République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider – Biskra
Faculté des sciences exactes et des sciences de la nature et de la vie
Département d'Informatique

N° d'ordre :
Série :

THÈSE

En vue de l'obtention du Doctorat en Sciences
Spécialité : Informatique

Titre

UN MODÈLE DE RAISONNEMENT POUR UN SYSTÈME DE RECHERCHE SÉMANTIQUE D'INFORMATIONS SUR LE WEB BASÉ AGENTS

Présentée par : NESSAH Djamel
Soutenue le :06/11/2014

Devant les membres de jury :

Président : Pr. SAHNOUNE Zaidi
Professeur, Université Constantine 2

Rapporteur : Pr. KAZAR Okba
Professeur, Université Mohamed KHIDER-Biskra

Examinateurs : Pr. BOUFAIDA Zizette
Professeur, Université Constantine 2

Pr. BALLA Amar
Professeur, ESI Alger

Dr. BENNOUI Hammadi
Maître de Conférences «A»
Université Mohamed KHIDER-Biskra

Dr. TERISSA LABIB Sadek
Maître de Conférences «A»
Université Mohamed KHIDER-Biskra

Remerciements

Je voudrais tout d'abord exprimer ma gratitude au professeur Okba Kazar, mon directeur de thèse pour avoir dirigé ce travail, m'avoir soutenu et appuyé tout au long de ces années. Son aide, sa patience, et son support inestimable m'ont permis d'accomplir cette thèse.

Mes remerciements vont également au Professeur Aicha-Nabila Benharkat, laboratoire LIRIS à l'INSA de Lyon, pour ses conseils, son aide riche et complémentaire qui m'a permis de construire ce travail.

Je remercie Professeur Zaidi Sahnoune, université de Constantine, d'avoir accepté de présider mon jury avec autant d'intérêt.

Mes respects et ma gratitude vont également aux membres du jury : Zizette Boufaida Université de Constantine, Ammar Balla ESI d'Alger, Hammadi Bennoui et Terissa Labib Sadek Université Mohamed khider -Biskra qui m'ont fait grand honneur de juger ce travail et qui par leurs observations constructives m'ont permis de l'enrichir.

Je tiens aussi à remercier toutes les personnes qui, de près ou de loin ont contribué à la réalisation de cette thèse.

Enfin, je voudrais remercier ceux sans qui cette thèse n'aurait pas été possible et en particulier ma femme Leila, mes enfants qui ont su comprendre mon occupation. Leurs soutient m'était très réconfortant pendant certains moments difficiles.

Djamel

A :

La mémoire de mon père …

Ma mère …

Mon épouse …

Mes adorables enfants …

Abstract

The key task for the web, namely, web searches, is evolving towards some novel form of semantic web search. In fact, actually most information retrieval systems are based on static vectors representations like the space vector model, and various statistical and / or probabilistic approaches. However, for a given user of information need, some retrieved documents might be not really responding to the search context.

The main intention of the semantic search is to improve the search quality by various techniques, such as the query's expansion and reformulation. Often two major difficulties when a researcher uses current information retrieval (IR) systems are, how to filter out irrelevant documents (reduce the noise) and how to discover latest or more significant documents (increase the recall). Recently a very promising approach to semantic web search is based on combining standard web pages and search queries with ontological background knowledge. In this perspective we'll describe a model with the aim to enhance the robustness of semantic web search, as it holds noise, and incompleteness. For this, first we recommend an original method for segmenting a text into semantic fragments to more explicit the enclosed text's semantics. Then we will merge a syntactic keyword search with purely semantic search based domain ontology and a multi agent system as support to tackle the solving of such distributed problems. Then to perform ranking algorithm on returned documents, we proposed a new semantic similarity measure between concepts, based on the taxonomy structure and verifying some suitable properties to more measure the distance between documents and user's queries.

Keywords: semantic information search; multi-agent system; semantic web; web languages; ontology; semantic annotation; metadata; inference engine; similarity measure; WordNet.

Résumé

La recherche d'information sur le web est entrain d'évoluer vers une nouvelle forme de recherche dite sémantique utilisant des concepts au lieu de mots clés. En effet, actuellement les systèmes de recherche d'informations sont basés sur des modèles de représentations utilisant des vecteurs statiques comme le modèle d'espace vectoriel, et diverses approches statistiques et probabilistes. Malgré cela, pour un besoin utilisateur en information, certains documents sélectionnés peuvent ne pas être pertinents pour le contexte de la recherche.

Le but principal d'une recherche sémantique est d'améliorer la qualité des recherches par différentes techniques, comme l'expansion et la reformulation de requête. Souvent deux difficultés majeures sont rencontrées quand un utilisateur effectue une recherche : (i) comment filtrer les documents non pertinents pour réduire le bruit, et (ii) comment découvrir et sélectionner tous les documents pertinents pour augmenter le rappel.

Récemment, une approche très prometteuse pour la recherche sémantique sur le web est basée sur la combinaison de pages web et de requêtes soumises avec un background de connaissances structurées dans une ontologie. Dans cette perspective, nous présenterons dans le cadre de cette thèse un modèle de raisonnement dont le but est d'augmenter la robustesse des recherches sémantiques sur le web par la prise en compte du bruit dans les résultats et du manque d'expressivité (manque de connaissances) dans les représentations basées ontologies. Notre raisonnement tient sur trois aspects, en premier nous suggérons une nouvelle méthode de segmentation sémantique de texte en fragments sémantiques pour mieux élucider la sémantique enclose dans le texte.

Ensuite, et comme mode de raisonnement nous associons une recherche syntaxique basée expansion par mots clés, avec une recherche sémantique basée concepts en utilisant un système multi-agents pour résoudre ce type de problèmes distribués. Enfin et pour classer les documents résultats, nous proposons une nouvelle mesure de similarité entre concepts basée taxonomie et vérifiant des critères appropriés pour mesurer la distance entre les documents disponibles et les besoins utilisateurs en informations.

Mots clés : recherche sémantique d'information, système multi-agent, web sémantique, langages web, ontologie, annotation sémantique, métadonnée, moteur d'inférence, mesure de similarité, WordNet

Table des Matières

INTRODUCTION GENERALE …… 1

CHAPITRE I : MODELES CLASIQUES DE REPRESENTATION ET DE RECHERCHE D'INFORMATIONS

I.1 Introduction ……..…….…..…… 6

I.2 Problématique du processus de recherche d'informations ……...…… 6

I.3 Concepts de base pour la modélisation d'un système de recherche d'informations. 7

I.3.1 La requête ……..…….…….……. 8

I.3.2 Les documents ……...…..…….…… 9

I.3.3 Indexation ou Annotation ? ……..……...…….. 9

a. Indexation classique …….…….….. 9

b. Annotation sémantique et concept Métadonnée …… 14

I.3.4 Modèles de correspondance Document-Requête ……..……. 16

I.4 Modèles de Recherche d'informations ……...…….. 17

I.4.1 Le modèle booléen ……..…… 17

I.4.2 Le modèle à espace vectoriel (SVM)…… 18

I.4.3 Le modèle Latent Semantic Indexing (LSI) …… 19

I.4.4 Le modèle probabiliste ……. 20

I.5 Critères d'évaluation des modèles de recherche …….. 20

I.6 Systèmes classiques de recherche d'informations …… 22

Conclusion …… 26

CHAPITRE II : MODELES DE REPRESENTATION DES CONNAISSANCES, CONCEPTS DE BASE ET LANGAGES DU WEB SEMANTIQUE

II.1 Introduction ……. 28

II.2 Gestion des connaissances ……... 28

II.2.1 Donnée, information et connaissance ……...…… 29

II.2.2 Classification des connaissances ……..…… 30

II.2.3 Aspects du raisonnement ……..……... 31

II.3 Modèles de représentation des connaissances ……. 31

II.3.1 Les réseaux sémantiques …….....…… 32

II.3.2 Les graphes conceptuels(Gcs) …….……. 34

II.3.2.1 Sémantique logique des GCs …….…… 35

II.3.2.2 Raisonnements dans les GCs ……. 37
II.3.3 Les représentations logiques ……..…….. 39
II.3.3.1 Raisonnements logiques ……..…….. 40
II.3.3.2 Distribution du raisonnement ……... 41
II.3.3.3 Explication et planification du raisonnement ……... 41
II.3.4 Les logiques de description …… 42
II.3.5 Modèles basés ressources terminologiques ...……..…… 43
II.3.5.1 Vocabulaire contrôlé …… 43
II.3.5.2 Taxonomie ……... 44
II.3.5.3 Thésaurus ……. 44
II.3.6 Les ontologies …… 45
II.3.6.1 Buts d'utilisation des l'ontologie ……. 46
II.3.6.2 Eléments de base constituant l'ontologie ……..…… 46
II.3.6.3 Types d'ontologies ……... 48
II.3.6.4 Principes méthodologiques de construction d'ontologies …… 49
II.3.6.5 Méthodes de construction des ontologies …… 50
II.4 Le Web sémantique ……..…….. 52
II.4.1 Historique ……... 52
II.4.2 Définitions et standards du web sémantique ……..……... 53
II.4.3 Composants du Web sémantique ……..……. 55
II.4.3.1 Ontologie …….. 55
II.4.3.2 Ressources …… 55
II.4.3.3 Langages …….. 55
Conclusion …….. 64

CHAPITRE III : RECHERCHE SEMANTIQUE D'INFORMATION SUR LE WEB BASEE SYSTEMES MULTI-AGENTS (SMA)

III.1 Introduction …….. 65
III.2 Outils d'annotation sémantique ……..…... 66
III.3 Mesures de similarités sémantiques ……..…… 68
III.3.1 Méthodes basées « Edge Counting » …… 69
III.3.2 Méthodes basées « Information Content » ……... 71
III.4 Paradigme agent en intelligence artificielle …...……...……. 73
III.4.1 Architecture interne d'un agent …… 75

III.4.1.1 L'agent cognitif ……..…….. 75
III.4.1.2 L'agent réactif ……. 76
III.4.2 Environnement d'un système multi-agents ……..….. 76
III.4.3 Les système multi-agents (SMA)……..….. 77
III.4.3.1 Définitions …… 77
III.4.3.2 Communication dans les SMA …… 78
III.4.3.3 Protocoles d'interaction dans les SMA …… 80
III.4.4 Systèmes de recherche sémantique d'informations …….. 82
Conclusion……...…… 87

CHAPITRE IV : UN MODELE DE RAISONNEMENTS POUR UN SYSTEME DE RECHERCHE SEMANTIQUE D'INFORMATIONS SUR LE WEB BASE AGENTS.

IV.1 Introduction ……..…….. 88
IV.2 Contexte théorique du modèle …… 88
IV.3 Architecture du système ……. 89
IV.3.1 La taxonomie WordNet …… 89
IV.3.2 L'ontologie de domaine …… 91
IV.4 Processus d'annotation sémantique …… 91
IV.4.1 Extraction de termes ……. 91
IV.4.2 Segmentation sémantique ……. 92
IV.4.3 Analyse syntaxique des segments sémantiques ……. 96
IV.4.4 Génération d'annotations sémantiques ……. 96
IV.5 Architecture du système multi-agents …… 98
IV.5.1 Agent « Interface » ……... 99
IV.5.2 Agent « Requête » …… 99
IV.5.2.1 Recherche par expansion de requête …….. 100
IV.5.2.2 Recherche sémantique …… 101
IV.5.3 Agent « Information » ……. 103
IV.5.4 Agent « Classement (Ranking) » ……. 104
IV.5.5 Agent « Ontologie » ……. 108
Conclusion ……..……... 109

CHAPITRE V : ETUDE DE CAS ET ASPECTS D'IMPLEMENTATION DES COMPOSANTS DU MODELE 111

V.1 Introduction ……..…….….. 114

V.2 Segmentation sémantique ……..……...…….. 115

V.3 Mesure de similarité sémantique …….…… 117

V.4 Spécification de l'ontologie de domaine ……. 118

V.5.Génération et expansion de requête syntaxique ……..……... 118

V.6 Recherche Sémantique …….……..…… 118

V.6.1 Modèle d'inférence ……..……. 119

V.6.1.1 Schéma d'inférence ……..…… 122

V.6.1.2 Modèle de données ……. 123

V.7 Implémentation du SMA……..…….. 123

V.7.1 Plate Forme JADE (Java Agent Development Framework) …….. 125

V.7.2 Composants d'un agent Jadex …… 125

V.7.2.1 Structure de l'ADF …….. 125

a. Les croyances (beliefs) …….. 126

b. Les buts (desires) ……... 127

c. Les intentions (plans) …….

Conclusion ……..……

CONCLUSION GENERALE …… 129

BIBLIOGRAPHIE ……. 131

Table des Figures

Figure I.1 : Science de l'information multidisciplinaire ……..……...…… 6
Figure I.2 : Processus de la recherche d'information ……..……...…… 8
Figure I.3: Indexation classique et Annotation sémantique ……...…….. 9
Figure I.4 : Fréquence et discrimination de termes d'indexation ……..… 12
Figure I.5 : Annotation sémantique basée ontologie …… 15
Figure I.6 : Vecteurs documents et requêtes dans l'espace des termes …… 18
Figure I.7 : Matrice Terme x Document (nxm) …… 19
Figure I.8 : Répartition des documents envers une requête ……. 21
Figure I.9 : Architecture du système AGATHE …….. 24
Figure I.10 : Architecture d'un système multi-agents pour la RI sur le Web …….. 25
Figure II.1: Cycle de gestion de connaissances ……... 28
Figure II.2 : Relations données –Informations –Connaissances …….. 30
Figure II.3 : Exemple de réseau sémantique …… 32
Figure II.4 : Exemple de graphe conceptuel …… 34
Figure II.5 : Hiérarchies types de concepts, et types de relations binaires …….. 36
Figure II.6 : Projection du GC « H » dans le GC « G » …….. 38
Figure II.7:Architecture des bases de connaissances en LDs …….. 43
Figure II.8: Base de connaissances et interprétations en LDs ……. 43
Figure II.9 : Les relations dans un thésaurus …… 45
Figure II.10 : Le triangle sémiotique et l'interprétation ontologique …….. 47
Figure II.11: L'élément « Concept » ……..……. 47
Figure II.12: L'élément « Relation » …… 48
Figure II.13 : Type d'ontologies selon leur dépendance ……. 49
Figure II.14 : Phase de construction d'une ontologie …….. 50
Figure II.15 Vision du Web Sémantique ……. 53
Figure II.16 : Pyramide des langages du web sémantique …….. 54
Figure II.17.a : Exemple de document XML ……. 56
Figure II.17.b : DTD associé à un document XML ……. 56
Figure II.17.c : XSD associé à un document XML …….. 57
Figure II.18 : Graphe RDF d'un document XML …… 59
Figure II.19 : Origines du langage OWL ……. 61
Figure III.1 : Composants d'annotation sémantique …… 65

Figure III.2 : Action de l'agent sur l'environnement ……... 74
Figure III.3 : Positionnement des SMA dans l'IA ……... 77
Figure III.4 : Structure d'une performative FIPA …… 80
Figure III.5 Taxonomie de coordination …….. 81
Figure III.6 : Processus de recherche ……... 83
Figure III.7 : Architecture basée SMA pour la recherche sémantique d'informations …… 85
Figure IV.1 : Sous hiérarchie de WordNet relative au concept « Hôtel » …… 90
Figure IV.2 : Processus d'annotation sémantique …… 92
Figure IV.3 : Délimitation de segments sémantiques …….. 94
Figure IV.4 Les vecteurs X, Y et Vsim ……. 95
Figure IV.5 : Construction du graphe conceptuel …….. 97
Figure IV.6 : Architecture du Système Multi-Agents ……. 98
Figure IV.7: La Délibération dans un agent "BDI" ……..…….. 99
Figure IV.8: Diagramme de séquence AUML du SMA ……. 100
Figure IV.9 : Processus de recherche sémantique …… 101
Figure IV.10 : Structure de l'agent « Requête » …….. 102
Figure IV.10 : Structure de l'agent « Information » …… 103
Figure IV.11 : Mesure de similarité basée structure de l'hiérarchie …… 106
Figure IV.12 : Structure de l'agent « Classement » ……. 108
Figure IV.13 : Structure de l'agent « Ontology » …… 109
Figure V.1 : Similarités entre paragraphes Ph1 et Ph2 …….…….. 113
Figure V.2 : Similarités entre paragraphes Ph2 et Ph3 …….…….. 114
Figure V.3 : Résultats et comparaison des approches LDSim et Wu et Palmer ……. 115
Figure V.4 : Graphique des variations des similarités LDSim et Wu et Palmer ……. 115
Figure V.5 : Protégé-2000 pour spécifier l'ontologie « Hotellerie » …….. 116
Figure V.6 : Graphe des relations « IS-A » ……..…….. 116
Figure V.7 : Document et annotations RDF associées …… 117
Figure V.8 : Documents retrouvés ……..…….…… 117
Figure V.9: Modèle de données (Annotations) …….…….. 120
Figure V.10 : Graphe conceptuel de requête « Gr » ……. 121
Figure V.11 : Graphe conceptuel d'annotation « Gd » …… 121
Figure V.12 : Inférence sur le type « Hotel » ……. 122
Figure V.13 : Composantes d'un agent Jadex ……. 124
Figure V.14 : Chargement de l'ADF d'un agent ……. 124

Figure V.15 : Eléments de l'ADF …… 125

Liste des Tableaux

Tableau I.1 : Mesures de rappel et de précision ……... 22
Tableau V.1 : poids et longueur de mots extraits de Ph1,Ph2 ……. 112
Tableau V.2 : Matrice « Terme x Paragraphe » ……..….... 112
Tableau V.3 : Profondeur et mesure de similarités de mots entre Ph1,Ph2 …….…..... 112
Tableau V.4 : poids et longueur de mots extraits de Ph2,Ph3…….…..…. 113
Tableau V.5 : Matrice « Terme x Paragraphe » ……...……..... 113
Tableau V.6 : Profondeur et mesure de similarités de mots entre Ph2,Ph3 …...……..…..... 113

Introduction Générale

On ne peut pas aller de l'informel au formel par des moyens formels. *Alan Jay Perles*

- **Contexte de la recherche**

L'explosion quantitative de l'information produite par l'avènement d'Internet, plus précisément du Word Wide Web, a permis à des millions de personnes de créer, partager, diffuser et publier des quantités gigantesques d'informations sous formes de documents structurés, semi structurés, et non structurés. La conséquence de cette irruption des technologies de l'information, est que le domaine de la recherche d'informations, a connu à son tour un bouleversement majeur affectant l'ensemble des démarches et méthodes utilisées.

Etant un domaine fort de plusieurs années de travaux de recherche de bibliothécaires, de documentalistes, de professionnels du domaine et d'informaticiens, et avec l'accumulation documentaire toujours croissante et de manière exponentielle, il est devenu pressant à s'interroger sur les nouvelles stratégies à mettre en place pour accéder aux ressources d'informations pertinentes qui répondent aux besoins des requêtes utilisateurs.

Dans cette perspective, plusieurs moteurs de recherche sont nés (Excite et Yahoo 1994, AltaVista 1995, Google 1997/98, etc.), offrent des sélections multicritères et des services de filtrage avancés tout en mesurant la pertinence des résultats obtenus.

Cependant le majeur défi celui de satisfaire de manière précise et pertinente les besoins des utilisateurs demeure toujours d'actualité. Face à ce grand volume d'informations les moteurs de recherche se sont investis dans les méthodes de traitements statistiques et/ou probabilistes, qui s'appuient sur des modèles mathématiques, ayant certes enregistrés des progrès remarquables, mais sans une nette amélioration.

La recherche sémantique d'information constituait alors une nouvelle dimension, une vision qui considère un mot comme une référence à des concepts, et aux relations entre ces concepts et non plus comme une simple chaine de caractères diminuée de toute sémantique linguistique. La réflexion d'en faire du web actuel, un web sémantique s'est amplement maturée, l'enjeu est de rendre les ressources d'informations accessibles et manipulables par les utilisateurs certes, mais aussi par des agents logiciels.

Dans ce cadre, une série graduelle de langages de représentation des connaissances ont été développés, leur but est d'offrir les primitives nécessaires à la description de nos connaissances conformément à un vocabulaire contrôlé défini dans des

ontologies, offrant la possibilité de pouvoir raisonner sur ces connaissances à travers des inférences valides. A cette fin, et pour structurer les informations sur le web, Tim Berners Lee, initiateur du web sémantique a défini en 1998, le concept de métadonnées formelle.

Selon Tim Berners-Lee, le web sémantique est un ensemble de méthodes et de technologies permettant à des agents logiciels de manipuler et de raisonner sur le contenu des ressources du Web. Cependant, le web actuel est encore loin de cet idéal. Cette nouvelle vision du Web futur dépend de la description et l'utilisation de métadonnées, ce sont des données qui définissent d'autres données, notamment leur sémantique. Cette amélioration repose sur la notion d'ontologie, qui selon Gruber elle définie comme une spécification explicite d'une conceptualisation.

En effet, les ontologies sont de nos jours au cœur des recherches modernes incluant plusieurs domaines comme l'ingénierie des connaissances, le traitement automatique du langage naturel, la recherche d'information et les systèmes collaboratifs de façon générale. Cette thèse se situe dans le contexte de la recherche sémantique d'information sur le web, plus précisément, ce travail vise l'élaboration d'un modèle de raisonnement pour un système multi agents qui effectue cette recherche.

- **Problématique**

La recherche d'informations pertinente sur le web est devenue avec le temps une tâche très complexe. Les ressources concernent divers types de documents, ce sont de simples pages Html, du son, des vidéos, des données structurées dans des bases de données, des logiciels et des connaissances représentées par divers formalismes. C'est ainsi que l'accès à ces ressources devient une tâche fastidieuse qui nécessite l'usage de nouveaux outils et méthodes de traitement de documents puissantes pour les caractériser dans les corpus et faciliter leur recherche et leur exploitation rationnelle.

L'opération d'indexation des documents est l'une de ces traitement fondamentaux, ce processus consiste à repérer dans le contenu du document et vis-à-vis d'un contexte donné, des termes particulièrement significatifs appelés termes d'indexation, ces termes sont mis en correspondance avec le document indexé. L'indexation en elle même ne couvre pas seulement les aspects d'accès aux données, c'est aussi une représentation sous une forme réduite d'un document par rapport à son contenu sémantique.

Les limites des moteurs de recherche d'informations actuels, dits plein texte, sont dues au fait que l'indexation se fait sur des entités lexicales, et se base sur des notions statistiques et de probabilités, ce qui génère un taux de bruit élevé tout comme le

taux de silence, car il est vrai qu'on ne peut garantir que toutes les ressources souhaitées seront sélectionnées. Cette sévère constatation a été à l'origine de l'émergence chez la communauté des chercheurs de nouvelles perspectives, l'idée tend vers l'usage de connaissances qui doivent représenter explicitement la sémantique des ressources documentaires. Cette formalisation devrait permettre une vision unique et partagée pour rendre ces contenus compréhensible et sémantiquement interprétable.

C'est dans ce contexte que se situe la problématique d'annotation sémantique des documents sur le web. C'est une structuration des connaissances pour permettre un raisonnement, elle peut être définie comme étant le processus qui fixe l'interprétation d'un document vis-à-vis d'un contexte donné, cette tâche s'accomplit en associant à ce document une sémantique explicite par l'ajout d'un ensemble de métadonnées.

L'annotation sémantique repose sur l'ensemble de connaissances détaillées et hiérarchisées d'un domaine donné, une ontologie permet de structurer et exploiter des métadonnées pour mettre en correspondance des éléments ontologiques avec des fragments documentaires.

- **Solution proposée**

Dans le cadre de cette thèse nous proposons un modèle de raisonnement pour un système multi-agent de recherche d'information sur le Web. La notion de raisonnement dans ce contexte est déterminée par la conceptualisation de modèle capable de doter les agents de capacités de comprendre le besoin utilisateur, et le contenu des ressources recherchées pour sélectionner et savoir filtrer les réponses pertinentes. Concrètement, l'annotation est le processus d'association de métadonnées ayant une sémantique définie dans une ontologie, à un segment du document.

Le segment pour être une phrase, un paragraphe, ou le document tout entier. L'ensemble d'annotations peut être stockée dans le document lui-même, ou dans un autre document référençant l'entité annotée par son URI. Par rapport à l'indexation classique, une représentation basé concepts devient nécessaire pour résoudre divers problèmes comme la synonymie et la polysémie dans la syntaxe des mots clés. Ainsi l'annotation sémantique représente les documents et les requêtes par des descripteurs appartenant à une terminologie prédéfinie. La terminologie peut être une ressource terminologique (vocabulaire, thésaurus) ou ontologique (taxonomies et/ou des ontologies de domaine).

Avant tout et pour comprendre la sémantique des contenus des documents manipulés, nous proposons une démarche d'annotation sémantique de documents

textuels, qui effectue une segmentation sémantique d'un texte, et utilise une ontologie de domaine pour produire les métadonnées d'annotations.

Ce processus utilise un outil d'indexation comme LUCENE pour produire une description des documents par les termes descripteurs. L'étape suivante consiste à segmenter les textes en plusieurs portions sémantiquement différentes, une démarche originale pour cette segmentation sera décrite.

L'étape d'après utilisera un outil d'analyse syntaxique comme « Connexor » pour produire à chaque segment un arbre de décomposition syntaxique décrivant les relations entre les mots du segment. La définition d'un ensemble de règles de transformation permettra à la dernière étape de ce processus de construire un graphe conceptuel exprimant la sémantique du segment associé. Aussi, partant du fait que la qualité d'un SRI dépend du classement des résultats obtenus par des comparaisons entre les concepts des documents et des requêtes, nous avons jugé nécessaire de proposer une nouvelle mesure de similarité entre concepts basée WordNet.

Cette mesure basée sur la structure de l'ontologie apporte une amélioration par rapport à la mesure de Wu et palmer. Le troisième pilier du modèle de raisonnement regroupe d'une part un processus d'expansion de requête basée WordNet, et d'autre part la recherche d'une opération de projection entre le schéma de la requête utilisateur et l'annotation associé au document. Cette opération repose sur l'exécution d'inférences sur les métadonnées associées aux documents et définis dans le schéma de l'ontologie.

- ***Plan du manuscrit***

Ce manuscrit est organisé en cinq chapitres :

Chapitre I : décrit l'Etat de l'Art destiné aux modèles de représentation et de recherche d'information.

Chapitre II : Se rapporte à l'Etat de l'Art qui concerne les modèles de représentations des connaissances et les langages de description.

Chapitre III : Concerne le concept d'annotation sémantique, les mesures de similarité sémantique, le paradigme agent et les systèmes de recherche sémantique d'informations.

Chapitre IV : Ce chapitre est consacré à la description de notre modèle, les aspects structurel et fonctionnel sont abordés.

Chapitre V : Ce chapitre décrit les aspects d'implémentation des démarches proposées au chapitre 4, ainsi une application de la méthode de segmentation est effectuée sur un fragment de document en entrée, les segments sémantiques constituent les résultats.

De même une expérimentation de la mesure de similarité appliquée WordNet est montré, une requête utilisateur d'un exemple de recherche dans le domaine d'hotellerie a fait l'objet d'un exemple de recherche hybride combinant l'expansion de requête et la recherche sémantique proprement dite.

Nous terminons ce manuscrit par une conclusion générale dressant une synthèse suivie des perspectives futures.

CHAPITRE I

Modèles Classiques de Représentation et de Recherche d'Informations

I.1 Introduction

Il faut discuter des goûts et des couleurs. D'abord parce que toute dispute se réduit à cette espèce, et qu'il faut que l'on dispute. »

Paul Valéry, 1941.

Dans ce chapitre sont décrites les composantes et fonctionnalités principales d'un Système de Recherche d'Information (SRI). Il sera question d'exposer les démarches de représentation des documents, du concept d'indexation classique (les prétraitements) et les différentes démarches utilisées. Le concept d'annotation sémantique est présenté ainsi que les modèles de correspondance entre les requêtes et les documents.

Ce chapitre présente aussi les modèles classiques, leurs caractéristiques, et les critères d'évaluation quantitatifs et qualitatifs. Une présentation des SRI les plus utilisés est faite, des architectures sans ou avec agent sont détaillées à la fin du chapitre.

I.2 Problématiques du processus de recherche d'informations

De nos jours La recherche d'informations est apparue comme une discipline qui a l'objectif de satisfaire les besoins de gérer l'immense quantité d'informations toujours croissante. La recherche d'informations comme science se situe à l'intersection de deux disciplines qui sont les sciences de l'information et l'informatique, les sciences de l'information est aussi un autre carrefour de plusieurs autres disciplines comme le montre la figure I.1.

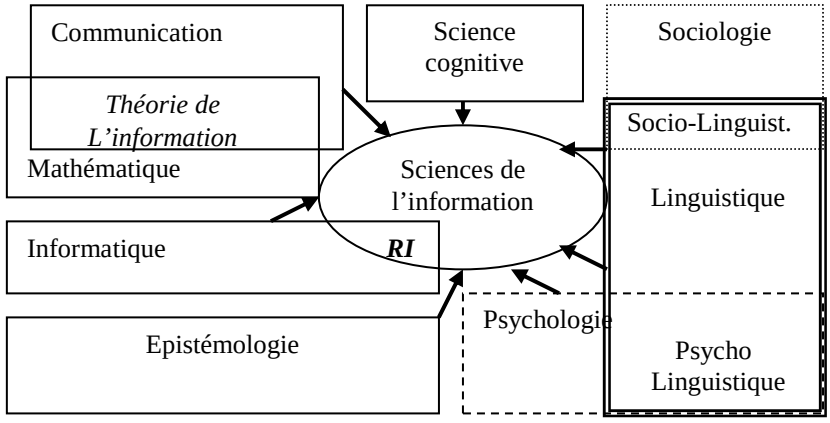


Figure I.1 : Science de l'information multidisciplinaire [yae,2009]

La définition [Rij,1979] énonce «*L'utilisateur exprime son besoin d'information sous la forme d'une requête en vue d'obtenir de l'information. La RI consiste à restituer les documents qui peuvent être pertinents par rapport au besoin d'information exprimé dans la requête.Il est probable que ce procédé soit réitéré puisque la requête demeure un moyen*

imparfait d'expression du besoin d'information et que les documents restitués à un moment donné permettent d'améliorer la requête utilisée pour la prochaine itération».

Les recherches se sont ensuite focalisées sur les systèmes de recherche de l'information « SRI », le but étant de mettre en œuvre des techniques et des mécanismes pour développer des outils puissants permettant la gestion automatique de l'information documentaire (les documents, leurs métas donnés et les relations inter documents). Les besoins sont présentés sous forme de requêtes, le modèle d'un SRI aura pour fonction de représenter l'ensemble des documents disponible dans un corpus, et aussi de disposer de mesures de comparaison pour évaluer la pertinence des résultats renvoyés.

Si l'on considère les moteurs de recherche d'informations couramment utilisés, à l'exemple de Google, Yahoo et autres, et vis-à-vis des besoins des utilisateurs, ces moteurs de recherche retournent certainement en partie des documents pertinents, mais aussi un ensemble de liens inutiles pour le contexte des recherches effectuées. Etant basés sur des approches de recherche par mots clés pour retrouver l'information, les documents et les requêtes sont traitées comme des ensembles de termes indépendants, le succès d'une recherche dépendra fortement du choix des mots clés utilisés pour formuler les requêtes.

L'amélioration des systèmes de RI, passe par la compréhension des modèles existants, la problématique de cette thèse est l'établissement d'une passerelle pour surpasser du concept syntaxique par un concept sémantique. L'état de l'art que nous présenterons dans ce chapitre reprend les modèles des SRI les plus fréquents dans les applications de recherche de l'information. Une classification des modèles fait ressortir les familles suivantes :

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

I.3 Concepts de base pour la modélisation d'un S.R.I

La croissance permanente du volume d'informations a vite suscité l'intérêt des spécialistes, notamment en ce qui concerne l'automatisation du stockage et la consultation de l'information. Aussi le développement d'outils et de méthodes pour gérer ces quantités d'informations est devenu une nécessité absolue.

Un SRI a la charge d'assurer plusieurs aspects fonctionnels, la Figure I.2 illustre les principales fonctionnalités et leurs articulations, nous expliciterons en particulier les modules de:

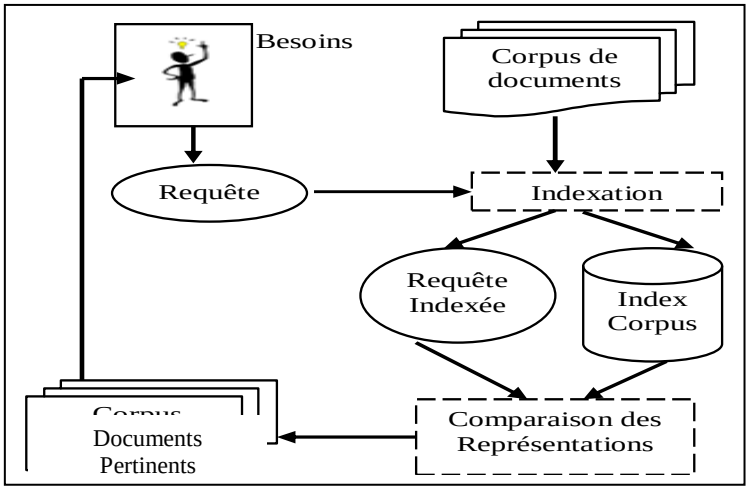


Figure I.2 : Processus de la recherche d'information

- Formulation de requêtes
- L'indexation des documents et des requêtes
- Comparaison requête-documents, et classement des documents par ordre de pertinence.
- Présentation des documents reconnus pertinents.

I.3.1 La requête

La requête qui traduit les besoins en informations est formulée ou reformulée par l'utilisateur pour initier le processus de recherche. La requête qui est le produit conceptuel d'une analyse des besoins est formée de concepts clés et de relations entre ces concepts. C'est pendant le processus de recherche que les besoins d'informations de l'utilisateur sont plus précis et mieux exprimés [Yae,2009], [Kuh,1990].

Cette constatation est due au fait que l'incertitude et la confusion décroissent avec les multiples tentatives de l'utilisateur à vouloir exprimer exactement ses besoins en choisissant avec plus de précision les termes de recherches. Une fois formulée la requête peut avoir généralement la forme d'une expression en langue naturelle, ou encore d'une liste de concepts avec éventuellement un degré d'importance associé, ou une formule logique de concepts coordonnés par des opérateurs logiques.

I.3.2 Les documents

Le terme « Document » trouve son origine du latin « Documentum » qui signifie leçon, exemple, modèle. Ce terme partage la même racine que « doctrine » ou docteur de la famille du verbe « docere » qui signifie instruire, enseigner [Yae,2009]. En conséquence, le document est porteur de sens et son contenu s'exprime en une forme interprétable par

l'utilisateur. Avec l'avènement des ordinateurs, le document devient numérique et on peut alors le stocker par une représentation qui s'apprête aux traitements informatiques.

Dans l'optique de cette thèse nous focalisons l'étude sur les documents textuels reconnus être le support le plus porteur d'informations partageables. De manière analogue à la requête les documents de corpus doivent être « indexés » pour être traité par un SRI.

I.3.3 Indexation ou Annotation ?

Les recherches menées lors de décennies sur les représentations de données textuelles sont très diverses [Lew,1990], [Sal,1986], [Spa,1974], leurs buts communs étant de réduire la complexité des représentations pour en faciliter les traitements automatiques. Un SRI, a l'objectif de renvoyer une liste de documents pertinents par rapport à une requête utilisateur, par conséquent, il est nécessaire de pouvoir rechercher les documents de la collection dont le contenu ressemble au contenu de la requête.

La recherche implique une méthode d'évaluation et de tri par comparaison des concepts du contenu documentaire candidat avec les concepts formulés dans la requête. Les concepts sont exprimés par des termes qui sont des unités linguistiques porteuses de sens et devant refléter au mieux et le contenu documentaire et le besoin utilisateur.

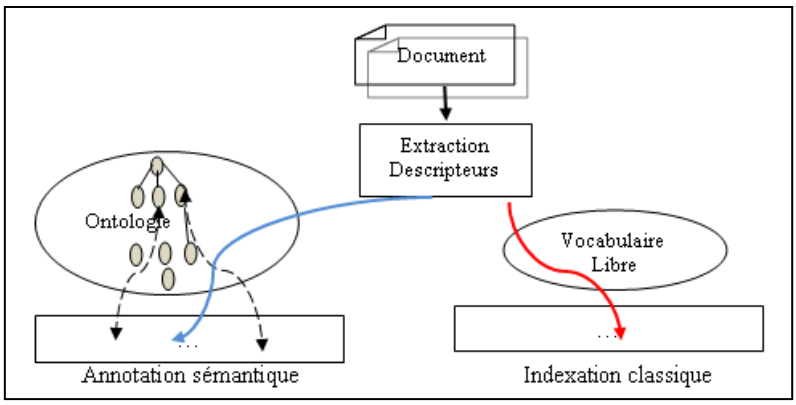


Figure I.3: Indexation classique et Annotation sémantique

a. Indexation classique

L'association française de normalisation (AFNOR NF Z 47-102, octobre 1993), définit l'indexation par [Mar,2004]: « *L'indexation est l'opération qui consiste à décrire et à caractériser un document à l'aide de représentations des concepts évoqués dans ce document, c'est-à-dire à transcrire en langage documentaire les concepts après les avoir extraits du document par une analyse.* »

L'opération d'indexation classique qu'elle soit manuelle, semi-automatique ou automatique nécessite une profonde analyse du contenu de chaque document de la collection

« Figure I.3 ». Cette analyse se fait sur plusieurs étapes [Yae,2009], le but étant d'extraire les termes représentatifs du contenu et d'évaluer leur pouvoir de représentation et de caractérisation du document dans lequel ils apparaissent. Pour choisir les termes d'indexation, nous avons plusieurs alternatives, les systèmes actuels représentent les concepts par des mots seuls ce sont des unités linguistiques faciles à reconnaître, et qui sont assez porteuse de sens.

Néanmoins les mots ne donnent pas toujours une description précise du concept, ainsi d'autres approches tentent de regrouper des mots pour former des termes, elles se basent soit sur une analyse syntaxique et/ou statistique, ou un dictionnaire de mots composés.

L'idée de choisir un groupe de mots comme représentant de concept semble être justifiée par une perte du sens lorsqu'on considère des mots isolés, par exemple le terme « recherche d'information » porte plus de sens que les mots « recherche » et « information » séparés, mais l'expérimentation n'a pas montré une nette amélioration parce que les termes posent aussi des problèmes d'interprétation contextuelle.

L'indexation s'effectue selon des étapes :

- **Analyse lexicale**

Cette étape du processus d'indexation transforme le document textuel en un ensemble de termes appelés des lexèmes.

- **Filtrage**

Lors de cette phase, plusieurs techniques peuvent être utilisées pour sélectionner les termes candidats à l'indexation. En premier les mots vides et mots fonctionnels (articles, pronoms, prépositions, certains adverbes, certains adjectifs …) seront supprimés car ces mots n'apportent aucun plus au contenu informationnel du document, ce processus utilise par exemple un stoplist (anti-dictionnaire) des mots vides comme pour SMART [Sal,1988].

- **Lemmatisation**

Beaucoup de mots ont des formes syntaxiques légèrement différentes, ce sont des variantes morphologiques, mais leur sens restent le même ou très similaire. C'est notamment le cas des mots conjugués. Les mots suivants ont des sens très similaires: transformer, transforme, transforment, transformation, transformateur, …La différence de forme syntaxique entre ces mots est inutile pour la RI, souvent on voudrait trouver des documents contenant « transformation » à partir d'une requête sur "transformer", il faut ramener ces mots à une forme identique, le « Lemme », par élimination des terminaisons (désinences) de mots, et garder seulement le radical (racine), c'est l'idée de la lemmatisation.

Les algorithmes de radicalisation des mots sont divers : un premier algorithme linguistique, dit « Porter » consiste à examiner la forme du mot pour déduire la racine en éliminant les affixes (suffixes et préfixes) [And,1971]. Cet algorithme élimine les terminaisons de mots en anglais en cinq grandes étapes: la première étape essaie de transformer le pluriel en singulier. Les étapes suivantes essaient d'éliminer au fur et à mesure les dérivations (par exemple -ness qu'on ajoute derrière certains adjectifs (happiness), -able ajouté derrière un verbe (adjustable).

Cet algorithme a fait preuve de son efficacité [And,1971], [Daw,1974], pour la plupart des cas il donne des résultats appréciables, il est considéré comme un algorithme classique souvent utilisé pour la lemmatisation, il a été appliqué à d'autres langues comme le français, italien et l'allemand. D'autre algorithmes se basent sur des méthodes statistiques comme par exemple les n-grammes [Ada,1974] ou être hybrides comme [Kro,1993], [Pai,1996]. Ils peuvent également se baser sur des lexiques afin de valider ou d'invalider une tentative de transformation d'un mot en radical [Sav,1993].

- **Pondération**

La pondération consiste à associer à un terme d'indexation un poids qui représente sa capacité représentative et discriminatoire des documents du corpus. Cette caractérisation traduit le pouvoir informatif du terme pour le document donné, en effet, à une information sont attachés un sens et une probabilité qui permet de quantifier, de mesurer son contenu d'informations, plus une information est probable moins elle sera informative.

Dans une liste de mots Zipf [Zip,1949], a montré que si l'on classe dans une liste les mots d'indexation par ordre décroissant de leurs fréquences, alors la distribution de la fréquence d'un mot est inversement proportionnelle à son rang de classement, c'est-à-dire : *rang* fréquence = constante*

Cette relation permet de choisir les termes d'indexation, qui doit être le plus informatif possible. L'informativité d'un terme mesure la quantité de sens qu'il porte, ainsi plus un terme est fréquent dans la collection, moins il sera discriminatoire, c'est la fréquence absolue. De même un terme peu fréquent dans un document ne peut être représentatif de son contenu, c'est la fréquence relative.

En limitant la valeur Rang*Fréquence entre les deux bornes « Seuil_Max » et « Seuil_Min », nous pouvons éliminer tous les termes dont la valeur dépasse « Seuil_Max » et aussi les termes ayant la valeur Rang*Fréquence au dessous de « Seuil_Min ». Les termes à

valeurs Rang*Fréquence dans l'intervalle [Seuil_Min, Seuil_Max] sont conservés et sont donc informatifs. Figure I.4.

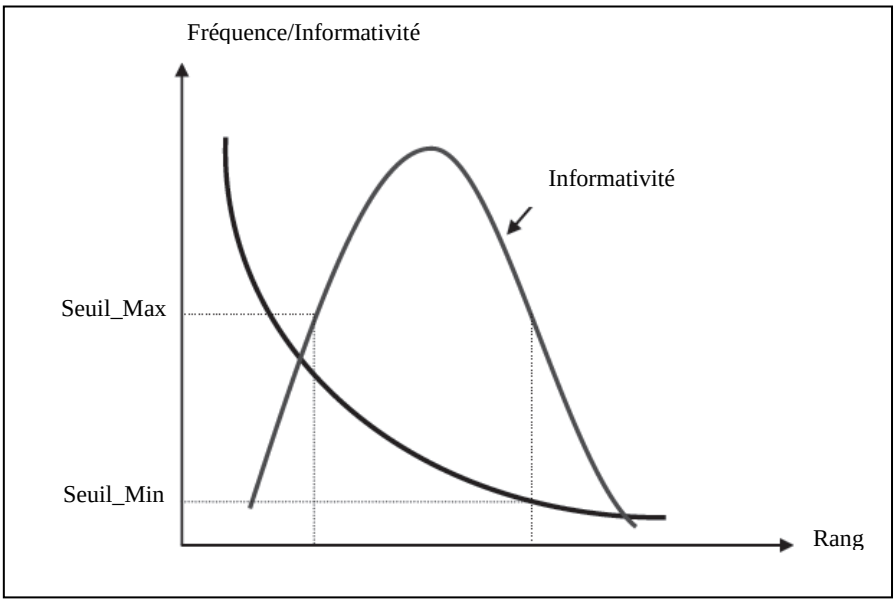


Figure I.4 : Fréquence et discrimination de termes d'indexation

Pour calculer le poids d'un terme, nous avons plusieurs approches qui s'adaptent aux modèles du SRI, nous présentons ci-dessous ces différentes approches en évoquant lors des descriptions le modèle SRI.

- ✓ **Basée sur {0 ou 1}**

Deux valeurs binaires sont associées aux termes d'indexation et expriment la présence (1) ou l'absence (0) d'un terme dans le document, s'adapte au modèle booléen.

- ✓ **Basée sur la fréquence d'occurrence**

Dans cette approche on sélectionne les mots qui représentent le mieux le contenu d'un document. On part de l'hypothèse qu'un mot qui apparaît souvent dans un texte représente un concept important. Ainsi, cette approche consiste à choisir les mots représentants selon leur fréquence d'occurrence f(t,d), t :terme,d :document, après avoir éliminé les mots vides. La façon la plus simple consiste à définir un seuil sur la fréquence: si la fréquence d'occurrence f(t,d) d'un terme « t » dans le document « d » dépasse ce seuil, alors il est considéré important pour le document.

- ✓ **Basée sur la valeur de discrimination**

Par discrimination on décrit le fait qu'un terme distingue un document des autres documents de la collection, un terme ayant une valeur de discrimination importante apparaîtra dans un petit nombre de documents, cette mesure discriminatoire est importante dans la mesure

où les termes discriminants sont gardés pour la constitution des index des documents. La valeur de discrimination d'un terme se calcule comme suit :

1- Calcul d'un vecteur moyen V (centroïde) du corpus : le poids de chaque terme dans ce vecteur est le poids moyen de ses poids dans l'ensemble des documents, c'est-à-dire: $Pj = (\sum_i Pij)/N$ (1) N : Nbre de documents du corpus
2- Calcul de l'uniformité du corpus par une mesure de similarité moyenne des documents avec le vecteur centroïde V ; $U1 = C * \sum_i Sim(Di, V)$ (2) C : constante de normalisation (par exemple : 1/N), et Sim (Di, V) est la similarité entre le document Di et le vecteur V exprimée dans [0,1]
3- On uniforme le poids du terme en question à « 0 », et on répète les étapes précédentes (1 et 2) pour obtenir une nouvelle valeur d'uniformité soit U2.
4- La valeur de discrimination du teme est U=(U2-U1) (3)

- ✓ **Basée sur Tf*Idf**

Tf: *Term-frequency* est la fréquence du terme dans le document c'est-à-dire le nombre d'occurrences d'un terme dans le document.Cette fréquence peut être évaluée selon les expressions suivantes :

- Tf=f(t,d) (4) terme est fonction de son nombre d'occurrences.
- Tf=log(1+f(t,d)) (5) par cette heuristique, on peut exploiter une propriété de la fonction logarithmique ,qui discrimine peu deux termes ayant leurs nombre d'occurrences proche et élevés, et les différencie de façon importante dans le cas contraire.
- $Tf = \frac{f(t,d)}{\max(f(t_i,d))}$ (6)

Considère l'importance du terme « t » par rapport à la fréquence du terme le plus présent dans le document, c'est une forme qui offre l'avantage de normalisation.

Idf : Inverse *of Document Frequency* est la fréquence absolue inverse, c'est un facteur qui varie inversement proportionnel au nombre « *n* » de documents où un terme apparaît dans une collection de « N » documents. La fréquence absolue inverse peut avoir l'expression:[Sal,1987] [Sal,1971] :

- Idf = log (N/n) (7) N: le nombre total de documents dans la collection et *n* le nombre de documents où le terme apparaît.

Le *poids* d'un terme « j » dans le document « i » s'écrit [Spa,2004]:

Wij = Tfij×Idfj (8) Tfij : fréquence d'apparition du terme j dans le document i et Idfj est la fréquence absolue inverse du terme j dans le corpus.

Ainsi le poids d'un terme augmente si celui-ci est fréquent dans le document et décroît si celui-ci est fréquent dans la collection.

Le poids « w_{ij} » du terme «t_i » dans le document « j » est généralement donné par l'expression : [Sch,2005]

$$w_{ij} = \frac{f(t_i, d_j)}{\max(f(t_i, d_j))} * \log\frac{N}{n} \qquad (9)$$

D'autres approches utilisent des combinaisons de schémas de calcul de poids, et exploitent des résultats issus d'expériences pratiques pour choisir l'expression donnant les meilleurs résultats possibles [Tan,2005]. En plus ces schémas permettent de différencier entre les poids de termes de documents et les poids associés aux termes des requêtes.

$$w_{ij} = (0.5 + 0.5 * \frac{T_{f\,ij}}{\max T_{f\,ij}}) * \log\frac{N}{n} \quad (10)$$

$$w_{ik} = \frac{W^{*}_{ik}}{\sqrt{\sum_{i=1}^{m} W^{*2}_{ik}}} \qquad (11) \quad \text{et} \quad w^{*}_{ik} = T_{f\,ik} * \log\frac{N}{n_i}$$

Aussi, une normalisation des valeurs du poids peut être effectuée par les expressions suivantes [Yae,2009] :

$$w_{ij} = \frac{Tf_{ij} * Idf_j}{\sum_{k=1}^{t} Tf_{ik} * Idf_k} \quad (12) \qquad ; w_{ij} = \frac{Tf_{ij} * Idf_j}{\sqrt{\sum_{k=1}^{t} (Tf_{ik} * Idf_k)^2}} \quad (13)$$

t : nombre de termes

b. Annotation sémantique et concept de métadonnée

Le terme « *annotation »* est défini dans le dictionnaire comme étant « une note explicative ou critique qui accompagne un texte- une note de lecture que l'on inscrit sur un livre ». Ce terme dérive du terme latin « *Annotare* » qui signifie « accompagner un texte de notes, de remarques et commentaires » et parce que nous annotons un sujet, une annotation seule n'a aucun sens, elle est associée a l'objet qui a été annoté.

Nous nous intéressons principalement à l'annotation de document par le contenu, l'un des aspects du web sémantique est de pouvoir créer et manipuler les annotations sémantiques, ce sont des Métadonnées de documents [Tua,2006].

La sémantique de l'annotation est fondée sur des vocabulaires dans les ontologies spécifiées explicitement par un langage de représentation, par conséquent l'annotation sémantique est le processus le plus adapté pour le partage du contenu des documents.

Une annotation sémantique peut alors être définie comme :<<*Une représentation formelle d'un contenu, exprimée par des concepts, des relations, et instances décrits dans une ontologie reliée à le ressource documentaire*>>**.**

Les buts de l'annotation sémantique peuvent être classés en trois catégories :

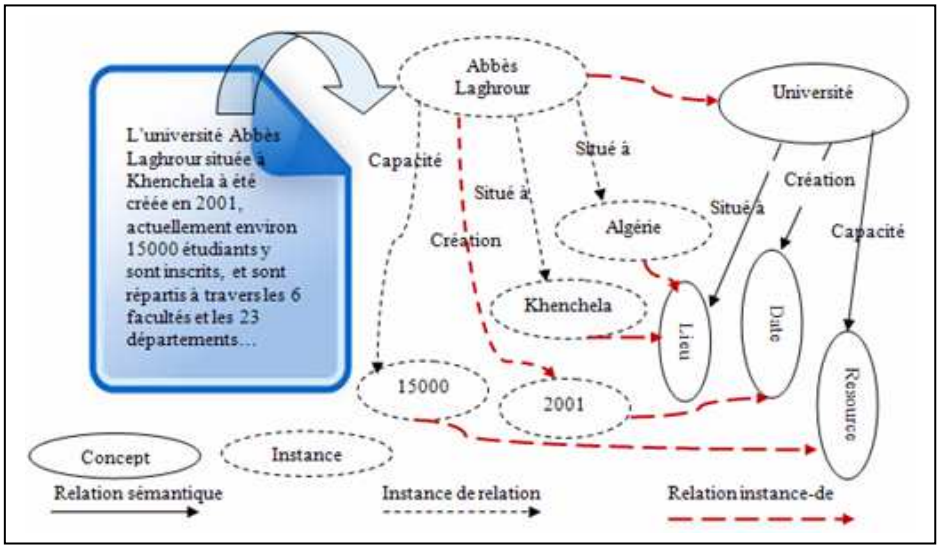


Figure I.5: Annotation sémantique basée ontologie

1) Spécifier une sémantique implicite intégrée, pour la compréhension des objets annotés.

2) Identifier la sémantique commune aux objets annotés depuis différentes ressources, pour faciliter leurs transferts, leurs échanges et leur mise en correspondances.

3) Permettre aux machines de comprendre, raisonner et vérifier les objets annotés.

Les approches d'annotation sémantique se distinguent par les types de connaissances ontologiques utilisés, et par la nature et la granularité des parties de document annotées. Les éléments ontologiques utilisés pour l'annotation peuvent être classés en cinq classes:

- Les concepts : termes définis dans une ontologie.
- Les instances de concepts.
- Les relations.
- Les instances de relations.
- Les valeurs littérales.

Le terme de métadonnées a été principalement associé à la balise <meta> du langage HTML, une forme d'indexation des pages Web « Figure I.5 ». Ensuite, ce terme a été élargi à

des ressources pour véhiculer des notions variées, les métadonnées sont très diverses, elles peuvent être des données créées par l'humain ou la machine; ou des données destinées à l'humain ou à la machine.

Dans [Deb,2012], une métadonnée est définie par « *une représentation ré-interprétable, sous forme conventionnelle convenant à la communication, à l'interprétation ou au traitement* ». Les métadonnées doivent être explicites modélisées et exprimées de façon formelle, pour cela les ontologies constituent le réceptacle de cette modélisation.

Une autre définition utile [Haa,2004]: *« Une métadonnées est une donnée, qui transporte des connaissances relatives décrivant un sujet, sans requérir l'inspection et l'examen du sujet en lui-même.»*

I.3.4 Modèles de correspondance Document-Requête

Cette étape évalue la pertinence des documents retournés par rapport à la requête soumise, plusieurs méthodes d'évaluation de cette ressemblance existent, les plus connues sont décrites ci-dessous [Yae,2009], [Sal,1987]. Le document et la requête sont représentés par leurs vecteurs poids issus de l'étape d'indexation, donc un document « D », et une requête « Q » auront les représentations suivantes : $D=((t_0,w_{d0}),(t_1,w_{d1}),\ldots,(t_k,w_{dk}))$ et $Q=((q_0,w_{q0}),(q_1,w_{q1}),\ldots,(q_k,w_{qk}))$

Les « w_{di} » et « w_{qi} » représentent les poids du terme « t_i » dans le document « D » et dans la requête « Q ». « k » correspond au nombre de termes dans l'espace.

Étant donnés les vecteurs poids « D » et « Q » leur degré de ressemblance est évalué par diverses approches, les plus connues sont:

- Le produit scalaire : $\text{Similarite}(D,Q) = \sum_i w_{di} * w_{qi}$ (14)
- La formule du cosinus $\text{Similarite}(D,Q) = \dfrac{\sum_i w_{di} * w_{qi}}{\sqrt{\sum_i w_{di}^2 * \sum_i w_{qi}^2}}$ (15)
- Le coefficient de Dice : $\text{Similarite}(D,Q) = \dfrac{\sum_i w_{di} * w_{qi}}{\frac{1}{2}((\sum_i w_{di}^2) + (\sum_i w_{qi}^2))}$ (16)
- La mesure de Jaccard : $\text{Similarite}(D,Q) = \dfrac{\sum_i w_{di} * w_{qi}}{\sum_i w_{di}^2 + \sum_i w_{qi}^2 - \sum_i w_{di} * w_{qi}}$ (17)

Les systèmes de recherche après avoir évalué cette similarité retourne à l'utilisateur une liste de documents classés par leur degré de pertinence. L'indexation des documents et de

la requête et l'évaluation de la pertinence sont des paramètres déterminants pour juger la qualité du modèle de la recherche d'informations.

En exploitant les feedback utilisateur, il est souvent initié un algorithme de reformulation de requête par la technique dite de réinjection de pertinence. Il peut être envisagé une réadaptation du vecteur requête par ajout de nouveaux termes ou par une redistribution de poids. Dans [Sel,1997], nous avons une étude comparative des méthodes de réinjection de pertinence.

I.4 Modèles de recherche d'informations

Un modèle de recherche d'information spécifie pour le système qui l'utilisera les démarches à suivre pour l'accomplissement des étapes décrites précédemment. Ce sont la représentation des documents et des requêtes, l'indexation, la recherche proprement dite, l'évaluation des correspondances documents-requêtes et si besoin en est, la reformulation de la requête.

I.4.1 Le modèle booléen

Dans ce modèle une recherche d'information consiste à trouver les documents qui contiennent les mêmes termes (éventuellement avec poids) que la requête construite à base de mots clés. Dans ce sens, un document se rapportant à « professeurs » ne sera pas retrouvé pour répondre à une requête qui concerne les « chercheurs » lorsque le document ne contient pas le terme « professeurs », il est évident que « professeurs » est un type de « chercheurs ».Les requêtes peuvent être formulée par des termes reliés par les opérateurs logiques de base, à savoir « AND », « OR » et la négation « NOT ». Comme indiqué plus haut le document est représenté par son vecteur index, c'est-à-dire : $d = t_1, t_2, \ldots t_n$. La requête est représentée par une expression logique de termes avec des opérateurs logiques.

Parmi les avantages de ce modèle nous avons l'exactitude des représentations des concepts, en plus ce modèle permet de résoudre partiellement le problème de synonymie en utilisant l'opérateur « OR » [Fra,1992]. Il convient aussi pour exprimer un terme (ensemble de mots) par l'opérateur « AND », et il est facile à implémenter.

Les insuffisances, concernent la nécessité de savoir utiliser et interpréter les formulations booléennes, l'autre désavantage est que les documents ne sont pas présentés par ordre de pertinence, tous les documents retournés ont la même mesure de similarité envers la requête soumise.

I.4.2 Le modèle à espace vectoriel (SVM)

Le modèle vectoriel, est un modèle mathématique [Sch,2005] [Sal,1971]. Il permet de représenter les documents et les requêtes par des vecteurs d'un espace à «n » dimensions, les « n » repères étant constitués par les termes d'indexation.

Le principe de l'approche vectorielle consiste en une transformation des données textuelles en une représentation numérique, utilisant des vecteurs et des matrices et des techniques statistiques et d'analyses matricielle pour la découverte des caractéristiques de connections dans un ensemble de documents. Dans le modèle vectoriel chaque document et chaque requête est représenté par un vecteur de « t » dimensions (t : nombre de termes).

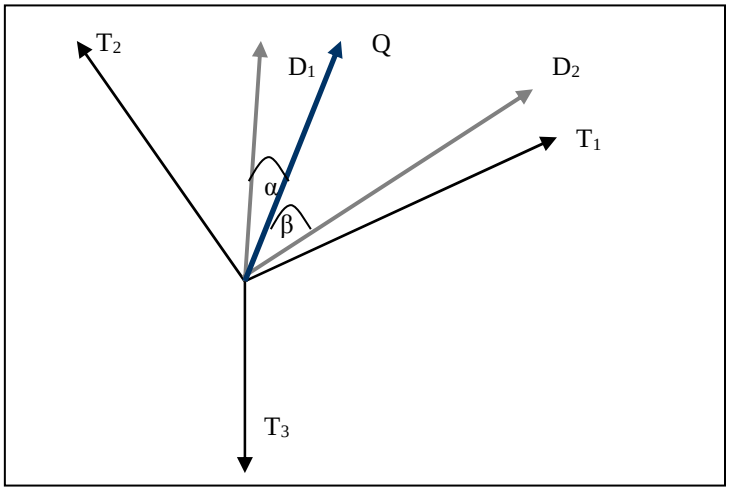


Figure I.6 : Vecteurs documents et requêtes dans l'espace des termes

Chaque terme du corpus représente une dimension de l'espace considéré, ensuite le codage de la représentation vectorielle est réalisé soit par une fonction booléenne, ou par une fonction de nombre d'occurrences de termes dans les documents.

La Figure I.6 est un exemple de représentation de deux document « D1 » et « D2 » et d'une requête « Q » dans un espace de trois dimensions (T1,T2,T3). En appliquant la similarité donnée par la formule du cosinus vue précédemment, nous remarquons que plus deux vecteurs sont similaires, plus l'angle formé est petit, et plus le cosinus de cet angle est grand [Mar,2004]. Dans la Figure I.6, le document « D_1 » est plus similaire à la requête « Q » que le document « D_2 ». A la différence du modèle booléen, la fonction de similarité évalue une correspondance partielle entre un document et une requête, ce qui permet de retrouver des documents qui satisfont approximativement la requête. Les résultats peuvent donc être ordonnés par ordre de pertinence décroissante.

Le modèle vectoriel ne considère pas les relations entre les termes, les mots clés sont indépendant, d'où l'orthogonalité des dimensions de l'espace, c'est une insuffisance apparente,

mais dans la pratique il semble que la prise en compte de ces dépendances n'a pas conduit à améliorer notablement la qualité du modèle [Rag,1986].

Dans le modèle à espace vectoriel, on utilise une matrice dite « Terme x Document » pour représenter les termes d'indexation et les documents du corpus. Chaque ligne représente un document, et chaque colonne représente un terme de l'index comme illustré en Figure I.7.

	T_1	T_2		T_j		T_n
D_1	W_{11}	W_{12}		W_{1j}		W_{1n}
D_2	W_{21}	W_{22}		W_{2j}		W_{2n}
D_i	W_{i1}	W_{i2}		W_{ij}		W_{in}
D_m	W_{m1}	W_{m2}		W_{mj}		W_{mn}

Figure I.7 : Matrice Terme x Document (nxm)

Les différentes évaluations réalisées ont montré que le modèle vectoriel donne des résultats satisfaisants [Bae,1999]. Les performances en temps de réponse et la qualité des résultats restent appréciables même quand le nombre de dimensions est grand, par conséquent nous avons choisi de l'utiliser dans nos travaux relatifs à cette thèse.

I.4.3 Le modèle Latent Semantic Indexing (LSI)

Le modèle LSI est une variante du modèle vectoriel standard, qui pour améliorer les représentations et les performances, cherche à réduire le nombre de dimensions des vecteurs. L'idée suppose l'existence d'une structure sémantique latente dans un corpus de documents, étant donnée une matrice Terme x Document A(m,n), une colonne de cette matrice est un document donné par le vecteur d'occurrence des termes qui le composent [Vac,2005].

Cette matrice est projetée dans un espace de dimensions plus faible, où les descripteurs considérés ne sont plus de simples termes (les termes apparaissant ensemble sont projetés sur une même dimension), c'est une représentation qui vise à résoudre partiellement les problèmes de synonymes et des termes polysèmes [Sch,2005].

Le contexte mathématique :

A(t,d) :matrice termes par documents

Il existe pour A une factorisation de la forme : $A_{txd} = T_{txn} S_{nxn} D^T{}_{nxd}$ (18)

T : est une matrice unitaire (txn) orthogonale

S : est une matrice (nxn) dont les éléments diagonaux sont des réels positifs, et tous les autres sont nuls, c'est une matrice diagonales, les éléments diagonaux sont les valeurs singulières de la matrice A.

D^T : est une matrice (nxd) orthogonale.

Le rang de la nouvelle matrice Â est égal au nombre des valeurs singulières non nulles.

$$\hat{A}_{txd} = T_{txk} S_{kxk} D^T{}_{kxd} \qquad (19)$$

I.4.4 Le modèle probabiliste

Ce modèle s'appuie sur des théories de probabilité, et considère que les termes d'indexation sont indépendants, et que leur probabilité d'apparition est la même avec ou sans la présence des autres termes. Sous cette hypothèse, le problème revient à estimer la probabilité qu'un document retourné soit pertinent par rapport à la requête. Dans cette perspective des théories de probabilités des approches [Rij,1979] [Boo,1983] [Fuh,1989] ont été développées.

Le modèle probabiliste tente d'estimer la probabilité de la pertinence (respectivement la non pertinence) d'un document notée P(*PERT*|*D*) (resp. P(*NPERT*/D)) . Seules la présence et l'absence de termes dans les documents et dans les requêtes sont considérées comme des caractéristiques observables, 0 (absent) ou 1 (présent).

La similarité entre une requête *q* et un document *d* est déterminée par :

$$\text{Similarité}(d,q) = \frac{P(\text{Pert}/d,q)}{P(\text{NPert}/d,q)} \qquad (20)$$

Plus cette proportion est élevée pour un document, plus ce document est pertinent pour la requête, les formule de Bayes sont introduites pour le calcul de ces probabilités.

I.5 Critères d'évaluation des modèles de recherche

Les critères de mesure de qualité des modèles de recherche et des ensembles de données ont été développés afin de tester ces systèmes sur une base commune. Des systèmes de recherche d'information basés sur des modèles différents donnent des résultats différents. Les éléments à considérer sont de nature qualitative et quantitative, mais, il est difficile de disposer d'une démarche analytique formelle pour évaluer les différents SRI. Il est souvent procédé par une démarche expérimentale d'évaluation [Yae,2009] [Mar,2004].

La qualité d'un système doit être mesurée en comparant les réponses du système avec les réponses que l'utilisateur espère, pour réaliser une telle évaluation, l'expérimentation utilise les éléments suivants:

- Un ensemble de documents.

- Un ensemble de requêtes.
- La liste de documents pertinents pour chaque requête.
- Des mesures et des critères quantifiables.

Parmi les mesures quantitatives nous avons « Disc Access Cost » ou « DAC » qui mesure le nombre d'accès au disque et le temps total pris pour l'indexation et la recherche [Vac,2005]. Suivant la requête, les documents de la collection sont répartis en partitions selon deux caractéristiques comme indiqué en Figure I.8.

- Les documents sélectionnés et les documents non sélectionnés.
- Les documents pertinents et les documents non pertinents.

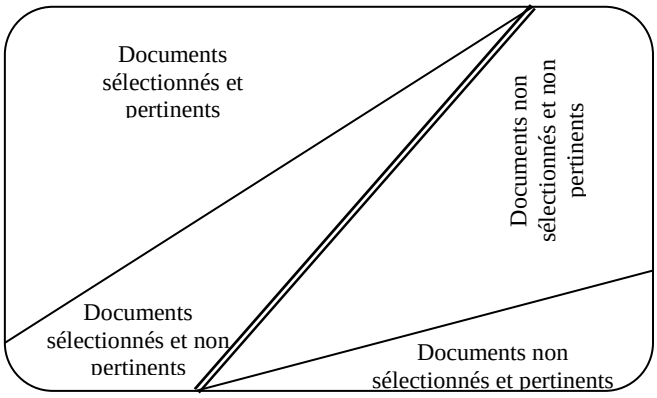


Figure I.8 : Répartition des documents envers une requête

Les mesures qualitative concernent deux paramètres essentiels : la précision et le rappel.

a) **Le Rappel (Recall):** La mesure de rappel est définie comme étant le rapport entre le nombre de documents pertinents trouvés et le nombre de documents pertinents disponibles dans la collection.

b) **La Précision(Precision):** C'est la proportion de documents pertinents parmi l'ensemble de documents sélectionnés.

	Pertinents	Non Pertinents
Sélectionnés	A	b
Non Sélectionnés	C	d
Total	P	np

Tableau I.1 : Mesures de rappel et précision

$$0 \le \text{Precision} = \frac{a}{a+b} \le 1 \quad (21)$$

$$0 \le \text{Rappel} = \frac{a}{a+c} \le 1 \quad (22)$$

Une faible précision se traduit par le fait que l'utilisateur devra prendre du temps à lire des informations qui ne l'intéressent pas, c'est une conséquence d'une forte présence de polysémie alors qu'un faible rappel signifie que l'utilisateur n'aura pas accès un ensemble d'informations pertinentes et souhaitables, c'est l'effet que provoque les mots synonymes.

Egalement on définit le bruit et le silence comme étant des notions complémentaires de la précision et du rappel, on a donc : Bruit = (1-Precision) et Silence= (1-Rappel).

La précision et le rappel sont fortement liés par une relation inversement proportionnelle, en effet quand l'une augmente, l'autre diminue. L'explication de cette observation est triviale, car plus nous avons beaucoup de document sélectionnés, moins ils traiteront tous précisément du sujet de la requête, et inversement [Yae,2009]. L'idéal serait d'équilibrer ces deux métriques, par exemple en augmentant la précision mais pas trop au dépend du rappel.

c) La mesure F Précision

Soit la précision « Pn » et le rappel « Rn » relatives au sous ensemble des documents constitués des « n » premiers documents retournés. Il serait intéressant d'analyser ces mesures, comme on l'a vu ces deux mesures évoluent souvent de façon opposée. La précision est globalement décroissante au fur et à mesure que le SRI restitue des documents, alors que le rappel est globalement croissant.

On peut choisir la mesure F comme valeur synthétique exploitant la précision et le rappel, son expression est :

$$F = (2 \times \text{Rappel} \times \text{Précision}) / (\text{Rappel} + \text{Précision}) \qquad (23)$$

Pour évaluer l'ordonnancement, il est possible de calculer la mesure Fn à chaque rang n :

$$Fn=2 \times Rn \times Pn / (Rn+Pn). \qquad (24)$$

I.6 Systèmes classiques de recherche d'informations

Dans cette section, nous exposons quelques travaux relatifs au développement de systèmes de recherche d'information, nous mettrons l'accent sur les travaux dont les démarches de développement partagent des fondements théoriques ressemblent à ceux utilisés dans ce travail.

Le modèle exposé dans [Vac,2005] se base sur le modèle LSI « Latent Semantic Indexing », il décrit une démarche de mappage de termes avec les concepts de l'ontologie WordNet pour réduire les dimensions des vecteurs représentatifs, et améliorer le rappel et la

précision. Les auteurs justifient leur démarche par le fait que plus les grandeurs dimensionnelles sont élevées plus l'efficacité d'un SRI se dégradent.

Aussi en cherchant un document donné, nous ne sommes pas sûrs d'avoir choisi les termes exacts pour la requête, ce qui peut influencer les résultats en général. Le modèle LSI est une méthode numérique qui permet de découvrir la sémantique latente du document en créant des concepts à partit de termes.

Le modèle est une version du modèle vectoriel, après la décomposition de la matrice initiale A(m,n) dite terme par documents en valeurs singulières, seules les plus grandes valeurs singulières de rang « k » seront retenues pour ramener la décomposition de la matrice « A » en une décomposition singulière de rang « k ». $A_K = U_K \sum_k V_K^T$

A la place de A_K une matrice dite concept par document définie par $D_K = \sum_k V_K^T$ Avec k lignes est utilisée pour l'exécution de la requête Q dans le nouveau espace des concepts. Un vecteur réduit associé à la requête est créé $Q_K = U_K^T Q$.

Appliquée à WordNet, cette démarche utilise pour un terme donné, les hypernyms d'un niveau « i » à un niveau « j » dans l'hiérarchie pour remplacer les termes par les concepts. Lorsque i=j=0 alors on obtient pour les termes du document leurs correspondants synsets dans WordNet, les poids des concepts sont calculés par la somme des poids des termes pondérés par un coefficient évalué par la distance séparant le terme des hypernyms.

Nous exposons ce travail parce qu'il présente une ressemblance avec l'idée que nous expliciterons ultérieurement et dans le cadre de cette thèse. Cela concerne en particulier le processus d'expansion de requête basé WordNet, les deux démarche favorisent l'amélioration du rappel mais au détriment de la précision. Evidemment, les dimensions seraient réduites, car chaque synset contient plusieurs termes, en particulier pour les grandes collections le nombre de synsets utilisés serait inférieur au nombre de termes initialement retenus.

Un autre travail est présenté dans [Esp,2007], l'idée de base repose sur la recherche d'information dans des domaines restreints. Il est espéré d'améliorer les performances d'un SRI, c'est dans cette orientation que les auteurs proposent une architecture de collecte d'information sur des domaines particuliers du web, en s'appuyant sur des ontologies qui décrivent ces domaines pour prendre en compte le contexte de la recherche.

L'architecture générique proposée « Agent information Gathering » ou AGATHE utilise des agents logiciels pour effectuer une collecte coopérative d'information. D'un point de vue l'intégration des agents est justifiée par la modularité, la vitesse d'exécution due au parallélisme et à la fiabilité.

D'un autre coté l'utilisation de l'ontologie répond à un besoin de représentation de connaissances déclaratives pour devoir traiter des données semi-structurées ou non structurés, l'expressivité, les possibilités d'inférence et l'héritage multiple. L'architecture AGATHE fait ressortir trois sous systèmes en interactions comme indiqué en Figure I.9, le premier consacré à la recherche proprement dite, a la charge d'interroger des moteurs de recherche externes, comme Google, Yahoo etc. Il reçoit des requêtes en provenance des clusters d'extraction située dans un autre sous système d'extraction, et récupère des documents web en réponse aux requêtes. C'est un système multi agents où sont situés des agents coopérants.

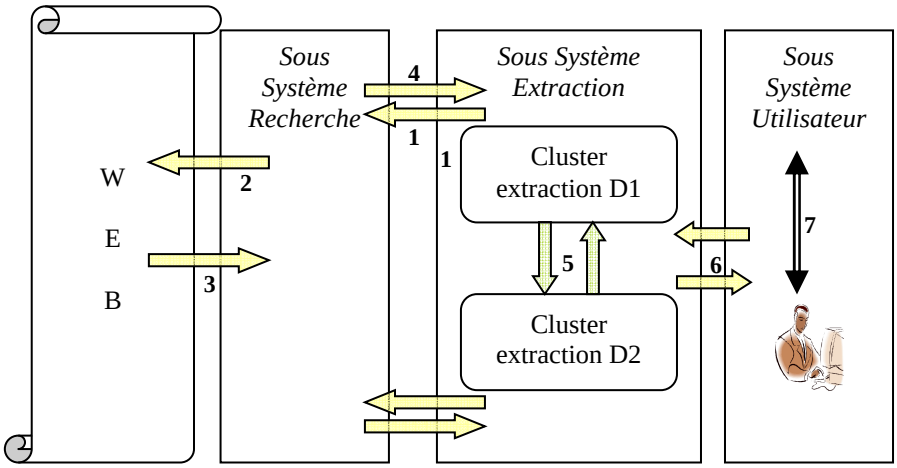


Figure I.9 : Architecture du système AGATHE [Esp,2007]

Le sous système d'extraction, regroupe des clusters d'extraction auquel sont rattachées des ontologies, c'est aussi un système multi agents d'agents extracteurs d'informations, de classifications sémantique, d'a*gents préparateurs qui* reçoivent les pages web renvoyées par la recherche, ils sont créés et supprimés par un agent superviseur du même cluster. Ces agents réalisent un premier traitement des pages web qui consistent à la validation des pages, d'agents de stockage, agent de recommandation et de supervision.

Le troisième sous système a la charge de maintenir les interactions entre AGATHE et les utilisateurs. Il est composé de deux principaux composants qui sont : le médiateur et l'interface utilisateur. La RI dans l'architecture AGATHE ne tient compte que des pages HTML, autrement dit les documents d'autres types ne seront pas validés.

L'analogie du système AGATHE avec notre travail se situe au niveau d'allocations de tâches aux agents, de l'existence d'agents superviseur qui coordonne les activités du système, et aussi la création d'agents de recherche sur le web en fonction de la charge.

Dans le même ordre d'idée et selon une architecture semblable, d'un système multi agents de recherche d'informations sur le web est présenté dans [Wog,1999], les connaissances

à propos des agents sont formalisées dans une base de connaissances sous forme de règles structurées dans la base de faits (tableau noir) selon trois niveaux.

Un niveau concerne les compétences des agents, un autre niveau dédié à la communication (messages et modes d'envoi/réception), dans le dernier nous avons le problème, et les stratégies de résolution.

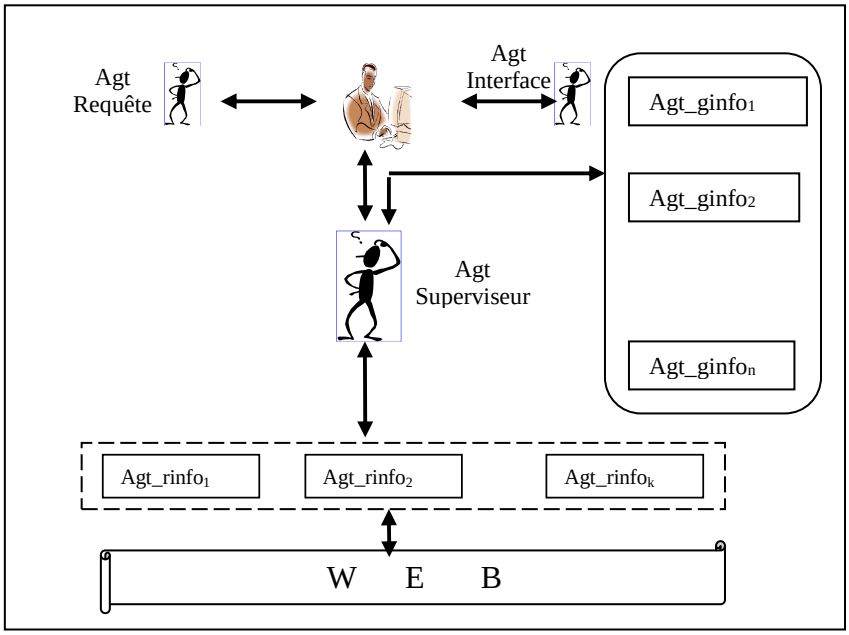


Figure I.10 : Architecture d'un système multi-agents pour la RI sur le Web

L'architecture comme le montre la Figure I.10, comporte trois ensemble d'agents, le premier regroupe un agent interface utilisateur, un agent requête et des agents gestionnaires d'informations, dans le second nous avons les agents et des outils de recherche d'informations, et dans le dernier ensemble sont situés deux agents , l'un joue le rôle de superviseur, l'autre a un rôle d'administrateur. L'utilisateur choisit l'interface qui lui convient et compose sa requête avec l'assistance de 1'agent requête. Une fois la requête raffinée, elle est envoyée à l'agent superviseur qui le décompose et l'analyse, et connaissant le modèle de chaque agent dans I' environnement, il choisit le ou les agents les plus aptes à effectuer la recherche. Les agents sélectionnés effectuent les recherches, les résultats obtenus sont traites par les agents gestionnaires d'informations et présentés à l'utilisateur.

Pour terminer nous exposons brièvement deux systèmes largement utilisés dans le domaine d'indexation et de recherche d'information qui sont SMART et LUCNE

Lucene, est un système permettant d'ajouter des fonctionnalités de recherche plein-texte, il est possible d'indexer et retrouve des "documents". Le document est une structure de données constituée de champs. Un champ est une donnée possédant un nom (titre, auteur, date de

publication, contenu, ..) et à laquelle est associé du texte. C'est ce texte qui est indexé et recherché, les documents indexés sont regroupés au sein d'une collection de documents appelée "index".

Un index peut contenir un très grand nombre de documents (milliers ou millions) et il est possible de créer autant d'index différents qu'il est nécessaire. Si le texte qui est à indexé est contenu dans des fichiers Excel, Word, PDF ou HTML, il est envisagé par l'utilisateur d'en extraire de contenu textuel qui sera indexé.

SMART est l'un des premiers systèmes développé à base du modèle vectoriel [Sal,1971], dans ce système les documents textes et les requête sont analysés automatiquement par une variante de procédures syntaxiques, statistiques et sémantiques. Il fournit des possibilités d'usage de différentes méthodes et l'analyse et la comparaison de l'information et des requêtes ce qui permet d'ajuster les recherches en fonction des résultats obtenus lors de différents cycles de traitements. Cette flexibilité est assurée par un superviseur contrôle qui effectue les fonctions centrales suivantes :

- lecture de données en entrée, spécifiant les requêtes, et les documents de la collection sous une forme qui s'apprête au traitement informatique (vecteurs de poids de termes).
- les documents sont groupés sous divers critères pour accélérer les algorithmes de recherches.
- sélections de groupes de documents susceptibles de contenir l'information
- recherche de l'information proprement dite dans la collection des documents sélectionnés
- évaluation des performances des recherches exécutées.

Conclusion

Les moteurs de recherche d'informations sur le web se basant sur les modèles de représentation décrits dans ce chapitre, tenant compte de certaines spécificités et vis-à-vis des besoins des utilisateurs en informations, retournent certainement une partie des documents pertinents, mais aussi un ensemble de liens inutiles pour le contexte des recherches effectuées. Etant basés sur des approches de recherche par mots clés pour retrouver l'information, les documents et les requêtes sont traitées comme des ensembles de termes indépendants, le succès d'une recherche dépendra fortement du choix des mots clés utilisés pour formuler les requêtes et représenter les documents.

Ces modèles bien que simples n'emploient aucune notion sémantique et ne permettent pas de différencier entre les documents qui même s'ils partagent des termes similaires, ils présentent différentes relations, l'exemple de termes « *school library* » et « *library school* »

montre que le sens ne parvient pas seulement en mettant ensemble des termes, mais de la relation qui existent entre les mots.

La principale faiblesse dont souffrent ces modèles est le fait de négliger le contexte des recherches, l'une des raisons qui explique pourquoi un système de recherche d'information n'a pas un domaine d'intérêt explicite, est que la plupart des utilisateurs ont tendance à utiliser peu de termes pour formuler les requêtes en langage naturel. Par conséquent le système sera dans l'incapacité de comprendre le contexte de la requête et génère des résultats imprécis, la requête utilisateur représente le contexte et le but de la recherche, il est évident qu'en incluant le contexte, et en définissant les relations qui existent entre les mots on améliore les recherches.

Enfin, un système de recherche ne peut pas à lui seul répondre aux besoins variés des utilisateurs en matière d'informations sur le web et à propos de n'importe quel sujet, le succès relatif des systèmes experts est essentiellement dû à leur opérationnalisation sur des domaines spécialisés (retreints). C'est une constatation valable aussi pour les systèmes de recherche d'informations sur le web, c'est dans cette orientation que la vision du web sémantique est née, nous retrouvons la problématique de la recherche sémantique d'information, une discipline en plein expansion focalisé sur le développement de systèmes de recherche d'informations qui utiliseraient des outils et des mécanismes leurs permettant de comprendre le contenu sémantique des documents manipulés, pour pouvoir satisfaire les besoins utilisateurs de façon plus adéquate.

Le deuxième chapitre sera consacré pour la description de cette nouvelle dimension, le trait sera fait autour des connaissances et des modèles de leurs représentations, une notion fondamentale qui constitue la base du raisonnement.

CHAPITRE II

Modèles de Représentation des Connaissances, Concepts de base et Langages du Web Sémantiques

II.1 Introduction

La connaissance est devenue depuis le développement des systèmes à base de connaissances un véritable actif stratégique. Etant un support de base du savoir et du savoir-faire, elle est devenue l'objet de toutes les attentions dans toutes les activités humaines. Les organisations, désormais évaluent leur capital de connaissances et améliorent les processus d'acquisition et de préservation qui sont valorisés comme une accumulation de capital.

C'est donc une ressource qu'il faut maintenir et développer, une fois acquise la connaissance est mémorisée pour être réutilisée, les différents supports de mémorisation présentent des caractéristiques spécifiques, notamment les mémoires individuelles, et les mémoires collectives.

Dans ce chapitre, après un exposé sur la gestion de la connaissance la classification et les aspects du raisonnement, des modèles de représentation des connaissances sont présentés avec les modes de raisonnements associés. L'accent est mis particulièrement sur les ontologies comme concept den base de notre modèle, dans ce sens les éléments de base constituant une ontologie sont décrits, les types d'ontologies, les critères et les méthodes de leurs construction les plus connues sont présentées.

L'autre partie sera consacrée aux technologies du web sémantique, notamment il sera question des langages disponibles pour formaliser l'ontologie. Nous passerons en revue l'état de l'art du processus d'annotation sémantique des documents, des différentes mesures de similarités qui sont au cœur du modèle proposé le cadre de notre thèse.

II.2 Gestion des connaissances

L'acquisition des connaissances est un domaine de recherches qui s'est développé avec l'avènement des systèmes experts au début des années 80, et par la suite les systèmes à base de connaissances.

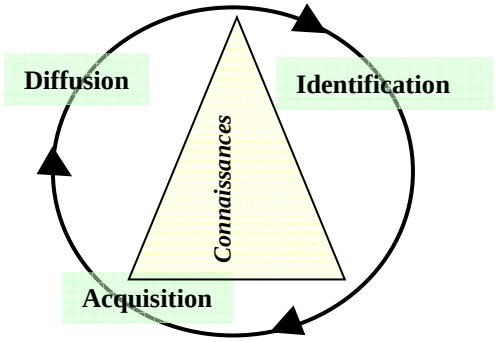


Figure II.1: Cycle de gestion de connaissances [Mon,2008]

La question de la modélisation de l'acquisition est apparue comme une problématique cruciale donnant lieu à de nombreux travaux de recherche, abordant l'aspect cognitif et les

niveaux de représentation des connaissances, le cycle de gestion des connaissances comporte l'identification des connaissances, leur acquisition et leur diffusion.

Dans [Hol,2002] il est arboré la compréhension des phénomènes liés à la gestion des connaissances qui dépendent de : « Figure II.1 ».

- Caractérisation des ressources des connaissances.
- Identification et élucidation des activités des manipulations.
- Reconnaissances des facteurs qui affectent l'aboutissement du processus.

Le cycle de vie des connaissances, fait ressortir que la compréhension des activités de gestion repose sur les macro-processus suivants [Moa,2012]

- **Identification :** (production et acquisition) : Correspond aux choix d'éléments contextuels, plusieurs approches ont été définies à ce niveau, la plus connue est commonKADS, de Schreiber et al [Sch,1999] [Sta,2009].
- **Formalisation** : Outre la définition classique des objets et leurs annotations descriptives, la formalisation devra tenir compte des nouvelles initiatives de recherche, telles que le text-mining, le Natural Language Processing (NLP), Information Retrieval (IR) etc., tout comme les ontologies pour améliorer la dimension sémantique des concepts modélisés
- **Préservation :** La préservation est définie par l'ensemble des processus fonctionnels et applicatifs permettant de conserver la connaissance dans le système pour des utilisations ultérieures.
- **Transfert et partage:** les processus qui se rapportent aux activités de conversion, de communication, d'archivage, et de traductions.
- **Réutilisation :** La dissémination mesure la capacité du système de gestion de connaissances à satisfaire les besoins utilisateurs. Cette activité est souvent prise en compte lors de la conception des modèles de préservation des connaissances.

II.2.1 Donnée, Information et Connaissance

Souvent, nous utilisons plusieurs vocables soit pour définir un même concept, soit pour invoquer des concepts différents comme: donnée, information et connaissance. Une donnée est un fait objectif qui relate un événement, comme une simple observation, c'est un élément brut livré en dehors de tout contexte.

Une donnée n'a aucune valeur en soi, elle représente une entrée à un processus d'interprétation, alors qu'une information réfère à des données qui, compilées ensemble et dans

un contexte donné, peuvent véhiculer un message informatif venant d'une source émettrice à l'intention d'une source réceptrice [Gri,2008] [Mon,2008].

La connaissance, dans ce sens est l'aboutissement de la compréhension et de l'assimilation des règles qui régissent les modèles mentaux sous jacent à ces relations. "Figure II.2".

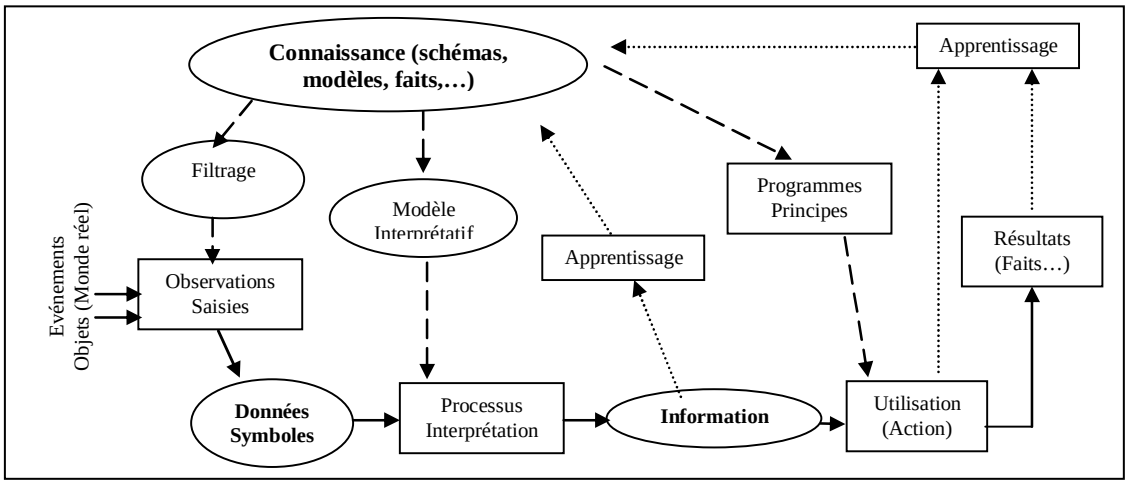


Figure II.2 : Relations données –Informations –Connaissances [Rei,2005]

II.2.2 Classification des connaissances

Les premiers travaux qui ont trait avec la classification des connaissances remontent à ceux de Michael Polanyi [Pol,1966]. Il part du fait que nous pouvons connaître plus que nous pouvons dire :«we can know more than we can tell».

a) Les connaissances tacites et connaissances explicites

Les connaissances tacites est tout le cumule d'un savoir-faire cognitif acquis avec le temps et selon l'intensité des expériences vécues par l'individu, et qu'il peut détenir intuitivement dans sa tête, sans pouvoir les formaliser, de manière objective ou les expliciter de façon communicable et compréhensible pour autrui [Gri,2008] [Lam,2010]. A l'opposite, la connaissance explicite, peut être exprimée dans un langage formel, elle se présente sous diverses formes qui peuvent être des formulations, des spécifications et des schémas, des manuels de procédure, des images et du son.

b) Connaissance déclaratives et connaissances procédurales

La connaissance déclarative, est un savoir exprimé par des propositions et/ou des descriptions de la réalité, il comprend des faits, des objets et des principes attachés à un domaine donné. Les taxonomies de connaissances représentent l'une des plus importantes formes de connaissances déclaratives [Erm,2000]. La connaissance procédurale, s'attache

plutôt aux stratégies et méthodes (conditions et tâches) d'utilisations des connaissances déclaratives pour la résolution de problèmes, c'est le savoir faire.

II.2.3 Aspects du raisonnement

Le raisonnement est un enchainement d'énoncés conduit en fonction d'un but. Le but peut être une démonstration, élucidation, interprétation, ou explication, etc. [Hat,1997].

La nature d'enchainement est une caractéristique importante du raisonnement qui en général nécessite des retours en arrières. Le modèle du raisonnement est inhérent aux connaissances sur lesquelles il opère, ainsi on peut distinguer :

- Le raisonnement formel : qui se concrétise par la manipulation syntaxique de structures symboliques, le raisonnement logique en fait partie.

- Le raisonnement procédural : les connaissances, leurs modes d'utilisation, ainsi que le raisonnement sont expliqués dans des algorithmes ou des automates. Les mécanismes d'attachement de démons dans les systèmes de frames mettent en évidence un tel raisonnement.

- Le raisonnement par analogie : qui observe des situations déjà vécus et résolues pour les adapter aux situations actuelles. C'est le principe du raisonnement naturel humain**.**

- Le raisonnement par abstraction et généralisation : Lié à l'apprentissage par induction, peut mettre en évidence les mécanismes d'inférence par héritage et classification.

II.3 Modèles de représentation des connaissances

L'objectif de l'intelligence artificielle est de reproduire le comportement humain dans les activités de raisonnement pour obtenir de la machine un comportement jugé intelligent, c'est un facteur déterminant dans l'évolution des réalisations dites intelligents.

La représentation des connaissances exprime la modélisation adéquate des connaissances d'un domaine, sous une forme interprétable et manipulable par les opérateurs humains et logiciels, la modélisation est liée à un ensemble de types de raisonnements qui s'intéressent aux problèmes d'activités humaines telles que la perception, la prise de décision, le diagnostic, la planification, la compréhension du langage naturel, l'apprentissage, etc. qui nécessitent une exploitation raisonnée d'une grande quantité de connaissances [Gri,2008].

Le raisonnement est défini comme étant un processus d'élaboration d'inférences par combinaison de connaissances alors que le modèle de connaissance est constitué d'un langage de représentation des connaissances et un ensemble de types de raisonnement sur ces connaissances [Hat,1997].

Le modèle de connaissance doit par ailleurs présenter quelques propriétés à savoir:

- *la clarté* : la correspondance entre la connaissance du domaine, et sa représentation symbolique doit être simple, pour garantir une interprétation rigoureuse.
- *La puissance d'expression* : doit être suffisante, pour que toutes les connaissances nécessaires à la construction du raisonnement puissent être représentées.
- *La simulation :* L'espace mémoire et temps d'exécution des méthodes du modèle.

II.3.1 Les réseaux sémantiques

Les réseaux sémantiques sont apparus suite aux travaux de Quillian en 1968, et les résultats obtenus en psychologie cognitive sur les mémoires associatives. C'est un modèle déclaratif d'une représentation graphique qui exprime le contenu sémantique des concepts d'un domaine, il est fondé sur la notion de graphe formé de nœuds qui représentent les concepts et d'arcs qui relient les concepts [Ber,2001] [Hen,2008].

Un concept donné n'acquiert un sens qu'en considérant les relations qu'il a avec les autres concepts du graphe, le graphe associé à la phrase « Pacha est un chien labrador, il possède une niche »,"Figure II.3".

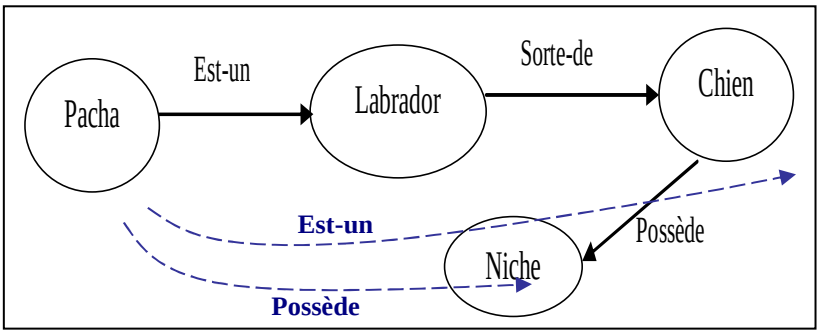


Figure II.3 : Exemple de réseau sémantique

Les types de réseaux d'après [Sow,2013], sont le réseau sémantique définitionnel, le réseau sémantique assertionnel, devenu graphe existentiel puis graphe conceptuel. Le réseau sémantique exécutable et réseau sémantique d'apprentissage. Il y a des modes de raisonnement dans les réseaux sémantiques :

a. Héritage

Lorsque la terminologie exprime des généralité/spécificité des concepts impliqués, alors un arc « IS-A » entre deux nœuds fournit la base pour l'héritage des propriétés. Un concept plus spécifique qu'un autre, hérite les propriétés du concept plus général, dans l'exemple Figure II.3, un « Chien » hérite toutes les propriétés du sous types « Labrador ».

b. Composition de relations

Ce type de raisonnement consiste à inférer l'existence d'une relation « R » entre deux nœuds « N_1 » et « N_2 » s'il existe un chemin entre ces deux nœuds qui fait intervenir « K» relations. $R=R_1oR_2o\ldots oR_K$

Dans l'exemple de la Figure II.3, nous avons l'inférence:

« Pacha » *possède* « Niche » = (« Pacha » *Sorte-de* « Chien » o « Chien » *possède* « Niche »)

(Sorte-de o Possède= Possède)

Ce type de raisonnement se heurte à deux difficultés majeures, la première est que les relations ne sont pas toutes composables, la deuxième a trait aux ambigüités liées à la définition de la relation composée.

c. Propagation d'activation

Cette technique, a l'objectif d'inférer une relation « R » entre deux ou plusieurs nœuds, en se basant sur les arcs qui existent entre ces nœuds, et d'autres nœuds du réseau. Le principe de la technique de propagation d'activation consiste selon l'algorithme de Fahlman, 1979 à :

- Marquage des nœuds sources, par exemple (N1,N2,..)
- Propagation depuis la source du marquage vers l'extrémité le long des arcs « Est-un » et « Sorte-de » des nœuds N1,N2,…
- S'il existe un arc portant la relation « R » entre les nœuds marqués, la relation « R » existe, sinon on poursuit le marquage, jusqu'à ce que le réseau soit saturé.

Ayant un pouvoir expressif satisfaisant du point de vue cognitif, un réseau sémantique peut décrire le « quoi » de domaines de connaissances assez complexes, néanmoins le « comment » du raisonnement n'est pas complet, par le fait qu'il est souvent difficile de fournir des explications sur le raisonnement.

L'autre insuffisance de réseaux sémantique est qu'ils ne disposent pas d'une sémantique formelle, la signification peut être décidée par l'intuition des utilisateurs. De même ils ne fournissent aucune primitive pour représenter la négation et la disjonction, ce besoin critique a initié l'émergence des réseaux sémantiques partitionnés.

II.3.2 Les graphes conceptuels (GCs)

Le modèles des graphes conceptuels « GCs » introduits par Sowa en 1984, est un modèle formel et expressif de représentation des connaissances d'un domaine d'application.

Les Gcs se veulent un langage intermédiaire entre des formalismes destinés aux traitements automatiques et le langage naturel [Gri,2008] [Sow,2000]. Nous mettrons l'accent en détails sur ce formalisme parce qu'il a été à la base des représentations de notre démarche d'annotation sémantique.

Les graphes conceptuels tiennent leurs origines des réseaux sémantiques de Quillian 1968, qui depuis plusieurs extensions ont été proposées, aussi un graphe conceptuel n'a pas de sens propre, c'est au travers des réseaux sémantiques que les concepts et les relations sont rattachés au contexte. Le graphe conceptuel illustré par la Figure II.4, correspond à la phrase « Chaque employé est recruté par quelques managers à certaines dates».

Les connaissances structurelles du domaine sont définies à travers un support qui les organise en taxonomies de type « Sorte-de » et qui hiérarchisent les concepts et les relations, ensuite le graphe conceptuel est crée pour représenter les connaissances factuelles. Cette séparation explicite confère au GCs une grande clarté lors de l'utilisation de ce modèle, la représentation graphique est un graphe fini, orienté, connexe, et étiqueté qui comporte deux types de nœuds (biparti) :

- Des nœuds concepts, étiquetés par des noms correspondant à des types de classes sémantiques, et de référents individuels ou génériques, définis dans le support.
- Des nœuds relations qui sont étiquetés par les noms de relations conceptuelles entre concepts définies dans le support.

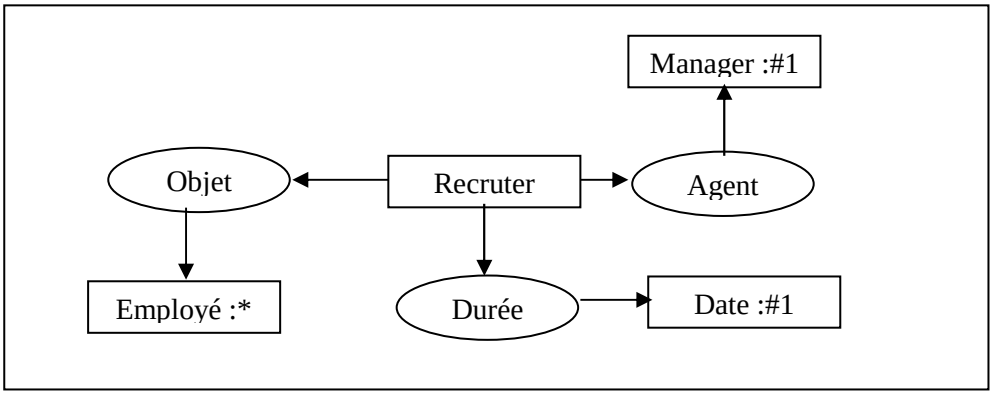


Figure II.4 : Exemple de graphe conceptuel [Sow,2000]

Formellement, de manière simplifiée et selon la définition de Sowa, nous avons :

- $S=(T_C,T_R,\varphi)$, est un support tel que :

(T_C,≤) est l'ensemble partiellement ordonné des types de concepts, avec « ⊤ » le super type de tous les types (universel), et « ⊥ » le sous type de tous les types (absurde).

(T_{Ri}, ≤) est l'ensemble partiellement ordonné des types de relations d'arité « i »

$T_R = T_{R1} \cup T_{R2} \cup \ldots \cup T_{Rp}$

Φ : l'ensemble des référents individuels, auquel on ajoute le référent (marqueur) générique noté « * ».

Ainsi, on définit un graphe conceptuel par [Cro,2007] :

- CG= [S, G, λ] où :

S : est le support

G = (V_C, V_R,E) : graphe orienté biparti.

V=($Vc \cup V_R$) est l'ensemble des nœuds du graphe avec :

Vc : l'ensemble fini des nœuds concepts.

V_R : l'ensemble fini des nœuds relations.

E : l'ensemble des arcs {v_r,v_c) tel que $v_r \in V_R$ et $v_c \in Vc$.

λ : V→ S, est une application label telle que :

pour $v \in Vc$ $\lambda(v) = (type_v, ref_v)$; $type_v \in T_C$ and $ref_v \in (\Phi \cup \{*\})$;

si $r \in V_R$ alors $\lambda(r) \in T_R$.

II.3.2.1 Sémantique logique des graphes conceptuels

Les graphes conceptuels sont menés d'une sémantique formelle, donc ils permettent à la machine de disposer d'un formalisme directement exploitable t.

A chaque support et à chaque graphe conceptuel on peut associer une formule de la logique du premier ordre, cette transformation s'obtient en :

- A tout type de concept du support, on associe un prédicat unaire ayant le nom du type.
- A tout type de relation on associe un prédicat de même arité que le type relation.
- A tout référent individuel, on associe une constante.

Plus précisément si l'on note Φ cette transformation nous obtenons :

Au support S, la transformation Φ(S) associe les formules logiques suivantes :

- Au type universel ⊤, nous avons la formule $\forall x, \top(x)$.
- A tout couple de types de concepts (t_1, t_2) de T_C tel que $t_1 \leq t_2$ on associe la formule :

$\forall\ x, t_1(x) \rightarrow t_2(x)$

- A tout couple de types de relations (t_1, t_2) de T_{Rp} / $t_1 \leq t_2$ correspond la formule :

  $\forall\ x_1 \ldots x_p,\ t_1(x_1, \ldots, x_p) \rightarrow t_2(x_1, \ldots, x_p)$, où p est l'arité de t_1 et t_2.

- A tout $\delta(t_r) = (t_1, \ldots, t_p)$ d'un type de relation tr de T_{Rp} , on associe la formule :

  $\forall\ x_1 \ldots x_p,\ t_r(x_1, \ldots, x_p) \rightarrow t_1(x_1) \wedge \ldots \wedge t_p(x_p)$,

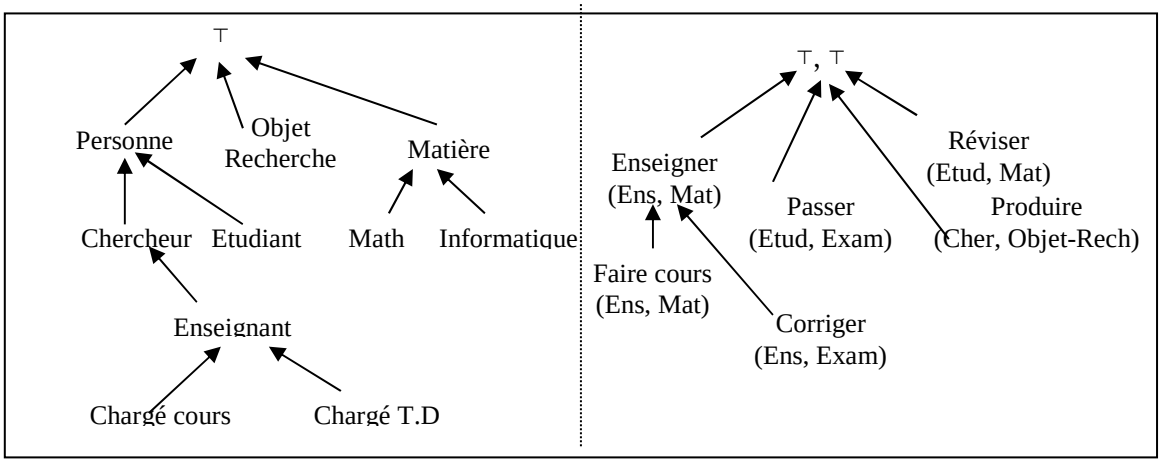


Figure II.5 : Hiérarchies types de concepts, et types de relations binaires

Le support défini en Figure II.5, sur les concepts et les relations du domaine, se traduit à l'ensemble des formules Φ (S) suivant :

$\forall$ x, ⊤(x)
$\forall$ x, Personne(x)→ ⊤(x)
…
$\forall$ x, Chercheur(x)→ ⊤(x)
…
$\forall$ x, Chargé-cours(x)→ Enseignant(x)
$\forall$ x, Enseignant(x)→ Chercheur(x)
…
$\forall$ x, $\forall$ y, Enseigner(x, y)→ T_2 (x, y)
$\forall$ x, $\forall$ y, Corriger(x, y)→ Enseigner (x, y)
…
$\forall$ x, $\forall$ y, Corriger(x, y)→ Enseignant(x) ∧ Examen (y)
…
Matière(Informatique)
…

Pour tout graphe conceptuel G sur un support S, l'application Φ associe une formule de la logique du premier ordre, notée Φ (G), elle est construite comme suit [Gri,2008]

- Pour tout nœud concept c, on associe l'atome $t_c(id_c)$, tel que t_c est le prédicat associé au type de c et id_c le terme (variable ou constante) associé à c.
- Pout tout nœud relation r, on associe l'atome $t_r(id_1, \ldots, id_p)$, où t_r est le prédicat associé au type de r, p l'arité de ce prédicat, et chaque id_i est le terme associé au i ème voisin de r dans G.

La formule Φ (G) est obtenue par la fermeture existentielle de la conjonction de ces atomes. Pour le graphe conceptuel de la Figure II.4, nous obtenons :

Φ(G)= x(manager(m)^employé(e) ^date(d)^ recruter(x)^ agent(x, m)^ objet(x, e) ^ durée(x, d))

II.3.2.2 Raisonnements dans les graphes conceptuels

Plusieurs opérations sur les graphes conceptuelles sont vues comme étant des types de raisonnements.

a. La simplification

Lorsque deux ou plusieurs relations conceptuelles d'un GC sont identiques, c'est-à-dire ayant le même type, même arité, reliant les mêmes types de concepts, alors on procède à une simplification de GC. La simplification consiste à supprimer les informations redondantes.

b. La Restriction

Ce type d'opération concerne la restriction de concepts et la restriction de relations, dans les deux cas, la restriction est le raisonnement qui spécialise des types de concepts/relations génériques par des types de concepts/relations plus spécialisés.

c. La Jointure

Etant donné un Graphe conceptuel « C », ayant $C_1,C_2,\ldots,C_n$ des concepts identiques aux concepts $D_1,D_2,..D_n$ d'un autre graphe conceptuel « D ». L'opération de jointure des deux GCs « C » et « D » produit un graphe conceptuel « R », qui est obtenu en enlevant les concepts $D_1,D_2,..D_n$ et en reliant à $C_1,C_2,\ldots,C_n$ les arcs qui été reliés aux nœuds $D_1,D_2,..D_n$ dans le graphe « D ».

d. La Projection

Cette opération est un morphisme de graphes, elle est à la base du mécanisme de recherche dans les graphes conceptuels, elle permet le calcul de la relation de spécialisation. La recherche d'une projection d'un graphe « H » dans un graphe « G » est vue comme la recherche de « l'inclusion » de l'information représentée par « H » dans « G ». Si c'est le cas, on dit que « G » est une spécialisation de « H » ou « H » subsume « G ».

Le théorème suivant est une preuve « *Etant donnés « G » et « H » deux graphes conceptuels, il existe une projection de « H » dans « G » si et seulement si* $G \leq H$.

Une projection d'un graphe conceptuel $H=(R_H, C_H, U_H, étiq_H)$ dans un graphe conceptuel $G = (R_G, C_G, U_G, étiq_G)$ est un couple d'applications $\Pi =(f, g)$ telle que

$f : R_H \rightarrow R_G$, et $g : C_H \rightarrow C_G$ [Mug,1996]

1) Restreindre les étiquettes des nœuds, pour sommet « r » de R_H, $étiqG(f(r)) \leq étiqH(r)$

Et pour tout sommet « c » de C_H, étiqG(g(c)) ≤ étiqH(c)

2) Conserve les arcs et la numérotation des arcs.

Pour tout arc « rc » de U_H, f(r)g(c) est un arc de U_G , si c = H_i(r), alors g(c) = G_i(f(r)).

Π(H) : est le sous-graphe de G image de H, comme le montre la Figure II.6.

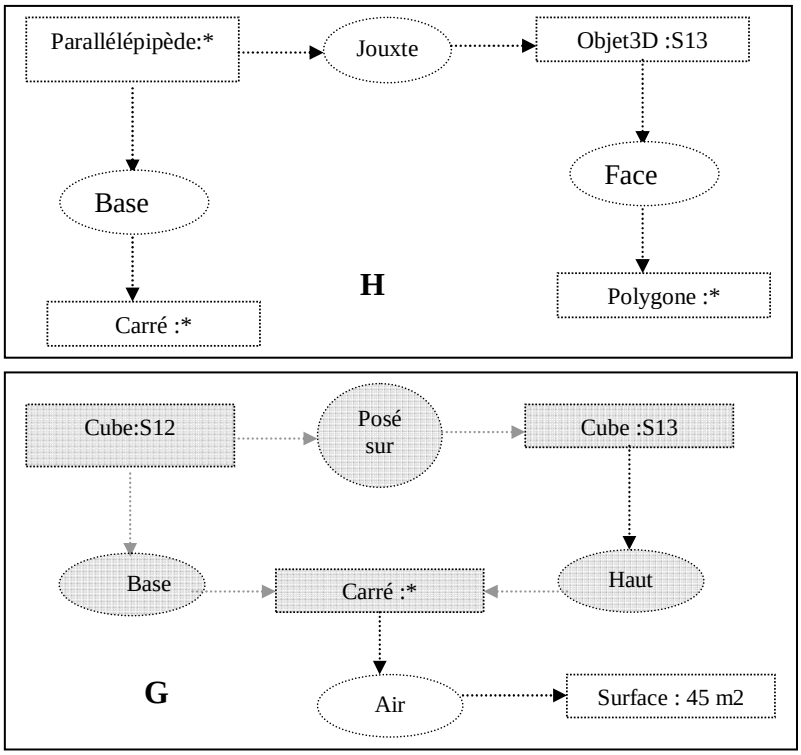


Figure II.6 : Projection du GC « H » dans le GC « G »

Π(H) : le sous graphe du graphe « G », image du graphe « H », qui vérifie une projection de « H » dans « G » est la partie grisée du graphe. L'opération de projection dans le domaine de la recherche d'informations, est un raisonnement très puissant, il permet de représenter par un graphe conceptuel « R » une requête utilisateur, à la quelle on cherchera une projection dans graphe conceptuel « C » qui modélise les connaissances du domaine sous forme d'annotations sémantiques.

Les graphes conceptuel de la Figure II.6, illustre ce raisonnement, le graphe « G » peut être une réponse pertinente à la requête utilisateur : « *Les parallélépipèdes ayant une base carrée, qui jouxtent des objets 3D, dont la face est polygonale* ».

II.3.3 Les représentations logiques

Les logiques mathématiques constituent le modèle de représentation des connaissances les plus célèbres dans la communauté de gestion des connaissances. Cette

popularité est soutenue par leurs solides fondements mathématiques et les méthodes de preuve efficaces [Hat,1997].

L'objectif de telles représentations, est de produire des formules logiques (expressions structurées) suivant une syntaxe bien définie, telle que ces expressions n'admettent pas plusieurs interprétations vis-à-vis de leurs réelles intensions. L'intérêt majeur de ces représentations est la disponibilité de règles d'inférences fondées sur ces formules solides comme Modus Ponens, Modus Tolens, la règle de résolution, la réfutation, et la spécialisation.

De manière générale, une représentation logique est définie par :

- Une syntaxe sur un langage L défini par un alphabet α, qui définit les règles d'écriture des formules bien formées.
- Une sémantique qui interprète des valeurs de vérité et associe des modèle aux formules.
- Axiomes, qui définissent un sous ensemble de L de formules valides, ou tautologies.
- Règles d'inférence, qui définissent le procédé pour les formules qui sont les conséquences logiques d'autres formules, c'est le raisonnement que supporte le modèle en question.

a. Logique des prédicats du 1er ordre

De nos jour, la logique des prédicats du premier ordre et malgré la limitation à pouvoir représenter des connaissances complexes, incomplètes, approximatives, et/ou incertaine [Kyu,2004], est une technique qui connait un large succès d'utilisation car :

- C'est une méthode formelle pour le raisonnement.
- Aptitude à pouvoir exprimer un grand nombre de concepts de domaine, dans une symbolique forme qui s'approche du sens de ces concepts.
- Ces représentations logiques s'apprêtent aux traitements automatiques, pour produire des faits par l'application de mécanismes de raisonnements corrects.

Cependant, ce formalisme pose le problème de la décidabilité des raisonnements, car en général la logique des prédicats du premier ordre est complète, mais semi-décidable, voire indécidable. L'autre point de faiblesse, est l'étendue restreinte, par exemple il n'est pas possible de représenter des connaissances relatives aux langages naturels, et c'est dans cette perspective que plusieurs autre pseudo-logiques telle que les logiques modales, la logique temporelle, la logique on-monotone, la logique floue, a logique des défauts et la logique probabiliste ont été proposées.

b. Logique non monotone :

La logique non monotone émet la supposition suivante, à partir de connaissances de certains faits, si une proposition ne peut pas être prouvée, alors il est raisonnable de la

considérer comme étant fausse. Ce qui signifie que si P(a) n'est pas prouvable alors ¬(P(a)) est vraie, en complétant la base de connaissances par la négation des faits non dérivables on preuve la complétude.

c. Logique floue :

Proposée par Zadeh en 1975, la logique floue consistait en l'issue pour remédier aux points faibles de la logique des prédicats du premier ordre. En effet souvent les expressions des experts humains sont imprécises, la logique floue peut fournit des interprétations pour ce genre de connaissances [Kyu,2004]. Dans ce formalisme, la qualité du fait est représentée par un nombre entre [0, 1], donc les propositions sont menée d'un degré de vérité ou de fausseté pour être par la suite raisonnées par les mécanismes appropriés.

d. Logique Temporelle :

La logique temporelle est une logique intuitionniste, elle traite le temps de manière linaire, c'est un modèle qui peut être utilisé dans un environnement distribué, pour raisonner sur des connaissances qui évoluent dans le temps, de façon discrète (points) ou continue (intervalles). Des opérateurs temporaux sont introduits pour exprimer des grandeurs de temps, parmi les plus connus nous avons "Always", "Done", "Future", "Until", et "Eventual".

II.3.3.1 Raisonnements logiques

Le raisonnement sur des représentations logiques, consistera en un enchainement d'opérations sur les structures symboliques, tout en préservant la cohérence et l'exactitude des déductions

-Raisonnement déductif : Valide et rigoureux, les déductions de connaissances valides se fait par l'application de connaissances générales aux cas particuliers. C'est donc la vérification de théories qui s'appuie sur des schémas comme le "Modus Ponens" [Hat,1997]

-Raisonnement inductif : Infère des lois et développe des théories, à partir de l'abstraction de faits expérimentaux. C'est donc le passage au cas général ou le développement d'une théorie, ce raisonnement est dit plausible.

-Raisonnement abductif : Plausible, il cherche à attacher des causes aux prémisses, émet des hypothèses à partir de l'observation de faits particuliers.

- Raisonnement approximatif : A partir d'hypothèses approximatives, il est nécessaire de disposer de mécanismes de raisonnements approximatifs, efficaces et capables de prendre en compte les imperfections de ce type de connaissances, pour ce faire, il s'agit de :

- Définir une représentation de l'incertitude et de l'imprécision.
- Etendre le schéma du raisonnement, pour tenir comptes de nouveaux aspects.

-Raisonnement analogique : chez l'humain l'analogie est un raisonnement cognitif très naturel dans divers domaine tel que la compréhension du langage naturel, ou l'analyse de scènes visuelles. Le raisonnement par analogie consiste à la mise en correspondance de deux situations dans un univers donné, et déduire la solution d'une nouvelles situation en fonction des solutions connues de situations déjà rencontrées.

Les principales étapes du raisonnement analogique (à partir de cas) sont :

- *La remémoration* : sélection d'une situation jugée similaire à la nouvelle situation.
- *L'adaptation* : résolution de la nouvelle situation, en s'appuyant sur la situation similaire sélectionnée.
- *La mémorisation* : apprentissage, validation de la nouvelle situation, et son éventuel mémorisation.

II.3.3.2 Distribution du raisonnement

L'intelligence artificielle distribuée IAD, présente un intérêt capital du fait du caractère distribué de certains problème, cette distribution peut être géographique, spatiale, fonctionnelle (tâches à effectuer) ou méthodologique (solutions partielles) [Hat,1997].

- L'intérêt d'u système distribué réside dans la réduction de la complexité par la modularité et la sureté du fonctionnement.

II.3.3.3 Explication et planification du raisonnement

L'ensemble d'informations qui tracent le raisonnement suivi pour parvenir au résultat. Expliquer est un processus important à plusieurs titres :

- Mise au point de systèmes par détections d'anomalies.
- Permet de s'approprier du raisonnement, et se faire familier avec les utilisateurs.

La planification du raisonnement a pour but de raisonner sur des actions et des plans. Ce concept est à l'origine de l'émergence des notions de causalité, temps, perception, multi-agent etc.

II.3.4 Les logiques de description

A leurs origines, les logiques de description « LDs », devaient fournir une sémantique formelle aux réseaux sémantiques et aux frames de Minsky, afin que ces modèles soient pourvus d'outils permettant de raisonner. C'est une famille de formalismes de représentation

de connaissances (KR), dans un domaine d'application basées sur le langage KL-One de *Brachman*, qui lui-même est issus des réseaux sémantiques et des frames. Les principaux constructeurs des logiques de descriptions sont :

Concept: expression utilisée pour dénoter des ensembles d'individus.

Rôle: Dénote des relations binaires entre individus.

Dans la base de connaissance, on peut distinguer clairement des connaissances intensionnelles qui sont des connaissances décrivant la terminologie du domaine, et des connaissances extensionnelles qui spécifient un problème donné [Baa,2003] [Nar,2003].

Suivant cette analogie, une base de connaissances présentée par les logiques de description, est formé du couple (T-Box, A-Box), telle que : « Figure II.7 »et « Figure II.8 »

- TBox : c'est la terminologie, ou le vocabulaire qui décrit les concepts et les rôles du domaine d'application, cette composante spécifie les connaissances structurelles, ce sont les connaissances intensionnelles.

En plus des définitions atomiques des concepts et des rôles les DLs permettent de construire des définitions plus complexes en utilisant un ensemble de constructeurs de concepts (négation, conjonction, disjonction, restriction etc.). Par exemple on définit le concept "Femme" comme étant une "personne" de sexe "féminin", par la déclaration: Femme ≡ Personne ∩ Femelle.

- ABox. : dite aussi assertionelle, contient des individus qui sont des assertions sur les concepts et sur les relations, avec certaines précautions pour les relations qui ne sont pas primitives, ces assertions sont les connaissances factuelles ou extensionnelles du domaine.

Femelle ∩ Personne(Amina) est une formule LDs.

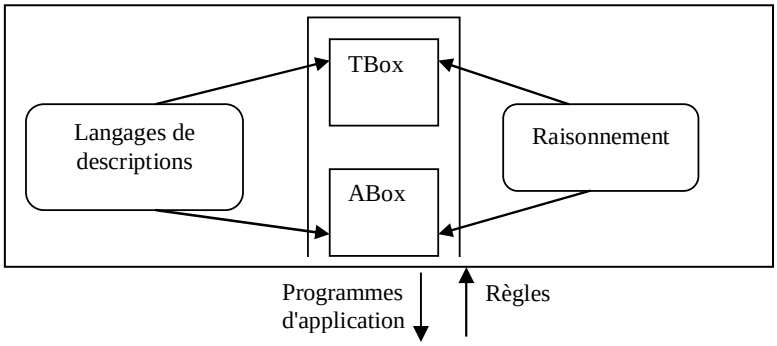


Figure II.7:Architecture des bases de connaissances en LDs [Baa,2003]

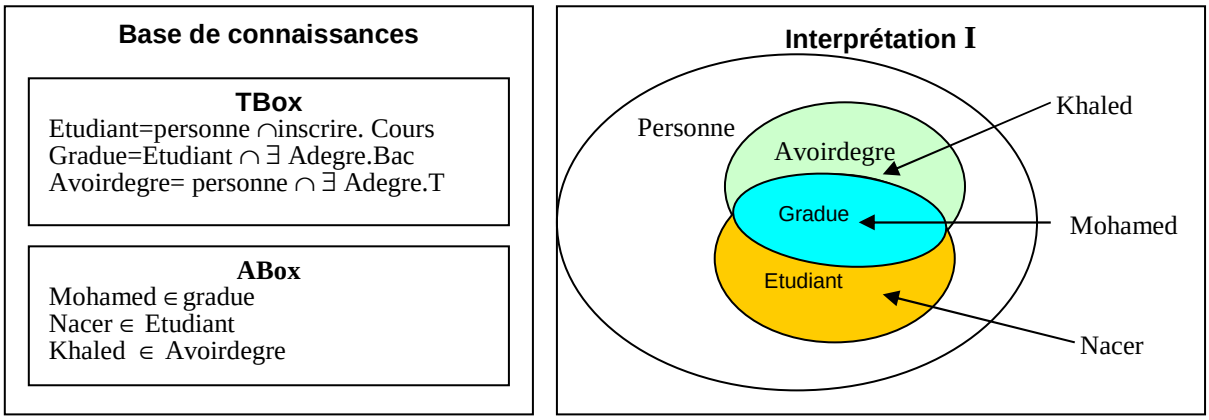


Figure II.8: Base de connaissances et interprétations en LDs

II.3.5 Modèle basés ressources terminologiques

Les ressources terminologiques se situent à l'intersection des domaines de la terminologie et de l'ingénierie des connaissances en intelligence artificielle.

Plusieurs sortes de ressources constituent de ce fait une articulation solide pour développer des modèles de représentation des connaissances d'un domaine, cela inclut les simples indexes, les taxonomies et les thésaurus.

II.3.5.1 Vocabulaire contrôlé

Souvent, un vocabulaire contrôlé est associé aux documents techniques et/ou spécialisés, il comporte des sujets nommés qui décrivent le contenu de ces ressources et servent de références pour la recherche d'information et la classification des entités du domaine.

La distinction entre "terme" et "concept" est formulée en déclarant que le premier est le nom de concept, et qu'un même concept peut avoir plusieurs noms, comme aussi, un terme peut désigner plusieurs concepts (sujets) [Gar,2004].

II.3.5.2 Taxonomie

Les systèmes de représentation des connaissances et de raisonnement en intelligence artificielle ont besoin de représenter l'existant de manière formelle. Une taxonomie est toute structure abstraite, d'une collection formelle de termes se rapportant à un vocabulaire contrôlé. Elle classifie ses entités suivant un ensemble de propriétés choisies pour cette hiérarchisation.

Cette hiérarchie se modélise par une structure arborescente, qui traduit une relation de subsomption « is-a » entre les concepts de la taxonomie, les propriétés algébriques de cette relation permettent d'enrichir la sémantique des connaissances modélisées [Ama,2007]. Les parties qui composent la taxonomie sont :

- *Relations hiérarchiques:* relatent les concepts du plus général qu'est la racine aux plus spécifiques, c'est-à-dire les feuilles, c'est une relation transitive.
- *Niveaux:* Une hiérarchie possède plusieurs niveaux, le plus élevé, est aussi le plus abstrait, ainsi les éléments d'un même niveau auront approximativement le même degré d'abstraction.
- *Racine:* C'est le sommet de la structure, le domaine ou la source de la structure.
- *Nœuds* : représente les concepts de la structure, la plupart sont des nœuds père-fils
- *Chemin:* Est une séquence de nœuds traversés pour atteindre un nœud particulier.

II.3.5.3 Thésaurus

Le terme thésaurus a été utilisé tout comme les taxonomies, pour décrire toute sorte de structure de classification de termes, il représente une extension de celle-ci. Un thésaurus dote une taxonomie d'aptitude à pouvoir décrire le domaine de connaissances non seulement par l'hiérarchisation de ses termes, mais aussi en permettant la construction de nouvelles déclarations concernant les termes, et en fournissant des relations de description de ces termes: « Figure II.9 ».

- *Synonymie :* un terme X est le synonyme d'un terme Y.
- *Homonymie:* un terme X a la même forme orale ou écrite qu'un terme Y alors qu'ils ont des sens différents.
- *Associative*: un terme X est associé à un terme Y s'il y a une sorte de relation non sémantiquement spécifiée entre les deux.[Gar,2004].

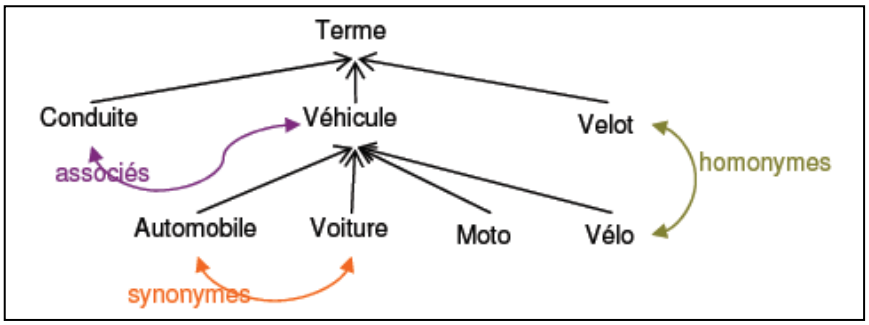


Figure II.9 : Les relations dans un thésaurus

En résumé, un thésaurus tel que le dictionnaire électronique « WordNet » que nous avons utilisés lors du processus d'expansion de requête sont très utilisé dans les applications de l'intelligence artificielle, servant à résoudre des problèmes pratiques de désambigüisation, classification et de recherche de documents.

II.3.6 Les ontologies

En philosophie, le terme ontologie revoie étymologiquement à la « théorie de l'existence ». D'après [Gri,2008] et [Mon,2008], l'ontologie est une branche de la métaphysique qui traite de la nature des êtres, en tant qu'être, elle capture et structure les connaissances de façon à permettre leur partage et leur réutilisation. En intelligence artificielle, l'ontologie signifie l'artefact informatique qui permet de représenter et de manipuler les connaissances d'un domaine en donnant à ses composants une sémantique tout en précisant leurs relations.

Apparue en informatique dans les courants des années 1990, les chercheurs ont adopté ce terme dans leur propre langage, et pour eux une ontologie est la spécification explicite d'une conceptualisation partagée qui présente une vue du monde réel dans un domaine spécifique.

Dans cet esprit, plusieurs définitions du concept ontologie existent, souvent dépendantes du contexte d'utilisation, parmi ces définitions nous citons la définition fréquemment admise qui est celle de Gruber 1993, «*Une ontologie est une spécification explicite d'une conceptualisation* » [Gru,1993]. « Explicite »: définir les concepts, les propriétés, les relations, et les axiomes de l'ontologie ainsi que les conditions de leur utilisation de manière explicite.

«Conceptualisation » : concevoir un modèle d'abstraction d'une réalité, par l'identification des concepts clés de cette réalité.

Nous *complétons* cette définition par celle d'Uschold [Usc,1996] « *Une ontologie peut prendre différentes formes, mais elle inclura nécessairement un vocabulaire de termes et une spécification de leur signification. Cette dernière inclut des définitions et une indication de la façon dont les concepts sont reliés entre eux, les liens imposant collectivement une structure sur le domaine et contraignant les interprétations possibles des termes* ».

Dans une récente publication, [Gru,2009], Gruber propose une définition plus détaillée : « *Une ontologie est un ensemble de primitives de représentation qui permettent de modéliser un domaine de connaissances ou un domaine de discours. Les primitives de représentation sont typiquement des classes (ou des ensembles), des attributs (ou des propriétés), et des relations (ou relations parmi les membres des classes). La définition des primitives de représentation inclut des informations à propos de leur sens et des contraintes sur la consistance logique de leur application* ».

II.3.6.1 Buts d'utilisation des l'ontologie

Plusieurs aspects guident l'utilisation des ontologies dans les application IA :

- La communication : permettre de communiquer sans ambiguïté entre les humains et/ou les organisations, par opposition au langage naturel, les termes définis dans l'ontologie possède chacun sa propre sémantique.
- L'interopérabilité: l'ontologie sert de modèle intermédiaire pour la traduction entre les modélisations de différentes collections d'objets, c'est un format d'échange.
- L'ingénierie des systèmes: l'ontologie peut assister le processus de construction de spécification de système, modéliser les documents du processus et évite l'ambiguïté dans la spécification.

II.3.6.2 Eléments de base constituant l'ontologie

Les schémas de représentation des connaissances, informels ou formels reposent sur le triangle sémiotique, « Figure II.10 » qui montre les relations entre des idées/abstractions et leurs concrètes expressions dans le monde réel [Bar,2009]. Les éléments de base de l'ontologie sont :

a. **Objet :** c'est une représentation de l'objet réel, du point de vue ontologie c'est l'individu.

b. Propriété : une propriété d'un ou d'une collection d'objet est une qualité, une caractéristique de l'objet, son existence est inhérente à l'objet.

c. Prédicat: est une assertion à propos de l'abstraction du domaine, acceptée comme vraie, et exprime des propriétés et des contraintes.

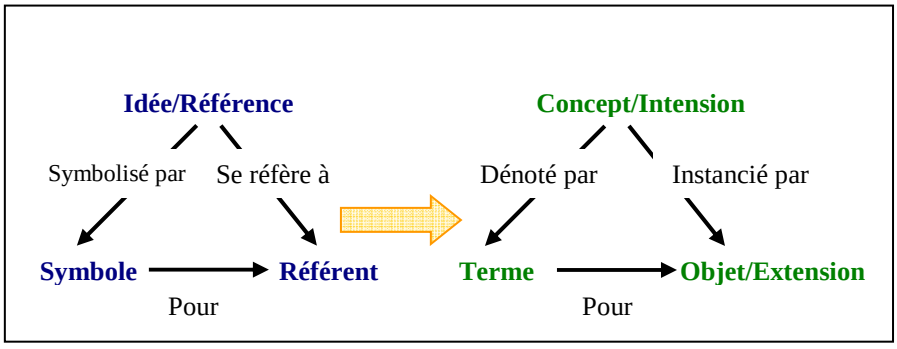


Figure II.10 : Le triangle sémiotique et l'interprétation ontologique [Bar,2009]

d. Classe: et une abstraction d'une collection d'objets ayant des propriétés communes. Du point de vue des ontologies c'est une définition d'un ensemble d'objets avec leurs attributs.

e. Terme: est un nom simple, une expression, une formule ou un symbole qui désigne l'objet.

f. Intension : l'intension est l'ensemble de propriétés qui caractérise un concept.

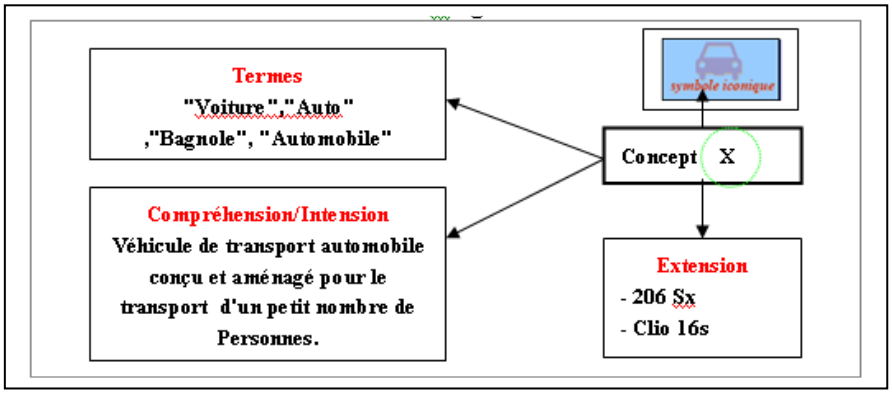


Figure II.11: L'élément Concept

g. Extension : Dénote un ensemble d'objets qui correspondent à un concept.

h. Concept: Un concept est l'unité de connaissances, il est caractérisé par la notion d'intentionnalité, le concept représente un objet qui est exprimé par un terme, « Figure II.11 ». Les concepts sont organisés en taxonomie par des relations de subsomption, l'extension d'un concept est l'ensemble de ses instances, alors que son intension est l'ensemble des attributs, propriétés et contraintes.

i. Relation/Fonction : Les relations sont l'ensemble des associations et interactions entre les concepts qui permettent de construire des représentations complexes de la connaissance du domaine. Ces relations permettant de structurer hiérarchiquement les concepts du domaine incluent les associations de : généralisation - spécialisation (sous-classe), partie de (agrégation), etc. Les fonctions constituent des cas particuliers des relations. « Figure II.12 ».

L'extension d'une relation est l'ensemble d'instances, ce sont donc des réalisations effectives de cette relation entre des êtres. L'intension d'une relation ou sa compréhension est l'ensemble des attributs, propriétés et contraintes communes aux réalisations.

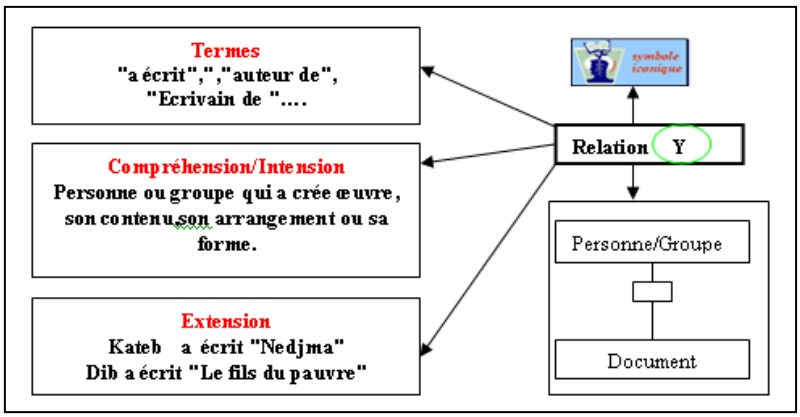


Figure II.12: L'élément Relation

II.3.6.3 Types d'ontologies

Dans la littérature, plusieurs types de classification des ontologies existent [Ama,2007] [Stu,1998] [Asu,1999]

a. Ontologie générique (haut niveau) : Abstraite**,** elle est utilisable dans différents domaines, les concepts qui y sont définis sont génériques et décrivent des conceptualisations très générales.

b. Ontologie de domaine : L'ontologie de domaine exprime une conceptualisation d'un domaine spécifique, elle décrit les entités du domaine, leurs propriétés et les liens associés.
L'intérêt, de ce type d'ontologie est qu'elles sont réutilisables pour différentes applications dans le même domaine. Généralement les concepts d'une ontologie de domaine sont des spécialisations des concepts d'une ontologie de haut niveau.

c. Ontologie de tâche: Modélise une activité générique par la description d'un ensemble de concepts et le rôle de chacun dans le raisonnement, qui conduit à la résolution du problème.

d. Ontologie d'application: Elle est restreinte et contient les connaissances spécifiques à une application (exécution de tâches). Ce type d'ontologie peut être vu comme une spécialisation d'une ontologie de domaine par rapport à une ontologie de tâche permettant ainsi de modéliser une activité spécifique dans un domaine donné.

e. Ontologie de représentation (méta-ontologie)**:** n'est pas spécifique à un domaine particulier, elle est utilisée pour formaliser un domaine de connaissances. L'exemple de l'ontologie de frame qui définit les primitives pour exprimer de la connaissance dans un environnement implémentant les langages de Frame.

N.Guarino [Gua,1998], propose une autre classification, "Figure II.13" où les ontologies de haut niveau décrivent les concepts généraux communs à tous les domaines, (temps, espace, objet, évènement,…), elles ressemblent donc aux ontologies génériques.

Les ontologies de domaine et les ontologies de tâche spécialisent les termes des ontologies de haut niveau, alors que l'ontologie d'application décrit les concepts qui dépendent simultanément des ontologies de domaine et des ontologies de tâche.

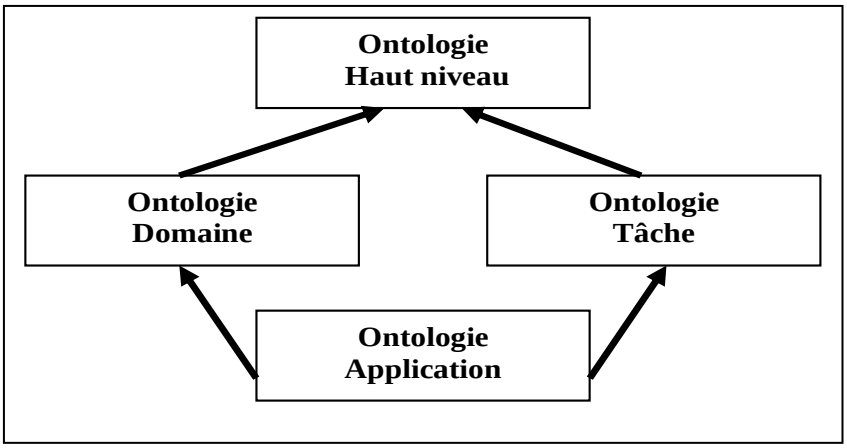


Figure II.13 : Type d'ontologies selon leur dépendance [Gua,1998]

II.3.6.4 Principes méthodologiques de construction d'ontologies

a. Choix de termes : L'intention d'une ontologie étant de définir le sens de concept qui peut être interposé entre plusieurs termes [Mon,2008]. Le sens de terme doit être défini de manière claire et objective.

b. Exhaustivité : le degré de spécificité détermine avec précision la définition du concept, par conséquent mieux un concept est défini, moins il sera ambigu, pour ce faire [Asu,1999], préconise d'exprimer les définitions par des conditions nécessaires et suffisantes au lieu de l'exprimer seulement par une condition nécessaire, ou seulement par une condition suffisante.

c. Cohérence : Une ontologie doit être cohérente afin de formuler des inférences consistantes avec les définitions [Asu,1999]. La cohérence dépend des termes qui sont associés, et de la nature de leur association.

d. Extensibilité : L'ajout de nouveaux termes, ne doit pas nécessiter des révisions ou remettre en cause les définitions existantes.

e. Principe de distinction ontologique : Les classes de l'ontologie doivent être bien séparées (disjointes), pour faciliter leur compréhension.

f. Multiplicité des hiérarchies/Héritage multiple : Par multiplicité on vise la désignation de l'emplacement d'un terme dans multiples hiérarchies, et nous devons distinguer le critère de multiplicité, de l'ambiguïté sémantique.

L'héritage multiple, signifie qu'une sous classe pourra avoir plus d'une classe parent, offrant plus de flexibilité et de richesse d'expressions dans le modèle de l'ontologie, mais aussi cela induit plus de complexité dans le modèle [Asu,1999].

h. Modularité : par ce critère, on prévoit réduire le couplage entre les différents modules.

II.3.6.5 Méthodes de construction des ontologies

Une méthodologie comprend l'ensemble de procédures, de techniques de processus d'activités, et de directives qui assistent le développement de l'ontologie durant son cycle de vie et suivant une approche donnée (bottom-up, top -down, et middle-out) [Cas,2011].

Les ontologies, des composants logiciels, leur développement s'appuie sur les mêmes principes du *génie logiciel,* en particulier une ontologie a son propre cycle de vie, qui désigne dans son processus de développement une succession d'états (phases) à travers lesquels s'effectue le processus de transformation de données brutes en une ontologie opérationnelle.

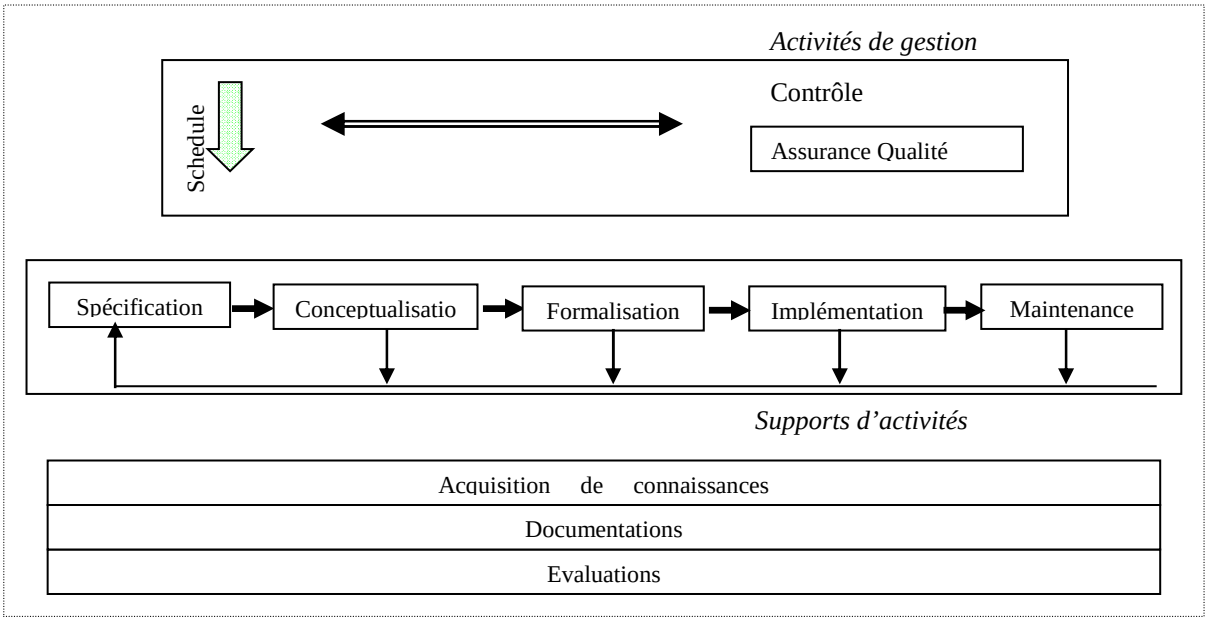


Figure II.14 : Phase de construction d'une ontologie

Ce processus s'accomplit par plusieurs raffinements progressifs, qui impliquent la révision de certains choix acceptés pour une phase donnée, ces phases sont : *Spécification, Conceptualisation, Formalisation, Intégration Implémentation et Maintenance*. "Figure II.14"

Parmi les méthodologies pertinentes utilisées actuellement, nous citons :

- La méthodologie initiée par Lenat and Guha en 1990, pour modéliser l'ontologie « Cyc »

1- Extraction manuelle et semi automatique des connaissances du sens commun.

2-Gestion des connaissances extraites auparavant et stockées dans « Cyc ».

- La méthodologie d'Uschold, King et Grüninger, distingue plusieurs étapes :

1- Identification du rôle et l'intention d'usage.

2-Détermination du niveau de formalité.

3-Construction de l'ontologie suivant plusieurs étapes.

4-Evaluation, révision, et vérification des critères de clarté, de consistance et de cohérence.

La troisième méthodologie que l'on décrit est « CommonKADS », qui est destinée à construire des modèles de connaissances. Elle inclut des activités qui assistent la construction dans la définition des différents rôles :

1-L'identification des connaissances, en sources d'informations, scénarios, etc. et identification des composants de modèles pour leur réutilisation.

2-Spécification des connaissances**,** qui inclut le choix de modèle de tâches, la décomposition, la construction d'une conceptualisation initiale du domaine, compléter la conceptualisation.

3-Raffinement de connaissances, validation du modèle de connaissances, prototypes de raisonnements, compéter la base de connaissances.

Nous terminons ce background des méthodologies par la célèbre METHONTOLOGY, décrite par Fernandez-Lopez et al. 1997, 1999 et 2003. Cette méthodologie décrit à la fois les phases du processus de conceptualisation d'une ontologie et le aussi durant le développement de son cycle de vie.

1-Spécification : est la phase durant laquelle sont élucidées les buts, l étendue de l'ontologie et les utilisateurs futurs.

2-Conceptualisation : correspond aux processus d'organisation des connaissances acquises, cela se traduit par, la définition des termes, leurs classifications en taxonomies de concepts associés, définitions des relations binaires entre le concepts, la construction du dictionnaire de concepts (attributs), définitions de détails de concepts comme les cardinalités, les types de rôles, et certaines propriétés de relations comme les relations inverses. Durant ce processus on peut utiliser un ensemble de représentations intermédiaires semi-formelles (des tables et des graphes).

3-Formalisation, Implémentation et Evaluations : correspondent aux choix d'outils pour la formalisation, coder l'ontologie dans un langage d'ontologie formel, l'implémentation et les prototypes d'évaluation par rapport aux besoins spécifiés.

Les méthodologies Enterprise, Tove et Methontology sont les plus représentatives pour construire des ontologies formelles. La méthode TOVE s'intéresse principalement à la construction d'ontologies représentées par la logique du premier ordre.

Les méthodes ENTERPRISE et METHONTOLOGY se distinguent par le fait qu'elles commencent par identifier le but de l'ontologie à créer ainsi que les entités cognitives du domaine de connaissances. Une fois les connaissances (*concepts et relations*) acquises,

ENTERPRISE propose de passer directement de l'acquisition de connaissances à la codification de l'ontologie en utilisant un langage formel.

METHONTOLOGY suggère plutôt d'exprimer l'idée sous forme d'un ensemble représentations intermédiaires, semis-formelles, à travers une étape de conceptualisation avant de passer à la codification de l'ontologie.

II.4 Le Web Sémantique

II.4.1 Historique

Le web tel que connu actuellement a été créé par Tim Berners-Lee au début des années 1990, et avait pour objectif de permettre à des agents humains ou logiciels d'échanger leurs savoirs. Depuis, il a fortement évolué par le fait que la quantité gigantesque des documents de nature hétérogène, structurés, et non structurés ne cessent de s'accroitre remettant en cause les démarches de la recherche d'informations.

Le HTML (Hyper Text Markup Language) qui dérive du langage SGML (Standard Generalized Markup Language) apparu dans les années 1970 constituait l'un des premiers langages utilisés pour structurer l'information sur le web, mais avec cette diversité, on a vite pris conscience de l'incapacité du HTML à satisfaire les besoins utilisateurs par seulement la mise en forme de l'information, ainsi le besoin d'explorer d'autres démarches plus efficaces est mis en évidence. Dans cette perspective, le web actuel a atteint ses limites, une réorganisation non plus structurelle, mais se basant sur le contenu documentaire, semble être pressante.

L'idée du web sémantique initiée par le consortium W3C (World Wide Web Consortium) fût adoptée, l'objectif du consortium est de structurer les informations disponibles sur le web, qui doit être perçu comme un vaste espace d'échange de ressources, entre humains et machines. Tim Berners Lee, définit le consortium comme suit : « *Nous avons crée un environnement neutre, capable de servir les intérêts de tous, depuis l'individu jusqu'aux plus grandes entreprises et aux états. La communauté industrielle en particulier a compris qu'il est de son intérêt de disposer d'un web stable et évolutif, fondé sur un accord commun* ».

Le rôle du consortium consiste donc à inventer et à favoriser l'expansion des langages et protocoles universels, afin de permettre une évolution homogène, décentralisée et standardisée.

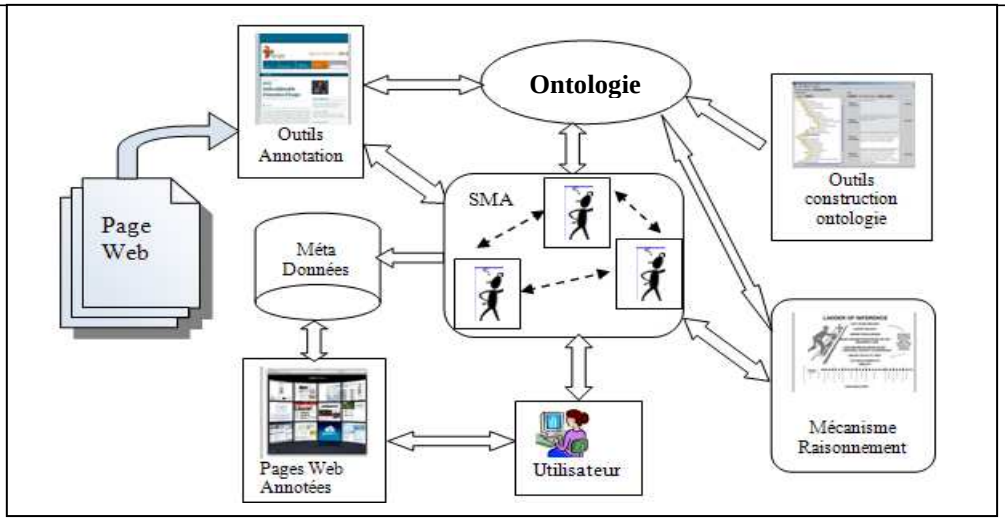


Figure II.15 Vision du web sémantique

II.4.2 Définitions et standards du web sémantique

En Mai 2001, Tim Berners-Lee, James Hendler et Ora Lassila présentèrent le Web sémantique comme étant [Ber,2001] « The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.» Figure II.15. Le but étant de disposer de nouveaux langages, tels que le XML, RDF(S) et OWL et de nouveaux outils pour permettre de comprendre et manipuler les contenus sémantiques des documents afin de réaliser un meilleur partage et mieux fixer leurs interprétations [Rai,2008].

L'utilisation de ces langages pour modéliser les connaissances d'un domaine passe par le langage XML (eXtensible Markup Language), qui offre une libre syntaxe pour organiser les documents, RDF (*Resource Description Framework*), [Kly,2004] qui structure par des *triplets* les connaissances factuelles du domaine, et par le langage RDFS (Resource Description Framework Scheme) [Bri,2004] pour la définition des connaissances structurelles du domaine. L'usage de RDF et RDFS est souvent notée RDF(S).

Le langage OWL - pseudo acronyme de *Web Ontology Language* - est la génération suivante de RDF(S) et dispose d'une très grande richesse expressive avec ses trois versions OWL_LT, OWL_DL et enfin OWL FULL. Il s'agit d'un langage formel pour décrire des concepts, des faits et des relations dans une ontologie et les diffuser sur le Web [Gui,2004] [Dea,2004]. OWL repose sur la syntaxe XML/RDF, étend le vocabulaire de RDFS et possède une sémantique logique issue des logiques de descriptions [Baa,2003]. Ce langage offre ainsi l'avantage d'être formel et de pouvoir disposer de raisonnements issus de ces logiques et applicables sur les connaissances qu'il modélise. L'architecture des langages du web

sémantique est une pyramide proposée par Tim Berners-Lee, elle représente des connaissances sur le web en satisfaisant les critères de standardisation, d'interopérabilité et de flexibilité.

Cette architecture en couches « Figure II.16 » permet une approche graduelle dans les processus de standardisation et d'acceptation par les utilisateurs. Un langage de la couche haute doit être une extension du langage de la couche au-dessous. Aujourd'hui seules les couches basses sont relativement stabilisées. Les fonctions principales de chaque couche dans l'architecture du web sémantique sont :

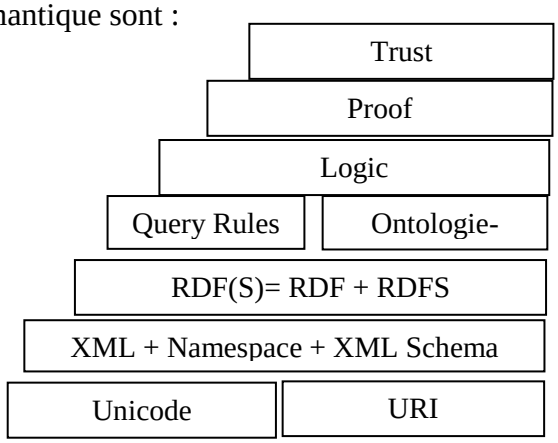


Figure II.16 : Pyramide des langages du web sémantique

- XML est utilisé comme couche de base syntaxique du web sémantique. Le langage XML est considéré comme un standard pour le transport de données sur le web.
- La couche RDF représente les métadonnées pour les ressources web.
- La couche « OWL-Ontologie », fondée sur une formalisation commune, spécifie la sémantique de métadonnées fournies dans le web sémantique.
- La couche « Logique » s'appuie sur des règles d'inférence qui permettent le raisonnement intelligent exécuté par des agents logiciels.

L'infrastructure du web sémantique s'appuie sur un certain niveau de consensus pour faciliter le partage de la connaissance et sa réutilisation, elle doit d'autre part contribuer à assurer le plus automatiquement possible l'interopérabilité et la mise en œuvre de calculs et de raisonnements complexes et valides. La mise en place du web sémantique, permettra d'apporter des améliorations profondes au web actuel à savoir [Ama,2007] :

- Amélioration de l'efficacité des moteurs de recherche d'information, en dotant les requêtes de capacités à exploiter la structure d'une ontologie, et à inférer de la connaissance à partir des informations existantes dans l'ontologie.
- Donner les moyens d'accès à l'information pertinente, et permettre aux utilisateurs d'exploiter la connaissance représentée dans leurs applications.

- Réaliser l'interopérabilité entre différents systèmes d'information, en effet, l'annotation basée sur l'utilisation d'une ontologie commune, fournit un cadre commun pour l'intégration d'informations provenant de sources hétérogènes.

L'émergence du paradigme agent, permet d'introduire un ensemble d'individus, capables d'interagir, d'intentions et de capacités d'évolution, a amené les spécialistes à orienter les travaux vers l'organisation et l'étude des interactions entre ces entités.

II.4.3 Composants du Web sémantique

II.4.3.1 Ontologie

Etant un modèle de représentation des connaissances supportant des mécanismes d'inférences puissants, l'ontologie a été vue en détail dans la première partie de ce chapitre.

II.4.3.2 Ressources

Une ressource, dans le web sémantique, désigne est une entité élémentaire de représentation des connaissances. Elle peut donc être du niveau structurel ou du niveau factuel dans une modélisation. L'apparition des langages du web sémantiques comme l'XML et a permis de décrire les ressources portant sur un domaine au moyen d'un vocabulaire adapté.

II.4.3.3 Langages

Selon le consortium W3C, le web sémantique est une nouvelle vision fondée sur l'idée de devoir définir et organiser les informations sur le web de manière à ce qu'elles soient utilisables par les machines pour l'intégration, le partage et leur réutilisation.

À sa création au début des années 1990, le web était exclusivement destiné aux partages des informations sous forme de liens de pages HTML exploités par un navigateur Web.

a. XML (**eXtended Markup Language**) : Est une recommandation du W3C depuis 10 Février 1998, il utilise la notion de balisage, pour représenter un document de manière arborescente. C'est un méta langage qui permet de structurer un document en définissant ses propres balises en fonction des besoins et sans tenir compte de la signification de cette structure et des systèmes informatiques qui vont l'exploiter [Rai,2008].

Des standards comme XPath, et XQuery ont été développés afin de parcourir et d'interroger l'arborescence XML [Tua,2006]. La définition de la syntaxe et de la grammaire d'un nouveau langage basé sur XML se fait soit par une DTD *(Document Type Definition),* soit par un XSD *(XML Schema Definition)* [Bra,2006] . Principalement, un Schéma XML permet, en plus de ce qui est faisable avec une DTD, de typer les données (booléen, entier, chaîne de caractères, . . .), offrir une gamme de cardinalités plus large.

Un document XML est dit « bien formé » s'il est identifié comme étant XML par un entête et s'il satisfait les règles syntaxiques XML. Un document XML bien formé est « valide » s'il satisfait de plus les contraintes imposées par la syntaxe et la grammaire du langage, contenues dans une DTD ou un XSD [Tho,2006] [Pet,2006]. Les figures II.17.a II.17.b et II.17.c illustrent l'exemple d'un document XML, le DTD et le fichier XSD correspondants.

```
1 <?xml version="1.1" encoding="utf-8" ?>
2 <!DOCTYPE thème SYSTEM "thème.dtd">
3
4 <thème  lib = "Le web sémantique">
5 <référence>
6 <titre>les ontologies</titre>
7 <auteur>Mohamed</prénom>
8 <licence numéro = "11" />
9 </référence>
10 <référence>
11 <titre>Les langages du web sémantique</titre>
12 <auteur>Bachir</auteur>
13 <année>2010</année>
14 <licence numéro = "12" />
15 </référence>
16 </thème>
```

Figure II.17.a : Exemple de document XML

```
1 <!ELEMENT thème (référence+)>
2 <!ATTLIST thème  lib CDATA # REQUIRED >
3
4 <!ELEMENT référence (licence, titre, auteur, grade?)>
5 <!ELEMENT titre (#PCDATA)>
6 <!ELEMENT auteur (#PCDATA)>
7 <!ELEMENT année (#PCDATA)>
8 <!ELEMENT licence EMPTY>
9 <!ATTLIST licence numéro CDATA #REQUIRED>
```

Figure II.17.b : DTD associé au document XML, en figure II.17.a

Espace de nom et U.R.I Lorsqu'un document utilise plusieurs unités (DTD, ou XSD), ou lorsqu'on importe des éléments ou des attributs contenus dans des unités externes, Il peut y avoir conflit lorsqu'un même élément est défini de manières différentes dans plusieurs unités. Le parser XML, doit savoir lequel des schémas il doit appliquer, la solution consiste à utilise des espaces de noms, qui seront identifiés de façon uniques par des URIs.

On différencie les noms conflictuels, en qualifiant chaque nom par l'URI de l'espace de noms dont il provient et pour simplifier leur écriture (appelés des noms étendus) on associe à chaque espace de noms un ou plusieurs noms logiques appelés préfixe d'espace de nommage.

Un espace de nom est donc défini comme étant une collection de noms (éléments et propriétés) qui sont identifiée par une URI (***U****niform* ***R****essource* ***I****dentifier*).

```
1 <?xml version="1.1" encoding="utf-8"?>
2 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
3
4 <xsd:complexType name="thème">
5 <xsd:sequence>
6 <xsd:element name="référence" type="membreType" minOccurs="1" />
7 <xsd:attribute name="lib" type="xsd:string"/>
8 </xsd:sequence>
9 </xsd:complexType>
10
11 <xsd:complexType name="membreType">
12 <xsd:sequence>
13 <xsd:element name="titre" type="xsd:string" minOccurs="1" maxOccurs="1" />
14 <xsd:element name="auteur" type="xsd:string" minOccurs="1" maxOccurs="3" />
15 <xsd:element name="année" type="xsd:string" minOccurs="0" maxOccurs="1" />
16 <xsd:element name="licence" type="licenceType" minOccurs="1" maxOccurs="1" />
17 </xsd:sequence>
18 </xsd:complexType>
19
20 <xsd:simpleType name="licenceType">
21 <xsd:attribute name="numéro" type="xsd:integer"/>
22 </xsd:complexType>
23
24 </xsd:schema>
```

Figure II.17.c : XSD associé au document XML, en figure II.17.a

Ainsi, l'U.R.I désigne une ressource sur le web, souvent l'URI est une U.R.L(**U**niform **R**essource **L**ocator*)* qui est une chaîne de caractères indiquant l'emplacement d'une ressource sur le web. La notion d'espace de noms est une notion fondamentale puisque c'est grâce à elle que l'on peut étendre les schémas de métadonnées. Aussi, la déclaration des espaces de noms permet ensuite d'utiliser le préfixe abrégé de substitution pour référencer les URIs. En conclusion, nous pouvons dire que le langage XML représente des données structurées. Les DTD et les schémas XML, expriment des contraintes sur ces structures, cependant la sémantique des données n'est toujours pas accessible aux machines.

b. RDF(S) (**Resource Description Framework- Schema**) : L'utilisation du langage XML pour la structuration des informations, a donné lieu à diverses mises en forme qui varient selon que l'on choisit entre l'emploi d'une balise ou d'un attribut pour renseigner les données. En général aucune règle précise n'indique quand utiliser un attribut ou une balise.

C'est dans cet esprit de structurer l'information de façon plus homogène que le W3C a créé en 1999, le langage RDF(S), [Bri,2004] [Kly,2004] , ce langage préconise d'utiliser des attributs uniquement pour identifier ou référencer un élément XML. RDF, permet de représenter des objets (ressources) et des relations entre ces objets, il est à un niveau de description supérieur à XML qui est basé sur une structuration hiérarchique des documents.

En effet, le langage RDF utilise les attributs « rdf:about » ou « rdf:ID » pour identifier une ressource, et les attributs « rdf:resource » et « rdf:datatype » pour faire des références respectivement à une ressource ou à un datatype. De ce fait, RDF est alors la spécification d'un système d'expression d'assertions sémantiques simples.

La syntaxe de RDF est basée sur celle de XML. Le modèle de base de RDF est conçu pour permettre d'associer des attributs aux ressources du Web en utilisant la description de métadonnées sémantiques, c'est ainsi que RDF structure le web comme étant un ensemble de ressources reliées par les liens sémantiques [Tua,2006].

Un fichier XML/RDF a pour balise racine la balise <rdf:RDF>. Cette balise racine possède comme attributs les espaces de nommage utilisés dans le document RDF. Une assertion RDF est donnée par un triplet *(Ressource, Propriété, Valeur).*

- **Ressource** : toute expression RDF a pour but de décrire une ressource. Il s'agit d'une entité référencée par un identificateur unique (URIs). Les ressources web sont variées, et peuvent être : des pages web, un site web, ou un objet

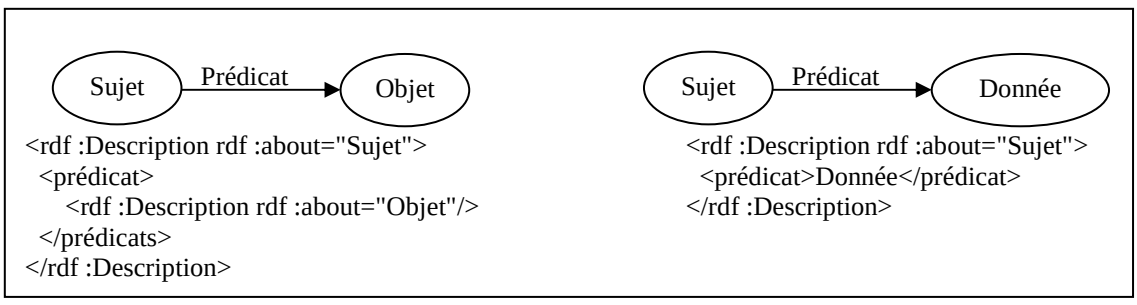
- **Propriété** : peut être un attribut ou une relation qui décrit la ressource.

- **Valeur** : c'est la valeur de la propriété associée à une ressource, elle peut être un autre énoncé (déclaration) RDF, c'est-à-dire une réification ou une ressource spécifiée par une URI, ou une chaîne de caractères simple (un littéral), ou encore un nœud blanc est un littéral (simple chaîne de caractères) ou une ressource.

Graphe RDF : l'expression RDF peut être représentée par un graphe appelé « *graphe RDF »*, il est orienté et étiqueté, le graphe RDF est composé de nœuds et d'arcs dirigés dans lequel chaque triplet est représenté par un lien « Noeud1-Arc-noeud2 ».

« Noeud1 » est étiqueté par le sujet, le second « Noeud2 » a pour étiquette l'objet, « Arc » est orienté du *sujet* vers *l'objet*, il a pour étiquette le *prédicat*. Lorsque l'objet est une simple donnée, il est représenté par un rectangle au lieu d'une ellipse qui généralement représente une ressource.

Ci après nous avons un exemple de graphe du triplet RDF <*sujet, prédicat, objet*> et du cas <*sujet, prédicat, donnée*>, et leurs descriptions en notation RDF/XML.



Le graphe RDF correspondant au document XML de la « Figure II.17.a » est en Figure II.18

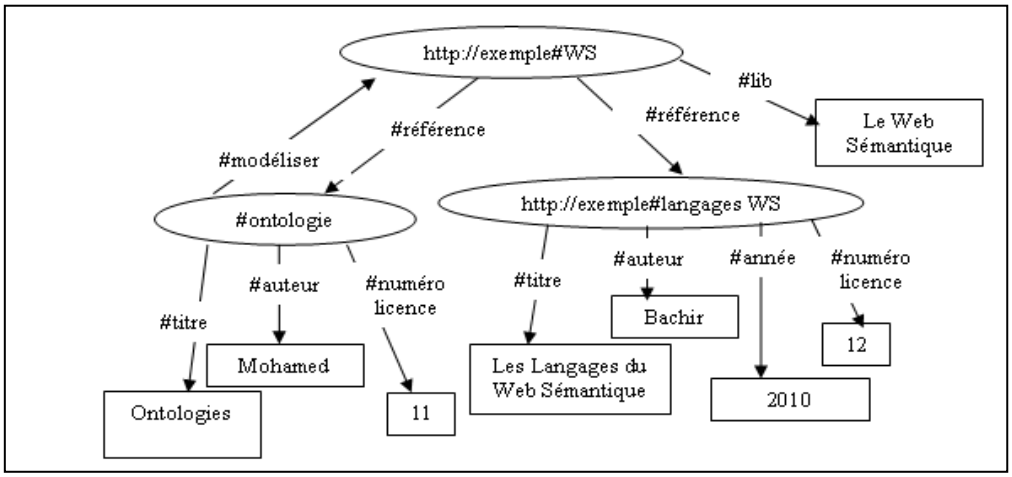


Figure II.18 : Graphe RDF du document XML en figure II.17.a

La description au format XML/RDF de ce graphe est :

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
3 xmlns="http://exemple.eu#" xml:base="http://exemple.eu">
4
5 <rdf:Description rdf:about = "#WS">
6 <lib>Le Web Sémantique</lib>
7 <référence>
8 <rdf:Description rdf:about = "#ontologie">
9 <titre>Ontologies</titre>
10 <auteur>Mohamed</auteur>
11 <numéro_licence>11</numéro_licence>
12 </rdf:Description>
13 </référence>
14 <référence>
15 <rdf:Description rdf:ID = "langages WS">
16 <titre>Les langages du Web Sémantique</titre>
17 <auteur>Bachir</auteur>
18 <année>2010</année>
19 <numéro_licence>12</numéro_licence>
20 </rdf:Description>
21 </référence>
22 </rdf:Description>
23 <rdf:Description rdf:about = "#ontologie">
24 <modéliser  rdf:resource="#WS" />
25 </rdf:Description>
26
27 </rdf:RDF>
```

Nous remarquons à travers cet exemple la puissance du langage RDF par rapport à la structuration en arbre de balises XML. En effet l'usage des attributs « rdf:ID », « rdf:about » et « rdf:resource » permettent une souplesse dans ces description. Il n'est pas possible en XML pur d'exprimer le fait que la ressource « #WS » fait référence à la ressource « #ontologie » et qu'en même temps, le ressource « #ontologie » modélise la ressource « #WS ».

Pour mieux structurer les ressources RDF, le langage RDFS pour RDF Schema a été introduit en 1999, il permet l'intégration des notions de classes et de propriétés, et leurs organisations en taxonomies de classes et de propriétés [Bri,2004].

Le langage RDFS définit une classe comme étant un type d'un ensemble de ressources, et une propriété comme étant un type d'un ensemble de prédicats. L'utilisation simultanée de RDF et de RDFS, est un outil très efficace qui permet de représenter les connaissances de domaine à travers les deux niveaux conceptuels et assertionnel. Les connaissances factuelles ou assertionnelles sont exprimées en RDF, et les connaissances structurelles sont modélisées par le langage RDFS.

c. OWL (Web Ontology Language): L'expressivité du langage RDF(S), comme décrite auparavant est limitée, RDF étant limité aux prédicats binaires, alors que RDF Schéma se limite lui aussi à décrire des hiérarchies de classe et de propriétés fondées sur les concepts de domaine et de Co-domaine (range).

Le groupe de travail W3C a identifié des caractéristiques de cas d'utilisation pour les ontologies web, qui exigent plus d'expressivité que ce qu'offre RDF(S), c'est à dire un langage puissant, les résultats des groupes de recherches ont aboutis au langage DAML+OIL, (DAML : proposition américaine, OIL version européenne) qui ensuite a été pris comme point de départ par le groupe du W3C pour définir le langage OWL qui est destiné a être un standard accepté comme langage d'ontologies pour le web sémantique « Figure II.19 ».

L'idée de vouloir juste étendre RDF(S) se heurte à la dualité entre la force d'expressivité et l'efficacité du support de raisonnement, car plus un langage est expressivement riche, plus son support de raisonnement devient inefficace, on a donc besoin d'un langage de compromis, qui constitue le support pour des raisonneurs efficaces et raisonnables tout en offrant suffisamment d'expressivité pour permettre de décrire des ontologies et des connaissances.

Ces besoins ont incité le groupe de travail du W3C à définir trois différents sous langages qui offrent des capacités d'expressivités croissantes et dont chacun est destiné à satisfaire une communauté différente d'utilisateurs.

OWL-Lite : Son expressivité est minimale, mais il a la calculabilité maximale, c'est un langage facile à comprendre et à implémenter, il convient aux utilisateurs qui désirent définir des hiérarchies de classification et de contraintes simples. Il supporte des contraintes de cardinalité.

OWL-DL : Il est plus complexe que OWL Lite, permettant une expressivité maximale, fondé sur les logiques de description d'où il tire son nom, il convient aux utilisateurs qui désirent

avoir le maximum d'expressivité tout en maintenant la complétude (toutes les inférences sont calculables) et la décidabilité (leurs calcul se fait en une durée finie). Il inclut tous les constructeurs RDF et OWL, mais leur utilisation est soumise à des restrictions, par exemple, lorsqu'une classe est sous classe de plusieurs autres classes, elle ne peut pas être une instance d'une autre classe.

Ces restrictions ont l'avantage de permettre d'écrire des supports de raisonnement efficaces.

OWL-Full : Etant la version la plus complète et la plus complexe d'OWL, elle offre le plus haut niveau d'expressivité, et de liberté syntaxique de RDF, sans aucune garantie sur la complétude et la décidabilité des calculs liés à l'ontologie, par exemple dans OWL-Full on peut traiter une classe comme étant à la fois une collection d'individus et comme un individu (instance). OWL-Full utilise toutes les primitives du langage autorisant aussi leurs combinaisons par RDF(S), son avantage est qu'il est entièrement compatible avec RDF syntaxiquement et sémantiquement, ainsi un document RDF est aussi un document OWL-Full légal, et toute conclusion RDF(S) valide est aussi une conclusion OWL-Full valide.

En OWL, une relation est appelée une *propriété* par référence à son origine RDF(S), ou bien un *rôle* en faisant correspondance avec les logiques de descriptions. Le langage OWL permet une modélisation des connaissances de domaine [Dea,2004], par une représentation se basant sur le regroupement de faits en ensembles de structures génériques (classes).

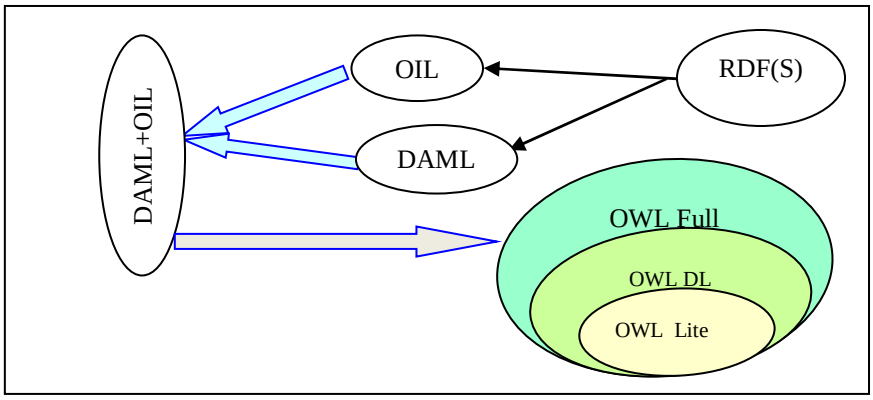


Figure II.19 : Origines du langage OWL

c1. Connaissances structurelles

Classe : Représente un type d'un ensemble d'individus ayant des caractéristiques communes. Par exemple, la classe *EtudiantInformatique* est l'ensemble des individus étudiants en informatique, ces individus ont pour caractéristiques communes d'être des étudiants de la filière informatique.

Il existe deux classes prédéfinies en OWL, qui sont les classes *« owl:Thing »* dont l'extension est l'ensemble de tous les s individus du domaine, et *« owl:Nothing »* qui a pour extension l'ensemble vide. Les classes sont d'une modélisation sont *élaborées* par des descriptions de classes ou des axiomes de classes. La définition d'une classe peut être réalisée par différentes descriptions, ou par des axiomes. Pour les descriptions il s'agit de : *restrictions de rôles, l'énumération, Intersection, union et classe complémentaire.*

Les axiomes se rapportent à la définition par des descriptions de classes, *les sous classe, la disjonction et l'équivalence.*

Rôles: Un rôle permet de mettre en relation une classe avec une autre classe ou bien une classe avec un type de donnée (datatype). En RDF(S) un rôle est aussi appelé une propriété.

Le langage OWL distingue deux catégories de rôles :

– Un rôle d'objet (Objectproperty) relie une classe à une autre ; ainsi une assertion d'un rôle d'objet liera un individu à un autre individu. Un rôle d'objet est de type « *owl:ObjectProperty* ».

– Un rôle de type de donnée (Datatypeproperty) relie une classe à un datatype ; ainsi une assertion de rôle de type de donnée liera un individu à une valeur de donnée. Un rôle de type de donnée est de type « *owl:DatatypeProperty* »

La description de rôles inclut la définition des domaines et co-domaine, par défaut, le domaine et le co-domaine d'un rôle sont la classe « *owl:Thing* ». La propriété « *rdfs:domain* » relie un rôle à une description de classe, et précise que les sujets de la propriété doivent appartenir à l'extension de la classe indiquée.

La propriété « *rdfs:range* » relie un rôle soit à une description de classe soit à un datatype définit dans le Schéma XML, et précise que les objets de la déclaration de rôle doivent appartenir à l'extension de la classe indiquée ou à une valeur du datatype.

Le description de rôle se fait aussi par les propriétés de *Transitivité, symétrie et inverse* qui sont applicables sur les rôles d'objets, les domaines et co-domaines sont des classes et non des datatype, une *Contrainte sur le rôle* est une contrainte de cardinalité sur le rôle. Les propriétés *« owl:FunctionalProperty »* et *« owl:InverseFunctionalProperty »* permettent d'établir des contraintes de cardinalités agissant respectivement sur le domaine et sur le co-domaine du rôle.

Les axiomes de rôles concernent les sous rôles: L'héritage entre rôles est exprimée par la propriété *« rdfs:subPropertyOf »*, et l'équivalence: La propriété « *rdfs:equivalentProperty »* permet de déclarer que deux rôles ont la même extension.

c2. Connaissances factuelles

Individu : Un individu est une instance ayant le type d'une ou de plusieurs classes, sa déclaration en XML/OWL se fait par une balise ayant le nom de la classe de son type, le nom de l'instance est renseigné par l'attribut *« rdf:about »* ou *« rdf:ID »*.

Par exemple, la déclaration que Mohamed est un étudiant se fait par :

<Etudiant rdf :about= "#Mohamed"/>

Cette déclaration est aussi équivalente à la déclaration ci-dessous qui est plus générale :

<owl:Thing rdf:about="#Mohamed">

<rdf:type rdf:resource="#Etudiant" />

</owl:Thing>

Avec OWL, des URIs différentes peuvent identifier ou non le même individu, pour les différencier nous avons les trois propriétés suivantes:

-La propriété*« owl:sameAs »* déclare que deux identifiants, se rapportent au même individu.

-La propriété *« owl:differentFrom »*déclare que deux identifiants ne font pas référence à la même instance.

-La structure *« owl:AllDifferent »* utilisée avec la propriété *« owl:distinctMembers »* constituent une forme d'écriture de la propriété *« owl:differentFrom »* pour différencier deux à deux les éléments d'un ensemble d'individus.

Valeurs de rôle : Un individu est une instance ayant le type d'une ou de plusieurs classes, sa déclaration en XML/OWL se fait par une balise ayant le nom de la classe de son type, le nom de l'instance est renseigné par l'attribut *« rdf:about »* ou *« rdf:ID »*, ou encore il peut ne pas être identifié dans ce cas il est anonyme, c'est le cas où il est valeur d'un rôle.

Un individu peut être lié à d'autres individus ou à des données par des assertions de rôles.

Conclusion

Dans ce chapitre nous avons passé en revue les principaux concepts ayant trait direct avec la notion de la sémantique des documents. Nous avons dans un premier temps abordé les notions fondamentales liées à la connaissance en tant qu'élément de base du savoir.

Le cycle de vie étant décrit brièvement, il a été question d'aborder les modèles de représentation des connaissances avec les mécanismes de raisonnement associés. L'accent a été mis particulièrement sur le modèle des graphes conceptuels. D'un coté le choix de ce modèle est justifié par les caractéristiques de formalisation basées logique, et d'un autre côté pour son expressivité et les mécanismes de raisonnements dont il dispose.

L'opération de projection étant à la base du raisonnement de notre modèle, nous avons traité les mécanismes de cette opération avec un plus de détails et d'intérêt. La description de l'ontologie de domaine, avec des connaissances factuelles et des connaissances structurelles, nous a conduits à examiner l'essentiel des concepts de la collection des langages du web sémantique. Ce sont les descriptions de classes et de rôles et les axiomes qui permettront d'exécuter les mécanismes de raisonnement lors des recherches d'informations que supporte notre modèle.

Le suivant chapitre, présentera quelques plateformes d'annotation sémantique de documents et des systèmes de recherche sémantique d'information basés sur les différents modèles vus dans ce chapitre.

CHAPITRE III

Recherche Sémantique d'Information sur le Web Basée Systèmes Multi-Agents.

III.1 Introduction

Concevez toujours une chose en la considérant dans un contexte plus large - une chaise dans une pièce, une pièce dans une maison, une maison dans un quartier, un quartier dans une ville.

Eliel Saarinen

L'annotation sémantique est un outils efficace permettant de sélectionner l'information pertinente sue le web, leur mise en œuvre est devenue une nécessité absolue. L'information dans une première phase de traitement doit être interprétée, l'intention de la requête utilisateur doit être révélée et bien réfléchie par la prise en compte du contexte.

[Lia,2011] [Bec,2002] distingue plusieurs types d'annotation :

- L'annotation textuelle, qui est l'ajout de commentaires et de notes aux objets.
- Lien d'annotation qui relie des objets à des contenus lisibles.
- Annotation sémantique qui consiste en une information sémantique lisible par la machine.

Suivant cette classification, l'annotation sémantique est considérée être des métadonnées compréhensible par les agents humains et logiciels.

Les composants de l'annotation sémantiques montrés en Figure III.1 sont :

- Les ressources à annoter (textes, images, vidéos, etc.)
- L'ontologie d'annotation
- Le modèle d'annotation sémantique.
- Les applications utilisateurs.

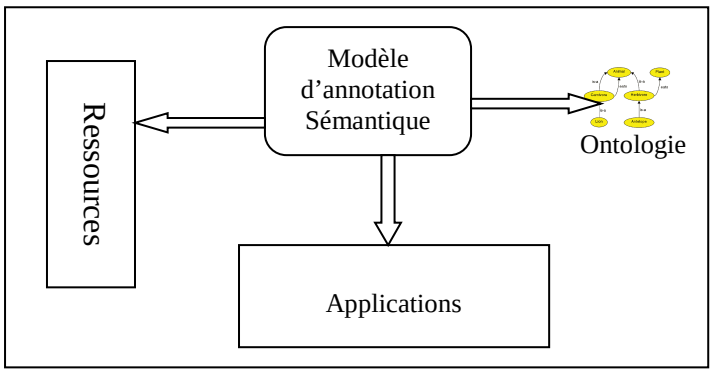


Figure III.1: Composants d'annotation sémantique [Lia,2011]

Dans ce chapitre, diverses architectures de systèmes d'annotation sémantique sont détaillées, ensuite un rappel sur les mesures de similarités sémantiques est donné pour conclure avec le paradigme agent. L'architecture interne d'un agent, l'environnement de l'agent, et les systèmes multi agents sont présentés.

La communication, étant d'un intérêt capital pour la coopération entre agents est traités, tout comme les types de communication, les langages et les protocoles d'interactions

sont revus. Ce chapitre se termine par la présentation de quelques systèmes de recherche sémantique d'information sans ou avec agent.

III.2 Outils d'annotation sémantique

Les systèmes d'annotation sémantique sont classés en fonction des méthodes utilisées. Il ya deux grandes catégories qui sont les méthodes basée modèle et les méthodes basées sur l'apprentissage.

Le système « Annotea » développé par le W3C est un environnement collaboratif pour l'annotation manuelle des documents web, les annotations peuvent être des commentaires, des explications, le format de Annotea est RDF, le type de document pouvant être annotés se limite aux documents structurés HTML/XML, le projet fournit dans Xpointer une méthode qui localise les annotations dans le document, Xpointer étant une recommandation du W3C qui identifie des fragments de ressources URIs [Mar,2005]. Annotea est intégré dans nombre d'outils d'annotation : «Amaya » et «Annozilla ».

Un autre système « Mangrove » permet aussi d'effectuer une annotation manuelle, Il fournit à l'utilisateur l'interface pour créer des balises de documents HTML et associer des étiquettes d'annotations aux textes.

Pour l'annotation semi automatique, nous citons le Framework « OntoMat », une implémentation de SCREAM (Semi automatic CREAtion of Metadata). Il offre un navigateur Web pour afficher la page annotée et fournit quelques fonctions utilisateur adaptées pour l'annotation manuelle [Han,2003]. OntoMat, utilise le système d'extraction d'information (IE) « Amilcare », l'utilisateur annote un document servant d'exemple et le système apprend comment reproduire l'annotation de l'utilisateur, pour pouvoir suggérer des annotations pour de nouveaux documents, c'est une annotation par apprentissage.

Aussi nous avons le système d'annotation « MnM » [Var,2002] qui intègre une ontologie pour l'annotation et repose sur l'outil linguistique d'extraction d'information « GATE ». Ce système fournit l'environnement pour annoter manuellement un corpus d'apprentissage, , il stocke les documents balisés en tant que versions étiquetées de l'original, plutôt que les formats RDF du Web Sémantique.

Il inclut un navigateur HTML pour afficher les documents et avec le fonctionnement d'un navigateur d'ontologie. Une force de MnM est qu'il fournit des APIs ouvertes pour se connecter aux serveurs d'ontologie et pour intégrer les outils d'extraction de l'information.

« Armadillo » est un système pour la création non supervisée des bases de connaissances à partir d'entrepôt et l'annotation de documents [Cir,2001]. Il utilise la redondance d'information dans des entrepôts pour amorcer l'apprentissage d'exemples choisis par l'utilisateur. L'extraction d'information est utilisée pour généraliser ces exemples et pour trouver de nouveaux faits.

La confirmation par plusieurs sources (documents) est alors exigée pour vérifier la qualité des données saisies. Après confirmation, l'apprentissage peut être lancé une nouvelle fois. Ce processus peut être répété jusqu'à ce que l'utilisateur soit satisfait de la qualité d'information issue après l'apprentissage. Armadillo utilise des techniques, comme les recherches basées sur des mots clés et l'outil « Amilcare » d'extraction de l'information.

La plateforme KIM (Knowledge Information Management), est une infrastructure qui fournit un ensemble de services dont un module d'annotation sémantique automatisé. Ce processus se base sur l'architecture d'ingénierie de texte GATE (General Architecture for Text Engineering), une plate-forme d'ingénierie linguistique qui offre les ressources nécessaires à la réalisation d'un moteur d'extraction d'information générique.

Les auteurs [Bor,2003] définissent les caractéristiques et les besoins du service d'annotation de la plateforme KIM par :

- Le système d'annotation sémantique requiert, l'usage d'une ontologie de haut niveau qui structure en relations les classes des entités nommées génériques.
- L'usage d'un langage de description telle que RDF(S), et OWL Lite, pour maintenir l'efficacité d'expressivité et du raisonnement.
- Le stockage de l'ontologie et de la KB se fait dans un repository SESAME Rdf.
- La recherche utilise une version améliorée de LUCENE basé mots clés.

L'architecture de la plateforme KIM, comporte l'ontologie KIMO, inspirée des ressources comme OpenCyc, WordNet 1.7, et DOLCHE, cette ontologie définit environ 200 classes d'entités d'ordre général, et 100 attributs et relations.

Les descriptions sémantiques des entités et de leurs relations sont tenues dans une base de connaissances qui regroupe environ 80000 entités, concernant 50000 lieux, 282 pays et 4700 villes avec des descriptions des montagnes , des rivières, des mers et des océans les plus communs. Aussi la KB définit les plus importante organisations comme UN, NATO, OPEC etc. au total nous avons environ 8400 instances d'organisations définies dans cette KB, et pour chaque entité extraite du texte il est établi:

- Un lien (URI), vers la classe la plus spécifique dans l'ontologie.

- Un lien vers l'instance spécifique dans la base de connaissance.

L'ontologie KIMO et la base de connaissances, sont maintenues par l'exploitation des technologies et les standards du web sémantique, à savoir les langages RDF(S), les middlewares, et des raisonneurs. Les autres composants de l'infrastructure KIM, sont l'API KIM server, une interface permettant diverses méthodes d'accès, et un explorateur de base de connaissances.

Les modules de l'API KIM server fournissent les services d'annotation sémantique, de gestion documentaire basée GATE, d'indexation et de recherche. L'évaluation et les performances des testes sont encourageant, en effet, les mesures de précision et de rappel obtenus sur un corpus de 100 documents, par rapport à l'annotation manuelle sont proches (de l'ordre de 84%).

Une autre conception d'une démarche d'annotation sémantique est décrite dans [Thi,2010]**,** dans cette description les auteurs révèlent une dépendance entre une méthode d'annotation et le niveau structurel du corpus documentaire, à savoir des textes libres, structurés ou semi structurés. Ce projet vise à annoter sémantiquement à l'aide d'une ontologie et de manière automatique non supervisée , une collection de documents hétérogènes comportant des parties structurés et des parties libres, l'ontologie comporte une composante lexicale où chaque concept est accompagné de plusieurs labels, d'un ensemble d'entités nommées, et des termes d domaine d écrivant les instances de concepts.

La démarche consiste à repérer des termes, des entités nommées ou des concepts dans les nœuds d'un arbre DOM (Document Objet Model) qui décrit la structure d'un document (HTML /XML). Les termes ou les entités nommées extraits sont rapprochés des termes ou labels de l'ontologie lexicale pour identifier les concepts candidats. Le mécanisme exploite la proximité structurelle des nœuds instances pour déduire la possibilité d'existence des relations sémantique.

L'architecture comporte trois composantes, l'extraction de termes et enrichissement de la composante lexicale ontologique, l'annotation des nœuds constituant l'arbre DOM, et la formulation de requête à l'aide de métadonnées.

III.3 Mesures de similarités sémantiques

Les recherches montrent que le fait de comparer deux choses produit un savoir, Miner [Min,1987] écrit « *Il est manifestement impossible de comparer ce qui est identique. Des différences doivent exister ou alors nous identifions plus que nous comparons. De la même*

façon, si les différences sont trop grandes, la comparaison devient infaisable, les résultats logiques ou pratiques ne satisfont pas.» Ainsi, il semble que comparaison et similarité soient étroitement liées.

La similarité est au cœur de plusieurs travaux dans différents domaines tels que l'analyse de données, le raisonnement à partir de cas, la reconnaissance des formes, la résolution de problèmes, l'apprentissage, le transfert, ...

La similarité sémantique qui exprime une liaison entre deux concepts est une capacité abstraite de l'homme, les machines n'ont pas la faculté de l'interpréter. Il est évident pour un humain que les concepts « stylo » et « papier » sont liés beaucoup plus que « température » et « chaise ». Cet état de fait, difficile à formaliser sans recours aux ressources sémantiques : les ontologies permettent de montrer les liens entre les concepts (hyperonymie, antonymie, etc.).

La similarité sémantique est un concept important pour le champ de recherche d'informations, en effet les problèmes de polysémie et de synonymie qui sont respectivement liés à la précision et au rappel d'un système IR, génèrent des ambiguïtés et incitent à passer au niveau sémantique. Avec une ontologie, il serait possible de savoir que l' « avocat » dans un document est un « fruit vert » est que celui d'une requête est un « défenseur », que « voiture » et « automobile » réfèrent tous les deux le même concept.

Le calcul des mesures de similarité entre concepts d'une ontologie, constitue aussi un autre défi auquel diverses solutions existent et qui sont classées en deux grandes catégories :

- Approches basées sur la structure de l'ontologie (edge counting)
- Approches utilisant le contenu informatif des concepts

III.3.1 Méthodes basées « Edge Counting »

La première catégorie regroupe un ensemble de mesures communément appelées des mesures basée sur la longueur de chemin liant deux concepts C_1 et C_2, la longueur est exprimée par le nombre d'arcs du plus court chemin entre C_1 et C_2.

Etant donné un graphe dont les nœuds représentent les concepts de le l'hiérarchie, selon [Lea,1998] [Li,2003] la démarche très intuitive pour mesurer la distance entre deux concepts C_1 et C_2 est de parcourir un chemin pour aller du concept C_1 au concept C_2. La similarité est évaluée en fonction de la longueur du plus court chemin liant les concepts C_1 et C_2 et de leurs positions (profondeurs) dans l'hiérarchie. Dans ce regard la similarité définie dans [Wu,1994] est liée à la distance (nombre d'arcs) et tient compte de la position du subsumant le plus spécifique des deux concepts comparés.

L'idée est de décrire la cohésion des concepts et normaliser leur différence. L'expression est donnée par l'équation III.1 où :

C : dénote le concept subsumant le plus spécifique « MSCS » des concepts C_1 et C_2 .

Depth(C) : la longueur (en terme profondeur) du concept C, à partir de la racine.

Depth(Ci) : la longueur du concept Ci , à partir de la racine.

$$Sim_{WP}(C_1, C_2) = \frac{2 * Depth(C)}{Depth(C_1) + Depth(C_2)} \qquad (III.1)$$

Les arcs sont affectés du poids unitaire et sans tenir compte de leurs orientations dans la taxonomie, mais certaines approches [Sed,2010] différencient les poids des arcs et tiennent compte de leurs directions, ce qui permet de représenter des relations entre concepts en fonction du poids et de la direction des arcs qui les interconnectent.

Pour reproduire l'influence des caractéristiques des arcs, des facteurs poids α et β sont introduits pour mesurer la similarité tenant compte des sens de spécialisation et de généralisation. L'expression est donnée en équation III.2

$$Sim_{WP}(C_1,C_2) = \underset{J=1...m}{Max}\left(\alpha^{S(P_J)}\beta^{G(P_J)}\right) \qquad (III.2)$$

α, β sont les poids associés en descendant (spécialisation) et en remontant (généralisation) respectivement dans l'hiérarchie.

$P_1,P_2,\ldots,P_m$ sont les arcs connectant les concepts C_1 et C_2 .

$S(P_J)$: nombre d'arcs dans la direction spécialisation.

$G(P_J)$: nombre d'arcs dans la direction généralisation.

Une autre mesure acceptée dans cette catégorie est donnée par l'équation III.3, où :

N_0: distance du concept C, le « MSCS » des concepts C_1, C_2 depuis la racine.

N_1: distance depuis C à C_1 et N_2: distance de C à C_2.

$$Sim_{WP}(C_1, C_2) = \frac{2 * N_0}{2 * N_0 + N_1 + N_2} \qquad (III.3)$$

De même, de manière similaire, plus la distance entre deux concepts est grande, moins ils seront similaires [Rad,1989] [Mil,1989], une relation inversement proportionnelle donnée par l'équation III.4

$$Sim_{Rada}(C_1, C_2) = \frac{1}{1 + Dist(C_1, C_2)} \qquad (III.4)$$

$Dist(C_1,C_2)$ est une distance métrique, qui indique le nombre d'arcs minimum à parcourir entre C_1 et C_2 exprimée en nombre d'arcs.

Leacock and Chodorow [Lea,1998], ont défini une mesure de similarité suivant la longueur du plus court chemin "IS-A" entre les mots compares, la formule est donnée par l'équation III.5.

$$Sim_{lcha}(C_1, C_2) = \underset{I}{Max}\left[-Log\left[\frac{length_I(C_1, C_2)}{2D} \right] \right] \quad \text{(III.5)}$$

$Length_I$ (C_1,C_2) : longueur du chemin "I". et D : profondeur maximale de l'hiérarchie.

D'ailleurs, dans [Ces,2003], nous avons une approche qui combine des notions de probabilités avec la méthode basée longueur de chemin. Pour N concepts C_1,C_2,…C_N , cette approche décrit un concept par un terme qui le dénote, une description du sens du concept et un ensemble de relations qui l'interconnectent aux autres concepts.

Dans ce contexte, un réseau sémantique représente l'ensemble des concepts et leurs relations, le poids associé à chaque arc reliant C_1 et C_2 exprime la force du lien sémantique entre ces concepts.

$$D(C_1, C_2) = \underset{i\in\{1,n\}}{Max}\left(\prod_{j=1}^{m_i} Pij \right) \quad \text{(III.6)}$$

La relation sémantique entre deux concepts peut alors être exprimé par des une fonction distance de probabilités donnée par l'équation III.6.

n: nombre de chemins de C_1 à C_2

m_i: nombre d'arcs du chemin "i".

Pij: la probabilité associée à l'arc "j" du chemin "i".

La distance D a les propriétés:

1. $D(C_1,C_2)\neq D(C_2,C_1)$
2. $D(C_1,C_2)=0$ SSI il n'existe aucun chemins de C_1 à C_2
3. $D(C_1,C_3)\neq D(C_1,C_2)+D(C_2,C_3)$

III.3.2 Méthodes basées « Information Content »

Il s'agit de considérer le contenu informatif (IC) des concepts de l'ontologie, Le contenu informationnel d'un concept traduit la pertinence d'un concept dans le corpus en tenant compte de sa spécificité ou de sa généralité. Le principe, plus un ensemble de concepts partagent de l'information, plus ils sont similaires [Var,2005] [Jia,1998].

Les approches utilisant cette notion, calculent la probabilité de trouver un concept ou l'un de ses descendants dans le corpus, pour cela on calcule une fréquence qui regroupe la fréquence d'apparition du concept lui-même et celles des concepts qu'il subsume.

Soit C le concept, et P(C) la probabilité mesurée (probabilité de retrouver une instance du concept C) alors le contenu informatif associé à C est défini par la relation :

$$IC_{Resnik}(C) = -\log((P(C)). \qquad \text{(III.7)}$$

P(C)=Fréquence(C)/N , N :Nombre total de concepts

Le mesure de Resnik, calcule la similarité entre deux concepts C_1,C_2 par le contenu informatif de leurs subsumant le plus spécifique « MSCS », [Res,1995] [Res,1999]

Cette même expression peut être réécrite $Sim(C_1,C_2)=IC(MSCS(C_1,C_2))$ (III.8.a)

$$Sim(C_1,C_2)=Max[IC(C)] ; \qquad \text{(III.8.b)}$$

$C \in S$ = ensemble des subsumants de C_1,C_2

Une autre approche, [Sec,2004] n'utilise pas un corpus, mais calcule le contenu informatif d'un concept à partir de WordNet, elle repose sur l'idée que plus un concept « C » a des descendants, moins il est informatif, les hyponymes de concepts sont donc utilisés pour calculer le c contenu informatif de concepts. Dans ce sens, les concepts feuilles de la taxonomie sont les plus informatifs

$$IC_{wn}(C) = \frac{\log\left(\frac{hypo(C)+1}{Max_{wn}}\right)}{\log\left(\frac{1}{\max_{wn}}\right)} = 1 - \frac{\log(hypo(c)+1)}{\log(\max_{wn})} \qquad \text{III.9}$$

hypo(c) : Nombre d'hyponyme du concept « C » ; max_{wn} : est une constante qui désigne le nombre de concepts de la taxonomie WordNet.

Cette formule, pourvoit que le contenu informatif décroit en partant des nœuds vers la racine.

Une autre métrique de cette catégorie est développée par Dekang Lin [Lin,1998]

Il annonce la définition de sa mesure par "*The similarity between C_1 and C_2 is measured by the ratio between the amount of information needed to state the commonality of C_1 and C_2 and the information needed to fully describe what C_1 and C_2 are.*"

Dans sa forme cette mesure est similaire à celle de Wu & Palmer, mais à la différence qu'elle considère les contenus informatifs des concepts, et non les longueurs de chemins. Formellement la mesure est exprimée par l'équation III.10

$$Sim_{Lin}(C_1, C_2) = \frac{2 * \text{Log}(P(C_0))}{\text{Log}(P(C_1)) + Log(P(C_2))} \qquad \text{(III.10)}$$

$P(C_0)$: probabilité de rencontrer le subsumant le plus spécifique C_0 de C_1 et C_2 .

P(Ci) i=1,2 : probabilité de rencontrer le nœud représentants le concept Ci ou une instance de celui-ci.

Enfin, Jiang et Conrath (Jcn) [Jia,1998] combinent la méthode de chemin avec celle du contenu informationnel de Resnik pour définir une distance entre les deux concepts C_1 , C_2
$Dist(C_1,C_2)=IC(C_1)+IC(C_2)-2*IC(MSCS(C_1,C_2))$:

$MSCS(C_1,C_2)$: Subsumant le plus spécifique des concepts C_1 et C_2.

Dans la littérature, les mesures de similarités sémantiques développées sont très diverses, mais se situant dans l'une ou l'autre des catégories indiquées précédemment.

Dans le cadre de cette thèse, nous avons proposé une mesure de similarité sémantique entre concepts basée WordNet, nous nous sommes inspiré des travaux de Wu & Palmer pour développer cette mesure, l'expérimentation donne des résultats très similaire à ceux de l'approche Wu & Palmer, l'amélioration se situe au niveau de la distribution des valeurs calculées dans l'intervalle [0, 1] et ce grâce à l'usage et l'exploitation des caractéristiques très intéressantes de la fonction logarithmique pour cette conceptualisation.

III.4 Le paradigme agent en intelligence artificielle

L'Intelligence Artificielle (IA), est apparue comme une discipline informatique qui vise la reproduction des activités intellectuelles de l'homme par la conception d'entité qualifiée d'intelligente. Dès lors, les domaines d'applications de l'IA, ne cessaient d'évoluer, les systèmes d'informations actuels sont développés dans des environnements distribués, ouverts et hétérogènes. Cette constatation remettait en cause les méthodes et stratégies utilisées par les programmes de l'IA. En effet, certains problèmes posés sont de caractère distribué et nécessitent la coopération de plusieurs entités à la fois. Cette nouvelle tendance a donné lieu à la naissance de l'intelligence artificielle distribuée (IAD).

L'IAD, se situe à l'interconnexion de plusieurs disciplines incluant l'intelligence artificielle, les systèmes informatiques distribués, le génie logiciel, la sociologie, …etc.

En 1999, G. Weiss [Wei,1999] proposa la définition suivante « *L'IAD est l'étude, la conception et la réalisation de système multi agents, c'es à dire de systèmes dans lesquels des agents intelligents qui interagissent, poursuivent un ensemble de buts ou réalisent un ensemble d'actions* ».

La distribution des fonctionnalités d'un système exige la division du problème en sous problèmes, les problèmes trop complexes imposent une vision locale permettant l'obtention de résultats qui souvent émergent des interactions locales. C'est ainsi que la notion d'agent a été présentée : des entités informatiques qui sont capables de raisonner et de coopérer, elles doivent

être dotées de capacités d'interaction, de perception et d'action sur l'environnement avec une certaine autonomie sur son comportement [Stu,1995].

Qu'est ce qu'un agent

Pour Jennings et Wooldridge, [Woo,2002] un agent par « *Un agent est un système informatique, situé dans un environnement, et qui agit sur cet environnement de façon autonome pour atteindre les objectifs pour lesquels il a été conçu.* »

Situé : L'agent reçoit des entrées de l'environnement, il est capable d'agir sur son environnement à partir de ces entrées.

Autonome : Cette définition aborde une notion essentielle : L'autonomie, un concept au centre de la problématique des agents, c'est la faculté d'avoir ou non le contrôle de son comportement sans l'intervention d'autres agents ou d'êtres humains.

L'autre définition, celle de [Fer,1995], très acceptée dans la communauté est «*On appelle agent une entité physique ou virtuelle qui:*

- *Est capable d'agir dans un environnement,*
- *peut communiquer directement avec d'autres agents,*
- *mû par un ensemble de tendances (sous la forme d'objectifs individuels ou d'une fonction de satisfaction, voire de survie qu'elle cherche à optimiser),*
- *possède des ressources propres,*
- *est capable de percevoir (mais de manière limitée) son environnement,*
- *ne dispose que d'une représentation partielle de cet environnement (et éventuellement aucune),*
- *possède des compétences et offre des services,*
- *peut éventuellement se reproduire,*
- *tend par son comportement à satisfaire ses objectifs, en tenant compte des ressources et des compétences dont elle dispose, et en fonction de sa perception, de ses représentations et des communications qu'elle reçoit.* »

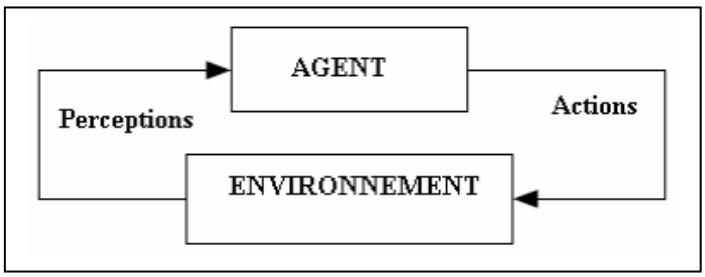


Figure III.2 : Action de l'agent sur l'environnement [Stu,1995]

A partir des définitions précédentes, les principales caractéristiques d'un agent sont « Figure III.2 » :

- *L'autonomie :* L'agent possède un état interne sur lequel il a un contrôle total. Cet état interne est inaccessible aux autres agents. De plus, l'agent prend des décisions qui sont basées sur cet état interne et selon ses objectifs sans intervention extérieure (humaine ou d'un autre agent).
- Un agent est *réactif* : il adapte ses actions en fonction de l'environnement qu'il perçoit. C'est un comportement de réponse à un stimulus. L'agent doit être capable d'élaborer des réponses aux changements de son environnement dans les temps requis.
- Un agent est *proactif* : il prend lui-même des initiatives, et choisit des actions en fonction de ses objectifs et des connaissances qu'il possède et aux bons moments.
- Un agent possède une *habilité sociale :* il est en mesure d'interagir avec d'autres agents pour satisfaire les tâches qui lui sont confiées, coopérer ; et résoudre des conflits (de ressources, ou de buts).

Selon [Fer,1995], les agents opérant dans un environnement commun de façon collective et décentralisée doivent accomplir des tâches, telles que :

- La résolution distribuée de problèmes complexes : utilisation d'un ensemble de spécialistes qui possèdent des compétences complémentaires.
- La résolution de problèmes distribués, qui comprend l'analyse, l'identification, et le contrôle de systèmes physiquement distribués, comme exemple le contrôle de réseau de communication.
- Résolution par coordination.
- La complexité des développements logiciels nécessitant la modularité et l'interopérabilité.

III.4.1 Architecture interne d'un agent

L'architecture interne désigne l'ensemble des structures de données et des processus internes à un agent lui permettant de prendre une décision (éventuellement rationnelle) consistant à choisir une action en vue de modifier l'environnement. On distingue deux types d'agents en fonction de leur architecture interne [Woo,2002]**:**

III.4.1.1 L'agent cognitif

Un agent cognitif a des capacités de raisonnement développées. Il possède :

- Une représentation explicite de ses objectifs ;
- Une représentation évoluée de l'environnement ;
- Une capacité à manipuler ces représentations pour anticiper ou réévaluer ces objectifs.

L'agent dispose d'une base de connaissance comprenant l'ensemble des informations et procédures du Le savoir-faire nécessaires à la réalisation de sa tâche et à la gestion des interactions avec les autres agents et avec son environnement. L'architecture BDI (Belief Desire Intention) constitue un type d'architecture d'agents cognitifs.

- Les croyances correspondent aux informations (éventuellement incomplètes et incorrectes) qu'à l'agent de son environnement ;
- Les désirs correspondent aux états de l'environnement que l'agent souhaiterait avoir.
- Les intentions correspondent aux projets de l'agent pour satisfaire ses désirs.

III.4.1.2 L'agent réactif

Un agent réactif agit selon des règles du type stimulus-réponses, il ne dispose pas de représentation interne explicite de son environnement. Les prises de décision d'un tel agent peuvent être représentées par une machine à états finis.

III.4.2 Environnement d'un système multi-agents

Russel propose une classification de l'environnement selon les propriétés suivantes [Stu,1995] :

- *Accessible/ inaccessible* : Dans un environnement accessible, l'agent a accès à l'état complet de l'environnement, toute information est connue et à jour, par conséquent Internet est un environnement inaccessible.

- *Déterministe/Non-déterministe*: Le prochain état de l'environnement est-il déterminé par son état courant et par l'action sélectionnée par l'agent ? Un environnement déterministe permet de garantir l'effet d'une action dans le temps.

- *Episodique/Séquentiel* : un environnement est discret s'il existe un nombre fini d'états que l'agent peut atteindre, dans l'environnement continu, il peut atteindre un nombre illimité d'états.

- *Statique/Dynamique*: L'environnement dynamique peut changer pendant la prise de décision de l'agent, par des processus et indépendamment de l'agent, un environnement statique est modifié seulement par les actions de l'agent.

III.4.3 Les Systèmes Multi-Agents

L'approche multi-agents en tant que méthodologie issue de l'IAD, propose aux concepteurs de systèmes informatiques de considérer ces systèmes comme étant composés d'agents organisés dans un environnement. Ces agents communiquent entre eux pour résoudre des problèmes complexes. « Figure III.3 ».

L'expression « agents organisées » signifie qu'il existe une ou plusieurs organisations, qui définissent les règles de coexistence et de travail collectif entre les agents (définition des rôles, de partage de ressources, dépendances de tâches …)

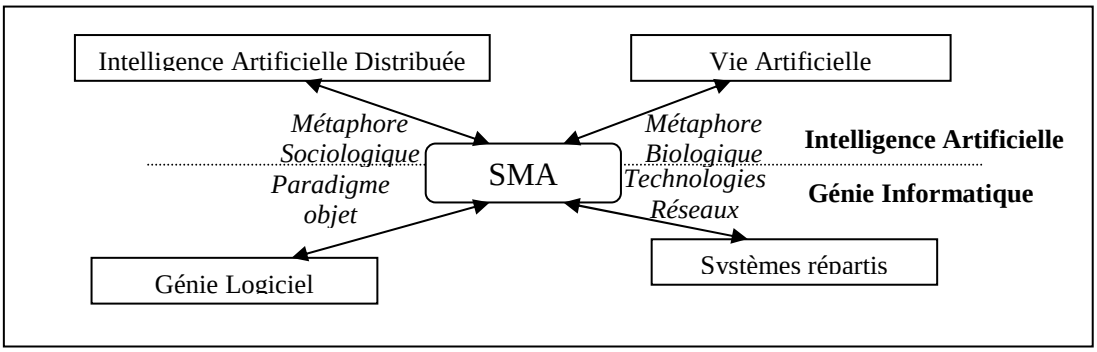


Figure III.3 : Positionnement des SMA dans l'IA

III.4.3.1 Définitions

Un système multi agents est un ensemble d'agents qui représentent des entités actives du système, ils sont situés dans un environnement, c'est-à-dire qu'à tout moment, on connaît leur position dans l'environnement. L'environnement comporte aussi des objets (entités passives pouvant être perçues, détruites, créées, ou modifiées par les agents) [Fer,1995]

Les agents sont unis par des relations, et disposent d'un ensemble d'opérations qui représentent leur capacité à percevoir leur environnement, par la production, la consommation, la transformation, et la manipulation des objets qui y sont situés.

L'environnement peut aussi être doté de lois dites "lois de l'univers" qui sont des opérateurs chargés de l'application des opérations décrites plus haut, ainsi que les réactions de l'environnement qui leur correspondent, donc un système multi-agents est essentiellement décentralisé, il permet de modéliser des systèmes hétérogènes, complexes, dynamiques, non linéaires et évolutifs, qui serviront à résoudre des problèmes, pour lesquels la somme des compétences des agents constitutifs ne le permet pas.

De la cohabitation et la coopération des agents relativement autonomes dans un système multi-agents émerge l'intelligence, et des capacités d'actions supérieures à celles des

agents qui le composent la solution émerge des interactions entre les différents agents du système. L'approche voyelle, de Yves Damazeau en 1995, décompose un SMA en [Fer,1995] :

Agents A : Définit les modèles et architectures utilisées pour les agents.

Environnement E : représente le milieu où évoluent les agents.

Interactions I : Les interactions proviennent de la mise en relation dynamique de plusieurs agents par le biais d'un ensemble d'actions réciproques. Il existe plusieurs types d'interactions, qui dépendent de trois paramètres que sont les buts, les ressources et les compétences. Cette partie regroupe les infrastructures, les langages et les différents protocoles d'interaction entre les agents.

Organisation O : structure les agents, leurs hiérarchies et leurs relations, les organisations constituent à la fois le support et la manière dont se passent les inter-relations entre les agents, c'est-à-dire dont sont réparties les tâches, les informations, les ressources et la coordination d'actions. L'approche "vowels" a deux principes :

- *Approche déclarative* : SMA=Agents+Environnement+Interactions+Organisation
- *Approche fonctionnelle* : La fonctionnalité d'un SMA s'exprime par la somme des fonctionnalités individuelles de ses agents, et de leur fonctionnalité collective.

Fonction(SMA)=(∑ Fonction(Agents) + Fonction collective)

III.4.3.2 La communication dans les SMA

Dans les systèmes multi-agents, la communication permet aux agents de s'envoyer (action) et de recevoir (perception) des messages et de les comprendre. L'environnement offre l'infrastructure de communication incluant des mécanismes d'interaction et des protocoles de communication. un modèle de communication doit répondre à [Maz,2001] :

- Pourquoi *communiquer* : La communication permet de mettre en œuvre l'interaction, par conséquent elle permet de coordonner et coopérer les actions des agents.
- Quand *communiquer* : Les agents sont confrontés à des situations où ils ont besoins d'interagir, ce qui revient à bien identifier les situations de communication.
- Avec *qui communiquer* : Lae communication est-t-elle diffusée à tous les agents, ou destinée à un ensemble restreint d'agents, ce choix dépend du voisinage de l'agent, et des connaissances qu'il possède sur les autres agents.
- *Quoi communiquer* : La communication porte sur des croyances, des intentions, et des tâches.
- *Comment communiquer* : La communication est mise en œuvre dans un langage compréhensible et commun à tous les agents.

a. Communication par envoi de messages

Dans ce type de communication les agents possèdent une représentation de l'environnement, un agent envoie donc des messages aux autres agents, selon divers protocoles et à base d'un langage commun. On distingue deux modes de transmission qui sont :

- Mode point à point, l'émetteur du message connait l'adresse des agents destinataires.
-Mode par diffusion, le message est diffusé à tous les agents du système.

b. Communication par partage de mémoire

La technique du tableau noir (Blackboard), en intelligence artificielle est utilisée pour spécifier une mémoire partagée [Jar,2002]. Pour les systèmes multi agent un tableau noir est une structure de données partagée entre divers agents qui l'utilisent pour écrire des messages, déposer des résultats de calculs et de solutions, obtenir des informations sur l'état d'un problème [Wei,1999].

c. Langages de communication

Les langages de communication se focalisent essentiellement sur la manière de décrire exhaustivement des actes de communication d'un point de vue syntaxique et sémantique, supportant un langage de représentation des connaissances

Le langage de communication KQML (Knowledge Query and Manipulation Language) est fondé sur la théorie d'actes de langage, son but est de permettre aux agents cognitifs de coopérer, son principe repose sur la séparation de la sémantique liée au protocole de communication (Indépendante du domaine d'application) de la sémantique liée au contenu des messages (dépendante du domaine d'application).

Un message KQML contient les informations nécessaires à sa compréhension

Le langage KIF (Knowledge Interchange Format) [Wei,1999] est un langage logique, qui a pour objectif de définir un format standard de représentation des connaissances manipulées par les agents. Le format ainsi défini doit être suffisamment générique pour permettre de développer des applications dans différents domaines, bien précis pour éviter toute interprétation ambiguë.

Le langage FIPA-ACL (FIPA Agent communication Language) [Woo,2002] de la « *Foundation for Intelligent Physical Agents"* doit spécifier des standards visant à assurer l'interopérabilité des applications et des services basée sur la paradigme agents. Le centre de cette initiative a été le développement de ACL, un langage de communication standard qui

comme KQML se base sur la théorie des actes de langages et s'appuie sur la définition de deux ensembles : [Maz,2001]

- Un ensemble d'actes de communication primitifs, peuvent être composés pour donner d'autres actes, Figure III.4.
- Un ensemble de messages prédéfinis, que tous las agents peuvent comprendre.

Les spécifications de FIPA-ACL décrivent chaque acte communicatif par une sémantique formelle basée sur les logiques modales et qui exprime des effets sur les attitudes mentales des agents expéditeurs et récepteurs.

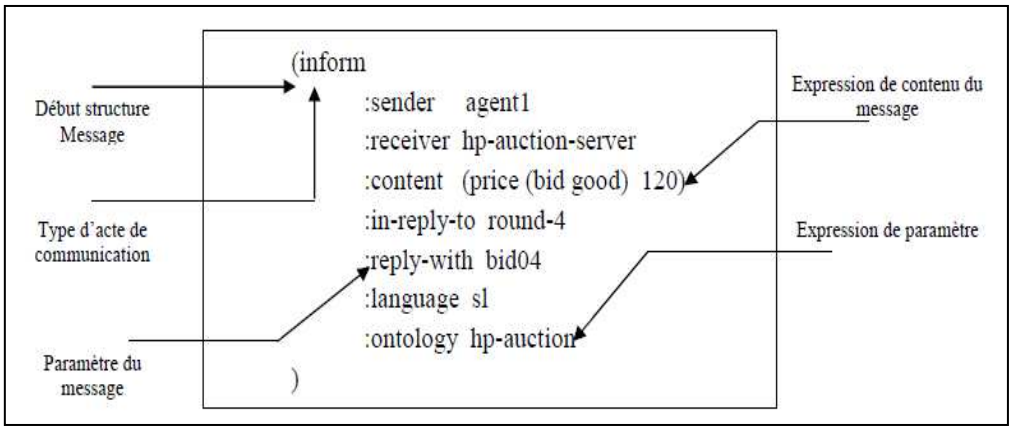


Figure III.4 : Structure d'une performative FIPA [Maz,2001]

III.4.3.3 Protocoles d'interaction dans les SMA

Un protocole d'interaction gère les échanges de messages entre agents, il existe différents types de protocole dépendant du système envisagé, en général on doit considérer les suivants aspects :

- Déterminer les buts partagés.
- Déterminer les tâches communes.
- Eviter les conflits inutiles.
- Partager les connaissances entre agents.

Les concepts de coopération et de coordination sont les formes d'interaction les plus utilisées pour réaliser conjointement des objectifs communs. Chaque agent contrôle ses propres actions, et gère ses interactions avec les autres agents du système [Wei,1999].

a. Interaction

Pour Ferber [Fer,1995], une interaction signifie la mise en relation dynamique d'un ensemble d'agents qui s'expriment par une série d'actions, ayant une influence sur le comportement futur des agents. Pour un agent l'interaction est à la fois la source de sa puissance et l'origine de ses problèmes: «On appellera situation d'interaction un ensemble de

comportements résultant du regroupement d'agents qui doivent agir pour satisfaire leurs objectifs en tenant compte des ressources plus au moins limitées dont ils disposent et de leurs compétences individuelles »

Les interactions sont généralement présentées sous forme d'exécution d'actions dans un système, qui a pour effet de modifier le comportement d'autres agents, permettant au système d'évoluer vers un but global en adoptant un comportement intelligent.

b. Coordination

Les agents communiquent pour mieux atteindre les buts du système dans le quel ils opèrent, ou les buts qu'ils se sont fixés [Wei,1999]. Les actions des agents doivent être coordonnées car :

- Il peut y avoir des dépendances ou des interférences entre les actions.
- Le besoin de maintenir des contraintes globales.
-Un agent seul, n'est pas suffisamment compétent, et ne dispose pas de toutes les ressources pour atteindre les buts du système. La coordination impose donc le partage de l'environnement, le degré de coordination indique le niveau des résolutions collectives recherchées (partage optimum de ressources, évitement de blocages, maintien de certaines propriétés…), selon la nature des agents on a plusieurs types de coordinations comme illustré par la Figure III.5.

En 1996 Jennings, avait défini le processus de coordination: *« The process by which an agent reasons about its local actions and the (anticipated) actions of others to try and ensure that the community acts in a coherent manner »*

La coordination est ainsi définie comme une interaction associée à une communication, et est liée aux capacités cognitives des agents.

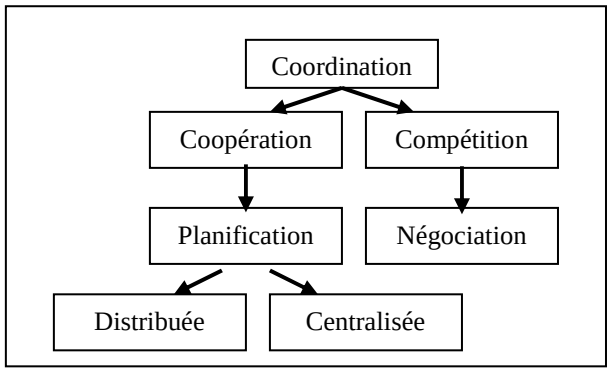


Figure III.5 Taxonomie de coordination [Wei,1999]

c. Protocole de coopération

Pour qu'un système multi-agent puisse parvenir à accomplir des tâches complexes dépassant une coordination entre agents réactifs, ils doivent être dotés d'attitudes propositionnelles comme les croyances, les désirs et les intentions envers leur environnement, en particulier ils doivent avoir des buts pour établir des comportements plus élaborés. Il existe une coopération entre agents dans un environnement "E" si et seulement si :

-Les agents interagissent.

- Ils sont dotés de buts.

Dans la définition ci-dessus, il n'est pas fait appel aux notions de travail commun, qui se rapportent à la collaboration, l'aspect d'une coopération entre agents apparaît au niveau du comportement global du système. Les agents coopératifs possèdent le modèle de leur environnement, donc des autres agents leur permettant de se considérer les uns et les autres, c'est cette perception qui permet à l'agent de déterminer, adopter un comportement, et procéder à la mise à jour des modèles des agents dont il dispose. On peut donc considérer qu'il y a des transferts d'information entre des agents coopérant au-delà de simples interactions.

La collaboration caractérise un ensemble d'agents qui coopèrent, pour atteindre les mêmes buts, dans ce cas de systèmes collaboratifs.

d. Planification

Dans les approches de l'intelligence artificielle classique, et avant que l'on s'intéresse aux aspects distribués la planification consistait en un processus (un seul agent planificateur) qui recherche un chemin entre un état initial et un état final, dans un espace d'états (environnement).

Une planification est dite statique lorsque l'environnement ne change que par les actions d'un planificateur, qui connaît tous les états possibles de l'environnement et les événements qu'il peut générer pour le modifier. En conséquence si d'autres événements inattendus peuvent changer l'état de l'environnement, c'est une planification dynamique, dans ce cas l'agent doit observer les conséquences de ces événements sur le déroulement de son plan, et si nécessaire il doit reconsidérer l'exécution de son plan [Stu,1995].

III.4.4 Systèmes de recherche sémantique d'informations

Dans cette section nous exposerons quelques architectures et fonctionnalités de systèmes de recherches sémantique d'informations avec ou sans le paradigme agents. Cet état de l'art, se veut être un repère pour notre approche par rapport aux travaux existants.

Notamment il sera question de repérer les outils de conceptualisations communs, des problèmes majeurs rencontrés lors des développements, les solutions envisagées. Les systèmes multi-agents, sont de nos jours largement utilisés pour concevoir des systèmes de gestion des informations hétérogènes et distribués. Ces entités autonomes coopèrent à l'échange de ressources documentaires, à la recherche d'informations, et utilisent les concepts de négociation, de mobilité pour fournir l'information requise aux utilisateurs potentiels.

Les recherches massives engagées ces dernières années, ont abouti à la réalisation de plusieurs systèmes qui considère la recherche sémantique d'informations sous divers angles, qui dépendent des buts envisagés, des stratégies et modèles utilisées.

La plus part des modèles proposés partagent des aspects communs qui se concentrent sur les suivants points:

- Des outils d'annotation
- Des représentations sémantiques des connaissances de domaines.
- Des stratégies de recherche (sémantiques, hybrides, etc.) basées ou non sur les agents.
- Des algorithmes de classements des résultats obtenus selon leur pertinence.

C'est ainsi que [Cas,2007] propose une recherche sémantique basée ontologie et utilisant une adaptation du modèle classique d'espace vectoriel. Des algorithmes de calcul de poids de termes et de classement sont aussi décris pour combiner ce type de recherche avec une recherche basée mots clés pour dépasser les défis liés aux limites d'expressivité des ontologies.

« *Implicit* » est un système de recherche d'information basé SMA, il expérimente un démarche pour satisfaire en informations des groupes de personnes ayant des buts et intérêts similaires [Bir,2012].

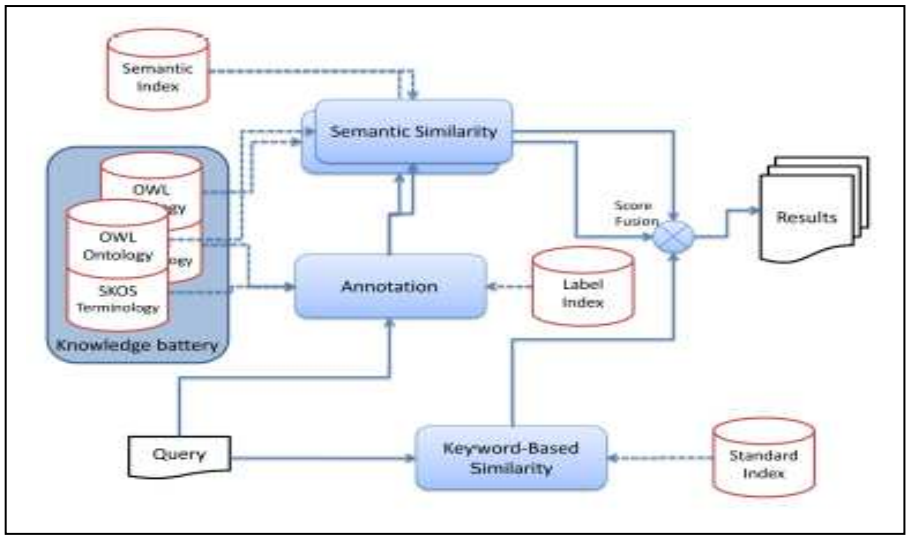


Figure III.6 : Processus de recherche

Les agents observent le comportement de leurs utilisateurs et apprennent la culture de la communauté pour certains intérêts spécifiques, ils facilitent le partage des connaissances des liens pertinents pour la communauté par des recommandations. Dans cette même orientation, un système multi agents *« Mars »* offre pour chaque utilisateur un agent, les agents interagissent pour s'échanger leurs connaissances via une ontologie.

Pour [Bus,2013], nous avons une description complète de « YaSemIR » est un système de recherche sémantique d'information basé « Lucene ». les travaux de En résumé ce système utilise une ou plusieurs ontologies OWL et pour chaque ontologie une terminologie pour indexer sémantiquement une collection documentaire.

Une terminologie sert à annoter les concepts des documents alors qu'une ontologie procure une taxonomie pour l'expansion de ces concepts par leurs subsumant. Les composants de cette architecture sont « Figure III.6 » :

- Un module d'annotation, qui identifie les occurrences des concepts dans les documents/requêtes.
- Un module d'indexation qui crée un index standard basé Lucene.
- Un module de classement, basé occurrence des mots et les correspondances avec les concepts.
- Une ou plusieurs ontologies, et des terminologies.

La première étape consiste à créer l'index label, qui fait correspondre les labels aux concepts, ensuite le module d'indexation sémantique, agit comme parseur de documents, chaque document est par traité par le module d'annotation qui atteint l'index label et retourne un ensemble de concepts $C_D=\{C_1,C_2,\ldots,C_n\}$.

Cet ensemble subi ensuite une expansion par les subsumants définis dans l'ontologie, le résultat est un ensemble $A_D=\{C_{n+1},C_{n+2},\ldots,C_m\}$.

Enfin, chaque document est annoté par $C_D \cup A_D$

Une cession de recherche est déclenchée par une requête en langage naturel, elle est analysée et traitée par le module d'annotation pour extraire l'ensemble $Q=\{C_1,C_2,\ldots,C_k\}$, qui sera recherché dans l'index sémantique. L'évaluation des similarités entre les documents retournés et la requête est accomplie par l'application de la formule de Wu & Palmer (section III.2.1 équation III.1. Pour faire le lien de notre travail avec l'architecture de ce système, on retient qu'il reprend une partie des fonctionnalités de notre proposition qui concerne aussi le processus d'expansion de requête mais de manière plus large à l'aspect de cette description.

Une autre architecture utilise une ontologie de domaine pour dépasser le problème d'hétérogénéité de l'information sur le web, [Bro,2005]**.** L'architecture présentée propose de:

1. Fournir aux utilisateurs les outils convenables pour annoter leurs pages web.
2. Déployer des applications multi-agents pour faciliter l'interrogation basée concepts.

Un logiciel est utilisé pour annoter le contenu de sites web par des triplets RDF, ce sont des instances OWL appropriées à l'ontologie de domaine. Dans ce modèle cognitif, une session de recherche se déroule comme suit, « Figure III.7 »

1- Un *agent coordinateur* demande à des *agents de domaines* de crawler le web et mettre à jour les ontologies de domaines d'intérêt, rechercher les sites annotés en respect à ces ontologies.

2- *Les agents de domaines* recherchent les ontologies, téléchargent leurs modèles et les transmettent à *un agent Jena*, ce dernier crée un modèle jena de l'ontologie et l'enregistre dans une base de données RDF.

3- *Les agents de domaines* crawlent le web et recherchent les pages annotées RDF, avec des espaces de nommage qui correspondent à leurs spécifiques ontologies. Ils extraient les annotations RDF de ces pages et les transmettent à l'*agent Jena.*

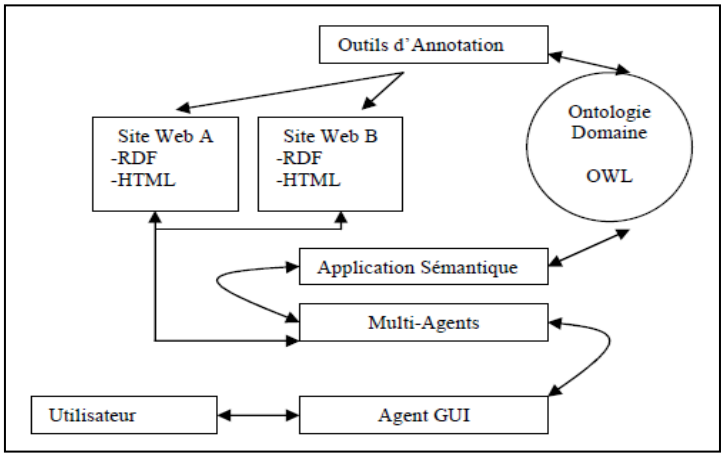


Figure III.7 : Architecture basée SMA pour la recherche sémantique

4- *L'agent Jena* enregistre les annotations reçues dans le modèle de l'ontologie qui relève du domaine, une copie du modèle étant auparavant enregistré dans la base de données RDF

5- L'utilisateur soumet une requête à un *agent interface utilisateur* « GUI », ce dernier la convertit au format XML et l'envoie à *l'agent interface.*

6- *L'agent interface* reçoit la requête, transforme les spécifications de tâches en spécifications techniques qu'il passe à *l'agent coordinateur.*

7- *L'agent coordinateur* divise la tâche en sous tâche, établit un plan et alloue les sous taches aux *agents de domaines.*

8- Les *agents de domaines*, formulent les solutions possibles, les convertissent en spécifications de requêtes qu'ils transmettent à l'agent Jena.
9- L'agent Jena convertit les spécifications de requêtes en requêtes RDQL, invoque le raisonneur « Racer » pour initier les requêtes en RDQL.
10- L'agent Jena retrouve les résultats par le raisonneur, les transmet aux agents de domaines pour le classement pour les présenter aux utilisateurs sous forme HTML.

Un travail similaire est présenté [Ker,2004], c'est une autre architecture multi-agents basée ontologies qui accède à différentes sources d'informations hétérogènes. Une collection d'agents coopératifs qui interagissent pour la spécification, le raffinement, la décomposition, le traitement, la classification, et la présentation des résultats associés à des requêtes interactives.

Les auteurs de l'approche, justifient l'usage du paradigme agent pour implémenter ce système de recherche par les propriétés d'autonomie, de proactivité et d'apprentissage adaptées pour ce type d'applications, de plus les agents du système ayant chacun une responsabilité locale coopèrent et communiquent via un ACL. Le système est conçu pour les objectifs suivants :

· Permettre aux utilisateurs d'effectuer des recherches sémantiques guidées par une ontologie exprimée en OWL.
· Raffiner les recherches par des feedback utilisateur.
· Accéder différentes sources d'informations hétérogènes.

L'architecture conceptuelle comporte aussi trois couches (utilisateur, gestion de connaissances et sources de données). Etant générique et modulaire, elle permet l'incorporation de nouvelles ontologies/sources d'informations.

Conclusion

Dans ce chapitre nous avons décris l'état de l'art associé aux aspects du processus sémantique de recherche d'information sur le web. Ces aspects s'articulent autour du concept de la sémantique des termes dans un texte, des concepts structurés dans une hiérarchie et des relations entre eux.

Nous avons en premier abordé en détail le processus d'annotation sémantique des documents. Le trait à été fait sur les composants de l'annotation sémantique, à savoir les ressources annotées et les métadonnées associées. De même, les systèmes et outils d'annotation et les méthodologies existantes ont pris part importante dans cette section. Il a été question de relater les systèmes développés, et les fondements théoriques des démarches suivies.

Nous avons donc sélectionné parmi les nombreux systèmes actuels, les plus communs dans la littérature, un bref expose pour chaque outils a été donné.

L'enchainement des concepts sémantiques nous a incité à évoquer les mesures de similarité sémantique entre concepts et par conséquent entre les documents et les requêtes. Nous avons dans cette perspective essayé de reprendre les principales mesures qui existent dans la littérature, en expliquant leurs théories, les structures sur lesquelles elles reposent et leurs formulations mathématiques proprement dites. Cette étude détaillée sera traduite dans le chapitre 4, par la description de notre mesure « LDSim » basé sur la méthode « edge-counting ».

Les systèmes multi-agents, étant partie intégrante dans notre travail, ont fait l'objet d'un état de l'art dans le quel nous nous sommes concentrés sur les caractéristiques intrinsèques des agents. En effet, après avoir passé l'essentiel des définitions d'un agent en tant qu'entité autonome, du paradigme des systèmes multi-agents et de leur positionnement dans le cadre des activités de l'intelligence artificielle distribuée. Nous avons jugé nécessaire de voir comment les agents d'un SMA interagissent pour accomplir les tâches dont ils ont la charge.

Dans cette orientation nous nous sommes intéressés aux principaux modes de communication, les langages de communication et les protocoles d'interaction. Ainsi, la coordination, la coopération et la négociation constituent les modes d'interaction entre agents ont été revues.

La dernière section de ce chapitre, nous l'avons consacré à la description du processus de recherche sémantique d'information basée ou non basée agents. De brèves descriptions de l'architecture de systèmes et des fonctionnalités ont été données, l'intérêt majeur est de localiser les insuffisances et tirer profits de leurs avantages pour tracer le cadre théorique de notre contribution qui fera l'objet du chapitre suivant.

CHAPITRE IV

Un Modèle de Raisonnements pour un Système de Recherche Sémantique d'Informations Sur le Web basé Agents.

IV.1 Introduction

La recherche sémantique est un processus complexe qui devrait être achevé sur plusieurs étapes ; par exemple nous avons un processus d'annotation sémantique, le traitement de la requête, et une importante phase qui consiste à classer les résultats obtenus pour en évaluer la pertinence. Dans cette orientation le web sémantique ne fonctionne plus seulement avec les relations d'hyperlien, il y a d'autres types de relations à retenir et qui lient ses différentes ressources. Dans cette approche, nous allons décrire une architecture basée agents pour la recherche sémantique d'information sur le web.

Nous allons donc nous intéresser à la conception d'un modèle de raisonnement qui permettra de prendre en charge le processus de recherche sémantique basée ontologies et utilisant le paradigme agent à travers lesquels émerge l'intelligence dans le processus de la recherche.

Ce raisonnement est projeté à travers les étapes du processus de recherche, allant de l'annotation sémantique où une première amélioration sera introduite et qui s'attache à la manière de délimiter les segments sémantiques dans un document textuel, pour mieux dégager la sémantique profonde enfoncée dans la structure documentaire. Le raisonnement se construit aussi lors du processus de classement de document par une originale mesure de similarité sémantique basée WordNet, qui s'approche de celle proposée par Wu et Palmer avec des propriétés souhaitables et plus tangibles [Nes 2013].

Le troisième pilier du raisonnement concerne l'hybridation de l'approche de recherche, d'une part nous proposons une recherche syntaxique-sémantique à travers un processus d'expansion de requête par des synonymes, des hyponymes et des hyperonymes. Ce processus devrait augmenter le rappel lorsque si nécessaire et permet de tenir compte des limites des représentations sémantiques qui sont généralement peu expressives. D'autre part et pour dépasser la manipulation syntaxique nous avons basé ce raisonnement sur la recherche d'une opération de projection du graphe de la requête dans le graphe de l'annotation rattaché au document. Ce processus constitue le noyau du raisonnement sémantique du modèle.

IV.2 Contexte théorique du modèle

Plusieurs formalismes de représentation des connaissances existent, les techniques de représentations conventionnelles (les bases de données relationnelles et les modèles orienté objets), les techniques d'intelligence artificielle (les formalismes logiques, les réseaux sémantiques, les graphes conceptuels, etc.) sont aujourd'hui largement utilisées.

Récemment, il a été reconnu dans les milieux académiques et universitaires, qu'aucune de ces techniques ne peut être appliquée avec plein succès pour représenter l'information contenue dans les documents en langage naturel. En effet, ces formalismes ne peuvent être retenus pour la conversion automatique des textes (documents en langage naturel, pages web, etc.) dans sous une forme structurée, sans perte d'informations.

Ainsi, il est devenu essentiel d'adopter une nouvelle vision pour structurer cette connaissance, le formalisme d'ontologie et les langages de description comme RDF(S) [Bri,2004], DAML + OIL et OWL [Gui,2004] et dans le cadre du web sémantique sont des modèles dotés de puissance structurelle et expressive permettant de représenter tout type de connaissances sous forme structurée et appropriée pour les traitements automatiques ultérieurs.

L'architecture de notre système inclut les composants suivants [Nes 2013] :

- La taxonomie WordNet.
- Une ontologie de domaine.
- Un module d'annotation sémantique.
- Une mesure de similarité sémantique.
- Un système multi-agents.

IV.3 Architecture du système

Nous allons dans ce qui suit décrire l'aspect structurel et fonctionnel de notre système.

IV.3.1 La taxonomie WordNet

WordNet est un thésaurus électronique, une référence lexicale largement utilisée pour l'anglais. Sa conception reposait sur des théories psycholinguistiques des mémoires lexicales humaines. Ce lexique comporte environ 180000 termes organisés dans 117597 synsets qui regroupent des noms, des verbes, des adjectifs et des adverbes organisés dans des ensembles de synonymes appelés « synsets », représentant les concepts lexicaux sous-jacents.

Cette ontologie a plusieurs définitions relatives aux besoins et aux objectifs utilisateurs, son origine remonte aux années 1986 à l'Université de Princeton [Fel,2010] où elle continue d'être développée et maintenue. Le psycholinguiste George A. Miller, était inspiré par les travaux expérimentaux en intelligence artificielle qui essayaient de comprendre la mémoire sémantique de l'humain. Partant du fait que les locuteurs possèdent des connaissances sur des dizaines de milliers de mots et les concepts exprimés par ces mots, il semblait raisonnable de penser à des mécanismes efficaces pour le stockage et l'accès aux concepts, Figure IV.1.

Ainsi, le modèle Collins et Quillian [San,2012] a proposé une structure hiérarchique des concepts, où les concepts plus spécifiques héritent des informations de leur super-ordonnée, les concepts plus généraux; donc seulement les connaissances particulières à des concepts plus spécifiques doivent être stockées. WordNet offre deux services distincts :

- Un vocabulaire décrivant les différents sens des mots.
- Une ontologie décrivant les relations sémantiques entre les mots.

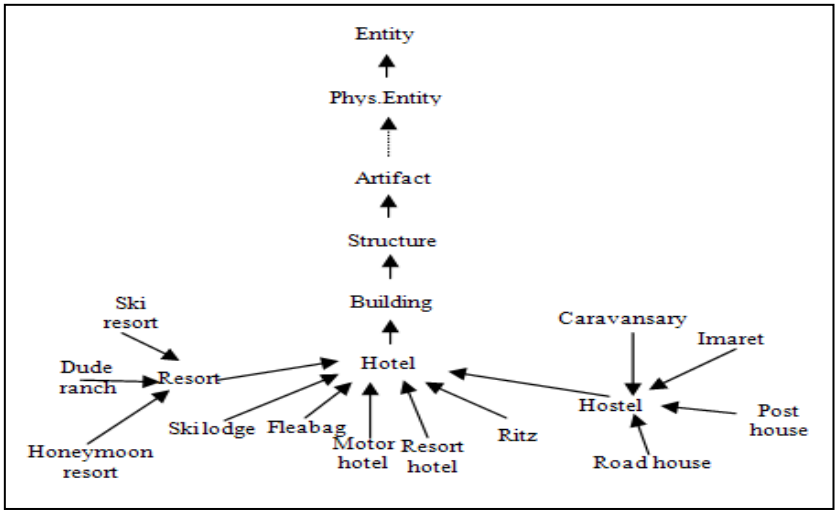


Figure IV.1 : Sous hiérarchie de WordNet relative au concept « Hôtel »

Les synsets sont connectés en haut / bas de la hiérarchie par différents types de relations, la plupart sont des relations « Is-a » pour les « hyperonyme / hyponyme », et des relations « Part-of », pour les « holonyme / méronyme »

Dans le cadre de cette thèse, nous utilisons uniquement les relations de« synonymes / hyperonymes/hyponymes », et les représentations de termes basée sur les noms d'objets qui sont généralement reconnu pour être les formes les plus représentatives de la sémantique d'un langage. Les noms extraits des documents et des requêtes, présentent certaines similarités, plusieurs méthodes de mesures de similarité sémantique qui ont été testées, essentiellement il ya deux catégories:

- Les méthodes basées sur la structure de l'ontologie.
- Les méthodes basées sur le contenu informatif du concept.

WordNet supporte des méthodologies d'évaluation pour des algorithmes d'expansion de WordNet, la tâche principale consiste à identifier pour un mot tous les sens possibles et toutes les instances des relations lexico-sémantiques auxquelles il participe [Brd,2012].

IV.3.2 L'ontologie de domaine

Le concept d'ontologie existe depuis longtemps, il est d'origine philosophie, en conséquence plusieurs définitions de l'ontologie ont été proposées. La plus communément utilisée est celle de Gruber «une ontologie est une spécification explicite d'une conceptualisation ». Dans les traitements du langage naturel, nous définissons une ontologie comme une déclaration formelle qui associe le nom d'entités de l'univers du discours (classes, relations et fonctions) avec le contenu des documents, en outre et au moyen de ce formalisme nous avons un ensemble d'axiomes qui limitent l'interprétation des ces entités.

Pourtant, la spécification du domaine pour la conceptualisation de l'ontologie est toujours une tâche complexe [Yu,2011]. D'une part, l'ontologie doit tenir sur les normes standards du domaine pour garantir l'interopérabilité, et d'autre part, elle devrait être appliquée dans des environnements hétérogènes. Le principal objectif de l'ontologie est donc de permettre la communication entre des systèmes informatiques, d'une façon qui soit indépendante des architectures et des domaines d'applications [Cha,2010].

Dans notre cas, ce modèle servira à la construction de base de connaissances d'annotation sémantique en définissant par un vocabulaire contrôlé, les entités du domaine et leurs relations sémantiques. La collection des technologies web sémantique, par exemple XML, RDF, OWL, RDQL, SPARQL, fournissent l'environnement adéquat où une application logicielle peut interroger ces données et effectuer des inférences sémantiques.

IV.4 Processus d'annotation sémantique

L'objectif principal est de construire pour un document son index sémantique basé ontologie de domaine, ce processus peut être divisé en étapes :

IV.4.1 Extraction de termes

Est une opération qui regroupe plusieurs tâches dites de prétraitement, telles que la tokenization [San,2012], la suppression des mots vides, le marquage grammatical (POS tagging), qui consiste en un marquage des mots du texte par leurs classes ou catégorie lexicale, et la lemmatisation des termes pour extraire les caractéristiques importantes du texte. Cette étape est effectuée par l'intégration d'un outil d'indexation approprié tel que LUCENE.

L'hypothèse sous-jacente est que les mots co-occurrents dans un segment (document ou paragraphe) sont susceptibles d'être les plus représentatifs [Hua,2012]. Le résultat de cette étape est un index plat que l'on représentera par une matrice qui définit une relation d'ordre sur

les termes et les segments. Les lignes correspondent aux segments du document et les colonnes représentent les termes retenus pour l'indexation sous forme de couples ($terme_i$, $poids_i$).

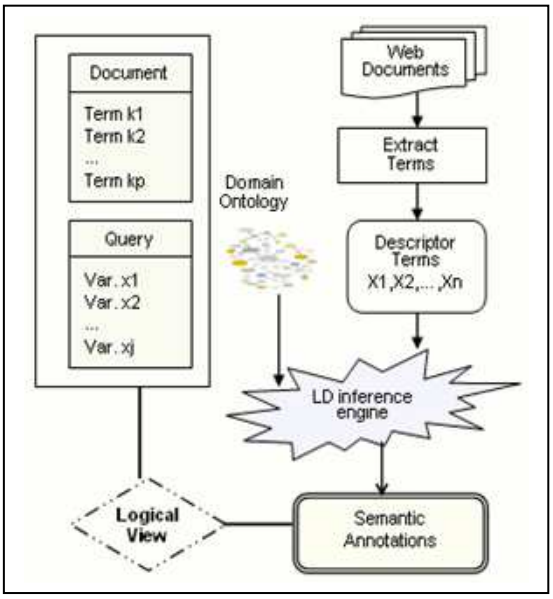


Figure IV.2 : Processus d'annotation sémantique

Dans l'étape suivante, nous proposons de segmenter le texte en plusieurs portions sémantiques (sous-thèmes) porteurs de sens. L'objectif de cette segmentation vise d'un coté à permettre l'annotation des portions profondes dans le document, d'un autre côté l'annotation par sous-thème permettra une description plus précise et simple qui simplifiera les mécanismes d'inférence appliqués aux sous-thèmes lors d'une session de recherche.

IV.4.2 Segmentation sémantique

La représentation classique d'un texte ne fournit aucune notion sur la sémantique des phrases, ni sur les thèmes, ni sur les mots qui la composent. La complexité se situe à différents niveaux et s'exprime par des concepts tels que la synonymie, la polysémie, d'ailleurs le fait d'isoler les mots d'une phrase n'est pas évident, sujet à plusieurs ambiguïtés.

La segmentation d'un texte en fragments thématiques est le processus de division du texte en portions sémantiquement différents, ces portions peuvent être des phrases, des paragraphes, et des fragments. Le problème n'est pas trivial, car dépendant de la langue et des marqueurs explicites qui délimitent ces unités recherchées.

La segmentation dans le cadre de notre approche consiste à repérer dans des le texte à annoter les limites des segments thématiques, dans la littérature, nous trouvons plusieurs approches qui accomplissent ce processus comme les méthodes basées similarités ou statistiques, les méthodes dites graphiques, des méthodes de chaînes lexicales et aussi des méthodes basée sur l'apprentissage. [Lab,2007]

Nous utiliserons dans le cadre de ce travail une nouvelle méthode basée sur le calcul de similarités entre les termes des paragraphes, cette méthode a fait l'objet d'une présentation à une conférence internationale ICMCS en 2012, [Nes 2012a]. Ce choix est inspiré essentiellement par l'efficacité de ces méthodes à détecter les frontières des fragments thématiques du texte. Aussi, l'idée de base qui guide notre démarche d'annotation sémantique est de créer pour chaque segment un graphe conceptuel qui exprime les connaissances d'annotation du segment, de ce fait la taille d'un bloc thématique ne nécessite pas l'usage de vecteurs de grades dimensions ce qui facilitent énormément les traitements.

L'hiérarchie WordNet est utilisée lors de cette phase pour calculer les similarités entre les termes repérés dans des phrases (paragraphes) adjacents pour ensuite évaluer leurs similarités sémantiques.

Durant la première étape d'extraction de termes, un paragraphe est représentée par l'ensemble des poids des termes qui le composent, c'est donc un vecteur définit par :

$S : \{Ph\} \rightarrow R^p$; où {Ph}: l'ensemble de paragraphes.

$S(Ph_i)=V_{Phi} =(x_1,x_2,\ldots,x_p)$, xi: poids du terme i.

L'idée, repose sur un calcul de similarité entre les vecteur Ph_1, Ph_2, et un vecteur Vsim. Les vecteurs poids Ph_1 et Ph_2 représentent respectivement les paragraphes Ph_1 et Ph_2.

Le vecteur Vsim que l'on devra construire, repose sur la notion mathématique de somme de vecteurs, c'est-à-dire nous considérons Vsim comme la résultante de Ph_1 et Ph_2.

$$\vec{Vsim}=\vec{Ph_1}+\vec{Ph_2} \qquad \text{(IV.1)}$$

Nous entendons ainsi exprimer l'éloignement ou le rapprochement des sens des deux paragraphes (unité de traitement sémantique), pour cela nous calculons les composantes du vecteur Vsim par la démarche suivante :

Les différents termes extraits des deux segments sont classés par une relation d'ordre.

Soit $Ph_1=(x_1,x_2,\ldots,x_i,\ldots,x_p)$ et $Ph_2=(y_1,y_2,\ldots,y_j,\ldots y_q)$. Les phrases sont représentées par les vecteurs poids des termes qui les composent.

Pour tout terme x_i de la phrase Ph_1, qui n'existe pas dans Ph_2, on calcul sa similarité avec l'ensemble de termes compostant la phrase Ph_2, ensuite on retiendra la similarité maximale que l'on multiplie par le poids du terme x_i, le résultat est considéré être la composante « i » du vecteur Vsim.

Le processus est répété pour tous les termes x_i de la phrase Ph_1 ; ensuite le même procédé est appliques aux termes y_j de la phrase Ph_2.

Cette démarche avantage la segmentation en segments de tailles réduites, c'est-à-dire des segments aussi courts possibles pour simplifier la construction d'arbre syntaxique associés au segment défini comme il montré sur la Figure IV.3. Les similarités entre concepts sont calculées selon une mesure que nous avons développée et qui sera exposé par la suite [Nes 2013].

La formulation mathématique de calcul des composantes du vecteur « Vsim » est :

Si x_i #0 alors
$Vsim_i$=Max(Sim(Tx_i ;Ty_k) y_k#0 ,k=1..q)*x_i
Si x_i =0 alors /* y_i #0
$Vsim_i$=Max(Sim(Ty_i ;Tx_k) x_k#0 ,k=1..p)*y_i

L'algorithme de construction du vecteur Vsim que nous avons proposé [Nes 2012a] et [Nes2012b] est le suivant:

Debut
Etablir une relation d'ordre sur les termes x_i et y_j / i=1 ..p et j=1..q
Pour i= 1 à P *faire*
Si x_i #0 *alors*
Pour k=1 à q *faire*
Calculer (Sim(Tx_i ;Ty_k)) *Fin*
Vsim(i)=Max(Sim(Tx_i ;Ty_k))*x_i
Sinon /*x_i =0 et y_i #0
Pour k=1 à P *faire*
Calculer (Sim(Ty_i ;Tx_k)) *Fin*
Vsim(i)=Min(Sim(Ty_i ;Tx_k))*y_i
Fin
Fin
Fin

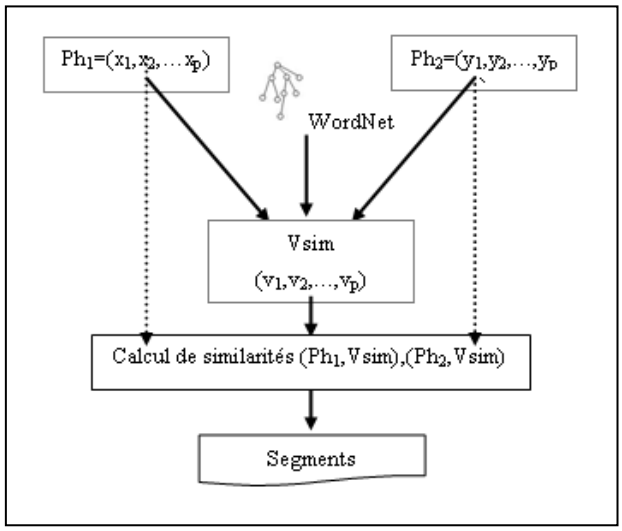


Figure IV.3 : Délimitation de segments sémantiques

Plusieurs mesures de similarités entre vecteurs existent, dans le contexte de ce travail nous utilisons la mesure du cosinus qui est la mieux adaptée aux traitements documentaires. La

détection des frontières des segments thématiques se fera en calculant les cosinus des angles que forment les vecteurs X,Y associés aux phrases éventuellement aux segments Ph_1 et Ph_2 , et le vecteur Vsim construit comme indiqué ci-dessus.

Soit α l'angle que forment les vecteurs X et Vsim

Soit β l'angle entre les vecteurs Y et Vsim.

En appliquant la formule du cosinus nous obtenons, Figure IV.4

$$Sim(X,Vsim)=Cos(\alpha)=\frac{\overrightarrow{X}.\overrightarrow{Vsim}}{|X|*|Vsim|} \qquad (IV.2)$$

$$Sim(Y,Vsim)=Cos(\beta)=\frac{\overrightarrow{Y}.\overrightarrow{Vsim}}{|Y|*|Vsim|} \qquad (IV.3)$$

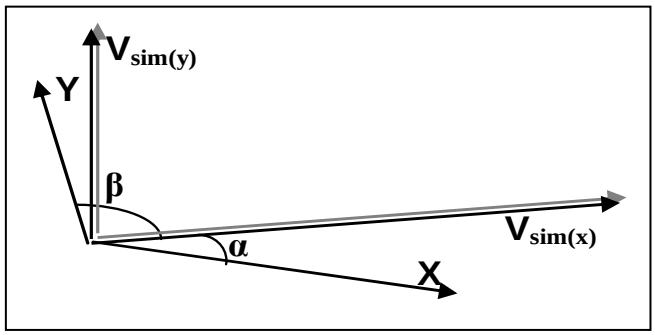


Figure IV.4 : Les vecteurs X, Y et Vsim

La comparaison de ces mesures de similarité nous permettra de déduire le rapprochement ou l'éloignement des vecteurs X et Y par rapport à leur résultante Vsim.

- Cos(α) > Cos(β) : dans ce cas α < β, le vecteur X est plus proche du vecteur Vsim que le vecteur Y, la sémantique de la résultante est emportée par la sémantique du vecteur X. Dans ce cas nous avons une continuation de la sémantique de X à travers le vecteur Y, les paragraphes Ph_1 et Ph_2 seront fusionnées dans un seul segment sémantique [Nes 2012a] [Nes 2012b].
- Cos(α) <= Cos(β) : c'est-à-dire α>=β, le vecteur Y est plus proche du vecteur Vsim que le vecteur X, la sémantique de la résultante est plus proche de la phrase représentée par Y, dans ce cas nous avons un délimiteur de segment sémantique et nous devons donc séparer les vecteurs X et Y, parce que Y s'éloigne (sémantiquement) et suffisamment de X.

Une fois le texte analysé et les segments sémantiques délimités, l'étape suivante du processus consistera à construire pour chaque segment un arbre de décomposition syntaxique.

IV.4.3 Analyse syntaxique des segments sémantiques

Cette étape a fait l'objet de plusieurs recherche en NLP, les méthodes qui ont été proposées appartiennent à trois grandes familles : les méthodes basées sur l'étude statistique, celles qui exploitent les contextes syntaxiques et des méthodes qui utilisent des marqueurs.

Les implémentations de ces méthodes sont des outils d'analyse et de construction d'arbre syntaxique comme (SygFran avec l'application SYGMART, Lexter, Connexor, Fips …). Ces outils produisent un arbre de décomposition syntaxique d'un texte soumis à l'analyseur. Rappelons que notre objectif est de produire un ensemble d'annotations sémantiques structurées que nous exprimons dans le formalisme des graphes conceptuels.

Parmi les outils qui effectuent une décomposition syntaxique d'un texte écris dans plusieurs langues comme le français et l'anglais nous avons *Connexor*, un analyseur syntaxique qui transforme une phrase écrite en langage naturel en un arbre syntaxique dont les nœuds sont les mots de la phrase et les arcs représentent les relations syntaxiques entre les mots.

IV.4.4 Génération d'annotations sémantiques

L'annotation sémantique des informations sur le web par les métadonnées RDF est un processus clé dans la plupart des applications du web sémantique, en particulier pour les systèmes de recherches sémantiques d'informations. L'annotation est le contexte d'instanciation des classes de l'ontologie attachées à la ressource documentaire, elle nécessite un traitement linguistique profond. Le but est de permettre aux agents logiciels de raisonner et retrouver des documents situés dans le voisinage sémantique d'une requête utilisateur.

Il y a différents types d'annotations qui se distinguent par la nature de l'élément ontologique auquel elles se rattachent. Certains mots et expressions renvoient à des instances, des entités nommées, et des concepts. Certains termes dénotent des rôles conceptuels, alors que certains fragments textuels renvoient à des relations entre instances, et des axiomes ontologiques qui expriment des relations de subsomption entre les concepts de l'ontologie [Yue,2009].

Notre choix de formaliser les connaissances d'annotations par un graphe conceptuel est motivé par le fait qu'un graphe conceptuel est une forme expressive de la logique qui a été largement utilisés dans les représentations sémantiques liées aux traitements des langages naturels. Aussi, un graphe conceptuel peut être traduit par un formalisme logique simple, rigoureux et formel et qui s'apprêtent mieux aux raisonnements.

A partir des fonctions grammaticales des mots d'un segment thématique, et des relations syntaxiques entre les mots qui sont identifiées par le graphe de décomposition syntaxique, et en utilisant un ensemble de règles de transformations, on peut générer l'ensemble de graphes qui décrivent les portions du document et dont chacun correspond à la décomposition syntaxique d'un fragment délimité auparavant « Figure IV.5 » [Nes 2012a]

Ces règles peuvent être décrites sous forme de prémisses et conclusions, la partie condition exprime donc une partie du graphe syntaxique et la partie conclusion définie l'élément du graphe conceptuel correspondant. Les relations qui composeront le graphe conceptuel résultat appartiennent à l'ensemble de relations prédéfinies utilisées dans ce formalisme, notamment nous avons : «AGNT» (agent), «PTNT» (Objet), «ATTR» (attribut), «MANR»(manière), «NUM»(nombre), « MOD »(modifier), «LOC»(Localisation), «TMP»(Temps), «DUR»(Durée), «CAUS»(Cause), «CHCR »(Caractéristique), « ISA» etc.

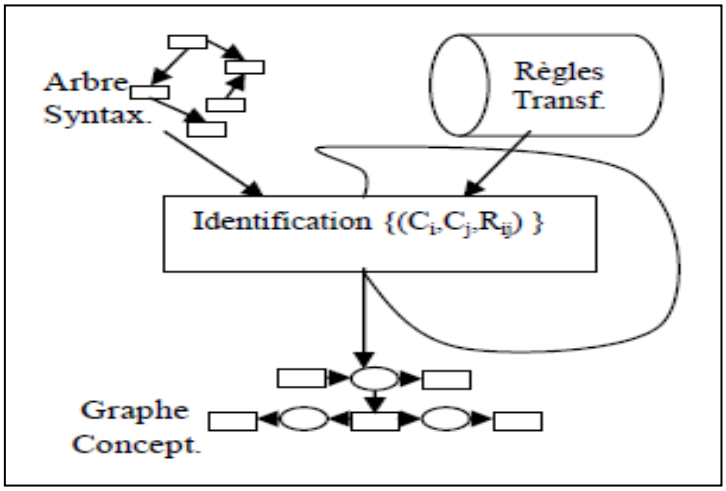


Figure IV.5 : Construction du graphe conceptuel [Nes 2012a]

Une règle de transformation peut avoir la forme suivante :

Si (sujet-verbe-complément) alors

Verbe → Concept : C1

Sujet → relation : Agnt

Complément → Concept : C2

L'idée générale sur laquelle s'appuie cette transformation est de considérer les verbes, les noms, les adjectifs comme des concepts, et leurs rôles grammaticaux comme des relations. On procède par construction de sous graphes associés aux portions de la phrase ou du segment, ensuite on applique une opération de jointure sur les sous graphes conceptuels obtenus comme nous l'avons décris dans [Nes 2012a] [Nes 2012b].

La composition de l'ensemble de règles de transformation de l'arbre syntaxique en un graphe conceptuel, est une étape décisive et fondamentale dans le processus de modélisation des connaissances d'annotation des segments.

IV.5 Architecture du système Multi-Agents

Le but principal de l'utilisation du paradigme agent est d'améliorer les applications liées à la recherche d'informations sur le web. En particulier, l'objectif est de concevoir et développer un système multi-agent qui prend en charge [Nes 2013]:

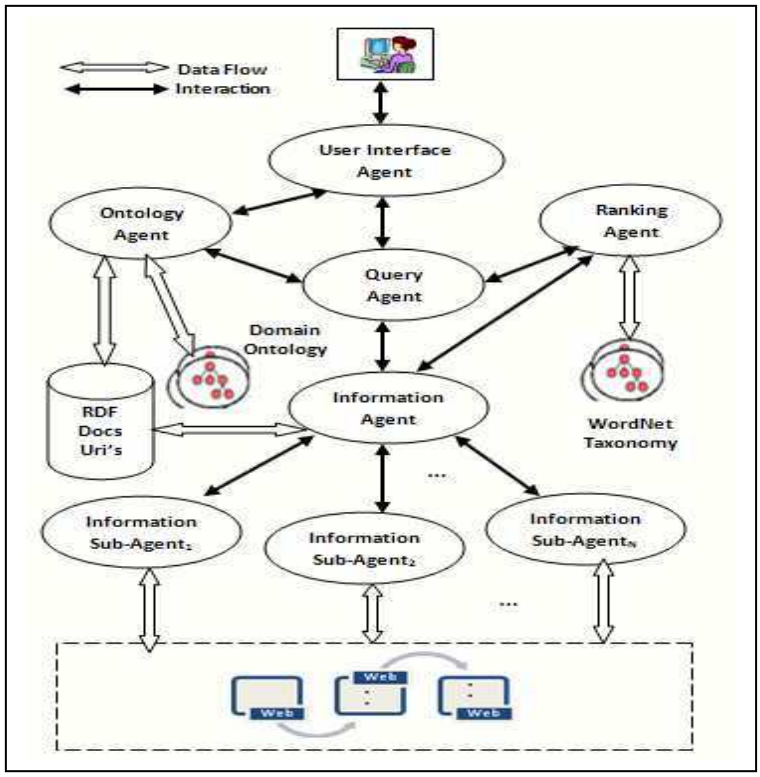


Figure IV.6 : Architecture du Système Multi-Agents

1- Répondre aux requêtes via des URI documents pertinents.

2- Parcourir automatiquement et simultanément plusieurs sites web pour rechercher les concepts liés à l'ontologie du domaine et veiller aux changements sur l'ontologie.

3- Rechercher, identifier et extraire des informations utiles.

L'architecture générique du système est présentée en Figure IV.6, les principales unités, illustré dans l'architecture générique sont:

- Unité « Interface Utilisateur »
- Unité de « Traitement de requête »
- Unité de « Recherche d'information »

L'agent « interface utilisateur » est considéré être le moyen par lequel l'utilisateur interagit avec le système. L'agent « information» recueille les ressources d'informations pertinentes pour l'utilisateur, tandis que l'agent «ontologie» inspecte et contrôle les changements dynamiques des informations. Dans l'unité de traitement, l'agent « requête » coordonne les activités du système.

Les agents assument un comportement intelligent, c'est dans cette perspective que nous avons choisis le type d'agents cognitifs « BDI» pour (croyance, désir, intention), qui communiquent par l'envoi de messages par le protocole de mise en forme FIPA-ACL. L'architecture interne, le mode délibération et le diagramme de séquence d'un agent BDI, sont donnés respectivement, sur les Figures IV.7 et IV.8

IV.5.1 Agent « Interface »

L'agent d'interface fournit une interface conviviale pour interagir avec le système. Lors d'une session de recherche, il enregistre la requête de l'utilisateur par le choix de concepts et de relations définies dans l'ontologie. Éventuellement l'utilisateur peut introduire diverses préférences de recherche et un ensemble de variables définissant les seuils de calcul.

En outre, l'agent présente les résultats obtenus et peut adopter un comportement intelligent en apprenant des expériences passées et des feedbacks utilisateur sur les précédentes requêtes.

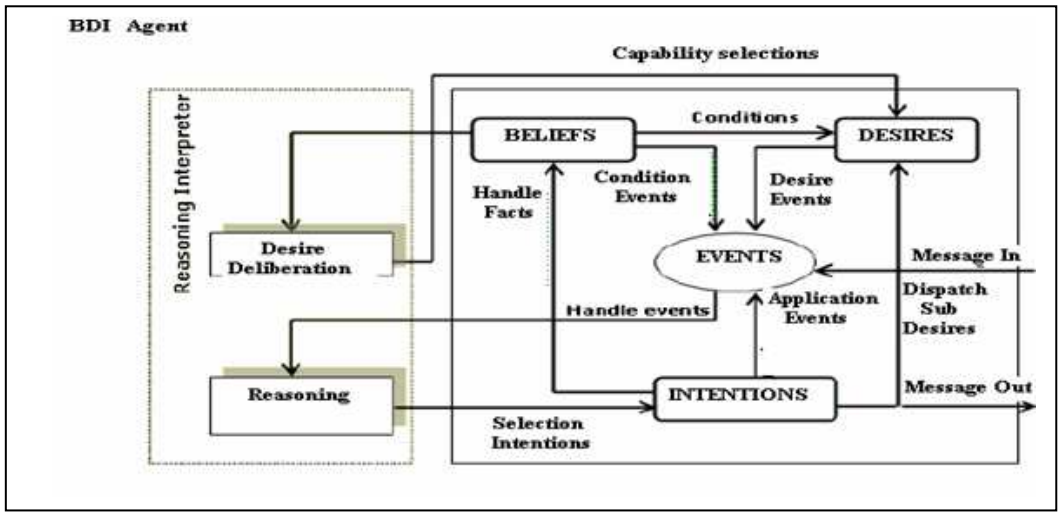


Figure IV.7: la Délibération dans un agent "BDI" [Nes 2013]

IV.5.2 Agent « Requête »

Supervise l'exécution coopérative d'une session de recherche. En premier, il interagit avec l'agent « Interface utilisateur» pour construire l'expression de requête syntaxique-sémantique exprimée dans un langage d'interrogation d'ontologie (RDQL / SPARQL). Cette

tâche s'accomplit en interaction avec l'utilisateur par la spécification explicite des concepts et des relations, et le choix des termes d'expansion de la requête.

L'agent « Requête » utilise une base de connaissance qui comprend les fichiers des définitions des autres agents du système et leurs habilités. En fonction de leurs aptitudes, il leurs alloue les tâches à accomplir pour atteindre leur objectif commun, les tâches principales sont suivantes :

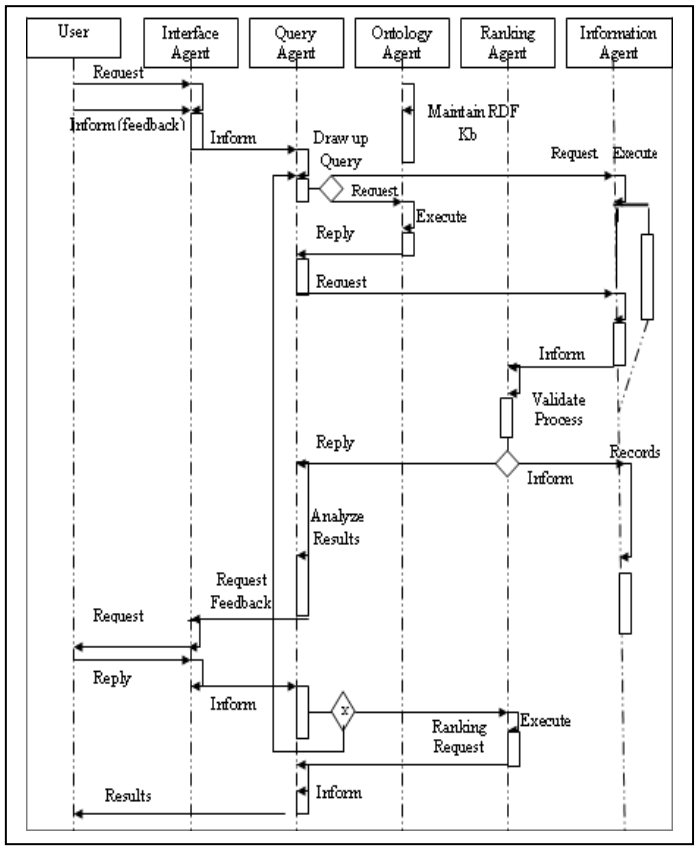


Figure IV.8: Diagramme de séquence AUML du SMA

IV.5.2.1 Recherche par l'expansion de requête

Ce processus vise à extraire de la taxonomie WordNet et pour un terme donné «Ti», un ensemble de synonymes, hyperonymes et hyponymes en tenant compte des préférences utilisateurs (profondeur ascendants/descendants). Ces ensembles ainsi formés constituent ce qu'on conviendra d'appeler «Classe d'Annotation » [Nes 2013].

L'agent « Requête » interagit avec l'agent « Information » pour exécuter la requête RDQL générée, le résultat est un ensemble de liens de documents qui satisfont la requête.

Supposons 'Ti' un terme, et {Ti} son classe d'annotation, donc nous pouvons avoir Ti1 ∈ {Ti}, Ti2 ∈ {Ti}…Tik ∈ {Ti}. Si 'D'est un document, et S1, S2, S3 trois phrases annotée par les instances Ti1, Ti1 et Ti2, alors du point de vue sémantique, les trois phrases

sont identiques même avec leurs différentes syntaxes puisque S1, S2 and S3 sont sémantiquement annotées par la même classe d'annotation {Ti}. Pour un terme donné Wi, on construit l'ensemble :

$$Cwi=\{Synonyms(wi) \cup Hypernyms_{depth}(wi) \cup Hyponyms_{depth}(wi)\} \quad (IV.4)$$

Le poids des classes d'annotations est calculé pour évaluer la pertinence du document, son adéquation pour la requête et pour mettre en œuvre un algorithme de classification.

Tenant compte du principe de génération de classes d'annotations à partir de la requête, nous avons adapté l'Algorithme 'Tf-Idf ", telle que décrite par Wang et al. (2011) et Castells et al. (2009), pour calculer le poids d'une classe, les termes d'expansions sont pondérés par leurs proportions de similarité avec le mot clé correspondant, l'expression de calcul est

$$[X_k]Q = \frac{\sum_{k=1}^{t} freq(A_k * sim(A_k, W_k))}{\max_y [X_y]Q} \quad (IV.5)$$

t : nombre de termes générés pour le mot "W_k", dans la classe $[X_k]$.

A_k: k^{ieme} terme d'annotation dans la classe $[X_k]$.

$[X_k]Q$: poids de la classe d'annotation « X_k » du mot clé "k" dans la requête "Q".

$Max_Y[X_Y]Q$: "Y" représente la classe d'annotation de poids maximum dans la requête Q.

IV.5.2.2 Recherche sémantique

La recherche sémantique proprement dite est effectuée par la recherche d'opération de projection entre la requête soumise et les schémas d'annotation des documents.

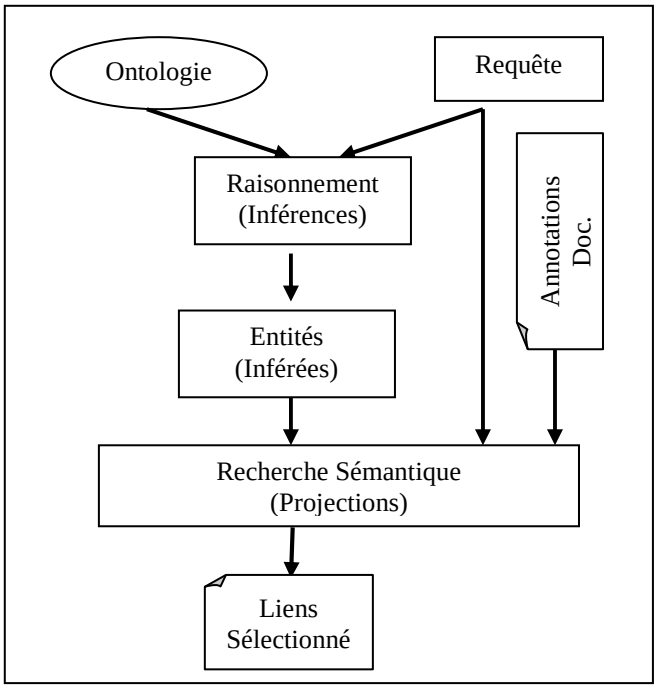


Figure IV.9 : Processus de recherche sémantique

La projection suppose un calcul d'opérations de spécialisations entre deux schémas conceptuels, cela nécessite l'utilisation d'un mécanisme d'inférences permettant de calculer les restrictions sur les types concepts et les types relations.

En utilisant les concepts et les relations soumis dans la requête utilisateur, on élabore des modèles d'inférences pour calculer des spécialisations, dans notre modèle nous avons travaillé sur deux modèles d'inférence « rdf:type » basés JENA. L'un utilise le concept de transitivité des sous classes et l'autre se base sur la transitivité des propriétés [Nes 2013].

Par exemple, dans le domaine « Tourisme », une requête peut porter sur le concept «Sahara », on essayera de voir ce que l'on peut inférer sur le modèle de l'ontologie par rapport à ce concept. Les entités inférées, par exemple « Oasis_region » et « désert » comme sous classes permettront de sélectionner des documents lorsque ces derniers sont annotés par ces entités. Ce processus est illustré par la Figure IV.9.

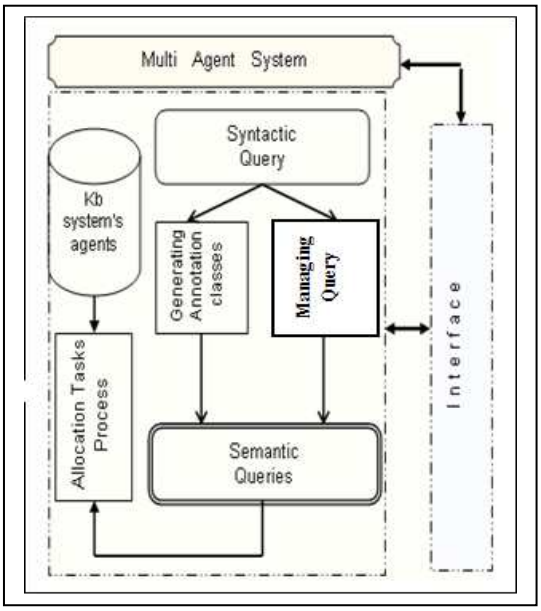


Figure IV.10 : Structure de l'agent « Requête »

La structure interne de l'agent « Requête » est donnée ci-dessus, « Figure IV.10 ».

Algorithm 1 Query-interactions
Begin
1. **Get** message (Interface)
2. **Manage** Query
- **Interface** (WordNet)
- **With** user's feedback
Repeat
- **Build** (Annotation classes (i))
- **Compute** new synonyms, hypernyms and hyponyms weights
- **Generate** (RDQL query)
- **Send** message (Information)
- **Send** message (Interface)

- **Get** message (interface)
  Until feedback
3. **Execute** Jena rdf: type Inference
4. **Send** message (Information)
5. **Send** message (Ranking)
6. **Send** message (Interface)

End

IV.5.3 Agent « Information »

L'agent information, peut contracter plusieurs autres agents pour accomplir divers types de recherches, le choix d'un agent dépendra de son aptitude, de la nature de la recherche (exécution de requêtes RDQL ou recherche d'opérations de projection), et de la nature des informations recherchée telle que des images, des vidéos, des textes etc.

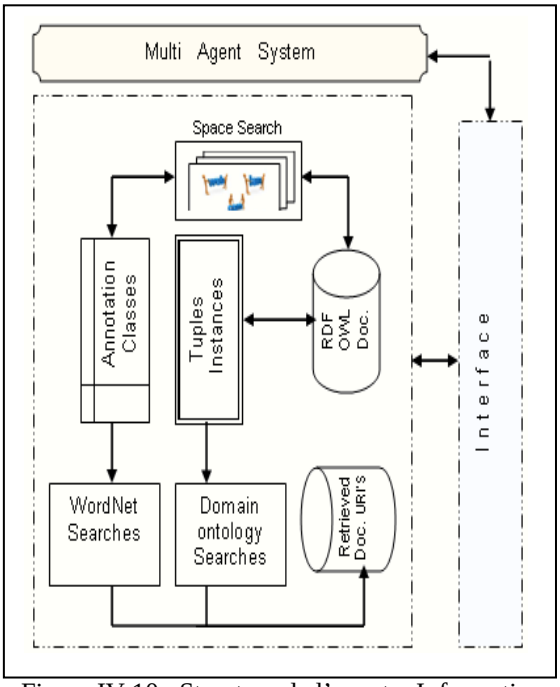


Figure IV.10 : Structure de l'agent « Information »

Le module de recherche par recherche de projection vérifies les spécialisations entre les instances inférées, et les triplets RDF du schéma d'annotation du document. La structure de l'agent « Information » est donnée en « Figure IV.11 »

Algorithm 2 Information
Begin
1- **Repeat**
  Get message (Query);
  Execute RDQL Query
  Record (document-links)
End repeat
2-**Get** message (Query);
3-**Execute** Projection operation
  R**ecord** (documents links)

4- **Send** Message (Query)
End

IV.5.4 Agent « Classement » (Ranking)

L'agent « information » stocke les liens des ressources pertinentes dans un fichier temporaire auquel l'agent «Classement» accède, c'est une mémoire d'échanges où l'agent télécharge l'ensemble des documents référencés pour effectuer les tâches suivantes:

- **LDSim une nouvelle mesure de similarité**

La mesure de similarité sémantique entre concepts est un problème générique dans de nombreux domaines de recherche tels que l'intelligence artificielle, biomédecine, la linguistique, les sciences cognitives et la psychologie. La difficulté est de comment simuler le processus des jugements humains sur la similarité des mots.

La mesure de similarité que nous proposons est liée aux travaux de [Wu,1994] , [Toc,2007], [Pir,2010] et [She,2012] et est basé sur la structure de l'hiérarchie WordNet à travers deux paramètres [Nes 2013]:

- La longueur du chemin entre les synsets.
- La profondeur du subsumant le plus spécifique dans la hiérarchie.

Nous supposons que «C_1» et «C_2» sont les synsets liés aux concepts C_1 et C_2.

L (C1) et L (C2) désignent la longueur des chemins de "C1" et "C2".

Sim (C1, C2) représente la similarité sémantique entre "C1" et "C2".

D (C) dénote la profondeur du concept de «C» dans l'hiérarchie

Dans ce travail, le principe de calcul de la similarité est basé sur la méthode « Edge counting », qui utilise le nombre d'arcs à partir de la racine de l' hiérarchie, elle est limitée aux liens taxonomiques et considère la position de deux concepts C_1 et C_2 dans l'hiérarchie et par rapport à la profondeur de leurs subsumant les plus spécifique « MSCS ».

Comme il peut y avoir plusieurs parents pour chaque concept, deux concepts peuvent partager des parents par de multiples chemins. Le MSCS désigne le parent commun avec le nombre minimum d'arcs allant du MSCS vers les concepts C_1 et C_2.

Avec L (C_1) ≤ L (C_2), nous mesurons la similarité sémantique entre C_1 et C_2 par

$$\mathrm{LDSim}(C_1, C_2) = \frac{L(C_1)}{L(C_2)} \qquad \text{(IV.6)}$$

Comme MSCS dénote le subsumant le plus spécifique de C_1 et C_2, il est évident que :

D(MSCS) ≤L(C_1) and D(MSCS) ≤L(C_2) ; D: désigne la profondeur.

Supposons que « P » un concept situé à la profondeur « N » et représente le parent direct du concept « C ». Sur la base de l'équation (IV.6) nous obtenons:

$$\mathrm{LDSim}(P, C) = \frac{N-1}{N} = 1 - \frac{1}{N} \qquad \text{(IV.7)}$$

Etant donnée l'équation IV.7, nous retenons une importante et souhaitable propriété de notre mesure de similarité qui assigne une forte similarité aux termes rapprochés (en termes de longueur du chemin) et inférieure dans la hiérarchie (les plus spécifiques), que pour les termes qui sont également rapprochés mais plus haut dans la hiérarchie (les plus généraux).

La similarité sémantique entre le subsumant le plus spécifique «MSCS», et le concept C_1 (respectivement, C_2) est calculée par l'expression:

$$\mathrm{LDSim}(MSCS, C_1) = \frac{D(MSCS)}{L(C_1)} \qquad \text{(IV.8)}$$

$$\mathrm{LDSim}(MSCS, C_2) = \frac{D(MSCS)}{L(C_2)} \qquad \text{(IV.9)}$$

Pour estimer la similarité sémantique entre les concepts C_1 et C_2, l'idée que nous avons explique dans [Nes 2013] consiste à appliquer une notion de probabilité qui stipuler que la probabilité de l'intersection de deux événement s'exprime par le produit de leurs probabilités.

Se basant sur ce fondement mathématique et tenant compte des équations (IV.8) (IV.9) nous obtenons l'expression :

$$(\mathrm{LDSim}(C_1, C_2))^2 = \frac{D(MSCS) * D(MSCS)}{L(C_1) * L(C_2)} \qquad \text{(IV.10)}$$

Finalement, la similarité entre les concepts C_1 et C_2 est donnée par la formule:

$$\mathrm{LDSim}(C_1, C_2) = \frac{D(MSCS)}{\sqrt{L(C_1)} * \sqrt{L(C_2)}} = \frac{D(MSCS)}{\sqrt{L(C_1) * L(C_2)}} \qquad \text{(IV.11)}$$

Cette mesure que nous proposons, donne approximativement les mêmes résultats que pour la formule de Wu-Palmer [Wu,1994]. Les résultats et les interprétations sont donnés dans le chapitre des expérimentations.

Nous définissons la similarité sémantique par une application "LDSim" [Nes 2013] :

$$\mathrm{LDSim}: \{N \times N\} \rightarrow R$$

$$A,B \rightarrow \mathrm{LDSim}(A, B) = \frac{C}{\sqrt{A * B}} \qquad \text{(IV.12)}$$

A,B : entiers (longueurs des chemins des concepts C_A et C_B

C: entier (profondeur du MSCS (C_A,C_B)

Si le concept de C_B est un parent du concept C_A, on applique:

$$\mathrm{LDSim}(A, B) = \frac{C}{\sqrt{A * B}} + \frac{|\mathrm{Log}(\mathrm{LDSim}(B, D)|}{A} \qquad \text{(IV.13)}$$

D: est un concept de frère du concept B avec MSCS(B,D)=C.

LDSim(B,D): similarité sémantique entre les concepts C_B,C_D.

Un exemple d'application de l'équation (IV.13) est donné en figure 7.

$LDSim(C_A,C_D) = 2/(4*3)0,5 = 0,5773$

$LDSim(C_A,C_B) = 0,5773+(|log(LDSim(C_B,C_D)|)/4$

$|Log(LDSim(C_B,C_D))| = |Log(2/(3*3)0,5| = 0,1760$

$LDSim(C_A,C_B) = 0,5773 + 0,1760/4 = 0,7533.$

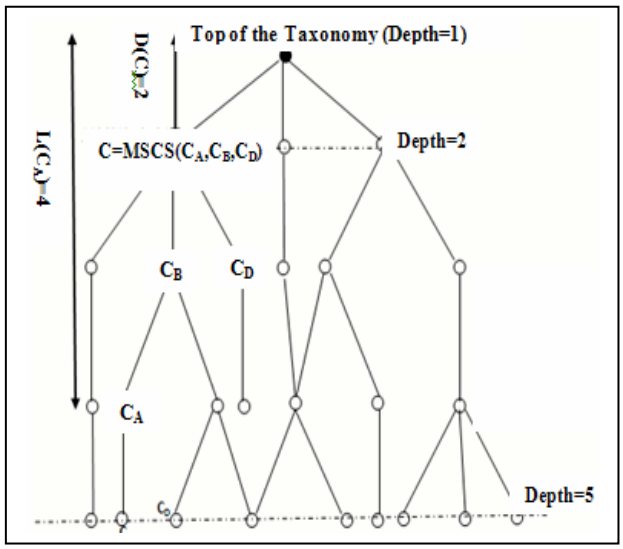


Figure IV.11 : Mesure de similarité basée structure de l'hiérarchie

Nous avons appliqué la fonction logarithme pour ses spécifiques caractéristiques requises ces applications, à savoir :

- En valeur absolue, elle tend à zéro lorsque la similarité augmente (voisinage de 1).
- Incrémentation rapide (en valeur absolue), lorsque la similarité diminue (voisinage de 0).

Interprétation: Lorsque la similarité entre deux concepts est au voisinage de l'unité '1', la fonction logarithme tend vers '0'. Comme décrit précédemment, cela signifie que ces concepts sont très proches (IS-A père direct – fils) et situés assez bas dans la taxonomie.

On peut déduire que La similarité sémantique entre ces concepts (descendants et frères) est à peu près identique. Lorsque la similitude tend à '0', cela signifie que les concepts sont assez génériques, et sont situés en haut de l'hiérarchie. Leurs similarité dans ce cas, doit être distinguée. Il est évident que la similarité d'un concept avec son parent direct doit être plus grande que sa similarité avec un frère d'un parent direct. Cette différence est donnée par la valeur absolue du logarithme de la similarité des son père avec l'un de ses frères, pondérée par la profondeur du concept en question. (Figure IV.11).

La similarité LDSim satisfait les propriétés suivantes:

Soit C, C_1, C_2, Hi, Si des concepts de la hiérarchie.

1- $\forall$ C_1, C_2; LDSim (C_1,C_2)$\in$]0,1]

En effet, de l'équation « IV.13 », nous concluons que la similarité maximale est obtenue, comme il a été interprété ci-dessus, lorsqu'elle est calculée entre un concept « N » et son parent direct « N-1 » par rapport à leurs subsumant le plus spécifique qui est dans ce cas le deuxième parent de « N » c'est-à-dire « N-2 ». En effectuant dans l'équation IV.13, les substitutions suivantes :
A par N ; B et D par (N-1) ; C par (N-2) nous obtenons :

$$LDSim(A,B) = \frac{(N-2)}{\sqrt{(N(N-1)}} + \frac{\left|Log(LDSim(B,D)\right|}{A} \qquad (IV.14)$$

$$LDSim(B,D) = \frac{(N-2)}{(N-1)}$$

L'équation (IV.14) devient : $$LDSim(A,B) = \frac{(N-2)}{\sqrt{(N(N-1)}} + \frac{\left|Log\frac{(N-2)}{(N-1)}\right|}{N} \qquad (IV.15)$$

Pour N (profondeur de l'hiérarchie) assez grande, la valeur de LDSim(A,B) tend vers « 1 ».

2- $\forall$C ; LDSim (C,C) = 1.
3- $\forall$ C1,C2 ; LDSim (C1,C2) = LDSim(C2,C1) (Symmetry)
4- $\forall$ C_1,C_2 ; LDSim (C_1,C_2)=1 $\Rightarrow$ $C_1 \equiv C_2$
5- $\forall H_I$ hyponyme 'C' ; $\forall$ S_I synonyme 'C' ; LDSim(C, HI) ≤ LDSim(C, SI).
6- Depth(R) = 1 where R is the root node.

Aussi nous avons : $$Log(N) = \left(\sum_{K=1}^{N} \frac{1}{K}\right) - \alpha$$; α : constante d'Euler. (IV.16)

Sur la base des équations (IV.13) et (IV.16) et pour N assez grand nous obtenons:

$$LDSim(A,B) = \frac{C}{\sqrt{A*B}} + \frac{1}{N} \qquad (IV.17)$$

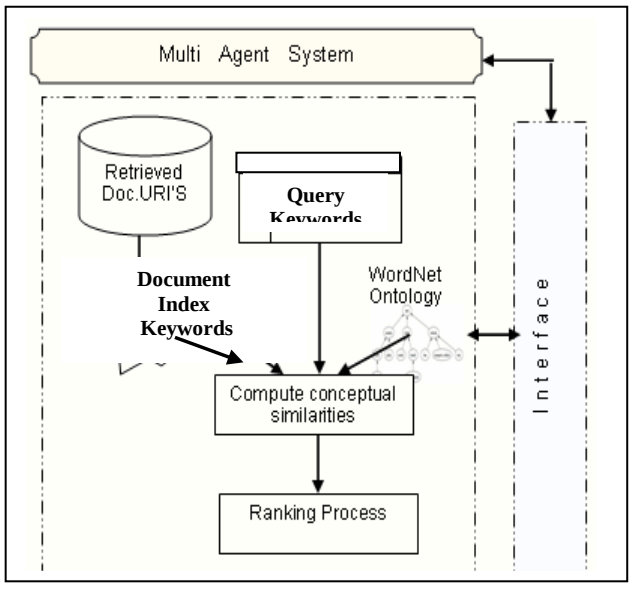


Figure IV.12 : Structure de l'agent « Classement »

Les mots-clés de la requête sont pondérés par leurs occurrences. Par conséquent, la requête elle-même est représenté par un vecteur Vq = (w_{1q}, w_{2q}, …, w_{iq}, …, W_{nq}), w_{iq}: poids du mot-clé «i» dans la requête q. L'adaptation d'une représentation par le modèle d'espace vectoriel consiste à affecter aux annotations sémantiques des pondérations reflétant l'importance de la classe d'annotation pour le document.

La structure de l'agent « Classement » est donnée en Figure « IV.12 »

La formule du cosinus est utilisée pour calculer la similarité Requête-Document, son expression:

$$Sim(D_j, Q) = \frac{\overrightarrow{D_j} * \overrightarrow{Q}}{|D_j| * |Q|} \qquad (IV.18)$$

Pour évaluer la pertinence des résultats retournés, la mesure calculée est comparée à un seuil minimum « Pmin », le processus se termine lorsque tous les documents seraient analysés.

IV.5.5 Agent « Ontologie »

Cet Agent est attaché à l'ontologie et doit avoir les habilités nécessaires pour maintenir les changements dynamiques de l'ontologie. Ces tâches requièrent l'usage d'un éditeur d'ontologie pour insérer, modifier et supprimer des concepts, des instances et des relations conceptuelles. Un autre rôle de cet agent est de crawler le Web à des intervalles réguliers à la découverte des documents annotés RDF / OWL compatible avec l'ontologie de domaine spécifiée. Les liens des documents retrouvés seront sauvegardés dans une base de données de documents annotées RDF. Ainsi, l'agent par un comportement proactif peut tenir compte de la dynamique des changements de l'information sur le web. La structure de l'ontologie « Ontologie » est en Figure « IV.13 ».

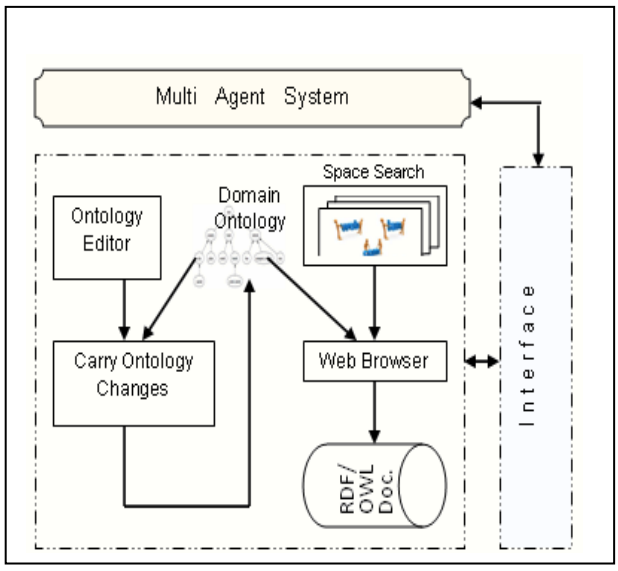


Figure IV.13 : Structure de l'agent « Ontology »

Conclusion

Dans les chapitres précédents nous avons exposé l'état de l'art relevant du domaine de notre investigation. Ainsi, les modèles et les démarches relatives à ce domaine de recherche ont fait l'objet de descriptions et d'analyses inhérentes à leurs caractéristiques. C'est un domaine large qui connait d'intenses recherches qui portent sur plusieurs aspects, car il se situe à la connexion de plusieurs disciplines comme le traitement automatique du langage naturel, le web sémantique, et le paradigme agent qui reste toujours d'actualité.

Nous nous sommes inspiré des travaux et des résultats obtenus pour proposer notre modèle de raisonnement pour la recherche sémantique d'information basée agents. Nous avons d'abord abordé la problématique de l'annotation sémantique des documents sur le web, c'est en fait le traitement pivot du processus de recherche sémantique, après nous avons exploré les techniques de traitement du langage naturel qui sont utilisées pour l'extraction de termes descripteurs (mots clés). Ensuite nous avons décrit notre nouvelle démarche de délimitation des segments sémantiques dans un texte, car il semblait nécessaire de distinguer les différents thèmes sémantiques dans un texte pour pouvoir les repérer par des instances d'annotation.
L'analyse syntaxique d'un segment est aussi un passage nécessaire dans ce processus, pour cela il existe plusieurs outils linguistiques comme Connexor, SygFran, et GATE. Ces outils morphosyntaxiques analysent un texte et fournissent en sortie des structures arborescentes qui tracent les connexions syntaxiques et les rôles grammaticaux des termes et leurs relations.

Nous avons choisi d'utiliser le formalisme des graphes conceptuels pour exprimer la sémantique des connaissances (concepts et relations) que fournies l'analyse syntaxique, ce choix se justifie par : i) les graphes conceptuels, sont par essence bien adapté aux traitements des langages naturels, notamment à la représentation des textes et ii) ils sont une forme très expressive de la logique, sur lesquels nous avons la possibilité de raisonnements complets.

La transformation de la structure syntaxique produite par un analyseur syntaxique en un graphe conceptuel est un processus déterminant dans l'expression de la sémantique véhiculée par une portion d'un texte, pour cela nous avons opté pour l'usage d'un ensemble de règles de transformation qui peut être amélioré au fur et à mesure de l'expérimentation et des résultats obtenus. La dernière étape consiste à utiliser les langages du web sémantique pour exprimer les connaissances d'annotation du graphe conceptuel.

Nous avons ensuite présenté un modèle de recherche sémantique basé agent, le principe est de montrer le potentiel de l'utilisation de ces technologies pour faire face aux défis

de la recherche sémantique d'information. C'est une tentative pour surmonter deux défis majeurs du web sémantique qui sont:

- Comment rendre le contenu Web disponibles par le sens.
- La difficulté à intégrer et à maintenir des ressources Web hétérogène et dynamiques en utilisant le paradigme d'ontologie.

Pour cela, nous avons présenté une démarche hybride pour remédier aux lacunes des recherches purement syntaxiques et purement sémantiques. Ainsi, une recherche par expansion de requête est un processus qui vise à répondre aux insuffisances des recherches sémantiques car les modèles de représentations sémantiques restent d'expressivité limitée.

Aussi pour dépasser les recherches syntaxiques, nous avons choisis l'ontologie comme modèle de représentation des connaissances assez expressif, permettant des raisonnements, en particulier, nous avons utilisé le raisonnement basé sur la recherche d'opération de projection entre les graphes requête et annotation document. L'existence d'une telle opération suffit pour conclure que le document vérifiant les critères de projection est une réponse pertinente à la requête en question.

De même, pour savoir ordonner les résultats, nous avons présentée une nouvelles mesure de similarité entre concepts basée WordNet et utilisant la méthode « edge-counting ». Les expérimentations montrent que cette mesure est une variante améliorée de la mesure « Wu-Palmer », les résultats donnés sont jugés bons. L'évaluation de notre contribution fera l'objet du chapitre suivant.

CHAPITRE V

Etude de Cas et Aspects d'Implémentation des Composants du Modèle

V.1 Introduction

Pour expérimenter notre modèle, nous avons focalisé le travail sur les parties qui concernent les raisonnements proposés. Ainsi, en premier, nous allons donner des résultats qui expriment la conceptualisation de l'approche de segmentation sémantique du texte, en traitant un texte en entrée pour restituer les segments produits comme résultats.
Ensuite nous montrerons comment calculer et analyser les résultats de la nouvelle mesure de similarité sémantique.

En dernier nous allons définir le concept de modèle d'inférence Jena, les instances inférées par le raisonneur basées sur la transitivité des classes et des propriétés. Nous signalons à cet effet que les modules de la démarche globale sont en cours de tests individuellement, les fonctionnalités du système complet sont en cours de développement.

V.2 Segmentation sémantique

Pour analyser les résultats de l'approche de segmentation sémantique détaillée au chapitre IV, et qui avait fait l'objet d'une publication dans le journal référencé dans la rubrique de nos contributions. Nous avons pris un texte à segmenter de trois paragraphes, par ce test nous désirons savoir si ces paragraphes expriment une même sémantique ou alors nous avons plusieurs segments sémantiques. Soit le texte:
"Algerian ***Sahara*** is very vast; ***Ghardaïa*** *serves as the northern entry pointing the* ***Sahara****. While it is in the* ***desert****, it is not what you have in mind,* ***sand dunes****! No, there are no* ***sand dunes*** *in* ***Ghardaïa****, but mainly rocky* terrain.
Vast majority of the ***Sahara*** *is actually rocky terrain and not* ***sand dunes****. The main investments in the* ***Sahara*** *are* ***petroleum*** *industry and* ***tourism****.*
In the ***M'zab*** valley, we discover its traditionnel system of ***watershed*** and its wonderful architecture which still fascinates today by its refined and functional aesthetics, its customs rooted centuries ago and which allowed the preservation of the picturesque and specific aspect of this region in the Algerian ***Sahara***. "

Le module semi automatique d'extraction de termes donnera les mots en gras dans le texte, on notera ici que certains termes qui n'existent pas dans WordNet ont été pris comme mots. Par exemple nous avons considéré le mot « Petroleum » pour le terme « petroleum industry », le terme « sand dunes » figure dans le dictionnaire, nous l'avons traité comme terme. De l'hiérarchie WordNet, nous obtenons leurs longueurs depuis la racine, ainsi que la profondeur de leurs subsumants le plus spécifique.

On suppose égale à l'unité « 1 » la longueur d'un mot qui n'existe pas dans WordNet (profondeur est donc de 2). C'est-à-dire descendant direct du concept « Entity », comme pour le mot « Ghardaïa ». Le poids d'un mot est calculé par le rapport entre la fréquence de ses occurrences, et la fréquence des occurrences du mot le plus répété dans les deux paragraphes.

$$Poids(Wi) = \frac{Freq(Wi)}{Max(Freq(Wj)), j = 1, p}$$ p : nombre de termes retenus dans les deux paragraphes

Pour comparer les deux premiers paragraphes Ph_1 et Ph_2, nous reportons dans le tableau V.1, les poids et les positions (profondeurs) des termes extraits des deux paragraphes. Les termes qui figurent dans les deux paragraphes sont ignorés dans le calcul des similarités entre termes.

Terme	Occur.	Poids	Longueur
Ghardaïa	2	1	2
Desert	1	0,5	8
Petroleum	1	0,5	9
Tourism	1	0,5	11

Tableau V.1 : Poids et longueur de termes dans Ph_1 et Ph_2

La matrice « Terme x Paragraphe » pour les paragraphes Ph_1 etPh_2 est donnée au tableau V.2

	Ghardaïa	Desert	Petroleum	Tourism
Ph_1	11	12		
Ph_2			23	24

Tableau V.2 : Matrice « Terme x Paragraphe »

Le tableau V.3 donne les profondeurs du subsumant le plus spécifique, et des mesures de similarités par application de équation « IV.13 » décrite au chapitre IV.

Terme(Ph_1)/ Terme(Ph_2)	Depth (MSCS)	Similarité	Terme(Ph_2)/ Terme(Ph_1)	Depth (MSCS)	Similarité
11/23	1	0,236	23/11	1	0,236
11/24	1	0,213	23/12	2	0,236
12/23	2	0,236	24/11	1	0,213
12/24	1	0,107	24/12	1	0,107

Tableau V.3 : Profondeur et mesure de similarités de mots entre paragraphes Ph1 et Ph2

Les termes extraits des paragraphes Ph_2 et Ph_3 sont donnés en tableau V.4.

Dans le tableau V.5 nous avons la matrice « Terme x Paragraphe » relative à ces paragraphes, les données relatives aux longueurs, aux profondeurs du subsumant le plus spécifique et aux similarités calculées entre les termes des paragraphes Ph_2 et Ph_3 sont présentés dans le tableau V.6

Terme	Occur.	Poids	Longueur
Petroleum	1	1	2
Tourism	1	1	8
M'zab	1	1	9
Watershed	1	1	11
Sahara	1	1	8

Tableau V.4 : Poids et longueur de termes dans Ph_2 et Ph_3

	Petroleum	Tourism	M'zab	Watershed	Sahara
Ph_2	21	22			
Ph_3			33	34	35

Tableau V.5 : Matrice « Terme x Paragraphe » pour Ph_2, Ph_3

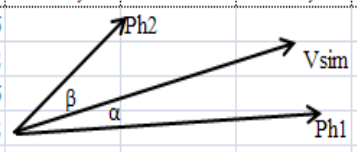
Terme(Ph_2)/ Terme(Ph_3)	Depth (MSCS)	Similarité	Terme(Ph_3)/ Terme(Ph_2)	Depth (MSCS)	Similarité
21/33	1	0,236	…	…	…
21/34	2	0,272	34/22	3	0,123
21/35	2	0,236	35/21	2	0,236
22/33	1	0,213	35/22	1	0,107

Tableau V.6 : Profondeur et mesure de similarités de mots entre paragraphes Ph_1 et Ph_2

La figure V.1, et V.2 donnent respectivement les vecteurs Vsim construits à base de cette démarche, ainsi que leurs mesures de similarités avec les vecteurs des paragraphes du texte. Les mesures des vecteurs sont données par le cosinus de l'angle entre vecteurs. Dans la figure V.1 on observe que l'angle entre les vecteurs Ph_1 et Vsim est plus petit que l'angle entre Ph_2 et Vsim.

Similarité entre (Paragraphe 1 et Paragraphe 2)

	Vsim	Vsim^2	Ph1	Ph1^2	Ph2	Ph2^2	Ph2*Vsim	Ph1*Vsim
Ghardaia	**0,236**	0,056	**1,000**	1,000	**0,000**	0,000	0,000	0,236
Desert	**0,118**	0,014	**0,500**	0,250	**0,000**	0,000	0,000	0,059
Petroleum	**0,118**	0,014	**0,000**	0,000	**0,500**	0,250	0,059	0,000
Tourism	**0,107**	0,011	**0,000**	0,000	**0,500**	0,250	0,054	0,000
		0,308		**1,118**		**0,707**	**0,113**	**0,295**
COS(Vsim,Ph1)	**0,856**							
α°	**31,12**							
COS(Vsim,Ph2)	**0,516**							
β°	**58,92**							



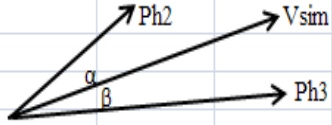
les paragraphes :Ph1 et Ph2 forment un même segment sémantique car le vecteur resultant 'Vsim' est proche de Ph1.

Figure V.1 : Similarités entre paragraphes Ph_1 et Ph_2

Dans ce cas, le vecteur Ph_2 porte une sémantique proche de celle du vecteur Ph_1 car la résultante Vsim est proche de Ph_1. On déduit que les deux paragraphes Ph_1 et Ph_2 forment un même segment sémantique. Aussi, dans la figure V.2, le vecteur Vsim est plus proche du vecteur Ph_3, ce qui signifie que les paragraphes Ph_2 et Ph_3 portent des sémantiques différentes.

Similarité entre (Paragraphe 2 et Paragraphe 3)

	Vsim	$Vsim^2$	Ph2	$Ph2^2$	Ph3	$Ph3^2$	Ph2*Vsim	Ph3*Vsim
Petroleum	0,272	0,074	**1,000**	1,000	**0,000**	0,000	0,272	0,000
Tourism	0,213	0,045	**1,000**	1,000	**0,000**	0,000	0,213	0,000
M'zab	0,236	0,056	**0,000**	0,000	**1,000**	1,000	0,000	0,236
Watershed	0,272	0,074	**0,000**	0,000	**1,000**	1,000	0,000	0,272
		0,499		**1,414**		**1,414**	**0,485**	**0,508**
COS(Vsim,Ph2)	**0,687**							
α°	**46,59**							
COS(Vsim,Ph3)	**0,720**							
β°	**43,96**							



Les paragraphes Ph2 et Ph3 représentent deux segments sémantiques car le vecteur resultant 'Vsim' est proche de Ph3

Figure V.2 : Similarités entre paragraphes Ph_2 et Ph_3

V.3 Mesure de similarité sémantique

Pour expérimenter la mesure de similarité proposée, nous avons pris pour l'hiérarchie WordNet la profondeur (D=16). Des mesures de similarités entre deux concepts C_A,C_B sont calculées en faisant varier leurs longueurs dans WordNet en combinaison avec des profondeurs de leur subsumant le plus spécifiques sur divers nivaux (de 1= profondeur de la racine jusqu'au niveau le plus bas=15).

Ainsi nous avons montré l'utilité d'usage de la composante logarithmique dans la formule « IV.13 » du chapitre IV. Les résultats obtenus sont montrés par « Figure V.3 » et « Figure V.4 ». Dans les deux premières colonnes de la figure V.3, nous avons donné plusieurs longueurs de concepts comparés, la troisième colonne indique la profondeur du subsumant le plus spécifique. Nous avons aussi introduit un booléen pour caractériser les occurrences où l'un des concepts comparé est un parent direct de l'autre concept.

La colonne suivante donne les résultats de notre mesure de similarité « LDsim » comme décrite au chapitre IV, et la dernière colonne montre les résultats que fournit la mesure de Wu et Palmer pour les mêmes concepts. Dans la figure IV.4, nous avons une interprétation graphique des résultats des deux mesures, pour saisir leurs variations en fonction des valeurs choisies pour les tests.

Nous observons alors que les courbes tracées se confondent sur plusieurs intervalles, notre mesure semble fournir une meilleure distribution, due à l'utilisation de la composante logarithmique qui en effet discrimine plus les similarités à différents niveaux de l'hiérarchie.

Lenght (C_A)	Lenght (C_B)	D(C) C=MSCS(C_A,C_B)	Boolean C_B (Ancestor) C_A	LDSim (C_A,C_B)	WP-Sim (C_A,C_B)	Lenght (C_A)	Lenght (C_B)	D(C) C=MSCS(C_A,C_B)	Boolean C_B (Ancestor) C_A	LDSim (C_A,C_B)	WP-Sim (C_A,C_B)
2	2	1	0	5,0000	5,0000	8	4	3	0	5,3033	5,0000
3	2	1	0	4,0825	4,0000	10	4	2	1	3,8612	2,8571
3	2	1	1	6,4124	4,0000	8	5	1	0	1,5811	1,5385
4	2	1	1	5,2830	3,3333	8	5	1	1	1,9574	1,5385
5	2	1	0	3,1623	2,8571	8	5	4	0	6,3246	6,1538
6	2	1	0	2,8868	2,5000	9	5	4	0	5,9628	5,7143
3	3	2	0	6,6667	6,6667	10	5	2	0	2,8284	2,6667
3	3	1	0	3,3333	3,3333	6	6	2	0	3,3333	3,3333
4	3	1	0	2,8868	2,8571	8	6	4	0	5,7735	5,7143
4	3	1	1	4,1939	2,8571	9	6	4	0	5,4433	5,3333
7	3	1	0	2,1822	2,0000	10	6	5	0	6,4550	6,2500
9	3	2	0	3,8490	3,3333	8	7	6	1	9,1842	8,0000
10	3	2	0	3,6515	3,0769	9	7	6	1	8,5960	7,5000
4	4	3	0	7,5000	7,5000	8	8	3	0	3,7500	3,7500
4	4	2	0	5,0000	5,0000	8	8	5	0	6,2500	6,2500
4	4	1	0	2,5000	2,5000	10	8	3	1	3,9281	3,3333
5	5	1	0	2,0000	2,0000	9	9	7	0	7,7778	7,7778
5	4	3	0	6,7082	6,6667	9	9	5	0	5,5556	5,5556
5	4	3	1	8,4583	6,6667	9	9	2	0	2,2222	2,2222
7	4	2	0	3,7796	3,6364	10	9	8	0	8,4327	8,4211
7	4	2	1	4,7782	3,6364	10	9	2	1	2,4550	2,1053
7	4	3	0	5,6695	5,4545	10	10	8	0	8,0000	8,0000
7	4	3	1	6,9196	5,4545	10	10	4	0	4,0000	4,0000
7	7	6	0	8,5714	8,5714	10	10	1	0	1,0000	1,0000

Figure V.3 : Résultats et comparaison des approches LDSim et Wu et Palmer

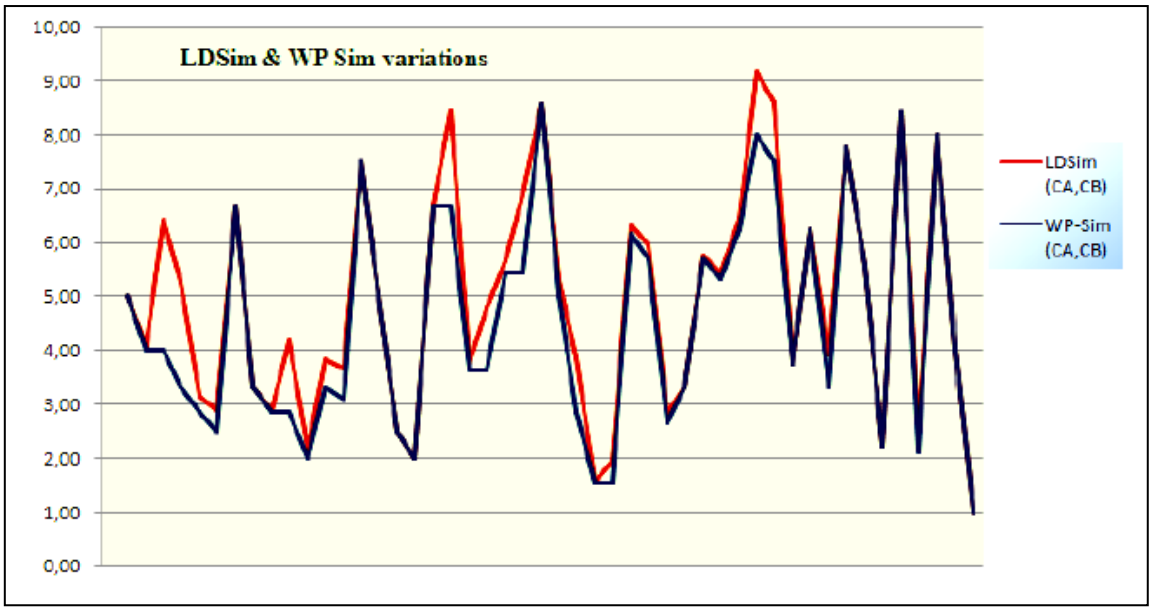


Figure V.4 : Graphique des variations des similarités LDSim et Wu et Palmer

V.4 Spécification de l'ontologie de domaine

Nous avons utilisé l'éditeur d'ontologie « Protégé 2000 » pour spécifier une ontologie du domaine de tourisme, plus précisément le sous domaine « hôtellerie ». Ainsi des classes de

concepts, des propriétés et des rôles ont été spécifiées, et des instances ont été définies pour pouvoir appliquer l'inférence basée Jena. L'efficacité du mécanisme d'inférence, dépend de l'expressivité du langage de spécification qui concerne les restrictions de classes, sur les rôles entre classes et les axiomes. Par exemple on spécifie qu'un hotel classé 5, doit avoir le service d'hotel «visites-guidée »par la classe: visites-guid= ((hotel) ∩ (> = 5 classe. etoile)). L'interface protégé 2000, de spécification des composants et le graphe associé sont donnés par les figures V.5 et V.6.

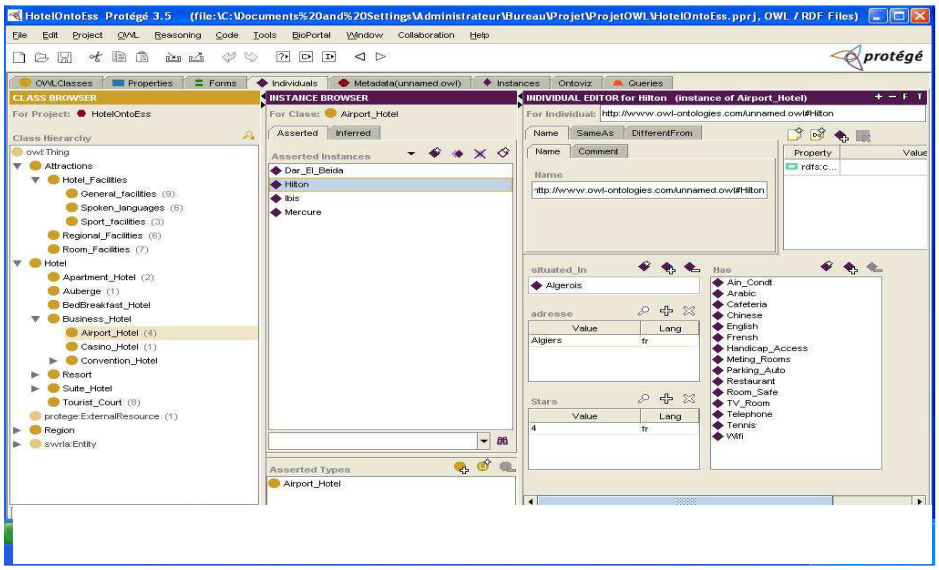


Figure V.5 : Protégé-2000 pour spécifier l'ontologie « hotellerie »

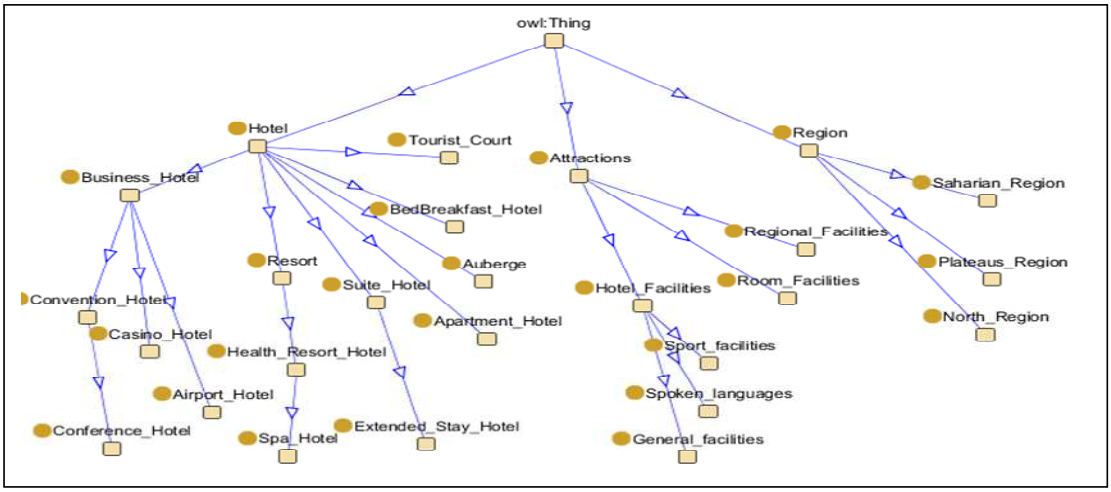


Figure V.6 : Graphe des relations « IS-A » de l'ontologie « hotellerie »

L'expérimentation porte sur le raisonneur RDF de l'API Jena, en premier nous appliquerons un raisonnement basé sur la transitivité de la relation de sous classes, ensuite nous essayerons une inférence basé sur la relation de sous propriétés. La figure V.7 montre un

exemple d'annotations sémantiques associé à un document web décrivant l'hotel nommé « Cirta », situé à Constantine en Algerie, avec les attractions offertes.

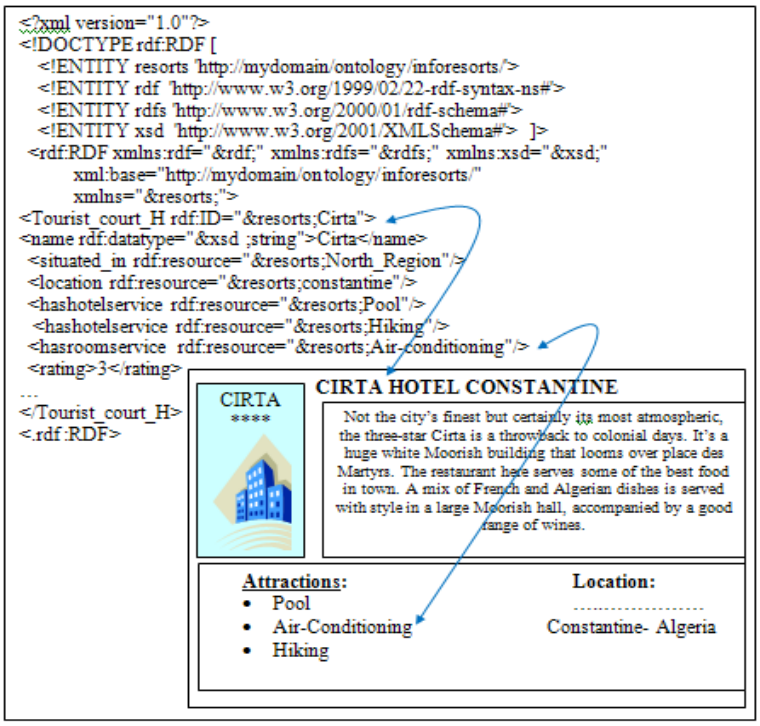


Figure V.7 : Document et annotations RDF associées

V.5 Génération et expansion de requête

On suppose la requête utilisateur suivante : Nom et Adresse des hôtels dans les Oasis Algérien avec les services d'hôtel Internet Wifi et Piscine.

Les concepts de cette requête sont :

"Nom hotel"," Adresse Hotel", "Oasis Algerie", "Wifi", "Pool".

La requête RDQL générée pour cette requête devra être :

SELECT *?resort_name ?document_URI,?region*

WHERE *(?document_URI,<http://www.example.org/terms#name>,?resort_name),*

(?document_URI,<http://www.example.org/terms#hotelfacilities>,?hotel_facilities)

(?document_URI,<http://www.example.org/terms#situated>,?region)

AND *?hotel_facilities* ***EQ*** *"wifi",?hotel_facilities* ***EQ*** *"pool", ?region* ***EQ*** *"oasis Algerie"*

La figure V.8, donne les résultats d'exécution de cette requête qui peut être reformulée par l'utilisateur en lui proposant d'autres concepts reliés par IS-A relation.

Resort_Name	Document_URI	Region
"El-Ouaha"	<Http://...#Document1>	"Bechar"
"Touat"	<Http://...#Document2>	"Adrar"
"El-Mountazah"	<Http://...#Document4>	"Adrar"
"El-Djamil"	<Http://...#Document5>	"Ouargla"

Figure V.8 : Liens des documents retrouvés

V.6 Recherche Sémantique

Comme nous l'avons proposé dans la modélisation de notre approche, la recherche sémantique est réalisée par l'algorithme de recherche d'une projection entre les graphes requête et document. Le graphe de la requête étant le schéma conceptuel de la requête utilisateur, constitué des concepts, des relations et des instances en rapport avec l'ontologie de domaine. Le graphe du document représente les annotations sémantiques en référence à l'ontologie de domaine.

Le recherche d'une projection est la vérification de l'existence d'opérations d'instanciation des concepts et des relations du graphe requête dans le graphe d'annotation du document ainsi que les interconnexions des concepts et des relations. Les hiérarchies des types de concepts et des types de relations sont données par le schéma de l'ontologie. La vérification des opérations d'inclusion des classes et des propriétés nécessite l'utilisation d'un raisonneur, dans ce travail nous avons utilisé l'API Jena comme moteur d'inférence.

V.6.1 Modèle d'inférence

Le modèle d'inférence dans Jena est constitué de deux composantes qui sont :

V.6.1.1 Schéma d'inférence

C'est le schéma de l'ontologie du domaine, la syntaxe XML/RDF de l'ontologie de domaine « hotellerie » est donné ci-dessous, par exemple il est spécifié dans ce schéma que la classe « convention-hotel », est une sous classes de la classe « business-hotel » de la classe principale « hotel ». La classe « hotel » est liée par une propriété « Situated In » à la classe « Region ».

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
   <!ENTITY resorts 'http://mydomain/ontology/inforesorts/'>
   <!ENTITY rdf  'http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
   <!ENTITY rdfs 'http://www.w3.org/2000/01/rdf-schema#'>
   <!ENTITY xsd  'http://www.w3.org/2001/XMLSchema#'>
```

```
]>
<rdf:RDF xmlns:rdf="&rdf;" xmlns:rdfs="&rdfs;" xmlns:xsd="&xsd;"
        xml:base="http://mydomain/ontology/inforesorts/"
        xmlns="&resorts;">
<rdf:Description rdf:about="&resorts;Hotel">
  <rdfs:comment rdf:datatype="&xsd;string">
          Main class for the hotel_ontology</rdfs:comment>
</rdf:Description>
…
<rdf:Description rdf:about="&resorts;ranking">
       <rdfs:domain rdf:resource="&resorts;Hotel"/>
       <rdfs:range rdf:resource="&xsd;string"/>
</rdf:Description>
<rdf:Description rdf:about="&resorts;situated_In">
       <rdfs:domain rdf:resource="&resorts;Hotel"/>
       <rdfs:range rdf:resource="&resorts;Region"/>
</rdf:Description>
<rdf:Description rdf:about="&resorts;Region">
       <rdfs:comment rdf:datatype="&xsd;string">
          Region where the hotel is located</rdfs:comment>
</rdf:Description>
<rdf:Description rdf:about="&resorts;North_Region">
       <rdfs:subClassOf rdf:resource="&resorts;Region"/>
</rdf:Description>
…
<rdf:Description rdf:about="&resorts;Hotel_Facilities">
       <rdfs:subClassOf rdf:resource="&resorts;Attractions"/>
</rdf:Description>
…
<rdf:Description rdf:about="&resorts;hasroomservice">
   <rdfs:subPropertyOf rdf:resource="&resorts;hashotelservice"/>
  </rdf:Description>
  <rdf:Description rdf:about="&resorts;hashotelservice">
   <rdfs:domain  rdf:resource="&resorts;hotel"/>
   <rdfs:range rdf:resource="&resorts;Attractions"/>
  </rdf:Description>
  …
  <rdf:Description rdf:about="&resorts;Convention_H">
   <rdfs:subclassOf rdf:resource="&resorts;Business_H"/>
  </rdf:Description>
<rdf:Description rdf:about="&resorts;Business_H">
   <rdfs:subclassOf rdf:resource="&resorts;Hotel"/>
  </rdf:Description>
```

</rdf:RDF>

V.6.1.2 Modèle de données (instances)

Ce modèle est l'ensemble d'annotations des documents spécifiés au regard de l'ontologie de domaine (Schéma). Dans l'exemple, nous avons sélectionné l'hotel « Cirta » de la classe « conference hotel », l'hotel « Touat » comme instance de la classe « convention hotel » et « Aurassi » comme instance de la classe « business hotel » Figure V.9.

Web documents	**Syntactic keywords**	**Equivalent classes (generated)**	**Semantic annotations**
Document1 Hotel :Cirta Location:Constantine Rating : 3	cirta hotel constantine, constantine hotels, reservation cirta, cirta.dz	[cirta hotel constantine, cirta resort constantine,cirta ritz constantine, cirta building constantine] , [constantine hotel, constantine resort, constantine ritz, constantine building], [reservation cirta],[cirta.dz]	Name spaces declarations … <Conference_hotel rdf:about="&resorts;Cirta"> <name rdf:datatype="&xsd ;string">Cirta</name> <situated_in rdf:resource="&resorts;north_region"/> <location rdf:resource="&resorts;constantine"/> … <hasroomservice rdf:resource="&resorts;wifi"/> <situated_In rdf:resource="&resorts;North_Region"/> … </Conference_hotel> … </rdf:RDF>
Document2 Hotel:Touat Location:Adrar Rating: 3	touat hotel, touat adrar, adrar hotel,hotel sahara, hotel oasis, algeria sud, algeria tourism	[touat hotel, touat resort, touat tourist, touat building], [touat adrar],[adrar hotel, adrar resort, adrar tourist, adrar ritz, adrar building],[hotel sahara, hotel oasis,resort sahara,resort oasis,tourism sahara, tourism oasis, building sahara, building oasis], [Algeria sud],[Algeria tourism]	Name spaces declarations … <Convention_hotel rdf:about="&resorts;touat"> <situated_in rdf:resource="&resorts;oasis_region"/> <location rdf:resource="&resorts;Adrar"/> … <hashotelservice rdf:resource="&resorts;restaurant"/> <hashotelservice rdf:resource="&resorts;hiking"/> <hasroomservice rdf:resource="&resorts;telephone"/> <situated_In rdf:resource="&resorts;Oasis_Region"/> … </Convention_hotel> … </rdf:RDF>
…	…	…	……
Document20 Hotel:Aurassi Location:Algiers Rating :5	algiers hotel, hotel alger, aurassi hotel, Algiers aurassi alger aurassi.	[Algiers, alger blanche], [algiers hotel, algiers resort, algiers building, algiers tourist],[aurassi hotel, aurassi resort,aurassi ritz,aurassi building, aurassi structure],[algiers aurassi],[alger aurassi]	Name spaces declarations … <Business_hotel rdf:about="&resorts;aurassi"> … <hashotelservice rdf:resource="&resorts;pool"/> <hashotelservice rdf:resource="&resorts;parking"/> <hasroomservice rdf:resource="&resorts;safetyroom"/> <ranking>4</ranking/> … </Business_hotel> …

Figure V.9: Le modèle de données (annotations de documents)

Les inférences sont basées sous classes pour (Entity1 et Entity2), et sous propriété pour (Entity3). Les annotations attachées dans un document donné sont assimilées comme une instanciation de l'ontologie. Le moteur de recherche sémantique pendant l'inférence retourne « Cirta » et « Touat » comme des réponses pertinentes aux requêtes de recherche d'un hotel de tout type. C'est une inférence basé sur la relation de sous classe.

Similairement, le raisonneur dérive des connaissances additionnelles du domaine, par exemple nous avons l'inférence que « Entity3 » est aussi un hotel, par l'exploitation des caractéristiques de « HasRoomService » qui est une sous propriété de la propriété « HasHotelService ». Comme le prédicat « HasHotelService » est défini sur le domaine « hotel » et a comme objet le co-domaine «Attraction », nous pouvons inférer que le building « Aurassi » est un hotel.

Si nous reprenons la requête utilisateur du paragraphe V.4 « Nom et Adresse des hôtels dans les Oasis Algérien avec les services Wifi Internet et Piscine. » alors nous pouvons lui associer le graphe conceptuel suivant :

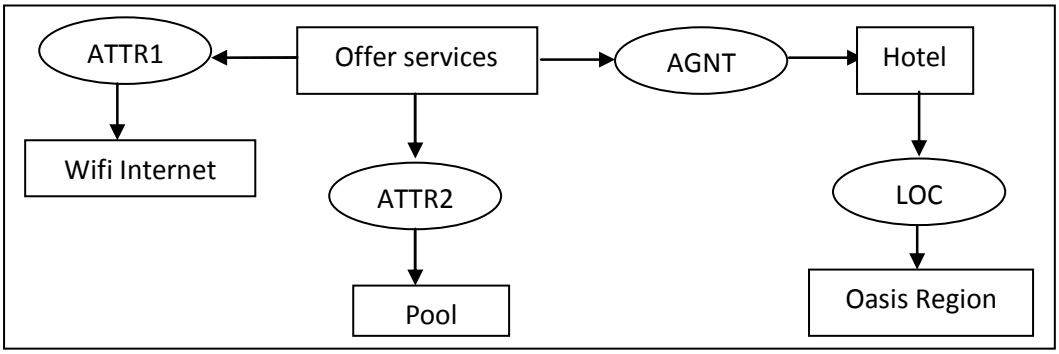


Figure V.10 : Graphe conceptuel de requête « Gr »

Un document annoté par le graphe « Gd » montré en figure V.11, est une réponse pertinente à la requête représentée par le graphe « Gr ». En effet il existe une correspondance entre les deux graphes qui vérifie les critères d'une opération de projection du graphe « Gd » dans le graphe « Gr », on peut déduire dans ce cas que le taux de rappel sera élevé au dépend de la mesure de précision.

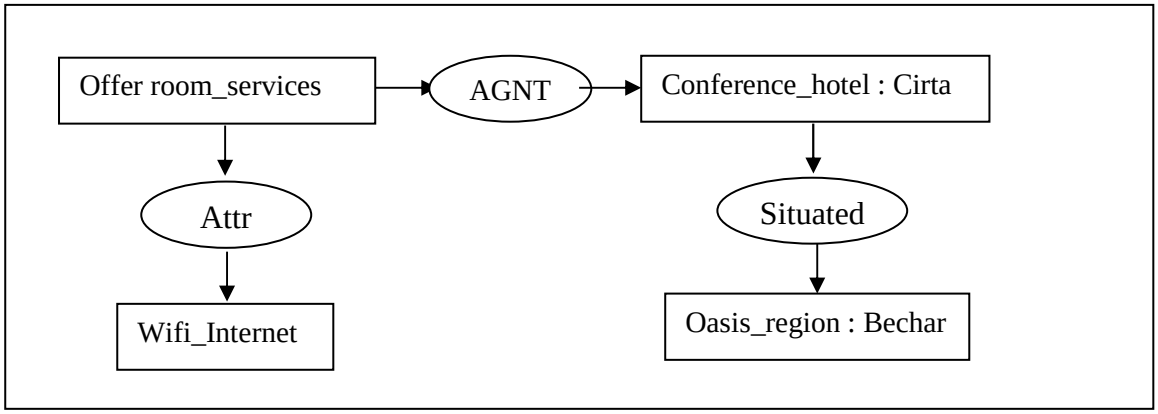


Figure V.11 : Graphe conceptuel d'annotation « Gd »

Entity1 has type :
-<http://Mydomain/Ontology/Inforesorts/**Cirta** rdf:type http://Mydomain/Ontology/Inforesorts/**Conference_hotel**>
-<http://Mydomain/Ontology/Inforesorts/**Cirta** rdf:type http://Mydomain/Ontology/Inforesorts/**Hotel**>
-<http://Mydomain/Ontology/Inforesorts/**Cirta** rdf:type rdfs:**source**

Conference_hotel has type :
-<http://Mydomain/Ontology/Inforesorts/**Conference_hotel** rdf:type rdfs:Class>
-<http://Mydomain/Ontology/Inforesorts/**Conference_hotel** rdf:type rdfs:Resource>

Entity2 has type :
-<http://Mydomain/Ontology/Inforesorts/**Touat** rdf:type http://Mydomain/Ontology/Inforesorts/**Convention_hotel**>
-<http://Mydomain/Ontology/Inforesorts/**Touat** rdf:type http://Mydomain/Ontology/Inforesorts/ **Hotel** >
-<http://Mydomain/Ontology/Inforesorts/**Touat** rdf:type rdfs:**source**

Convention_hotel has type :
-<http://Mydomain/Ontology/Inforesorts/**Convention_hotel** rdf:type rdfs:Class>
-<http://Mydomain/Ontology/Inforesorts/**Convention_hotel** rdf:type rdfs:Resource>

Entity3 has type :
-<http://Mydomain/Ontology/Inforesorts/**Aurassi** rdf:type http://Mydomain/Ontology/Inforesorts/**Building**>
-<http://Mydomain/Ontology/Inforesorts/**Aurassi** rdf:type http://Mydomain/Ontology/Inforesorts/ **Hotel** >
-<http://Mydomain/Ontology/Inforesorts/**Aurassi** rdf:type rdfs:Resource>

Figure V.12 : Inférence sur le type « Hotel »

Pour la mesure du rappel, la performance du modèle de raisonnement doit répondre à la nécessité de résoudre les problèmes majeurs de synonymie, d'hyponymie et d'hyperonymie des termes utilisés. Par ailleurs, pour la recherche sémantique la mesure de précision est très appréciable; ce résultat répond au problème de la polysémie de mots clés.

Si on note {Ds} et {Dq} les ensembles d'URIs de documents retrouvés, respectivement pour la recherche syntaxique et sémantique, et en fonction des préférences de l'utilisateur, il existe deux possibilités:

- Pour obtenir un important rappel, il est recommandé de considérer l'union des ensembles:

  {Dr} = {Ds} ∪ {Dq}; {Dr}: ensemble des URI avec un rappel maximal.

- Pour une meilleure précision, il est évident pour considérer l'intersection de ces ensembles:

  {Dp} = {Ds} {∩ Dq}; {Dp}: ensemble des URI avec une précision maximale.

V.7 Implémentation du Système Multi Agents

Pour l'implémentation nous avons utilisé les outils suivants :

V.7.1 Plate Forme JADE (Java Agent Development Framework)

C'est une plate forme multi agents développée en Java, permettant le développement d'applications basées systèmes multi agents et conformes aux spécifications de la norme FIPA, cet environnement permet aux agents d'exécuter des actions concurrentes et offre une bibliothèque de comportements implémentés en Java.

L'environnement logiciel JADEX est une extension de JADE, qui lui apporte le support du modèle BDI. L'agent est perçu comme une boite noire qui reçoit et émet des messages, c'est donc une architecture « Message-Driven » dans le sens où l'agent réagit aux messages et événements internes qui surviennent. Dans JADEX, il y a deux types de composants d'un agent :

- Composant Global : Il correspond aux mécanismes de réaction et de délibération qui associent les événements reçus avec les intentions(plans) sélectionnés de la bibliothèque des plans, les modules de réaction et de délibération sont responsables de chercher et de sélectionner les intentions et les exécuter depuis la librairie des intentions ou la structure contenant les intentions en cour d'exécution.

  Le module « Desire Deliberation », va choisir dans l'ensemble des désires (buts) ayant été déclenchés ceux qui vont effectivement être activés, suspendus ou bien abandonnés, et ce selon la politique choisie.
- Composant capacité : sont des unités regroupant des éléments BDI d'un agent (croyances, buts, plans, événements).

Les systèmes BDI représentent les croyances d'un agent sous forme de prédicats logiques du premier ordre ou sous forme de modèle relationnel, la base de croyances dans l'environnement JADEX joue un rôle actif dans l'exécution de l'agent. Un agent pour réaliser ses buts sélectionne et exécute des plans, lors des délibérations les croyances en cours sont prises en considération, les plans qui sont en cours d'exécution peuvent influencer les croyances, et de ce fait la modification des croyances peut provoquer des événements internes qui à leur tour impliquent l'adoption de nouveaux buts et l'exécution d'autres plans.

V.7.2 Composants d'agent Jadex

Le développement d'applications basées agents passe par la spécification de l'ensemble des agents, sur la plate forme *Jadex* la spécification d'un agent comporte deux types de fichier :

- Fichier XML de définition de l'agent ADF (Agent Definition File).
- Des classes Java pour l'implémentation des comportements (plans) de l'agent.

```
<agent name= "Interface">
<Beliefs>
…
<Goals>
…
<Plans>
…
</agent>
```

```
public class AfficherResultatsPlan extends Plan
{
        public void body()
        { …
        }
        …
        }
```

Figure V.13: Composantes d'un agent Jadex

L'ADF est un fichier XML utilisé pour définir l'agent, les éléments qu'il comporte et qui constituent son noyau consiste en des croyances (Beliefs), des buts (Goals), des Plans ainsi que des balises optionnelles comme <imports>, <expressions>, <capabilities> et autres.
La structure est illustré par la figure V.13, ainsi, pour son démarrage sur la plateforme *Jadex*, en premier il faut charger son ADF et ce grâce au composant JCC (Jadex Control Center). La figure V.14 illustre l'exemple de chargement de l'ADF de l'agent « Interface ».

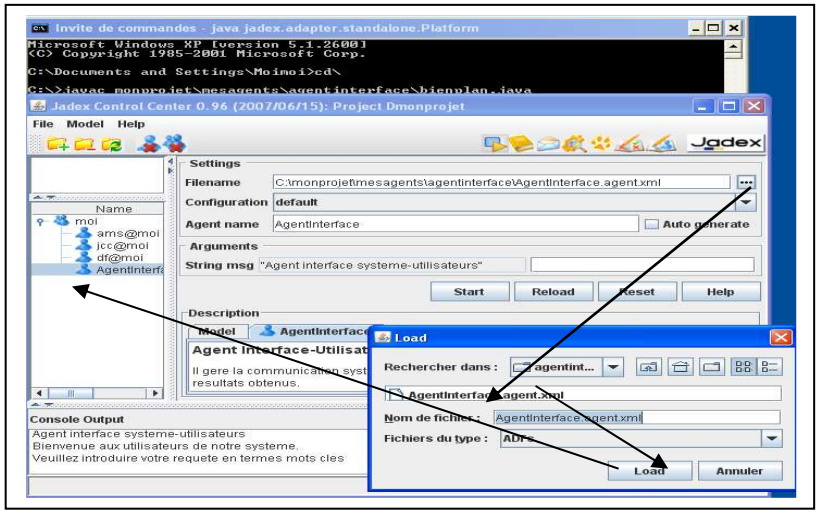


Figure V.14 : Chargement de l'ADF

Une fois l'ADF chargé, l'agent est initialisé par le contenu de son ADF. Il spécifie les éléments de capacités de l'agent, l'agent est un ensemble de (capabilities), muni d'un processus de raisonnement partagé, donc une capacité est similaire à l'agent lui-même et pour la définir nous utilisons aussi un fichier semblable à un ADF. Le chargement d'un ADF provoque la création des objets Java correspondant aux éléments XML, c'est-à-dire les croyances, les buts et les plans. L'ordre d'occurrence des éléments composant l'ADF est prédéfini, nous nous ne pouvons pas par exemple déclarer les plans avant de déclarer les croyances de l'agent, l'ordre d'apparition de ces éléments est schématisé par la figure V.15.

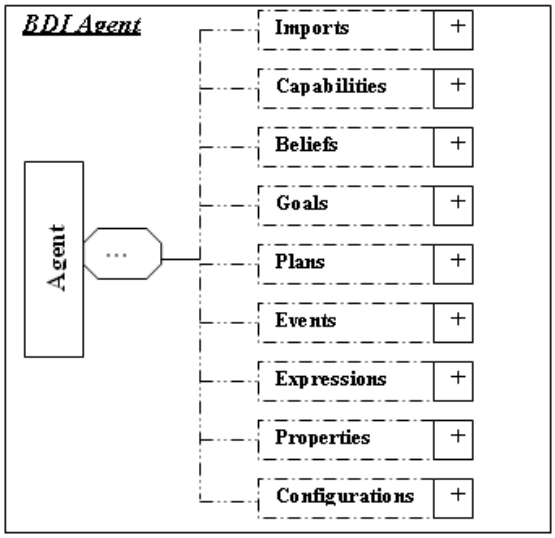


Figure V.15 : Eléments de l'ADF

V.7.2.1 Structure de L'ADF

a. Les croyances (beliefs)

La base des croyances de l'agent représente les connaissances dont il dispose à propos de son environnement, cette base est similaire à une structure de stockage de données qui permet aux différents plans de communiquer via des croyances partagées. Dans Jadex tous les objets Java peuvent être stockés comme des croyances dans cette base.

Jadex distingue entre deux types de croyances, dans le premier type, l'utilisateur ne peut stocker qu'un seul fait (la balise <belief>) alors que le deuxième type utilise des « beliefset » pour plusieurs faits, (balise <beliefs>). Les croyances représentent les faits connus par l'agent et définies dans l'ADF, la base de croyances est accédée par les plans.

b. Les buts (desires)

L'un des concepts clé du paradigme agent est la programmation orientée but, ce concept dénote l'engagement de l'agent à satisfaire un objectif pour cela l'agent va tenter les différentes possibilités pour atteindre son but.

Jadex appuie l'idée du concept BDI qui définit les buts comme des désirs concrets et momentanés de l'agent. Le cycle de vie d'un but comporte les états « ***option », « actif »*** et ***« suspendu ».*** Ces états sont gérés par un mécanisme de délibération spécifique et en relations avec le contexte et les croyances de l'agent.

c. Les intentions (Plans)

Un agent Jadex est essentiellement basé événements qui lorsqu'ils occurrent déclenche l'agent, cette réaction ne veut pas dire que l'agent est purement réactif, en effet Jadex ne supporte pas uniquement les événements extérieurs (Messages ACL) qui représentent une communication entre agents, mais aussi différents types d'événement interne à l'agent qui sont des types de communication interne à l'agent et qui sont nécessaire lorsque plusieurs plans veulent échanger des informations.

Les événements sont traités par des plans, l'occurrence d'un événement interne est caractérisée par le fait qu'une information doit être communiquée à des plans qui ont déclarés leur intérêt à ce type d'événement. Tous les événements extérieurs (messages en émission et en réception) doivent être spécifiés dan l'ADF de l'agent.

L'exemple ci-dessous est tirée de l'ADF de l'agent « Query »

- active: Booléen qui indique si une croyance «état actif».
- préférences: les croyances de l'agent qui pouvant être un liste des préférences utilisateur, des seuils de calculs, etc.
- Feedbacks : des croyances relatives à la pertinence des résultats obtenus.
- Ex_F: un fichier XML, pour les échanges de messages.
- $Word_1$, $Word_2$,…, $Word_k$ sont des chaines de mots clés, des synonyms, des hyponymes et des hyperonymes.

```
<Agent xmlns="http://jadex.sourceforge.net/jadex"
  xmlns:xsi="http://www.w3.org/2001XMLScheminstance"
  xsi :schemalocation="http://jadex.sourceforge.net/jadex
  http://jadex.sourceforge.net/jadex-0.96.xsd"
  name ="Query"
  package="myprojet.mesagents.queryagent"
  <beliefs>
```

```
<belief name= "msg" class="string" exported="True">
<fact>Query  agent</fact>
</belief>
<belief name="state" class="Boolean">
<fact>true</fact> </belief>
<belief name="preferences" class="String">
<fact> threshold-Res</fact>
<fact> search-eng</fact></belief>
<belief name="feedback" class="Boolean">
<fact>true</fact>
</belief>
<beliefset name="keywords" class="string">
<fact>word1</fact>
<fact>word2</fact>
…
</beliefset>
</beliefs>
</Agent>
```

L'un des objectifs de l'agent « Query » est de substituer les mots-clés par leurs synonymes, hyponymes et hyperonymes. Ce but est réalisé lorsque les «feedback» et la croyance sont en état valides. Ainsi, la partie de l'ADF correspondant à cette fonction est:

```
<goals>
  <performgoal name="substit"
  <parameter name="keyword" class="String"/>
  <parameter name="result" class= "String"/>
  <contextcondition>
  $beliefbase.Est_pret&&$beliefbase.feedbacks
  </contextcondition>
  </performgoal>
</goals>
```

Conclusion

Dans ce chapitre nous avons exposé l'étude de cas pour tester le modèle selon les trois aspects décris auparavant. Pour le premier aspect du raisonnement nous avons segmenté un texte de trois paragraphes, le résultat produit deux segments sémantiquement différents.

Le deuxième aspect du raisonnement repose sur un calcul du rapprochement des documents résultats par rapport à la requete utilisateurs, nous avons testé notre mesure et l'avons comparée à celle de Wu Palmer, les résultats sont satisfaisants. Le troisième aspect du raisonnement concerne le mécanisme d'inférence. Nous avons montré un type de raisonnement via Jena (transitivité de classes, et type de domaine et de co-domaine des propriétés). Cette inférence est la base de la recherche sémantique qui se traduit par la recherche d'une opération de projection entre le graphe requête et le graphe d'annotation du document. La projection, s'effectue par la recherche d'opérations de spécialisation de concepts et de propriétés des les deux graphes, si par exemple nous avons dans un graphe d'annotation « Gd » une projection d'un graphe de requête « Gr », alors nous somme en mesure d'affirmer que le document annoté par « Gd » est une réponse pertinente à la requête représentée par « Gr » avec une précision élevé.

Conclusion Générale

Le modèle de notre contribution a l'objectif d'affecter un système multi agents de capacités de raisonnements pour accomplir une recherche sémantique d'informations sur le web. Le raisonnement devrait se matérialiser par une amélioration significative des mesures de rappel et de précision qui sont très interdépendantes car l'augmentation de l'une se répercute par une diminution des performances de l'autre.

Tenant compte de cette relation inversement proportionnelle, nous avons construit ce modèle par des démarches hybrides pour satisfaire au mieux et de manière adéquate ces deux mesures. Nous avons choisi le paradigme agent pour raisonner et l'ontologie pour représenter les connaissances du domaine, ce choix est motivé par l'opportunité de reprendre la notion de résolution coopérative de problèmes complexes et distribués.

L'architecture du système montre que les agents ainsi conçus ont des compétences individuelles et des connaissances relatives à la résolution de parties du problème. Ils coopèrent ensemble pour l'objectif commun celui de présenter à l'utilisateur les meilleurs résultats possibles en termes de documents relevant vis-à-vis de ses besoins.

Notre modèle de recherche combine une recherche sémantique basée ontologie, et une recherche classique à base de mots clés, c'est une recherche syntaxique qui exploite les relations entre les mots clés et des ensembles de synonymes, d'hyponymes et d'hyperonymes (*synsets*) définis dans la taxonomie WordNet pour réaliser des expansions de requêtes dans le but d'avoir un plus de rappel. La requête générée en langage d'interrogation d'ontologie à partir de mots clés est ensuite exécutée pour retrouver les documents annotés par ces entités et qui constituent des réponses pertinentes.

Le support de la recherche sémantique, est un mécanisme de raisonnement basé sur le type de ressources recherchées, ainsi nous nous somme basé sur les inférences de sous classes et de sous propriétés pour rechercher une projection entre la requête et le document.

Une nouvelle mesure de similarité a été proposée, elle est fondée sur la structure de la taxonomie WordNet pour évaluer les similarités entre les mots clés et les *sysnets* correspondants, cette similarité est ensuite utilisée pour calculer le rapprochement des résultats vis-à-vis des requêtes. La formule du cosinus très connue est utilisée pour calculer la similarité Requête-Document pour décider de la pertinence des documents, et pour effectuer le classement final.

Nous notons ici que les représentations formelles souffrent de certaines limites, qui se traduit par l'incapacité de représenter certaines connaissances complexes, le système hérite donc de problèmes inhérents aux processus de construction et de partage des ontologies.

Les perspectives de recherche dans ce domaine sont diverses et encourageantes, particulièrement concernant le modèle proposé nous trouvons utile d'enrichir les connaissances de l'agent qui gère la requête par une base de connaissances rassemblant différentes techniques de traitement et de formulation de requêtes, par exemple un ensemble de règles explicites et des politiques de décision sous forme de prédicats logiques. Il est vrai que plus la requête est bien formulée, meilleurs seront les résultats obtenus.

NOS CONTRIBUTIONS

[Nes,2012a] D.Nessah, O.Kazar : "Document Analysis to Provide Semantic Metadata based ontologies", The 3rd International Conference on Multimedia Computing and Systems, pp:725-731,Tangier Morocco, IEEE, 10-12 may 2012.

[Nes,2012b] D.Nessah, O.Kazar: "A Multi-Agents System for Semantic Annotation Based Conceptual Graph Formalism." IJWA V4(3): pp:134-150, September 2012.

[Nes,2013] D.Nessah, O. Kazar: "An improved semantic information searching scheme based multi-agent system and an innovative similarity measure." IJMSO V8(4): pp:282-297, December 2013.

BIBLIOGRAPHIE

[Ada,1974] Adamson, G., Boreham J. "The use of an association measure based on character structure to identify semantically related pairs of words and document titles". In Information Storage and Retrieval,10, pp: 253–60, 1974.

[Ama,2007] Florence Amardeilh, "Web Sémantique et Informatique Linguistique : propositions méthodologiques et réalisation d'une plateforme logicielle", Thèse de doctorat. Université Paris X – Nanterre, Mai 2007.

[And,1971] Andrews, K. "The Development of a Fast Conflation Algorithm for English.", Dissertation for the Diploma in Computer Science, Computer Laboratory, University of Cambridge, 1971.

[Asu,1999] Asunción Gómez Pérez, et V. Richard Benjamins, "Overview of Knowledge Sharing and Reuse Components: Ontologies and problem-solving methods". Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods. Stockholm, Sweden, August, 1999.

[Baa,2003] Franz Baader, Warner Nutt "Basic Description Logics" , the Description Logic Handbook, Theory, Implementation, Applications , pp:43-95, Cambridge University Press, UK, 2003.

[Bae,1999] R. Baeza-Yates, B. Ribeiro-Neto. "Modern Information Retrieval." ACM Press Series /Addison-Wesley, 1999.

[Bar,2009] Patrick Barlatier, "Conception et implantation d'un modèle de raisonnement sur les contextes basé sur une théorie des types et utilisant une ontologie de domaine", Thèse de doctorat. Université de Savoie. Juillet 2009.

[Bec,2002] Bechhofer, S., Carr, L., Goble, C., Kampa, S. and Miles-Board, T. "The Semantics of Semantic Annotation." In: Proceedings of the 1st International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems. pp:1151-1167 . 2002.

[Ber,2001] Berners-Lee T., Hendler J. et Lasilla O. "The semantic web", Scientific American Vol. 284/5 pp:34-43. May 2001.

[Bir,2012] Aliaksandr Birukou, Enrico Blanzieri , PaoloGiorgini "Implicit: a multi-agent recommendation system for web search", Multi-Agent Syst 24 pp:141-174. 2012.

[Boo,1983] Bookstein A. "Outline of a general probabilistic retrieval model." Journal of *Documentation*, vol. 39/2: pp:63-72, 1983.

[Bor,2003] Borislav Popov, Atanas Kiryakov, Angel Kirilov, Dimitar Manov, Damyan Ognyanoff, Miroslav Goranov, "KIM - Semantic Annotation Platform." International Semantic Web Conference. pp-834-849. 2003.

[Bra,2006] Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, François Yergeau, and John Cowan, "Extensible Markup Language (XML) 1.1 (Second Edition)." Technical report, 2006. W3C. http://www.w3.org/TR/xml11/

[Brd,2012] Broda, B., Kurc, R. and Piasecki, M. "Evaluation method for automated Wordnet expansion", *International Joint Conferences, SIIS 2012*, , pp.422–433. Warsaw, Poland 2012.

[Bri,2004] Brickley, D., Guha, R.V., (Eds.) "RDF Vocabulary Description Language 1.0: RDF Schema." W3C Recommendation.(2004). http://www.w3.org/TR/rdf-schema/

[Bro,2005] Brooke Abrahams ; Wei Dai "Architecture for automated annotation and ontology based querying of semantic web resources", Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence. 2005.

[Bus,2013] Davide Buscaldi, Haïfa Zargayouna ,"YaSemIR: Yet Another Semantic Information Retrieval System ", ACM ESAIR'13, October 2013, San Francisco, CA, USA.

[Cas,2007] Pablo Castells, Miriam Fernández, and David Vallet "An Adaptation of the Vector-Space Model for Ontology- Based Information Retrieval", Transactions on Knowledge and data engineering. V 19/2, IEEE, Feb. 2007.

[Cas,2011] N. Casellas , "Methodologies, Tools and Languages for Ontology Design". Law, Governance and Technology Series 3, Springer Science + Business Media B.V. 2011.

[Ces,2003] C.Cesarano, A. d'Acierno, A.Picariello, "An intelligent search agent system for semantic information retrieval on the internet", ACM WIDM'03 New Orleans Louisiana USA 2003.

[Cha,2010] Chauhan, R., Goudar, R., Rathore, R., Singh, P. and Rao, S. "Ontology based automatic query expansion for semantic information retrieval in sports domain", *ICECCS* 2012, Kochi India, pp:422–433.2012.

[Cir,2001] Ciravegna F. Adaptive Information Extraction from Text by Rule Induction and Generalisation, in Proceedings of the 17th International Joint Conference on Artificial Intelligence . IJCAI , Seattle. 2001.

[Cro,2007] Madalina Croitoru, et Kees van Deemter. "A Conceptual Graph Approach to the Generation of Referring Expressions», IJCAI, Hyderabad -India Jan 2007.

[Daw,1974] Dawson, J.L. "Suffix Removal and Word Conflation". ALLC Bulletin, Michaelmas pp:33-46, 1974.

[Dea,2004] M. Dean, D. Connolly, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F.

Patel-Schneider, and L. A. Stein. “OWL Web Ontology Language Reference.” Technical report, 2004. W3C Recommendation. http://www.w3.org/TR/owl-ref/

[Deb,2012] Jean Debaecker “ De l’usage des métadonnées dans l’objet sonore”, Thèse de doctorat. Université Charles-de-Gaulle, Lille 3, Octobre 2012.

[Erm,2000] Ermine JL, “La gestion des connaissances, un levier stratégique pour les entreprises”. IC’00, Toulouse, 2000.

[Esp,2007] B. Espinasse, S. Fournier et F. Freitas, “AGATHE : une architecture générique à base d’agents et d’ontologies pour la collecte d’information sur domaines restreints du Web.” pp : 367-383 CORIA Mars 2007.

[Fel,2010] Fellbaum, C. “WordNet”, Princeton University, NJ, USA, pp.231–243. 2010.

[Fer,1995] J.Ferber, “Les systèmes multi-Agents – vers une intelligence collective”, inter éditions 1995.

[Fra,1992] Frakes, W.B. “Stemming Algorithms.” In: Frakes, W.B., Baeza-Yates, R. (eds.): “Information Retrieval Data Structures and Algorithms.” Prentice Hall, New Jersey p. 131-160, 1992.

[Fuh,1989] Fuhr, N. “Models for retrieval with probabilistic indexing.” Information processing and management, vol. 25/1, pp:55–72, 1989.

[Gar,2004] Lars Marius Garshol, “Metadata? Thesauri? Taxonomies? , Topic maps! Making sense of it all.” Journal of Information Science, pp. 378–39, 30(4) 2004.

[Gri,2008] Malika Grim-Yefsah, “Gestion des connaissances et externalisation informatique apports managériaux et techniques pour l’amélioration du processus de transition ”, Thèse de doctorat. Université Paris-Dauphine. Novembre 2008.

[Gru,1993] T. R. Gruber. “A translation approach to portable ontologies specifications”. Knowledge Acquisition, 5(2) :pp:199-220, 1993.

[Gru,2009] T. R. Gruber, “Ontology”. The Encyclopedia of Database Systems, Springer-Verlag. 2009.

[Gua,1998] Nicola Guarino “Formal Ontology and Information Systems”. Proceedings of FOIS’98, Amsterdam, IOS Press, pp:3-15. Italy, June 1998.

[Gui,2004] McGuinness, D. and van Harmelen, F. (Eds) “*OWL WebOntology Language Overview”: W3C ecommendation*. (2004). http://www.w3.org/TR/owl-features

[Haa,2004] Kenneth Haas, “ Context for Semantic Metadata”, MM’04, New York, USA. ACM October 10–16, 2004.

[Han,2003] Siegfried Handschuh, Steffen Staab, Rudi Studer: “Leveraging Metadata Creation for the Semantic Web with CREAM.” KI: pp:19-33. 2003.

[Hat,1997] Jean Paul Haton et al. “Le raisonnement en intelligence artificielle, modèles techniques et architectures pour les systèmes à base de connaissances. ” Editions Dunod ,480p, Décembre 1997.

[Hen,2008] Hendler, James A., et Frank van Harmelen; “The Semantic Web: webizing knowledge representation”,pp:821-839. 2008.

[Hol,2002] C.W.Holsapple, K.D.Joshi, “Knowledge Management: A Threefold Framework ”, the information society, Vol 18, pp 18-47, Francis and Taylor, 2002.

[Hua,2012] Huang, Y., Gan, M. and Jiang, R. “Ontology-based genes similarity calculation with TF-IDF”, I*CICA 2012*, Chengde, China, pp.600–607. 2012.

[Jar,2002] Imed Jarras et Brahim Chaib-Draa, , “ Aperçu sur les systèmes multi-agents ” Montréal juillet 2002.

[Jia,1998] J.Jiang, D.Conrath, “Semantic similarity based on corpus statistics and lexical taxonomy”, In International conference on research in computational linguistics , p.19-33 Taiwan 1998.

[Ker,2004] Larry Kershberg et al. “"Knowledge Sifter: Agent based ontology-driven search over heterogeneous databases using semantic web services”, Goerge Mason University USA 2004.

[Kly,2004] G. Klyne et J. J. Carroll. “Resource Description Framework (RDF): Concepts and Abstract Syntax.”, Technical report, 2004. W3C Recommendation. http://www.w3.org/TR/rdf-concepts/

[Kro,1993] Krovetz, R. “Viewing morphology as an inference process”. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp: 191-202, 1993.

[Kuh,1990] Kuhlthau, C.; Turock, B.; George, M. & Belvin, R. “Validating a model of the search process: a comparison of academic, public and school library users”, Library & Inf. Science Research, pp:1-12, 1990.

[Kyu,2004] Kyung Hoon Yang, David Olson, Jaekyung Kim, “Comparison of first order predicate logic, fuzzy logic and non-monotonic logic as knowledge representation methodology”. Expert Systems with Applications, Elsevier Ltd, pp:501–519, 2004.

[Lab,2007] A .Labadié, J. Chauché : “Segmentation Thématique par Calcul de Distance Sémantique.” pp : 355-366 EGC Namur, Belgique .2007.

[Lam,2010] Lamari, M. “Le transfert intergénérationnel des connaissances tacites: les concepts utilisés et les évidences empiriques démontrées”, Télescope, vol. 16, nº1, pp : 39-65 ,2010.

[Lea,1998] C.Leacock, M. Chodorow “Combining local context and WordNet similarity for word sense identification in WordNet”, In C. Fellbaum editor, An electronic lexical database Pp:265-283 MIT Press 1998.

[Lew,1990] Lewis D. D. and Croft.W. B, “Term clustering of syntactic phrases”. Université: Massachusetts, Collins Technique Report pp: 71-90, 1990.

[Li,2003] Y.Li, Z.A.Bandar, D.McLean. “An Approach for Measuring similarity between words using multiple information sources” , IEEE Tras. On knowledge and data engineering, 15/4 pp:871-882. 2003.

[Lia,2011] Yongxin Liao, Mario Lezoche, Hervé Panetto, Nacer Boudjlida, “Semantic Annotation Model Definition for Systems, Interoperability” OTM 2011 Workshops 2011 - 6th International Workshop on Enterprise Integration, Interoperability and Networking (EI2N)

Hersonissos, Crete : Greece 2011.

[Lin,1998] D.Lin, “An information- theoretic definition of similarity”, Proc. Of the international conference on machine learning , ICML’98 1998.

[Mar,2004] Jean Martinet. “Un modèle vectoriel relationnel de recherche d’information adapté aux images”, Université Joseph Fourier Grenoble I, décembre 2004.

[Mar,2005] Marja-R. Koivunen: “Annotea and Semantic Web Supported Collaboration”. ESWC, Greece. May 2005.

[Maz,2001] Hamza Mazouzi, “ Ingénierie des protocoles d’interaction: des systèmes distribués aux systèmes multi-agents ”, thèse. Université paris IX Déc. 2001.

[Mil,1989] R.Rada, H.Mili,E.Bicknell, M.Blettner, “Development and application of a metric on semantic nets”, IEEE Trans. on systems man and cybernetics 19/1 p.17-30 Jan/Feb 1989.

[Min,1987] Miner E., “Some Theoretical and Methodological Topics for Comparative Literature” p:137, 1987.

[Moa,2012] Néjib Moalla, Hervé Panetto, Xavier Boucher, “ Interopérabilité et partage de connaissances ”, Ingénierie des Systèmes d'Information (ISI) 17/4, pp :7-17, Aout 2012.

[Mon,2008] Monticolo Davy “Une approche organisationnelle pour la conception d’un système de gestion des connaissances fondé sur le paradigme agent”. Thèse de doctorat. Université de Technologie de Belfort Montbéliard et Université de Franche Comté. Février 2008.

[Mug,1996] M. L. Mugnier et M. Chein. “Représenter des connaissances et raisonner avec des graphes”, *Revue d’Intelligence Artificielle (R.I.A)*, 10(1) , pp : 7–56, 1996.

[Nar,2003] D.Nardi, et R.J Brachman “An Introduction to Description Logics” , the Description Logic Handbook, Theory, Implementation, Applications pp 1-40, Cambridge Univ.Press, UK, 2003.

[Pai,1996] Paice, C.D. “Method for evaluation of stemming algorithms based on error counting.” Journal of the American Society for Information Science 47 (8). 1996.

[Pet,2006] David Peterson, Paul V. Biron, Ashok Malhotra, and C. M. Sperberg-McQueen, “XML Schema 1.1 Part 2: Datatypes.”. Technical report, 2006. W3C. http://www.w3.org/TR/xmlschema11-2/

[Pir,2010] Pirro, G. and Euzenat, J. “A feature and information theoretic framework for semantic similarity and relatedness”, *ISWC 2010*, pp.615–630. Springer Shanghai, China 2010.

[Pol,1966] Polanyi M. “The tacit dimension”. First published, Doubleday & CO 1966.

[Rad,1989] R.Rada, ,E.Bicknell “Ranking documents with a thesaurus”, JASI6 40/5 pp:304-310 September 1989.

[Rag,1986] V.V. Raghavan and S.K.M.Wong. “A critical analysis of vector space model for information retrieval.” Journal of the American Society for Information Science, 37/5 p :279–287, 1986.

[Rai,2008] Thomas Raimbault, « Transition de modèles de connaissances. Un système de connaissance fondé sur OWL, Graphes conceptuels et UML», Thèse de doctorat. Université de Nantes.

Septembre 2008.

[Rei,2005] Reix R. “Systèmes d’information et management des organisations”, Vuibert 2005.

[Res,1995] Resnik P. “Using information content to evaluate semantic similarity in a taxonomy.” Proceeding of the 14th international joint conference on artificial intelligence. pp:448-453 Montreal QC Canada 1995.

[Res,1999] Resnik P. “Semantic similarity in a taxonomy: An information based measure and its application to problems of ambiguity in natural language.” Journal of artificial intelligence research, pp:95-130 July 1999.

[Rij,1979] C. J. C.J.Van Rijsbergen, “Information Retrieval” 2nd Edt. Butterworths: London, 1979.

[Sal, 1988] Salton G. and Buckley, C. “On the use of spreading activation methods in automatic information retrieval.” 11th ACM-SIGIR Conference. pp: 147-160, 1988.

[Sal,1971] Salton G. “The SMART Retrieval System.”, Prentice Hall, 1971.

[Sal,1986] Salton G. “Another look at automatic text-retrieval systems”ACM 29.7, pp:648-656, 1986.

[Sal,1987] Salton G., Buckley C., “Term Weighting Approaches in Automatic Text Retrieval”, Cornell University, Ithaca, NY, 1987. Information processing & Management Vol 24/5 pp:513-523

[San,2012] Sandhya, N. and Govardhan, A. (2012) “Analysis of similarity measures with Wordnet based text document clustering”’, *ICISDIA 2012*, Visakhapatnam, India, pp:703–714. 2012.

[Sav,1993] Savoy, J. “Stemming of French words based on grammatical categories.”, Journal of the American Society for Information Science, 44(1), pp: 1-9, 1993.

[Sch,1999] Schreiber G. et al. “Knowledge engineering and management: the CommonKADS Methodology”. MIT presse 1999.

[Sch,2005] Didier Schwab. “Approche hybride –lexicale et thématique – pour la modélisation, la détection et l’exploitation des fonctions lexicales en vue de l’analyse sémantique de texte ”, Thèse doctorat. Université Montpellier II Décembre 2005.

[Sec,2004] N.Seco, T.Vaele, J.Hayes, “An Intrinsic information content metric for semantic similarity in WordNet”, Tech.report University College Dublin Ireland 2004.

[Sed,2010] Md Hanif Seddiqui, Masaki Aono, “Metric of intrinsic information content for measuring semantic similarity in an ontolgy”, Proc 7th Asia-Pacific conference on conceptual modeling. Brisbane, Austria 2010.

[Sel,1997] Selberg E. “Information Retrieval Advances using Relevance Feedback.” UW Dept.CSE General Exam. 1997.

[She,2012] Shenoy, K.M., Shet, K.C. and Acharya, U.D. ‘A new similarity measure for taxonomy based on edge counting’, *International Journal of Web & Semantic Technology*, Vol. 3/4, 2012.

[Sow,2000] J.F. Sowa “Knowledge Representation: Logical, Philosophical, and Computational Foundations” Pacific Grove, CA: Brooks/Cole, hardbound, ISBN 0-534-94965-7 . 2000.

[Sow,2013] John F. Sowa: “Semantic Networks”, http://www.jfsowa.com/pubs/semnet.htm

October 2013.

[Spa,1974] Sparck Jones, K. “Automatic indexing.” J. Doc. 30/4, pp: 393-432, 1974.

[Spa,2004] Sparck Jones, K. “A statistical interpretation of term specificity and its Application In Retrieval”, Journal of Documentation Vol 60/5 pp:493-502 2004.

[Sta,2009] Staab S., Studer R. «Handbook on Ontologies. » Springer, (2nd edition), 2009.

[Stu,1995] Stuart J Russel, Peter Norvig, “Artificial intelligence : A modern Approach”, 1995.

[Stu,1998] Studer R., Benjamins V.R. et Fensel D., «Knowledge engineering: principles and methods», in IEEE Transactions on Data and Knowledge Engineering, 25(1&2), pp:161-197. 1998.

[Tan,2005] Siddiqui, T.J, “Integrating notion of agency and semantics in information retrieval: an intelligent multi-agent model”»,
Intelligent Systems Design and Applications, pp: 160-165, September 2005.

[Thi,2010] Mouhamadou THIAM : Annotation Sémantique de Documents Semi structurées pour la Recherche d’Information. PhD thesis. Universities of South Paris and Gaston Berger. December 2010.

[Tho,2006] Henry S. Thompson, C. M. Sperberg-McQueen, Shudi (Sandy) Gao, Noah Mendelsohn, David Beech, and Murray Maloney, « XML Schema 1.1 Part 1: Structures». Technical report, 2006. W3C. http://www.w3.org/TR/xmlschema11-1/

[Toc,2007] Toch and Gal, A. “A semantic approach to approximate service retrieval”, *ACM Transactions on Internet Technology*, Vol. 8/1, ACM, New York.2007.

[Tua,2006] Tuan Dung CAO, “Exploitation du web sémantique pour la veille technologique”, Thèse de doctorat. Université Nice-Sophia Antipolis - UFR, Novembre 2006.

[Usc,1996] M. Uschold, « Building ontologies: toward an unified methodology», in proceedings of the 16th conference of the British Computer Society Specialist Group and Expert Systems.
Cambridge UK, 1996.

[Vac,2005] Václav Snášel, Pavel Moravec, Jaroslav Pokorný “WordNet Ontology Based Model for Web Retrieval.” Proceedings of the 2005 International Workshop on Challenges in Web Information Retrieval and Integration (WIRI’05) IEEE 2005.

[Var,2002] Maria Vargas, Enrico Motta et al “MnM Ontology driven tool for semantic markup.” In Proceedings of the workshop Semantic Authoring Annotation & Knowledge Markup. 2002.

[Var,2005] G.Varelas, E.Voutsakis, P.Raftopoulou, E.Petrakis, E.Milios, “Semantic similarity methods in WordNet and their application to information retrieval on the web”, ACM WIDM’S05
Bremen Germany November 2005.

[Wei,1999] G. Weiss, “Multi-agent systems: A modern Approach to distributed artificial intelligence”, MIT Press Cambridge UK 1999.

[Wog,1999] L. Woguia, S. Pierre, C.L Nguyen, “Modélisation d'un outil multi-agent d'aide à la recherche

d'informations sur Internet." Proceedings of the 1999 IEEE Canadian Conference on Electrical and Computer Engineering Shaw Conference Center, Edmonton, Alberta, Canada May 1999.

[Woo,2002] M.Wooldridge, "An Introduction to multi-agent systems", University of Liverpool UK. August 2002.

[Wu,1994] Z.Wu, M. Palmer, "Verb semantics and lexical selection", 32nd Annual meeting of the association for computational linguistics, ACL'94 p.133-138 Las Cruces, New Mexico, 1994.

[Wu,1994] Z.Wu, M. Palmer, "Verb semantics and lexical selection", 32nd Annual meeting of the association for computational linguistics, ACL'94 pp:133-138
Las Cruces, New Mexico, 1994.

[Yae,2009] Yaël Champclaux, "Un modèle de recherche d'information basé sur les graphes et les similarités structurelles pour l'amélioration du processus de recherche d'information ", Thèse de Doctorat, Université de Toulouse, décembre 2009.

[Yu,2011] Yu, C. and Yan, L. "Comparative research on methodologies for domain ontology development", ICIC 2011, Zhengzhou, China, pp:349–356. 2011.

[Yue,2009] Yue. Ma, L. Audibert, A. Nazarenko : "Ontologie Etendues pour l'Annotation Sémantique." pp :205-216, IC. Hammamet, Tunisia 2009.

[Zip,1949] G. Zipf. "Human Behaviour and the Principle of Least Effort." Addison-Wesley, 1949.