer sur », « 4 », etc.*
- Vérifier la qualité d'extraction des entités de \mathcal{T} , ie. la qualité d'affectation des étiquettes aux termes : il s'agit ici de mesurer l'exactitude des entités indexant le document D (vérifier qu'à tous les termes d'indexation correspondent les bonnes étiquettes) ainsi que la consistance de l'indexation au niveau de ces entités. La vérification se fait ici pour chacune des étiquettes. *Par exemple, affecter à « imprimante » l'étiquette « MACHINE », à « panneau avant » l'étiquette « COMPOSANTE », à « Appuyer sur » l'étiquette « ACTION », à « 4 » l'étiquette « ETAPE », etc.*
- Vérifier la qualité d'extraction des unités de \mathcal{U} , ie. la qualité d'affectation des relations entre les entités : il s'agit dans ce cas de mesurer l'exactitude des unités indexant le document D, autrement dit, de vérifier que les relations entre les entités ont bien été posées dans l'index (par type de relation). Il s'agit aussi de vérifier l'accord des annotateurs sur cette extraction (la consistance). *Par exemple, relier (ACTION, Appuyer sur) et (ETAPE, 4) par la relation « ActEtap ». On obtient ainsi l'unité ((ACTION, Appuyer sur), (ETAPE, 4), ActEtap).*

La vérification de la qualité se fait sur trois niveaux afin de savoir exactement, si les mesures ne sont pas satisfaisantes, à quel niveau (extraction des termes, des entités ou des unités) se situe le problème de construction de l'index.

Pour chacun des trois niveaux décrits, nous redéfinissons les mesures de complétude, de justesse et de consistance. Ces mesures sont présentées dans le Tableau 8.

	Exactitude de l'extraction des termes de \mathcal{T}_r	Exactitude de la construction des entités de \mathcal{T} pour chaque étiquette e	Exactitude de la construction des unités de \mathcal{U} pour chaque relation ρ
Complétude	$\frac{\text{nb termes indexant correctement le document}}{\text{nb termes qui devraient indexer le document}}$	$\frac{\text{nb termes correctement étiquetés par } e}{\text{nb termes qui devraient être étiquetés par } e}$	$\frac{\text{nb relations } \rho \text{ correctes entre les entités}}{\text{nb relations } \rho \text{ entre les entités qui devraient être dans l'index}}$
Justesse	$\frac{\text{nb termes indexant correctement le document}}{\text{nb total de termes indexant le document}}$	$\frac{\text{nb termes correctement étiquetés par } e}{\text{nb total de termes corrects étiquetés par } e}$	$\frac{\text{nb relations } \rho \text{ correctes entre les entités}}{\text{nb total de relations } \rho \text{ entre les entités dans l'index}}$
	Consistance au niveau de l'extraction des termes de \mathcal{T}_r	Consistance au niveau de la construction des entités de \mathcal{T} pour chaque étiquette e	Consistance au niveau de la construction des unités de \mathcal{U} pour chaque relation ρ
Kappa	$\frac{P_0 - P_e}{1 - P_e}$ (proportion d'accord sur l'extraction des termes)	$\frac{P_0 - P_e}{1 - P_e}$ (proportion d'accord sur l'étiquetage par e)	$\frac{P_0 - P_e}{1 - P_e}$ (proportion d'accord sur l'affectation des ρ)

Tableau 8 Mesures qualitatives d'indexation à langage complexe

1.4.2. Evaluation qualitative de l'indexation textuelle du graphique

L'indexation que nous proposons est fondée sur un langage complexe comprenant des termes, des entités (termes étiquetés) et des unités (entités reliées), et le processus que nous proposons pour l'extraction des termes et la construction de l'index est automatique. Ces deux points (langage complexe et processus automatique) mis ensemble laissent planer des doutes quand à la qualité de l'indexation. Pourtant, c'est sur une indexation de qualité que se base un système de recherche de qualité. Nous proposons donc de vérifier la qualité de notre indexation, afin de pouvoir l'englober dans le système de recherche.

Notre but ici n'est pas d'évaluer notre indexation par rapport à d'autres, mais de la valider qualitativement afin de l'intégrer dans un système de RI. Nous voulons donc, à partir de textes commentant les graphiques (voir l'exemple donné dans la Figure 62), vérifier que les **termes** d'indexation sont correctement extraits, que l'étiquetage de ces termes par un ensemble d'étiquettes prédéfini (entités) est correct et enfin que l'affectation de relations prédéfinies entre termes étiquetés (unités) est bien établie.

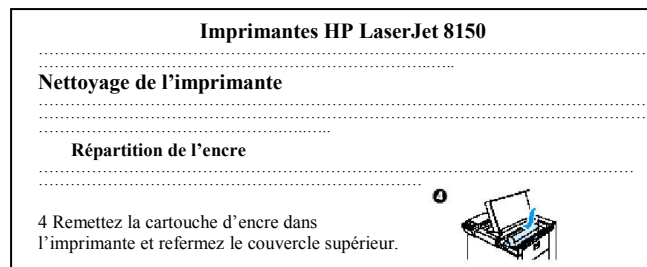


Figure 62 Exemple de document du corpus

Rappelons que nous nous intéressons dans cette partie aux vues symbolique et opératoire du modèle GRIM qui est un sous ensemble du langage décrit Chapitre VI. 2. Nous y incluons aussi à la relation de composition CONTIENT, définie dans la vue structurelle, qui peut être extraite du texte.

Remarques

-Nous ne nous sommes pas intéressés aux éléments illustratifs car ils sont visuels et ne peuvent être extraits du texte.

-Nous ignorons les relations d'ordre de la vue opératoire que notre processus ne permet pas d'extraire.

-Afin de faire le lien entre la vue symbolique et la vue opératoire sans faire intervenir la vue structurelle, nous définissons une relation *ComAct* faisant le lien entre une composante (une entité étiquetée par *Composant*) et l'action qui lui correspond (une entité étiquetée par *Action*) (voir Figure 63)

-Afin de faire le lien entre composante (une entité étiquetée par *Composant*) et le caractère relatif à l'énumération qui la désigne (une entité étiquetée par *Caractère*) sans faire intervenir la vue structurelle et l'entité étiquetée par *Illustratif*, nous définissons une relation *ComCar* faisant le lien entre ces deux entités (voir Figure 63)

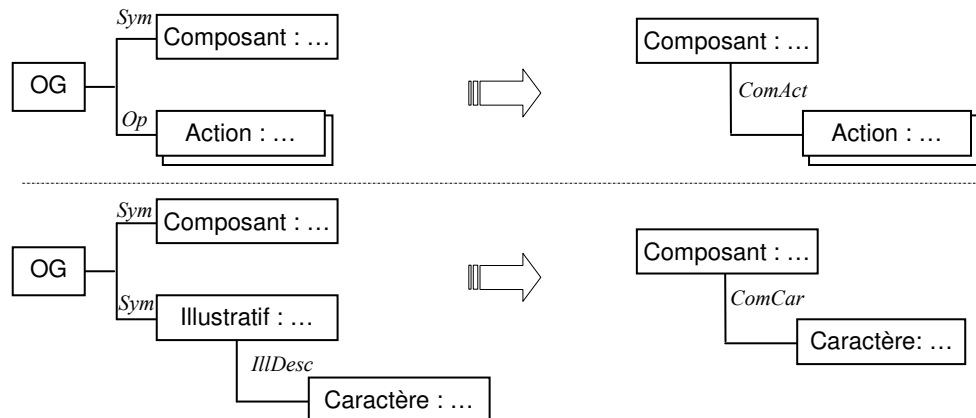


Figure 63 Simplification par suppression de la vue structurelle

La schématisation du sous langage GRIM_{Texte} correspondant aux vues symbolique et opératoire et à leurs éléments que nous pouvons extraire automatiquement à partir du texte sont schématisés dans la Figure 64 et décrits dans ce qui suit.

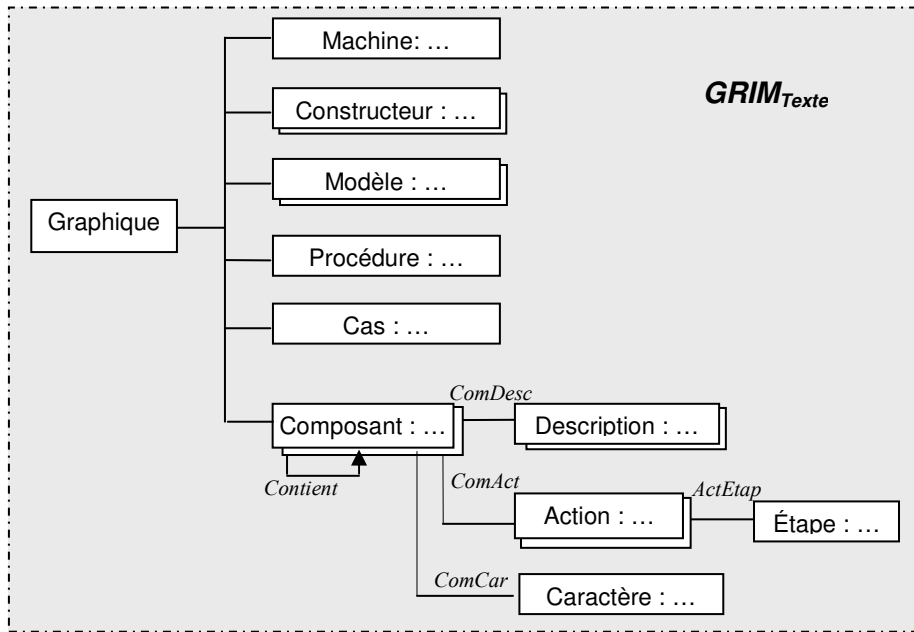


Figure 64 Schématisation du langage $GRIM_{Texte}$

Afin de faciliter la compréhension, nous notons le langage d'indexation qui nous intéresse ici :

$GRIM_{Texte} = \{ \mathcal{T}r_{Texte}, E_{Texte}, R_{Texte}, \mathcal{T}_{Texte}, \mathcal{U}_{Texte} \}$ où :

- L'ensemble des termes $\mathcal{T}r_{Texte} = \mathcal{T}r_{Sym} \cup \mathcal{T}r_{Op} \setminus \mathcal{T}r_{Illustr}$
 $\mathcal{T}r_{Texte} = \{ imprimante, panneau\ avant, appuyer\ sur, 4... \}$
- L'ensemble des étiquettes $E_{Texte} = E_{Sym} \cup E_{Op} \setminus \{ Illustratif \}$
 $E_{Texte} = \{ Machine, Constructeur, Modèle, Composant, Description, Caractère, Procédure, Cas, Action, Étape \}$
- L'ensemble des relations R_{Texte}
 $R_{Texte} = \{ ComDesc, ComCar, ComAct, ActEtap, contient \}$
- L'ensemble des entités $\mathcal{T}_{Texte} = E_{Texte} \times \mathcal{T}r_{Texte}$
- L'ensemble des unités $\mathcal{U}_{Texte} = \mathcal{T}_{Texte} \times \mathcal{T}_{Texte} \times R_{Texte}$

a. Le protocole d'évaluation

Pour évaluer notre processus d'indexation, nous avons procédé en deux étapes :

- Première étape

100 graphiques indexés, extraits de 12 manuels techniques de différents constructeurs, ont été répartis entre 11 personnes (validateurs) à raison de 16 graphiques par personne (certains documents ont été validés par deux validateurs différents). La validation qui a duré une heure

en moyenne par validateur, a été guidée par des questions (voir Figure 65). Pour chaque graphique indexé, nous avons présenté au validateur le graphique et le texte l'accompagnant dans le document technique.

A travers cette étape, nous voulons savoir si le validateur juge, pour chaque graphique, que :

- les termes de $\mathcal{T}_{\text{Texte}}$ extraits pour ce graphique par le processus sont corrects,
- les étiquettes de $\mathcal{E}_{\text{Texte}}$ sont correctement attribués (Les entités de $\mathcal{T}_{\text{Texte}}$ sont correctes), et
- les relations de $\mathcal{R}_{\text{Texte}}$ sont correctement affectées (Les unités de $\mathcal{U}_{\text{Texte}}$ sont correctes).

Nous voulons aussi savoir si les réponses de deux validateurs différents concordent sur ces trois mêmes niveaux.

- Deuxième étape

Nous avons comparé le résultat de l'indexation obtenue avec le processus automatique avec une indexation manuelle générée par un expert (indexation que nous utilisons comme référence). A travers cette comparaison, nous voulons vérifier, pour chaque graphique, si tous les termes de $\mathcal{T}_{\text{Texte}}$ ont bien été extraits, que l'étiquetage, aboutissant à $\mathcal{T}_{\text{Texte}}$, a bien été effectué et que l'affectation de relations, aboutissant à $\mathcal{U}_{\text{Texte}}$, a bien été établie.

- Liens protocole-mesures

Les deux étapes du protocole précédemment présentées nous permettent de mesurer la qualité de notre indexation selon les mesures définies précédemment :

- la justesse de $\mathcal{T}_{\text{Texte}}$, $\mathcal{E}_{\text{Texte}}$ et $\mathcal{U}_{\text{Texte}}$ est mesurée à l'aide des résultats de la première étape,
- la complétude est mesurée à l'aide des résultats de la deuxième étape
- le coefficient de kappa est calculé pour les trois niveaux ($\mathcal{T}_{\text{Texte}}$, $\mathcal{E}_{\text{Texte}}$ et $\mathcal{U}_{\text{Texte}}$) à l'aide des résultats de la première étape pour deux validateurs différents.

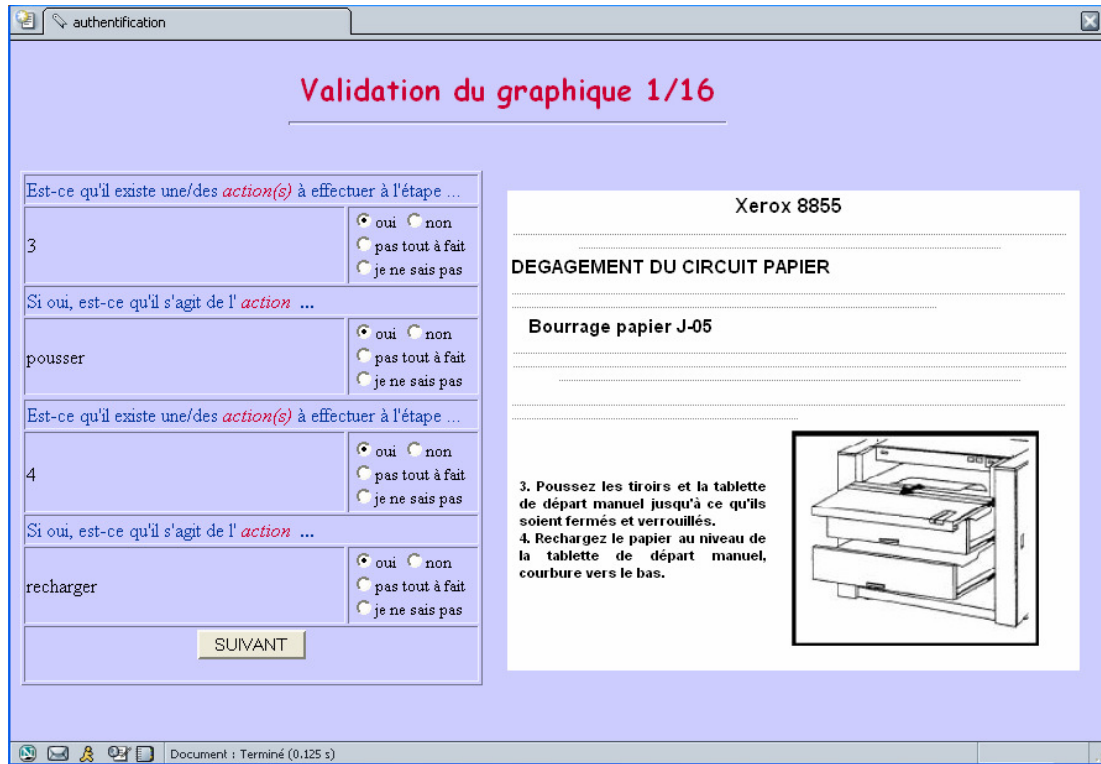


Figure 65 Exemple d'interface de validation de l'indexation

b. Les résultats expérimentaux

L'évaluation que nous avons proposée nous permet d'une part de vérifier si l'on peut faire confiance aux jugements des validateurs et d'autre part de cibler les problèmes, c'est-à-dire à quels niveaux ils se situent. Nous n'avons pas mesuré la complétude de $\mathcal{U}_{\text{Texte}}$ car, dans notre cas, nous sommes intéressés de savoir si les relations établies entre les termes extraits sont correctes ou pas et c'est ce que nous vérifions en calculant leur justesse. La complétude ne nous apporte pas d'informations intéressantes ici.

- Les résultats globaux

L'indexation manuelle de l'expert (2^{ème} étape du protocole décrit) a extrait 1297 termes d'indexation. L'indexation automatique en a extrait 1028 dont 946 jugés corrects par les validateurs. Les valeurs obtenues lors de notre évaluation, sont présentées dans le Tableau 9.

	exactitude des termes ($\mathcal{I}_{\text{Texte}}$)	exactitude des entités ($\mathcal{I}_{\text{Texte}}$)	exactitude des unités ($\mathcal{U}_{\text{Texte}}$)
Complétude	0.73	0.71	-
Justesse	0.92	0.96	0.95
	Consistance des termes ($\mathcal{I}_{\text{Texte}}$)	Consistance des entités ($\mathcal{I}_{\text{Texte}}$)	Consistance des unités ($\mathcal{U}_{\text{Texte}}$)
Kappa	0.92	0.95	0.99

Tableau 9 Résultats globaux de l'évaluation

- Le coefficient de kappa

Nous avons calculé les mesures de l'exactitude par rapport à la similitude entre l'indexation produite par le processus automatique et les jugements d'annotateurs humains. Or, s'il y a convergence entre l'index et le jugement de l'annotateur, considérer que le processus d'indexation est exact revient à présupposer que le jugement de l'annotateur est *a priori* valide, apte à servir de référence (réalité terrain), ce qui n'est pas vraiment démontré.

La consistance inter-annotateur permet de vérifier la confiance que l'on peut accorder au jugement humain en calculant l'accord entre deux annotateurs. Si les annotateurs sont en accord, il y a de fortes chances que leur jugement soit bon. Les mesures que nous avons effectuées du coefficient kappa nous indiquent que l'accord entre les annotateurs est très bon, ce qui nous permet de valider les résultats obtenus pour l'exactitude.

- L'exactitude

Nous considérons les trois niveaux : extraction des termes de $\mathcal{T}_{\text{Texte}}$, construction des entités de $\mathcal{T}_{\text{Texte}}$ et construction des unités de $\mathcal{U}_{\text{Texte}}$.

Exactitude de l'extraction des termes de $\mathcal{T}_{\text{Texte}}$

Comme le montre le Tableau 9, 92% des termes extraits ont été jugés corrects et seulement 73% des termes corrects ont été extraits. En utilisant dans le processus d'indexation un étiqueteur grammatical, les erreurs qu'il induit entraînent forcément des erreurs d'extraction. Si cela explique la valeur satisfaisante de la justesse, cela n'explique pas totalement la valeur de la complétude. Cette dernière, appliquée à l'ensemble des entités $\mathcal{T}_{\text{Texte}}$, nous permettra de cibler à quelles étiquettes correspondent les termes qui ne sont pas tous extraits.

Exactitude de la construction des entités de $\mathcal{T}_{\text{Texte}}$

Nous précisons que l'étiquetage n'est considéré que pour les termes jugés exacts. Nous remarquons que, comme pour l'extraction des termes, la justesse de l'affectation des étiquettes aux termes jugés exacts est satisfaisante. En effet, la moyenne globale de la justesse est égale à 96% (voir Tableau 9). Par contre, nous constatons que la complétude de l'affectation de certaines étiquettes (ou construction d'entité) est assez mauvaise (voir Tableau 10) comme pour l'étiquetage par *Composant* (la valeur de complétude est inférieure à 60%).

A part des erreurs de construction d'entités, ie. d'affectation termes/étiquettes, d'autres facteurs peuvent réduire les valeurs de complétude. D'un côté, ces valeurs ne sont pas complètement indépendantes des mesures relatives à l'extraction des termes : si un terme correct n'est pas détecté, il ne pourra être étiqueté. D'un autre côté, la valeur 0,58 de complétude de l'étiquetage par l'étiquette *Composant* induit une baisse au niveau de la complétude de l'étiquetage par certaines étiquettes qui lui sont rattachées telle que *Description*, *Caractère*, etc. : si un terme normalement étiqueté par *Composant* (nom de la composante matérielle) n'est pas détecté ou est mal étiqueté, on ne détectera pas les termes qui lui sont rattachés (qui auraient dû être étiquetés par *Description*, *Caractère*, etc.) Néanmoins, la validation des affectations des étiquettes (ou construction des entités), nous a permis de cibler les zones d'intérêts lors de l'étude des index afin de détecter les problèmes au niveau du processus d'indexation.

Nous avons constaté que pour la construction d'entités étiquetées par *Description*, le problème n'était pas très important, car dans tout le corpus, il n'y a que 42 instances de termes étiquetés par cette étiquette parmi lesquels 25 ont été extraites et affectées correctement.

Concernant l'affectation de l'étiquette *Machine* aux termes correspondants, ces derniers ont été correctement extraits et étiquetés. La valeur de la complétude (0,63) vient du fait que pour certains documents où le type de la machine n'est pas mentionné dans le texte, le terme correspondant à cette étiquette a été déduit à partir du graphique par l'annotateur humain.

Nous remarquons que la valeur de la variance de complétude est élevée pour des moyennes basses. Nous expliquons cela par le nombre d'occurrences de termes correspondant à certaines étiquettes dans les documents. En effet, si nous prenons l'exemple de l'étiquette *Description*, les termes lui correspondant apparaissent dans 30 documents 42 fois, ce qui fait une moyenne de 1,4 par document. Donc, prenons le cas où un seul terme apparaît dans le document (ce qui est souvent le cas), si l'affectation à l'étiquette *Description* n'est pas détectée, cela donne une complétude nulle, et si elle est détectée, cela donne une complétude égale à 1. D'où une moyenne basse égale à 0,5 et une variance maximale égale à 1.

		Etiquettes									
		Composant	Description	Caractère	Action	Etape	Constructeur	Modèle	Machine	Procédure	Cas
Moyenne	Complétude	0,58	0,71	0,69	0,87	0,60	0,91	0,72	0,63	0,94	0,84
	Justesse	0,84	0,97	1	0,96	0,99	1	1	1	1	1
Variance	Complétude	0,11	0,49	0,20	0,08	0,13	0,08	0,13	0,23	0,05	0,13
	Justesse	0,06	0,08	0,00	0,02	0,03	0,00	0,00	0,00	0,00	0,00

Tableau 10 Valeurs de l'exactitude de la construction des entités de $\mathcal{T}_{\text{Texte}}$ par type d'étiquette

Exactitude de la construction des unités de $\mathcal{U}_{\text{Texte}}$:

Les valeurs des mesures de justesse pour la construction des unités de $\mathcal{U}_{\text{Texte}}$ (affectation de relations de $\mathcal{R}_{\text{Texte}}$ entre deux entités) sont satisfaisantes (voir Tableau 11). Cela signifie qu'il est assez rare qu'une unité (deux entités reliées) de l'index soit mal construite.

		$\mathcal{R}_{\text{Texte}}$				
		ComDesc	ComCar	Contient	ComAct	ActEtap
Justesse	moyenne	0,99	1,00	0,90	0,92	0,96
	variance	0,08	0,00	0,05	0,02	0,03

Tableau 11 Valeurs de la justesse de la construction des unités de $\mathcal{U}_{\text{Texte}}$ par type de relation

c. Conclusion sur la qualité de notre indexation automatique

Les résultats obtenus indiquent une indexation globalement correcte. Les valeurs élevées du coefficient Kappa prouvent que l'on peut faire confiance à ces mesures. Nous constatons que peu d'index erronés sont extraits (bonne justesse), mais qu'un certain nombre d'index est omis (complétude moyenne). Une analyse plus détaillée des résultats nous a permis de situer précisément les problèmes de notre processus d'indexation. En effet, ceux-ci se situent au niveau de l'extraction des termes pour lesquels la complétude de la construction des entités (l'étiquetage) est moyennement satisfaisante, et donc au niveau de l'extraction des termes relatifs aux noms des composantes matérielles (termes ayant pour étiquette *Nom*), aux caractères qui les désignent (termes ayant pour étiquette *Caractère*) et aux étapes d'exécution des actions (termes ayant pour étiquette *Etape*).

2. Indexation visuelle : proposition d'un protocole d'évaluation

Dans la première partie de ce chapitre, nous nous sommes penchés sur l'indexation automatique du contenu sémantique du graphique en exploitant le texte qui l'accompagne dans le document technique. Dans cette deuxième partie, nous nous intéressons à l'indexation manuelle du contenu visuel du graphique, indexation qui est, rappelons-le, fondée sur la mémoire de l'utilisateur. C'est en combinant ces deux types d'indexation que nous obtenons l'index global du graphique.

Nous tenons à signaler, que si l'indexation textuelle a été menée jusqu'au bout (définition d'un processus automatique et validation qualitative), ceci n'est pas le cas de l'indexation visuelle. Cependant, même si nous n'avons pas disposé du temps nécessaire pour mettre en œuvre l'évaluation, nous proposons néanmoins le protocole expérimental la décrivant.

2.1. Le processus d'indexation manuel

L'indexation visuelle du graphique repose sur la mémoire de l'utilisateur. Et comme nous l'avons mentionné précédemment, la mémoire ne récupère pas tout dans le graphique, mais elle récupère assez d'informations pour pouvoir l'identifier parmi d'autres. Nous avons donc posé, lors de la définition du modèle d'indexation de la vue mémoire visuelle, des approximations concernant les types d'objets à retenir dans l'indexation (a), ainsi que leur caractéristiques (b). C'est en se basant sur ces approximations, rappelées ci-dessous, que nous indexons nos graphiques.

a. Types d'objets à retenir

Nous retenons dans l'indexation des graphiques trois types d'objets que les utilisateurs devraient mémoriser. Il s'agit des objets suivants :

- les objets illustratifs tels que les zooms, les flèches, etc.
- les objets de grandes tailles,
- les objets importants mentionnés dans le texte accompagnant le graphique.

b. Caractéristiques des objets graphiques

Pour chaque objet que nous retenons dans l'indexation, nous lui affectons les valeurs des quatre propriétés définies dans la partie 3 décrivant le modèle d'indexation GRIM. Il s'agit

d'approximer leur forme, leur taille, leur position et leur direction. Nous rappelons ici ces approximations :

- La forme est choisie dans l'ensemble *Formes* :

$Formes = \{ \text{cylindre}, \text{cube}, \text{carré}, \text{losange}, \text{triangle}, \text{cercle}, \text{crochet}, \text{double crochet}, \text{parabole}, \text{carré avec point}, \text{carré avec croix}, \text{carré empilé}, \text{carré empilé avec point}, \text{carré empilé avec croix} \}$

- La taille est choisie dans l'ensemble *Tailles* :

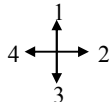
$Tailles = \{ \text{très_petit}, \text{petit}, \text{moyen}, \text{grand}, \text{très_grand} \}$

- La position est choisie dans l'ensemble *Positions*, selon les cases où l'objet se situe sachant que le graphique s'inscrit dans un rectangle quadrillé :

$Positions = \mathcal{P}(\{1, 2, 3, 4, 5, 6, 7, 8, 9\})$,

1	2	3
4	5	6
7	8	9

- La direction est choisie dans l'ensemble *Directions* :

$Directions = \{1, 2, 3, 4\}$ correspondant aux quatre directions 

L'index visuel d'un graphique est, ici, un ensemble d'objets, chaque objet étant défini par quatre propriétés relatives à sa forme, sa position, sa taille et sa direction.

2.2. Description de l'évaluation

Nous décrivons, ici, le protocole expérimental que nous proposons pour valider l'indexation visuelle du graphique.

2.2.1. Objectifs

L'évaluation de l'indexation que nous souhaitons mener a pour objectif de vérifier le bien-fondé des approximations posées pour construire l'index des graphiques, autrement dit de vérifier que cet index tel que nous l'avons défini reflète bien son encodage dans la mémoire de l'utilisateur. Cela revient à vérifier :

- si tous les objets graphiques retenus dans notre index sont corrects, et si aucun objet n'a été omis,
- si les caractéristiques affectées aux objets retenus sont adéquates (représentatives de ce dont se souvient l'utilisateur),
- si la modification d'un objet graphique influe sur la représentation globale du graphique (est-ce que l'utilisateur reconnaît quand même le graphique ou non?)

2.2.2. Corpus et utilisateurs

Les utilisateurs sont des réparateurs qui ont l'habitude du vocabulaire utilisé dans la documentation technique qu'ils ont l'habitude de consulter.

Notre corpus de test est constitué d'un ensemble de graphiques que nous avons indexés manuellement selon le modèle GRIM. Et pour chaque graphique indexé, nous associons un ensemble d'index altérés, c'est à dire modifiés légèrement par rapport à l'index initial (la

suppression d'un objet, la modification d'une forme, la modification d'une taille, la modification d'une position et la modification d'une direction).

Pour l'évaluation, nous proposons à chaque utilisateur :

- un « mini-document » construit à partir de pages sélectionnées dans 12 manuels techniques. Il contient des graphiques du corpus et leurs commentaires tels qu'ils apparaissent dans les documents.
- une liste de tâches rattachées au mini-document Chaque tâche (comme par exemple, « *dégager un bourrage papier dans le panneau avant de l'imprimante NI7* ») renvoi à des graphiques et leurs commentaires contenus dans le mini-document.

2.2.3. Déroulement

Étant donné que la mémorisation joue un rôle important dans cette évaluation, l'expérience se déroulée en trois temps, étalés sur sept jours.

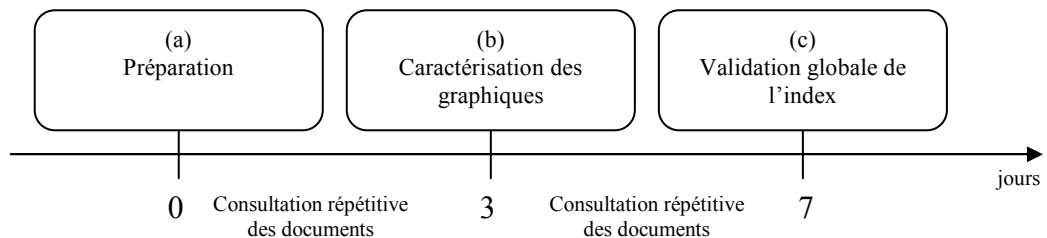


Figure 66 *Les trois étapes de l'expérience*

a. Préparation : consultation répétitive des documents

Cette phase a pour but de préparer les utilisateurs et de les familiariser avec les documents afin qu'il aient une mémoire de ces documents.

Chaque utilisateur se voit attribuer un mini-document, ainsi que liste des tâches rattachées à ce mini-document. Pendant trois jours, l'utilisateur est mis en situation de travail : il doit, à partir des tâches qui lui ont été attribuées, retrouver le graphique et son commentaire adéquats (qui répondent à son besoin) dans le mini-document et les étudier minutieusement. Cette consultation répétitive des documents, étalée sur une période de trois jours, permet de mettre l'utilisateur dans une situation proche de la réalité, et de le familiariser avec les documents. L'utilisateur a ainsi cette mémoire de ce qu'il a déjà consulté et voudra éventuellement retrouver.

b. Caractérisation des graphiques

Cette phase a pour but de confronter une schématisation des graphiques faite par les utilisateurs avec leur index que nous avons proposés.

Pour cela, à l'issue des trois jours, il est demandé à chaque utilisateur de schématiser certains graphiques contenus dans le mini-document qui lui a été attribué. Il s'agit de graphiques que nous lui avons demandé d'étudier via la liste des tâches.

Afin de viser un graphique particulier, nous montrons à l'utilisateur, pendant un laps de temps inférieur à une seconde, un nombre défini de pages dont une seule contient un graphique qu'il devrait avoir consulté (mémorisé). Nous lui demandons ensuite de le schématiser dans un temps assez court (moins d'une minute).

Outre un cadre pour dessiner le graphique, une deuxième zone permet à l'utilisateur d'indiquer ses doutes, difficultés ou remarques. Notons que le dessin est fait à la main sur un support papier, afin de faciliter la tâche à l'utilisateur, ce qui évite de le perturber par des outils informatiques qu'il ne maîtrise pas forcément.

A l'issue de cette phase, nous obtenons un ensemble de schématisations utilisateurs des graphiques de notre corpus, que nous allons pouvoir confronter avec les index que nous leur avons affectés.

c. Validation globale de l'index

Cette phase a pour but d'étudier les jugements des utilisateurs confrontés à une vision globale de l'index visuel des graphiques.

Pour cela, à l'issue de trois jours supplémentaires de consultation répétitive des documents, il est demandé à chaque utilisateur de visualiser des index, altérés ou non, de graphiques de notre corpus. Notons qu'ils ne connaissent pas forcément tous les graphiques que nous leur montrons. Pour chaque index visualisé, les utilisateurs doivent cocher l'affirmation adéquate parmi :

- « oui, je reconnais ce graphique »,
- « non, je ne reconnais pas du tout ce graphique »,
- « oui et non, je reconnais ce graphique mais certains éléments ne sont pas conforme à ce dont je me souviens »,
- « peut être, je ne sais pas ».

A l'issue de cette phase, nous obtenons pour chaque index, le jugement de l'utilisateur par rapport à l'encodage global présenté.

2.3. Exploitation des résultats de l'évaluation

L'évaluation décrite fournit pour chaque graphique de notre corpus:

- une schématisation effectuée par l'utilisateur,
- un jugement de l'utilisateur concernant son index correct et ses index altérés.

A l'aide des schématisations des utilisateurs, prises comme index de référence, et confrontées à notre indexation, nous vérifions (i) si tous les objets graphiques retenus dans notre index sont aussi ceux retenus par l'utilisateur et (ii) si les caractéristiques affectées, par notre indexation, aux objets retenus coïncident avec celles schématisées par l'utilisateur. Autrement dit, nous mesurons l'exactitude de notre indexation. Et à l'aide des jugements de utilisateurs, nous vérifions (iii) quels sont les critères qui font qu'un graphique reste reconnaissable, autrement dit, pour chaque type de modification effectuée sur un index, nous vérifions, si ce type de modification influe sur la représentation globale du graphique.

(i) Vérification de l'exactitude des objets retenus

Nous mesurons ici la complétude et la justesse pour tous les objets retenus lors de l'indexation sans distinction de type, ainsi que pour chacun des trois types (illustratifs, grands et mentionnés dans le texte) que nous avons identifié, et ce pour cibler, éventuellement, le type d'objet qui pose problème, au niveau de l'indexation. Nous mesurons donc :

- la complétude globale et la justesse globale qui renvoient respectivement à « parmi tous les objets schématisés par l'utilisateur, combien apparaissent dans notre index du graphique ? » et à « parmi tous les objets retenus dans l'index, combien sont aussi schématisés par l'utilisateur ? »
- la complétude et la justesse concernant les objets illustratifs, qui renvoient respectivement à « parmi les objets illustratifs schématisés par l'utilisateur, combien apparaissent dans notre index du graphique ? » et à « parmi les objets illustratifs retenus dans l'index, combien sont aussi schématisés par l'utilisateur ? »
- la complétude et la justesse concernant les objets de grandes tailles qui renvoient respectivement à « parmi les objets de grandes tailles schématisés par l'utilisateur, combien apparaissent dans notre index du graphique ? » et à « parmi les objets de grandes tailles retenus dans l'index, combien sont aussi schématisés par l'utilisateur ? »
- la complétude et la justesse concernant les objets importants mentionnés dans le texte qui renvoient respectivement à « parmi les objets importants schématisés par l'utilisateur, combien apparaissent dans notre index du graphique ? » et à « parmi les objets importants retenus dans l'index, combien sont aussi schématisés par l'utilisateur ? »

(ii) Vérification de l'exactitude au niveau des caractéristiques des objets

Nous mesurons ici la complétude et la justesse de l'affectation de chacune des quatre caractéristiques des objets graphiques, à savoir la forme, la taille, la direction et le sens. Pour chacune de ces caractéristique, nous mesurons la complétude et la justesse globale, puis la complétude et la justesse pour chacune des instances définissant la caractéristique en question (par exemple, pour la taille, nous mesurons la complétude pour les cinq tailles : petit, grand, très petit, très grand et moyen), et ce pour cibler, éventuellement, le type de caractéristique qui pose problème, au niveau de l'indexation.

Nous mesurons donc :

- la complétude au niveau de l'approximation des formes :
 - la complétude et la justesse concernant les formes en général qui renvoient respectivement à « parmi tous les objets schématisés par l'utilisateur, combien apparaissent, décrits par la même forme, dans notre index du graphique ? » et à « parmi tous les objets retenus dans l'index, combien apparaissent décrits par la même forme, dans la schématisation de l'utilisateur ? »

- la complétude et la justesse concernant les formes 'f' ('f' étant une des formes prédéfinies dans notre vocabulaire au § Chapitre VI. 1.3.a) qui renvoient respectivement à « parmi les objets de forme 'f' schématisés par l'utilisateur, combien apparaissent, décrits par cette même forme, dans notre index du graphique ? » et à « parmi les objets de forme 'f' retenus dans l'index, combien apparaissent décrits par cette même forme, dans la schématisation de l'utilisateur ? »
- la complétude au niveau de l'approximation des tailles :
 - la complétude et la justesse concernant les tailles en général qui renvoient respectivement à « parmi tous les objets schématisés par l'utilisateur, combien apparaissent, décrits par la même taille, dans notre index du graphique ? » et à « parmi tous les objets retenus dans l'index, combien apparaissent décrits par la même taille, dans la schématisation de l'utilisateur ? »
 - la complétude et la justesse concernant les tailles 't' ('t' étant une des tailles prédéfinies dans notre vocabulaire au § Chapitre VI. 1.3.a) qui renvoient respectivement à « parmi les objets de taille 't' schématisés par l'utilisateur, combien apparaissent, décrits par cette même taille, dans notre index du graphique ? » et à « parmi les objets de taille 't' retenus dans l'index, combien apparaissent décrits par cette même taille, dans la schématisation de l'utilisateur ? »
- la complétude au niveau de l'approximation des directions :
 - la complétude et la justesse concernant les directions en général qui renvoient respectivement à « parmi tous les objets schématisés par l'utilisateur, combien apparaissent, décrits par la même direction dans notre index du graphique ? » et à « parmi les objets retenus dans l'index, combien apparaissent décrits par la même direction, dans la schématisation de l'utilisateur ? »
 - la complétude et la justesse concernant les directions 'd' ('d' étant une des directions prédéfinies dans notre vocabulaire au § Chapitre VI. 1.3.a) qui renvoient respectivement à « parmi les objets ayant pour direction 'd' schématisés par l'utilisateur, combien apparaissent, décrits par cette même direction dans notre index du graphique ? » et à « parmi les objets ayant pour direction 'd' retenus dans l'index, combien apparaissent décrits par cette même direction, dans la schématisation de l'utilisateur ? »
- la complétude au niveau de l'approximation des positions :
 - la complétude et la justesse concernant les positions en général qui renvoient respectivement à « parmi tous les objets schématisés par l'utilisateur, combien apparaissent, situés dans la même position dans notre index du graphique ? » et à « parmi tous les objets retenus dans l'index, combien apparaissent décrits par la même position, dans la schématisation de l'utilisateur ? »
 - la complétude et la justesse concernant une position 'p' précise ne sont pas calculées en raison du nombre élevé des positions possibles.

(iii) Vérification des types de modifications sur un objet influant sur la représentation globale des graphiques

Nous vérifions, ici, pour chaque type de modification définie (suppression d'un objet, modification d'une taille, modification d'une forme, modification d'une position, modification d'une direction), si cette modification influe sur la représentation globale du graphique, c'est-à-dire si l'utilisateur :

- reconnaît le graphique sans remarquer cette modification,
- reconnaît le graphique mais remarque cette modification,
- ne reconnaît pas le graphique,

Ceci devrait nous permettre de reconnaître les types d'altérations qui sont tolérées et ceux qui ne le sont pas, et d'exploiter ces résultats de façon adéquate.

Remarque : nous pouvons, aussi, aller plus loin, en considérant d'autres types d'altération du graphique tels que la modification de plusieurs caractéristiques d'un même objet ou la modification des caractéristiques de plusieurs objets.

2.4. Conclusion sur l'indexation visuelle

L'indexation visuelle que nous proposons se base sur un processus manuel et sur un modèle fondé sur la mémoire visuelle de l'utilisateur donc sur des approximations que nous avons posées. A travers le protocole expérimental proposé, ici, il nous sera possible de vérifier le bien-fondé de ces approximations, autrement dit de vérifier que l'index tel que nous l'avons défini reflète bien l'encodage des graphiques dans la mémoire de l'utilisateur.

3. Conclusion

Nous nous sommes penché dans ce chapitre sur le processus d'indexation du graphique technique. Ce processus se base sur un enrichissement de la donnée graphique par le contenu des blocs textuels qui l'entourent dans un document technique et sur une approximation des caractéristiques visuelles du graphique et des objets qu'il contient.

Dans le premier cas, nous avons exploité la régularité dans la rédaction des documents techniques, régularité que l'on retrouve autant au niveau de sa structure qu'au niveau de la syntaxe de ses phrases, afin de définir des règles combinant des patrons morpho-syntaxiques et les positions des syntagmes dans le commentaire. Nous nous sommes, ensuite, penchés sur la validation qualitative de notre indexation. En effet, l'indexation étant un processus coûteux et souvent irréversible, nous insistons sur l'importance de sa validation en amont, avant de l'englober dans un système de recherche. Nous avons donc adapté les mesures de qualité existantes afin d'évaluer la qualité du processus d'indexation défini et qui est fondé sur le langage complexe GRIM (plus particulièrement GRIM_{Texte}). Cette évaluation nous a permis, d'un côté, de cibler les erreurs engendrées par notre processus d'indexation, et d'un autre côté, de contredire l'avis souvent répandu qui juge les langages complexes difficilement utilisables car peu fiables qualitativement. En effet, les résultats obtenus lors de l'évaluation étant satisfaisants, nous avons pût prouver que l'indexation à base de langages complexes peut aussi être de qualité et donc fiable, sur le domaine technique qui nous intéresse.

Dans le deuxième cas, nous avons proposé un protocole expérimental qui se base sur la mémoire visuelle de l'utilisateur professionnel, et ce afin de vérifier que l'approximation que nous proposons pour l'indexation visuelle des graphiques techniques est convenable, et donc que cette indexation est de qualité. La mise en place d'une telle expérience nécessitant du temps et surtout la participation d'un groupe d'utilisateurs professionnels, elle n'a pas été effectuée, mais fait partie des perspectives à court terme de ce travail.

Chapitre VIII. Formulation de la requête et correspondance : validation de l'ajout des critères d'obligation et de certitude

Dans le modèle, que nous avons défini dans la partie 2 de ce manuscrit, nous avons proposé une formulation de la requête enrichie par les deux critères d'obligation/option et de certitude/incertitude ainsi qu'une fonction de correspondance tenant compte des contraintes associées à un tel ajout. Nous objectif, dans ce chapitre, est de vérifier le bien fondé de cette proposition, autrement dit, nous voulons:

- vérifier que l'ajout de ces critères améliore la qualité du système, autrement dit, répondre à la question "est-ce qu'une requête sans ces critères donnerait les mêmes résultats"?
- vérifier que la fonction de correspondance telle que nous l'avons définie (injection avec contraintes) est nécessaire, autrement dit, répondre à la question "est-ce qu'il est possible de rendre compte de ces critères avec les modèles existants?"

Afin de répondre à ces deux questions, nous avons mené un ensemble d'expériences, dans le cadre applicatif de la recherche de graphiques technique. Nous nous sommes limité à l'index textuel du graphique, puisque nous avons validé sa qualité, et avons choisi de n'utiliser que les termes de l'index (sans les étiquettes, ni les relations), et ce dans le but de pouvoir comparer notre modèle avec des modèles largement utilisés : les modèles booléen, vectoriel, probabiliste et de langue, modèles dont nous étendons progressivement les fonctions de correspondance pour qu'elles gèrent les critères ajoutés à la requête (voir aussi [kefi,06]).

1. Modèle opérationnel considéré

Nous définissons ici, le modèle que nous nous proposons d'évaluer en définissant les pondérations utilisées (pour l'indexation et la correspondance) permettant de classer les documents renvoyés par notre système, ainsi qu'une solution possible pour le calcul de l'injection maximisant le score d'un document.

Description du modèle de recherche considéré

Dans cette expérimentation, nous nous intéressons uniquement à une partie du modèle que nous avons proposé. En effet, alors que le modèle complet considère la description d'un document ou d'une requête comme un ensemble de termes reliés par des relations, nous nous contentons ici d'une description à base de termes sans prendre en compte les relations. Cette restriction permet de vérifier dans un premier temps le bien fondé de notre approche (prise en compte de critères

supplémentaires) et de vérifier que nous nous positionnons correctement par rapport aux modèles standards de RI.

La Figure 67 donne une vue globale des ensembles entrant en jeu dans le modèle concernés par cette évaluation.

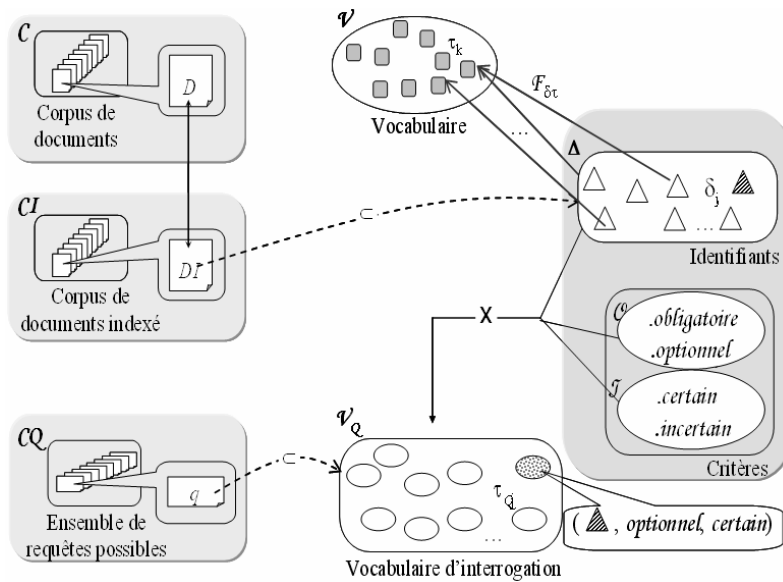


Figure 67 Vue globale du modèle considéré dans l'évaluation

Pondération des termes

Dans le modèle opérationnel, nous affectons un poids aux identifiants d'indexation des documents afin de pouvoir classer les documents retournés, et comparer ainsi les performances de notre modèle aux modèles existants.

Nous définissons le poids de l'identifiant δ_j ainsi:

$$w_{D,j} = \frac{\text{nb occurrences du terme } \mathcal{F}_{\delta_T}(\delta_j) \text{ dans le document } D}{\sum_{\delta_{D,i}} \text{nb occurrences du terme } \mathcal{F}_{\delta_T}(\delta_{D,i}) \text{ dans le document } D}$$

Ce poids reflète l'importance du terme d'indexation dans le document.

Fonction de correspondance

Plusieurs fonctions peuvent être utilisées afin de pondérer les documents retournés. Nous utilisons une variante de la fonction de similarité définie dans [Salton,83], soit:

$$S_1(q, DI, \mathcal{F}_{corresp}) = 1 - \sqrt{\frac{\sum_{\delta_{D,j} = \mathcal{F}_{corresp}^{-1}(\delta_{q,k})} (1 - w_{D,j})^2}{\sum_{\delta_{D,j} = \mathcal{F}_{corresp}^{-1}(\delta_{q,k})} 1}}$$

À cause des phénomènes d'incertitude, plusieurs injections peuvent exister entre (un sous-ensemble des) identifiants de la requête et (un sous-ensemble des) identifiants du document, chacune fournissant un score différent pour le document. Plusieurs choix concernant l'injection à retenir sont possibles : injection minimisant le score, injection maximisant le score, moyenne des scores, etc.

Parmi les documents répondant à une requête donnée, le document le plus « similaire » à cette requête sera celui qui a un maximum de termes (identifiants) en commun avec elle (en considérant aussi l'appariement des termes proches en cas d'incertitude), et encore plus s'ils, décrivant fortement le document. Donc, dans le but de classer les documents les plus « similaires » en haut de la liste des documents retournées par le système, l'injection à retenir parmi toutes celles qui sont possibles est celle qui maximise le score d'un document.

Ceci donne la fonction de similarité:

$$S_2(q, DI) = \max_{\mathcal{F}_{corresp} \in \text{Inj}(q, DI)} S_1(q, DI, \mathcal{F}_{corresp})$$

où :

- $S_1(q, DI, \mathcal{F}_{corresp}) = 0$ si $\mathcal{F}_{corresp}$ ne définit pas une injection
- $\text{Inj}(q, DI)$ est l'ensemble des injections possibles entre la requête q et le document indexé DI .

Les flots sur les réseaux pour une complexité acceptable

Dans notre modèle, un document est pertinent pour une requête s'il existe une injection entre certains de leurs identifiants respectant certaines contraintes, et, comme nous venons de le mentionner, nous cherchons une injection admissible (i.e. respectant les contraintes) qui maximise le score du document.

Le nombre d'injections possibles, a priori, est trop élevé pour qu'une recherche naïve puisse être adoptée. Il est toutefois possible de formuler le problème de la recherche de la « meilleure » injection comme un problème de flots dans des réseaux, car une telle injection définit un alignement de coût minimal. De telles considérations conduisent finalement à un algorithme d'appariement entre requête et document dont la complexité (voir [Ahuja,93]) est de l'ordre de :

$$O((m * \log(\mathcal{N}_d))(m + \mathcal{N}_d * \log(\mathcal{N}_d)))$$

où N_D est la longueur du document, et m le nombre de connections possibles entre les identifiants du document et ceux de la requête. Une telle complexité devient acceptable pour des collections moyennes.

2. Description des expériences

2.1. Collection de test

Les besoins en information proposées dans les collections de test existantes telles que TREC¹ ou CLEF² ne sont pas formulés dans un contexte de recherche où l'utilisateur a une mémoire des documents qu'il souhaite retrouver, ils n'expriment donc pas ses doutes : aucun terme ne peut être considéré comme étant incertain ou optionnel. C'est la raison pour laquelle nous avons construit notre propre collection de test. Même si cette collection est de petite taille, elle nous permet de valider notre approche et dès lors des collections plus larges pourront être construites et d'autres expériences sur ces collections pourront être menées.

La collection que nous avons construite comporte:

- un corpus de 1033 documents extraits de 12 manuels techniques. L'index de ces documents est une liste de termes extraits automatiquement par le processus proposé dans le chapitre précédent,
- 30 requêtes en langage naturel, chacune est traduite manuellement en un ensemble de triplets (identifiant référençant un *terme*, *critère d'obligation*, *critère de certitude*),
- Des jugements de pertinence.

Notons qu'une fonction de proximité a aussi été construite (manuellement).

Remarque

Une requête qui représentée dans notre modèle ne contient que des termes optionnels n'est pas réaliste. Nous posons donc la contrainte qu'au moins un terme de la requête soit obligatoire.

Exemple de requête

Un réparateur veut retrouver les graphiques représentant une "imprimante", probablement la "N17" avec peut être un "panneau avant".

La traduction de cette requête donne:

$q = \{(\delta 1: \text{imprimante, obligatoire, certain}), (\delta 2: \text{N17, obligatoire, incertain}), (\delta 3: \text{panneau avant, optionnel, certain})\}$.

Dans notre interface, l'utilisateur a accès à ces critères, c'est-à-dire qu'il peut entrer directement la requête traduite, telle que celle donnée en exemple (q).

¹ <http://trec.nist.gov/> www.clef-campaign.org

² <http://www.clef-campaign.org>

2.2. Expériences réalisées

Les expériences que nous avons menées et qui sont résumées dans le Tableau 12 sont de deux types : une comparaison interne du modèle proposé et une comparaison avec les modèles classiques de recherche d'information.

Termes Modèles	Tous obligatoires		Obligatoires/optionnels	
	Tous certains	Certains/incertains	Certains/incertains	Tous certains
Proposé	x	x	x	x
Booléen	X	X	x	
Vectorel	X	X	x	
Probabiliste	X	X	x	
De langue	X	X	x	

(1)
(2)
(3)

(b)

Tableau 12. Tableau récapitulatif des expériences réalisées

- une comparaison interne du modèle proposé ((a) dans le Tableau 12), où lors de la correspondance et selon le cas, les critères enrichissant les termes de la requête sont tous deux ignorés ou pris en compte séparément ou conjointement. Les résultats de cette comparaison ont pour but de valider l'apport de l'ajout des deux critères d'obligation/option et de certitude/incertitude dans l'interprétation de la requête.
- une comparaison avec les modèles classiques de recherche d'information, soient les modèles booléen, vectorel, probabiliste et de langue dont nous étendons progressivement la fonction de correspondance pour qu'elles gèrent les critères ajoutés à la requête ((b) dans le Tableau 12). Les résultats de cette comparaison ont pour but de valider la fonction de correspondance telle que nous l'avons définie (injection avec contraintes).

a. La comparaison interne du modèle proposé ((a) dans le Tableau 12)

Dans cette expérience, lors de la correspondance et selon le cas, les critères enrichissant les termes de la requête sont tous deux ignorés ou pris en compte séparément ou conjointement. Nous comparons donc différentes versions de notre modèle correspondant aux différentes combinaisons possibles de prise en compte des critères d'obligation/option et de certitude/incertitude dans les requêtes:

- la fonction de correspondance ne gère ni le critère d'obligation/option ni le critère de certitude/incertitude (tous les termes de la requête sont considérés comme obligatoires et certains),
- la fonction de correspondance gère uniquement le critère de certitude/incertitude (tous les termes de la requête sont considérés comme obligatoires),
- la fonction de correspondance gère uniquement le critère d'obligation/option (tous les termes de la requête sont considérés comme certains),
- tous les termes de la requête sont considérés certains mais une distinction est faite entre les termes obligatoires et ceux optionnels,
- la fonction de correspondance gère le critère d'obligation/option ainsi que le critère de certitude/incertitude (modèle complet).

Les résultats de cette comparaison ont pour but de vérifier l'apport de l'ajout des deux critères d'obligation/option et de certitude/incertitude dans l'interprétation de la requête (en terme de précision du système).

b. La comparaison avec les modèles classiques ((b) dans le Tableau 12).

Il s'agit de comparer notre modèle aux modèles booléen, vectoriel, probabiliste et de langue :

- pour le modèle booléen, la requête est considérée comme une conjonction des termes de la requête, chaque terme étant pondéré par le même poids que dans notre modèle.
- pour le modèle vectoriel et le modèle probabiliste, nous avons utilisé la pondération BM25tf (voir [Robertson,94]),
- pour le modèle de langue, le lissage utilisé est Jelinek-Mercer avec un coefficient $\lambda = 0.2$ (voir [Ogilvie,01])

Pour cette comparaison, nous procédons par la prise en compte progressive des critères en étendant progressivement la fonction de correspondance:

Première expérience (sans extension des modèles)

Cette première expérience ((1) dans le Tableau 12) ne fait intervenir aucun critère : il s'agit d'utiliser les modèles classiques tels qu'ils sont définis à la base. Tous les termes de la requête sont donc considérés comme étant tous obligatoires et certains.

Deuxième expérience (première extension des modèles classiques)

Cette deuxième expérience ((2) dans le Tableau 12) fait intervenir le critère de certitude/incertitude, mais pas le critère d'obligation/option : il s'agit d'une première extension de la fonction de correspondance des modèles classiques.

La procédure est assez simple puisqu'il suffit d'étendre la requête en s'appuyant sur la fonction de proximité. Cette extension ne concerne que les termes incertains.

Troisième expérience (deuxième extension des modèles classiques)

Cette troisième expérience ((3) dans le Tableau 12) fait intervenir les deux critères d'obligation/option et de certitude/incertitude : il s'agit, en plus de la première extension, d'une deuxième extension de la fonction de correspondance des modèles classiques.

Dans le modèle booléen, la procédure est simple : la fonction de correspondance ignore les termes optionnels pour vérifier si un document est pertinent ou pas mais les utilise lors du calcul du poids du document. Ainsi, les documents contenant en plus les termes optionnels se voient affecter un meilleur score que ceux contenant uniquement les termes obligatoires.

Avec les trois autres modèles, la manœuvre est plus délicate car il n'existe pas, à notre connaissance, une extension directe permettant de tenir compte du critère d'obligation/option. Nous avons donc proposé de procéder par fusion de requête. Il s'agit de considérer la requête comme étant la fusion de deux sous requêtes, l'une (q_{ob}) portant sur les termes obligatoires et l'autre (q_{op}) portant sur les termes optionnels. Une combinaison de ces deux sous requêtes est effectuée afin de donner un score global $S(q, D)$ au document D tel que :

$$S(q, D) = \alpha S(q_{ob}, D) + (1-\alpha) S(q_{op}, D),^1$$

S étant la fonction habituelle pour le calcul du score d'un document, et α une variable permettant de favoriser les termes obligatoires par rapport aux termes optionnels. Notons que nous avons utilisé une valeur de $\alpha=0,7$ dans cette expérience.

3. Résultats des expériences

Nous rapportons dans cette partie les résultats obtenus pour chacune des expérience décrites précédemment. Nous commençons par la comparaison interne du modèle proposé, puis nous décrivons les résultats de chacune des comparaisons du modèle proposé aux modèles classiques (modèles sans extension, 1ère extension des modèles et 2ème extension des modèles).

3.1. Comparaison interne

Le Tableau 13 et la courbe de rappel/précision, schématisée, dans la Figure 68 résument les résultats obtenus avec le modèle que nous avons proposé en considérant les quatre combinaisons de prise en compte des critères:

Cas / Précision	Modèle complet	Modèle sans prise en charge des critères	Modèle avec prise en charge du critère de certitude	Modèle avec prise en charge du critère d'obligation
Moyenne	0,658	0,371	0,542	0,528
à 5 docs	0,766	0,566	0,590	0,726
à 10 docs	0,553	0,458	0,361	0,573

Tableau 13. Comparaison interne du modèle proposé

¹ Dans ces expériences $\alpha =0,7$

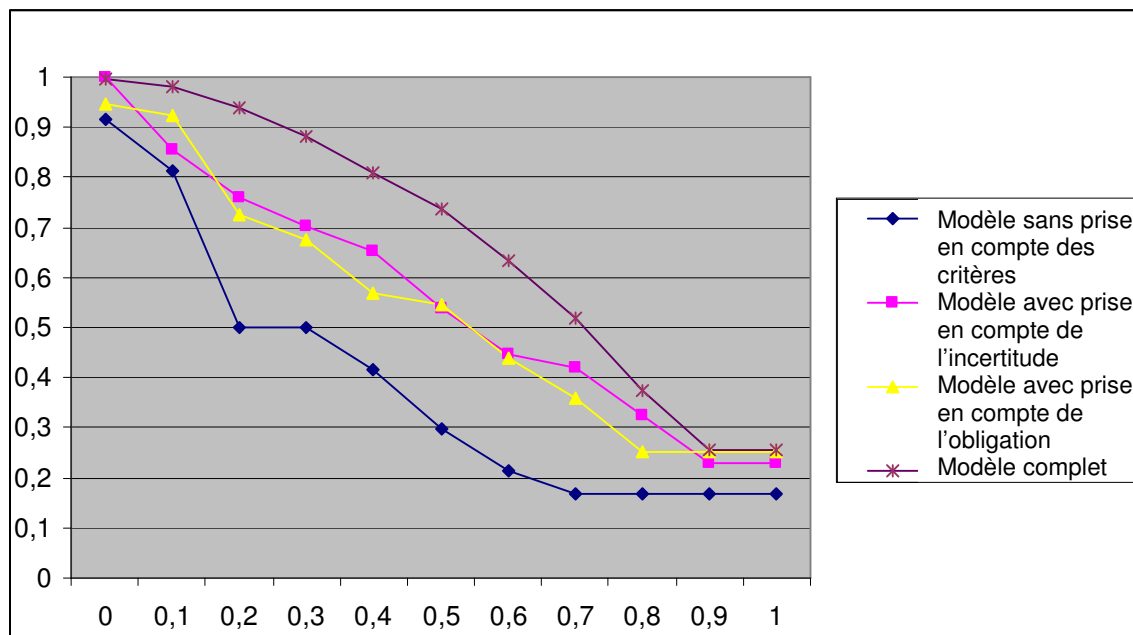


Figure 68 Courbe de rappel/précision pour la comparaison interne

Comme le montrent les résultats, chaque prise en compte d'un critère dans la requête augmente les performances du système en terme de précision :

- la prise en compte du critère d'obligation/option augmente la précision moyenne de 15,7%. En effet, dans le modèle qui ne prend pas en charge ce critère, les documents contenant tous les termes obligatoires et ne contenant pas un terme optionnel ne seront pas jugés comme pertinents par le système, alors qu'ils le sont : les documents retrouvés sont uniquement ceux contenant *tous* les termes qu'ils soient obligatoires ou optionnels.
- la prise en compte du critère de certitude/incertitude l'augmente de 17,1%. En effet, dans le modèle qui ne prend pas en charge ce critère, les documents contenant un terme proche d'un terme incertain de la requête mais ne contenant pas exactement ce terme ne seront pas jugés comme pertinents par le système, alors qu'ils le sont (en supposant qu'ils vérifient les critères sur les autres termes): les documents retrouvés sont uniquement ceux contenant *exactement* les termes de la requête (tels quels) qu'ils soient obligatoires certains ou incertains.
- la prise en compte conjointe des deux critères apporte un gain de 28,7% par rapport au modèle sans prise en compte de ces critères. Ceci nous permet de vérifier que la prise en compte conjointe des critères d'obligation/option et de certitude/incertitude améliore nettement les performances du système de recherche en terme de précision.

3.2. Comparaison avec les modèles classiques de recherche d'information

Nous décrivons ici les résultats des trois expériences menées avec extensions progressives de la fonction de correspondance.

a. Première expérience (Modèles sans extension)

Il s'agit, ici, de la comparaison de notre modèle avec les modèles classiques tels qu'ils existent à la base.

Modèles Précision	Modèle proposé	Modèle booléen	Modèle vectoriel	Modèle probabiliste	Modèle de langue
Moyenne	0,658	0,360	0,407	0,373	0,386
à 5 docs	0,766	0,512	0,606	0,52	0,553
à 10 docs	0,553	0,393	0,573	0,516	0,546

Tableau 14. Précision moyenne et à 5 et 10 documents avec les modèles de base

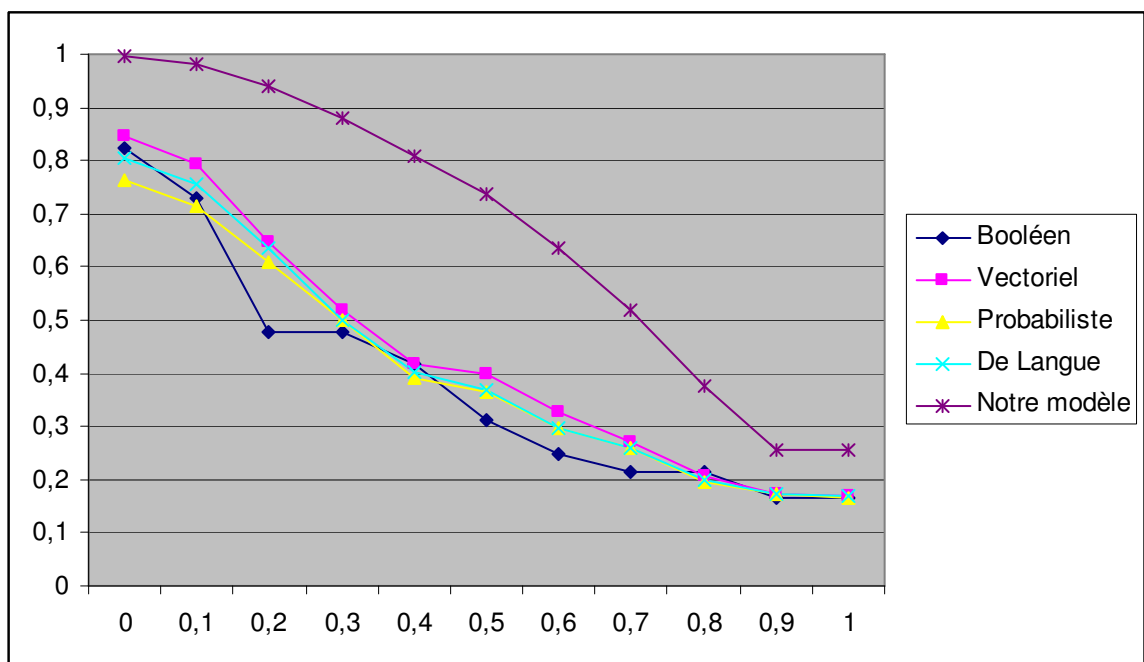


Figure 69 Courbe de rappel/précision pour les modèles de base

Comme le montrent les résultats dans le Tableau 14, les modèles existants utilisés directement tels qu'ils ont été définis donnent de mauvais résultats, la précision moyenne ne dépassant pas la moyenne (0,5). Nous pouvons conclure qu'une utilisation basique des modèles existants ne prend pas compte des critères pris en compte dans la formulation de la requête.

La raison des faibles valeurs obtenues par les modèles de base est que:

- (i) les documents ne contenant pas un terme incertain de la requête mais contenant un de ses proches ne sera pas considéré comme pertinent par ces modèles ou auront un faible score, alors qu'ils peuvent être ceux répondant le mieux à la requête.
- (ii) les documents ne contenant pas un des termes obligatoires de la requête mais contenant tous ses termes optionnels seront renvoyés par le système et ils seront en plus relativement bien classés si les termes optionnels ont une pondération élevée dans le document. Pourtant ces documents ne sont pas pertinents pour la requête.
- (iii) les documents contenant un unique terme qui est mentionné plusieurs fois dans la requête avec les critères obligatoires et certains seront considérés comme étant pertinents par le système, alors qu'ils ne répondent pas au besoin de l'utilisateur.

Une première extension des modèles classiques devrait remédier au premier problème décrit (i) en prenant en charge le critère de certitude/incertitude et une deuxième extension devrait aussi permettre de remédier au second problème décrit (ii) en prenant aussi en charge le critère d'obligation/option.

b. Deuxième expérience : 1ère extension des modèles

Il s'agit ici de la prise en compte du critère de certitude/incertitude par extension de requête dans les modèles classiques.

Modèles Précision	Modèle proposé	Modèle booléen	Modèle vectoriel	Modèle probabiliste	Modèle de langue
Moyenne	0,658	0,468	0,523	0,531	0,538
à 5 docs	0,766	0,561	0,58	0,6	0,62
à 10 docs	0,553	0,4238	0,536	0,566	0,57

Tableau 15. Précision moyenne et à 5 et 10 documents avec les modèles existants étendus afin de prendre en compte le critère de certitude/incertitude

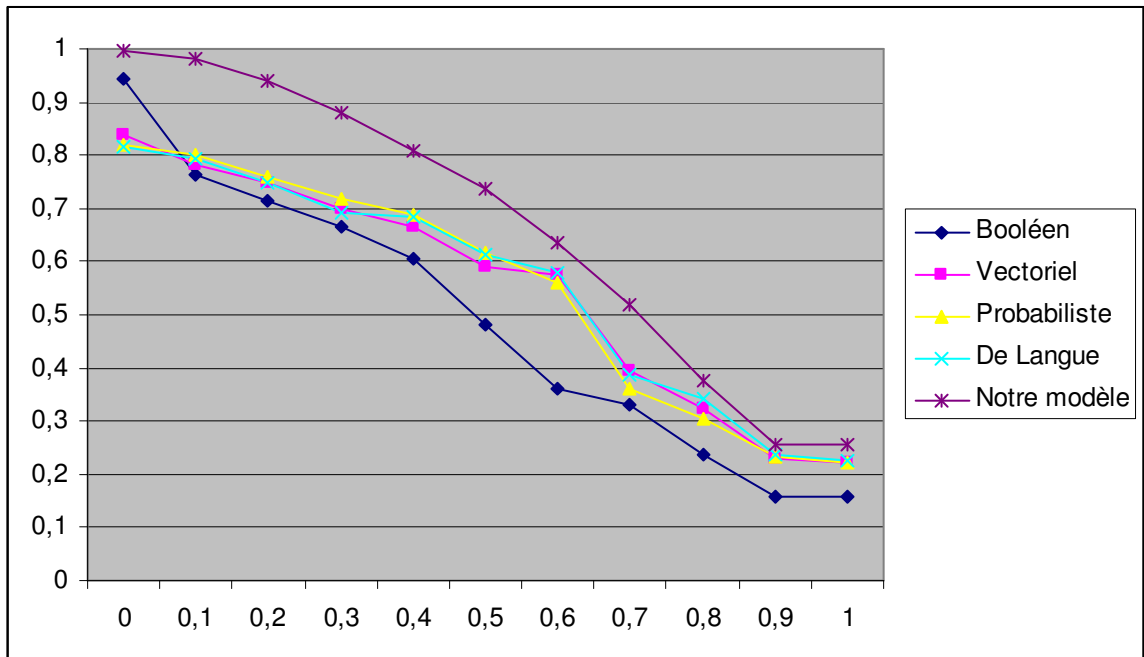


Figure 70 Courbe de rappel/précision pour la première évolution

Comme nous pouvons le constater (voir Tableau 15 et Figure 70), les résultats obtenus sont meilleurs que ceux obtenus, précédemment, avec les modèles de bases, ce qui confirme l'importance de la prise en compte d'un tel critère lors du processus de recherche. Cependant, ils restent relativement bas, dans l'absolu, et en comparaison avec notre modèle.

Si cette première extension permet d'éviter que les modèles classiques jugent non pertinents un document ne contenant pas un terme incertain de la requête mais contenant un de ces proches, alors que ce document répond au besoin de l'utilisateur, il ne permet cependant pas d'éviter les deux problèmes (ii) et (iii) décrits précédemment (§ 3.2.a). Une deuxième extension devrait permettre de remédier au problème (ii) en permettant, avec l'utilisation des modèles classiques, de juger pertinents les documents ne contenant pas des termes optionnels.

c. Troisième expérience : 2ème extension des modèles

Dans cette deuxième extension, les deux critères d'obligation/option et de certitude/incertitude sont gérés par les modèles classiques "étendus" par extension et fusion de requêtes.

Les résultats obtenus sont reportés dans le Tableau 16 et la courbe de rappel/précision correspondante est donnée dans la Figure 71.

Modèles Précision	Modèle proposé	Modèle booléen	Modèle vectoriel	Modèle probabiliste	Modèle de langue
Moyenne	0,658	0,594	0,589	0,603	0,6
à 5 docs	0,766	0,692	0,693	0,72	0,73
à 10 docs	0,553	0,526	0,576	0,606	0,596

Tableau 16. Précision moyenne et à 5 et 10 documents avec les modèles existants étendus afin de prendre en compte les deux critères conjointement

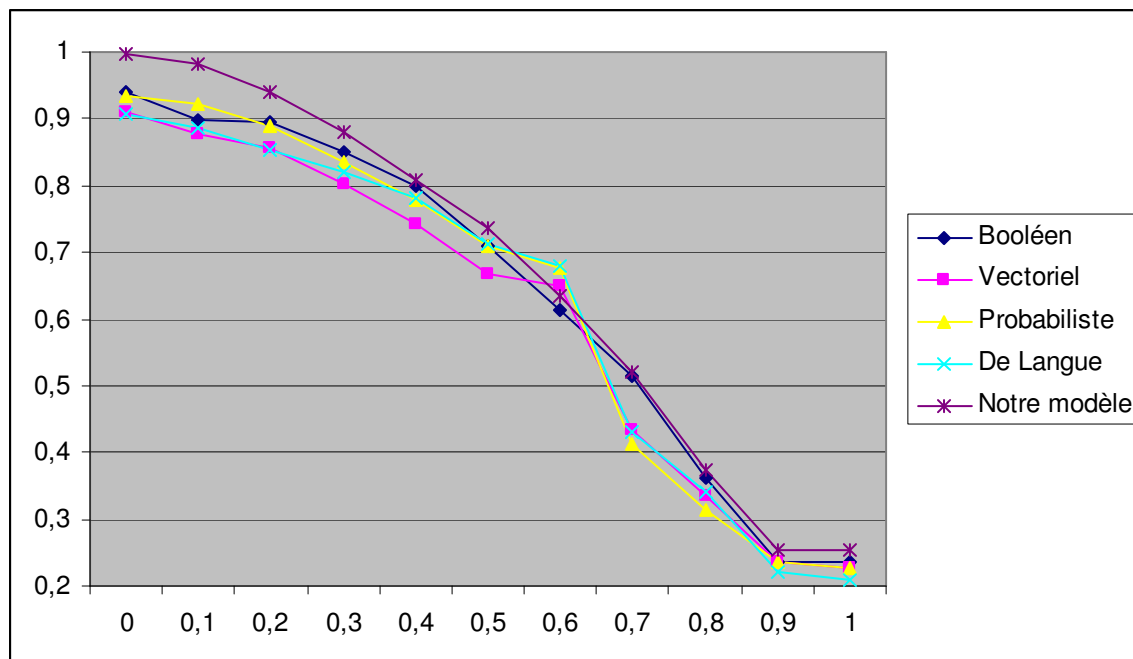


Figure 71 Courbe de rappel/précision pour la deuxième évolution

Les résultats obtenus sont meilleurs que les précédents. Cela nous conforte dans l'idée que la prise en compte des critères d'obligation/option et certitude/incertitude est bénéfique pour les performances de recherche du système, quelque soit le modèle utilisé : gain de 0,23 en moyenne (pour la précision moyenne) par rapport aux modèles de base et gain de 0,1 en moyenne par rapport à la première évolution (prise en compte du critère de certitude).

Les résultats présentés dans le Tableau 16 et la Figure 71 montrent que notre modèle se positionne au-dessus des autres en terme de précision moyenne et de précision à 5 documents, la

différence avec ces différents modèles (le notre et les classiques étendus) étant statistiquement importante en terme de précision moyenne.

En plus du fait notre modèle permet d'obtenir de meilleurs résultats que les modèles classiques étendus, il présente certains avantages par rapport aux autres.

En effet :

- l'utilisation des modèles classiques nécessite l'introduction d'une variable α qui doit être calculée pour chaque nouvelle collection,
- l'utilisation des modèles classiques n'est pas directe et nécessite de poser deux requêtes et de combiner leurs résultats,
- les modèles classiques ne permettent pas de résoudre le problème (iii) décrit dans le § 3.2.a concernant l'utilisation multiple d'un même terme. Ainsi pour une requête de type $q=\{(\delta 1: \text{panneau avant, obligatoire, certain}), (\delta 2: \text{panneau avant, obligatoire, certain})\}$, les modèles classiques, même étendus, retrouveront les documents contenant un unique terme « panneau avant », alors qu'un tel document ne répond pas au besoin exprimé dans la requête q . Cette non prise en compte de l'utilisation multiple d'un même terme dans l'index des documents et dans la requête influe sur la précision du système.
- la deuxième extension des modèles classiques permet de remédier au problème (ii) décrit précédemment, ainsi les documents ne contenant pas des termes optionnels seront jugés pertinents en donnant une importance plus grande aux documents contenant les termes obligatoires par rapport à ceux contenant les termes optionnels. Seulement, cette extension ne permet pas d'obtenir exactement le comportement attendu par l'utilisation du critère d'obligation/option, puisqu'en utilisant les modèles vectoriel, probabiliste et de langue, un document ne contenant pas un terme obligatoire peut se retrouver en tête de liste des documents retournés car les termes optionnels qu'il contient ont un poids élevé. Par exemple, dans le cas où le poids des termes optionnels dans un document $D1$ ne contenant pas les termes obligatoires ($S(q_{ob}, D1)=0$) est largement supérieur aux poids des termes obligatoires et optionnels contenus dans un document $D2$, le document $D1$ pourra être classé avant le document $D2$, alors que $D1$ n'est pas pertinent et que $D2$ l'est. C'est le cas lorsque :
 - ✓ $S(q, D1) = (1-\alpha) S(q_{op}, D1)$ et, $S(q, D2) = \alpha S(q_{ob}, D2) + (1-\alpha) S(q_{op}, D2)$
 - ✓ $(1-\alpha) S(q_{op}, D1) > \alpha S(q_{ob}, D2) + (1-\alpha) S(q_{op}, D2)$
- les modèles classiques ne supportent pas l'ajout de relations entre les termes, relations que notre modèle intègre en proposant de les considérer aussi comme obligatoires/optionnelles et certaines/incertaines, et qui sont nécessaires dans les systèmes orientés précision.

4. Conclusion

Rappelons que l'objectif visé par ces expériences est de répondre aux deux questions :

- "est-ce qu'une requête sans les critères d'obligation et de certitude donnerait les mêmes résultats"?
- "est-ce qu'il est possible de rendre compte des critères d'obligation et de certitude avec les modèles existants?"

La réponse à la première question est « non ». Nous avons pu constater cela lors de la comparaison interne, ainsi qu'en utilisant la version non étendue des modèles existants.

La réponse à la deuxième question n'est pas aussi catégorique, il s'agit plutôt d'un « pas tout fait ». En effet, notre modèle se positionne au dessus des autres (en terme de précision moyenne et de précision à 5 documents) et il présente, en plus, certains avantages par rapport à ces modèles (pas de variable α , gestion des relations, pas d'influence en rapport avec les pondérations des entités optionnelles).

Pour conclure, nous dirons que ces premiers résultats sont une confirmation du bien fondé de notre approche (autant au niveau de l'ajout des critères qu'au niveau du modèle en lui-même). D'autres expériences sur un corpus de test incluant les relations entre termes restent alors à mener.



Conclusion et perspectives

1. Synthèse et contributions

Rappelons, tout d'abord, l'approche proposée au début de ce manuscrit (voir Figure 72) et amenant au travail présenté dans le cadre de cette thèse.

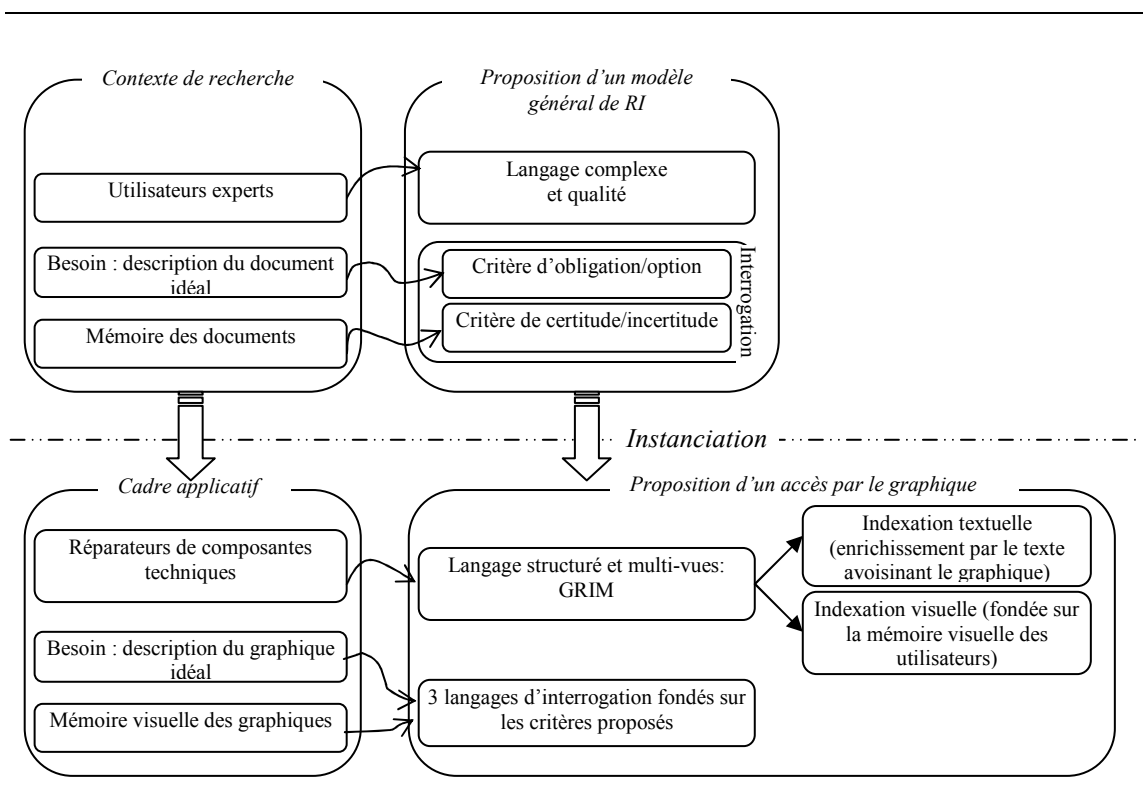


Figure 72 Schéma récapitulatif de l'approche

Ce travail s'intéresse à la recherche d'information dans un contexte particulier de recherche où :

- la formulation de la requête est une description du document 'idéal' recherché par l'utilisateur. L'utilisateur décrit donc le document qu'il souhaite retrouver en précisant ce qui est important ou moins important que ce document contienne (critères d'obligation ou d'option),
- les utilisateurs connaissent les documents et ils en ont donc un souvenir plus ou moins fiable. Lors de la formulation du besoin, la description du document idéal peut alors être la description du souvenir qu'à l'utilisateur du document recherché,

avec tout ce que cela comporte comme doutes (critères de certitude ou d'incertitude),

- les utilisateurs sont des professionnels avec une forte connaissance du domaine et des exigences en terme de précision du système (langage complexe).

Définir un modèle de recherche d'information nécessite la mise en place d'un certain nombre d'éléments, concernant l'indexation, la formulation du besoin et la fonction de correspondance, dont le choix est orienté par le contexte de recherche. Au cours de l'état de l'art que nous avons réalisé, nous avons étudié la modélisation des systèmes de recherches d'information, pour ensuite proposer une modélisation qui prenne compte le contexte dans lequel nous nous situons. C'est ainsi que nous avons proposé un modèle qui a les caractéristiques suivantes :

- il est fondé sur un langage complexe, faisant intervenir des entités (pouvant être des termes simples, des concepts, des objets graphiques, etc.) inter reliées, et permettant l'utilisation multiple d'une même entité dans la description d'un même document ou d'une même requête,
- la formulation du besoin fait intervenir deux notions : la première concerne l'importance que les éléments descriptifs de la requête (entité et relation) soient contenus dans les documents et la seconde concerne la certitude de l'utilisateur que ces éléments sont bien ceux qui devraient apparaître dans les documents. Nous avons proposé, pour intégrer ces deux notions dans la formulation du besoin, d'enrichir chaque entité et chaque relation de la requête par les deux critères d'obligation/option et de certitude/incertitude,
- la fonction de correspondance que nous avons mis en place, prend en compte les contraintes liées à la représentation des documents et des requêtes proposées (utilisation multiple d'entités, relations, critères), en s'appuyant sur une injection, sous contraintes, entre certains éléments de la requête et des documents,
- enfin, la formulation de la requête, fondé sur l'utilisation de critères associés, nous a permis de proposer deux types de classification des documents renvoyés par le système, en fonction des éléments de la requête que ces documents ont en commun. En combinant cette classification avec les jugements de pertinence des utilisateurs, concernant les documents renvoyés par le système, nous avons proposé une piste pour mettre en place un processus de reformulation de la requête.

Afin de mettre en application ce modèle de recherche d'information, nous avons choisi l'accès à l'information dans la documentation technique à usage professionnels, et plus particulièrement à l'accès par le média graphique à ces documents. L'utilisateur, qui consulte fréquemment la documentation technique a donc une mémoire des documents, et peut dès lors formuler sa requête en décrivant le « graphique idéal » qu'il souhaiterait retrouver.

En nous basant sur le modèle général proposé et sur une étude des graphiques tels qu'ils apparaissent dans la documentation technique, nous avons proposé :

- un modèle de recherche de graphiques techniques : en exploitant la complémentarité informationnelle entre le graphique et le texte, dans les documents techniques, nous avons proposé une modélisation structurée et multi-vues de leur contenu qui prend source dans leurs *propriétés visuelles* ainsi que dans le *contexte textuel* dans lequel ils apparaissent
- trois modes de formulation de la requête : nous donnons à l'utilisateur le choix entre un mode purement graphique, un mode purement textuel et un mode combinant

texte et graphique, pour formuler son besoin, selon ce dont il se souvient des graphiques qu'il désire retrouver. Ces trois langages permettent tous d'exprimer les notions relatives à l'importance des éléments de la requête (critère d'obligation/option) et aux doutes de l'utilisateur (critère de certitude/incertitude).

Cette modélisation a conduit à la mise en place d'un système opérationnel de recherche de graphiques techniques :

- nous avons d'abord proposé un processus d'indexation de ce média fondé sur (i) une indexation automatique qui permet de construire, à partir du texte accompagnant le graphique dans les documents techniques, l'index textuel et structuré du graphique et (ii) un processus manuel qui permet, à partir d'hypothèses concernant la mémorisation de ce type de graphiques, de construire son index visuel,
- nous avons ensuite mis en place un système de recherche, qui se base sur l'index textuel des graphiques, extraits automatiquement, en considérant uniquement les termes extraits (sans structure), et ce afin de valider le bien fondé de notre approche. Des expérimentations nous ont permis de confirmer l'apport des critères que nous avons ajoutés dans l'expression du besoin, et de vérifier que les modèles existants (booléen, vectoriel, probabiliste et de langue), même en leur apportant certaines extensions pour gérer les critères proposés, ne permettent pas de rendre compte convenablement de ces critères, ce qui confirme l'utilité de la fonction de correspondance proposée.

Enfin, la qualité de l'indexation étant une caractéristique importante dans les systèmes de recherche d'information, notamment lorsque le langage est complexe et le processus est automatique, nous avons proposé une adaptation des mesures de qualités d'indexation conçues pour des langages simples, afin qu'elles permettent l'évaluation qualitative d'indexation fondées sur des langages complexes. Ces mesures nous ont permis d'évaluer la qualité de notre indexation textuelle. L'expérimentation réalisée auprès de 11 participants a établie la qualité globale de notre indexation, et a permis de cibler les parties de l'index qui posent certains problèmes au cours de l'extraction.

2. Perspectives

Comme tout travail de thèse, ce travail donne lieu à de nombreuses perspectives.

A court terme, certains points sont à compléter :

- la validation de l'indexation visuelle du graphique : nous avons posé, ici, le protocole expérimental permettant une telle validation et souhaitons mettre en place l'expérience correspondante,
- la mise en place d'une seconde validation de notre système : nous avons validé le bien fondé de notre approche en nous limitant à l'utilisation des termes dans l'indexation des documents. Cette première évaluation étant prometteuse, nous souhaitons donc mettre en place une seconde évaluation faisant intervenir, dans la représentation des documents et des requêtes, les relations entre entités. La comparaison pourrait s'effectuer avec un modèle fondé sur le formalisme des graphes conceptuels.

- la mise en place du processus de reformulation : nous avons proposé, ici, différentes situations envisageables, fondées sur la distribution des documents jugés pertinents dans les classes de pertinences, ainsi que les actions, permettant la modification de la requête initiale, se rapportant à chacune de ces situations. Nous souhaitons mettre en place un processus de reformulation de la requête, dans notre cadre applicatif, en tenant compte de ce qui peut, dans les particularités visuelles des graphiques, influencer sur les jugements de pertinence de l'utilisateur (comme la symétrie des objets, par exemple),

A long terme, certains points semblent être intéressants à envisager.

- Nous avons proposé une indexation manuelle de la partie visuelle du graphique, néanmoins, une réflexion sur ce qui peut être extrait automatiquement de ce média particulier (en utilisant une approche syntaxique) devra être menée. Les objets illustratifs, par exemple, nous semblent être un bon point de départ pour cette réflexion.
- Nous avons proposé, dans ce travail, un système permettant de retrouver, parmi les graphiques contenus dans les documents techniques, ceux susceptibles d'intéresser l'utilisateur. Une amélioration de ce système, serait de l'intégrer dans un système hypermédia, reliant des unités documentaires des documents techniques (textuelles et des graphiques) de manière à ce que l'utilisateur puisse naviguer dans un document reconstruit tournant autour de son besoin. Ceci pourrait être envisagé, en partant d'un travail, effectué dans l'équipe, qui combine un système hypermédia à système de recherche [Kheirbek,95].
- Les utilisateurs professionnels utilisant la documentation technique, se déplacent dans leur travail. Accéder à l'information, qui les intéresse dans la documentation technique, lorsqu'ils sont chez un client, devrait être possible, à partir d'un dispositif mobile. Une adaptation de la recherche (interface d'interrogation, contenu des documents retournés) au support utilisée devrait être envisagée. Ainsi, selon le support utilisé et la qualité du réseau, par exemple, l'utilisateur se verra retourner uniquement du texte, ou des graphiques commentés, etc.



Annexes

Annexe A : Modèles classiques de recherche d'information

Différents modèles de RI textuelle, dits « classiques » [Van Rijsbergen,79], sont fondés sur ce type de langage. Parmi ces modèles figurent le modèle booléen, le modèle vectoriel et le modèle probabiliste. Ces modèles ont en commun le vocabulaire (des termes ou syntagmes) et seuls le modèle de document, le modèle de requête et l'évaluation de la correspondance changent.

1. Modèle booléen

Dans ce modèle, le document est représenté comme une conjonction logique de termes et la requête comme une expression logique de termes (utilisant les opérateurs *ET* (\wedge), *OU* (\vee) et *NON* (\neg)). Lors de la correspondance, l'opérateur *ET* est interprété comme la présence de deux termes dans les documents, le *OU* comme la présence de l'un ou de l'autre et le *NON* comme l'absence du terme dans le document. Ainsi, un document est soit pertinent pour une requête soit il ne l'est pas. En conséquence, le système détermine un ensemble de documents non ordonnés comme réponse à une requête.

Le modèle booléen a l'avantage d'être très simple à mettre en œuvre, et de posséder une fonction de correspondance très facile à mettre en œuvre. Ce qui lui a valu sa popularité et son utilisation dans de nombreux systèmes. Cependant, dans sa version pure, ce modèle souffre de certaines lacunes dont l'absence d'ordonnement des documents retrouvés. En effet, cette absence d'ordonnement des réponses ne satisfait pas les utilisateurs qui doivent encore fouiller dans cet ensemble de documents pour trouver des documents qui les intéressent. Une extension du modèle booléen standard basée sur les ensembles flous tente de remédier à cette lacune en tenant compte de la pondération des termes dans les documents. Du côté requête, elle reste toujours une expression booléenne classique. Un des avantages de ce modèle est qu'on peut mesurer le degré de correspondance entre un document et une requête dans l'intervalle $[0, 1]$. Ainsi, on peut ordonner les documents dans l'ordre décroissant de leur correspondance avec la requête. Au niveau de la représentation, on a également une représentation plus raffinée des documents : On peut exprimer dans quelle mesure un terme est important dans un document.

Une autre extension du modèle booléen permet d'associer une importance à chaque terme de la requête. Cependant, ce type de modèle n'a jamais gagné du terrain dans la pratique. La raison principale est qu'il est difficile de comprendre le sens de la pondération associée à un terme.

Dans tous les cas (modèle booléen simple ou étendu), même si le langage d'interrogation offre une grande flexibilité aux usagers pour exprimer leurs besoins, il reste difficile à utiliser. En effet, les mots "et" et "ou" du langage naturel ne correspondent pas parfaitement aux opérateurs logiques *ET* et *OU*. Par exemple, quelqu'un qui cherche des documents en "logique des propositions et logique de prédicats" peut en réalité vouloir chercher des documents sur "la

logique de prépositions" ou sur "la logique de prédicat". Ici, le mot "et" doit plutôt être traduits en *OU*. Ceci implique que les expressions logiques données par l'utilisateur correspondent souvent mal à son besoin. La qualité de la recherche souffre en conséquence.

2. Modèle vectoriel [Salton71]

Dans le modèle vectoriel, le document et la requête sont représentés comme un vecteur de poids d'un espace à n dimensions (les dimensions étant constituées par les termes du vocabulaire d'indexation). Chaque poids désigne l'importance d'un terme dans le document ou la requête. Le degré de correspondance entre les deux vecteurs est déterminé par leur similarité qui peut être calculée en fonction du cosinus de l'angle formé par les deux vecteurs. Plus deux vecteurs sont similaires, plus l'angle formé est petit et son cosinus est grand.

La Figure 73 montre un exemple d'espace vectoriel composé de 3 termes t_1 , t_2 et t_3 . Les index de deux documents d_1 et d_2 et une requête q sont représentés dans cet espace.

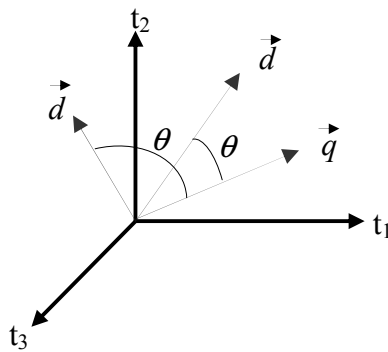


Figure 73 Représentation vectorielle de deux documents d_1 et d_2 et une requête q dans l'espace (t_1, t_2, t_3)

Un des avantages du modèle vectoriel est qu'il permet l'ordonnement des réponses retournées par le système. Le fait qu'il soit en plus robuste, en font sans doute le modèle le plus utilisé en RI.

3. Modèle probabiliste [Robertson,76]

Le modèle probabiliste essaye d'estimer la probabilité qu'un utilisateur a de trouver un document d pertinent. L'idée est de retrouver les documents qui ont en même temps une forte probabilité d'être pertinents et une faible probabilité d'être non pertinents.

Un document d et une requête q sont représentés par un vecteur comme dans le modèle vectoriel, mais les poids sont binaires.

La fonction de correspondance évalue la pertinence du document d pour la requête q ainsi :

$$\text{Similarité}(d, q) = P(R/d) / P(\neg R/d)$$

où $P(R/d)$ est la probabilité que le document d soit pertinent pour la requête q et $P(\neg R/d)$ est la probabilité que le document d ne soit pas pertinent pour la requête q .

Typiquement, ces probabilités sont calculées selon des méthodes paramétriques : on suppose que la distribution des mots suit une certaine norme parmi les documents pertinents (et non pertinents). Il peut s'agir de la distribution de poisson, par exemple. En fonction de la distribution des mots dans chacun des deux ensembles de documents échantillons, documents pertinents et documents non pertinents, il est possible d'estimer les probabilités des mots pour la pertinence et donc de calculer $P(R/d)$.

4. Modèle de langue

L'idée de base des modèles de langue (LM) est de déterminer la probabilité que la requête q puisse être générée à partir du document d . Cette formulation est similaire à l'idée derrière les modèles probabilistes, cependant, la façon de calculer cette probabilité diffère. En effet, dans le LM, on ne tente pas de modéliser la notion de pertinence dans le modèle mais on considère que la pertinence d'un document face à une requête est en rapport avec la probabilité que la requête puisse être générée par le modèle de langue du document (Notons qu'un modèle de langue a pour objectif de capter la régularité linguistique par une ou plusieurs fonctions probabilistes).

Ainsi, pour chaque document d , on tente de construire le modèle de langue M_d qui lui correspond. Le score de ce document face à une requête q est alors déterminé par la probabilité que son modèle génère la requête :

$$\text{Similarité}(d, q) = P(q / M_d)$$

Il est aussi possible de construire un modèle de langue pour la requête M_q . Le score d'un document d face à une requête q est alors déterminé par une comparaison entre les deux modèles.

Annexe B : De EMIR² à GRIM

Nous avons choisi de nous inspirer du modèle EMIR² pour aboutir à un modèle qui prend en compte les aspects particuliers des graphiques issus des documents techniques. EMIR² est un modèle permettant la représentation du contenu des images. Ce choix se justifie par le fait qu'il « considère comme représentation du contenu d'une image diverses interprétations de l'ensemble des objets images et des relations qui les lient » [Mechkour,95]. Ce point de vue concorde avec notre besoin de prendre en compte les aspects multi-facettes et structurel des graphiques que nous désirons représenter.

Dans cette partie, nous commençons par rappeler synthétiquement le modèle EMIR² avant de décrire les aspects de ce modèle que nous devons repenser pour aboutir à un modèle capable de représenter complètement les graphiques de notre corpus.

1. Structure de EMIR²

Pour rappeler brièvement le modèle EMIR² détaillé en Annexe A, nous donnons une représentation des différents composants du modèle d'images ainsi que les liens de dépendances entre eux. Cette représentation est illustrée par la Figure 74. Ainsi, la vue structurelle est l'élément pivot de cette modélisation car elle assure le lien et la cohérence entre les différentes vues pour former une vision globale d'une image.

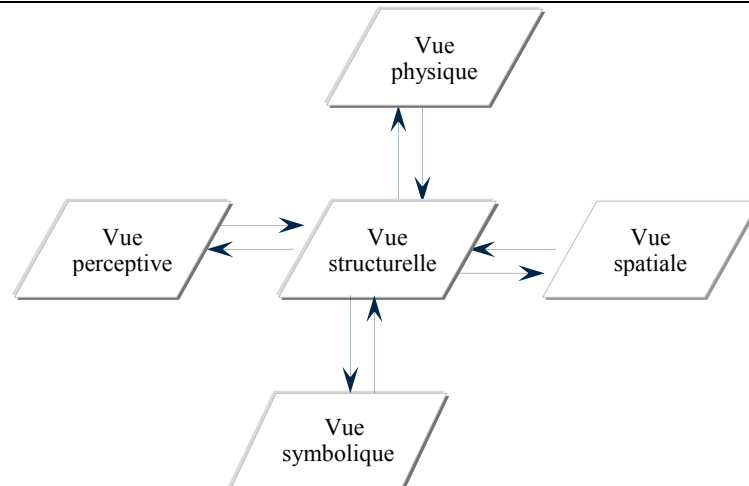


Figure 74 Schématisation du modèle EMIR²

2. Vers notre modèle de représentation

EMIR² est un modèle étendu de représentation des images. Il permet de représenter un grand nombre de type d'images. Ce modèle fait abstraction du domaine des images, de l'application particulière qui les manipule et du type des utilisateurs auxquels elle est destinée.

Le modèle que nous cherchons à obtenir doit représenter les graphiques des documents techniques. Nous sommes donc, en premier lieu, face à un autre type de média que sont les graphiques et, en second lieu, face à des utilisateurs spécialisés. Ces deux hypothèses nous amènent à des reconsidérations de EMIR² :

Tout d'abord, manipuler des graphiques et non plus des images induit que :

- la vue perceptive est inutile,
- des vues structurelle et spatiale adaptées au graphiques doivent être définies.

Par ailleurs, dédier le système à des utilisateurs spécialisés entraîne que :

- la prise en compte d'une sémantique particulière est nécessaire. Nous le ferons dans la vue dite opératoire,
- la prise en compte de la mémoire visuelle de l'utilisateur est également fondamentale pour notre modèle.

Ces différents points sont détaillés et justifiés ci-après.

2.1. Pour un corpus particulier de graphiques

2.1.1. Vue perceptive inutile

Le fait que nous nous intéressions au type particulier de média, que sont les graphiques, fait que la vue perceptive n'a pas de raison d'être. Ceci est dû à la représentation de ce média qui se fait par un ensemble de primitives géométriques dessinées en noir. Les aspects visuels tels que la couleur, la texture ou la brillance n'ont alors plus de sens lorsque nous parlons de graphiques. Il en résulte l'inutilité d'une vue perceptive dans notre modèle de représentation (voir Figure 75).

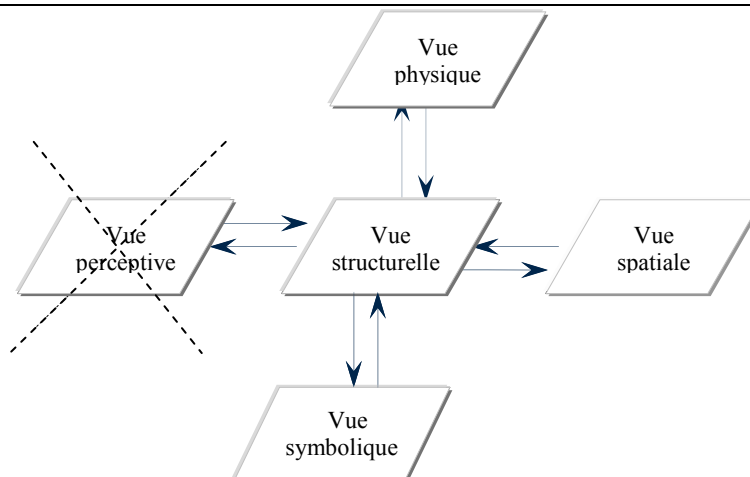


Figure 75 Inutilité d'une vue perceptive

Comme nous l'avons vu précédemment, les graphiques de notre corpus ont des caractéristiques au niveau de la structure et d'un point de vue spatial. Ces caractéristiques influent sur les vues structurelle et spatiale qui doivent donc être reconsidérées.

2.1.2. Une nouvelle vue structurelle


Au niveau de la structure, les graphiques des documents techniques contiennent des objets qui leur sont propres. Ces objets ne sont pas une projection d'un objet réel sur le graphique, mais des objets ajoutés par l'auteur pour clarifier ou mettre en évidence un aspect des objets réels représentés dans le graphique. Il s'agit des « objets illustratifs ».

Nous distinguons :

- la représentation d'éléments zoomés :

Dans ce cas, un même élément réel est représenté deux fois. La sémantique des deux objets graphiques est la même puisqu'il s'agit du même élément, mais d'un point de vue structure, nous avons deux objets, liés certes, mais distincts. Cette double représentation d'un même élément (c'est à dire ayant une même sémantique) influe sur le modèle de vue structurelle.

- La représentation d'éléments numérotés :

Dans ce cas, la décomposition d'un objet contenu dans le graphique est perçue par une énumération pointant les éléments composants : . Les objets représentant une telle énumération sont des composants à représenter dans la vue structurelle.

- La représentation d'éléments qui subissent une action :

Dans ce cas, une flèche ou une main est représentée dans le graphique pour indiquer que telle action (tirer, pousser, etc.) est à appliquer sur un objet réel donné, représenté dans le graphique. Même s'il ne représente pas un objet réel, cet élément véhicule un sens (l'action). Il est donc pertinent dans le graphique, et doit, dans ce cas, être pris en compte dans la vue structurelle

- La représentation à plat d'éléments tridimensionnels :

Comme pour le zoom, dans ce cas, un même élément est représenté deux fois, mais différemment. La représentation à plat peut être par exemple, la projection de la vue de dessus d'un objet tridimensionnel. Cette représentation double d'un même objet est à retenir dans la vue structurelle.

2.1.3. Une nouvelle vue spatiale

Au niveau spatial, le fait que les graphiques de notre corpus soient tridimensionnels, fait que trois de leurs faces sont visibles. Cette caractéristique a un impact sur la représentation spatiale du graphique, qui est, dans ce cas, plus précise. La vue spatiale comporte alors plus d'objets spatiaux par rapport à ceux définis à partir de la vue structurelle (l'objet tridimensionnel + ces faces visibles), et d'autres types de liens s'ajoutent au modèle de cette vue pour relier l'objet tridimensionnel à ses faces visibles.

2.2. Pour un utilisateur spécialisé

2.2.1. Une sémantique particulière

L'utilisateur des documents techniques effectue une activité opératoire. Il désire donc être informé non seulement des propriétés d'un dispositif technique mais aussi de la réalité de son fonctionnement. Les graphiques présents dans la documentation technique offrent cette information opératoire à l'utilisateur.

Ainsi, une interprétation du graphique pourrait être l'ensemble des actions que l'utilisateur doit effectuer sur les différents objets contenus dans ce graphique. Autrement dit, une des interprétations possibles du graphique tourne autour de l'aspect opératoire. Dans la représentation sémantique du graphique, nous distinguons alors un second aspect autre que l'aspect descriptif.

Pour décrire un graphique, il nous faut représenter son contenu descriptif dans la vue symbolique et son contenu opératoire dans une vue qui y est rattachée que nous appelons « la vue opératoire ». Plus de détails sur cette vue seront donnés dans la section suivante.

Dans la figure suivante, nous pouvons voir où se situe l'ajout de la vue opératoire.

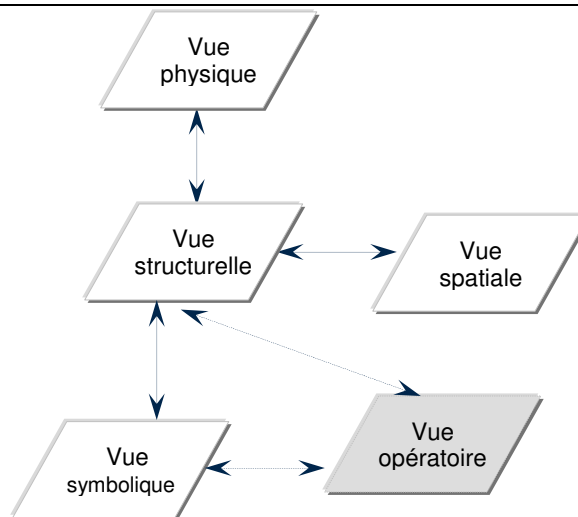


Figure 76 Prise en compte d'une vue opératoire

2.2.2. Mémoire visuelle

Le fait que les utilisateurs forment un groupe homogène de professionnels, influe sur leur perception du graphique. Ainsi, certains éléments du graphique ayant une plus grande importance que d'autres auront plus d'impact sur leurs mémoires visuelles. Il en est de même pour les relations spatiales entre ces objets.

Prenons l'exemple de la Figure 77 : Dans la représentation géométrique (b) correspondant au graphique (a) la majorité des éléments composant l'imprimante sont représentés. Pourtant, tous ces éléments n'auront pas le même impact sur la mémoire visuelle de l'utilisateur. Certains éléments uniquement seront encodés. (c) est un exemple représentant ces éléments.

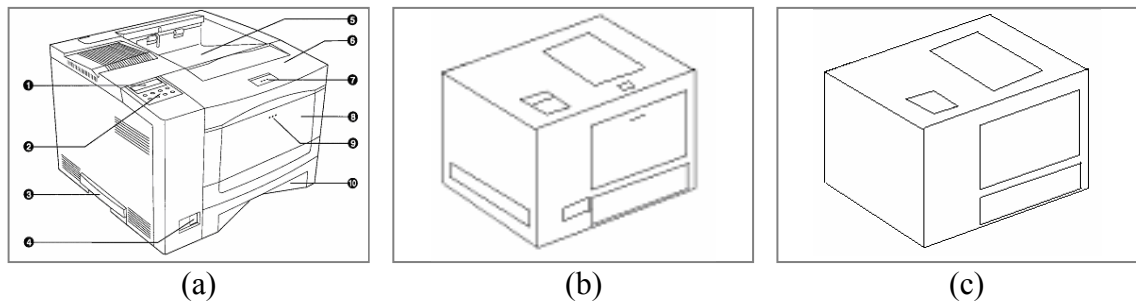


Figure 77 Impact de la mémoire visuelle sur la représentation spatiale d'un graphique

Il est donc pertinent de mettre en évidence les objets les plus significatifs pour les utilisateurs (d'un point de vue spatial).

Nous avons donc fait intervenir dans notre modèle, au niveau de la vue spatiale, une vue intermédiaire permettant de représenter l'encodage du graphique dans le mémoire visuelle de l'utilisateur. Cette vue que nous appelons « la vue mémoire visuelle » remplace la vue spatiale. (voir Figure 78)

Cette vue intermédiaire est en fait une induction de la vue spatiale.

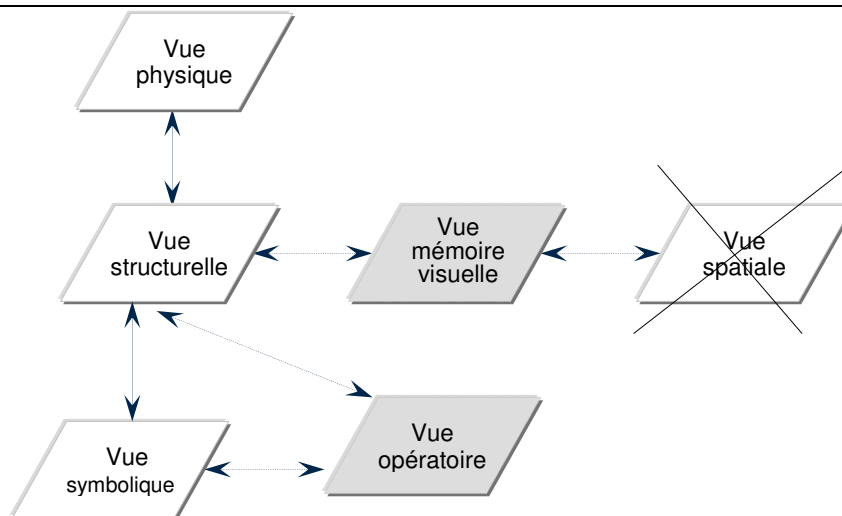


Figure 78 Prise en compte de la vue mémoire visuelle

3. Bilan

Pour proposer notre modèle de représentation des graphiques des documents techniques, nous nous sommes inspirés de la philosophie d'un modèle existant, EMIR², qui considère l'image comme un objet complexe, construit à partir d'éléments correspondant à des interprétations différentes de son contenu.

En tenant compte du média particulier que doit représenter notre modèle, et du groupe d'utilisateurs auxquels il est destiné, nous avons abouti à un modèle dont les différentes vues et leurs interrelations sont schématisés dans la Figure 79.

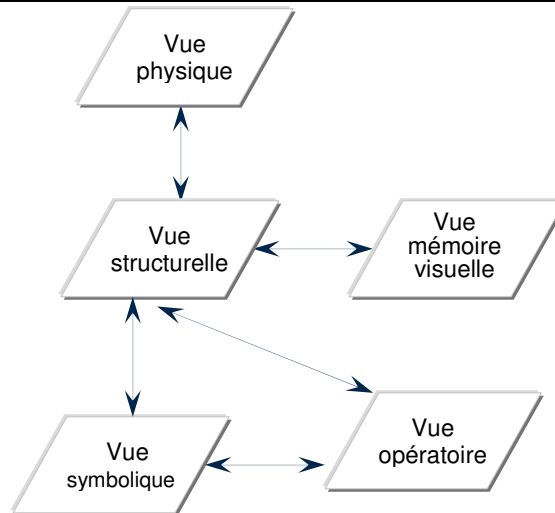


Figure 79 Une synthèse de notre modèle

Bibliographie

- [Ahuja,93] R. Ahuja, T. Magnanti et J. Orlin : *Network flows*. Prentice Hall, 1993.
- [Akdag,94] H. Akdag et F. Khoukhi : Une approche logico-symbolique pour le traitement des connaissances nuancées", *proceedings of the fifth IPMU*, Paris, 1994.
- [Allwood,97] C.M. Allwood et T. Kalen : Evaluating and improving the usability of user manual, *Behavior and Information Technology*, Vol.16(1), pp 43-57, 1997
- [Atkinson,68] R.C. Atkinson et R.M. Shiffrin : Human memory: A proposed system and its control processes, K.W. Spence & J.T. Spence (Eds.), *The psychology of learning and motivation*. New York: Academic Press, Vol. 2, pp 89-195, 1968.
- [Baddeley,95] A. Baddeley : *La mémoire humaine : théorie et pratique*, Presses Universitaires de Grenoble, 1995.
- [Badjo,00] B. Badjo-Monne et M. Bertier : Vers un modèle d'indexation des documents techniques, *Document numérique*, Vol. 4(1), Hermes, 2000.
- [Baeza-Yates,99] R. Baeza-Yates et B. Ribeiro-Neto : *Modern Information Retrieval*, ACM Press Series/Adison-Wesley, 1999
- [Berrut,88] C. Berrut, *Une méthode d'indexation fondée sur l'analyse sémantique de documents spécialisés. Le prototype RIME et son application à un corpus médical*, thèse de doctorat, Université Joseph Fourier, 1988.
- [Berrut,89] C. Berrut et Y. Chiaramella : Indexing medical reports in a multimedia environment: the RIME experimental approach, in *ACM-SIGIR Conference on Research and Development in Information Retrieval*, Boston, Massachusetts, USA, pp 187-197, 1989.
- [Berrut,90] C. Berrut : Indexing Medical Reports : The RIME Approach, *Information Processing and Management*, vol. 1(26), pp 93-109, 1990.
- [Blair,90] D.C. Blair: *Language and Representation in Information Retrieval*, Amsterdam: Elsevier, 1990.
- [Bordogna,91] G. Bordogna, C. Carrara et G. Pasi : Query Term Weights as Constraints in Fuzzy Information Retrieval, *Information Processing & Management*, vol. 27, pp 15-26, 1991.
- [Bosc,94] P. Bosc et H. Prade : An Introduction to the Fuzzy Set and Possibility Theory-Based Treatment of Soft Queries and Uncertain Or Imprecise Databases, *Uncertainty Management in Information Systems*, pp 285-324, 1994
- [Bourigault,96] D. Bourigault, I. Gonzalez et C. Gros : LEXTER, a Natural Language Tool for Terminology Extraction, *Proceedings of the seventh EURALEX International Congress*, Goteborg, Suède, 1996.

-
- [Bourigault,05] D. Bourigault , C.Fabre, C.Fr erot, M.-P. Jacques et S. Ozdowska : Syntex, analyseur syntaxique de corpus, *Actes des 12 emes journ ees sur le Traitement Automatique des Langues Naturelles*, France,2005.
- [Buell,81] D.A. Buell et D.H. Kraft : A model for a weighted retrieval system, *Journal of the American Society for Information Science*, Vol.32(3), pp 211-216, 1981.
- [Cater,87] S.C. Cater et D.H. Kraft : TIRS: A Topological Information Retrieval System Satisfying the Requirements of the Waller-Kraft Wish List, *SIGIR conference on Research and development in information retrieval*, pp 171-180,1987.
- [CEP,83] *Guide pratique   l'usage des fabricants et fournisseurs pour la r edaction des notices destin ees aux acheteurs et utilisateurs*, Saint etienne, Le H enaff.
- [Clavier,97] V. Clavier, C. Froissart et C. Paganelli : Objects and Actions : Two concepts of major interest in information retrieval in full-text databases, *NLDB '97, Workshop on Applications of Natural Language to Information Systems*, Vancouver, 1997.
- [Cohen60] J. Cohen : A coefficient of agreement for nominal scales, *Educational and Psychological Measurement*, Vol. 20, pp 27-46, 1960.
- [Cooper,69] W.S. Cooper : Is interindexer consistency a hobgoblin?, *American documentation*, Vol. 20, pp 268-278, 1969.
- [Craik,72] F.I.M. Craik et R.S. Lockhart : Levels of processing: A framework for memory research, *Journal of Verbal Learning and Verbal Behavior*, Vol. 11, pp 671-684, 1972.
- [Croft,86] W.B. Croft : User-Specified Domain Knowledge for Document Retrieval, *Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval*, pp 201-206, Italie, 1986.
- [Cunningham,02] H. Cunningham, D. Maynard, K. Bontcheva et V. Tablan : GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications, *ACL'02*, 2002.
- [Daille,94] B. Daille : *Approche mixte pour l'extraction automatique de terminologie : statistique lexicale et filtres linguistiques*, Th ese de doctorat en informatique, Universit  Paris7, 1994.
- [David,90] S. David et P. Plante : De la n ecessit  d'une approche morphosyntaxique dans l'analyse de textes, *Intelligence artificielle et sciences cognitives au Qu ebec*, Vol.3 (3), pp 140-154, 1990.
- [Denos,97a] N. Denos : *Mod elisation de la pertinence en recherche d'information - mod le conceptuel, formalisation et application*, th ese de doctorat, Universit  Joseph Fourier, Grenoble I, 1997.
- [Denos,97b] N. Denos : *Modelling system relevance through user criteria - A conceptual and a formal model*, Rapport de recherche,  quipe Mrim, CLIPS-IMAG, TR-97-001, 1997.
- [Fourel,97] F. Fourel : *Mod elisation, indexation et recherche de documents structur s*, Th ese Informatique de l'universit  Joseph Fourier, Grenoble I, 1998
- [Ganier,02] F. Ganier : L'analyse des fonctionnements cognitifs : un support   l'am elioration de la conception des documents proc eduraux, *Psychologie Fran aise*, Vol. 47(1), pp 53-64, 2002.
- [Ganier,03] F. Ganier et L. Heurley : La prise en compte de l'utilisateur et de son utilisation des documents proc eduraux : une pr econdition n ecessaire   la conception de documents adapt s, *Production, compr ehension et usages des  crits techniques au travail*, pp 19-20, 2003.

-
- [Gardin,70] N. Bely, A. Borillo, N. Siot-Decauville, J. Virbel : *Procédures d'analyse sémantique appliquée à la documentation scientifique*, préface J.C. Gardin, Gauthiers-Villars, 1970.
- [Guarino,99] N.Guarino, C. Massolo, G. Vetere : *OntoSeek : Content-Based Access to the Web*, *IEEE Intelligent Systems*, Vol. 14(3), pp 70-80, 1999.
- [Herrera,01] E. Herrera-Viedma : *Modeling the Retrieval Process of an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach*, *Journal of the American Society of Information Science*, Vol. 52(6), pp 460-475, 2001
- [ISO/CEI,95] *Guide ISO/CEI 37 : Instructions d'emploi pour les produits présentant un intérêt pour les consommateurs*.
- [Jacquemin,97] C. Jacquemin : *Variation terminologique : reconnaissance et acquisition automatique de termes et de leurs variantes en corpus*, Habilitation à diriger des recherches, Université de Nantes, 1997.
- [Joly,93] M. JOLY : *Introduction à l'analyse de l'image*. NATHAN Université, 1993.
- [Kefi,02] L. Kefi, *Modèle d'indexation des graphiques des documents techniques*, rapport de DEA, Groupe MRIM - CLIPS-IMAG, juin, 2002.
- [Kefi,03] L. Kefi, C. Berrut, E. Gaussier : *Modèle d'indexation de données peu symboliques dans des documents structurés : L'exemple du graphique dans un corpus de documents techniques*, *Inforsid*, pp 69-86, Nancy, 2003.
- [Kefi,05] L. Kefi, C. Berrut, E. Gaussier : *Indexation Complexe de documents: vers une vérification qualitative*. *Inforsid*, pp 521-538, Grenoble, 2005.
- [Kefi,06] L. Kefi, C. Berrut, E. Gaussier : *Un modèle de RI basé sur des critères d'obligation et de certitude*, *CONFérence en Recherche d'Information et Applications CORIA'06*, Lyon, 2006.
- [Kendall,71] M.G. Kendall : *Rank correlation methods*, Hafner Pub.Co, New-York, 1962.
- [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, 1995.
- [Khoukhi,96] F. Khoukhi : *Une approche logico-symbolique dans le traitement des connaissances incertaines dans les systèmes à base de connaissances*, 3èmes *rencontres nationales des jeunes chercheurs en Intelligence Artificielle*, Nantes, 1996.
- [Kraft,83] D.H. Kraft, D.A. Buell : *Fuzzy sets and generalized Boolean retrieval systems*, *International Journal of Man-Machine Studies*, Vol. 19, pp 45-56, 1983.
- [Kraft, 94] D.H. Kraft, G. Bordogna et G. Pasi : *An Extended Fuzzy Linguistic Approach to Generalize Boolean Information Retrieval*, *Journal of Information Science-Applications*, Vol. 2(3), pp 119-134,1994.
- [Lalmas,96] M. Lalmas: *Theories of Information and Uncertainty for the modelling of Information Retrieval: an application of Situation Theory and Dempster-Shafer's Theory of Evidence*. Thèse de doctorat, University of Scotland, Glasgow, Scotland, 1996.
- [Lalmas,04] M. Lalmas et P.Vannoorenberghe : *Indexation et recherche de documents XML par les fonctions de croyance*, *Première Conférence en Recherche d'Information et Applications, CORIA'2004*, pp143-160, 2004.

-
- [Lorenz 94] O. Lorenz et G. Monagan: Retrieval of line drawings. *Proceedings of The Third Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, USA, 1994.
- [Lorenz 95] O. Lorenz et G. Monagan : Automatic indexing for storage and retrieval of line drawings, *IS&T/SPIE's Symposium on Electronic Imaging Science & Technology*, San Jose - California, USA, 1995.
- [Malandain,00] N. Malandain : Automatic geographical hypertext "multi-scaled links" generation, *Proceedings of Fifth International Workshop on Principles of Digital Document Processing*, Germany, 2000.
- [Martinet,04] J. Martinet : *Un modèle vectoriel relationnel de recherche d'information adapté aux images*, Thèse de doctorat, Université Joseph Fourier, Grenoble, 2004.
- [Mechkour,95] M. Mechkour, *EMIR2 : 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.
- [Michel,97] D. Michel : *Langage et cognition spatiale*, Paris-Masson, 1997.
- [Mihalcea,00] R. Mihalcea et D.I. Moldovan : Using WordNet and Lexical Operators to Improve Internet Searches, *IEEE Internet Computing*, Vol.4(1), pp 34-43, 2000.
- [Montmollin,96] M. De Montmollin : *L'ergonomie* (3ème édition), Paris : La Découverte, 1996.
- [Ogilvie,01] P. Ogilvie et J. Callan : Experiments using the Lemur Toolkit. *Proceedings of the 10th Text Retrieval Conference, TREC*, 2001.
- [Ouerfelli,00] T. Ouerfelli, G. Lallich-Boidin : Pratiques d'indexation dans les Bases Textuelles Structurées : Application aux Textes Techniques sous Format HTML, *Proceedings of the 28th Annual Conference CAIS 2000: Dimensions of a global information science*, 2000.
- [Ouesleti,96] R.Ouesleti, P. Frath et F. Rousselot : A corpus-based method for acquisition and exploitation of terms, *CLIM'96-Student Conference in Computational Linguistics in Montreal*, Canada, 1996.
- [Ounis,97] I. Ounis et M. Pasca : An extended inverted file approach for information retrieval, *IDEAS'97-International Database Engineering and Application Symposium*, IEEE Computer Society Press, pp 397-402, Canada, 1997.
- [Paganelli,97] C. Paganelli : *La recherche d'information dans des bases de documents techniques en texte intégral : Etude de l'activité des utilisateurs*. Thèse de Doctorat en Sciences de l'Information et de la Communication, Grenoble 3, 1997.
- [Paganelli,99] C. Paganelli, E. Mounier : L'accès à l'information pertinente dans les documents techniques volumineux, *Secondes Journées du chapitre Français de l'ISKO*, Lyon, 1999.
- [Paganelli,02a] C. Paganelli, E. Mounier : Vers un système de consultation des documents techniques volumineux par des utilisateurs experts : le système Sysrit, *Interaction homme-machine et recherche d'information*, Paris- Hermès, pp195-228, 2002.
- [Paganelli,02b] C. Paganelli : *Interaction homme-machine et recherche d'information*, Paris-Hermès, 2002.
- [Rabitti,90] F. Rabitti et P. Savino: Image analysis for semantic Image database, *IEI-CNR Tech. Rep. B8-34*, Italy, 1990.
- [Rabitti,91] F. Rabitti et P. Savino: Image query processing Based on multi-level Signatures, *Proceedings of 14th annual international ACM/SIGIR conference on research and development in IR*, September, 1991.

-
- [Rocacher,03] D. Rocacher : On fuzzy bags and their application to flexible querying, *Fuzzy Sets and Systems*, 140, pp 93–110, 2003.
- [Robertson,76] S.E. Robertson et K. Sparck Jones : Relevance weighting of search terms. *Journal of the American Society for Information Science*, Vol. 27, pp 129-46, 1976.
- [Robertson,94] S.E. Robertson et S. Walker : Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. *Proceedings of SIGIR '94*, pp 232-241, New York, USA, 1994.
- [Rocchio,71] J.J. Rocchio : *Relevance feedback in information retrieval*, Prentice Hall, 1971.
- [Rock,59] I. Rock et P.Engelstein : A study of memory for visual form, *American Journal of Psychology*, Vol. 72, pp 221–229, 1959.
- [Rohini,94] K. Rohini Srihari et D. T. Burhans : Visual Semantics: Extracting Visual Information from Text Accompanying Pictures, *Proceedings of AAAI '94*, Seattle, USA, 1994.
- [Rohini,95] K. Rohini Srihari : Automatic Indexing and Content-Based Retrieval of Captioned Images, *IEEE Computer Magazine*, vol.28(9), pp.49-56, 1995.
- [Rowe,96] N. C. Rowe : *Precise and efficient access to captioned picture libraries: The MARIE project*, Technical report, Department of Computer Science, Naval Postgraduate School, 1996.
- [Salton,71] G. Salton : *The SMART Retrieval System: Experiments in Automatic Document Processing*, Prentice-Hall, 1971.
- [Salton,83] G. Salton, E. A. Fox, et H. Wu : Extended Boolean information retrieval, *Communications of the ACM*, Vol. 26(12), pp 1022-1036, 1983.
- [Schank,80] R. Schank : Language and memory, *Cognitive Science*, Vol. 4, pp 243-284, 1980.
- [Schank,81] R. Schank : *Representing Meaning : An Artificial Intelligence Perspective Cognitive Science*, Technical Report 1, Yale University, 1981.
- [Schmid,94] H. Schmid : Probabilistic part-of-speech tagging using decision trees, *Proceedings of International Conference on New Methods in Language Processing*, Manchester, 1994.
- [Schutze,95] H. Schutze et J. Pedersen : Information Retrieval based on Word Senses, *Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval*, pp 161-175, Las Vegas, USA, 1995.
- [Soergel,94] D. Soergel : Indexing and Retrieval performance : the logical evidence, *Journal of the American Society for Information Science*, Vol. 45(8), pp 589-599, 1994.
- [Sowa,84] J.F. Sowa : *Conceptual Structures*, Addison-Wesley, Reading,MA ,1984.
- [Standing,70] L. Standing, J. Conezio et R. Haber : Perception and Memory for Pictures: Single-trial learning of 2500 visual stimuli, *Psychonomic Science*, Vol. 19, pp 73-74, 1970.
- [Tabbone,01] S. Tabbone, L. Wendling et K. Tombre : Indexing of Technical Line Drawings Based on F-Signatures, *Proceedings of 6th International Conference on Document Analysis and Recognition*, pp 1220-1224, Washington, USA, 2001.
- [Tabbone,03] S. Tabbone, L. Wendling et K. Tombre : Matching of graphical symbols in line-drawing images using angular signature information, *International Journal on Document Analysis and Recognition*, Vol. 6(2), pp 115-125, October 2003.
- [Tricot, 04] A. Tricot : *La prise de conscience du besoin d'information : une compétence documentaire fantôme*, Publié en ligne sur Docs pour Docs, 2004.

- [Van Rijsbergen,79] C. J. Van Rijsbergen : *Information Retrieval*, 2e édition, Dept of Computer science, Université de Glasgow, 1979.
- [Vigner,76] G. Vigner : *Le français technique*, Paris-Hachette, 1976.
- [Waller,79] W.G. Waller et D.H. Kraft : A mathematical model for a weighted Boolean retrieval system, *Information Processing and Management*, Vol. 15(5), pp 235-245, 1979.
- [Wright,77] P. Wright : Presenting technical information : A survey of research findings, *Instructional Science*, Vol. 6, pp 93-134, 1977.
- [Wright,99] P. Wright, Printed Instructions: can research make difference? In H. Zwaga, T. Boersema et H. Hoonout (eds.), *Visual Information for everyday use*, pp 54-66, London, 1999.
- [Zweigenbaum,94] P. Zweigenbaum: MENELAS: an access system for medical records using natural language, *Comput Methods Programs Biomed.* Vol. 45(1-2), pp 117-20, 1994.

Résumé

En recherche d'information, les particularités relatives au contexte de recherche de l'utilisateur induisent certains besoins qu'il est nécessaire de prendre en compte dans la modélisation du système de recherche. Dans notre travail de thèse, nous nous situons dans un contexte où l'utilisateur a une mémoire des documents qu'il désire retrouver : son besoin est alors une description d'un document idéal, reflet du souvenir qu'il a de ces documents. Dans ce contexte de recherche particulier, nous proposons un modèle de recherche d'information fondé sur (i) un langage complexe (des entités inter reliées avec utilisation multiple d'une même entité dans la description du document et du besoin), (ii) des critères d'obligation/option et de certitude/incertitude, rattachés aux éléments de la requête, qui reflètent les doutes de l'utilisateur quant au contenu des documents susceptibles de l'intéresser et (iii) une fonction de correspondance prenant en compte les contraintes liées à la représentation des documents et des requêtes ainsi qu'une approche pour la reformulation du besoin fondée sur les jugements de pertinence de l'utilisateur et sur les caractéristiques communes des documents retenus (par rapport aux critères rattachés à la requête).

Ce modèle est par la suite appliqué dans le cadre concret d'une application : la recherche de graphiques dans les documents techniques par des utilisateurs professionnels. À travers cette application, nous validons notre approche (prise en compte des critères d'obligation/option et de certitude/incertitude) en comparant notre modèle aux modèles classiques existants.

Mots Clés : *Modèle de recherche d'information, langage complexe, option et incertitude*

Abstract

In the information retrieval (IR) task, characteristics related to the context of the user search induce some needs it is necessary to take into account in the modeling of the IR system. In this work we consider that the user has a memory about the documents he wants to find: his need consists of a description of the ideal document w.r.t his memory of the content of these documents.

In the aim to tackle this need, we propose an information retrieval model based on (i) a complex language (inter-connected entities with multiple use of the same entity to describe the document and the user query), (ii) additional criteria on query terms, focusing on obligation/optionality, and certainty/uncertainty, in order to express user doubts and its vague needs, and (iii) a matching function which takes into account constraints related to the document/query representation, as well as a query reformulation approach based on characteristics of documents that are considered relevant by the user.

This model is applied thereafter within a concrete application: graphics retrieval by professionals in technical documentation. Through this application, we compare our model with classical IR models in order to validate our approach (e.g. obligation/optionality, and certainty/uncertainty criteria).

Keywords: *Information retrieval model, complex language, optionality and uncertainty*
