



HAL
open science

Web sémantique et Mémoire d expériences sur l analyse du transcriptome

Khaled Mohamed Khelif

► **To cite this version:**

Khaled Mohamed Khelif. Web sémantique et Mémoire d expériences sur l analyse du transcriptome. Réseaux et télécommunications [cs.NI]. Université Nice Sophia Antipolis, 2006. Français. NNT : . tel-00328114

HAL Id: tel-00328114

<https://theses.hal.science/tel-00328114>

Submitted on 9 Oct 2008

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

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

UNIVERSITE DE NICE-SOPHIA ANTIPOLIS - UFR Sciences
Ecole Doctorale Sciences et Technologies de l'Information et de la
Communication (S.T.I.C)

T H È S E

pour obtenir le grade de
Docteur en Sciences
de l'UNIVERSITE de Nice-Sophia Antipolis

Spécialité

Informatique

présentée et soutenue publiquement par

Mohamed Khaled KHELIF

Le 4 avril 2006

**Web sémantique et mémoire d'expériences pour
l'analyse du transcriptome**

Thèse dirigée par *Rose Dieng-Kuntz*

Jury :

Président	Peter Sander
Rapporteurs	Amedeo Napoli Gilles Kassel
Examineurs	Nathalie Aussenac-Gilles Pascal Barbry

Table des matières

Introduction	1
Chapitre 1 - État de l'art	5
1. Le web à la recherche d'une sémantique	7
1.1. Introduction et définition	7
1.2. Les principales composantes du web sémantique	7
1.2.1. La représentation des connaissances	8
1.2.1.1. XML (eXtensible Markup Language)	8
1.2.1.2. RDF (Ressource Description Framework)	9
1.2.1.3. RDFS (Ressource Description Framework Schema)	10
1.2.1.4. DAML+OIL	10
1.2.1.5. OWL (Web Ontology Language)	10
1.2.2. Les ontologies	11
1.2.2.1. A la recherche d'une définition	11
1.2.2.2. Les différents types d'ontologies	12
1.2.2.3. La réutilisation des ontologies	13
1.2.3. Les annotations sémantiques	14
1.2.3.1. Annotea	15
1.2.3.2. MnM	15
1.2.3.3. KIM	15
1.2.3.4. S-CREAM	16
1.3. La recherche d'information guidée par les ontologies	16
1.4. Des ontologies pour un web sémantique biomédical	17
1.4.1. GALEN	18
1.4.2. Menelas	18
1.4.3. Gene Ontology	18
1.4.4. SNOMED	18
1.4.5. UMLS	19
2. Le web sémantique d'entreprise	19
2.1. Mémoire d'entreprise et Web Sémantique	20
2.1.1. Définition de mémoire d'entreprise	20
2.1.2. L'approche ACACIA	21
2.1.2.1. Samovar	22
2.1.2.2. CoMMA	23
2.1.2.3. CORESE	23
2.1.3. Autre approche pour le développement des WSO	23
3. Extraction des connaissances à partir des textes	24
3.1. Quelques aspects sur le TALN	25
3.1.1. Les différentes étapes de l'analyse d'un texte	25
3.1.2. Notion de syntagme	26
3.1.3. Variations liées aux syntagmes	26

3.2.	Les approches d'extraction de candidats-termes	27
3.2.1.	Les approches statistiques	27
3.2.2.	Les approches syntaxiques	27
3.2.3.	Les approches mixtes	28
3.3.	Les approches d'extraction de relations	29
3.3.1.	Extraction des relations par étude statistique	29
3.3.2.	Extraction des relations par exploitation des contextes syntaxiques	29
3.3.3.	Extraction des relations par l'utilisation des marqueurs	30
4.	Fouille des textes pour le domaine biologique	31
4.1.	Rôles des techniques de TALN dans le domaine biomédical	31
4.1.1.	Construction des bases de connaissance biologiques	31
4.1.2.	La recherche d'informations	32
4.2.	Méthodes et outils de TALN en biologie	33
4.2.1.	Identification des termes	33
4.2.1.1.	Méthodes basées sur les dictionnaires	33
4.2.1.2.	Méthodes basées sur les règles	33
4.2.1.3.	Méthodes basées sur les techniques d'apprentissage	34
4.2.2.	Identification des interactions	34
4.2.3.	Quelques outils	35
4.2.3.1.	Medminer	35
4.2.3.2.	PubMiner	35
4.2.3.3.	Textpresso	36
5.	Conclusion	36
Chapitre 2 - Proposition de MEAT		37
1.	Introduction	39
2.	Contexte général	39
2.1.	Les expériences des puces à ADN	Erreur ! Signet non défini.
2.2.	Le processus de validation et d'interprétation d'une expérience	40
2.2.1.	Traitement des résultats bruts	42
2.2.2.	Analyse et exploitation des résultats	42
3.	Capitalisation des connaissances pour l'analyse du transcriptome	43
3.1.	Que mémoriser?	43
3.2.	Inventaire de l'existant	44
3.2.1.	MEDIANTE	44
3.2.2.	Les bases documentaires	45
3.2.3.	Les analyses des experts	45
3.3.	Comment mémoriser?	46
4.	Démarche générale	47
4.1.	Modélisation du domaine : définition des ontologies	48
4.2.	Extraction des connaissances à formaliser	48
4.3.	Recherche d'informations guidée par les ontologies	48

5.	Conclusion	50
Chapitre 3 - Choix et construction des ontologies : MeatOnto		51
1.	Introduction	53
2.	Description de l'ontologie MeatOnto	53
3.	Une ontologie pour Les expériences des puces à ADN : MGED	54
3.1.	Présentation générale	54
3.2.	Correspondance entre MGED et MEDIANTE	55
4.	Une ontologie pour le domaine biomédical : UMLS	56
4.1.	Présentation générale	56
4.1.1.	Le métathésaurus	56
4.1.2.	Le réseau sémantique	57
4.2.	UMLS : Une ontologie?	58
4.2.1.	Etude ontologique du métathésaurus	58
4.2.2.	Etude ontologique du réseau sémantique	60
4.3.	Enrichissement de UMLS	62
4.4.	Formalisation du réseau sémantique	64
4.4.1.	Formalisation de la hiérarchie des concepts	64
4.4.2.	Formalisation de la hiérarchie des relations	65
5.	L'ontologie DocOnto	69
5.1.	Des métadonnées sur les annotations	69
5.2.	Construction informelle	70
5.3.	Formalisation de DocOnto	71
6.	Une ontologie pour une mémoire d'expériences	72
7.	Conclusion	74
Chapitre 4 - Génération automatique des annotations : MeatAnnot		77
1.	Présentation générale	79
1.1.	Introduction	79
1.2.	Vue d'ensemble	79
1.2.1.	Motivation	79
1.2.2.	Notre approche	80
1.3.	Outils de TALN utilisés	82
1.3.1.	GATE	82
1.3.2.	TreeTagger	83
1.3.3.	RASP	83
1.4.	Identification des termes de UMLS	84
2.	Démarche de l'extraction des connaissances à partir des textes	85
2.1.	Analyse morpho-syntaxique des textes	85
2.2.	Détection des relations sémantiques	85
2.2.1.	Repérage des relations sémantiques dans les textes	85
2.2.2.	Les grammaires de détection de relations	86

2.2.2.1.	JAPE : un langage d'expression de grammaires pour le TALN	87
2.2.2.2.	Définition des grammaires de détection pour UMLS	87
2.3.	Extraction des candidats-termes	89
2.3.1.	Le processus d'extraction des termes	89
2.3.2.	Expansion de la liste des candidats-termes	90
2.4.	Génération de l'annotation	92
2.4.1.	Processus général	92
2.4.2.	Exemple d'exécution	93
3.	Validation et évaluation de la méthodologie	95
3.1.	Le processus de validation	96
3.2.	Résultats de l'évaluation	98
4.	Discussion et conclusion	100
Chapitre 5 - Exploitation des annotations : MeatSearch		101
1.	Introduction	103
2.	Vue d'ensemble	103
2.1.	Motivations	103
2.2.	Corese (Conceptual Resource Search Engine)	105
3.	La recherche d'informations dans Meat	108
3.1.	Exemples d'interfaces	108
3.1.1.	Exemple A : Interface de recherche libre	108
3.1.2.	Exemple B : Recherche d'expériences	110
3.1.3.	Exemple C : Le panier de gènes	112
3.2.	Exploitation des règles	114
3.3.	Exploitation des métadonnées	115
4.	Conclusion	116
Conclusions et perspectives		119
1.	Conclusion : contributions scientifiques	121
2.	Limites et perspectives	123
Bibliographie		125
Annexe I : Les grammaires de détection des relations		133
1.	Description des relations sémantiques dans UMLS	134
2.	Exemples de grammaires de détection des relations	138

Liste des tableaux

Tableau 1 - Quelques concepts de l'ontologie MGED	55
Tableau 2 - Correspondance entre les concepts de MGED et les tables de MEDIANTE	56
Tableau 3 - Exemple d'un concept du métathésaurus de UMLS	57
Tableau 4 - Exemples de concepts de l'ontologie DocOnto	72
Tableau 5 - Exemples de relations de l'ontologie DocOnto	72
Tableau 6 - Résultat de TreeTagger sur l'exemple précédent	83
Tableau 7 - Les résultats de l'extracteur de termes sur le corpus GENIA	98
Tableau 8 - Résultats de l'évaluation des suggestions	99
Tableau 9- Correspondance entre RDFS/RDF et GC	107

Liste des figures

Figure 1 - Exemple d'une annotation RDF	9
Figure 2 - Le problème de réutilisation/utilisation selon [Gómez-Pérez et al., 2003]	14
Figure 3 - Cycle de vie de la mémoire d'entreprise [Dieng-Kuntz et al., 2005]	21
Figure 4 - Architecture d'un web sémantique d'entreprise [Dieng-Kuntz, 2004b]	22
Figure 5 - Le principe d'une expérience puce à ADN	40
Figure 6 - Cycle de validation et d'interprétation d'une expérience puce à ADN	41
Figure 7 - Première vue sur la structure de la mémoire d'expérience	46
Figure 8 - Démarche générale de la construction de MEAT	47
Figure 9 - Architecture de Meat	49
Figure 10 - Représentation du type sémantique « Human » dans le RS de UMLS	57
Figure 11 - Une portion du réseau sémantique de UMLS	58
Figure 12 - Cycle direct dans le métathésaurus de UMLS [Bodenreider, 2001]	59
Figure 13 - Cycle indirect dans le métathésaurus UMLS [Bodenreider, 2001]	60
Figure 14 - Exemple d'une relation dans le réseau sémantique	61
Figure 15 - Définition de la relation 'affects'	63
Figure 16 - Spécialisation de la relation 'affects'	63
Figure 17 - Exemple de classes RFDS de UMLS	65
Figure 18 - Définition de la propriété 'process_of' en RDFS	66
Figure 19 - Représentation de la relation « process_of » en OWL	67
Figure 20 - Représentation de la relation « contains » en OWL	68
Figure 21 - Représentation d'une annotation d'un document	71
Figure 22 - Les concepts et les relations de l'ontologie DocOnto	71
Figure 23 - Structure de l'ontologie pour la mémoire d'expériences	74
Figure 24 - Les étapes de la génération des annotations basées à partir du texte	80
Figure 25 - Vue d'ensemble de la méthodologie	81
Figure 26 - Résultat de la requête « development of lung » renvoyé par l'UMLS SKS	84
Figure 27 - Exemple de repérage d'une relation sémantique dans le texte	86
Figure 28 - La grammaire de détection de la relation 'has_a_role_in'	88
Figure 29 - Résultats de la phase d'extraction de termes	90
Figure 30 - Le schéma de l'extracteur de termes	92
Figure 31 - Exemple1 : génération d'une annotation	94
Figure 32 - Exemple2 : génération d'une annotation	95
Figure 33 - L'interface de validation des suggestions	99
Figure 34 - Vue d'ensemble de MeatSearch	105
Figure 35 - Exemple d'une requête et un résultat SPARQL	106
Figure 36 - Architecture de CORESE	107
Figure 37 - Exemple d'interface de recherche libre	109
Figure 38 - Résultat d'une recherche sur un gène	110

Figure 39 - Lien entre le résultat d'une requête et la ressource annotée	110
Figure 40 - Interface de recherche d'expériences	111
Figure 41 - Résultat d'une recherche d'expériences	112
Figure 42 - Exemple de l'interface du panier de gènes	113
Figure 43 - Résultat d'une classification de gènes	114
Figure 44 - Exemple de métadonnées intégrées dans une annotation d'un document	116
Figure 45 - Calcul de chemin entre deux entités	117

Remerciements

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

Je remercie tout d'abord Mr. Peter SANDER, Professeur à l'université de Nice Sophia Antipolis, qui m'a fait l'honneur de présider le jury de cette thèse.

Je remercie Mr. Gilles KASSEL, Professeur à l'université Jules Verne d'Amiens ainsi que Mr. Amedeo NAPOLI, Directeur de recherches CNRS au Loria pour avoir accepté de rapporter ce manuscrit, ainsi que pour l'intérêt qu'ils ont manifesté à l'égard de ce travail de thèse.

Je remercie Mme. Nathalie AUSSENAC-GILLES et Mr. Pascal BARBRY d'avoir accepté d'examiner mes travaux.

Je remercie Mme. Rose DIENG-KUNTZ, Directeur de recherches INRIA, pour avoir encadré mon travail, et pour son aide précieuse, sa patience, et son support inestimable durant ce travail.

Je remercie tous les membres d'ACACIA pour leur accueil dans l'équipe, et pour leurs questions pertinentes lors des réunions, qui ont fait avancer ce travail.

Je remercie les membres de l'équipe de Mr. Pascal BARBRY, spécialement Mr. Kevin LE BRIGAND, qui ont mis à ma disposition leur expertise du domaine des biopuces.

Je remercie tous les membres de l'INRIA Sophia Antipolis pour leur accueil.

Je remercie la région PACA (Provence Alpes Côte d'Azur) pour avoir financé en partie ces travaux.

Je remercie surtout ma famille : mon père Ahmed, ma mère Souad, ma grand-mère Saidouda, mes beaux-parents, Aicha, Hend, Rym, Imed, Karim, Hichem, Karim KEFI, Nizar, Hinda et très spécialement ma femme Leila, pour leur soutien constant à travers ces longues années.

Je remercie mes amis : Nicolas, Cécile, Emilie, Mohammad, Mehdi, Khaled, Mohamed, Moez, Mouna, Sami...

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

Introduction

Contexte industriel et scientifique

Il fut un temps où le potentiel cognitif d'une équipe de recherche se limitait aux connaissances contenues dans ses publications internes ou dans des revues scientifiques spécialisées, ainsi qu'aux connaissances provenant de l'interprétation de ses propres résultats expérimentaux. Grâce à l'apparition de nouvelles technologies informatiques (Internet, systèmes de gestion de documents électroniques, base de données...), les chercheurs ont désormais la possibilité de partager leurs connaissances et d'accéder ainsi à de nouvelles connaissances scientifiques à travers les documents publiés sur le Web et les informations stockées dans les bases en ligne.

Ces connaissances qui sont indispensables à la vérification, la validation et/ou l'enrichissement du travail des chercheurs d'un domaine particulier, sont difficilement exploitables, en raison de la grande quantité des données provenant des sources autant internes qu'externes aux organisations.

Un des domaines concernés par ce problème de détection, stockage et exploitation d'énormes masses de données, est le domaine de la biologie moléculaire et en particulier, le domaine des expériences des puces à ADN (biopuces). En effet, dans ce domaine, les biologistes manipulent de grandes quantités de données dans différentes conditions expérimentales et doivent se référer à des milliers de publications scientifiques liées à leurs expériences. Afin de s'y retrouver dans cette masse énorme de données, nos collègues de l'IPMC (Institut de Pharmacologie Moléculaire et Cellulaire¹) ont manifesté leur intérêt pour un support méthodologique et logiciel qui les aiderait pour la validation et l'interprétation de leurs résultats et qui leur faciliterait la planification de nouvelles expérimentations.

C'est dans ce contexte que s'est formé un partenariat entre la plate-forme biopuces de Sophia Antipolis basée à l'IPMC et l'équipe de recherche ACACIA de l'INRIA Sophia Antipolis. Ce travail de thèse a été mené dans le cadre de ce partenariat.

Problématique et objectifs poursuivis

L'hétérogénéité des sources d'informations, la difficulté d'accéder à la connaissance et la perte du savoir-faire ou du cheminement d'une interprétation rentrent dans le cadre de la problématique de la gestion des connaissances au sein d'une communauté. Des chercheurs en intelligence artificielle se sont intéressés à cette problématique et ont proposé une solution qui consiste à doter la communauté ou l'organisation d'une mémoire (par analogie avec la mémoire

¹ <http://www.ipmc.cnrs.fr/>

humaine) permettant la capitalisation, le partage et la diffusion des connaissances. De façon générale, adopter une telle solution revient à répondre aux questions suivantes :

Comment détecter l'information ?

Il s'agit de définir les différentes sources d'informations pouvant constituer les ressources de la mémoire. Ces sources sont généralement hétérogènes (documents, bases de données, interprétations humaines ...).

Comment modéliser et formaliser les connaissances ?

Il s'agit de choisir un modèle (par exemple les ontologies) pour représenter formellement les connaissances afin de faciliter leur partage et l'interopérabilité entre les différents acteurs de la mémoire (les humains et les machines).

Comment alimenter la mémoire ?

Il s'agit d'offrir des moyens pour l'extraction des connaissances à partir des sources déjà répertoriées et ce afin d'alimenter la base de connaissances.

Comment diffuser les connaissances décrites dans la mémoire ?

Il s'agit de définir des scénarios d'interaction avec les utilisateurs pour faciliter la navigation dans la mémoire et l'exploitation des connaissances qu'elle contient.

C'est dans ce contexte que nous nous positionnons, en proposant la construction et la gestion d'une mémoire d'expériences pour une communauté de biologistes réalisant des expériences de puces à ADN. Le but principal de cette mémoire est d'apporter des aides méthodologiques et logicielles pour la capitalisation et la valorisation des connaissances au sein de la communauté étudiée.

En réponse aux besoins exprimés par nos collègues biologistes et à la problématique générique de construction d'une mémoire de communauté, nous nous sommes fixés les objectifs suivants :

- Proposer une architecture générique pour la conception d'une mémoire d'expériences ;
- Intégrer les techniques actuelles du Web Sémantique dans chacune des étapes de la construction de la mémoire.

Contributions et champs de recherche concernés

En tenant compte des objectifs indiqués dans la section précédente, nous proposons le système MEAT (Mémoire d'Expériences pour l'Analyse du Transcriptome) qui implémente une méthodologie de construction d'une mémoire d'expériences. Cette mémoire possède les caractéristiques suivantes :

- Elle englobe les différentes **sources d'informations** du domaine considéré, à savoir les articles scientifiques, les bases d'expériences et les documents décrivant les interprétations humaines.
- Elle repose sur les ontologies pour **la description et la formalisation des connaissances** du domaine.
- Elle intègre des techniques d'extraction de connaissances lui permettant de **s'alimenter au fur et à mesure de son cycle de vie**. Nous proposons une méthodologie d'extraction de connaissances à partir des textes et de génération d'annotations sémantiques basées sur l'ontologie.
- Elle fournit **un accès aux connaissances capitalisées** en offrant des mécanismes assez poussés de recherche d'informations (exploitation de la structure hiérarchique de l'ontologie et inférences sur les annotations sémantiques).

Comme nous pouvons le remarquer, ce système est proposé pour répondre aux besoins des biologistes travaillant sur des expériences sur les puces à ADN. Cependant, la méthodologie de capitalisation et de gestion de connaissances proposée peut être généralisée à d'autres domaines des sciences de la vie requérant des expérimentations et traitant un grand flux de données (protéomique, chimie, etc.).

Notre contribution se situe au carrefour de trois disciplines, à savoir :

- L'**ingénierie des connaissances** : le but principal de ce travail est de rendre les connaissances nécessaires pour la validation et l'interprétation d'une expériences sur les puces à ADN, visibles et accessibles pour tous les acteurs travaillant sur ce type d'expériences.
- Les technologies du **Web Sémantique** : nous avons choisi de matérialiser notre mémoire d'expériences en utilisant les techniques du Web Sémantique telles que les ontologies et les annotations sémantiques (approche proposée par [Dieng-Kuntz et al., 2004b]).
- Le **traitement automatique de la langue naturelle** : nous considérons que le texte est une source essentielle et riche pour l'acquisition des connaissances. Le volume des textes à traiter (i.e. nombre d'articles scientifiques essentiellement intéressants) justifie le recours à l'automatisation de ce traitement.

Organisation du document

Dans le premier chapitre, nous présentons la nouvelle vision du Web, à savoir le Web Sémantique, en nous focalisant sur ses principales composantes (représentation des connaissances, ontologies avec des exemples pour le domaine biomédical et les annotations sémantiques). Ensuite, nous nous penchons sur la problématique de gestion de connaissances, et tout particulièrement sur la construction d'une mémoire d'entreprise en utilisant les techniques du Web Sémantique. Enfin, nous nous intéressons à la façon d'alimenter cette mémoire, et ce en présentant les méthodes et les outils décrits dans la littérature et permettant l'extraction des connaissances à partir des textes (avec un zoom sur les textes biomédicaux).

Dans le deuxième chapitre, nous décrivons la problématique de validation et d'interprétation des résultats rencontrée par les biologistes travaillant sur les puces à ADN. Puis nous présentons la méthodologie adoptée pour résoudre cette problématique en décrivant le principe du système MEAT et en dévoilant son architecture.

Dans le troisième chapitre, nous commençons à détailler les différents éléments de MEAT en décrivant l'ontologie sur laquelle se base notre système. Nous nous intéressons aux besoins et aux contraintes qui nous ont poussés à faire nos choix et nous détaillons les différentes composantes de cette ontologie en précisant leurs rôles.

Dans le quatrième chapitre, nous poursuivons avec la description de la méthodologie adoptée et celle de l'outil développé pour l'extraction des connaissances à partir des textes et la génération des annotations sémantiques basées sur l'ontologie. Nous présentons une validation de l'approche (centrée utilisateur) qui nous a permis d'évaluer cet outil et d'améliorer les techniques d'extraction utilisées par l'outil.

Dans le cinquième chapitre, nous abordons la thématique de recherche d'information guidée par les ontologies en présentant un outil (basé sur un moteur de recherche sémantique existant) permettant d'exploiter les connaissances contenues dans les annotations sémantiques générées. Nous donnons des exemples de recherches qui illustrent la potentialité d'une telle approche.

Chapitre 1 - État de l'art

Notre travail touche à plusieurs disciplines : les techniques du Web Sémantique, l'ingénierie et la gestion des connaissances, la linguistique (en particulier dans le domaine biomédical) et la recherche d'informations, ce qui constitue la richesse du sujet.

Ce chapitre présente ces différentes problématiques :

La première partie s'intéresse à la nouvelle vision du Web à savoir le Web Sémantique. Nous présentons les principales composantes du web sémantique, les rôles des ontologies dans la recherche d'informations et quelques exemples d'ontologies dans le domaine biomédical (notre domaine d'application).

Ensuite, nous abordons la problématique de gestion des connaissances en nous focalisant sur l'aspect mémoire d'entreprise. Nous détaillons une méthode de construction d'une mémoire d'entreprise qui consiste à utiliser les techniques du web sémantique déjà présentées.

Enfin, considérant que les textes constituent une source très riche pour la construction et l'alimentation d'une telle mémoire (construction des ontologies et des annotations sémantiques), nous nous penchons sur la problématique de l'extraction des connaissances à partir des textes avec un zoom sur les textes biologiques.

1. Le web à la recherche d'une sémantique

1.1. Introduction et définition

Depuis sa création, le web a connu un succès gigantesque et est en train de devenir peu à peu le premier outil pour la production, la publication, la diffusion et le partage de l'information. Cependant la répartition à travers le monde d'un tel réseau d'informations, la croissance accrue du nombre de publications et la liberté totale d'y accéder ont révélé plusieurs limites et inconvénients. En effet, le web actuel ne dispose pas d'outils pour décrire et structurer ses ressources de manière satisfaisante afin de permettre un accès pertinent à l'information. Par exemple, les liens entre les pages web, bien que porteurs de sens pour les utilisateurs, n'ont aucune signification exploitable par les machines.

C'est pour pallier ces insuffisances que Tim Berners Lee a proposé dans [Berners-Lee et al., 2001] d'étendre le web actuel vers un web où l'information posséderait un sens bien défini permettant ainsi aux applications d'exploiter directement la sémantique des ressources et de coopérer avec l'utilisateur afin de lui faciliter ses tâches (recherche, commerce électronique...).

Ce futur web baptisé web sémantique a été défini comme un « web intelligent » où les informations stockées dans les machines seraient en plus comprises par ces dernières afin de répondre efficacement aux requêtes lancées par les utilisateurs.

1.2. Les principales composantes du web sémantique

Le Web Sémantique a été proposé en se basant sur les critiques adressées au web actuel : (i) certes HTML a permis de tisser tout un réseau d'informations par ses liens hypertextes, mais il n'a donné aucune sémantique à ces liens ce qui les rend pratiquement inexploitable par les

machines, (ii) les métadonnées utilisées sont non structurées et limitées dans leurs usages, et (iii) il est difficile de faire des inférences et des raisonnements sur les connaissances décrites dans les documents publiés sur le web vu l'absence de modèles permettant la représentation sémantique de ces connaissances.

Tous ces problèmes ont fait l'objet de différents travaux de recherche qui ont convergé vers plusieurs solutions parmi lesquelles nous présentons celles qui nous semblent les plus essentielles :

- Proposer des langages et des formalismes de représentation et de structuration des connaissances. Ces langages permettent de modéliser et de représenter le contenu sémantique des ressources du web (§1.2.1).
- Rendre disponibles des ressources conceptuelles (des modèles) représentées dans ces langages modélisant les connaissances et facilitant leur accès et leur partage : les ontologies (§1.2.2).
- Proposer des métadonnées explicites, c'est-à-dire qui suivent un modèle et qui sont exprimées dans des langages définis formellement (§1.2.3).

1.2.1. La représentation des connaissances

Le fonctionnement du Web Sémantique est fondé sur le fait que les machines puissent accéder à l'ensemble des informations éparpillées sur le web. Le W3C² (World Wide Web Consortium) ainsi que les chercheurs travaillant dans le domaine de l'intelligence artificielle ont beaucoup travaillé sur ce point et ont proposé plusieurs langages de représentation des connaissances afin de faciliter cet accès.

1.2.1.1. XML (eXtensible Markup Language)

C'est un méta-langage³ proposé par le W3C permettant de représenter un document textuel de manière arborescente en utilisant un système de balisage.

Ce langage a été élaboré pour faciliter l'échange, le partage et la publication des données à travers le web. Ainsi, la majorité des langages/modèles proposés pour le web sémantique sont exprimés en XML.

XML permet de structurer un document en définissant ses propres balises en fonction des besoins et sans tenir compte ni de la signification de cette structure ni des systèmes informatiques qui vont l'exploiter. Des standards comme XPath⁴ et XQuery⁵ ont été développés afin de parcourir et d'interroger l'arborescence XML des documents.

Etant donné que ce langage est un langage de structuration et non de représentation de données, le W3C a proposé le langage XSL (eXtensible StyleSheet Language) [Clark, 1999] pour effectuer la représentation des données des documents XML. XSL est lui-même défini

² <http://www.w3.org/>

³ <http://www.w3.org/XML/>

⁴ <http://www.w3.org/TR/xpath>

⁵ <http://www.w3.org/TR/2005/WD-xquery-20050915/>

avec le formalisme XML et il permet de définir des feuilles de style pour les documents XML. Outre la représentation des données, XSL permet aussi de retraiter un document XML afin d'en modifier totalement sa structure, ce qui permet à partir d'un document XML de générer d'autres types de documents (PostScript, HTML, Tex, RTF, ...) ou bien un fichier XML de structure différente.

1.2.1.2. RDF (*Ressource Description Framework*)

RDF [Lassila et Swick, 2001] est une recommandation du W3C développée pour décrire les ressources du web. Pour ce faire, RDF procède par une description de savoirs (données ou métadonnées) à l'aide d'expressions de structure fixée. En effet, la structure fondamentale de toute expression en RDF est une collection de triplets, chacun composé d'un sujet, un prédicat et un objet (ou {ressource, propriété, valeur}). Un ensemble de tels triplets est appelé un graphe RDF. Ceci peut être illustré par un diagramme composé de noeuds et d'arcs orientés, dans lequel chaque triplet est représenté par un lien noeud-arc-noeud (d'où le terme de "graphe").

A ce modèle est associée une syntaxe écrite en XML et basée sur les triplets :

- Ressource (Sujet) : une entité d'informations pouvant être référencée par un identificateur. Cet identificateur doit être une URI⁶.
- Propriété (prédicat) : l'attribut ou la relation utilisé(e) pour décrire une ressource.
- Valeur (objet) : la valeur d'une propriété associée à une ressource spécifique.

En utilisant ce modèle, il est possible de modéliser le fait que 'le livre identifié par l'ISBN 2-10-006300-6 est écrit par l'équipe ACACIA' comme suit :

```
<rdf:Description rdf:about='ISBN-2-10-006300-6'>  
  <auteur>équipe ACACIA</auteur>  
</rdf:Description>
```

Cette assertion peut être aussi représentée par un graphe étiqueté orienté (Figure 1).



Figure 1 - Exemple d'une annotation RDF

RDF est considéré comme le premier pilier du Web Sémantique et commence à être utilisé dans plusieurs applications, notamment les applications permettant d'annoter les documents sur le web/intranet afin de faciliter leur accès et leur partage.

⁶ Uniform Resource Identifiers : <http://www.w3.org/Addressing/>

1.2.1.3. RDFS (*Ressource Description Framework Schema*)

RDFS [McBride, 2004] est un méta modèle recommandé par le W3C permettant la définition de schéma/modèle décrivant l'univers sémantique des déclarations RDF.

RDFS fournit ainsi un système de typage pour les déclarations RDF. Il permet la définition des classes et des sous-classes (`rdfs:Class`, `rdfs:subClassOf`) décrivant les ressources à annoter et donnant un sens aux propriétés associées aux ressources. Il permet aussi la formulation de contraintes sur les valeurs associées à une propriété afin de lui assurer une signification (`rdfs:domain`, `rdfs:range`).

Dans le contexte du web sémantique RDFS est utilisé pour formaliser les ontologies sur lesquelles vont se baser les annotations RDF.

1.2.1.4. DAML+OIL

DAML+OIL [Van Harmelen et al., 2001] est la fusion de deux langages de représentation des connaissances DAML⁷ et OIL [Fensel et al., 2000] basés essentiellement sur les logiques de descriptions [Napoli, 1997] et qui ont été proposés comme langage de description d'ontologies.

Le but de DAML+OIL est d'étendre RDFS en lui ajoutant des primitives plus expressives pour la définition des classes et des propriétés d'une ontologie. Parmi ces extensions nous pouvons citer :

- l'intersection (`daml:intersectionOf`), l'union (`daml:unionOf`) et la négation (`daml:complementOf`);
- la collection d'individus (`daml:oneOf`);
- la restriction sur l'application des propriétés (`daml:Restriction`);
- l'équivalence des ressources (classes et propriétés) (`daml:equivalentTo`, `daml:sameClassAs` et `daml:samePropertyAs`);
- Etc.

Ce langage a fait l'objet d'une note du W3C.

1.2.1.5. OWL (*Web Ontology Language*)

Le développement rapide des applications basées sur les ontologies et la nécessité de modéliser des connaissances de plus en plus complexes, ont fait émerger quelques limites de RDF/RDFS. En effet, RDFS offre un vocabulaire simple, limité à une hiérarchie de classes, une hiérarchie de relations et des définitions des domaines d'application de ces dernières (« domain » et « range »).

OWL [McGuinness et Van Harmelen, 2004] a été recommandé par le W3C (en particulier par le groupe WebOnt déjà à l'origine de DAML+OIL) afin d'enrichir RDFS en définissant un vocabulaire plus complet pour la description d'ontologies complexes. Cette richesse par rapport

⁷ <http://www.daml.org/>

à RDFS se matérialise par l'ajout de nouvelles notions telles que : l'équivalence des classes, l'équivalence des relations, la symétrie et la transitivité des relations, la cardinalité, etc.

Ce nouveau langage est divisé en trois sous-langages définis par une syntaxe expressive avec une sémantique formelle et rigoureuse :

- OWL Lite : c'est la version légère de OWL qui reprend RDFS et l'enrichit avec de nouvelles primitives.
- OWL DL : contient toutes les primitives de OWL (y compris OWL Lite) avec des contraintes particulières sur leur utilisation qui assurent la décidabilité du langage.
- OWL Full : plus flexible que OWL DL ce qui le rend vraisemblablement indécidable.

OWL est basé essentiellement sur le formalisme des logiques de descriptions et tire profit des inférences et des mécanismes de raisonnements associés à ces formalismes. Pour ses trois sous langages, seuls les deux premiers maintiennent les tâches d'inférence principales à savoir la satisfiabilité et la classification.

Notons que, de plus en plus, de grandes ontologies sont en train d'être publiées en OWL (comme par exemple, dans le domaine biomédical : la Gene Ontology et Galen détaillées dans le §1.4). Ce gain de notoriété, provient en partie de l'extension de l'outil le plus utilisé pour l'édition d'ontologies Protégé [Noy et al., 2001] par un greffon (plugin) dédié à l'édition d'ontologies OWL [Knublauch et al., 2004].

Une autre application prometteuse de OWL est OWL-S [Martin et al., 2004], qui est un formalisme présenté comme une ontologie OWL pour la description des services Web sémantiques.

Dans cette partie, nous avons présenté un sous ensemble des langages disponibles (les plus utilisés) pour la représentation des connaissances dans le cadre du Web Sémantique. Ces langages, offrent un degré important d'expressivité pour représenter les connaissances simples et complexes (sous forme d'ontologie ou de méta-données) décrites dans les ressources du Web.

1.2.2. Les ontologies

1.2.2.1. A la recherche d'une définition

Terme d'origine philosophique, ontologie désigne la théorie d'étude de la « nature de l'existant ». Cette notion a été reprise par les chercheurs dans le domaine de l'intelligence artificielle et utilisée dans le cadre de construction des systèmes à base de connaissances. L'idée était de séparer, d'un côté, la modélisation des connaissances d'un domaine, et d'un autre côté, l'utilisation de ces connaissances (i.e. le raisonnement).

Dans ce contexte, plusieurs définitions des ontologies ont été proposées.

La première a été proposée par [Neches et al., 1991]: « Une ontologie définit les termes et les relations de base du vocabulaire d'un domaine ainsi que les règles qui permettent de combiner les termes et les relations afin de pouvoir étendre le vocabulaire »⁸.

Cette définition descriptive donne un premier aperçu sur la manière de construire une ontologie, à savoir l'identification des termes et des relations d'un domaine ainsi que les règles pouvant s'appliquer sur ces derniers.

Deux années plus tard, [Gruber, 1993] donne la définition qui est devenue la plus utilisée dans la littérature :

«Une ontologie est une spécification explicite d'une conceptualisation»⁹.

La conceptualisation se réfère ici à l'élaboration d'un modèle abstrait d'un domaine du monde réel en identifiant et en classant les concepts pertinents décrivant ce domaine. La formalisation consiste à rendre cette conceptualisation exploitable par des machines.

Dans cette même logique [Guarino et Giaretti, 1995] proposent leur définition : « *Une ontologie est une théorie logique proposant une vue explicite et partielle d'une conceptualisation* »¹⁰.

Depuis, de nombreuses définitions, à la fois complémentaires et précises, ont vu le jour.

[Aussenac-Gilles et al., 2000] soulignent la dépendance entre la formalisation de l'ontologie et l'application dans laquelle elle va être utilisée : « *Une ontologie organise dans un réseau des concepts représentant un domaine. Son contenu et son degré de formalisation sont choisis en fonction d'une application* ».

Nous retiendrons donc qu'une ontologie traduit un consensus explicite sur la formalisation des connaissances d'un domaine afin de faciliter le partage et la réutilisation de ces connaissances par les membres d'une communauté ou par des agents logiciels.

C'est dans cette optique que les ontologies se présentent comme un pilier du web sémantique, car elles permettent de faire communiquer les hommes et les machines en utilisant la sémantique partagée par les différents acteurs du web et en décrivant ses ressources.

1.2.2.2. Les différents types d'ontologies

[Van Heijst et al., 1997] définissent deux grandes typologies d'ontologies : (i) une typologie fondée sur la structure de la conceptualisation et (ii) une typologie fondée sur le sujet de la conceptualisation.

Dans la première typologie, ils distinguent trois catégories à savoir

- les ontologies terminologiques (lexiques, glossaires...);
- les ontologies d'information (schéma d'une BD);

⁸ Traduction de "An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary"

⁹ Traduction de "An ontology is an explicit specification of a conceptualization"

¹⁰ Traduction de "A logical theory which gives an explicit, partial account of a conceptualization"

- les ontologies des modèles de connaissances.

Dans la deuxième typologie, qui est la plus citée, ils distinguent quatre catégories :

- les ontologies d'application : elles contiennent toutes les informations nécessaires pour modéliser les connaissances pour une application particulière.
- les ontologies de domaine : elles fournissent un ensemble de concepts et de relations décrivant les connaissances d'un domaine spécifique.
- les ontologies génériques (dites aussi de haut niveau) : elles sont similaires aux ontologies de domaine, mais les concepts qui y sont définis sont plus génériques et décrivent des connaissances tels que l'état, l'action, l'espace et les composants. Généralement, les concepts d'une ontologie de domaine sont des spécialisations des concepts d'une ontologie de haut niveau.
- Les ontologies de représentation (dites aussi méta-ontologies) : elles fournissent des primitives de formalisation pour la représentation des connaissances. Elles sont généralement utilisées pour écrire les ontologies de domaine et les ontologies de haut niveau. Exemples : Frame Ontology [Gruber, 1993] et RDF Schema Ontology [McBride, 2004].

Dans cette même typologie on peut retrouver d'autres catégories telles que les ontologies de tâches et de méthodes définies dans [Mizoguchi et Vanwelkenhuysen, 1995].

1.2.2.3. La réutilisation des ontologies

La réutilisation était considérée comme l'un des principaux apports de l'intégration des ontologies dans les systèmes de gestion des connaissances. Avec le développement de plusieurs applications basées sur des ontologies, cette question est de plus en plus mise en cause et alimente de longs débats.

En effet, [Charlet et al., 1996] et [Aussenac-Gilles et al., 2000] (et plus généralement les membres du groupe TIA¹¹) considèrent qu'une ontologie est difficilement réutilisable. D'après eux, une ontologie garde toujours la trace de la tâche pour laquelle elle a été développée. En d'autres termes, la construction d'une ontologie est toujours guidée par son application, ce qui rend le souhait d'avoir une ontologie universelle totalement utopique.

D'un autre côté, [Uschold et Grüninger, 1996] proposent une solution originale à ce problème de réutilisation qui consiste à construire des bibliothèques d'ontologies, où ces dernières pourront ensuite être combinées pour pouvoir générer des nouvelles ontologies. Les auteurs proposent des recommandations pour faire ces combinaisons.

La réutilisation d'une ontologie telle qu'elle dans deux tâches ou deux applications différentes semble donc difficile. Ainsi, comme le souligne [Gómez-Pérez et al., 2003], il existe une dépendance forte entre l'utilisation et la réutilisation des ontologies : « plus elle est réutilisable, moins elle est utilisable ».

¹¹ <http://www.biomath.jussieu.fr/TIA/>

La Figure 2 représente le degré de réutilisation d'une ontologie par rapport à son degré d'utilisation.

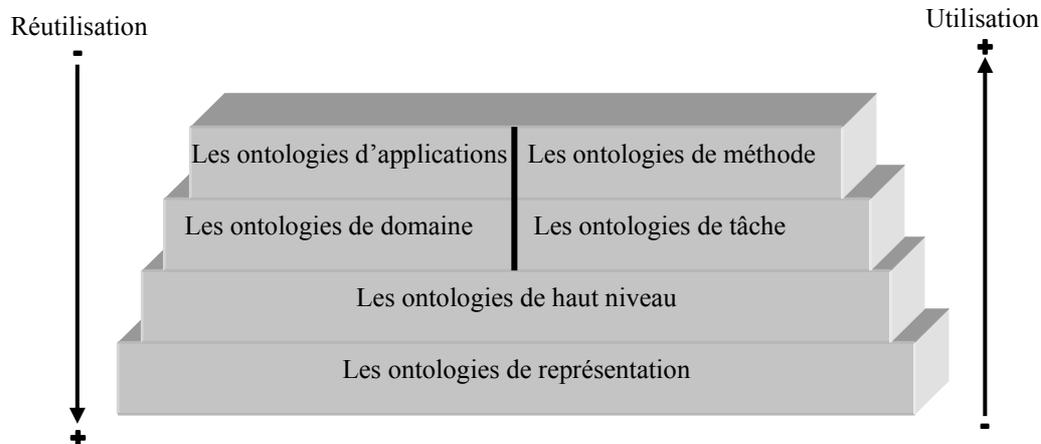


Figure 2 - Le problème de réutilisation/utilisation selon [Gómez-Pérez et al., 2003]

Ce constat a motivé plusieurs travaux sur la réutilisation des ontologies de haut niveau qui répertorient des concepts assez génériques pour décrire la société humaine. Parmi ces travaux, nous pouvons citer le groupe de travail SUO¹² qui propose l'ontologie SUMO [Niles et Pease, 2001] qui se veut être une ontologie universelle réutilisable dans n'importe quelle application. Cette réutilisation est en effet réalisable, si nous arrivons à spécialiser les concepts génériques de cette ontologie en des concepts spécifiques du domaine étudié.

Dans notre travail, nous avons pu réutiliser deux ontologies de domaine, la première a été proposée comme standard pour décrire les puces à ADN, et la deuxième se considère comme une ontologie générique pour le domaine biomédical : nous les détaillerons dans le chapitre 3.

1.2.3. Les annotations sémantiques

Une annotation (ou métadonnée) est une information descriptive facilitant l'accès, la recherche et l'utilisation d'une ressource. Se baser sur un modèle de connaissances déjà défini (i.e. se baser sur une ontologie) enrichit l'annotation en lui attribuant une sémantique et en la rendant utilisable comme telle par un agent logiciel.

En terme de documentation, les annotations sémantiques décrivent le lien entre les entités se trouvant dans le document et leurs descriptions sémantiques représentées dans l'ontologie. Elles permettent ainsi de désambiguïser le contenu du document pour un traitement automatique (ex. recherche documentaire, résumé...).

Avec l'expansion du Web Sémantique, plusieurs outils ont été proposés pour permettre la génération automatique ou manuelle de ces annotations sémantiques. Nous présentons ci-dessous quelques exemples.

¹² <http://suo.ieee.org/>

1.2.3.1. *Annotea*

C'est un système de génération (manuelle) d'annotations RDF pour les pages web [Kahan et al., 2001] développé au sein du W3C. L'idée principale est de proposer à chaque utilisateur un outil lui permettant (i) d'annoter un document en le consultant, (ii) de consulter toutes les annotations associées à un document et (iii) de typer les annotations en leur associant des méta-données (par exemple dire que cette annotation est un commentaire ou un erratum pour le document) en se basant sur un modèle prédéfini en RDFS.

Annotea a été intégré dans le navigateur web Amaya¹³.

1.2.3.2. *MnM*

MnM fournit un environnement pour la génération semi-automatique d'annotations sémantiques associées aux documents Web [Vargas-Vera et al., 2002]. Il est basé essentiellement sur des techniques d'apprentissage et des méthodes de TALN (Traitement Automatique de la Langue Naturelle).

Dans un premier temps, un corpus d'apprentissage est fourni aux utilisateurs afin de l'annoter manuellement en se basant sur une ontologie du domaine. Ce corpus est ensuite passé à l'outil Amilcare [Ciravegna, 2003] qui génère un ensemble de règles d'extraction qui seront ensuite appliquées sur les autres documents pour générer automatiquement les annotations.

Deux types de règles sont générés à savoir :

- les règles d'étiquetage qui permettent de repérer la partie du texte à annoter et d'y insérer une étiquette sémantique,
- les règles de correction qui permettent d'explorer le texte étiqueté et de détecter les étiquettes incorrectes et de les corriger tout en se basant sur les informations recueillies lors de la phase d'apprentissage.

1.2.3.3. *KIM*

KIM [Popov et al., 2004] fournit une plate-forme de génération automatique d'annotations sémantiques et de recherche documentaire basée sur ces annotations.

L'approche de KIM est basée sur l'extraction des entités nommées présentes dans le texte à annoter afin d'instancier les concepts d'une ontologie de haut niveau (KIMO) représentée en RDFS. Ces instances sont ensuite utilisées pour annoter les documents et pour enrichir la base de connaissances de KIM.

Ces annotations permettent ainsi de faire des recherches documentaires plus précises en utilisant les restrictions sémantiques offertes par l'ontologie.

Le processus d'extraction d'informations de KIM est basé sur l'outil GATE¹⁴.

¹³ <http://www.w3.org/Amaya/>

¹⁴ Détaillé dans le chapitre 4, §1.3.1

1.2.3.4. S-CREAM

C'est une plate-forme pour la création semi-automatique d'annotations sémantiques basées sur une ontologie [Handschuh et al., 2002]. Cette plate-forme fournit deux approches, toutes les deux implémentées dans l'outil OntoMat¹⁵ :

- La première est basée sur une phase d'apprentissage manuelle (similaire à celle de MnM) qui sert comme entrée à l'outil Amilcare qui génère ensuite des règles d'extraction d'instances des concepts de l'ontologie.
- La deuxième est basée sur une méthode originale nommée PANKOW [Cimiano et al., 2005] qui est entièrement automatique. Le système implémentant PANKOW extrait dans un premier temps des candidats termes à partir du texte à annoter (utilisation de techniques de TALN). Ensuite, en se basant sur des patrons de « génération d'hypothèses », il construit des requêtes en combinant chaque terme avec les concepts de l'ontologie. Enfin, en comparant les résultats des requêtes (Google), il déduit à quels concepts il doit associer le terme (Exemple de patron : <CONCEPT>s such as <INSTANCE>).

Beaucoup d'autres systèmes ont été proposés pour la génération (semi-)automatique d'annotations sémantiques [Uren et al., 2006]. Ils proposent, le plus souvent, des annotations en RDF (devenu un standard pour la représentation des méta-données). Par ailleurs, nous avons noté que la majorité de ces systèmes s'intéressent surtout à la tâche d'instanciation de concepts et proposent des techniques d'instanciation de formes simples de relations pouvant exister entre ces concepts (synonymie, paronymie...). Dans notre travail, nous considérons que les instances des relations (représentées dans les ontologies) présentes dans les textes jouent un rôle très important dans la description du contenu sémantique des documents et que leur représentation enrichisse les annotations.

1.3. La recherche d'information guidée par les ontologies

L'utilisation typique du Web actuel consiste en la recherche d'information qui peut être d'ordre professionnel (veille stratégique/technologique, recherche d'articles...) ou d'ordre personnel (recherche de personnes ou de produits).

Pour faciliter ces tâches, plusieurs moteurs de recherche ont vu le jour (Google, Yahoo, Altavista...). Ces outils, bien qu'ils répondent à une bonne partie des besoins des utilisateurs, présentent quelques problèmes critiques :

- la masse énorme des documents retournés,
- la sensibilité au vocabulaire utilisé dans la requête,

¹⁵ <http://annotation.semanticweb.org/ontomat/index.html>

- le résultat fractionné en pages Web, ce qui entraîne le besoin de faire plusieurs requêtes pour obtenir tous les documents pertinents et après en extraire manuellement la partie intéressante,
- la variabilité des langages utilisés sur le web et la non structuration des documents, ce qui rend cette tâche de plus en plus laborieuse.

La réflexion sur le web sémantique a été essentiellement fondée sur ce problème de la recherche d'informations. En effet, les ontologies peuvent améliorer la pertinence d'une recherche et ce, en recherchant des documents faisant référence à un concept précis au lieu de se baser sur des mot-clés qui peuvent être ambigus.

Prenons l'exemple d'une personne anglophone qui cherche à trouver l'adresse d'un installateur de fenêtres ; en tapant la requête « windows installation » dans n'importe quel moteur de recherche, elle obtiendra des milliers de pages traitant l'installation du système d'exploitation de Microsoft et les problèmes qui en résultent, mais elle aura beaucoup de mal à trouver l'information qu'elle recherchait.

Avec l'utilisation d'une ontologie, un moteur de recherche fera la différence entre un site sur lequel 'Windows' désigne un logiciel et un autre sur lequel il désigne une fenêtre.

Cette recherche basée sur les ontologies se présente comme une recherche intelligente qui repose sur la sémantique des ressources et sur les concepts contenus dans les documents qui leur sont associés. Ces ontologies peuvent ainsi, d'une part, guider la création d'annotations sous la forme de métadonnées sur les ressources, et d'autre part, décrire leurs contenus de manière à la fois formelle et signifiante pour être exploitable aussi bien par les humains que par les machines.

Dans cette optique, plusieurs systèmes de recherche d'informations à base d'ontologies ont été proposés, parmi lesquels nous pouvons citer : Ontobroker [Decker et al., 1999], Sesame [Broekstra et al., 2002] et CORESE [Corby et al., 2004]. La différence entre ces systèmes réside essentiellement dans le langage de représentation et le moteur d'inférence sur les connaissances imbriquées dans les annotations : Ontobroker utilise F-Logic [Kiffer et al., 1995], Sesame utilise SQL92SAIL (du SQL adapté à RDF), et CORESE utilise les graphes conceptuels [Sowa, 1984].

1.4. Des ontologies pour un web sémantique biomédical

Comme dans la plupart des domaines de recherches, les chercheurs dans le domaine biomédical visent à représenter, partager et réutiliser leurs connaissances. Par conséquent, plusieurs systèmes terminologiques ont été proposés et développés : des vocabulaires contrôlés pour annoter des gènes et classer les documents, et des thesaurus pour guider et faciliter la recherche d'informations. Néanmoins, le succès de ces systèmes est limité en raison de leur dépendance à des cas et des tâches spécifiques et de l'absence de possibilités de raisonnement.

Afin de compenser les limites de ces ressources, la communauté biomédicale s'est intéressée aux ontologies qui visent à représenter les connaissances indépendamment de leur cadre d'utilisation et qui offrent des mécanismes pour faire des inférences poussées sur ces connaissances. Dans cette partie, nous présentons quelques projets de construction d'ontologies.

Ces ontologies sont utilisées dans différentes tâches : annotations des gènes, annotations de documents, interopérabilité des systèmes biomédicaux et partage des connaissances.

1.4.1. GALEN

GALEN (General Architecture for Language, Encyclopedia and Nomenclature) est un projet européen (1992-1999) [Rector et al., 1996] qui avait pour but de proposer des terminologies réutilisables et partageables pour le domaine médical. Une ontologie (Common Reference Model : CRM) a été développée et représentée dans un langage de représentation propre à GALEN, appelé GRAIL [Rector et al., 1997]. Cette ontologie propose une représentation des concepts médicaux indépendamment de l'application choisie, dans le but de fournir une base pour la création de terminologies en combinant les concepts.

La version actuelle de GALEN comprend une hiérarchie assez riche de concepts (~ 25000 concepts) ainsi qu'un ensemble de relations associatives permettant de définir des structures complexes.

1.4.2. Menelas

Ce projet européen avait pour but de proposer une approche d'accès aux informations et aux connaissances médicales à travers différentes langues [Zweigenbaum, 1994]. Une ontologie couvrant le domaine des maladies coronariennes a été développée dans le cadre d'une application pilote. Cette ontologie a été construite à partir de plusieurs sources incluant l'analyse des résumés d'articles et les interviews avec les spécialistes, et elle contient une hiérarchie de 1.800 concepts et une hiérarchie de 300 relations.

1.4.3. Gene Ontology

Gene Ontology (GO) [Ashburner et al., 2001] fournit un vocabulaire partagé, structuré et contrôlé ayant pour but l'annotation et la description des gènes. Elle comprend trois sous-ontologies : (i) la Biological Process ontology (BP), qui décrit les rôles biologiques des gènes, (ii) la Molecular Function (MF), qui spécifie les activités moléculaires d'un gène, et (iii) la Cellular Component ontology (CC), qui décrit les zones cellulaires qu'un gène peut activer.

La version actuelle de GO (2005) contient 18137 concepts représentés dans un graphe orienté sans circuits (DAG). Elle est disponible dans différents formats (XML, RDF, SQL) et peut être interrogée par plusieurs outils tels que AmiGo¹⁶ et DAG-Edit¹⁷.

1.4.4. SNOMED

SNOMED [Price et Spackman, 2000] est une ontologie du domaine de la santé ayant pour but de rendre les connaissances dans le domaine médical accessibles et partageables par toute la communauté médicale. Les concepts de SNOMED permettent l'interopérabilité de plusieurs

¹⁶ <http://www.godatabase.org/cgi-bin/amigo/go.cgi>

¹⁷ <http://www.godatabase.org/dev/java/dagedit/docs/>

systèmes médicaux (par exemple les applications du dossier patient, de la surveillance des maladies, d'indexation d'images médicales...).

La version actuelle contient plus de 366.170 concepts structurés en plusieurs hiérarchies et ayant des identificateurs uniques. Ces concepts sont spécifiés en 933.420 termes médicaux qui sont eux mêmes reliés par 1.46 million relations sémantiques.

Elle est disponible en trois langues à savoir, le français, l'anglais et l'espagnol.

1.4.5. UMLS

Ce projet élaboré par la NLM (National Library of Medicine de Bethesda), déjà à l'origine de MeSH et de Medline, propose depuis 1986 de mettre au point un langage médical unifié [Humphreys et Lindberg, 1993]. Pour ce faire, ce langage repose sur : (1) un métathésaurus qui énumère tout le vocabulaire médical existant et qui comprend des millions de termes ; (2) un réseau sémantique constitué d'une hiérarchie de types sémantiques et d'une hiérarchie de relations ; il représente une classification de tous les concepts représentés dans le métathésaurus ainsi que les relations pouvant exister entre eux.

Ce projet va être détaillé dans le chapitre 3 §4.

Dans ce travail, nous nous intéressons à la description de la totalité du domaine biomédical. Nous notons ainsi, qu'à part UMLS, les ontologies présentées ont été développées, soit pour des cas spécifiques (GO : biologie moléculaire et MENELAS : maladies coronariennes), soit pour des domaines assez larges mais qui ne couvrent qu'une partie du domaine auquel nous nous intéressons (GALEN et SNOMED : le domaine médical).

Notons qu'au-delà de la construction d'ontologies, plusieurs projets biomédicaux exploitant les technologies du Web Sémantique ont vu le jour : [Dieng-Kuntz et al., 2004a] propose un système pour la gestion des connaissances au sein d'un réseau de soin, [D'Acquin, 2005] présente un portail sémantique pour la gestion, la diffusion et l'évolution des connaissances en oncologie et [Dameron et al., 2004] exploite les services Web sémantiques pour faciliter l'accès aux différentes ressources biomédicales.

Décrivons maintenant l'intégration des techniques du Web Sémantique pour la gestion des connaissances.

2. Le web sémantique d'entreprise

L'intégration d'un système de gestion des connaissances dans une entreprise est l'un des aspects les plus importants dans le Knowledge Management. En effet, un tel système offrant un accès global à l'ensemble des sources d'informations, peut jouer un rôle important de support pour le transfert et le partage des connaissances dans l'entreprise.

Au cours des années, plusieurs systèmes ont été proposés pour gérer les informations et les connaissances de l'entreprise : GED (Gestion électronique des documents), base de données, intranet, etc. Ces systèmes, possédant chacun un modèle de données propre, ont posé des problèmes de coopération et d'intégration et présentent un coût non négligeable de développement et de maintenance.

2.1. Mémoire d'entreprise et Web Sémantique

2.1.1. Définition de mémoire d'entreprise

Cette approche de gestion des connaissances a été introduite (il y a une trentaine d'années) par la communauté travaillant dans le domaine de l'intelligence artificielle. En analogie avec la mémoire humaine qui nous permet d'utiliser nos expériences antérieures pour éviter de répéter les mêmes erreurs, une mémoire d'entreprise sert à capitaliser les connaissances des différentes sources de l'organisation afin de les utiliser dans les futures tâches.

Elle est définie par [Van Heijst et al., 1996] comme « *une représentation explicite, persistante, et désincarnée des connaissances et des informations dans une organisation* ». Ces connaissances peuvent porter, par exemple, sur la stratégie de l'entreprise, les procédés de travail, les clients, etc. Ces connaissances stockées peuvent par la suite être combinées afin de faciliter, d'un côté, le processus de création et, d'un autre côté, le processus d'apprentissage.

D'après [Pomian, 1996], le but de la construction d'une mémoire d'entreprise consiste à « préserver afin de les réutiliser plus tard ou le plus rapidement possible, les raisonnements, les comportements, les connaissances, même en leurs contradictions et dans toute leur variété ». En effet, une telle mémoire représente un atout important pour l'entreprise car elle lui permet de pérenniser des connaissances explicites/implicites, de réutiliser de façon meilleure les leçons apprises lors des précédents échecs/succès et de créer de nouvelles connaissances.

[Dieng et al., 2005] qui définit la mémoire d'entreprise comme « la matérialisation explicite et persistante des connaissances et informations cruciales d'une organisation pour faciliter leur accès, partage et réutilisation par les membres de l'organisation dans leurs tâches individuelles et collectives », propose un cycle de vie de cette mémoire (Figure 3) qui repose sur les étapes suivantes :

1. Détection des besoins en mémoire d'entreprise : définition des scénarios d'utilisations.
2. Construction de la mémoire : choix des techniques et matérialisation de la mémoire d'entreprise (base d'expériences, mémoire documentaire, portail, base de connaissances, etc.).
3. Diffusion de la mémoire : utilisation de l'Intranet de l'entreprise par exemple pour diffuser les connaissances via des collecticiels ou des serveurs de connaissances.
4. Utilisation de la mémoire : recherche des informations stockées dans la mémoire en utilisant des moteurs de recherche (méthode « pull ») ou dissémination proactive des informations vers les utilisateurs selon leurs centres d'intérêt (méthode « push »).
5. Évaluation de la mémoire : reposer sur les retours d'expériences du point de vue usage et technique.

6. Maintenance et évolution de la mémoire : adapter la mémoire (qui peut être centralisée ou distribuée) aux nouveaux besoins et contextes afin de la garder à jour par rapport à la stratégie de l'entreprise. Les modifications doivent être dynamiques et cohérentes.

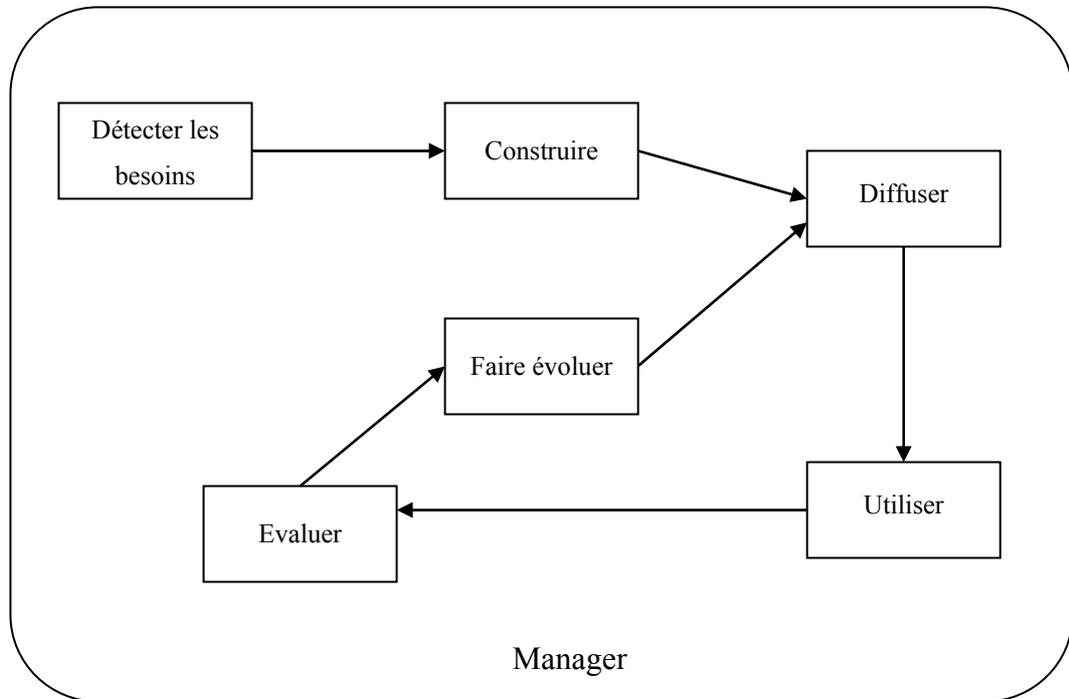


Figure 3 - Cycle de vie de la mémoire d'entreprise [Dieng-Kuntz et al., 2005]

2.1.2. L'approche ACACIA

L'approche de l'équipe ACACIA repose sur la combinaison des technologies du Web sémantique et les moyens de communication de l'entreprise (comme les intranets et les intrawebs) afin de caractériser et construire une mémoire d'entreprise.

En faisant l'analogie entre les ressources du web et les ressources d'une entreprise, [Dieng, 2004b] propose de matérialiser la mémoire d'entreprise à travers un «web sémantique d'entreprise (WSE)» (ou « web sémantique d'organisation (WSO)») en utilisant les ontologies qui fournissent un cadre formel pour décrire les différentes sources de connaissances de l'organisation et qui guident la création d'annotations sémantiques facilitant la description, le partage et l'accès à ces sources.

Les principales composantes de ce web sémantique d'entreprise sont les suivantes :

- Les ressources : peuvent être des bases de données, des personnes, des documents (dans tous les formats), des services/logiciels, etc.
- Les ontologies : décrivant le vocabulaire partagé par les différentes communautés de l'entreprise.

- Les annotations sémantiques : décrivant des méta données sur les ressources en se basant sur les concepts et les relations de l'ontologie.

La Figure 4 représente l'architecture d'un tel web sémantique d'entreprise.

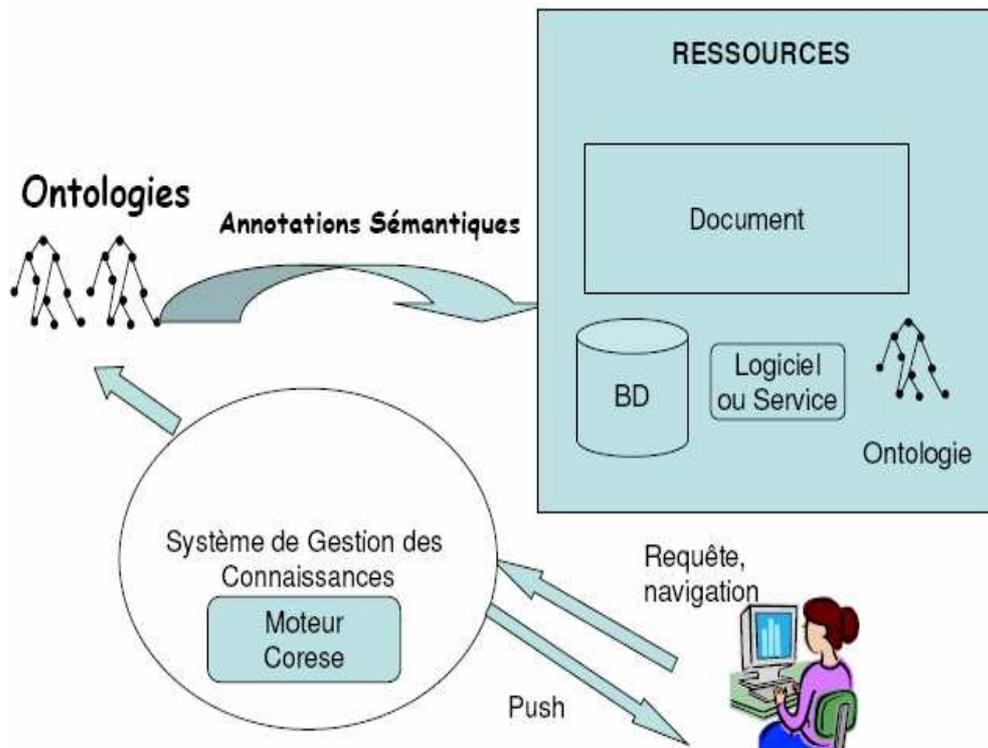


Figure 4 - Architecture d'un web sémantique d'entreprise [Dieng-Kuntz, 2004b]

Dans ce qui suit, nous présentons des projets basés sur cette approche et réalisés au sein de l'équipe ACACIA.

2.1.2.1. Samovar

Samovar [Golebiowska et al., 2001] est un système/méthode de capitalisation de connaissances dans le domaine automobile. L'objectif de ce projet était d'améliorer l'exploitation des informations stockées dans un système de gestion de problèmes afin de les mettre à disposition pour les projets futurs chez Renault.

L'approche Samovar repose sur l'utilisation de plusieurs ontologies (Problème, Pièce, Prestation et Projet) construites à partir de deux sources différentes, à savoir (a) les données structurées contenues dans les bases de données et (b) les données textuelles extraites automatiquement des notes des concepteurs (stockées dans des champs textuels de la base de données). Ces ontologies sont représentées en RDFS et servent à créer des annotations RDF sur

la base des problèmes, et à faciliter la recherche d'informations en utilisant le moteur CORESE (voir §3.2.3).

Les résultats des tests effectués sont intéressants et montrent l'intérêt de l'utilisation des techniques du web sémantique dans la construction d'une mémoire d'entreprise.

2.1.2.2. *CoMMA*

CoMMA [Gandon, 2003] (« Corporate Memory Management through Agents ») est un projet européen qui a permis de construire une mémoire d'entreprise matérialisée dans une base documentaire annotée par des annotations sémantiques basées sur l'ontologie O'CoMMA. Une société d'agents coopérant et guidés par l'ontologie permettait la recherche d'information dans cette mémoire d'entreprise, l'ajout d'annotations dans la base d'annotations et l'interaction avec les utilisateurs en intégrant CORESE dans un agent « Moteur de recherche ».

Deux scénarios ont été étudiés dans ce projet :

- L'insertion des nouveaux employés d'une entreprise;
- L'assistance à la veille technologique.

Le projet CoMMA a traité aussi l'aspect distribué de la mémoire d'entreprise. Des agents dédiés aux connexions ont été proposés, ces agents proposaient des services de pages jaunes et de pages blanches pour fournir des informations sur un agent pouvant offrir un service particulier.

2.1.2.3. *CORESE*

CORESE (COncceptual REsource Search Engine) [Corby et al., 2004] est un moteur de recherche sémantique qui peut être utilisé pour interroger les différentes ressources d'un web sémantique d'entreprise. Nous détaillons les fonctionnalités de ce moteur dans le Chapitre 5 §2.2.

2.1.3. Autre approche pour le développement des WSO

[Fortier, 2005] a proposé récemment une approche générique pour la conception et le développement de webs sémantiques d'organisations. Cette approche est dotée de deux ontologies de haut niveau, l'une dédiée à la description de l'organisation et l'autre à la description des ressources textuelles de cette organisation. Lors du développement d'un nouveau web sémantique d'organisation, les concepts abstraits de ces deux ontologies sont spécialisés par des concepts spécifiques à cette organisation.

Comme dans CoMMA, l'approche proposée dans ce travail traite le problème de distribution des connaissances en proposant un méta-modèle décrivant les sources de connaissances en faisant abstraction de leur spécification et de leur localisation. Ce méta-modèle est ensuite utilisé pour la construction de documents virtuels qui contiennent des informations provenant de sources hétérogènes et distribuées.

Un environnement logiciel nommé K^2M^3 supportant cette approche a été développé et évalué dans un projet de construction d'un web sémantique d'organisation visant à faciliter la gestion des dossiers patients dans les hôpitaux.

Dans les sections précédentes, nous avons présenté le Web Sémantique ainsi que les différents travaux réalisés autour de ce thème de recherche. Nous avons aussi mis l'accent sur les approches de construction de la mémoire d'entreprise en exploitant les techniques du Web Sémantique. Ces approches se basent essentiellement sur les ontologies et les annotations sémantiques pour la description du contenu sémantique des ressources de l'entreprise. Dans ce travail, ce qui nous intéresse davantage, c'est l'automatisation de la création de ces annotations sur les ressources textuelles d'un groupe (les personnes travaillant sur les expériences des puces à ADN) dans le but d'utiliser ces annotations pour alimenter une mémoire d'expériences.

3. Extraction des connaissances à partir des textes

Malgré ses avantages, la création d'une annotation sémantique est un processus difficile et coûteux (temps, personnes...). Ceci a amené les chercheurs à travailler sur la possibilité de (semi-)automatiser cette tâche en utilisant des techniques d'acquisition des connaissances à partir des textes.

La problématique est donc d'acquérir, à partir d'un document, un ensemble de connaissances utiles pour la construction d'une annotation sémantique pour ce document. Le premier objectif consiste à extraire des candidats-termes représentatifs des connaissances du domaine (i.e. des termes désignant des instances des concepts de l'ontologie). Le deuxième objectif est d'extraire des relations entre ces termes (i.e. des instances des relations de l'ontologie).

Ces objectifs ne sont pas très éloignés des objectifs de l'indexation automatique des documents pour les systèmes de recherche d'informations classiques. En effet, la plupart de ces méthodes essayent de capturer des termes représentatifs du contenu informationnel du corpus et des relations (simples) reliant ces termes. La grande différence entre ces deux problématiques (annotation sémantique et indexation à partir des textes) réside dans le fait que pour la première, (i) l'extraction des connaissances est guidée par un modèle déjà prédéfini du domaine (i.e. l'ontologie) et (ii) les termes extraits ne représentent pas uniquement le contenu informationnel du document mais représentent aussi les connaissances du domaine traitées dans ce document.

Dans cette partie, nous présentons quelques notions sur le TALN et nous proposons un panorama des outils de traitement automatique de la langue naturelle (TALN) utilisés dans plusieurs domaines et permettant l'extraction de termes et de relations à partir des textes.

3.1. Quelques aspects sur le TALN

3.1.1. Les différentes étapes de l'analyse d'un texte

L'analyse des textes comporte plusieurs étapes distinctes allant du simple découpage du texte en mots à la présentation de son contenu. Les différents systèmes de TALN implémentent, soit la totalité de ces étapes, soit une combinaison de certaines étapes.

L'analyse morphologique

Cette analyse permet de traiter les variations de surface de chaque mot du texte (chaînes de caractères séparées par un espace) en prenant en compte les formes fléchies ou variations apparentes du mot. Un analyseur morphologique permet ainsi de :

- Traiter les formes du pluriel d'un mot.
- Identifier les caractères minuscules ou majuscules, les abréviations, etc.
- Reconnaître les locutions, les expressions, les noms composés, etc.
- Isoler une seule forme canonique pour toutes les formes rencontrées d'un mot.

L'unité minimale produite après une telle analyse s'appelle morphème.

L'analyse lexicale

L'analyse lexicale permet, d'une part, de rechercher l'existence des mots et des expressions dans un dictionnaire linguistique et d'autre part, de confirmer ou d'infirmer l'existence des morphèmes identifiés par l'analyse morphologique.

L'analyse syntaxique

L'analyse syntaxique permet de représenter, sous forme symbolique ou graphique, la ou les structures syntaxiques d'un texte. En d'autres termes, il s'agit de la mise en évidence des structures d'agencement des catégories grammaticales (nom, verbe, adjectif, etc.), afin d'en découvrir les relations formelles ou fonctionnelles (par exemple, sujet, verbe et complément).

Une grammaire probabiliste contient des informations statistiques concernant la fréquence d'utilisation de ses règles. Ces statistiques « guident » l'analyseur syntaxique lorsque celui-ci a le choix entre plusieurs règles pour choisir celle qui est la plus probable.

L'analyse sémantique

L'analyse sémantique est l'étude linguistique du sens. L'objectif principal de cette analyse est de déterminer le sens des mots des phrases. Les mots et les structures des phrases identifiés lors des analyses morphologique, lexicale et syntaxique, constituent des indices pour la détermination du sens. Les analyseurs sémantiques utilisent généralement des lexiques sémantiques représentés sous forme de graphes associés à l'ensemble des mots utilisés dans le domaine.

L'analyse pragmatique

L'analyse pragmatique permet d'utiliser les connaissances pragmatiques afin d'interpréter des situations du monde réel. Cette étape est importante pour le processus de compréhension d'un texte ou d'une phrase, elle représente le lien entre l'analyse linguistique et le monde réel.

3.1.2. Notion de syntagme

« Un syntagme est un ensemble de mots formant une seule unité catégorielle et fonctionnelle, mais dont chaque constituant, parce que dissociable (contrairement au mot composé), conserve sa signification et sa syntaxe propres. Un syntagme constitue donc une association occasionnelle, libre, alors que le mot composé est une association permanente (lorsqu'un syntagme se fige, il devient bien sûr un composé détaché, soit une locution) »¹⁸.

Il s'agit donc d'un groupe de mots formant une unité à l'intérieur de la phrase. Dans le cadre de l'analyse syntaxique d'une phrase, il s'agit d'une segmentation en unités fonctionnelles appelées syntagmes. Par exemple, on peut citer les types de syntagme suivants : syntagme nominal, syntagme verbal, syntagme adjectival, etc.

Exemples de syntagmes extraits de notre corpus de test :

Increase the expression: Syntagme verbal

Mouse's cells: Syntagme nominal

3.1.3. Variations liées aux syntagmes

La variabilité morphosyntaxique est un problème récurrent dans la reconnaissance automatique des syntagmes. Certains systèmes, comme FASTR [Jacquemin, 1997] ont pris en compte ce problème en traitant les variations suivantes :

Les variations morphologiques

Un mot constituant le syntagme peut être au passé, au pluriel, conjugué ou remplacé par un mot de même racine mais de nature syntaxique différente (peupler, peuple, peuplement).

Les variations syntaxiques

Les modifications d'expressions peuvent aussi provenir de variations syntaxiques de différentes natures :

- *coordination* : combinaison de deux termes avec un mot tête commun ou un argument commun par exemple « élections présidentielle et législatives » est une variation de coordination du syntagme « élections législatives ».
- *substitution/modification* : la substitution est le remplacement d'un mot par un syntagme, la modification est l'insertion d'un modificateur sans référence à un autre terme : par exemple « Satellite géostationnaire de communication » est une substitution de « Satellite de communication » si « Satellite géostationnaire » est un syntagme (sinon, c'est une modification).

¹⁸ <http://fr.wikipedia.org/wiki/Syntagme>

Les variations sémantiques

Un mot du syntagme peut être remplacé par un synonyme.

Dans notre travail, nous nous intéressons particulièrement à certains aspects de l'analyse morphologique (i.e. la lemmatisation, les formes fléchies des verbes), de l'analyse syntaxique (repérage de syntagmes) et de l'analyse sémantique (association du terme au concept qu'il décrit).

Dans ce qui suit, nous introduisons la notion de candidat-terme qui peut être dans notre cas soit un mot simple, soit un syntagme.

3.2. Les approches d'extraction de candidats-termes

Cette tâche consiste à re-traiter le document après l'avoir analysé afin d'en extraire, sous la forme la moins ambiguë possible, un ensemble de candidats-termes, qui soient stables quelque soit le contexte. Pour effectuer cette tâche, plusieurs approches sont possibles, parmi lesquelles nous citons les approches statistiques, les approches syntaxiques et les approches mixtes que nous détaillons dans ce qui suit.

3.2.1. Les approches statistiques

Ces approches utilisent seulement les co-occurrences de mots. Le principe est que si deux mots co-occurrent souvent dans un certain type de contexte, alors ils peuvent être regroupés dans un terme.

Le calcul de co-occurrences varie selon le contexte et selon les besoins. Il peut se faire dans le même document, le même paragraphe, la même phrase, ou dans une certaine distance.

[Ouesleti et al., 1996] présente une méthode basée sur une approche statistique appelée 'méthode des segments répétés'. Cette méthode permet la détection de chaînes constituées de morceaux (mots, symboles, ponctuation...) apparaissant ensemble plusieurs fois dans le même texte. Cette méthode est implémentée dans un prototype appelé LIKES¹⁹ qui renvoie une liste des termes extraits filtrés avec une fréquence minimale d'apparition.

3.2.2. Les approches syntaxiques

Ces approches utilisent certaines informations syntaxiques dans le choix des termes et supposent que le document a déjà subi une analyse morphologique et une analyse syntaxique. Parmi ces approches, nous citons deux familles :

L'utilisation de patrons morpho-syntaxiques

C'est l'une des techniques les plus utilisées pour l'extraction de termes. Les systèmes basés sur cette technique supposent que les termes à extraire obéissent à des régularités syntaxiques stables. Ces systèmes prennent en entrée un ensemble de patrons constitués d'une suite de catégories grammaticales et qui peuvent être par exemple :

¹⁹ http://www.atala.org/article.php3?id_article=177

NOM NOM / ADJQ NOM / NOM PREP NOM ...²⁰

Toutes les occurrences de mots correspondant à ces patrons sont extraites comme des candidats-termes potentiels.

NOMINO [David et Plante, 1990] est considéré comme l'un des premiers systèmes à avoir utilisé cette technique. Proposé à la base pour la construction de bases de connaissances, cet outil permet aussi le repérage de syntagmes nominaux appelés UCN (Unités Complexes Nominales). NOMINO implémente toutes les étapes du traitement linguistique, il détecte ainsi les noms présents dans le document/corpus et en s'appuyant sur des règles d'expansion (une grammaire de patrons morpho-syntaxiques) propose une liste d'UCN triée, soit par fréquence, soit par ordre alphabétique.

Contrairement à NOMINO, LEXTER [Bourigault et al., 1996] prend en entrée un corpus préalablement étiqueté et désambiguïté. Cet outil permet également l'extraction de candidats termes sous forme de syntagmes nominaux décomposés en tête et expansion. LEXTER implémente une méthode originale qui consiste à éliminer d'abord les mots ne pouvant constituer un terme (verbe, conjonction, pronom...) pour ensuite relever des syntagmes nominaux maximaux.

[Bourigault et Fabre, 2005] propose une évolution de LEXTER vers un nouveau système appelé Syntex en rajoutant deux extensions importantes : (i) la prise en compte de l'anglais, et (ii) l'extension de la couverture du système à l'extraction des syntagmes verbaux.

Ces trois systèmes proposent un réseau de termes dont les relations lexicales (tête et expansions) peuvent conduire à des relations sémantiques.

L'utilisation des règles de transformation

Ces méthodes permettent d'extraire des termes complexes à partir de connaissances extérieures servant de référence. Généralement, elles identifient des variantes de termes fournis par un thésaurus ou un vocabulaire contrôlé.

FASTR [Jacquemin, 1997] permet de repérer des variations de termes à partir d'une liste de termes initiaux. Il prend comme entrée un ensemble de mots simples étiquetés et applique des règles (modification, substitution, permutation...) pour détecter les variantes.

3.2.3. Les approches mixtes

Ces approches combinent des méthodes à orientation statistique et des méthodes à orientation syntaxique. Elles utilisent généralement des calculs statistiques afin d'affiner leurs méthodes d'extraction linguistique.

[Daille, 1994] utilise une méthode mixte statistico-syntaxique dans son système ACABIT qui effectue des calculs statistiques sur des termes composés repérés dans le corpus. Il s'agit de repérer des candidats-termes à partir de schémas syntaxiques puis de les filtrer à l'aide de méthodes statistiques. Cette méthode établit une liste de types élémentaires de composés

²⁰ ADJQ : adjectif qualificatif ; PREP : préposition

nominaux du domaine de télécommunication ainsi qu'une topologie des moyens dont dispose la langue pour engendrer des formes complexes à partir de formes élémentaires: la surcomposition, la modification et la coordination. Parmi ces divers types de composés, elle ne retient que certains composés de longueur 2: Nom Nom, Nom Adjectif et Nom Préposition Nom avec quelques possibilités d'insertion de modificateurs (Adjectif, Nom ou Adverbe) au sein des schémas retenus. ACABIT peut être appliqué à des corpus bilingues et propose une liste de termes avec leurs traductions.

Le système Xtract proposé par [Smadja, 1993] se base sur un filtrage statistique sur les fréquences des co-occurrences des mots composant une collocation (une co-occurrence de mots ayant une forme syntaxique précise). Il repose sur deux hypothèses, à savoir : (1) les mots dans une collocation apparaissent ensemble plus fréquemment que par hasard et (2) les mots apparaissent dans une fenêtre de plus ou moins 5 mots correspondant à des contraintes syntaxiques particulières.

3.3. Les approches d'extraction de relations

Après avoir présenté quelques approches d'extraction de candidats-termes, notre but est maintenant d'étudier les différents travaux proposés pour extraire des relations sémantiques entre ces termes. Nous présentons trois grandes familles d'approches d'extraction de relations à savoir : l'étude statistique, l'exploitation des contextes syntaxiques et l'utilisation de marqueurs.

3.3.1. Extraction des relations par étude statistique

Ces approches reposent sur le principe que les termes qui co-occurrent ensemble ont de fortes chances d'être liés par des relations sémantiques. Elles exploitent donc la distribution des termes dans le document/corpus en utilisant des techniques de fouille basées sur des méthodes statistiques.

Parmi ces travaux, nous pouvons citer ceux de [Smadja, 1993] qui étudie les fréquences des co-occurrences des termes en anglais afin de proposer des relations entre ces termes. [Toussaint et al., 1997] a également étudié ce phénomène de co-occurrence pour le français et propose de construire des regroupements de termes (clusters). Une fois ceux-ci présentés à un expert, ce dernier peut décider de la nature de la relation qui regroupe l'ensemble des termes (synonymie, hyponymie ou méronymie).

Ces méthodes n'extraient pas vraiment des relations mais proposent un nuage de termes, à partir duquel un expert pourrait déduire des relations ou des classes conceptuelles.

3.3.2. Extraction des relations par exploitation des contextes syntaxiques

Comme pour les premières, ces approches exploitent le principe de co-occurrence des termes pour la détection des relations. Par contre, elles utilisent la distribution syntaxique des termes à la place des calculs statistiques pour extraire les relations.

En effet, dans [Grefenstette, 1994], le système SEXTANT extrait les contextes des termes sous une forme syntaxique : Adjectifs-Noms, Noms-Noms et Verbes-Noms. L'hypothèse avancée par Grefenstette consiste à dire que tous les termes partageant les mêmes contextes

peuvent être liés sémantiquement. Ce système a été proposé pour la construction automatique de thésaurus.

Dans la même optique, [Assadi, 1998] propose le système LexiClass qui se base sur les résultats générés par LEXTER pour extraire des relations entre les termes proposés par ce dernier. Ce système exploite la décomposition syntaxique des syntagmes nominaux proposés par LEXTER en « syntagme + adjectif » afin de les regrouper dans des contextes adjectivaux, et ce en utilisant des méthodes de classification automatique hiérarchique ascendante. Ainsi, et à partir de « activité du Na primaire » et de « activité du sodium primaire », LexiClass peut déduire une relation de synonymie entre « activité du Na » et « activité du Sodium ».

Cette même méthode est utilisée dans le système Upery [Bourigault, 2002], qui, lui, exploite les résultats de Syntex [Bourigault 2005].

3.3.3. Extraction des relations par l'utilisation des marqueurs

Ces approches se basent sur les traces linguistiques qui signalent les relations sémantiques dans le texte (ces traces peuvent être liées, soit à la langue, soit au domaine) pour construire des marqueurs permettant la détection de ces relations. Un marqueur peut être considéré comme une formule linguistique que les mots désignant une relation dans le texte doivent vérifier.

Plusieurs travaux utilisant cette méthode ont été réalisés, parmi lesquels nous citons SEEK [Jouis, 1993] et CAMELEON [Séguéla et Aussenac-Gilles, 2000].

SEEK est un système qui permet l'extraction de relations dites 'statiques' (identification, incompatibilité, mesure, comparaison, inclusion, appartenance, localisation, possession et attribution) entre des termes déjà identifiés dans un texte.

Il se base sur une méthodologie qui consiste à explorer les textes pour trouver des traces linguistiques particulières pouvant jouer un rôle de marqueur pour la détection de relations. Il utilise des listes de marqueurs (3300 marqueurs dans 240 listes) et des règles morphologiques qui permettent de faire de l'inférence de relations sémantiques. Le langage de déclaration des règles d'exploration est du type : SI <conditions> ALORS <actions> OU <conclusions>. Les prémisses des règles contiennent des combinaisons d'éléments lexicaux et des marqueurs possédant des exceptions.

CAMELEON est à la fois un système et une méthode de gestion et de réutilisation de bases de marqueurs pour la génération de relations sémantiques. Un marqueur est défini comme un patron lexico-syntaxique désignant dans le discours une relation entre deux termes et il implique la spécification de trois éléments : une relation, un identifiant et le schéma permettant l'extraction.

Un exemple de marqueur défini dans CAMELEON :

Relation : HYPONOMIE

Identifiant : Y_EST_LE_X_LE_PLUS

Schéma : Y ETRE ()*mots ARTICLE_DEFINI X ARTICLE_DEFINI (plus|moins)

CAMELEON propose des marqueurs génériques utilisant des connaissances linguistiques de la langue générale et offre la possibilité de générer des marqueurs spécifiques pour un domaine particulier (en fonction du corpus). Les relations traitées sont la méronymie et l'hyponymie.

Nous avons fait, dans cette partie, un tour d'horizon sur les techniques de base pour l'extraction des connaissances, que ce soit au niveau des termes ou des relations. Les méthodes actuelles tournent autour de ces techniques en y apportant certaines améliorations ou extensions...

Etant intéressés, dans notre travail, par les connaissances contenues dans les textes biologiques, il nous semble utile de faire, en plus, le point sur l'extraction des connaissances à partir des textes (communément appelé fouille de texte) biologiques. En effet, les travaux qui s'intéressent à ce domaine, prennent en compte des informations liées au domaine biomédical, ce qui ajoute à leur efficacité par rapport aux systèmes génériques de fouille de texte.

4. Fouille des textes pour le domaine biologique

Au cours de ces dernières années, il y a eu une croissance énorme de la quantité de données biomédicales expérimentales et informatiques, spécifiquement dans les secteurs de la génomique et de la protéomique. Cette croissance est accompagnée d'une augmentation accélérée du nombre de publications biomédicales discutant les résultats. Cette abondance de littérature a poussé la communauté biomédicale à s'intéresser de plus près aux techniques de fouille de textes dans le but d'extraire des informations pouvant être réutilisées dans leurs tâches d'analyse.

Comme nous l'avons fait dans la partie précédente pour le traitement linguistique général, nous nous intéressons, dans cette partie, aux techniques d'extraction de termes dans le domaine biomédical ainsi qu'aux techniques d'extraction de relations pouvant exister entre ces termes. Mais avant de détailler ces techniques, nous présentons quelques apports de la fouille des textes biomédicaux.

4.1. Rôles des techniques de TALN dans le domaine biomédical

4.1.1. Construction des bases de connaissance biologiques

Une des motivations de l'utilisation des techniques de TALN dans le domaine biomédical est la création et l'enrichissement de base de connaissances biologiques.

L'idée est de collecter des informations à propos d'entités biologiques/médicales et de les stocker directement dans des bases de données ou de les formaliser dans une ontologie et ce en se basant sur l'hypothèse qu'une grande partie de ces informations est contenue dans la littérature. Le but de l'utilisation du traitement automatique des textes dans une telle approche est de réduire le coût en temps généré par les approches manuelles.

Le système PubGene²¹ représente un bon exemple dans ce domaine, son principe repose sur le fait que deux protéines (resp. deux gènes) n'apparaissent pas ensemble dans un texte par hasard. Donc après l'extraction des noms de gènes et de protéines, le système calcule les co-occurrences de ces termes dans les textes afin de définir des interactions entre eux et de les stocker dans une base de données accessible à tout le monde.

Dans la communauté française, plusieurs équipes se sont intéressées à la capitalisation des connaissances biomédicales à partir des textes. [Zweigenbaum et al., 2003] propose de construire un lexique biomédical français UMLF, inspiré de UMLS en extrayant des ressources lexicales à partir de corpus spécialisés.

Dans [LeMoigno et al., 2002], les auteurs rapportent la construction d'une ontologie dans le domaine de la réanimation chirurgicale fondée sur des outils d'analyse syntaxique et d'analyse distributionnelle de corpus.

4.1.2. La recherche d'informations

Rappelons que la tâche de recherche d'informations consiste généralement à trouver dans une base documentaire, des documents pertinents qui contiennent des informations répondant à une question/requête de l'utilisateur. Dans le domaine biomédical, nous retrouvons tous les problèmes standards liés à cette tâche (i.e. le silence et le bruit), auxquels viennent s'ajouter des problèmes spécifiques au domaine.

En effet, les moteurs de recherche sur le web (google, yahoo...) traitant nos requêtes, nous renvoient généralement des milliers de documents parmi lesquels un petit nombre correspond 'exactement' à notre requête : c'est le problème classique qui consiste à rechercher 'une aiguille dans une botte de foin'. En revanche, un moteur de recherche spécialisé (i.e. Pubmed) traitant une requête qui porte sur un gène particulier, peut renvoyer aussi des milliers de documents mais qui sont cette fois en majorité pertinents pour les utilisateurs : c'est un problème plus complexe qui consiste à rechercher 'une aiguille dans une botte de foin constituée d'aiguilles' (analogie faite par [Cohen et Hunter, 2005]).

L'extraction automatique des connaissances à partir des textes peut jouer un rôle très important dans la résolution de ce problème. Ces connaissances peuvent servir à :

- Mieux annoter les documents, ce qui améliore la qualité du calcul de correspondance entre un document et une requête;
- Classer les documents retournés aux utilisateurs selon les informations qu'ils contiennent, ce qui rend la navigation dans les résultats plus simple;
- Améliorer la visualisation des résultats en faisant un zoom sur les informations intéressantes contenues dans les documents.

Néanmoins, la spécificité des textes biomédicaux et plus spécialement les textes biologiques, nécessite des adaptations des techniques de fouille de textes classiques déjà présentées dans ce chapitre. C'est dans cette direction que plusieurs travaux de recherche ont été menés dans le but de proposer des méthodes et techniques de TALN pour le domaine de la biologie.

²¹ <http://www.pubgene.org/>

4.2. Méthodes et outils de TALN en biologie

Dans ce qui suit, nous détaillons les techniques de traitement automatique de la langue utilisées dans le domaine biomédical pour l'identification des termes et des interactions entre eux, et nous présenterons quelques outils basés sur ces techniques.

4.2.1. Identification des termes

4.2.1.1. Méthodes basées sur les dictionnaires

Ces méthodes utilisent des ressources terminologiques existantes (dictionnaire, lexique, thésaurus...) dans le but de localiser les occurrences des termes dans les textes.

L'application de la version simple de ces méthodes, c'est-à-dire faire la correspondance directe entre les entrées du dictionnaire et les entités textuelles ne donne pas de résultats satisfaisants du point de vue précision et rappel.

Ces mauvais résultats sont dus essentiellement à des problèmes d'homonymie (i.e, en anglais par exemple, des mots communs comme 'and', 'by' ou 'for' sont détectés comme noms de gènes) et des problèmes de variations linguistiques liés à (i) la ponctuation (mdm-2 et mdm2), (ii) l'utilisation de l'alphabet grec (p53alpha et p53a), et (iii) l'ordre des mots (integrin alpha4 et alpha4 integrin).

Afin de remédier à ces problèmes, beaucoup d'améliorations ont été ajoutées à ces méthodes telles que l'utilisation de dictionnaire de synonymes, le filtrage des mots vides et le traitement des variations. [Krauthammer et al., 2000] proposent de coder les dictionnaires et les textes avec le code nucléique (l'alphabet formé de 4 lettres {A, C, G, T}) et d'utiliser l'algorithme BLAST [Altschul et al., 1994] utilisé pour l'alignement des séquences ADN pour identifier les termes qui ont une similarité forte. L'expérience menée sur un corpus de test et la base GenBank²² a donné de bons résultats.

4.2.1.2. Méthodes basées sur les règles

Ces méthodes reposent sur la création (manuelle) de règles d'extraction basées sur les particularités spécifiques à une classe de termes. Ces particularités peuvent être (i) morphologiques : les mots se terminant par -ase et -in peuvent être considérés comme des enzymes ou des protéines et (ii) orthographiques : les termes vérifiant l'expression régulière [a-z]+[0-9] peuvent être considérés comme des gènes (une séquence de lettres suivi d'une séquence de chiffres).

[Fukuda et al., 1998] propose une méthode pour la reconnaissance automatique des noms de protéines ; Ils exploitent le fait que les noms des protéines sont souvent en majuscules et comportent des caractères spéciaux et des chiffres.

Quant à [Hobbs, 2000], il a adapté un outil de reconnaissance automatique d'entités nommées standard (FASTUS [Hobbs et al., 1997]) pour la reconnaissance des noms de gènes et

²² <http://www.psc.edu/general/software/packages/genbank/genbank.html>

de protéines. Cet outil est basé sur une cascade de transducteurs à états finis qui permettent de reconnaître des unités complexes (par exemple : '3,4-dehydroproline').

D'autres utilisent des règles d'associations qui permettent de mettre en évidence des corrélations entre des éléments textuels. Un corpus prétraité est utilisé pour l'extraction de ces règles qui sont ensuite présentées à un expert du domaine pour les valider. Une fois validées, les règles d'associations sont classifiées selon des mesures probabilistes et appliquées sur les textes afin d'extraire des termes du domaine. [Cherfi et al., 2005] présentent une méthodologie de fouille de textes biologiques en utilisant les règles d'associations.

4.2.1.3. Méthodes basées sur les techniques d'apprentissage

Comme pour toutes les méthodes basées sur les algorithmes d'apprentissage, ces méthodes ont pour principe de détecter des particularités caractérisant une classe de termes à partir de données d'apprentissage (corpus déjà traité manuellement en affectant les termes à des classes prédéfinies).

A chaque classe, l'algorithme affecte des caractéristiques souvent orthographiques (i.e. combinaison de lettres et de chiffres, terme commençant par une lettre majuscule) ou morpho-syntaxiques (les patrons d'extraction). Ces informations sont ensuite utilisées par des algorithmes de classification standard qui classent les termes dans leurs catégories adéquates.

Plusieurs expériences ont été réalisées en utilisant différents algorithmes de classification, par exemple [Collier et al., 2000] se sont basés sur les chaînes cachées de Markov (HMM) alors que [Kazama et al., 2002] ont utilisé les machines à vecteurs de support (SVM). Ces méthodes sont gourmandes en temps et en ressources; en outre, elles sont confrontées à un autre problème qui est le manque de corpus déjà traité pour effectuer leur apprentissage. En effet, la majorité des expériences sont réalisées sur le même corpus GENIA [Kim et al., 2003].

Le projet pluridisciplinaire CADERIGE [Nédellec et Nazarenko, 2001] fait intervenir plusieurs équipes françaises de compétences différentes (biologie, apprentissage et TALN) dans le but de concevoir des outils d'analyse de données biologiques à partir des textes et en se basant sur les techniques d'apprentissage. Un éditeur d'annotation a été développé et une méthode d'apprentissage de patrons d'extraction a été mise au point.

4.2.2. Identification des interactions

L'explication de phénomènes biologiques, pharmacologiques ou médicaux, se base généralement sur la détection d'une interaction entre gènes, protéines ou molécules. Bien qu'une partie de ces interactions soit stockée dans des bases de données, une grande partie d'entre elles est exprimée en langue naturelle et donc stockée dans les publications du domaine. Plusieurs méthodes de fouille de textes biomédicaux pour l'extraction de ces interactions ont été proposées.

Pour la détection d'interaction de types gène-gène ou gène-protéine, [Nédellec et al., 2004] propose une méthode composée de trois étapes : (1) la sélection d'un ensemble de fragments de textes contenant ce genre d'interactions, (2) l'utilisation d'algorithmes d'apprentissage sur ces fragments pour définir des règles d'extraction et (3) l'application des règles sur les documents pour extraire les interactions.

[Shatkay et al., 2002] propose une méthode d'extraction de relations fonctionnelles entre les gènes. L'hypothèse consiste à dire que si deux gènes apparaissent régulièrement dans des documents traitant le même phénomène (même séparément), alors une relation pourrait exister entre ces deux gènes. Ils font appel à des modèles statistiques qui décrivent la fréquence des mots dans les documents afin de les classifier selon des thèmes pour déduire ensuite les fonctions des gènes qui apparaissent dans ces documents.

[Rindflecsh et al., 2000] propose un système d'extraction d'informations sur les relations qu'entretiennent gènes, médicaments et cellules. Il s'agit de trouver des relations du type : 'Dans les cellules de type C, l'expression du gène G est inhibée (ou activée) par le médicament M', ou du type : 'Les cellules du type C acquièrent une résistance (ou une sensibilité) au médicament M quand le gène G s'exprime'. Le système est basé sur la reconnaissance de la co-occurrence dans une même phrase d'un gène, d'un type cellulaire et d'un médicament.

D'autres travaux sur le même thème sont présentés dans [Staab, 2002] et [Shatkay et Feldman, 2003]. Les résultats de ces systèmes permettent de créer des réseaux d'interaction entre gènes et protéines qui peuvent jouer un rôle important dans l'interprétation des résultats d'une expérience.

4.2.3. Quelques outils

Après avoir détaillé les différentes méthodes et techniques de fouille de textes biomédicaux, nous présentons brièvement quelques outils d'extraction et de gestion des connaissances biomédicales à partir des textes.

4.2.3.1. *Medminer*

Medminer²³ est un système qui a été conçu spécialement pour les biologistes travaillant sur l'expression des gènes. Il permet d'effectuer des recherches sur plusieurs gènes à la fois dans la base documentaire PubMed afin de trouver les corrélations entre eux. Les résultats renvoyés par le moteur de recherche de PubMed sont ainsi filtrés, classifiés dans des catégories prédéfinies par le biologiste. Le filtrage est fait, soit par le calcul des fréquences des termes dans les documents, soit par le calcul des co-occurrences entre les termes.

4.2.3.2. *PubMiner*

Pubminer [Eom et Zhang, 2004] combine des techniques d'apprentissage (HMM et SVM) avec des techniques de TALN pour traiter les résumés de la base PubMed afin d'en extraire des entités nommées (gène, protéine) et de possibles interactions entre elles.

Ce système permet la visualisation des résultats sous la forme d'un graphe, où les nœuds représentent les noms des gènes et des protéines et les arcs représentent les interactions possibles ; l'utilisateur garde toujours un lien entre le graphe et les documents textes traités.

²³ <http://discover.nci.nih.gov/textmining/main.jsp>

4.2.3.3. *Textpresso*

[Muller et al., 2004] propose un système d'extraction et de recherche d'informations pour les articles du domaine biomédical. TextPresso se base sur une ontologie définie pour décrire les connaissances présentes dans les textes. Il identifie les termes (instances des concepts de l'ontologie) en utilisant des expressions régulières et les encadre avec des balises XML directement dans le texte. Il offre aussi un ensemble d'interfaces dédiées pour interroger efficacement les annotations en se basant sur l'ontologie.

Notons que Textpresso (i) intègre les annotations (XML) directement dans le texte ce qui rend leur utilisation par un autre système très difficile et (ii) nécessite la définition d'un nombre très grand d'expressions régulières (des milliers) pour pouvoir extraire les termes pertinents.

5. Conclusion

Avec la proposition du **Web Sémantique** comme nouvelle génération du Web, nous avons vu émerger plusieurs travaux (proposition de standards et de modèles) offrant des techniques facilitant le partage des connaissances contenues dans les ressources du Web et par conséquent leur réutilisation. Ces travaux rejoignent les travaux réalisés dans le domaine de **la gestion des connaissances** et en particulier dans le domaine des **mémoires d'entreprise ou d'organisation** visant à apporter une solution à la problématique de capitalisation du capital cognitif d'une organisation afin de faciliter son accès, son partage et sa réutilisation.

L'approche du **Web Sémantique d'Entreprise** fait l'analogie entre ces différents travaux en proposant de construire les mémoires d'entreprise en utilisant les techniques du Web Sémantique (**ontologies, annotations sémantiques et langages formels de représentation**). En ce qui concerne le problème que nous cherchons à résoudre : *la construction d'une mémoire d'expériences pour le domaine de l'analyse du transcriptome*, nous pensons que l'application de cette approche est très prometteuse.

Cependant, pour résoudre un problème de capitalisation et de gestion de connaissances, il faut tout d'abord identifier les sources de ces dernières et définir des techniques pour les extraire. Vu que le texte a toujours été considéré comme le moyen le plus sûr pour stocker et pérenniser les connaissances, une étude des techniques et des méthodes **d'extraction des connaissances à partir des textes** (en particulier les textes biologiques) nous a permis : (i) d'identifier le type d'informations pouvant être extraites automatiquement à partir des textes, (ii) d'étudier les différentes méthodes d'extraction génériques et d'explicitier plusieurs contraintes particulières du domaine étudié, et (iii) de faire les choix adaptés pour la résolution de notre problème.

Nous détaillons dans le chapitre suivant la problématique que nous cherchons à résoudre et présentons globalement la solution que nous proposons.

Chapitre 2 - Proposition de MEAT

1. Introduction

MEAT (**Mémoire d'Expériences pour l'Analyse du Transcriptome**) est un projet de construction d'une mémoire d'expérience, qui vise à faciliter la validation et l'interprétation des expériences puces à ADN.

L'objectif principal de cette mémoire est d'organiser les connaissances de ce domaine, qui proviennent de plusieurs sources hétérogènes (documents, base de données, connaissances humaines...) afin de faciliter leur partage et leur réutilisation. Nous visons donc à offrir aux biologistes un accès transparent et 'intelligent' à l'ensemble de ces connaissances.

MEAT peut être vu comme une méthodologie (i) d'intégration de différentes sources de connaissances, (ii) d'extraction et de gestion de connaissances, et (iii) d'assistance à la génération de nouvelles connaissances [Khelif et al., 2005, 2006].

Dans ce chapitre, nous présentons le contexte général de ce projet et nous évoquons les besoins qui ont motivé son élaboration. Nous donnons, ensuite, un aperçu global sur l'architecture de MEAT ainsi que sur ses différentes composantes.

2. Contexte général

2.1. Les expériences des puces à ADN

Le protocole des puces à ADN ou biopuces (*microarray*, *biochip*) est un protocole expérimental proposé en 1987. Il permet de mesurer et de visualiser très rapidement les différences d'expression entre les gènes et ceci à l'échelle d'un génome complet. Si la mise en oeuvre de la technique est assez compliquée, son principe est très simple.

A partir de la fixation à intervalles réguliers sur un support (de l'ordre de quelques cm²) de sondes composées de chaînes de nucléotides, l'expérience quantifie l'expression comparée des gènes ciblés entre une culture de cellules cible et une culture témoin par la lecture optique de 2 degrés de fluorescence (rouge et vert). Le rapport d'expression de chaque gène est alors acquis dans un tableau sous un format informatique adéquat après une analyse de l'image (i.e. de la coloration de chaque sonde). Ces rapports d'expressions permettent aux biologistes de déduire de nouvelles fonctions des gènes ou de nouvelles interactions entre eux.

La Figure 5 montre le principe d'une expérience puce à ADN²⁴.

²⁴ Plus de détails sur : <http://www.transcriptome.ens.fr/sgdb/presentation/principe.php>

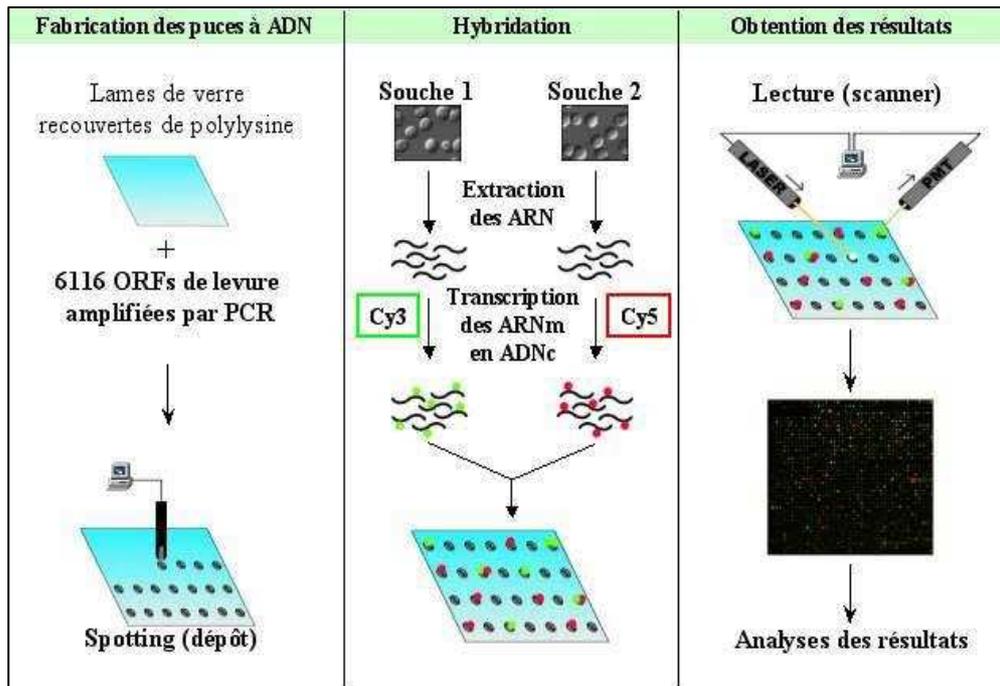


Figure 5 - Le principe d'une expérience puce à ADN

L'IPMC héberge la plate-forme biopuces de Sophia Antipolis, où les biologistes s'intéressent principalement aux applications pharmacologiques ou médicales de cette technologie, à savoir :

- la recherche de gènes exprimés lors de traitements pharmacologiques ou lors de situations physiologiques particulières (infection par des bactéries, inactivation spécifique d'un gène, etc.),
- le diagnostic clinique (mise en évidence ou classification de cancers permettant d'envisager des approches curatives mieux adaptées, en fonction des particularités génétiques du patient – pharmacogénomique),
- la recherche de profils d'expression associés de façon spécifique à des médicaments, afin d'accélérer la détermination des cibles pharmacologiques de nouvelles molécules et de mettre plus rapidement en évidence leur toxicité.

Pour MEAT, nous avons effectué un travail d'observation au sein de cette plate-forme afin de bien cerner les besoins et les difficultés de ce domaine.

2.2. Le processus de validation et d'interprétation d'une expérience

Chaque expérience de puces à ADN quantifie dans une ou plusieurs conditions entre 3.000 et 10.000 gènes. Le chercheur intervient à deux moments cruciaux : le choix des conditions d'expériences, qui est déterminée par le cadre général de son étude, et l'interprétation du résultat, ou plutôt *une* interprétation possible du résultat. Le résultat de l'expérience se présente

alors pour l'expert sous la forme d'un immense tableau de l'ordre de 100.000 entrées où il faut trouver des explications à un phénomène donné.

Dans notre travail, nous nous sommes intéressés au cycle de validation et d'interprétation des résultats d'une expérience puce à ADN. La Figure 6 présente les deux phases principales de ce cycle.

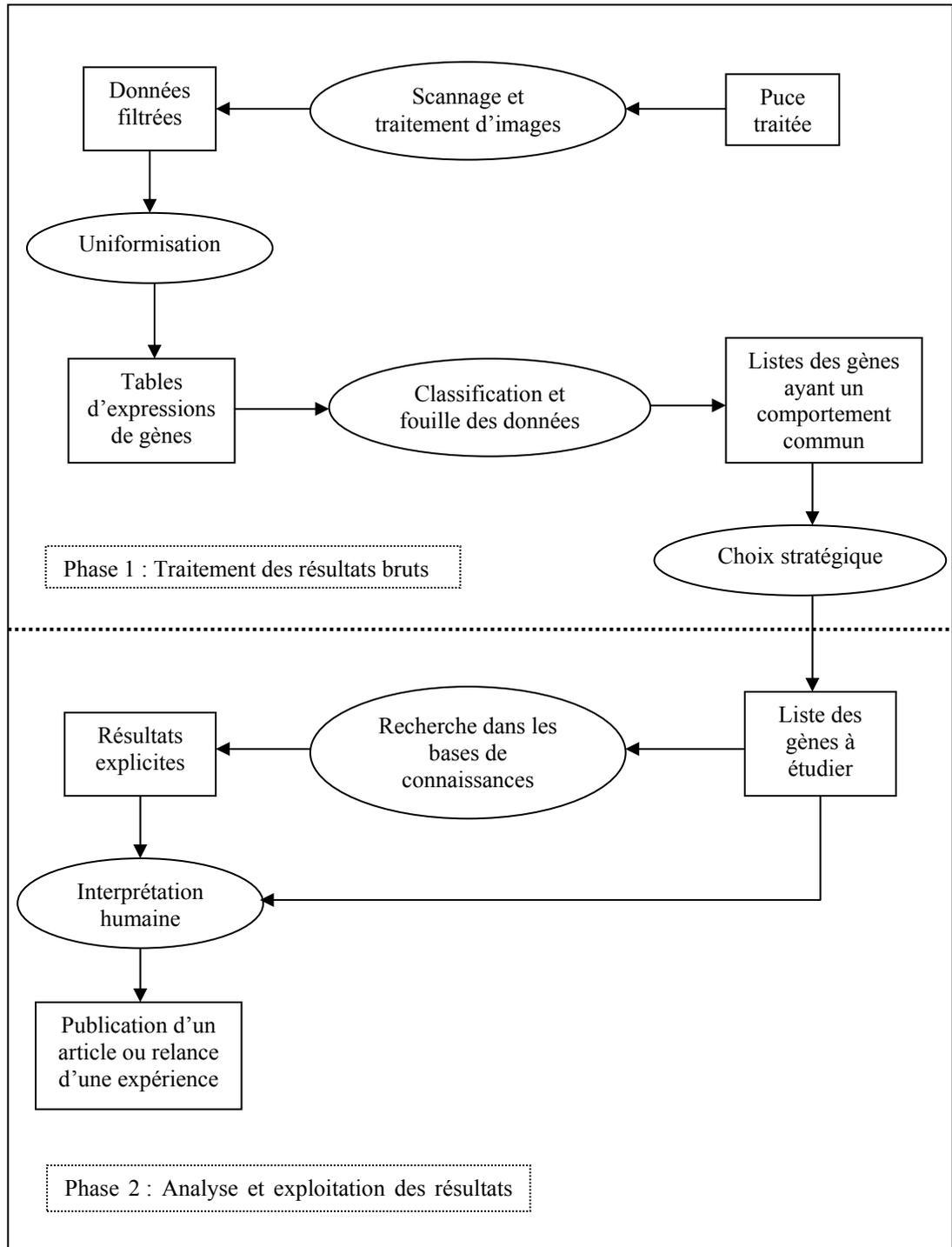


Figure 6 - Cycle de validation et d'interprétation d'une expérience puce à ADN

2.2.1. Traitement des résultats bruts

Le résultat initial d'une expérience puce à ADN est constitué de deux images montrant les degrés d'expression de chaque gène placé sur la puce. A l'aide de logiciels de traitement d'images, ces degrés d'expressions sont convertis en des valeurs allant de 0 à 1. Ensuite, des techniques de fouille de données sont appliquées sur cet ensemble de valeurs, afin de classer les gènes ayant des comportements communs lors de l'expérience. Ces classes sont présentées aux biologistes qui choisissent la liste des gènes stratégiques à étudier (suivant les hypothèses déjà fixées).

Dans notre travail, nous ne nous sommes pas intéressés à cette phase : (1) vu qu'il existe plusieurs outils (par exemple GeneCluster²⁵ et TreeView²⁶) déjà utilisés par les biologistes et considérés comme satisfaisants, et (2) les techniques utilisées ne concernent pas nos champs de recherche.

2.2.2. Analyse et exploitation des résultats

Une fois le sous-ensemble de gènes potentiellement intéressants sélectionné, le biologiste procède à une validation du déroulement global de l'expérience. Par exemple, il étudie des gènes dont le comportement est connu dans le cadre général d'une expérience et ce en se basant, soit sur ses connaissances implicites, soit sur des publications de référence. Si aucun comportement anormal n'est détecté, il peut alors passer à l'interprétation des nouveaux résultats.

Cette interprétation porte essentiellement sur des gènes apparentés (ayant la même fonction) ou dont l'expression lui paraît anormalement faible ou élevée. Il peut dès lors rechercher des analogies dans l'évolution de leurs expressions : s'ils réagissent de la même manière ou s'il n'y a pas un même facteur déclenchant les mêmes effets. Le raisonnement par analogie devient aussi un élément de preuve pour inférer la fonction d'un gène encore inconnue.

Nous pouvons ainsi résumer le cycle d'interprétation et de validation en une suite d'hypothèses, que le biologiste essaie de confirmer ou d'infirmer en se basant sur des sources de connaissances différentes et hétérogènes :

- Les bases documentaires : l'ensemble des publications en relation avec les gènes ou les phénomènes biologiques étudiés, ces documents peuvent être stockés localement ou peuvent provenir des portails en ligne (par exemple PubMed).
- Les bases d'expériences : les bases de données (locales ou en ligne) stockant les résultats des expériences antérieures traitant les mêmes gènes ou le même phénomène biologique.
- Les annotations sur les gènes : elles peuvent être formelles (les annotations de la Gene Ontology disponibles en ligne), ou informelles (faisant partie du

²⁵ <http://www.broad.mit.edu/cancer/software/genecluster2/gc2.html>

²⁶ <http://taxonomy.zoology.gla.ac.uk/rod/treeview.html>

patrimoine commun de biologistes ou apprises tout au long de leurs recherches).

Notre travail consiste à proposer une méthode de gestion et d'exploitation de ces connaissances afin d'assister les biologistes dans cette phase.

3. Capitalisation des connaissances pour l'analyse du transcriptome

Rappelons que l'enjeu majeur du projet MEAT est de capitaliser les connaissances du domaine de l'analyse du transcriptome sous la forme d'une mémoire d'expériences dans le but de proposer une aide méthodologique et logicielle aux biologistes dans la phase d'interprétation et de validation des résultats de leurs expériences. Dans cette partie, nous présentons notre méthode de construction de cette mémoire.

3.1. Que mémoriser?

Une grande partie des connaissances du domaine des biopuces est déjà représentée de manière informatique, en interne (exemple de la base d'expériences Mediante, cf. §3.2.1) ou en externe (base de données et bases bibliographique en ligne). Ces connaissances bien qu'accessibles via plusieurs moyens (recherche sur le web, interfaces dédiées...), manquent d'organisation et de visibilité, ce qui complexifie leur réutilisation et leur partage.

En effet, les publications du domaine sont nombreuses mais elles sont éparpillées sur le web et dans les bases documentaires internes. Le stockage des résultats d'expériences dans les bases de données (sans interprétation), fait souvent perdre le cheminement du chercheur, la méthodologie employée, et même les erreurs et les hypothèses qui se sont révélées infructueuses et dont un autre chercheur pourrait profiter, soit en évitant de commettre cette erreur, soit en découvrant une hypothèse à laquelle il n'a pas pensé et qui pourrait porter ses fruits dans le cadre de ses travaux. Quant aux notes, il s'avère qu'elles sont peu ou prou réutilisées, et trop liées au contexte de l'expérience. Nous pouvons penser que nous retombons sur un problème typique : la difficulté de formaliser les connaissances des experts ainsi que les connaissances du domaine.

La question « que mémoriser ? » sera donc centrale pour toute une première phase du projet MEAT. En fait, il faut voir ce que nous souhaitons désigner par « mémoire d'expérience » ... Le terme lui-même caractérise la mémoire d'une expérience vécue, dont l'homme peut se souvenir des années plus tard. On peut aussi la voir comme une spécification du terme « mémoire de projet »²⁷, qui vient lui-même d'une spécification du concept de mémoire d'entreprise.

²⁷ « Une mémoire des connaissances et des informations acquises et produites au cours de la réalisation des projets » [Matta et al., 1999]

La mémoire d'expérience serait alors une mémoire de projet dans le cadre d'un ou de plusieurs protocoles expérimentaux. Elle doit alors comporter la définition, c'est-à-dire le contexte scientifique de l'expérience, ses objectifs (c'est à dire ce qu'elle vise à mettre en évidence), la description des protocoles, le déroulement opératoire de l'expérience, les résultats et l'interprétation par le scientifique, qui peut elle-même devenir multiple si on prend en compte des points de vue différents ou des méthodologies, des hypothèses et des conclusions différentes. Cette mémoire doit aussi contenir des informations ou des liens sur des sources de connaissances externes qui sont considérées pertinentes pour le domaine (par exemple les articles des journaux scientifiques).

Dans le cadre du projet MEAT, cette mémoire d'expériences doit fournir :

- Une vue sur les expériences connexes : essayer d'identifier des relations entre les expériences (bases de données locales ou en ligne) et de découvrir de nouvelles pistes à explorer.
- Une aide à la validation des résultats expérimentaux : en proposant un accès 'intelligent' aux articles traitant des phénomènes étudiés afin de trouver des informations qui argumentent, confirment ou infirment les hypothèses de départ.
- Une aide à l'interprétation des résultats : en proposant des méthodes et des outils d'aide au raisonnement sur les connaissances contenues dans la mémoire afin d'identifier de nouvelles relations entre gènes et/ou les interactions pouvant exister entre eux, avec des composants cellulaires ou des processus biologiques.

3.2. Inventaire de l'existant

Après avoir défini notre mémoire d'expérience, nous nous sommes intéressés de plus près aux différentes sources de connaissances du domaine des biopuces, qui constitueront cette mémoire. Dans ce travail, nous nous intéressons particulièrement aux ressources de la plateforme biopuces de Sophia Antipolis, basée à l'IPMC.

3.2.1. MEDIANTE

Le projet MEDIANTE²⁸ a été développé dans le but de faciliter et d'améliorer la recherche de sondes (séquence d'ADN synthétique) utilisables lors d'analyses du transcriptome. Cette interface permet le suivi des commandes et le renseignement des données d'hybridation des lames. Elle permet en outre de visualiser directement les spots des lames hybridées et offre la possibilité de commencer une analyse des données transcriptomes. Ce système d'informations permet aussi la gestion des projets d'expériences au sein de la plateforme en partant de la conception des puces jusqu'au stockage des résultats.

Le cœur de MEDIANTE est une base de données qui se trouve alimentée tout au long des expériences, de la mise au point d'une puce à son utilisation pour la quantification du

²⁸ <http://www.microarray.fr:8080/mediante/index?language=fr>

transcriptome d'une culture de cellules cible. Le modèle utilisé est conséquent : une cinquantaine de tables et des centaines de champs. Ce modèle a été conçu en suivant les recommandations du groupe MGED (Microarray Gene Expression Data) permettant ainsi le partage des informations dans la base de données en ayant la possibilité d'importer (resp. d'exporter) des données (resp. vers) d'autres bases.

3.2.2. Les bases documentaires

Ces bases peuvent être internes à l'équipe ou externes (en ligne). Les biologistes les utilisent essentiellement pour valider leurs méthodes de travail en vérifiant si les gènes qu'ils mettent en évidence ont déjà été cités dans la littérature correspondante. Ils pourraient aussi trouver de nouveaux gènes, non-cités ... mais la portée de leurs résultats dépendrait alors de la validité de leur méthode. Il y aura donc toujours besoin de chercher d'anciens résultats acceptés par la communauté avant de proposer une nouvelle approche.

Bases internes

Pour chaque projet, les biologistes constituent un corpus de documents concernant les gènes qui sont à priori intéressants pour l'expérimentation. Ces articles sont généralement sélectionnés à partir de journaux renommés (Annual Reviews, Physiological Reviews, Pharmacological Reviews...). L'hypothèse qu'ils posent consiste à dire que cette sélection garantit la qualité et l'authenticité du contenu. Cette sélection constitue une source d'information très importante pour les biologistes, elle leur permet (i) de valider les hypothèses posées pour une expérience, et (ii) de vérifier la cohérence des résultats obtenus.

Chaque expérience stockée dans MEDIANTE possède un lien vers un ensemble de documents internes.

Bases externes

La base biomédicale documentaire en ligne la plus utilisée est MedLine, devenue récemment PubMed²⁹. Elle est disponible sur le site du NCBI (*National Center for Biotechnology Information*, USA), site portail qui permet un accès aux banques de données génétiques les plus utilisées (PubMed, GenBank, CDD, Swiss-Prot, LocusLink).

La recherche PubMed se fait sur des termes biomédicaux précis, des auteurs ou des références d'articles. Un ajustement terminologique est réalisé au niveau des requêtes à travers le thesaurus biomédical MeSH (*Medical Subject Headings*). Cette base regroupe un nombre très important d'articles scientifiques dans le domaine biomédical et constitue une source riche pour la recherche d'informations.

3.2.3. Les analyses des experts

Elles concernent les hypothèses émises par un biologiste pour la réalisation d'une expérience et la méthode entreprise pour la validation et l'interprétation des résultats obtenus. Elles regroupent les compétences innées ou acquises, le savoir-faire et l'expérience de l'expert.

²⁹ <http://www.ncbi.nlm.nih.gov/Entrez>

Ces connaissances interviennent tout au long du processus de réalisation d'une expérience. Elles peuvent être représentées sous trois formes : (1) des champs informels dans la base MEDIANTE décrivant l'interprétation des résultats d'une expérience, (2) les cahiers d'expériences décrivant le processus de réalisation ainsi que les hypothèses d'une expérience, et (3) les points de vue des biologistes sur la connexion et la corrélation des expériences et des phénomènes biomédicaux.

Ces connaissances n'étant pas explicites sont généralement difficiles à formaliser.

3.3. Comment mémoriser?

Disposant de sources hétérogènes de connaissances, et ayant un besoin de proposer une mémoire organisationnelle facilitant l'exploitation et le partage de ces connaissances, nous avons décidé de matérialiser notre mémoire d'expérience via un web sémantique d'entreprise (WSE). Les principales composantes de ce WSE sont :

- Les ressources : les bases de données des expériences, les biologistes et les articles provenant de sources internes (base documentaire locale de chaque biologiste) ou de sources externes (articles provenant du web).
- Les ontologies : nous proposons MeatOnto, une ontologie modulaire formalisant les connaissances (biomédicales, techniques et générales) du domaine des biopuces.
- Les annotations sémantiques : elles décrivent les expériences stockées dans les bases de données (résultats, interprétations), les connaissances extraites des articles scientifiques et les autres ressources du domaine. Ces annotations peuvent être générées manuellement (en utilisant un éditeur d'annotations) ou automatiquement (grâce à techniques de fouille de textes).

La Figure 7 présente une première vue sur la structuration de cette mémoire.

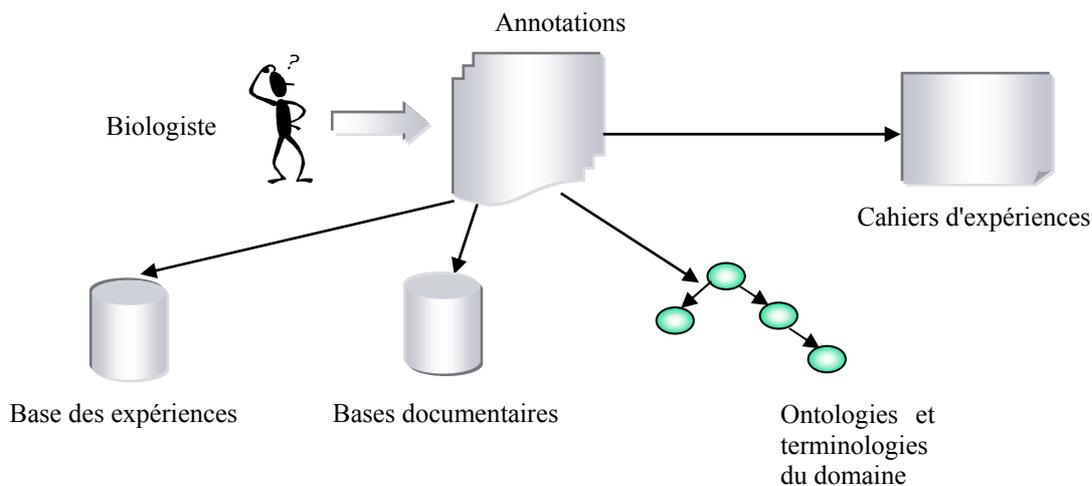


Figure 7 - Première vue sur la structure de la mémoire d'expérience

Dans ce qui suit, nous présentons l'architecture globale de la mémoire d'expériences MEAT, ainsi que ses différentes composantes.

4. Démarche générale

Après avoir présenté les besoins, la solution possible et les différentes sources de connaissances dont nous disposons, nous nous sommes penchés sur la manière de procéder pour proposer une architecture globale et cohérente de MEAT.

La Figure 8 montre les différentes étapes de notre démarche :

- Extraction des connaissances : proposer des moyens et des outils afin faciliter l'extraction de connaissances pertinentes du domaine à partir des différentes sources d'informations.
- Formalisation des connaissances : utiliser des modèles du domaine (i.e. les ontologies) pour représenter formellement les connaissances extraites et faciliter ainsi leur partage.
- Exploitation : fournir des moyens (recherche, navigation et raisonnement) pour une utilisation enrichissante des connaissances ainsi formalisées.
- Visualisation : faciliter la tâche d'interprétation des résultats d'une recherche d'informations en proposant des représentations à la fois pertinentes et conviviales (graphes, tableaux...).

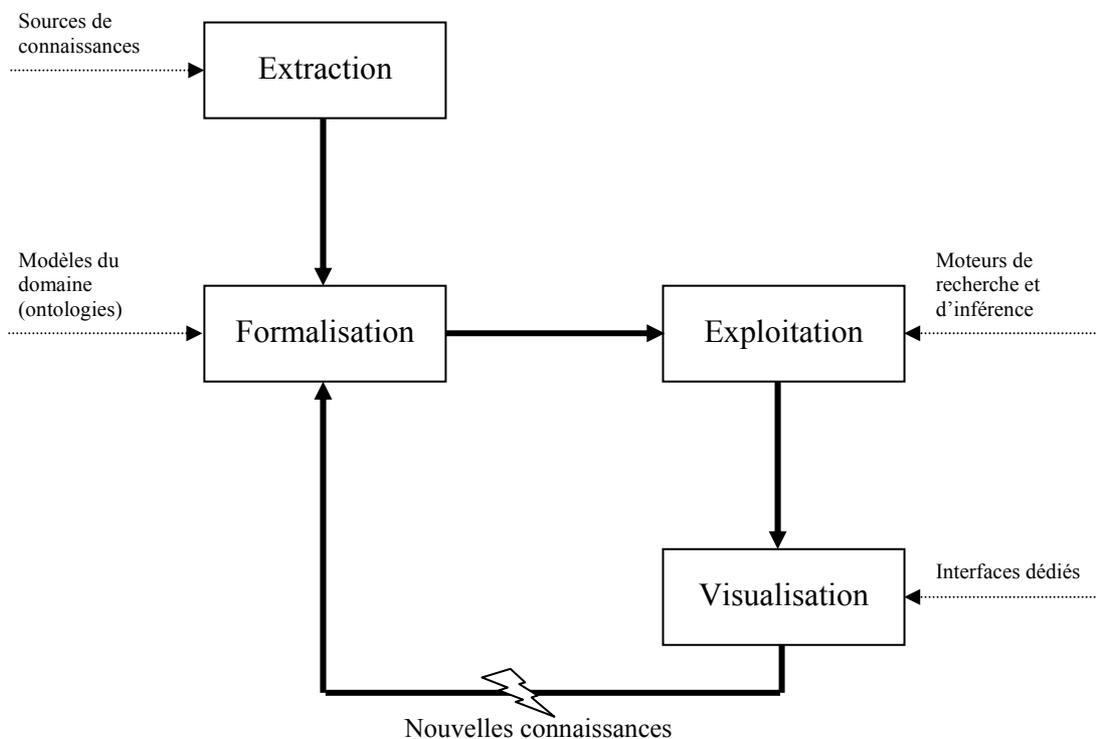


Figure 8 - Démarche générale de la construction de MEAT

4.1. Modélisation du domaine : définition des ontologies

Le cœur de Meat repose sur les ontologies, ces dernières offrent un cadre formel aux connaissances capitalisées, facilitant ainsi leur partage et leur réutilisation.

L'hétérogénéité des connaissances (connaissances techniques, connaissances scientifiques...) et la multiplicité des sources (base de données, documents, experts..) rend la tâche de définir un modèle unique du domaine difficile. Nous avons donc opté pour la structuration de ces connaissances sous la forme d'une ontologie modulaire composée de trois ontologies et que nous avons baptisé **MeatOnto** :

La première ontologie décrit les connaissances techniques nécessaires pour la réalisation d'une expérience puce à ADN, partant de l'élaboration d'une puce jusqu'au stockage des résultats.

La deuxième ontologie correspond à la modélisation du vocabulaire biomédical et aux connaissances scientifiques nécessaires pour la description et l'interprétation des résultats d'une expérience.

La troisième ontologie représente la structure des ressources et fait le lien avec les concepts des deux autres ontologies. Elle permet aussi de décrire des métadonnées sur les ressources (par exemple la source).

4.2. Extraction des connaissances à formaliser

Cette tâche consiste à proposer des méthodes et des techniques pour l'extraction de connaissances pertinentes permettant de générer des annotations sémantiques basées sur l'ontologie MeatOnto.

Nous distinguons trois types importants de connaissances provenant de sources différentes :

- Les connaissances contenues dans les bases d'expériences : nous proposons une phase d'extraction des données à partir des champs structurés des bases de données retenues (dans notre cas MEDIANTE) afin de les formaliser avec **MeatOnto**.
- Les connaissances présentes sous forme textuelle (articles scientifiques, notes ...) : nous avons proposé une méthode qui, à partir d'un texte, permet la génération d'une annotation structurée, basée sur MeatOnto et qui décrit le contenu sémantique de ce texte. Cette méthode a été implémentée dans le système **MeatAnnot**.
- Les points de vues et les connaissances des experts : nous offrons un éditeur basé sur MeatOnto permettant de collecter ces informations et de générer des annotations sémantiques. Cet éditeur est nommé **MeatEditor**.

4.3. Recherche d'informations guidée par les ontologies

Après avoir défini notre ontologie et élaboré des techniques pour la création d'annotations sémantiques, nous avons proposé un moyen d'accès, de navigation, d'exploitation et de visualisation de ces connaissances.

Un système nommé **MeatSearch** et basé sur le moteur de recherche sémantique Corese a été proposé : (i) il permet de faire de la recherche d'informations guidée par MeatOnto, (ii) il offre des interfaces dédiées pour l'interrogation et la visualisation des connaissances contenues dans la base d'annotations et (iii) il exploite le moteur d'inférence de Corese afin d'aider à faire des raisonnements et d'expliquer des comportements particuliers.

L'architecture générale de Meat, fruit de l'intégration des différents modules cités ci-dessus est présentée dans la Figure 9.

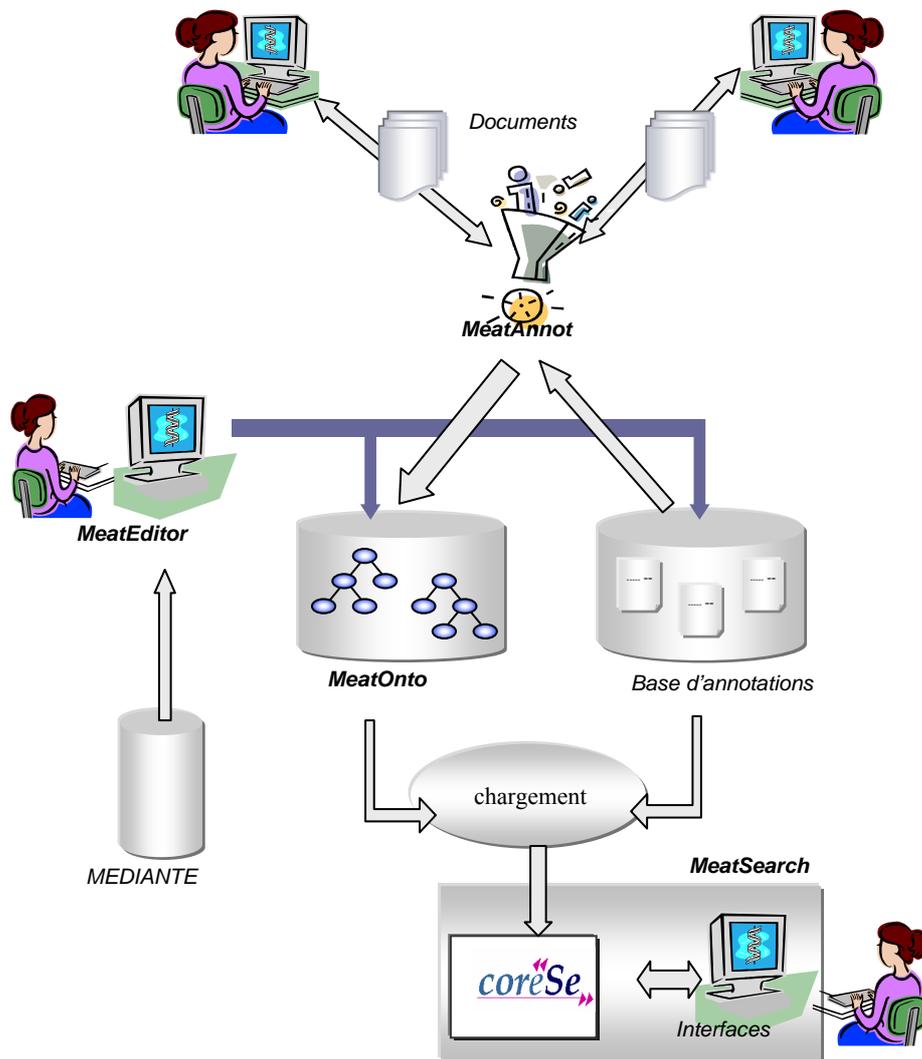


Figure 9 – Architecture de Meat

5. Conclusion

Nous avons proposé une démarche pour la construction d'une mémoire d'expériences à travers un Web Sémantique d'entreprise. Cette approche est axée sur la construction d'une ontologie décrivant les différentes connaissances du domaine et sur la l'extraction et la gestion des connaissances provenant de différentes sources (base de données, documents...). La recherche d'information guidée par les ontologies et exploitant les annotations sémantiques fournit une aide à la validation et à l'interprétation des résultats des expériences.

Une architecture du système implémentant cette mémoire a été proposée.

Les chapitres suivants présentent les différentes composantes de ce système : dans le troisième chapitre nous décrivons l'ontologie MeatOnto. Ensuite, dans le quatrième chapitre nous présentons une méthodologie pour la génération automatique d'annotations à partir des textes implémentée dans MeatAnnot. Enfin, le cinquième chapitre est consacré aux techniques utilisées dans MeatSearch pour l'exploitation des annotations déjà construites.

Dans la suite, nous commençons par décrire les choix et les techniques qui nous ont menés à construire l'ontologie MeatOnto.

Chapitre 3 - Choix et construction des ontologies : MeatOnto

1. Introduction

Ayant adopté une approche fondée sur les techniques du Web Sémantique pour la capitalisation et la gestion des connaissances dans le domaine des puces à ADN, notre première tâche consiste alors à définir une ontologie couvrant ce domaine. Cette ontologie que nous avons baptisée MeatOnto servira à guider (i) la génération d'annotations sémantiques sur les différentes ressources du domaine et (ii) la recherche d'information dans la mémoire d'expériences.

La construction d'une telle ontologie peut se faire selon plusieurs techniques, parmi lesquelles nous pouvons citer :

- Manuellement : en identifiant et en formalisant les connaissances du domaine, ce qui nécessite une certaine maîtrise du domaine et des interactions avec les experts. Cette technique a été appliquée par exemple dans [Gandon, 2002].
- Automatiquement : en extrayant automatiquement les informations jugées pertinentes pour la construction de l'ontologie à partir des différentes sources du domaine étudié (bases de données, notes papiers, corpus techniques...). Cette technique a été explorée dans [Golebiowska et al., 2001].
- Par réutilisation : en adaptant et en complétant des ontologies déjà existantes couvrant soit la totalité, soit une partie du domaine étudié. Dans notre travail nous avons choisi d'utiliser cette technique.

Dans ce qui suit, nous détaillons la construction de l'ontologie MeatOnto tout en justifiant les choix adoptés. L'exploitation de cette ontologie sera détaillée par des exemples d'utilisation dans le deux chapitres suivants : génération et exploitation des annotations.

2. Description de l'ontologie MeatOnto

L'objectif principal de l'ontologie MeatOnto est de couvrir et de formaliser les connaissances sur les puces à ADN. Ce domaine étant au carrefour de plusieurs domaines de recherche (biologie moléculaire, médecine, plateformes d'expérimentations), ces connaissances présentent un certain degré d'hétérogénéité. En effet, en l'étudiant plus précisément, nous pouvons distinguer deux grandes catégories de connaissances nécessaires pour la description de ce domaine :

- Connaissances « techniques » : concernant le processus de l'expérimentation allant de la préparation d'une puce jusqu'à la récupération des résultats;
- Connaissances biomédicales : nécessaires pour décrire les phénomènes biologiques étudiés et les gènes mis en jeu lors d'une expérience.

L'ontologie MeatOnto doit donc couvrir ces deux classes auxquelles s'ajoutent d'autres informations ayant pour but de faciliter l'exploitation de ces connaissances à des fins

d'annotation et de navigation. Ce constat nous a mené à opter pour une ontologie multi-composantes où chaque composante est dédiée à la description d'une catégorie précise de connaissances :

- Une ontologie pour les expériences : permettant de décrire l'aspect technique (plate-forme, technologie utilisée...) et d'autres informations générales concernant une expérience puce à ADN (Voir §3).
- Une ontologie biomédicale : couvrant le vocabulaire utilisé dans la communauté biomédicale (Voir §4).
- Une ontologie « fonctionnelle » : décrivant (i) des informations générales concernant les ressources à annoter (documents, expériences..), et (ii) le contexte de la création d'une annotation (source, thème général...). Cette ontologie permet le pilotage du processus de création d'une annotation en faisant le lien entre les différents concepts des autres ontologies (Voir §5).

3. Une ontologie pour les expériences des puces à ADN : MGED

3.1. Présentation générale

Le groupe MGED (Microarray Gene Expression Data) a été créé afin de proposer des méthodes pour la gestion et la représentation des expériences dans le domaine de la génomique avec un zoom sur les expériences des puces à ADN.

Ce groupe a proposé une ontologie spécifique au domaine des expériences des puces à ADN, mais aussi adaptable à d'autres types d'expériences. Le but de cette ontologie est (i) de servir de modèle pour annoter les expériences et (ii) de faciliter le partage des résultats par les différentes équipes.

Cette ontologie comprend deux grandes catégories de concepts :

- la première concerne les concepts décrivant l'aspect technique des expériences à savoir les protocoles, les matériels et les logiciels utilisés.
- la deuxième concerne l'aspect biologique de l'expérience comme par exemple les traitements utilisés et les organismes étudiés.

D'autres concepts décrivant des facteurs pouvant influencer les résultats d'une expérience comme l'âge, le sexe, etc. ont été rajoutés. Cette ontologie a été construite en identifiant les besoins des biologistes et en dialoguant avec ces derniers afin de fournir un vocabulaire compréhensible et utilisable par toute la communauté des biopuces.

Le Tableau 1 montre quelques exemples de concepts de l'ontologie MGED.

Concept	Définition	Type
Protocol	Description des différentes étapes de l'expérimentation	Aspect technique
SoftwareVariation	Les effets des logiciels utilisés sur les résultats obtenus	
IndividualGeneticCharacteristics	Caractéristiques génétiques de l'organisme étudié	Aspect biologique
Host	Les organismes utilisés pour la culture des cellules cibles	
Sex	Utilisé pour différencier les sexes des organismes	Autres
Age	La période de temps écoulée depuis un point identifiable dans le cycle de vie d'un organisme	

Tableau 1 - Quelques concepts de l'ontologie MGED

La version actuelle de l'ontologie (MGED 1.6) comprend 114 concepts et 82 relations et est disponible dans différents formats, y compris en RDFS et en OWL. Elle est utilisée par une grande partie de la communauté des biopuces, soit directement pour annoter les expériences en utilisant les concepts et les relations, soit indirectement en s'inspirant de sa structure pour définir des schémas de base de données pour le stockage de leurs expériences (le cas de MEDIANTE).

3.2. Correspondance entre MGED et MEDIANTE

Dans le projet Meat, nous avons adopté l'ontologie MGED pour la génération des annotations sur les expériences des puces à ADN. Le but de ces annotations, une fois intégrées dans la base de connaissances, est de permettre aux biologistes en utilisant un moteur de recherche adéquat (i) d'avoir une vue globale sur l'ensemble des expériences déjà réalisées, (ii) de déduire des corrélations entre les expériences, et (iii) d'avoir des idées pour la planification de nouvelles expériences (Détails dans le Chapitre 5).

Par ailleurs, et étant donné que les expériences réalisées au sein de l'IPMC sont stockées dans la base de données MEDIANTE, la génération des annotations sur les expériences consiste

à faire la correspondance entre les champs et les tables de cette base et les concepts ainsi que les relations de l'ontologie MGED. Cette correspondance a été réalisée manuellement à l'aide de l'ingénieur responsable de la base de données MEDIANTE au sein de l'IPMC. Lors de cette phase, nous avons réussi (i) à identifier les tables qui correspondent aux concepts génériques qui nous intéressent, et (ii) à identifier dans ces tables, les champs dont les valeurs vont être considérées comme des instances des concepts spécifiques utilisés pour annoter une expérience.

Le Tableau 2 montre quelques concepts de l'ontologie MGED ainsi que la description des tables qui leur correspondent dans la base de données.

Concept de MGED	Description de la table correspondante dans MEDIANTE
ExperimentDesignType	Description générale de l'expérience (titre, but...)
BibliographicReference	Description des publications liées à l'expérience
BiologicalFactorCategory	Description des entités biologiques liées à l'expérience
Contact	Informations sur les expérimentateurs

Tableau 2 - Correspondance entre les concepts de MGED et les tables de MEDIANTE

4. Une ontologie pour le domaine biomédical : UMLS

4.1. Présentation générale

Ce projet élaboré par la NLM (National Library of Medicine de Bethesda), propose depuis 1986 d'unifier la totalité du vocabulaire biomédical afin de mettre au point un langage médical unifié [Humphreys et Lindberg, 1993].

4.1.1. Le métathésaurus

Le mot méta indique ici le fait que cette base de connaissances intègre une centaine de thésaurus et de terminologies biomédicaux, parmi lesquels nous citons MESH³⁰, CIM-10³¹, SNOMED...

A ce jour, le métathésaurus de UMLS contient les définitions de plus de 2.500.000 termes. Certains sont des termes préférentiels, les autres sont des variantes lexicales, des synonymes et des abréviations. Les termes préférentiels représentent des concepts biomédicaux bien identifiés,

³⁰ <http://www.nlm.nih.gov/mesh/>

³¹ <http://www.icd10.ch/index.asp>

chacun étant considéré comme un ensemble de termes synonymes (l'équivalent des synsets dans WordNet³²). Dans la version 2005, nous répertorions plus de 900.000 concepts.

Chaque concept possède un identificateur et est lié à un type sémantique du réseau sémantique de UMLS. Des relations entre concepts existent aussi et peuvent, soit provenir des différentes sources constituant le métathésaurus, soit être créées spécifiquement.

Le Tableau 3 présente le concept 'Fibrillation' ainsi que l'ensemble des termes le constituant.

Concept	CUI	Termes associés	Type sémantique
Fibrillation	C0232197	Fibrillation Cardiac fibrillation Fibrillated Heart	Disease or Syndrome

Tableau 3 - Exemple d'un concept du métathésaurus de UMLS

Récemment la Gene Ontology a été intégrée dans UMLS [Lomax et McCray, 2004]. Les termes de GO ont été, soit alignés avec des concepts existants, soit liés manuellement aux autres concepts de UMLS. Cette intégration offre un important lien entre les termes biomédicaux et les termes de la génomique qui sont très utiles dans un domaine tel que le domaine des expériences des puces à ADN.

4.1.2. Le réseau sémantique

La version actuelle³³ du réseau sémantique (RS) de UMLS contient 134 types sémantiques et 54 relations. Ce réseau est composé de deux hiérarchies, une pour les entités et une pour les événements, et chaque type sémantique est lié à son parent par un lien de spécialisation is-a.

La Figure 10 représente le type sémantique «Human» issue de six spécialisations du type sémantique racine «Entity».

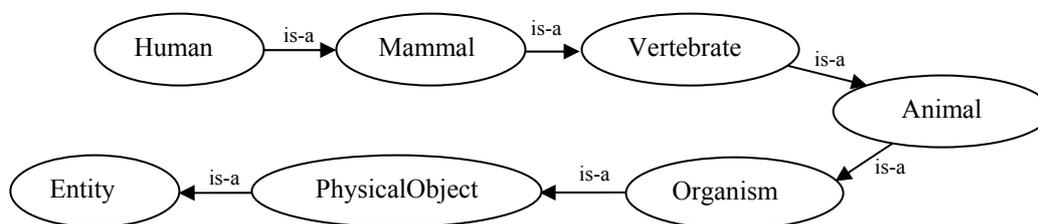


Figure 10 - Représentation du type sémantique « Human » dans le RS de UMLS

³² <http://ordnet.princeton.edu/>

³³ Version 2005

Le réseau sémantique contient aussi une hiérarchie de relations permettant de relier les types sémantiques entre eux. La Figure 11 montre une portion du RS illustrant quelques relations.

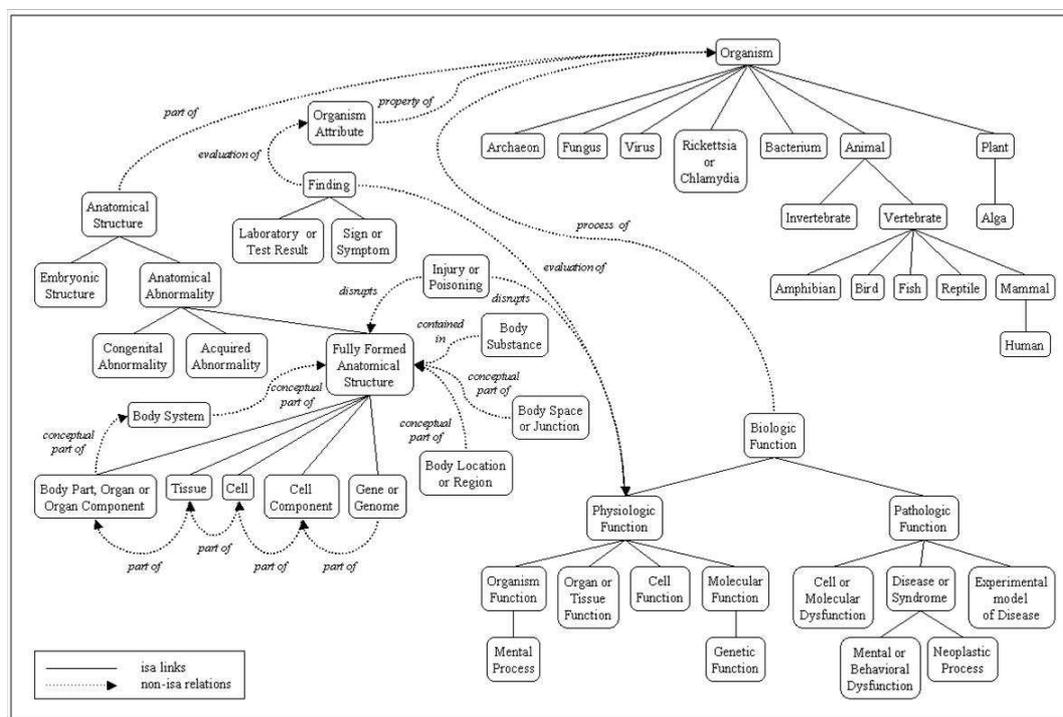


Figure 11 - Une portion du réseau sémantique de UMLS

4.2. UMLS : Une ontologie ?

Comme nous pouvons le constater à partir la description ci-dessus, le but initial de UMLS, que l'on pourrait traduire par « système d'unification de la langue médicale », était essentiellement d'unifier plusieurs produits terminologiques biomédicaux afin de fournir un accès facile et partagé à ces ressources pour la communauté biomédicale.

Cependant, et vu que ce système intègre plusieurs ontologies existantes (GO, Snomed...) et possède un réseau sémantique organisant ces ressources, nous avons procédé à une étude ontologique afin de décider si nous pouvons le considérer comme une ontologie ou non.

4.2.1. Etude ontologique du métathésaurus

Le métathésaurus est le produit d'une superposition de différentes sources terminologiques. Cette construction ascendante a généré quelques problèmes liés à la structuration hiérarchique des termes au sein du métathésaurus.

En effet, du fait que les différentes ressources constituant le métathésaurus ont été élaborées séparément et par différentes équipes, un premier problème qui est apparu était celui de l'alignement des termes similaires. Ce problème d'alignement est dû essentiellement à (i) l'ambiguïté des termes utilisés (deux termes peuvent être synonymes dans une terminologie alors qu'ils sont complètement différents dans d'autres) et (ii) l'emploi spécifique attribué aux

termes ('topographic regions' dans la SNOMED désigne un concept plus générique que 'body system', alors qu'ils sont considérés comme synonymes dans le métathésaurus [Zweigenbaum, 2004]).

[Mcray et Nelson 1995], évoquent d'autres problèmes liés à la hiérarchisation des concepts obtenue lors de la construction du métathésaurus. Les concepts n'ayant pas la même portée d'une terminologie à une autre, des cycles dans la hiérarchie sont apparus (un concept se retrouvait à la fois ancêtre et descendant d'un autre concept).

Comme le montrent la Figure 12 et la Figure 13 citées dans [Bodenreider 2001], ces cycles peuvent être directs ou indirects.

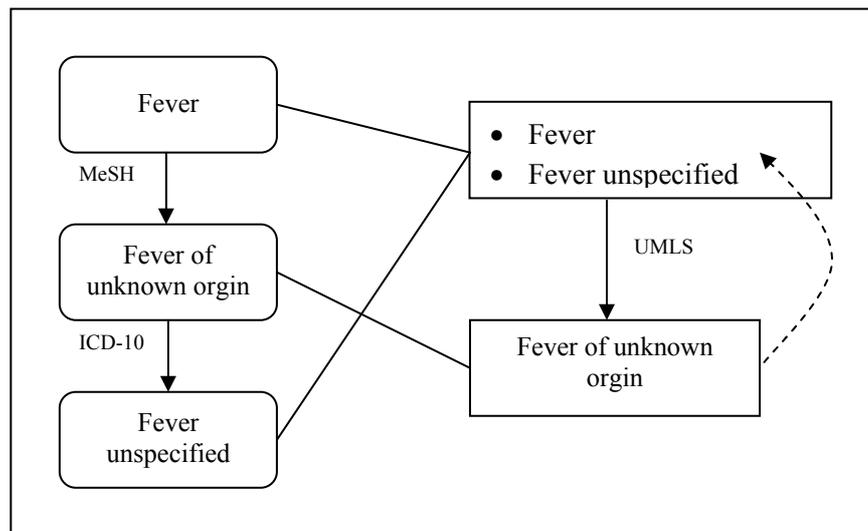


Figure 12 - Cycle direct dans le métathésaurus de UMLS [Bodenreider, 2001]

Dans cet exemple, nous pouvons voir que dans Mesh, le concept 'Fever' est parent de 'Fever of unknown source' qui est lui-même parent de 'Fever unspecified' dans ICD-10. Lors de l'intégration dans UMLS, les concepts 'Fever' et 'Fever unspecified' ont été regroupés dans un même concept qui est devenu à la fois père et fils du concept 'Fever of unknown source'.

Dans le cas précédent, le cycle est apparu directement, mais ce genre de problème peut surgir à n'importe quelle profondeur de la hiérarchie (comme le montre la figure suivante).

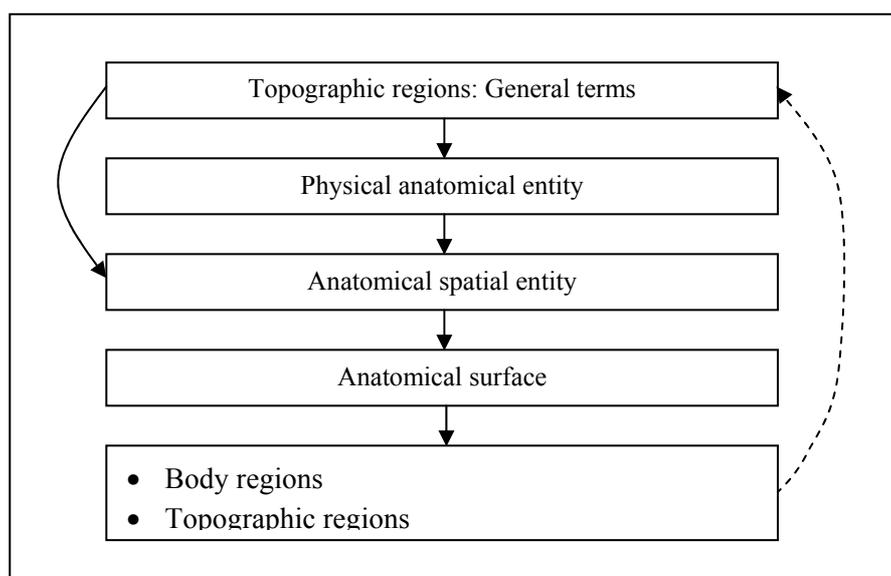


Figure 13 - Cycle indirect dans le métathésaurus UMLS [Bodenreider, 2001]

Le regroupement des termes sous la forme de concepts a généré aussi des cycles dus à des relations réflexives : c'est le cas où un père et un fils sont regroupés dans le même concept, ce qui oblige ce dernier de pointer sur lui-même.

[Bodenreider, 2001] fait état d'au moins 1.800 cycles directs, 120 cycles indirects et 13.000 cycles dû à des relations réflexives.

Ce constat nous a laissé perplexes sur la représentation du métathésaurus sous une forme ontologique, vu que le traitement des problèmes cités ci-dessus est pratiquement impossible à automatiser et nécessite la présence de plusieurs experts du domaine.

4.2.2. Etude ontologique du réseau sémantique

Contrairement au métathésaurus, la construction du réseau sémantique a été réalisée de façon descendante. En effet, le réseau sémantique a été développé au fur et à mesure, afin de proposer un cadre conceptuel de haut niveau pour tous les concepts du métathésaurus.

Ce réseau sémantique est présenté par [McCray, 2003] comme une ontologie générique fournissant une modélisation consistante et cohérente du domaine biomédical.

Cependant, pour vérifier l'aspect ontologique de cette modélisation, nous avons décidé d'étudier de plus près la structure et la sémantique de ce réseau. Pour ce faire, nous nous sommes reposés sur trois critères de validation de la construction d'ontologies proposés par [Bachimont, 2000] :

- l'engagement sémantique : fixant le sens linguistique des concepts de l'ontologie ;
- l'engagement ontologique : fixant le sens formel de ces concepts;
- l'engagement computationnel : déterminant l'exploitation effective de ces concepts.

Engagement sémantique et ontologique du réseau sémantique

Le réseau sémantique est constitué de 134 types sémantiques qui sont considérés comme des concepts de haut niveau décrivant le domaine biomédical. Parmi ces concepts, nous trouvons (i) des concepts généraux (Entity, Physical Object...) et (ii) des concepts propres au domaine (Sign or Symptom, Gene or genome...).

Chaque concept possède une définition lui donnant un sens linguistique. Par exemple, la définition du concept 'Mammal' est 'Un vertébré ayant une température constante et caractérisé par la présence de poils et de glandes mammaires'. Ces définitions permettent ainsi de restreindre les interprétations possibles des termes (i.e. les termes du métathésaurus).

D'autre part, les types sémantiques sont formalisés dans un arbre strict (pas de pères multiples) par la relation de spécialisation is-a. Cet arbre permet de couvrir 'tout' le domaine biomédical puisque chaque terme du métathésaurus possède un ou plusieurs types sémantiques. Cette formalisation permet ainsi de donner un aspect ontologique à la hiérarchie des types sémantiques.

Engagement computationnel du réseau sémantique

Outre la relation is-a, le réseau sémantique de UMLS comporte une hiérarchie de relations sémantiques : des relations physiques (part of, contains...), des relations spatiales (location_of, surrounds...), des relations temporelles (precedes, co-occurs_with...), des relations conceptuelles (measures, diagnoses...) et des relations fonctionnelles (affects, manages...). Chacune des relations possède une signature définissant les types sémantiques qu'elle peut relier.

Ces relations renforcent la structure du réseau sémantique et définissent les opérations possibles sur l'ensemble des types sémantiques.

Un exemple de relation est présenté dans Figure 14.

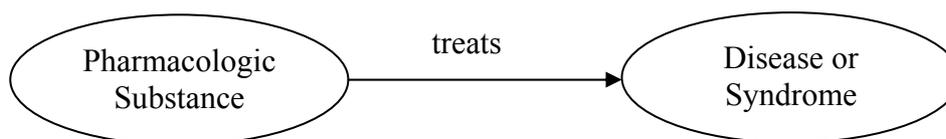


Figure 14 - Exemple d'une relation dans le réseau sémantique

Cette assertion montre une interaction possible entre les deux types sémantiques 'Pharmacologic Substance' et 'Disease or Syndrome' caractérisée par le lien 'treats'.

La richesse de structuration du réseau sémantique et sa couverture quasi-totale du domaine étudié nous a encouragé à le considérer comme une ontologie générale du domaine biomédical. Les termes du métathésaurus peuvent être désormais considérés comme des instances possibles des concepts de cette ontologie.

4.3. Enrichissement de UMLS

Comme nous l'avons indiqué dans le paragraphe §4.2.2, le réseau sémantique de UMLS comporte une hiérarchie de relations sémantiques (54 relations dans la version 2005) composée de cinq familles :

- Les relations physiques : reliant les termes ayant des caractéristiques physiques communes (exemple : *branch_of*);
- Les relations spatiales : reliant les termes en fonction de leur localisation (exemple : *location_of*);
- Les relations fonctionnelles : exprimant une fonction ou une activité reliant les termes (exemple : *interacts_with*);
- Les relations temporelles : reliant les termes dans le temps (exemple : *precedes*);
- Les relations conceptuelles : reliant les termes selon un certain concept abstrait, une pensée ou une idée (exemple : *measures*).

Après une étude approfondie de ces différentes familles et des discussions avec nos collègues biologistes, il est apparu (i) que pour annoter un phénomène biologique lié à une expérience, les deux familles qui les intéressent essentiellement sont : les relations conceptuelles et les relations fonctionnelles (au nombre de 35, ces relations représentent ainsi 65% de l'ensemble des relations), et (ii) que bien que ces relations couvrent la totalité des liens pouvant exister entre les concepts du réseau sémantique, quelques-unes sont trop génériques et peuvent avoir un effet négatif sur le niveau de précision et de finesse d'une annotation.

Prenons l'exemple de la relation fonctionnelle '*affects*' qui est définie comme étant la production d'un effet direct par une entité biologique sur une autre, cet effet pouvant être le résultat de l'une des actions suivantes : {*has a role in, alters, influences, predisposes, catalyzes, stimulates, regulates, depresses, impedes, enhances, contributes to*}. Cette définition préconise que toutes ces actions peuvent être considérées comme des 'synonymes' et doivent être annotées par la relation '*affects*', ce qui peut générer du bruit dans les résultats d'une recherche d'informations sur ces annotations. En effet, un biologiste cherchant à trouver toutes les entités biologiques ayant été stimulées par un gène particulier aura en plus des entités qui l'intéressent d'autres qui ont été altérées, catalysées ... par ce même gène.

Notre but étant d'utiliser cette ontologie pour annoter des ressources et faciliter la tâche de recherche d'informations, nous avons décidé d'enrichir le réseau sémantique par des relations plus spécifiques afin d'avoir des annotations plus précises. Pour ce faire nous avons procédé en deux étapes :

Etape 1 : Exploitation des définitions des relations

Dans cette étape, nous nous sommes basés sur les définitions attribuées à chaque relation du réseau sémantique. Comme le montre la Figure 15, ces définitions comportent un ensemble de termes pouvant avoir indiqué un sens plus précis de la relation concernée.

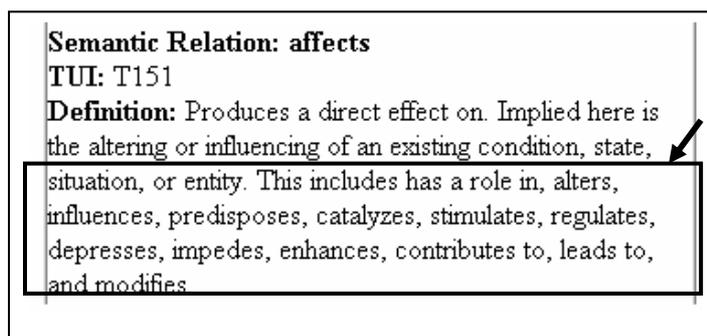


Figure 15 - Définition de la relation 'affects'

Nous avons constaté que ces termes ne peuvent pas être tous considérés comme des synonymes de la relation concernée et certains termes doivent être plutôt considérés comme correspondant à des relations sémantiques implicites plus précises. Cette hypothèse nous a donc permis de spécialiser les relations de UMLS par de nouvelles relations. La Figure 16 montre le résultat de cette spécialisation sur la relation 'affects'.

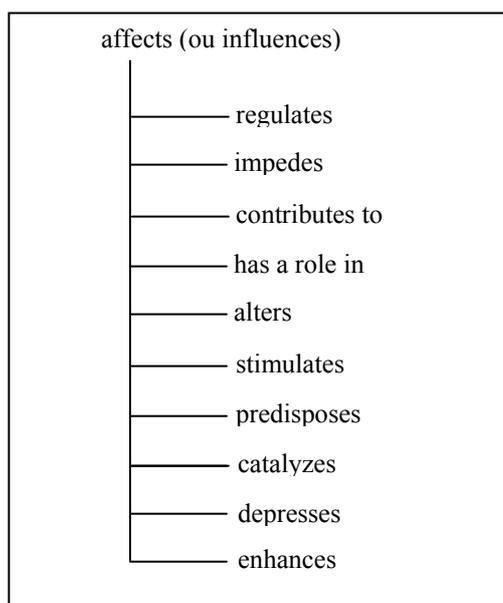


Figure 16 - Spécialisation de la relation 'affects'

L'exemple ci-dessus montre aussi que le terme 'influences' a été considéré comme un synonyme de la relation 'affects'.

Etape 2 : prise en compte des suggestions des biologistes

Lors de nos discussions, les biologistes nous ont proposé quelques relations spécifiques à leur domaine que nous n'avons pu détecter lors de la première étape car elles n'apparaissent pas dans les listes de termes caractérisant les relations de UMLS. Cette étape nous a permis par

exemple d'ajouter la relation '*activates*' et la relation '*inhibits*' comme étant deux spécialisations de la relation '*performs*'.

Au total, nous avons réussi à rajouter 24 nouvelles relations au réseau sémantique de UMLS. Ces nouvelles relations ont la même signature que la relation à laquelle elles sont rattachées.

4.4. Formalisation du réseau sémantique

Cette formalisation a été faite sur deux temps. Nous avons commencé par traduire la hiérarchie des types sémantiques en une hiérarchie de concepts, ensuite nous avons traduit la hiérarchie des relations en une hiérarchie de propriétés en définissant la signature de chacune.

4.4.1. Formalisation de la hiérarchie des concepts

À cette étape, nous avons appliqué un script que nous avons développé sur la version textuelle³⁴ du réseau sémantique. Chaque type sémantique a été traduit en une classe RDFS (`rdfs:Class`) et les liens de spécialisation entre les types sémantiques (*is-a*) ont été traduits en une propriété de subsomption de RDFS (`rdfs:subClassOf`).

La Figure 17 montre une portion de la hiérarchie des concepts traduite en RDFS.

³⁴ Disponible sur <http://semanticnetwork.nlm.nih.gov/Download/UnitFile/SU>

```

<rdfs:Class rdf:ID="Entity">
  <rdfs:subClassOf rdf:resource="#UMLS"/>
  <rdfs:comment>
    A broad type for grouping physical and conceptual entities.
  </rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Event">
  <rdfs:subClassOf rdf:resource="#UMLS"/>
  <rdfs:comment>
    An object perceptible to the sense of vision or touch
  </rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Physical_Object">
  <rdfs:subClassOf rdf:resource="#Entity"/>
  <rdfs:comment>
    A broad type for grouping activities, processes and states
  </rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Organism">
  <rdfs:subClassOf rdf:resource="#Physical_Object"/>
  <rdfs:comment>
    Generally, a living individual, including all plants and animals
  </rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Animal">
  <rdfs:subClassOf rdf:resource="#Organism"/>
  <rdfs:comment>
    An organism with eukaryotic cells, and lacking stiff cell walls, plastids and
    photosynthetic pigments
  </rdfs:comment>
</rdfs:Class>
.....
<rdfs:Class rdf:ID="Cell">
  <rdfs:subClassOf rdf:resource="#Fully_formed_anatomical_structure"/>
  <rdfs:comment>
    The fundamental structural and functional unit of living organisms
  </rdfs:comment>
</rdfs:Class>
<rdfs:Class rdf:ID="Gene_or_genome">
  <rdfs:subClassOf rdf:resource="#Fully_formed_anatomical_structure"/>
  <rdfs:comment>
    A specific sequence, or in the case of the genome the complete sequence,
    of nucleotides along a molecule of DNA or RNA (in the case of some
    viruses) which represent the functional units of heredity
  </rdfs:comment>
</rdfs:Class>

```

Figure 17 - Exemple de classes RDFS de UMLS

4.4.2. Formalisation de la hiérarchie des relations

Dans cette étape, nous avons utilisé un script amélioré par rapport à celui utilisé lors de la première phase. Ce script permet (i) la traduction de la hiérarchie des relations sémantiques en une hiérarchie de propriétés RDFS (`rdfs:Property`) en remplaçant le lien `is-a` en un lien d'héritage RDFS (`rdfs:subPropertyOf`) et (ii) la définition des signatures des propriétés qui

représentent le domaine d'application de ces dernières (rdfs:range et rdfs:domain). Cette signature autorise (mais n'implique pas) le fait qu'une relation puisse exister entre deux concepts.

La Figure 18 montre la définition de la propriété 'process_of'.

```
<rdfs:Property rdf:ID= process_of >
  <rdfs:subPropertyOf df:resource="#occurs_in">
  <rdfs:domain rdf:resource= #BiologicFunction >
  <rdfs:range rdf:resource= #Organism >
  <rdfs:comment>Action, function, or state of </rdfs:comment>
</rdfs:Property>
```

Figure 18 - Définition de la propriété 'process_of' en RDFS

Malgré nos efforts pour automatiser cette tâche, des complications sont apparues lors de la définition des relations. Ces complications sont dues à deux types de problèmes :

(1) Des relations polymorphes :

Ce sont des relations ayant plusieurs concepts sources et plusieurs concepts cibles. Par exemple, la signature de la relation 'contains' est la suivante :

```
contains(Body_Space_Or_Junction, Body_Part_Organ_Or_Organ_Component)
contains(Body_Space_Or_Junction, Body_Substance)
contains(Body_Space_Or_Junction, Tissue)
contains(Embryonic_Structure, Body_Substance)
contains(Fully_Formed_Anatomical_Structure, Body_Substance)
```

(2) Des blocages d'héritage :

Ce sont des relations qui ne sont pas systématiquement héritées par les concepts plus spécifiques que les concepts formant sa signature. Prenons l'exemple donné dans la Figure 18 :

```
domain(process_of)=Biologic_Function et range(process_of)=Organism
```

Si cette relation est héritée, nous pouvons avoir :

```
domain(process_of)=Mental_Process et range(process_of)=Plant
```

Cela signifie qu'une plante peut avoir un processus mental, ce qui est absurde car cette fonction biologique est propre aux animaux.

Ces problèmes ont été soulevés dans [Kashyap, 2003] en essayant de formaliser le réseau sémantique en OWL. Depuis, plusieurs travaux tentent d'utiliser ce langage qui offre des primitives supplémentaires par rapport à RDFS, telles que la restriction, la cardinalité, l'intersection et l'union des classes pour résoudre ce genre de problèmes.

La Figure 19 et la Figure 20 montrent deux exemples de solutions que nous proposons en OWL pour résoudre les deux classes de problèmes présentés ci-dessus.

```
<owl:ObjectProperty rdf:ID="process_of">
  <rdfs:domain rdf:resource="#Biologic_Function"/>
  <rdfs:range rdf:resource="#Organism"/>
</owl:ObjectProperty>

<owl:Class rdf:ID="Mental_Process">
  <rdfs:subClassOf rdf:resource="#Biologic_Function"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#process_of" />
      <owl:allValuesFrom rdf:resource="#Animal" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

Figure 19 - Représentation de la relation « process_of » en OWL

La Figure 19 est la traduction en OWL de l’assertion suivante qui définit la signature globale de la relation « process_of » en distinguant les cas particuliers :

Domain(process_of) = “Biologic_Function”
Range(process_of) = “Organism”
 $\exists x,y$ tel que process_of(x,y)
Si $x \in$ “Mental_Process” Alors $y \in$ “Animal”

```

<owl:ObjectProperty rdf:ID="contains">
  <rdfs:domain>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:resource="#Body_Space_Or_Junction" />
        <owl:Class rdf:resource="#Embryonic_Structure" />
        <owl:Class rdf:resource="#Fully_Formed_Anatomical_Structure" />
      </owl:unionOf>
    </owl:Class>
  </rdfs:domain>
  <rdfs:range>
    <owl:Class>
      <owl:unionOf rdf:parseType="Collection">
        <owl:Class rdf:resource="#Body_Part_Organ_Or_Organ_Component" />
        <owl:Class rdf:resource="#Body_Substance" />
        <owl:Class rdf:resource="#Tissue" />
      </owl:unionOf>
    </owl:Class>
  </rdfs:range>
</owl:ObjectProperty>

<owl:Class rdf:ID="Embryonic_Structure">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#contains" />
      <owl:allValuesFrom rdf:resource="#Body_Substance" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="Fully_Formed_Anatomical_Structure">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#contains" />
      <owl:allValuesFrom rdf:resource="#Body_Substance" />
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>

```

Figure 20 - Représentaion de la relation « contains » en OWL

La Figure 20 représente la définition de la relation polymorphe « contains » en définissant sa signature comme étant l'union de toutes les signatures possibles et en excluant les cas particuliers.

Nous pensons que d'un point de vue modélisation, la prise en compte de ce genre de contraintes est nécessaire pour formaliser correctement un domaine. Par contre d'un point de vue usage, ces informations complexifient l'exploitation de l'ontologie. En effet, prenons par exemple les deux scénarios d'usage auxquels nous nous intéressons dans ce travail, à savoir la recherche d'information guidée par les ontologies et la génération d'annotations sémantique basées sur les ontologies :

- Dans le premier scénario, un moteur de recherche guidé par une telle ontologie doit prendre en compte, à chaque fois, toutes les contraintes modélisées afin d'inférer correctement sur les annotations, même si ces dernières ne présentent aucune incohérence par rapport à l'ontologie.
- Dans le deuxième scénario, l'outil d'annotation automatique doit vérifier tous les cas particuliers modélisés dans l'ontologie afin de ne pas générer des annotations erronées (sémantiquement, voire logiquement).

La façon avec laquelle nous avons traité ces deux problèmes est présentée dans les chapitres suivants.

5. L'ontologie DocOnto

5.1. Des métadonnées sur les annotations

Le but de l'utilisation des annotations sémantiques dans la construction d'une mémoire d'entreprise et en particulier dans la construction d'une mémoire d'expérience est de faciliter le partage et la réutilisation des connaissances des différents acteurs de la communauté.

Dans le cadre du projet MEAT, bien que ces annotations fournissent des informations importantes sur les différentes ressources de l'équipe et en particulier sur les articles scientifiques utilisés pour la validation et l'interprétation des résultats obtenus lors d'une expérimentation, elles ne contiennent aucune information concernant leur création.

En effet, le même article scientifique peut être lié à deux expériences différentes par deux personnes différentes afin de décrire deux phénomènes différents ; ce qui peut avoir un impact non négligeable sur le contenu de l'annotation créée pour ce document.

Nous avons donc suggéré que l'identification et la représentation de ces informations peuvent s'avérer utiles pour faire des raisonnements poussés sur les annotations. Pour ce faire, nous avons proposé de rajouter pour chaque document des métadonnées sur les annotations le concernant.

Ces métadonnées décrivent essentiellement le 'contexte de création' des annotations et concernent les trois points suivants :

- la traçabilité de la connaissance : pour faire le lien entre la ressource annotée et sa source. Dans notre cas, la source est le biologiste qui a fourni l'article à annoter ou celui qui a fait l'expérience. Cette information permet (i) de classer les annotations selon leurs sources et (ii) de mettre en place un système de restriction d'accès aux annotations.
- Le thème général de l'annotation : dans le cas des annotations des documents, il s'agit du sujet principal (ou le plus intéressant) traité par le document. Notons que les biologistes peuvent avoir différents centres d'intérêt à propos du même article. Ce thème peut être une instance d'un concept de l'ontologie du domaine (i.e UMLS).

- La génération de l'annotation : cela permet d'indiquer si l'annotation a été générée automatiquement par un outil (i.e MeatAnnot) ou ajoutée/validée par l'utilisateur. Cette information donne une idée sur l'authenticité et la pertinence de l'annotation.

En se basant sur ces métadonnées, un moteur de recherche sémantique pourra renvoyer des résultats qui sont à la fois contextuels et multi-points de vue.

La formalisation de ces métadonnées a été réalisée par l'ajout de nouveaux concepts et de nouvelles relations dans l'ontologie, décrivant les points évoqués ci-dessus. Ces ajouts sont détaillés dans les paragraphes suivants.

5.2. Construction informelle

L'ontologie DocOnto a été proposée dans le but de faire le lien entre les articles et les concepts de l'ontologie UMLS et d'intégrer de nouvelles connaissances concernant les documents et les métadonnées décrites ci-dessus.

La construction informelle de cette ontologie a été réalisée progressivement afin de couvrir nos besoins concernant la description des connaissances contenues dans les textes. Lors de cette construction, nous avons adopté une approche ascendante qui consiste à définir la structure des annotations dont nous avons besoin et ainsi à en déduire la structure du modèle sur lequel se baseront ces annotations.

Nous avons donc commencé par énumérer les informations que nous avons jugées pertinentes et qu'une annotation doit contenir. Ces informations peuvent être organisées en trois sous parties :

- Des informations d'ordre général : elles sont presque communes à tous les systèmes d'annotations et concernent par exemple le titre du documents, ses auteurs et son éditeur.
- La structure du document : un document texte peut être décomposé naturellement en un ensemble de phrases. Ces phrases peuvent être candidates à être annotées si et seulement si elles sont porteuses d'informations. Dans notre cas, elles doivent contenir une ou plusieurs interactions entre des instances des concepts de l'ontologie UMLS.
- Les méta données : concernent les informations déjà détaillées dans le paragraphe précédent.

La Figure 21 représente graphiquement l'ensemble de ces informations.

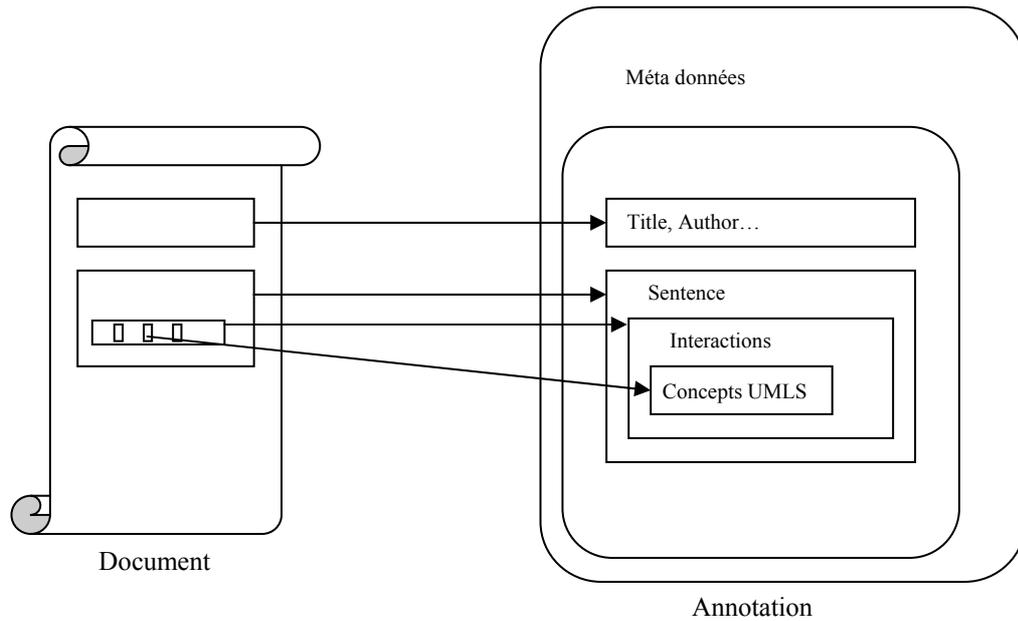


Figure 21 - Représentation d'une annotation d'un document

5.3. Formalisation de DocOnto

La première phase nous a permis de repérer toutes les informations dont nous avons besoin pour annoter un document. Ces informations ont été ensuite formalisées dans l'ontologie sous forme de concepts et de relations. Les deux hiérarchies sont présentées dans la Figure 22.

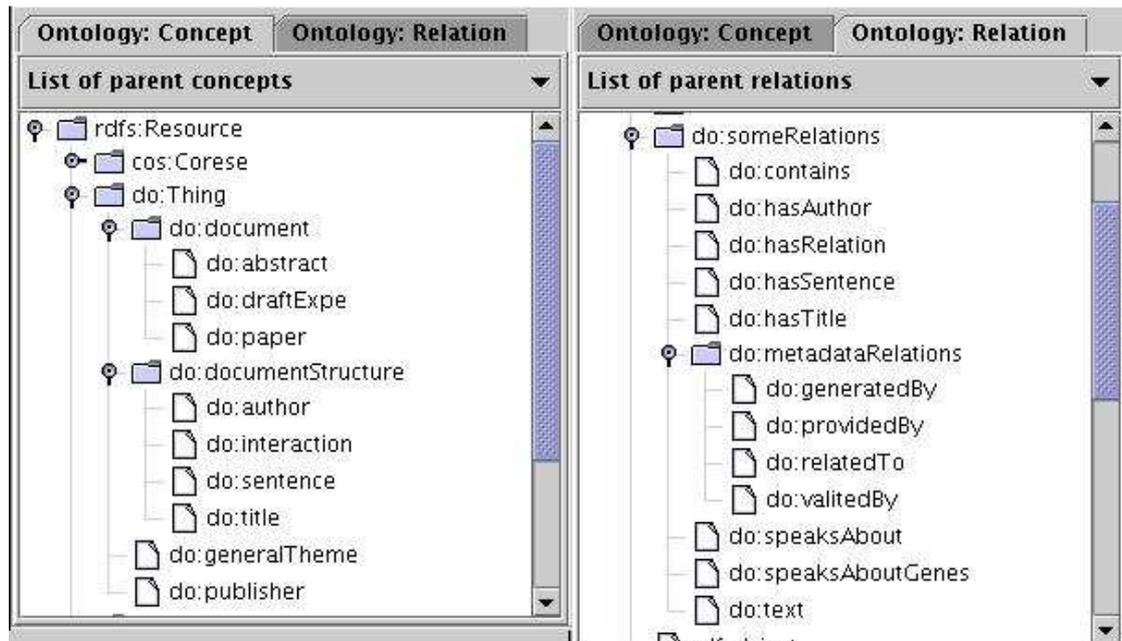


Figure 22 – Les concepts et les relations de l'ontologie DocOnto

Exemples de concepts

Concept	Définition
Document	Entité comprenant des éléments de représentation de la pensée
Sentence	Unité linguistique exprimant un ou plusieurs jugements et comprise entre deux points dans l'expression écrite
generalTheme	Sujet, idée ou proposition développé dans le document

Tableau 4 - Exemples de concepts de l'ontologie DocOnto

Exemples de relations

Relation	Signature	Définition
hasTitle	Domain= document Range = String	Désignation du document
relatedTo	Domain = document Range = un concept UMLS	Relation dénotant que le document est lié à un thème exprimé sous la forme d'un concept biomédical
validatedBy	Domain= interaction Range = String	Relation dénotant que l'interaction annotée est validé par une personne

Tableau 5 - Exemples de relations de l'ontologie DocOnto

De même que les deux autres ontologies, DocOnto a été codée en RDFS.

6. Une ontologie pour une mémoire d'expériences

Dans ce travail, nous présentons une approche orientée Web Sémantique pour la construction d'une mémoire d'expériences. Dans cette approche, l'ontologie constitue un composant très important et peut jouer des rôles variés [Dieng, 2004]: (i) l'ontologie peut être destinée à être exploitée par l'utilisateur final, (ii) l'ontologie peut être considérée comme une référence pour annoter sémantiquement la mémoire à des fins d'amélioration de la recherche de ressources ou d'informations dans la mémoire, et (iii) l'ontologie peut constituer une base pour la communication et l'échange d'information entre des programmes ou agents logiciels. Cependant la structure de cette ontologie peut différer d'un projet à un autre ou d'un type de mémoire à un autre.

Suite à notre expérience acquise lors du projet Meat, nous avons conçu une structure 'type' de l'ontologie qui peut être utilisée dans la construction d'une mémoire d'expériences en général indépendamment du domaine considéré. Cette structure est partiellement inspirée de celle l'ontologie O'CoMMA [Gandon, 2002].

Cette ontologie, comme le montre la Figure 23, est structurée en trois niveaux :

- Un haut niveau : constitué de concepts abstraits et réutilisables permettant en général de définir la structure hiérarchique de l'ontologie (par exemple les concepts *Thing*, *Entity* et *Event*);
- Un niveau médian : constitué de concepts facilitant les scénarios que la mémoire d'expérience traite (recherche d'information, aide à la validation...);
- Un niveau spécifique : constitué de concepts typiques de l'activité considérée (dans notre cas c'est l'expérience biopuce), donc peu réutilisable dans d'autres activités. Ce niveau se décompose en quatre sous-parties :
 - Domaine : décrit le domaine spécifique étudié par les expériences (dans notre cas c'est le domaine biomédical couvert par UMLS);
 - Expériences : comprend la description technique des expériences ainsi que leurs caractéristiques (c'est le cas de l'ontologie MGED pour les expériences biopuces);
 - Annotations : comprend des concepts permettant de rajouter des métadonnées sur les annotations afin de décrire le point de vue ou le contexte d'une annotation (cette partie est comprise dans l'ontologie DocOnto de notre travail);
 - Participants et tâches : décrit les membres de l'équipe effectuant les expériences ainsi que leurs tâches spécifiques (dans Meat, cette partie concerne les personnes validant et interprétant les résultats d'une expérience).

Notons ici que des interactions peuvent exister entre les différentes parties présentées dans le niveau spécifique. En effet, les connaissances du domaine étudié semblent indispensables (i) dans la description (même technique) des expériences et (ii) dans la définition des métadonnées sur les annotations (point de vue, contexte de l'expérience...).

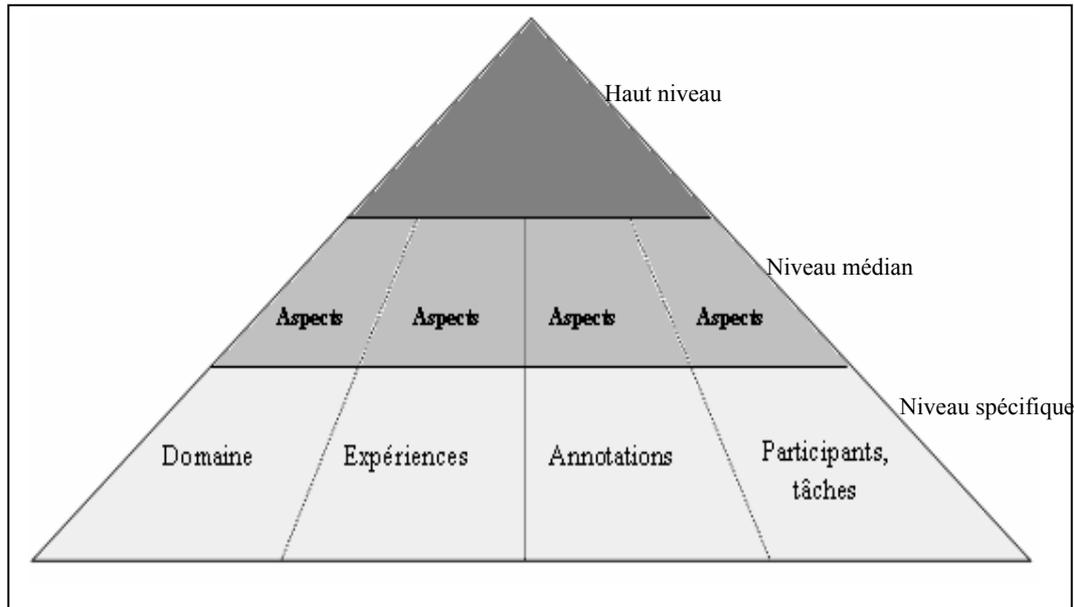


Figure 23 - Structure de l'ontologie pour la mémoire d'expériences

7. Conclusion

Dans cette partie, nous avons présenté l'ontologie MeatOnto, composant sur lequel va se baser le reste de notre travail et duquel nous nous sommes inspirés pour proposer une structure générique du composant ontologie dans la construction d'une mémoire d'expériences.

Nous avons montré aussi que des ontologies qu'elles soient de haut niveau (le réseau sémantique de UMLS) ou de domaine (MGED) peuvent être réutilisées si elles ont été développées selon un consensus ontologique fort dans lequel participent les différents acteurs du domaine étudié.

Par ailleurs, pour la formalisation de nos ontologies, nous avons utilisé le langage RDFS (et quelques primitives du langage OWL Lite) et ce pour trois raisons :

- C'est un standard du W3C pour la construction d'ontologies légères (le cas de nos ontologies) permettant ainsi leur partage et leur réutilisation dans d'autres applications.
- Le niveau d'expressivité de ce langage nous a paru suffisant pour la formalisation des ontologies utilisées (mis à part quelques problèmes discutés dans le §4.4) ce qui nous a amené à ne pas utiliser un langage plus complexe tel que OWL DL ou FULL.
- L'adoption de ce langage nous permet d'utiliser l'outil Corese pour la recherche d'information guidée par l'ontologie (Voir Chapitre 5). En effet cet outil implémente le langage RDFS et quelques primitives du langage OWL.

Après avoir présenté l'ontologie MeatOnto qui constitue la base de notre mémoire d'expérience, nous décrivons dans le chapitre suivant une contribution principale de notre travail, à savoir la méthodologie de génération automatique d'annotations sémantiques basées sur cette ontologie.

Chapitre 4 - Génération automatique des annotations : MeatAnnot

1. Présentation générale

1.1. Introduction

Rappelons que le but de la mémoire d'expériences que nous proposons est de rendre facilement accessibles certaines connaissances cruciales aux biologistes réalisant ces expériences. Ces connaissances peuvent provenir de plusieurs sources (i.e. humaines, BD...). Cependant, dans un domaine de recherche scientifique tel que le domaine des puces à ADN, les publications représentent la source la plus importante et la plus riche. En effet, les connaissances contenues dans ces documents permettent aux biologistes à la fois, d'émettre des hypothèses sur leurs expériences, de valider les résultats obtenus et enfin de pouvoir interpréter ces derniers.

Nous nous posons donc la question : comment faciliter le plus possible l'accès à ces connaissances ?

Comme nous l'avons décrit dans le chapitre 2, dans notre travail, nous avons opté pour l'utilisation des techniques du Web Sémantique et en particulier pour l'utilisation des annotations sémantiques basées sur les ontologies pour faciliter l'accès aux connaissances contenues dans les textes.

Mais bien que ces annotations permettent de décrire rigoureusement le contenu sémantique des documents, leur création reste un processus difficile et coûteux pour les biologistes (temps, personnes ...).

L'objectif de cette partie de notre travail est de proposer une méthodologie et un système pour la génération (semi-) automatique d'annotations sémantiques à partir des textes [Khelif et Dieng-Kuntz, 2004].

1.2. Vue d'ensemble

1.2.1. Motivation

Les biologistes nous ont tout d'abord fourni plusieurs articles de revue dans lesquels nous leur avons demandé de surligner les informations essentielles transmises par ces articles papiers : nous avons considéré celles-ci comme constituant donc les annotations de ces biologistes sur les articles, annotations que nous devons reproduire automatiquement si possible. À partir de ces annotations manuelles, nous avons pu déterminer les points importants soulignés par les biologistes comme intéressants pour caractériser le contenu sémantique des articles et guider leurs recherches : les gènes supposés a priori intéressants pour l'expérience, les substances, les protéines étudiées, les phénomènes biologiques et les fonctions cellulaires, etc. ainsi que les interactions entre ces différents entités.

Toutes ces informations nous ont permis d'élaborer une méthodologie pour la génération automatique d'annotations sémantiques. Ces annotations décrivent les relations entre les termes

contenus dans le texte et qui sont jugés intéressants pour le domaine, tout en se basant sur les concepts et les relations de l'ontologie.

1.2.2. Notre approche

L'approche standard de la modélisation d'un domaine à partir des textes consiste à (i) identifier les termes caractérisant le domaine, pour ensuite (ii) extraire les relations sémantiques qui les unissent.

Cette approche peut être utilisée, en inversant l'ordre des étapes, dans un contexte de génération à partir des textes d'annotations sémantiques basées sur un modèle déjà défini du domaine (i.e. l'ontologie). Nous préconisons ici, que le repérage d'une instance d'une relation de l'ontologie dans une phrase de l'article, indique que cette phrase peut être porteuse d'informations. Donc, disposant de l'ensemble des relations et de l'ensemble des termes qui peuvent être liés par ces relations, l'approche que nous proposons pour la génération d'une annotation se décompose en deux étapes (présentées dans la Figure 24): (i) détection d'une instance d'une relation dans le texte, et (ii) identification des termes constituant les arguments de cette relation.

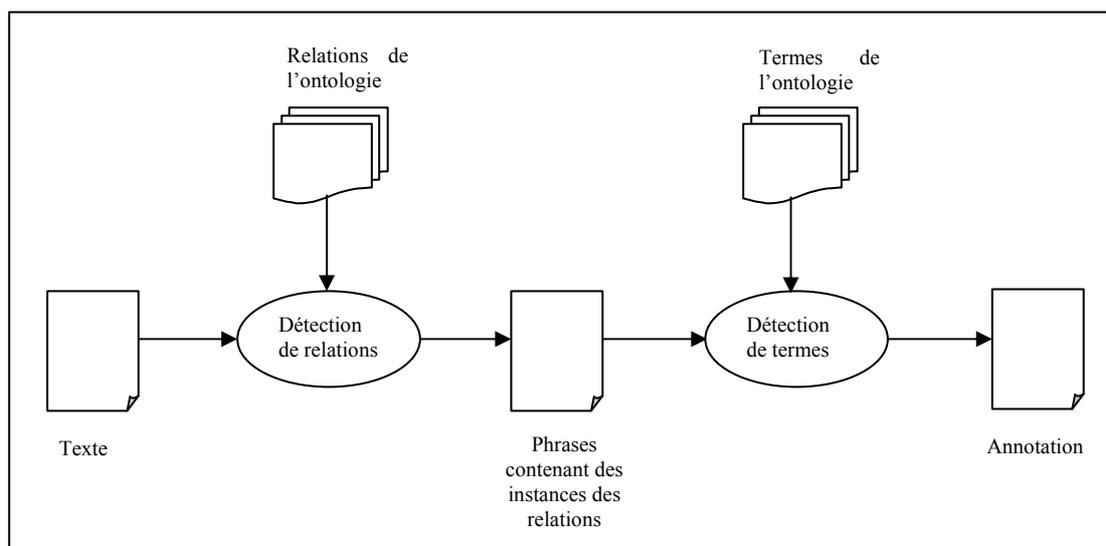


Figure 24 - Les étapes de la génération des annotations basées à partir du texte

Nous basant sur cette approche, nous avons proposé une méthodologie pour la génération d'annotations sémantiques à partir des textes. Cette méthodologie peut se décomposer en quatre grandes phases :

- La première phase consiste à effectuer une série d'analyses linguistiques sur le texte afin de le préparer pour les phases d'extraction;
- La deuxième vise à repérer les instances des relations de l'ontologie. Nous considérons que chaque relation est caractérisée par un ensemble de verbes et de syntagmes verbaux, et que l'apparition d'un de ces syntagmes dans le texte peut être considérée comme une instanciation de cette relation (Par

exemple la relation *prevents* est caractérisée par : « has a preventive effect », « prevents », etc.);

- La troisième phase consiste à analyser les phrases contenant une éventuelle instance d'une relation afin d'en extraire les instances des concepts de l'ontologie reliées par cette relation;
- Enfin, la quatrième et dernière phase consiste à collecter toutes les informations issues des phases précédentes pour générer une annotation structurée basée sur l'ontologie. Pour chaque document, une annotation globale décrivant ainsi son contenu est générée.

La Figure 25 montre le schéma général de cette méthodologie.

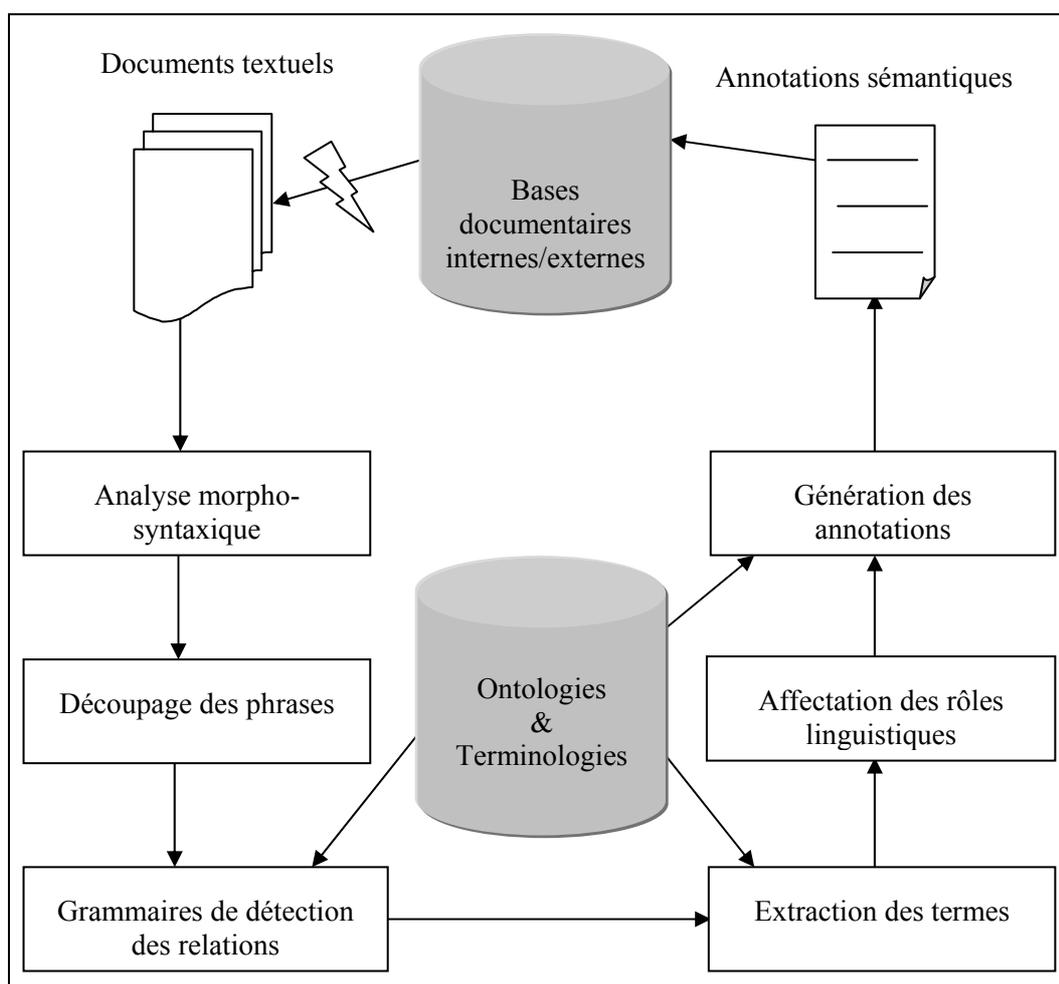


Figure 25 - Vue d'ensemble de la méthodologie

Remarque :

Une phase zéro qui consiste à convertir les documents depuis leurs différents formats de publication (pdf, ps, etc.) vers un format textuel est indispensable pour commencer le traitement linguistique.

Cette méthodologie a été implémentée dans l'outil que nous avons nommé MeatAnnot. Cet outil se base sur l'ontologie MeatOnto pour la génération d'annotations sémantiques sur les articles de référence du domaine des biopuces. Dans ce qui suit, nous détaillons les fonctionnalités et les composantes de cet outil.

1.3. Outils de TALN utilisés

Avant de détailler les fonctionnalités de MeatAnnot, nous présentons ici les outils de traitement automatique de la langue naturelle que nous avons utilisés pour la réalisation de ce travail.

1.3.1. GATE

GATE est une plate-forme d'ingénierie linguistique [Cunningam et al., 2002] qui repose sur l'application successive de transducteurs³⁵ aux textes. Conformément aux termes employés par ses concepteurs, nous parlons ici de ressources de traitement (Processing Resources : PR). Ces ressources de traitement utilisent le texte³⁶ modifié par les ressources précédemment appliquées pour ajouter de la structure au texte. Les ressources de traitement les plus courantes sont les segmenteurs (Tokenizers), les analyseurs morpho-syntaxiques (Part Of Speech ou POS Taggers), les lexiques (Gazetteers), les transducteurs (JAPE transducers), et les patrons d'extraction (Templates). Ils sont appliqués au texte au sein d'une cascade (chaîne de traitement ou pipeline).

GATE peut être utilisé de deux façons différentes : environnement de développement ou bibliothèque.

L'utilisation la plus simple est comme environnement de développement au travers des ressources développées par ses concepteurs. L'environnement de développement est constitué d'une interface graphique permettant aux utilisateurs de créer de nouvelles ressources ou paramétrer des ressources disponibles et de les appliquer aux textes au sein d'une chaîne de traitement. Nous avons utilisé cette interface pour tester nos modules d'extraction de termes et de relations et pour visualiser les résultats.

Le second niveau d'utilisation consiste à tirer parti des ressources disponibles dans le système. Nous pouvons les utiliser au sein de l'environnement de développement ou de façon embarquée dans des applications autonomes. De cette manière, il est possible de se passer de l'interface graphique de GATE et de traiter un texte dans un programme autonome hors de l'environnement de développement. C'est l'utilisation pour laquelle nous avons opté, nous avons ainsi bénéficié de l'architecture et de la bibliothèque de GATE afin d'intégrer nos différents modules et afin de les appliquer en chaîne sur les textes.

³⁵ Un transducteur est un automate à états finis qui, pour chaque état parcouru, produit une ou plusieurs informations

³⁶ Le texte est transformé en un arbre XML afin de permettre le partage d'informations entre les différents modules

1.3.2. TreeTagger

TreeTagger est un outil développé au sein l'institut de linguistique computationnelle de l'université de Stuttgart [Schmid, 1994]. Cet outil, utilisé pour l'étiquetage des textes en français et en anglais, spécifie pour chaque mot sa catégorie syntaxique et indique son lemme. L'estimation de la catégorie grammaticale d'un mot se base sur la construction récursive d'arbres de décisions binaires et un calcul de probabilité.

Le Tableau 6 représente un exemple de sortie pour l'entrée suivante "The TreeTagger is easy to use." :

Mot	Catégorie	Signification	Lemme
The	DT	Article	The
TreeTagger	NP	Nom propre	TreeTagger
Is	VBZ	Verbe conjugué au présent	Be
Easy	JJ	Adjectif	Easy
to	TO	TO : catégorie spéciale	to
Use	VB	Verbe à l'infinitif	Use
.	SENT	Ponctuation	.

Tableau 6 - Résultat de TreeTagger sur l'exemple précédent

Nous avons intégré cet outil dans la plate-forme GATE par le biais d'un traducteur (wrapper) que nous avons développé, et nous l'avons utilisé pour l'étiquetage grammatical et la lemmatisation des textes. Les informations concernant chaque mot sont intégrées dans la structure XML du document texte.

Notons ici que GATE propose son propre étiqueteur grammatical mais ce dernier ne calcule pas la forme lemmatisée des mots, information indispensable dans notre processus.

1.3.3. RASP

RASP (Robust Accurate Statistical Parsing) [Briscoe et Carroll, 2002] est un analyseur probabiliste pour l'anglais, il permet par des calculs statistiques de prédire la relation grammaticale (sujet, objet...) de chaque mot dans le texte. Il fournit comme résultat une forêt de mots interconnectés par des probabilités.

Considérons la phrase « RASP predicts grammatical relations. », le résultat de RASP sur cette phrase est le suivant :

```

("RASP" "predicts" "grammatical" "relations.")
(|ncsubj| |predict+s:2_VVZ| |RASP:1_NN1| |_)
(|dobj| |predict+s:2_VVZ| |relation.+s:4_NN2| |_)
(|ncmod| |relation.+s:4_NN2| |grammatical:3_JJ|)
    
```

Le résultat ci-dessus montre que le mot 'RASP' a été détecté comme étant le sujet du verbe 'predict' et que le mot 'relation' a été détecté comme étant l'objet de ce même verbe. Il montre aussi l'existence d'une dépendance entre les mots 'relation' et 'grammatical'.

Nous avons intégré cet outil dans la plate-forme GATE par le biais d'un traducteur (wrapper) que nous avons développé, et nous l'avons utilisé pour le calcul des relations grammaticales entre les termes. Comme pour TreeTagger, ce traducteur permet d'analyser le résultat de RASP et de rajouter des informations concernant chaque mot à la structure XML du texte traité.

1.4. Identification des termes de UMLS

Comme pour toute source terminologique, une des utilisations possibles de UMLS consiste à fournir une aide au repérage des termes dans le texte et à faire la correspondance entre ces termes et les concepts spécifiques au domaine étudié (le domaine biomédical dans notre cas). La complexité et la richesse du métathésaurus de UMLS rendent cette tâche assez complexe.

C'est dans cette optique que le NLM a proposé un service web nommé UMLS SKS (UMLS Knowledge Server) permettant l'interrogation du métathésaurus et du réseau sémantique. Ce service intègre le système MetaMap [Aronson, 2001] qui utilise un générateur de variations et qui tient en compte : les différentes formes des mots (lemmatisation), les synonymes/acronymes et quelques variations linguistiques ('development of lung' est reconnu comme 'lung development').

L'UMLS SKS peut être ainsi utilisé pour faire la correspondance entre des candidats-termes extraits à partir du texte et les concepts qui leur sont associés dans l'ontologie UMLS. La Figure 26 présente le résultat de UMLS SKS interrogé sur le terme « development of lung ».

```
<?xml version="1.0" encoding="UTF-8"?>
<SemanticTypeCollection version="1.0">
  <query>
    <getSemanticType version="1.0">
      <release>2005AB</release>
      <cui>C1160389</cui>
    </getSemanticType>
  </query>
  <release>2005AB</release>
  <cui>C1160389</cui>
  <cn>Lung Development</cn>
  <semanticType>
    <tui>T042</tui><sty>Organ or Tissue Function</sty><atui>AT08828147</atui>
  </semanticType>
</SemanticTypeCollection>
```

Figure 26 - Résultat de la requête « development of lung » renvoyé par l'UMLS SKS

Pour la phase d'extraction de termes, nous avons donc intégré la bibliothèque de fonctions permettant l'interrogation de ce service.

2. Démarche de l'extraction des connaissances à partir des textes

Dans cette partie, nous détaillons les différentes étapes de notre méthodologie présentée dans la Figure 25. Ces étapes, comme nous l'avons souligné précédemment, ont été implémentées dans le système MeatAnnot.

MeatAnnot utilise la plate-forme GATE pour intégrer plusieurs modules et les appliquer en cascade sur le texte brut. Ces différents modules sont décrits dans les sections suivantes.

2.1. Analyse morpho-syntaxique des textes

La manipulation automatique de textes écrits en langage naturel, quelle que soit la langue utilisée, nécessite souvent une première analyse des entités formant ces textes. L'objectif de cette étape est de récupérer toute information permettant de caractériser le comportement d'un mot dans son contexte d'énonciation.

Dans notre cas, cette analyse comprend :

- Le découpage du texte en phrases : cette étape permet de repérer les frontières de chaque phrase dans le texte afin de pouvoir la séparer pour un éventuel traitement spécifique.
- Le repérage des entités linguistiques de base (tokenisation), à savoir les mots (tokens). Cette étape permet aussi de distinguer la morphologie de ces entités (ponctuation, nombres, etc.) et de retrouver la racine de chaque entité (lemmatisation).
- L'étiquetage grammatical du texte : associer à chaque *token* une catégorie d'ordre grammatical (Nom, Verbe, Adjectif, etc.) en tenant compte du contexte de leur occurrence.

MeatAnnot intègre deux modules de GATE pour effectuer les deux premières étapes (à savoir le Sentence_Splitter et le Tokenizer) et utilise le TreeTagger pour l'étiquetage grammatical.

Cette phase fournit des informations de base sur les textes qui sont nécessaires pour les phases suivantes.

2.2. Détection des relations sémantiques

2.2.1. Repérage des relations sémantiques dans les textes

Nous rappelons que le but de cette phase est de repérer dans le texte, les relations sémantiques déjà modélisées dans l'ontologie afin de générer des annotations basées sur ces relations. Ce repérage consiste à essayer d'identifier les différentes formes syntaxiques d'apparition des relations dans le texte. Ces formes syntaxiques peuvent ainsi être considérées comme des instances possibles de relations formalisées dans l'ontologie.

Les relations sémantiques se décomposent généralement en deux grandes familles :

- Les relations syntagmatiques : qui sont identifiables par une étude directe des formes syntaxiques : chaque relation est caractérisée par un ou plusieurs syntagmes (verbaux, adjectivaux...). Ces relations sont liées à leurs arguments au niveau syntaxique.
- Les relations paradigmatisées : qui sont l'opposé des relations syntagmatiques car elle n'apparaissent pas sous la forme d'un lien syntaxique standard. C'est le cas, par exemple, de la synonymie ou l'hyponymie. La détection de ce type de relations nécessite généralement une validation humaine.

À des fins d'automatisation, nous nous sommes intéressés à la première famille de relations, à savoir, les relations syntagmatiques et en particulier les relations pouvant être caractérisées par des syntagmes verbaux. Cette caractéristique permet d'identifier plus « facilement » les arguments de la relation en se basant sur les catégories grammaticales des termes ayant une dépendance syntaxique avec le verbe caractérisant la relation.

La Figure 27 montre la déduction automatique de l'existence de la relation 'has a role in' entre deux termes d'une phrase.

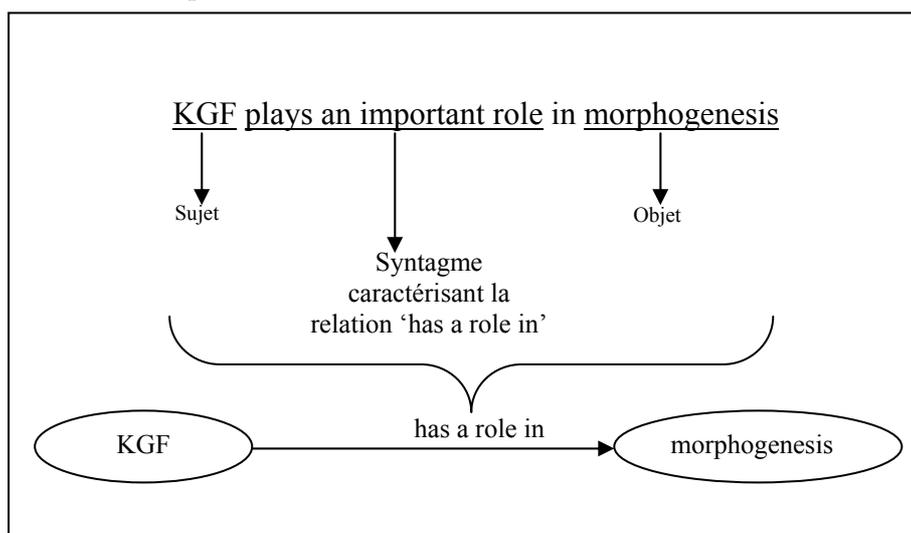


Figure 27 – Exemple de repérage d'une relation sémantique dans le texte

Dans ce qui suit, nous présentons la technique que nous proposons pour le repérage de ces relations.

2.2.2. Les grammaires de détection de relations

Une grammaire de détection est une grammaire qui couvre les différentes formes syntaxiques d'apparition d'une relation dans le texte. À chaque relation sont associées une ou plusieurs grammaires, l'application de ces dernières sur le texte permet d'identifier une éventuelle expression de cette relation.

2.2.2.1. JAPE : un langage d'expression de grammaires pour le TALN

Le langage JAPE (*Java Annotation Patterns Engine*), une variante du standard CPSL adapté au langage de programmation Java [Cunningham et al., 2002], a été proposé pour écrire des grammaires qui, une fois appliquées sur le texte, permettent d'y ajouter des informations en forme d'annotations.

Une grammaire de JAPE comporte un ensemble de phases, chaque phase étant un ensemble de règles sous forme de patron et action. Dans une règle, si les patrons sont satisfaits, alors une action (ex. attribution d'une étiquette d'annotation) pourra être déclenchée. La partie droite d'une règle qui contient des patrons doit être écrite en JAPE mais la partie gauche, qui contient les actions, peut être écrite en JAPE ou en JAVA.

Ci-dessous un exemple d'une grammaire JAPE permettant de détecter les adresses IP dans le texte.

```
Rule: IPAddress
( {Token.kind == number} {Token.string == "."}
  {Token.kind == number} {Token.string == "."}
  {Token.kind == number} {Token.string == "."}
  {Token.kind == number} )
:ipAddress --> :ipAddress.Address = {kind = "ipAddress"}
```

Etant donné que GATE dispose d'un transducteur permettant l'application des grammaires JAPE sur le texte, nous avons utilisé ce langage pour écrire nos grammaires de détection de relations.

2.2.2.2. Définition des grammaires de détection pour UMLS

Chaque relation sémantique ayant une forme verbale peut être caractérisée par un ensemble de verbes ou de syntagmes verbaux. Par exemple, la relation '*has_a_role_in*' est caractérisée par *{has a X role in, plays a X role in}* ; où X représente un adjectif qualificatif (important, vital, critical...).

Une idée pour repérer une instance d'une relation consiste à détecter l'apparition de l'un des verbes ou l'un des syntagmes verbaux dans le texte. Pour ce faire, nous avons utilisé le langage JAPE décrit précédemment pour écrire des grammaires permettant d'effectuer cette tâche.

Si nous reprenons l'exemple précédent concernant la relation '*has_a_role_in*', la grammaire décrivant cette relation est présentée dans la Figure 28 :

```

Rule:Has_role
Priority: 1
(
  ({Tag.lemme == "have"} |
  {Tag.lemme == "play"})
  {SpaceToken}
  ({Tag.lemme == "a"} |
  {Tag.lemme == "an"})
  {SpaceToken}
  ({Tag.cat == "JJ"} {SpaceToken})?
  {Tag.lemme == "role"}
  {Tag.lemme=="in"})

):has_role -->
:has_role.RelationShip = {kind = "has_role", rule=Has_role}

```

Figure 28 - La grammaire de détection de la relation ‘*has_a_role_in*’

Dans la grammaire ci-dessus, Tag.lemme correspond à la forme lemmatisée du verbe et Tag.cat correspond à la catégorie grammaticale (JJ=adjectif) du terme qui peut être présent entre le verbe et le terme ‘role’ (le ? signifie qu’il est optionnel).

Ces grammaires sont appliquées sur le texte déjà analysé et exploitent les informations collectées lors de la phase de l’analyse morpho-syntaxique, à savoir, la tokenisation, la lemmatisation et l’étiquetage grammatical.

Dans l’exemple précédent, le syntagme verbal ‘has a X role in’ caractérise exclusivement la relation ‘*has a role in*’ mais en étudiant chaque relation, nous avons remarqué qu’il existe des syntagmes verbaux pouvant caractériser plusieurs relations, c’est le cas du syntagme ‘has X effects’. En effet, la présence du syntagme verbal ‘has a preventive effect’ permet de repérer une éventuelle instance de la relation ‘*prevents*’, alors que la présence du syntagme ‘has a negative effect’ permet de repérer une éventuelle instance de la relation ‘*disrupts*’. Dans ces cas, et afin d’éviter des problèmes d’ambiguïté, le X est implicitement défini dans chaque grammaire. Ces informations intéressantes sur les caractéristiques linguistiques des relations ont été collectées par deux méthodes :

- en se basant sur les résultats de l’outil Syntex sur un corpus de test (une dizaine de documents fournis par les biologistes). En effet, la mise en correspondance que nous avons réalisée entre la liste des syntagmes verbaux extraits à partir des textes par Syntex et l’ensemble des relations de UMLS, nous a permis d’avoir beaucoup d’informations sur les formes syntaxiques d’apparition de certaines relations dans le texte.
- en se basant sur les remarques émises par les biologistes lors de la validation de l’approche sur ce corpus de test. En effet, lors de cette phase de validation, les biologistes nous ont signalé des phrases où les grammaires n’ont pas pu détecter la présence d’une relation. Ces phrases, une fois analysées, nous ont permis d’enrichir les grammaires.

Le transducteur exécutant les grammaires de détection, génère des informations concernant le type de la relation détectée, la phrase où la relation a été détectée et l'emplacement exact dans le texte. Ces informations sont ensuite utilisées dans la phase de génération de l'annotation.

2.3. Extraction des candidats-termes

Lorsqu'une instance d'une relation sémantique est détectée, MeatAnnot procède à l'identification des termes représentant des instances des concepts de l'ontologie qui sont dans la même phrase. Cette identification permet de déterminer les termes de l'ontologie pouvant être liés par la relation détectée.

2.3.1. Le processus d'extraction des termes

Comme nous l'avons indiqué, cette phase consiste à extraire les syntagmes (ou termes simples) susceptibles d'être des instances des concepts de l'ontologie. Cette extraction se décompose en deux étapes :

- La première consiste à traiter le texte et constituer une liste de candidats termes. Cette étape peut être réalisée à l'aide de n'importe quel extracteur de termes, à savoir par exemple Lexter, Nomino, Syntex, etc. (aucun outil particulier n'est imposé).
- La deuxième consiste à faire la correspondance entre la liste des candidats-termes et les termes de l'ontologie afin de filtrer la première et de n'en garder que les instances des concepts de l'ontologie.

Pour notre part, en ayant choisi UMLS comme ontologie pour décrire le domaine biomédical et disposant d'un service assez puissant permettant d'interroger cette ontologie tout en traitant les variations linguistiques des termes (i.e. UMLSKS, décrit dans §1.4), nous avons implémenté notre propre extracteur de termes qui combine les deux étapes citées précédemment.

Notre extracteur de termes implémente un algorithme assez simple, qui pour chaque phrase à traiter, utilise une fenêtre de taille n ³⁷ (n mots successifs peuvent constituer un candidat-terme) pour construire un terme, si ce terme est validé par UMLSKS, il passe au mot suivant, sinon la fenêtre d'extraction est diminuée jusqu'à ce qu'elle devienne vide.

Afin d'alléger cette phase d'extraction, l'extracteur de termes exploite le résultat de l'analyse morpho-syntaxique pour optimiser le traitement et intègre quelques restrictions, telles que : un terme ne peut ni commencer ni se terminer par un verbe, un terme ne peut ni commencer ni se terminer par une préposition, un terme ne peut être un nombre, etc.

Nous avons développé cet extracteur en utilisant la bibliothèque de fonctions de GATE afin de pouvoir l'intégrer automatiquement dans notre chaîne de traitement.

La Figure 29 présente une interface de GATE montrant le résultat de l'extracteur de termes sur une phrase contenant deux instances de deux relations de UMLS.

³⁷ Dans MeatAnnot nous avons choisi $n=4$

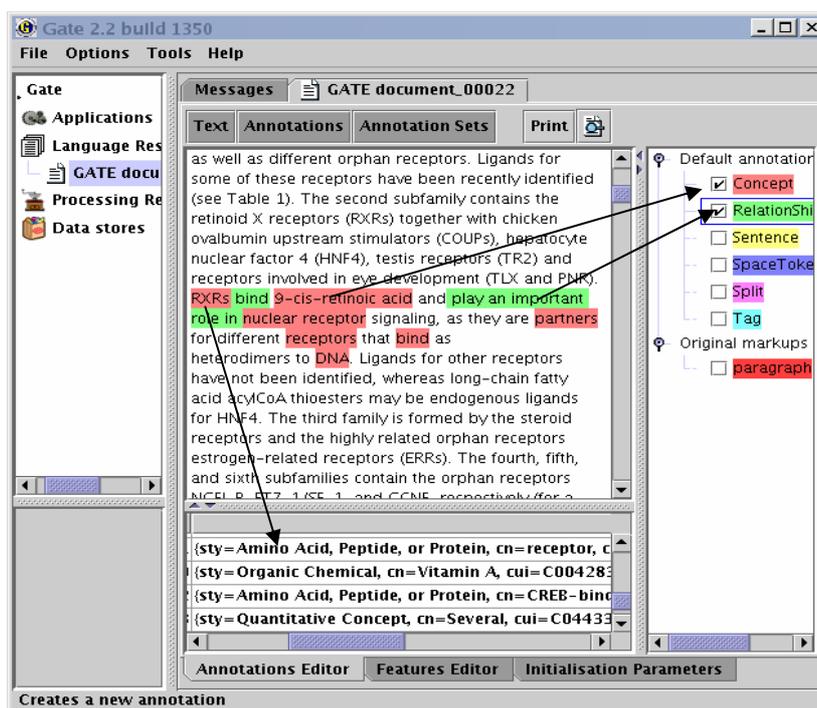


Figure 29 – Résultats de la phase d'extraction de termes

2.3.2. Expansion de la liste des candidats-termes

Comme nous pouvons le remarquer, la phase d'extraction de termes telle qu'elle est décrite dans le paragraphe précédent se base essentiellement sur les termes répertoriés dans le métathésaurus de UMLS. Cependant, malgré la richesse de ce métathésaurus, lors de la validation des résultats de cette phase, nous avons remarqué quelques limites liées à l'absence de quelques termes et que nous avons jugé intéressants pour le domaine étudié à savoir le domaine des biopuces.

En effet, pour pouvoir annoter des articles scientifiques traitant ces expériences dont le but essentiel est d'étudier les gènes, il est indispensable d'avoir une couverture quasi totale des noms des gènes recensés ainsi que de leurs fonctions.

Nous décrivons dans ce qui suit, les solutions que nous avons adoptées afin de remédier à ces deux manques, soit les noms des gènes d'un côté et de leurs fonctions d'un autre.

Solution 1 : Intégration d'un dictionnaire de gènes

Bodenreider a réalisé une évaluation du métathésaurus de UMLS dans le but de calculer cette couverture. Les résultats de cette évaluation (détaillés dans [Bodenreider, 2002]) montrent que le métathésaurus de UMLS ne couvre que 20% des noms de gènes recensés dans la base

LocusLink³⁸ et 15% de leurs alias. Ce manque est dû au souci des développeurs de UMLS à minimiser les ambiguïtés dans le métathésaurus.

Afin de remédier à cela, nous avons récupéré un dictionnaire élaboré au sein de l'IPMC et qui regroupe les noms de gènes fréquemment utilisés dans les expériences de puces à ADN. Ce dictionnaire qui comporte 23152 entrées (45250 noms de gènes en comptant les alias) a été intégré dans la phase d'extraction de termes pour retrouver les noms de gènes qui n'existent pas dans le métathésaurus de UMLS.

Solution2 : Intégration d'heuristiques

Le but de cette partie est d'extraire des familles de termes jugés intéressants par les biologistes mais n'existant pas dans le métathésaurus de UMLS, nous citons en particulier les fonctions génétiques et les fonctions moléculaires.

Prenons l'exemple de la phrase suivante :

“ERK-5 also plays a role in the AP-1 regulation”

L'extracteur de termes dans sa deuxième version (incluant le dictionnaire) permet d'extraire facilement les deux termes 'ERK-5' et 'AP-1' en tant que noms de gènes, ce qui permet à MeatAnnot de déduire la relation 'has_a_role_in' entre ces deux termes. L'annotation générée pour cette phrase manque en précision car le terme 'AP-1 regulation' qui n'appartient pas au métathésaurus de UMLS n'a pas été identifié alors qu'il aurait dû constituer le deuxième argument de la relation 'has_a_role_in'.

Afin de résoudre ce problème, nous avons proposé des heuristiques permettant de détecter les fonctions génétiques et les fonctions moléculaires qui n'appartiennent pas au métathésaurus de UMLS. H1 et H2 sont deux exemples d'heuristiques proposées :

H1: {term1.sty == 'Gene_or_Genome'}
{term2.string ∈ GF_termes} =>
{term3 = term1+term2; term3.sty = 'Genetic_Function'}

où :

GF_termes = {'induction', 'translation', 'regulation', 'expression', 'mutation', 'deletion'}

H2: {term1.sty == 'Amino_acid_Peptide_or_Protein'}
{term2.string ∈ MF_termes } =>
{term3 = term1+term2; term3.sty = 'Molecular_Function'}

où :

MF_termes = {'activity', 'binding', 'phosphorylation'}

³⁸ <http://www.ncbi.nih.gov/entrez/query.fcgi?db=gene> : une base de gènes considérée comme référence dans le domaine de la génomique

Dans les exemples précédents, H1 implique que, si un mot est détecté comme une instance d'un gène et qu'il est suivi par le mot '*regulation*' par exemple, alors la concaténation des deux mots peut être considérée comme une instance du concept 'Genetic_Function'.

La Figure 30 montre le schéma final de notre extracteur de termes.

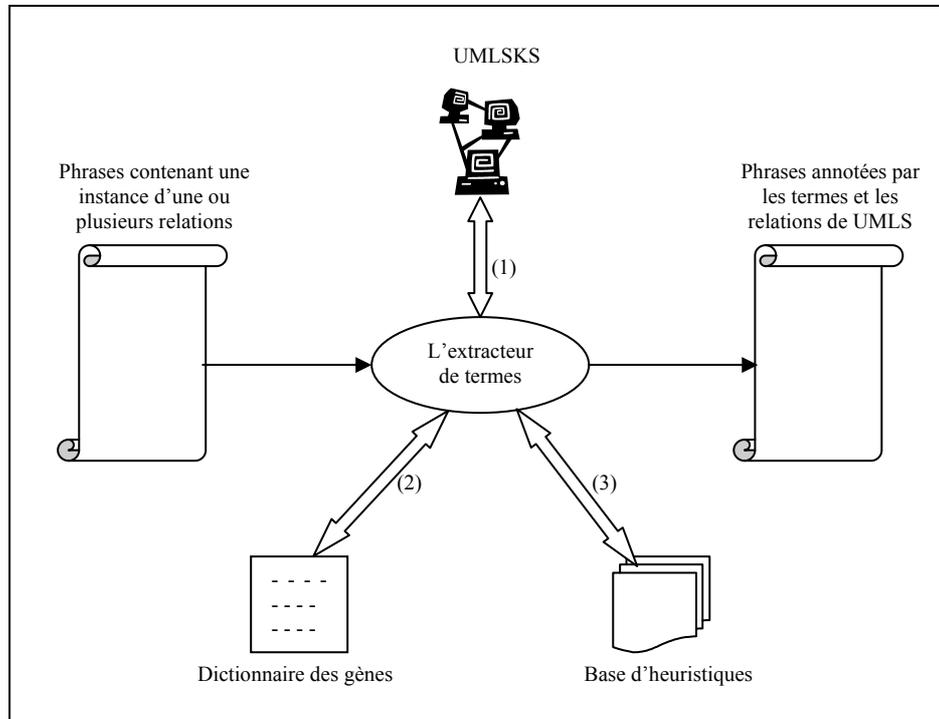


Figure 30 - Le schéma de l'extracteur de termes

2.4. Génération de l'annotation

2.4.1. Processus général

Une fois les relations détectées et les termes extraits, la dernière phase consiste à identifier les termes constituant les arguments de chaque relation afin de générer une annotation sémantique globale pour tout le document.

Dans cette phase, MeatAnnot exploite les informations générées lors des trois premières phases :

- Les phrases contenant des éventuelles instances des relations de UMLS ;
- Les termes représentant des instances des concepts de UMLS extraits à partir de ces phrases ;
- Les relations grammaticales de chaque terme, obtenues grâce à l'application de RASP sur les phrases en question.

Disposant de toutes ces informations, MeatAnnot applique alors l'algorithme de génération de l'annotation et qui se décompose en quatre étapes :

- Identifier les arguments de chaque relation en étudiant les relations grammaticales de chaque mot dans la phrase contenant le verbe (ou le syntagme verbal) caractérisant la relation : les 'sujets' et les 'objets' de ce verbe sont ainsi retenus comme les éventuels arguments de la relation.
Passer à l'étape (2)
- Vérifier que les termes retenus comme arguments ont été détectés comme étant des instances de concepts de UMLS. *Si c'est le cas, passer à l'étape (3), sinon récupérer la relation suivante et revenir à l'étape (1)*
- Vérifier que les arguments retenus vérifient bien la signature de la relation déjà définie dans l'ontologie. *Si c'est le cas, passer à l'étape (4), sinon récupérer la relation suivante et revenir à l'étape (1)*
- Générer une annotation RDF décrivant la relation et l'intégrer dans l'annotation globale du document. Récupérer la relation suivante et revenir à l'étape (1); s'il n'y a plus de relations générer l'annotation globale du document et la stocker dans la base d'annotations et FIN.

Notons que dans la première étape, RASP ne fournit que les informations concernant les termes simples, les relations grammaticales des multi-termes sont déduites automatiquement par MeatAnnot. Par exemple dans la phrase '*KGF is relatively tolerable but causes skin toxicity*', RASP affecte la relation grammaticale 'objet' à '*toxicity*' mais MeatAnnot réaffecte cette même relation au terme '*skin toxicity*' car ce dernier a été détecté comme un terme de UMLS.

2.4.2. Exemple d'exécution

Ci-dessous, nous présentons deux exemples d'exécution de MeatAnnot sur deux phrases extraites de deux documents différents. Le premier exemple (voir Figure 31) montre l'application des heuristiques permettant d'extraire des termes plus complexes ne se trouvant pas dans le métathésaurus de UMLS (dans ce cas c'est l'expression du gène traf6). Le deuxième exemple (Voir Figure 32) montre la capacité de MeatAnnot de traiter des phrases plus complexes (les conjonctions des termes et les verbes dans leurs formes passives).

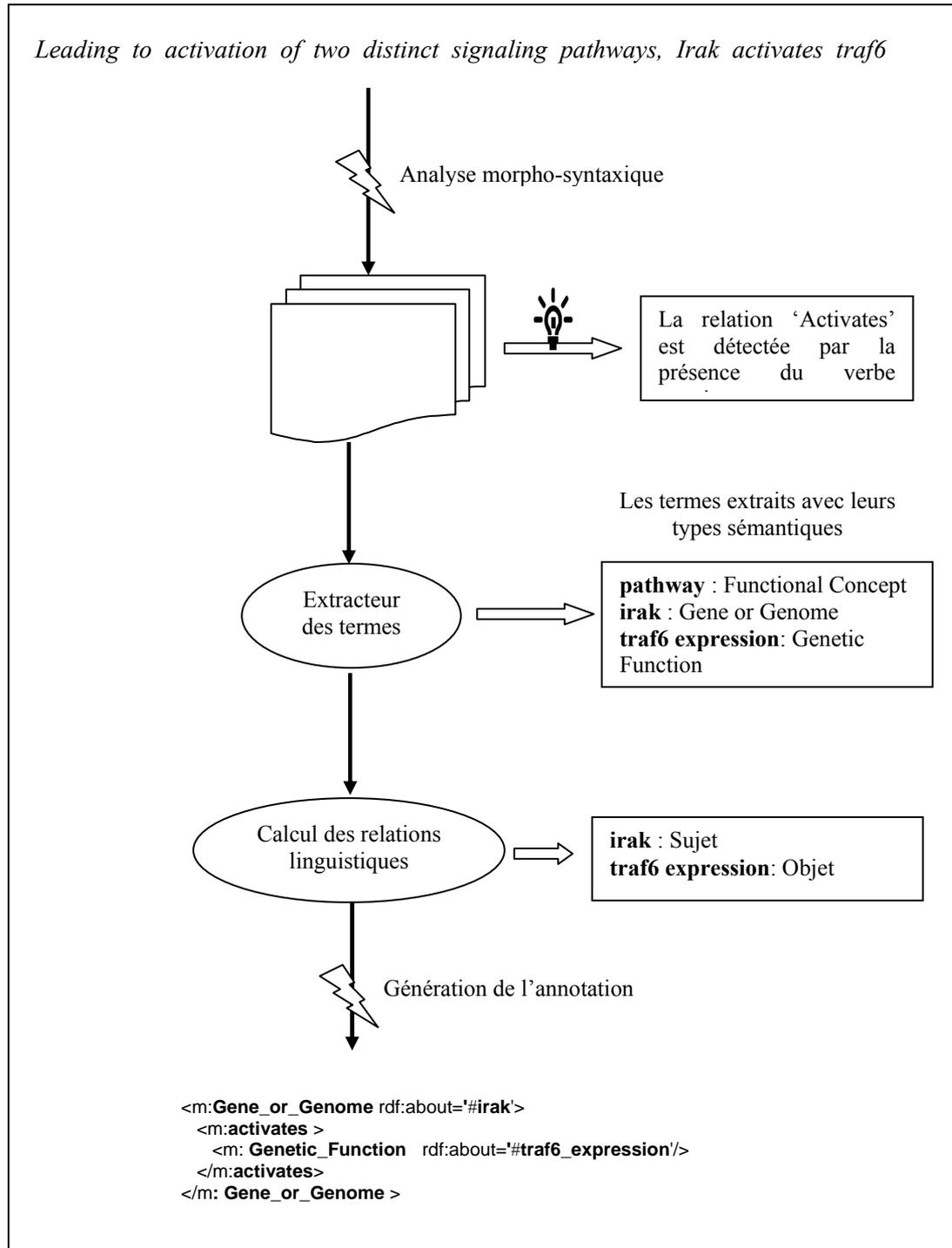


Figure 31 - Exemple1 : génération d'une annotation

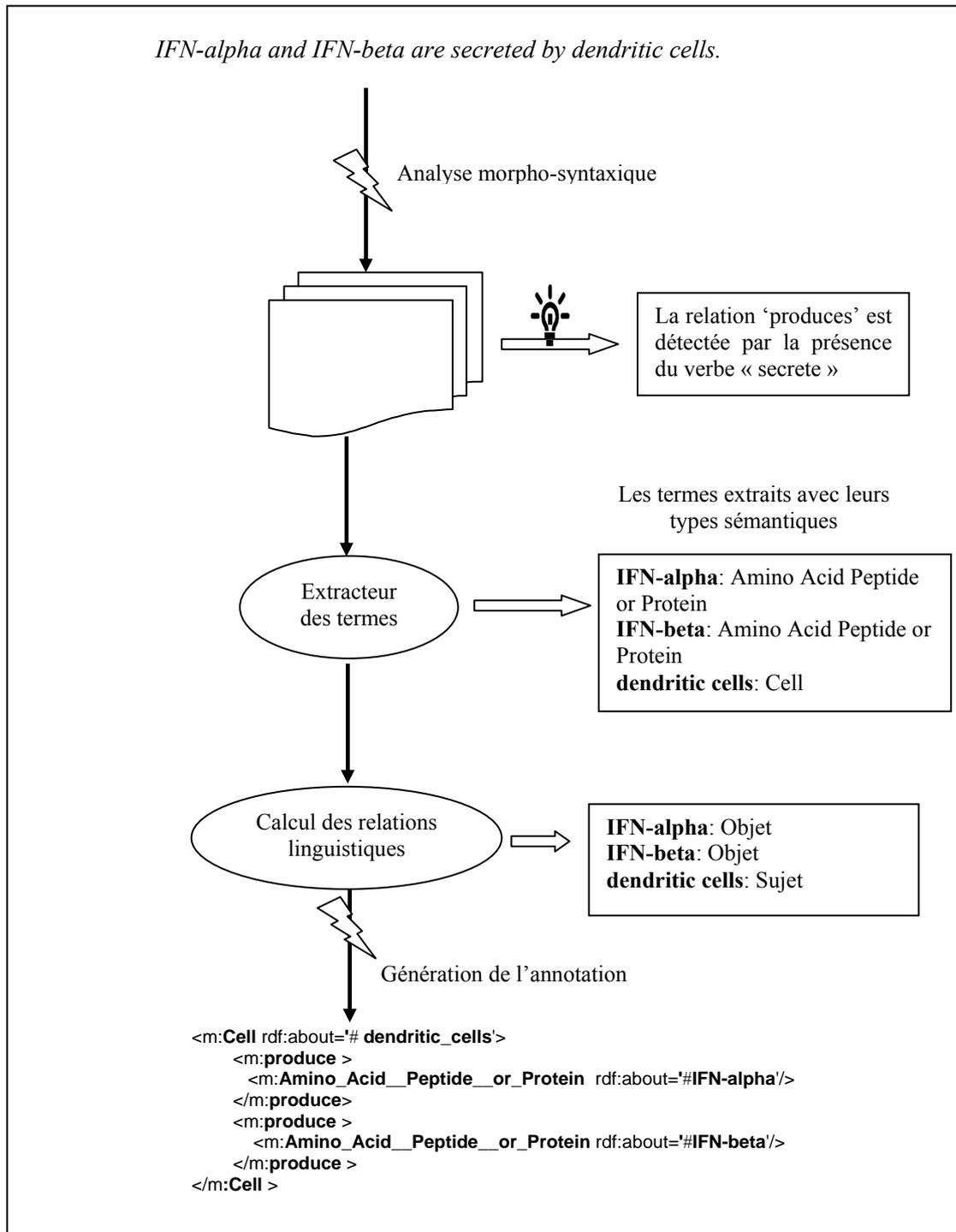


Figure 32 - Exemple2 : génération d'une annotation

3. Validation et évaluation de la méthodologie

Dans cette partie, nous présentons une étude à la fois quantitative et qualitative de la méthodologie de génération d'annotations afin de (i) valider son aspect automatique, et (ii) de

juger la qualité des annotations générées. Nous pensons qu'une phase d'évaluation des annotations en amont est nécessaire vu que la phase de génération est coûteuse et généralement irréversible.

3.1. Le processus de validation

La qualité d'une annotation générée automatiquement à partir des textes, dépend essentiellement de la qualité de la méthode d'extraction d'informations utilisée. Dans notre travail, évaluer la qualité des annotations générées consiste à évaluer :

- La capacité des grammaires de détection à détecter toutes les instances possibles d'une relation sémantique dans le texte ;
- La capacité de l'extracteur des termes à reconnaître toutes les instances de concepts pouvant être liées par ces relations ;
- La capacité à relier les bons termes par la bonne relation, tout en garantissant la cohérence avec le modèle du domaine (i.e. l'ontologie) et avec le sens véhiculé par le texte.

Lors du processus de validation, le deuxième critère a été testé automatiquement sur le corpus de référence GENIA [Kim et al., 2003]. Ce corpus est constitué de 670 résumés d'articles de Medline annotés manuellement et utilisé pour l'apprentissage et l'évaluation des outils d'extraction d'entités biomédicales (en particulier les gènes, les protéines, les acides aminés et les souches de cellules). Cette étape de validation nous a permis de calculer le taux d'erreurs ainsi que le taux de couverture de notre extracteur de termes vu que la dernière phase de la génération de l'annotation repose en partie sur la qualité de cette extraction.

Pour ce faire, nous avons utilisé deux mesures fréquemment utilisées dans l'évaluation des systèmes d'extraction d'informations, à savoir la précision et le rappel :

$$\text{Précision} = \frac{\text{Nombre de termes corrects}}{\text{Nombre total des termes extraits}}$$

$$\text{Rappel} = \frac{\text{Nombre de termes corrects}}{\text{Nombre des termes corrects qui auraient dus être extraits}}$$

La précision (P) est le pourcentage des termes correctement extraits, cela mesure donc l'absence de bruit dans l'extraction. Le rappel (R) est le pourcentage des termes extraits par rapport aux termes qui auraient dû être extraits, cela mesure l'absence de silence dans l'extraction. Une autre métrique peut être calculée en combinant les deux mesures, il s'agit de la F-mesure (dite aussi moyenne harmonique) :

$$\text{F-mesure} = \frac{2PR}{P + R}$$

Pour les deux autres critères d'évaluation définis précédemment, nous avons décidé de les combiner afin d'effectuer une seule validation centrée utilisateur. Dans cette phase de validation, totalement assistée par un environnement informatique, c'est le biologiste qui, soit valide une suggestion de relation entre deux termes (suggestion proposée par MeatAnnot), soit, rajoute une relation présente dans le texte mais qui n'est pas suggérée par l'outil.

Une suggestion est un triplet constitué d'une relation et deux termes (*terme1*, *relation*, *terme2*) extraits à partir d'une phrase. Cette suggestion n'est valide que si (i) la relation désigne bien la relation sémantique, (ii) le verbe caractérisant la relation n'est pas sujet à des modalités qui le nient dans le texte, et (iii) les deux termes sont bien liés par cette relation selon le sens du texte. Notons que les phrases présentées aux biologistes présentent une ou plusieurs suggestions.

Comme lors de la première phase de validation, la précision et le rappel ont été utilisés pour calculer la qualité des suggestions :

$$\text{Précision} = \frac{\text{Nombre de suggestions évaluées correctes}}{\text{Nombre total des suggestions extraites}}$$

$$\text{Rappel} = \frac{\text{Nombre de suggestions évaluées correctes}}{\text{Nombre des suggestions qui auraient dû être extraites}}$$

Au cours de cette phase, nous avons remarqué aussi que quelques suggestions de MeatAnnot ont été considérées correctes mais inutiles pour les biologistes car elles décrivaient soit des connaissances de base, soit des connaissances vagues. Nous avons donc introduit une nouvelle mesure de qualité nommée *utilité* pour mesurer le taux des suggestions utiles.

$$\text{Utilité} = \frac{\text{Nombre de suggestions jugées utiles}}{\text{Nombre des suggestions évaluées correctes}}$$

Notons que la valeur de cette mesure peut être subjective car elle est relative à un point de vue d'un utilisateur ou d'un groupe d'utilisateurs. Dans Meat, les annotations jugées inutiles par un utilisateur sont gardées dans la base d'annotations, une amélioration possible serait de rajouter une métadonnée sur ces annotations (de type *useless_for*) qui permettrait de faire un filtrage sur les réponses aux requêtes posées par cet utilisateur.

La validation des suggestions peut aussi s'intégrer dans le processus général de la génération de l'annotation (via des interfaces dédiées) assurant ainsi la qualité des annotations stockées dans la base de connaissances.

Dans ce qui suit, nous présentons le jeu de test utilisé pour chaque phase de validation ainsi que les valeurs des métriques d'évaluation présentées ci-dessus.

3.2. Résultats de l'évaluation

Pour la validation de l'extracteur de termes, nous avons choisi aléatoirement 27 résumés (142 phrases) du corpus GENIA, sur lesquels nous avons effectué l'extraction. Les résultats ont été ensuite comparés avec les annotations manuelles afin de calculer la précision, le rappel et la F-mesure (voir Tableau 7).

	Annotés	Extraits	Corrects	Rappel	Précision	F-mesure
Gènes et protéines	115	81	72	0.62	0.89	0.73
Souches de cellules	51	45	41	0.8	0.91	0.85

Tableau 7 - Les résultats de l'extracteur de termes sur le corpus GENIA

La cinquième colonne (i.e. le rappel) montre que les termes correctement extraits couvrent 80% des noms de souches de cellules et 62% des gènes et des protéines. La majorité des faux négatifs sont issus de l'insuffisance des variantes de termes dans le métathésaurus et la majorité des faux positifs sont des erreurs de typage dues à l'ambiguïté de quelques termes.

Un autre facteur pouvant expliquer cet écart consiste au fait que les termes annotés dans GENIA sont tous des entités nommées alors que les termes extraits par MeatAnnot sont des syntagmes de longueur maximale. Prenons l'exemple de la phrase : « *Our data suggest that lipoxxygenase metabolites activate ROI formation which then induce **IL-2 expression*** », dans cette phrase '*IL-2*' est annoté comme étant un gène dans GENIA alors que MeatAnnot extrait le syntagme '*IL-2 expression*' comme étant une fonction génétique, ce terme est donc considéré comme un faux négatif alors qu'il est correct.

Cependant, la colonne Précision montre que la qualité des termes extraits est très bonne car MeatAnnot arrive à extraire 89% des gènes et des protéines annotés ainsi que 91% des souches de cellules. Les valeurs de la F-mesure montrent aussi que nous avons un bon compromis entre le rappel et la précision.

Pour la validation des suggestions proposées par MeatAnnot, nous avons choisi au hasard un corpus de 20 documents (2751 phrases) parmi les documents proposés par les biologistes et nous avons présenté les suggestions aux biologistes à travers une interface de validation pour qu'ils évaluent leur qualité. Cette interface a été conçue de manière à présenter les annotations dans un format compréhensible (textuel) pour les biologistes, qui ne sont pas spécialistes des langages de représentation des connaissances tel que RDF/S (voir Figure 33).

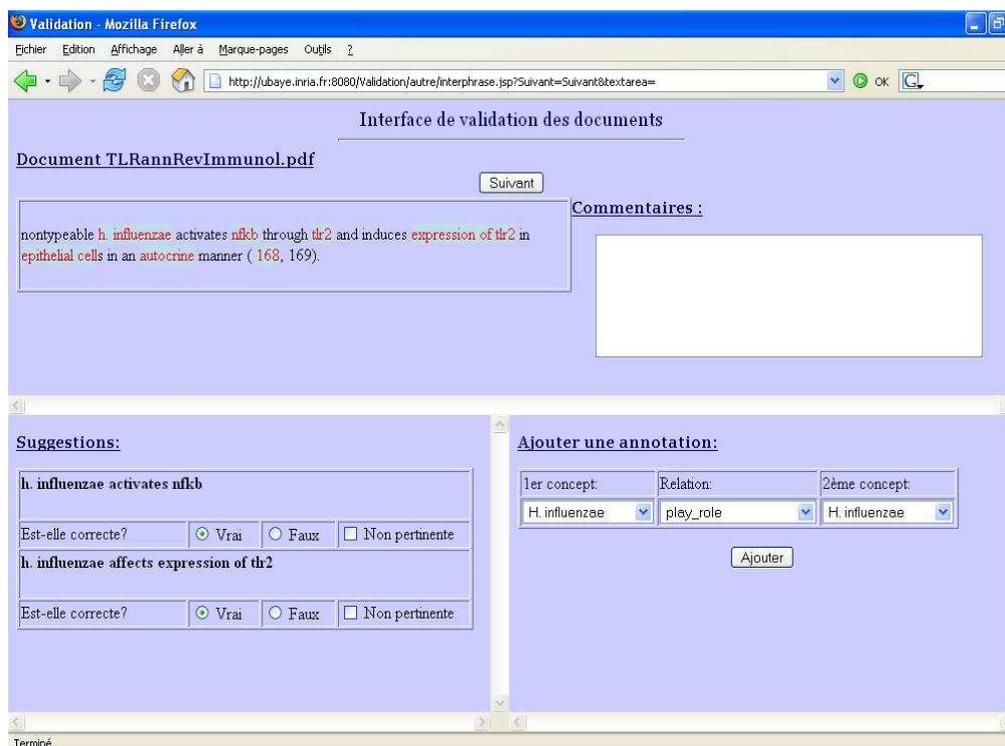


Figure 33 - L'interface de validation des suggestions

Les résultats de cette évaluation sont présentés dans le tableau suivant :

	Suggestions	correctes	Utiles	manquantes	Rappel	Précision	Utilité
Résultat	509	426	399	274	0.608	0.836	0.936

Tableau 8 - Résultats de l'évaluation des suggestions

La seconde colonne du Tableau 8 mentionne le nombre de relations extraites correctement à partir des textes. La différence avec le nombre de suggestions proposées par MeatAnnot est due principalement aux erreurs générées par les outils de TALN (catégorie grammaticale ou relation linguistique incorrecte) et aux termes manquants du thésaurus de UMLS (sujet ou objet non trouvé dans UMLS). Néanmoins, nous avons obtenu une bonne précision puisque 83% des suggestions sont correctes.

La troisième colonne décrit le nombre de relations existantes dans le texte mais que MeatAnnot n'a pas pu extraire. Ce silence est dû dans certains cas aux erreurs générées par les outils de TALN mais principalement aux relations déduites par les biologistes en lisant la phrase mais qui ne peuvent pas être générées automatiquement.

Un exemple de relation non extraite :

“Upon interferon-gamma induction, after viral infection for example, a regulator of the proteasome, PA28 plays a role in antigen processing.”

Dans cet exemple, MeatAnnot extrait automatiquement la relation “PA28 has_a_role_in antigen processing”, mais le biologiste en lisant la phrase peut déduire, en utilisant ses connaissances implicites, une autre relation qui est “interferon-gamma have_effect PA28”.

Enfin, MeatAnnot a une bonne *utilité* puisque 94% des suggestions correctes sont considérées utiles par les biologistes.

4. Discussion et conclusion

Nous avons présenté une méthodologie pour la génération d’annotations sémantiques basées sur une ontologie déjà existante. Ces annotations sont basées non seulement sur les instances des concepts mais en plus sur les instances des relations. Ces instances de relations peuvent relier les différents concepts de l’ontologie et non seulement les gènes et les protéines, comme c’est le cas dans la majorité des travaux sur la fouille des textes biologiques. Les annotations générées par MeatAnnot sont ainsi plus riches et décrivent rigoureusement le contenu sémantique du texte.

Par ailleurs, les documents utilisés par les biologistes (généralement des articles de revue) décrivent essentiellement des connaissances génériques et qui doivent être annotés par des termes génériques. Par exemple quand un article parle des poumons, les connaissances contenues dans cet article concernent les poumons en général et non pas les poumons d’une personne en particulier. Ce constat justifie notre choix sur l’utilisation des termes du métathésaurus comme étant des instances des concepts de l’ontologie.

Dans ce travail, nous avons aussi choisi d’associer à chaque document une annotation le décrivant, par opposition à l’approche qui consiste à créer une annotation pour chaque élément du domaine (par exemple un gène) tout en gardant un lien sur les documents. Ce choix est essentiellement motivé par le souhait de faciliter la gestion et le contrôle d’accès aux annotations. En effet, l’approche que nous avons choisie nous permet de filtrer plus facilement les annotations par rapport à leur provenance en utilisant les métadonnées que nous avons rajouté.

Une des perspectives pour MeatAnnot serait de rajouter la prise en compte des relations ayant une forme nominale (par exemple dans le cas de UMLS, les relations *degree_of*, *ingredient_of*). Car bien que, nous puissions toujours écrire des grammaires de détection pour ces relations, nous ne nous disposons pas de techniques pour la détermination de leurs arguments.

Après avoir présenté une méthodologie et un système permettant d’alimenter la base d’annotations de la mémoire d’expériences, nous présentons dans le chapitre suivant les mécanismes adoptées pour l’exploitation de ces annotations.

Chapitre 5 - Exploitation des annotations :

MeatSearch

1. Introduction

Comme nous l'avons souligné précédemment, notre approche se situe dans le cadre des travaux de l'équipe Acacia sur l'exploitation des techniques du Web Sémantique pour la construction de mémoire (dans notre cas une mémoire d'expériences). Cette construction se base essentiellement sur trois points :

- La définition et la construction d'une (ou plusieurs) ontologie(s) décrivant le vocabulaire partagé dans la communauté en question ;
- L'énumération et l'annotation des ressources du domaine tout en se basant sur l'ontologie ;
- L'exploitation des annotations construites en offrant un environnement permettant l'accès aux ressources annotées.

Ayant étudié les deux premiers points dans les chapitres précédents, dans ce chapitre nous présentons la démarche adoptée pour faciliter la recherche d'informations dans la mémoire d'expériences.

2. Vue d'ensemble

2.1. Motivations

Nous rappelons ici que la phase de validation et d'interprétation d'une expérience se base essentiellement sur une phase de recherche et d'extraction d'informations à partir de sources hétérogènes telles que les articles scientifiques traitant le phénomène étudié, les notes des biologistes et les expériences antérieures stockées dans des bases de données.

Recherche d'informations dans les bases documentaires :

Rechercher des informations dans des bases documentaires consiste à trouver des documents répondant à une requête envoyée par un utilisateur à un moteur de recherche. Ces moteurs de recherche qui peuvent être spécialisés (i.e. Pubmed en biologie) ou génériques (i.e. Google) se basent essentiellement sur des mots clés et présentent quelques défauts majeurs pour les biologistes :

- Des dizaines voire des centaines de documents peuvent répondre positivement à une requête, ce qui nécessite un effort considérable de la part des biologistes pour trier, explorer et comprendre les résultats obtenus;
- Le silence (resp. le bruit) dans les résultats dû par exemple à l'absence d'un terme et à la présence non détectée de son synonyme (resp. dû par exemple à l'ambiguïté du terme utilisé);
- L'obligation de rechercher et de relire le même article pour retrouver une information intéressante dont on a pas gardé une trace (par exemple une interaction spécifique entre deux gènes).

Recherche d'informations stockées dans les bases de données :

Avec l'apparition de nouveaux langages pour le web (PHP, JSP...), l'accès aux bases de données locales et distantes est devenu de plus en plus facile, et ce grâce à des interfaces dédiées reposant sur des requêtes types et permettant ainsi aux biologistes de rechercher les informations qui les intéressent dans ces bases de données. Cependant, le raisonnement sur les résultats des différentes expériences stockées représente toujours une tâche lourde et coûteuse, et ce pour deux raisons principales :

- Les descriptions et les interprétations des expériences sont stockées généralement sous une forme textuelle, ce qui nous renvoie à un problème similaire à celui de la recherche d'informations dans les documents textuels.
- Le lien entre les nouvelles connaissances obtenues lors de l'expérience et les connaissances implicites (celles des biologistes ou celles déduites lors de la lecture d'un article) est quasi-inexistant.

Dans le cadre de MEAT, l'utilisation des ontologies et des technologies du Web Sémantique nous a permis, dans un premier temps, de construire une mémoire d'expériences intégrant les différentes sources d'informations et de l'alimenter avec des annotations décrivant le contenu sémantique de ces sources. Dans cette partie, nous pensons que l'exploitation de ces annotations nous permettra de remédier aux problèmes de recherche d'informations cités ci-dessus. En effet, notre objectif rejoint l'objectif de [Berners Lee, 2001] pour la recherche d'informations dans le Web Sémantique mais à un niveau local qui est la mémoire d'expériences. Un moteur de recherche sémantique se basant sur les ontologies du domaine et des annotations sémantiques permettra (i) de surmonter l'hétérogénéité des ressources dans la mémoire, (ii) de répondre aussi bien à des requêtes génériques qu'à des requêtes plus fines croisant le contenu sémantique de chaque ressource, et (iii) de raisonner sur les connaissances contenues dans les annotations.

Dans ce qui suit, nous présentons le prototype MeatSearch développé pour naviguer dans la mémoire d'expériences et pour répondre aux requêtes des utilisateurs. La Figure 34 montre une vue d'ensemble de ce prototype qui se base le moteur de recherche sémantique Corese (décrit ci-dessous).

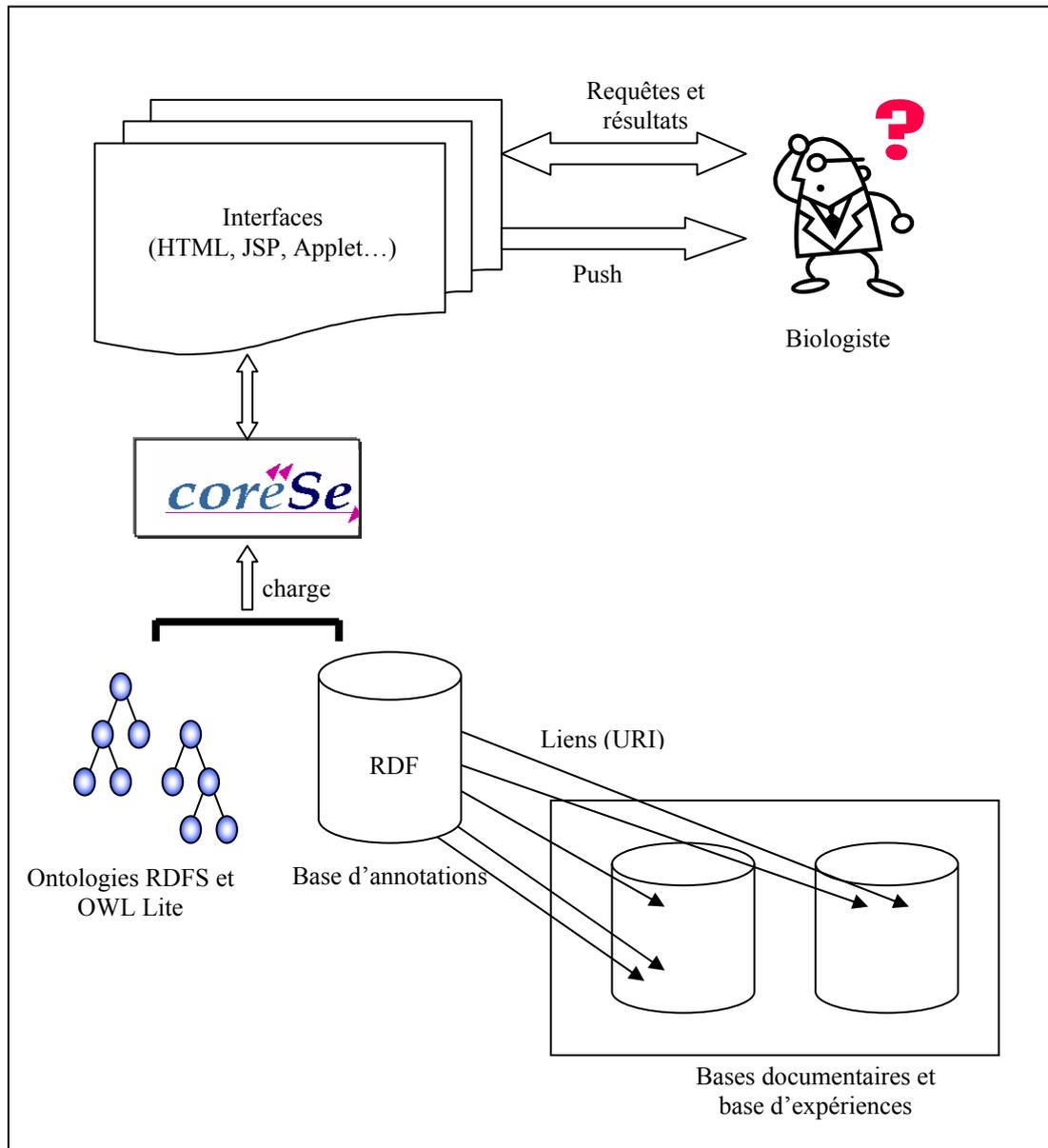


Figure 34 - Vue d'ensemble de MeatSearch

2.2. Corese (Conceptual Resource Search Engine)

Ce moteur de recherche sémantique proposé par l'équipe ACACIA [Corby et al., 2004,2006] permet (a) d'explorer et interroger les ontologies, (b) d'utiliser les annotations sémantiques pour la recherche d'information dans la mémoire et (c) de créer des interfaces d'interrogation et de visualisation des résultats en offrant un mécanisme de génération d'interfaces dédiées.

Corese implémente un moteur RDF(S) basé sur le formalisme des graphes conceptuels (GC) [Sowa, 1984]. Il traduit les classes et les relations RDFS en types de concept et en types de relation et les annotations RDF en base de GC (voir Tableau 9). Il intègre aussi un interpréteur

du langage de requêtes SPARQL³⁹ dédié à RDF. Ce langage en préparation au W3C permet à partir d'une base RDF de:

- Extraire de l'information : URI, Literal, Datatyped Literal;
- Extraire un sous graphe RDF;
- Construire un graphe résultat.

La Figure 35 montre la syntaxe d'une requête ainsi que le format d'un résultat SPARQL.

```
Une requête:
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?book ?title
WHERE { ?book dc:title ?title }
```

```
Le résultat:
<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  <head>
    <variable name="book"/>
    <variable name="title"/>
  </head>
  <results distinct="false" ordered="false">
    <result>
      <binding name="book"><uri>http://example.org/book/book1</uri></binding>
      <binding name="title">SPARQL</binding>
    </result>
  </results>
</sparql>
```

Figure 35 - Exemple d'une requête et un résultat SPARQL

Pour le calcul des résultats, Corese utilise l'opérateur de projection des graphes conceptuels qui permet d'extraire un graphe conceptuel résultat à partir d'une requête. Ce graphe résultat est traduit en un langage standard (RDF(S) ou XML) permettant ainsi sa réutilisation par un autre programme ou sa présentation à l'utilisateur.

³⁹ <http://www.w3.org/TR/rdf-sparql-query/>

RDF/RDFS	GC	
Rdfs:Class	Type de concept	Représentation des ontologies
Rdf:Property	Type de relation	
Rdfs:domain, rdfs:range	Signature des relations	
Resource	Concept	Représentation des annotations
resource Anonyme	Concept générique	
propriété	Relation	

Tableau 9– Correspondance entre RDFS/RDF et GC

Afin d'affiner les résultats et les rendre plus pertinents, CORESE offre la possibilité de décrire des règles de production qui permettent de compléter et enrichir la base d'annotations. Ces règles sont chargées et le moteur les exploite sur le graphe RDF global pour effectuer des inférences enrichissant la base d'annotations. Ces règles suivent la forme If condition THEN conclusion, où *condition* représente une requête à vérifier et *conclusion* représente une ou plusieurs nouvelles assertions RDF à rajouter dans le graphe global (la condition et la conclusion sont écrites en SPARQL).

Enfin, CORESE possède une fonction de recherche approchée qui permet de fournir une réponse proche à une requête dans le cas où aucune réponse n'existe. Cette fonction utilise une fonction de calcul de distance entre les concepts d'une ontologie en se basant sur le calcul de la longueur des chemins entre les concepts et en faisant varier cette longueur avec la profondeur ; ainsi, deux concepts frères de profondeur n+1 sont plus proches que deux concepts frères de profondeur n.

la Figure 36 qui décrit l'architecture globale de Corese.

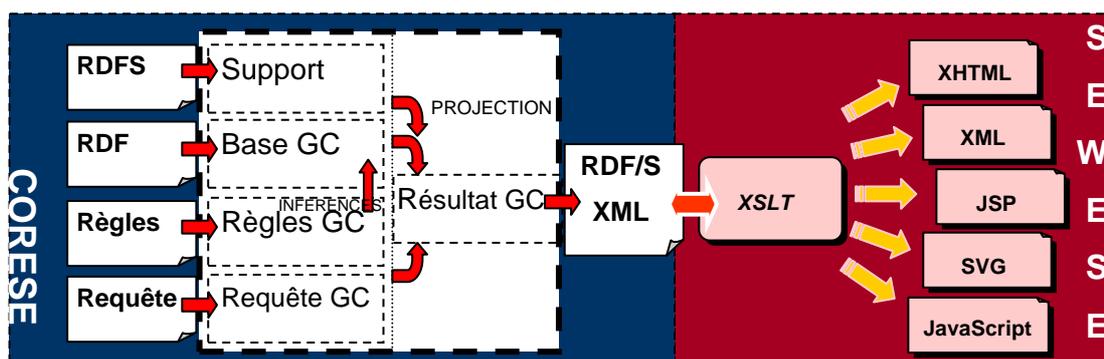


Figure 36 - Architecture de CORESE

Ce logiciel est disponible⁴⁰ en bibliothèque de fonctions (API) utilisable dans les applications du web sémantique. Il peut également être intégré dans un serveur Web sémantique (SEWEE) permettant la création dynamique d'applications Web à partir d'une base de connaissances

⁴⁰ <http://www-sop.inria.fr/acacia/soft/corese/>

représentées en RDF tout en se basant sur des formalismes de transformations (XSLT) et sur des technologies standards du Web (XML, JSP...).

3. La recherche d'informations dans Meat

Comme nous l'avons souligné précédemment, la recherche d'informations dans Meat se base sur le prototype MeatSearch, qui lui-même exploite Corese afin de permettre aux utilisateurs de :

- Naviguer dans la base d'annotations en tenant compte de la structure hiérarchique des ontologies. En effet, Corese utilise les liens de subsumption entre les concepts et les relations de l'ontologie afin d'enrichir les résultats d'une requête.
- Visualiser les résultats sous des formes compréhensibles par les biologistes (graphes, listes...).
- Ajouter des règles qui complètent la base d'annotations en utilisant le langage de règles de Corese. Ces règles sont proposées par les biologistes en se basant sur leurs connaissances implicites du domaine (voir l'exemple § 3.2)
- Raisonner sur la totalité de la base d'annotations construites à partir de sources différentes et hétérogènes (articles, base d'expériences...): cela permet aux biologistes de déduire des connaissances à la fois implicites et explicites sur un gène.
- Utiliser différents niveaux d'accès (administrateur, public, groupe...) et les métadonnées sur les annotations afin d'avoir différentes vues sur les connaissances contenues dans les annotations.

Dans ce qui suit, nous présentons des exemples des différentes fonctionnalités de MeatSearch.

3.1. Exemples d'interfaces

3.1.1. Exemple A : Interface de recherche libre

Ce premier exemple montre une interface générique de MeatSearch permettant de trouver des ressources (articles ou expériences) en se basant sur les concepts et les relations de l'ontologie. Ces interfaces peuvent être générées automatiquement en se basant sur la structure hiérarchique de l'ontologie. Dans cet exemple, l'utilisateur cherche à trouver toutes les interactions que peut avoir un gène nommé 'Annexin_A2' avec n'importe quelle entité biologique (voir Figure 37).

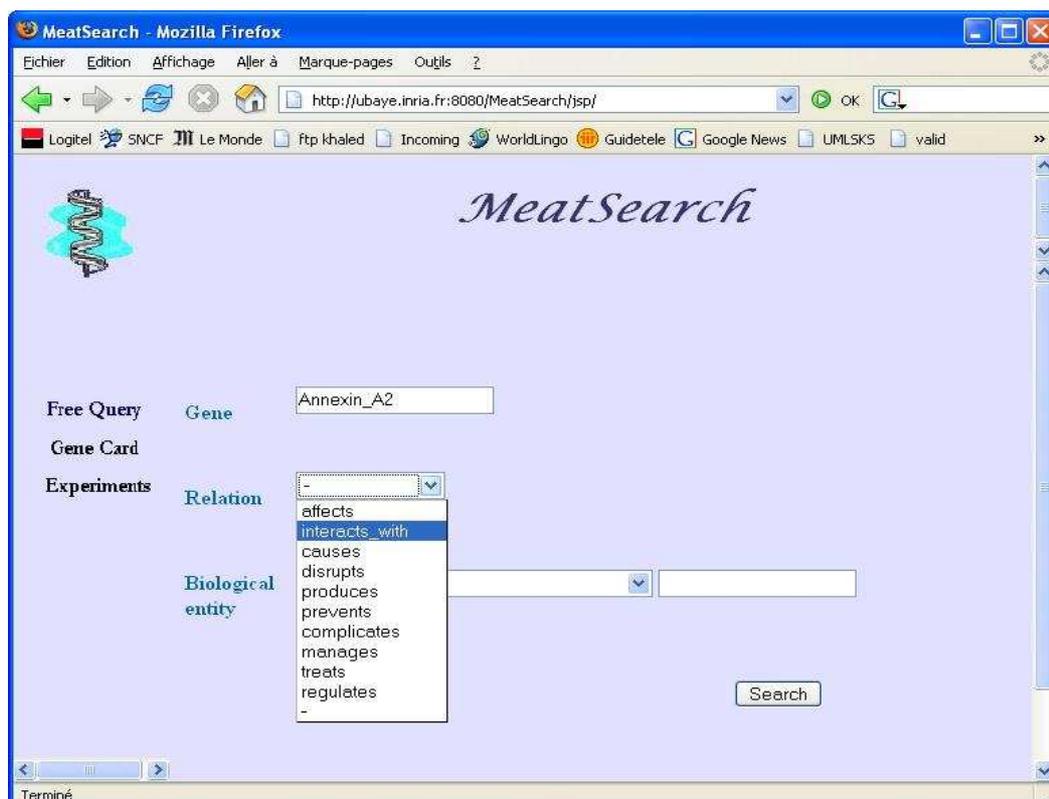


Figure 37 - Exemple d'interface de recherche libre

La réponse de MeatSearch à une telle requête peut être présentée sous plusieurs formes : un ensemble de documents, un ensemble d'expériences ou un graphe représentant les interactions demandées.

La Figure 38 montre le résultat de la requête précédente. Cette requête a été exprimée automatiquement par MeatSearch en SPARQL et envoyée à Corese, le résultat XML obtenu est transformé à l'aide d'une feuille de style XSLT et envoyé à son tour à une Applet Java qui permet de dessiner un graphe d'interactions. Le résultat obtenu représente une vue sur un sous ensemble des connaissances contenues dans la base d'annotations. Ces connaissances peuvent être extraites de plusieurs articles (deux en ce qui concerne cet exemple).

MeatSearch joue aussi le rôle d'un 'concordancier' en offrant un lien hypertexte entre chaque interaction dans le graphe (en cliquant sur l'arête) et le(s) texte(s) d'où elle a été extraite (voir Figure 39). Cette fonctionnalité permet, d'une part, de recadrer l'annotation par rapport à son contexte d'origine, et d'autre part, de renvoyer l'utilisateur à un ensemble de documents lui permettant de creuser la nouvelle connaissance acquise.

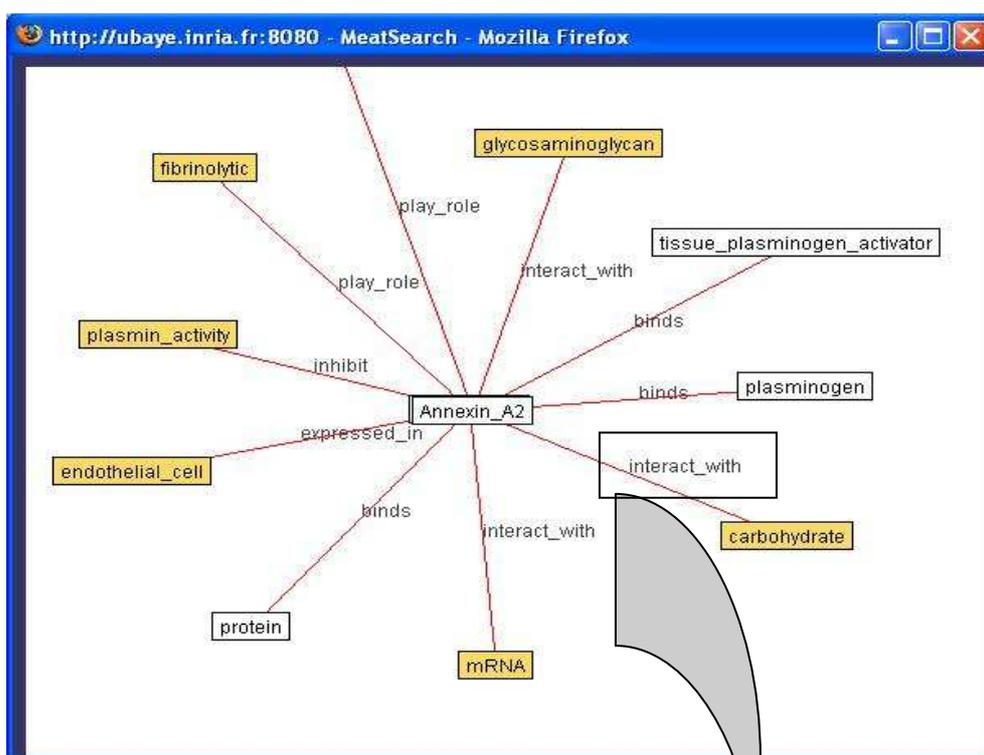


Figure 38 - Résultat d'une recherche sur un gène

The screenshot shows a web browser window with the address bar displaying 'http://ubaye.inria.fr:8080 - Mozilla Firefox'. The main content area displays the following information:
Document :
http://www-sop.inria.fr/meat/Gerke_2002.xml
Sentence :
Annexins A2, A4, A5, A6, and the *Caenorhabditis elegans* protein annexin B7 interact with carbohydrates, in particular *glycosaminoglycans*, and in some cases the binding sites have been mapped to certain regions within the respective annexin molecule.
[Return To graph](#)

Figure 39 - Lien entre le résultat d'une requête et la ressource annotée

3.1.2. Exemple B : Recherche d'expériences

La Figure 40 montre une interface de recherche d'expériences, cette recherche se base, soit sur des mots clés décrivant l'expérience, soit sur les gènes mis en cause dans l'expérience, soit

sur les personnes ayant effectué l'expérience. Dans l'exemple ci-dessous, l'utilisateur cherche à trouver une expérience traitant le foie et qui a été réalisée par un biologiste nommé 'Bernard'.

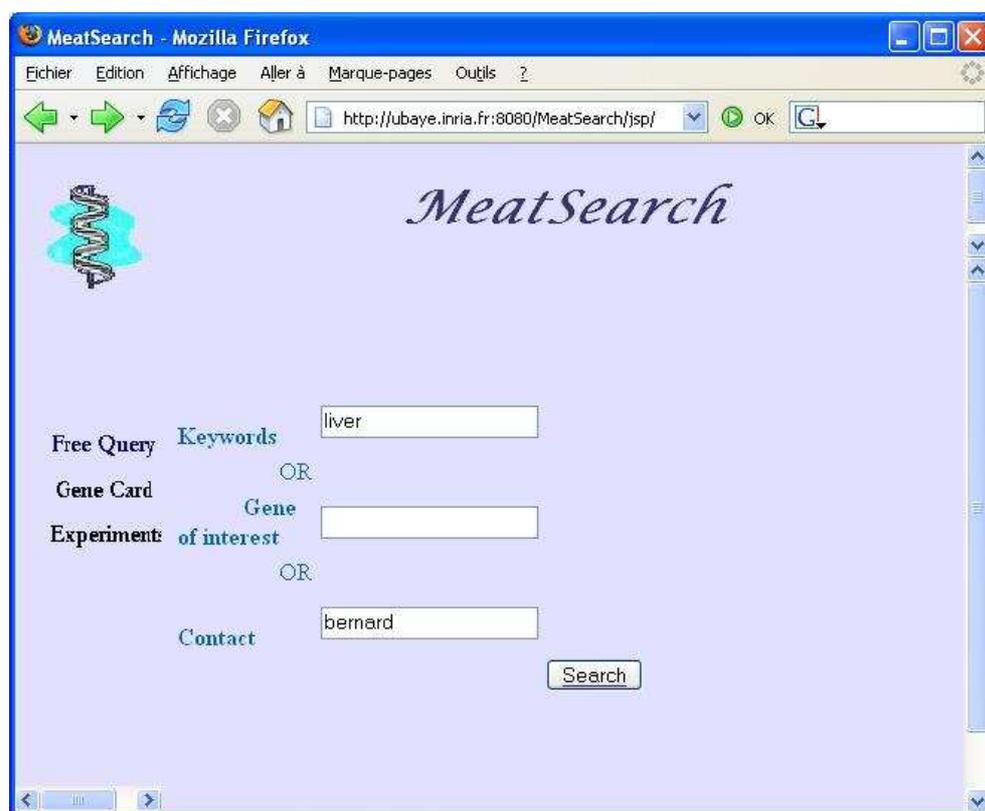


Figure 40 - Interface de recherche d'expériences

La réponse de MeatSearch à cette requête est présentée dans la Figure 41. Cette réponse contient une description de l'expérience, les gènes mis en cause lors de la manipulation, les personnes ayant effectué l'expérience et un ensemble d'articles qui ont un lien avec cette expérience.

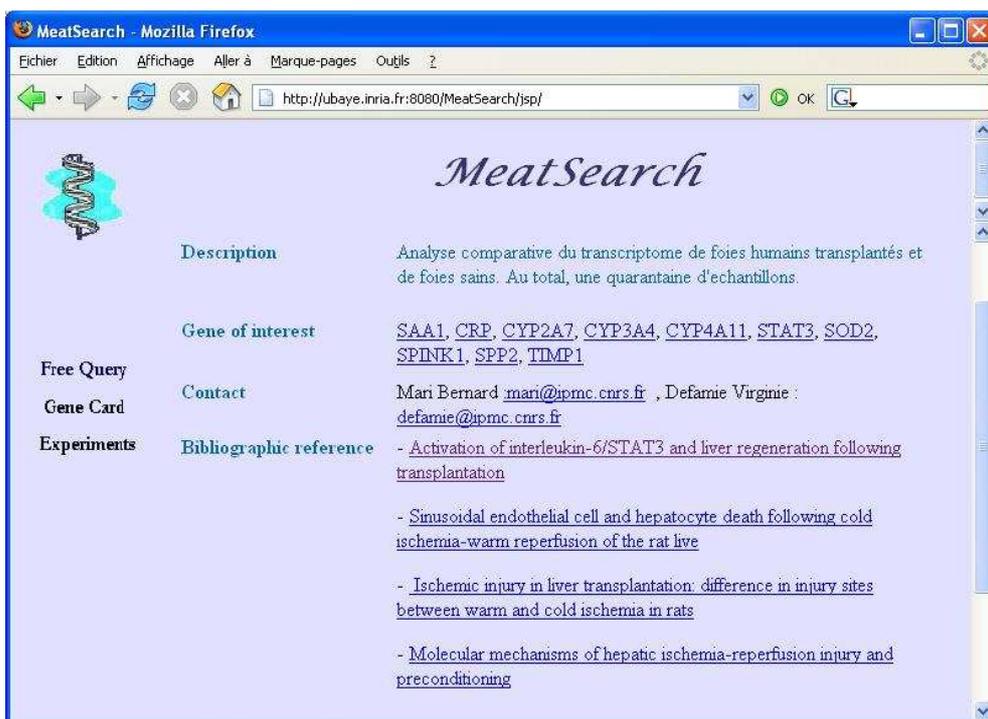


Figure 41 - Résultat d'une recherche d'expériences

Notons ici que chaque gène possède un lien hypertexte formulant une requête qui sélectionne toutes les informations le concernant dans la base d'annotations.

3.1.3. Exemple C : Le panier de gènes

Le dernier exemple montre une interface que nous avons nommé 'Panier de gènes'. Cette interface peut être utilisée tout au long du déroulement d'une expérience, elle permet de représenter un ensemble de gènes auxquels le biologiste s'intéresse ainsi qu'un lien avec la base d'annotations lui permettant de comprendre des phénomènes apparus lors de la manipulation.



Figure 42 - Exemple de l'interface du panier de gènes

Nous avons aussi rajouté une fonctionnalité à cette interface permettant de réaliser une classification des gènes sélectionnés, en essayant de calculer une similarité sémantique entre les gènes. Ce calcul de similarité se base sur le calcul de distance entre les ensembles de termes de la Gene Ontology (GO) caractérisant chaque gène.

En effet, la plupart des gènes ont été annotés par un ensemble de termes de GO les caractérisant, ce travail a été réalisé par plusieurs équipes de recherches en biologie et est disponible sur le site de GO⁴¹.

Notre idée consiste à dire que si deux gènes sont annotés par des termes proches par rapport à la structure de la Gene Ontology, alors ces gènes possèdent quelque chose de commun (même fonction génétique, une dépendance moléculaire...). Nous pensons qu'une telle classification peut aider à interpréter une classification déduite des résultats trouvés lors d'une expérience. Nous avons donc proposé une fonction de calcul de distance sémantique qui se base sur la fonction de distance entre concepts de l'ontologie [Gandon et al., 2005b] implémentée dans Corese:

$$\text{distance}(G_i, G_j) = \text{moyenne}(\text{dist}_c(C_{ki}, C_{ej}))$$

$\text{dist}_c(C_{ki}, C_{ej})$ est la distance dans GO (calculée par Corese) entre le concept C_k annotant le gène G_i et le concept C_e annotant le gène G_j :

⁴¹ <http://www.geneontology.org/GO.current.annotations.shtml>

$$\text{dist}_c(c_1, c_2) = \frac{1}{2^{\text{depth}(\text{LCST}(c_1, c_2)) - 2}} - \frac{1}{2^{\text{depth}(c_1) - 1}} - \frac{1}{2^{\text{depth}(c_2) - 1}}$$

où $\text{depth}(c)$ est la profondeur du concept c par rapport à la racine de l'ontologie et $\text{LCST}(c_1, c_2)$ est le plus petit ancêtre commun aux concepts c_1, c_2 dans l'ontologie.

Une fois la matrice des distances calculée, MeatSearch exécute un algorithme standard de classification hiérarchique (type Average Linkage [Seifoddini et Wolfe, 1986]) sur les gènes. Le résultat de la classification est ensuite représenté sous la forme d'un dendrogramme et affiché à travers une Applet Java. Un exemple de classification est montré dans la Figure 43.

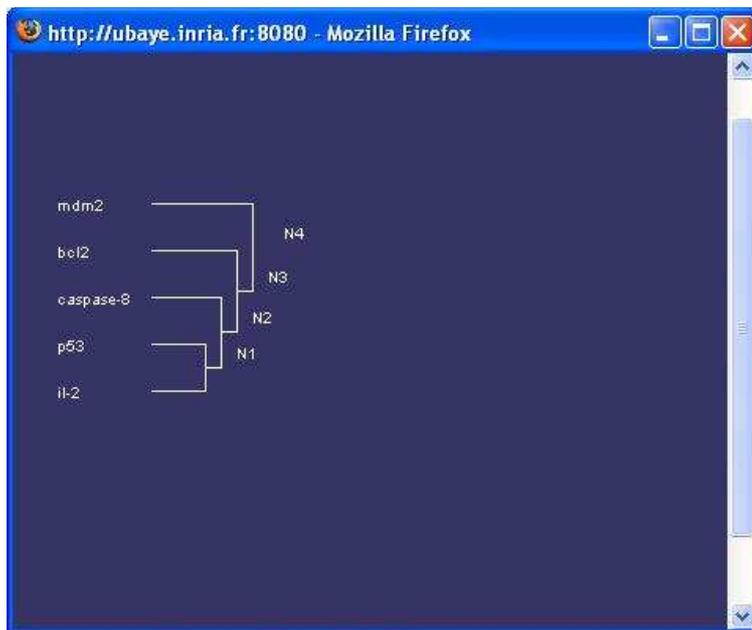


Figure 43 - Résultat d'une classification de gènes

L'exemple ci-dessus montre par exemple qu'il est probable (resp. peu probable) que le gène 'il-2' ait des similarités avec le gène 'p53' (resp. 'mdm2').

3.2. Exploitation des règles

Nous rappelons que Corese offre un langage de règles [Corby et al., 2006] permettant de déduire de nouvelles connaissances à partir de celles qui existent. Les règles de production sont appliquées sur les annotations dans le but de rajouter de nouvelles informations qui peuvent réduire le silence lors d'une phase de recherche d'informations.

Après des discussions avec nos partenaires, nous avons intégré progressivement quelques règles dédiées aux expériences des puces à ADN. Une règle permet ainsi de partager une connaissance acquise par un biologiste lors de son parcours avec les autres utilisateurs de la mémoire d'expérience. Les règles rajoutées nous ont été fournies par les biologistes sous une forme informelle (textuelle) que nous avons ensuite traduite manuellement dans le langage de règles de Corese (basé sur SPARQL).

Un exemple de règle :

“Pour chaque récepteur qui active une fonction moléculaire, si cette fonction joue un rôle dans le fonctionnement de l’organisme alors le récepteur joue le même rôle”

Cette règle est exprimée comme suit:

```
IF ?r rdf:type m:Receptor
    ?r m:activates ?mf
    ?mf rdf:type m:Molecular_Function
    ?mf m:play_role ?of
    ?of rdf:type m:Organism_Function
THEN
    ?r m:play_role ?of
```

Ces règles, une fois chargées par Corese, permettent d’enrichir la base d’annotation et peuvent être exploitées par MeatSearch afin d’améliorer la qualité de la recherche d’information en terme de précision et de rappel.

3.3. Exploitation des métadonnées

Nous rappelons ici l’intérêt des métadonnées que nous avons proposé de rajouter sur les annotations (Chapitre 3, §5.1). En effet ces annotations permettent d’intégrer de nouvelles informations concernant :

- La source de la ressource : le biologiste qui a fourni l’article à annoter ou celui qui a fait l’expérimentation.
- La source de l’annotation : si elle a été générée automatiquement par MeatAnnot ou ajoutée/validée par un biologiste.
- Le thème général de l’annotation : les biologistes peuvent avoir différents centres d’intérêt à propos de la même expérience.
- Le thème général d’une annotation constitue une instance d’un concept de l’ontologie UMLS (par exemple une maladie, une partie du corps...).

Les informations offertes par les métadonnées une fois combinées peuvent nous fournir des annotations contextuelles et multi-points de vue. L’annotation ci-dessous (voir Figure 44) décrit un article fourni par un biologiste nommé Pascal et parlant du développement des poumons.

```
<do:paper rdf:about='http://www-sop.inria.fr/acacia/meat/lungrepair.pdf'>
  <do:providedBy>Pascal</do:providedBy >
  <do:relatedTo >
    <m: Organ_or_Tissue_Function rdf:about='lung_development'/'>
  </do:relatedTo >
  ....Annotation...
  <do:generatedBy>MeatAnnot</do:generatedBy>
  <do:validatedBy>Pascal</do:validatedBy>
  ...Annotation...
</do:paper>
```

Figure 44 - Exemple de métadonnées intégrées dans une annotation d'un document

MeatSearch peut utiliser ces métadonnées pour proposer différentes vues sur la base d'annotations qui sont liées aux utilisateurs (source d'annotation), au contexte (thème général de l'annotation) et à la méthode de génération de l'annotation (automatique ou manuelle). D'autre part, l'interrogation de ces métadonnées par Corese nous permet d'avoir plus d'informations sur les annotations et de vérifier leurs cohérences afin de les valider.

Ayant aussi utilisé l'ontologie pour définir le thème général d'une annotation, MeatSearch pourra effectuer des filtrages sur les annotations en utilisant les raisonnements de Corese sur cette métadonnée. Prenons l'exemple d'un biologiste qui demande à avoir tous les articles ayant un thème général une instance du concept 'Anatomical Abnormality', en utilisant la structure de l'ontologie, Corese pourra renvoyer non seulement les articles annotés par ce concept mais aussi les documents annotés par 'Congenital Abnormality' et 'Acquired Abnormality' représentés comme des sous classes de 'Anatomical Abnormality' dans l'ontologie.

Dans cette même optique, [Bringay et al., 2005] proposent l'utilisation de métadonnées sur des annotations informelles dans le cadre de gestion des connaissances du dossier patient dans le but de faciliter la manipulation des documents électroniques annotés par les praticiens.

4. Conclusion

Dans ce chapitre, nous avons présenté le prototype MeatSearch permettant d'interroger la mémoire d'expériences. A des fins de réutilisation et d'évolution, cet outil se base essentiellement sur des standards recommandés par le W3C (SPARQL implémenté dans Corese, RDF, XSLT...).

MeatSearch offre en plus de la fonctionnalité de recherche d'informations, un mécanisme de raisonnement basé sur l'ontologie et qui permet de découvrir de nouvelles connaissances :

- l'utilisation de la structure hiérarchique de l'ontologie pour l'expansion des requêtes;
- l'utilisation des règles et des métadonnées;
- la déduction des liens entre les différentes ressources (articles, expériences) et entre les annotations.

L'utilisation des techniques de TALN implémentées dans MeatAnnot pour l'extraction d'information nous a permis de proposer une nouvelle approche pour la recherche documentaire. En effet, un biologiste envoyant une requête à un moteur de recherche classique est souvent confronté à une liste de documents comme résultat, dans ce même cas MeatSearch offre la possibilité d'avoir une vue sur toutes les informations répondant à la requête avec un lien sur les documents concernés.

Une extension envisageable à MeatSearch consiste à fournir aux biologistes une interface d'édition et de génération automatique de règles. Cette interface peut être utile pour des biologistes qui ne sont pas très habitués à l'utilisation de langages tel que SPARQL.

Une autre évolution possible consiste à incorporer le calcul automatique de chemins entre deux entités utilisé par [Gandon et al., 2005a] pour l'agrégation des services Web sémantiques. Dans notre cas, par exemple, le biologiste peut déduire un lien entre un gène et une maladie en étudiant le chemin calculé automatiquement à l'aide d'une requête posée sur la base d'annotations, entre ces deux entités (Voir Figure 45).

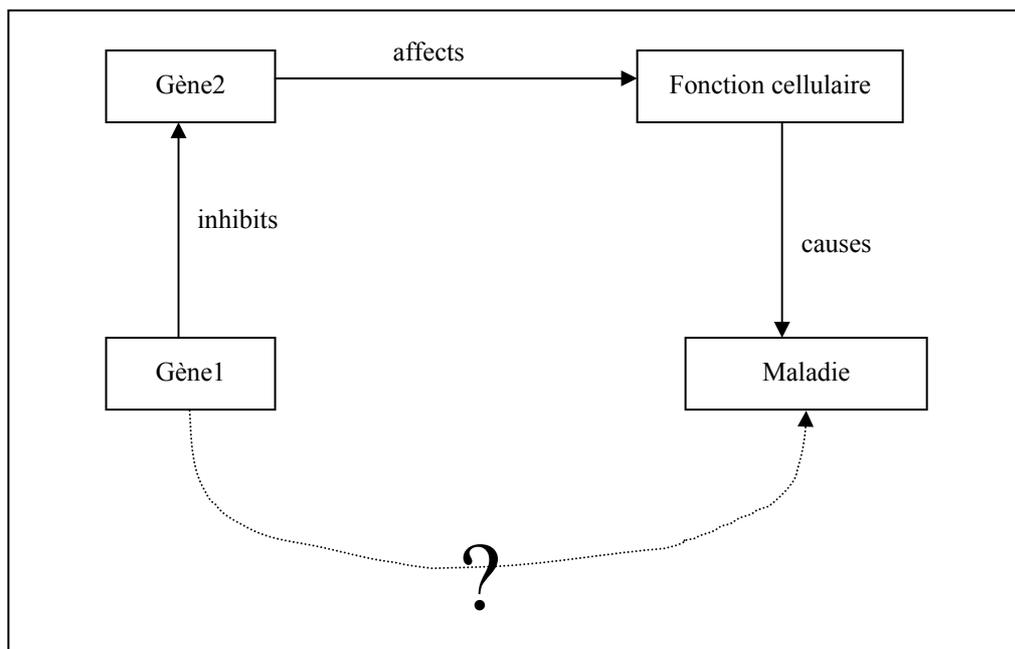


Figure 45 - Calcul de chemin entre deux entités

Dans le chapitre suivant, nous récapitulons l'ensemble de nos contributions et nous présentons quelques perspectives concernant ce travail.

Conclusions et perspectives

1. Conclusion : contributions scientifiques

« Avoir un support méthodologique et technique qui aiderait les biologistes travaillant sur les puces à ADN à valider et interpréter leurs résultats d'expériences » : Tel était le besoin pour lequel ce travail a été initié.

Ce besoin rentre dans le cadre d'une problématique générique de gestion des connaissances au sein d'une communauté scientifique travaillant sur des expériences. Les personnes appartenant à ce type de communauté sont souvent confrontées au problème d'hétérogénéité des sources d'informations. En effet, les connaissances d'une équipe de recherche sont distribuées dans de multiples sources d'informations : (i) les articles scientifiques du domaine, (ii) les bases de données stockant les résultats des expériences ainsi que leurs interprétations, et (iii) les personnes acquérant de nouvelles connaissances tout au long de leur parcours scientifique.

Afin d'apporter des solutions à cette problématique, nous nous sommes inspirés des travaux liés au Web Sémantique ainsi que des travaux qui portent sur la notion de mémoire d'entreprise, et plus particulièrement la mémoire de projet. A partir de là, nous avons proposé une mémoire d'expériences (MEAT) permettant d'intégrer et de faire partager les connaissances du domaine des expériences de puces à ADN.

MEAT représente le fruit des réponses aux hypothèses que nous nous avons choisies au début de nos travaux pour aborder les problèmes suivants:

- **La capitalisation et la modélisation des connaissances** du domaine : Nous avons proposé d'utiliser (i) les ontologies qui offrent un cadre formel pour la représentation et l'organisation des connaissances du domaine et (ii) les annotations sémantiques qui décrivent le contenu sémantique des ressources de la communauté (par analogie avec les ressources du Web).
- **La création des annotations sémantiques** : Nous avons proposé d'utiliser les connaissances contenues dans les textes en se basant sur une méthodologie d'extraction de connaissances textuelles que nous avons élaborée.
- **La diffusion de ces connaissances** : Nous avons choisi d'exploiter les annotations générées, d'une part, pour faciliter la recherche d'information au sein de la mémoire, et d'autre part, pour offrir des mécanismes de raisonnement sur les connaissances contenues dans cette mémoire.

La capitalisation et la modélisation des connaissances

MEAT repose essentiellement sur l'ontologie modulaire que nous avons proposée pour la modélisation et la formalisation des connaissances dans le domaine des biopuces. Cette modularité nous a permis de couvrir toutes les connaissances du domaine (i.e UMLS pour les connaissances biomédicales, MGED pour les connaissances concernant les expériences et DocOnto pour les métadonnées sur les annotations et pour la structuration des ressources) afin

d'offrir un moyen pour décrire les annotations sur les ressources et de guider la recherche d'information au sein de la mémoire.

Plusieurs chercheurs ont émis des doutes sur la possibilité de réutiliser une ontologie : ils insistent sur l'influence de l'application dans les choix de modélisation de l'ontologie. Notre expérience dans MEAT montre clairement que l'enrichissement et la réutilisation d'une ontologie existante peut néanmoins aider dans la génération automatique d'annotations sémantiques. En effet, l'utilisation d'une ontologie de référence (le réseau sémantique de UMLS dans notre cas) couplée à une riche terminologie (le métathésaurus de UMLS) facilite la tâche d'extraction de connaissances à partir des textes et permet de générer des annotations consistantes et partageables.

La création des annotations sémantiques

En ce qui concerne la génération des annotations, nous avons proposé une méthodologie innovante et générique basée sur les techniques de traitement automatique du langage naturel (TALN). Elle diffère ainsi des méthodes basées sur les techniques d'apprentissage qui nécessitent une phase d'entraînement sur des corpus annotés manuellement pour générer leur règles d'extraction (nous pouvons citer par exemple les travaux de [Handschuh et al., 2002] et [Vargas-Vera et al., 2002] pour les documents textuels en général et les approches présentées dans [Shatkey et Feldman, 2004] pour les textes biologiques).

Le système implémentant cette méthodologie permet de traiter des documents textuels entiers et d'extraire des instances de concepts reliées par des instances de relations de l'ontologie, ce qui contribue à la création d'annotations riches facilitant ainsi la recherche d'informations. Cette façon de procéder rejoint celle du groupe TIA qui propose une méthodologie d'analyse de corpus textuels à des fins d'acquisition de ressources termino-ontologiques. Cependant, alors que ce groupe s'intéresse à l'utilisation des techniques de TALN pour la construction d'ontologies ou de terminologies [Aussenac-Gilles et al., 2000], nous nous y intéressons pour la création des annotations basées sur des ontologies existantes.

L'extraction des relations, un des points forts de notre méthodologie, a été étudiée dans CAMELEON [Séguéla et Aussenac-Gilles, 2000], qui a proposé des marqueurs de détection de relations comparables à nos grammaires JAPE. Par ailleurs, MeatAnnot pousse le traitement linguistique à un niveau plus fin (calcul des rôles linguistiques par le biais de RASP) pour extraire les termes liés par ces relations. En outre, CAMELEON ne s'appuie pas sur une ontologie existante et ne visait aucune exploitation dans un contexte web sémantique.

La diffusion de ces connaissances

Pour la diffusion des connaissances, nous avons proposé un prototype qui se base sur un moteur de recherche sémantique (i.e Corese) pour exploiter la base d'annotations contenant les annotations générées automatiquement à partir des textes, les annotations décrivant les expériences, et les métadonnées que nous avons proposées. Ces métadonnées permettent d'offrir d'avantage de raisonnement et d'informations sur les annotations. Cette étape de notre travail

nous a permis d'améliorer la tâche de recherche d'informations en la rendant plus efficace et en offrant des capacités de raisonnement aidant à découvrir de nouvelles connaissances non explicitées dans les sources d'informations.

Proposant cette recherche d'informations sémantique avancée, MeatSearch répond particulièrement aux besoins de nos collègues biologistes et plus généralement à des besoins évoqués au sein du groupe d'intérêt du W3C : 'Semantic Web Health Care and Life Sciences Interest Group'⁴².

Enfin, ce travail nous a permis de proposer une approche originale pour la construction d'une mémoire d'expériences pour une communauté de scientifiques (en particulier une communauté de pratique) et ce, en réussissant l'intégration de plusieurs technologies innovantes à savoir : (i) les ontologies et les annotations sémantiques, (ii) les techniques de TALN pour l'extraction des connaissances, et (iii) des standards et des outils du Web Sémantique (RDF, Corese...) permettant la réutilisation et le partage.

2. Limites et perspectives

La méthodologie proposée pour la génération d'annotations sémantiques à partir des textes, en extrayant automatiquement des instances des concepts et des instances des relations de l'ontologie, est totalement généralisable pour n'importe quel domaine. Néanmoins, les bons résultats obtenus par MeatAnnot, le module implémentant cette méthodologie, sont positivement influencés par la richesse du métathésaurus de UMLS qui a surtout facilité la phase d'extraction des instances des concepts. Nous pensons donc qu'une réutilisation de ce module en utilisant une autre ontologie devrait nécessiter un travail en amont sur l'enrichissement et le peuplement de cette ontologie. Ce travail pourrait être facilité par l'application des outils de TALN sur des corpus proposés par les experts du domaine, comme par exemple :

- L'utilisation d'outils tels que Nomino [David et Plante, 1990] ou Likes [Ouesleti et al., 1996], pour peupler ou enrichir la hiérarchie des concepts (approche utilisée dans [Golebiowska et al., 2001]) ;
- L'utilisation de Syntex [Bourigault et al., 2005] afin d'extraire des syntagmes verbaux pouvant être candidats pour l'enrichissement de la hiérarchie des relations.

Perspectives à court et moyen terme

Une première perspective à court terme pour ce travail consiste à étudier l'extraction d'informations à partir des graphiques et des tableaux se trouvant dans les articles, compte tenu

⁴² <http://www.w3.org/2001/sw/hcls/>

Conclusions et perspectives

de leur importance pour les biologistes, de manière à intégrer un nouveau module à MeatAnnot pour les prendre en compte.

Une autre amélioration de MeatAnnot consiste à trouver des mécanismes pour prendre en compte deux types d'informations :

- les informations contextuelles lors de l'extraction des connaissances. Prenons l'exemple de la phrase suivante : “*In vitro* assays demonstrated that only *p38alpha* and *p38beta* are inhibited by *csaids*.”. Dans cette phrase, MeatAnnot réussit à détecter que ‘*p38alpha*’ et ‘*p38beta*’ sont *inhibés* par ‘*csaids*’ mais ne prend pas en compte le fait que cette inhibition est observée ‘*in vitro*’ alors que cette information peut s’avérer importante pour l’interprétation d’un résultat particulier ;
- les formes nominales des relations qu’actuellement nos grammaires de détection ne cherchent pas à extraire. En effet, une relation comme ‘*has_a_role_in*’ peut apparaître dans les textes sous une forme non verbale comme par exemple la forme ‘*the role played by*’.

Nous évoquons aussi l’importance de la gestion de l’évolution des ontologies et des annotations tout au long du cycle de vie de la mémoire d’expériences. En effet, une évolution de l’ontologie, que ce soit au niveau de sa structure ou au niveau des connaissances qu’elle modélise, peut générer des incohérences dans la base d’annotations et induire en erreur toute les inférences effectuées sur les connaissances contenues dans ces annotations.

En ce qui concerne MeatSearch, il serait intéressant d’impliquer encore plus les biologistes dans la création des scénarios d’utilisation, afin de leur créer des requêtes typiques facilitant l’utilisation de notre système et la navigation contextuelle dans la base d’annotations.

Perspectives à long terme

L’approche sémantique basée sur les ontologies et les annotations sémantiques que nous avons adoptée, peut être utilisée pour d’autres tâches que la recherche d’informations et l’aide au raisonnement. Prenons l’exemple des algorithmes de fouille de données expérimentales qui permettent de créer des classes d’entités (dans notre cas les gènes) ayant le même profil tout en se basant sur des calculs statistiques. Les annotations structurées selon un modèle ontologique devraient aider ces algorithmes à raffiner leurs classifications en leur fournissant une information sémantique sur les entités à classer.

Finalement, il serait fort intéressant de tester toute l’approche avec des biologistes travaillant sur d’autres sujets que les biopuces, ainsi que dans d’autres communautés de chercheurs effectuant des expériences (chimie, physique...) et ayant un besoin d’aide à la validation et l’interprétation des résultats.

Bibliographie

- [Altschul et al., 1994] Altschul S., Boguski M., Gish W. et Wootton J. (1994) : *Issues in searching molecular sequence databases*. Nature Genetics 6:119-129.
- [Aronson, 2001] Aronson A.R. (2001) : *Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program*. Proc. AMIA Symp., 17–21.
- [Ashburner et al., 2001] Ashburner M., Ball C., Blake J., Butler H., Cherry J., Corradi J., Dolinski K., Janan T., Eppig T. et Harris M. (2001) : *Creating the Gene Ontology resource: design and implementation*. Genome Research, 1425–1433.
- [Assadi, 1998] Assadi H. (1998) : *Construction d'ontologies à partir de textes techniques, Application aux systèmes documentaires*. Thèse de doctorat, Université Paris 6.
- [Aussenac-Gilles et al., 2000] Aussenac-Gilles N., Biébow B., Szulman N. (2000) : *Revisiting Ontology Design: a method based on corpus analysis*. Proc of EKAW'2000. Juan-Les-Pins (F). Oct 2000. Lecture Notes in Artificial Intelligence Vol 1937. Springer Verlag. pp. 172-188.
- [Bachimont, 2000] Bachimont B. (2000) : *Engagement sémantique et engagement ontologique: conception et réalisation d'ontologies en ingénierie des connaissances*. In Charlet, Zacklad, Kassel, Bourigault, Ingénierie des connaissances Evolutions récentes et nouveaux défis. Eyrolles.
- [Berners-Lee et al., 2001] Berners-Lee T., Hendler J. et Lassila O (2001) : *The semantic web*. Scientific American.
- [Bringay et al., 2005] Bringay S., Barry C., Charlet J. (2005) : *Les annotations pour gérer les connaissances du dossier patient. 16èmes journées francophones d'ingénierie des connaissances IC'2005*. Nice, France.
- [Blaschke, 2002] Blaschke C, Valencia A (2002) : *Molecular biology nomenclature thwarts information-extraction progress*. IEEE Intelligent System 17: 73-76.
- [Bodenreider, 2001] Bodenreider O. (2001) : *Circular Hierarchical Relationships in the UMLS: Etiology, Diagnosis, Treatment, Complications and Preventions*. Proc AMIA Symp.; 57-61.
- [Bourigault et al., 1996] Bourigault D., GONZALEZ I. et GROS C. (1996) : *LEXTER, a Natural Language Tool for Terminology Extraction*. In Proceedings of the seventh EURALEX International Congress, Goteborg, Suède.
- [Bourigault et Fabre, 2000] Bourigault D. et Fabre C. (2000) : *Approche linguistique pour l'analyse syntaxique de corpus*. Cahiers de grammaire, Vol.25, 131-151.
- [Bourigault, 2002] Bourigault D. (2002) : *Upery : un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus*, Actes de la 9^e conférence annuelle sur le Traitement Automatique des Langues (TALN 2002), Nancy, p. 75-84.
- [Bourigault et al., 2005] Bourigault D., Fabre C., Frérot C., Jacques M.-P. et Ozdowska S. (2005) : *Syntex, analyseur syntaxique de corpus*. in Actes des 12^{èmes} journées sur le Traitement Automatique des Langues Naturelles, France.

Bibliographie

- [Broekstra et al., 2002] Broekstra J., Kampan A., et van Harmelen F. (2002) : *Sesame: A generic architecture for storing and querying RDF and RDF Schema*. In proc of ISWC'02, pages 54-68
- [Briscoe et Carroll, 2002] Briscoe E. et Carroll J. (2002) : *Robust accurate statistical annotation of general text*. In Proceedings of the Third IC LR E, Las Palmas, Gran Canaria. 1499-1504.
- [Charlet et al., 1996] Charlet J., Bachimont B., Bouaud J. et Zweigenbaum P. (1996) : *Ontologie et réutilisabilité: expérience et discussion*. Acquisition et ingénierie des connaissances, Cépadues-Editions, Toulouse, pp. 69-87.
- [Cherfi et al., 2005] Cherfi H., Napoli A. et Toussaint Y. (2005) : *Towards a Text Mining Methodology Using Association Rules Extraction*. Soft Computing Journal.
- [Chiang et al., 2004] Chiang J.H., Yu H.C. et Hsu H.J. (2004): *GIS: a biomedical text mining system for gene information discovery*. Bioinformatics 20(1) 120-121.
- [Cimiano et al., 2005] Cimiano P., Ladwig G., et Staab S. (2005) : *Gimme' the context: Context-driven automatic semantic annotation with C-PANKOW*. In Proceedings of the 14th World Wide Web Conference.
- [Ciravegna, 2003] Ciravegna F. (2003) : *Designing adaptive information extraction for the Semantic Web in Amilcare*. In S. Handschuh and S. Staab, editors, Annotation for the Semantic Web, Frontiers in Artificial Intelligence and Applications. IOS Press.
- [Clark, 1999] Clark J. (1999) : XSL Transformations (XSLT) Version 1.0, W3C Recommendation. <http://www.w3.org/TR/xslt>.
- [Cohen et Hunter, 2005] Cohen, K. B. et Hunter, L. (2005) : *Natural Language Processing and Systems Biology*. In proc. of Artificial Intelligence and Systems Biology. Springer.
- [Collier et al., 2000] Collier N., Nobata C., et Tsujii J. (2000) : *Extracting the Names of Genes and Gene Products with a Hidden Markov Model*. In Proc. of COLING 2000, pages 201–207, 2000.
- [Corby et Faron-Zucker, 2002] Corby O. et Faron-Zucker C. (2002) : *Corese: A Corporate Semantic Web Engine*. In WWW'02 Workshop on Real World RDF and Semantic Web Applications, Hawai..
- [Corby et al., 2004] Corby O., Dieng-Kuntz R. et Faron-Zucker C. (2004) : *Querying the Semantic Web with the CORESE engine*. In R. Lopez de Mantaras and L. Saitta eds, Proceedings of the 16th European Conference on Artificial Intelligence (ECAI'2004), Valencia, Spain, IOS Press, p.705-709
- [Corby et al., 2006] Corby O., Dieng-Kuntz R., Faron-Zucker C. et Gandon F. (2006) : *Searching the Semantic Web: Approximate Query Processing Based on Ontologies*. In IEEE Intelligent Systems Vol.21 No.1 pp. 20-27.
- [Craven, 1999] Craven, M. and Kumlien, J. (1999) : *Constructing Biological Knowledge Bases by Extracting Information from Text Sources*. In Proceedings of the Seventh international Conference on intelligent Systems For Molecular Biology, AAAI Press, 77-86
- [Cunningham et al., 2002] Cunningham H., Maynard D., Bontcheva K. et Tablan V. (2002) : *GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications*. ACL'02.

- [D'Aquin, 2005] D'Aquin M. (2005) : *Un portail sémantique pour la gestion des connaissances en cancérologie*. Thèse de doctorat en informatique. Université Henri Poincaré – Nancy 1.
- [Daille, 1994] Daille B. (1994) : *Approche mixte pour l'extraction automatique de terminologie : statistique lexicale et filtres linguistiques*. Thèse de doctorat en informatique. Université Paris 7.
- [Dameron et al., 2004] Dameron O., Noy N., Knublauch H., et Musen M. (2004) : *Accessing and manipulating ontologies using web services*. In Proceeding of the Third International Semantic Web Conference (ISWC2004), Semantic Web Services workshop
- [David et Plante, 1990] David, S. et Plante, P. (1990) : *De la nécessité d'une approche morpho-syntaxique dans l'analyse de textes*. Intelligence artificielle et sciences cognitives au Québec, 3 (3), pp.140-154.
- [Decker et al., 1999] Decker S., Erdmann M., Fensel D., et Studer R. (1999) : *Ontobroker: Ontology based access to distributed and semi-structured information*. In R. Meersman et al., editor, Semantic Issues in Multimedia Systems. Proceedings of DS-8, pages 351–369. Kluwer Academic Publisher.
- [Dieng-Kuntz, 2004a] Dieng-Kuntz R. (2004) : *Capitalisation des Connaissances via un Web Sémantique d'Entreprise*. Chapitre 12 de Management des Connaissances en Entreprise, Imed Boughzala et Jean-Louis Ermine eds, Hermès.
- [Dieng-Kuntz et al., 2004b] Dieng-Kuntz R., Minier D., Corby F., Ruzicka M., Corby O., Alamarguy L., Luong H. (2004): *Medical Ontology and Virtual Staff for a Health Network*. EKAW 2004: 187-202.
- [Dieng-Kuntz et al., 2005] Dieng-Kuntz R., Corby O., Gandon F., Giboin A., Golebiowska J., Matta N., Ribière M. (2005) : *Knowledge management: Méthodes et outils pour la gestion des connaissances, 3ème édition*, DUNOD.
- [Eom et Zhang, 2004] Eom J. et Zhang B. (2004) : *PubMiner: Machine Learning-based Text Mining for Biomedical Information Analysis*. In Genomics & Informatics Vol. 2(2) 99-106.
- [Fensel et al., 2000] Fensel D., Horrocks I., Van Harmelen F., Decker S., Erdmann M., et Klein M. (2000) : *Oil in a nutshell*. In 12th International Conference on Knowledge Engineering and Knowledge Management EKAW2000, Juan les-Pins, France.
- [Fortier, 2005] Fortier J.Y. (2005) : *Vers une gestion des connaissances au niveau des informations*. Thèse de doctorat en informatique. Université de Picardie Jules Verne.
- [Fukuda et al., 1998] Fukuda K., Tsunoda T., Tamura A., et Takagi T. (1998) : *Toward information extraction: identifying protein names from biological papers*. PSB, pages 705–716.
- [Gandon, 2002] Gandon F. (2002) : *Distributed Artificial Intelligence and Knowledge Management: ontologies and multi-agent systems for a corporate semantic web*. Thèse de doctorat en informatique. Université de Nice Sophia Antipolis.
- [Gandon, 2003] Gandon F. (2003) : *Agents handling annotation distribution in a corporate semantic Web*. In Web Intelligence and Agent Systems, IOS Press International Journal, (Eds) Jiming Liu, Ning Zhong, Volume 1, Number 1, pp 23-45, ISSN: 1570-1263, WI Consortium.
- [Gandon et al., 2005a] Gandon F., Lo M., Corby O. et Dieng-Kuntz R. (2005) : *Managing enterprise applications as dynamic resources in corporate semantic webs: an application*

Bibliographie

scenario for semantic web services. In W3C Workshop on Frameworks for Semantics in Web Service, <http://www.w3.org/2005/04/FSWS/>.

[Gandon et al., 2005b] Gandon, F., Corby, O., Giboin, A., Gronnier, N. et Guigard, C. (2005) : Graph-based inferences in a Semantic Web Server for the Cartography of Competencies in a Telecom Valley, ISWC, Lecture Notes in Computer Science, Galway.

[Golebiowska et al., 2001] Golebiowska J., Dieng-Kuntz R., Corby O. et Mousseau D. (2001) : *Building and Exploiting Ontologies for an Automobile Project Memory*. First International Conference on Knowledge Capture (K-CAP), Victoria, October 23–24.

[Gomez-Perez et al., 2003] Gomez-Perez A., Fernandez-Lopez M., et Corcho O. (2003): *Ontological Engineering*. Springer Verlag.

[Grefenstette, 1994] Grefenstette G. (1994): *Explorations in automatic thesaurus discovery*. Dordrecht, The Netherlands: Kluwer.

[Gruber, 1993] Gruber T. (1993): *A translation approach to portable ontology specifications*. Knowledge Acquisition. 5(2):199–220, 1993.

[Guarino et Giaretti, 1995] Guarino N. and Giaretti P. (1995) : *Ontologies and knowledge bases: Towards a terminological clarification*. In Towards Very Large Knowledge Bases. N. J. I. Mars, Ed., IOS Press: 25-32.

[Handschuh et al., 2002] Handschuh S., Staab S. et Ciravegna F. (2002) : *S-CREAM – Semi-automatic CREation of Metadata*. The 13th International Conference on Knowledge Engineering and Management (EKAW 2002), ed Gomez-Perez, A., Springer Verlag,

[Hobbs, 2000] Hobbs, J.R. (2000) : *Information extraction from biomedical text*. Journal Biomedical Informatics. In Proceedings of *Pac Symposium Biocomputers*. p. 541-552.

[Hobbs et al., 1997] Hobbs, J.R., Appelt D., Bear J., Israel D., Kameyama M., Stickel M. (1997) : *FASTUS: A Cascaded Finite-State Transducer for Extracting Information From Natural-Language Text*. Finite-State Language Processing., Cambridge: MIT press. 383-406.

[Humphreys et Lindberg, 1993] Humphreys B. et Lindberg D. (1993) : *The UMLS project: making the conceptual connection between users and the information they need*. Bulletin of the Medical Library Association 81(2): 170.

[Jacquemin, 1997] Jacquemin C. (1997) : *Variation terminologique : reconnaissance et acquisition automatique de termes et de leurs variantes en corpus*. Habilitation à diriger des recherches, Université de Nantes, France.

[Jouis, 1993] Jouis C. (1993) : *Contributions à la conceptualisation et à la modélisation des connaissances à partir d'une analyse linguistique de textes. Réalisation d'un prototype : le système SEEK*, Thèse de doctorat EHESS, Paris.

[Kahan et al., 2001] Kahan J., Koivunen M., Prud'Hommeaux E., et Swick R. (2001) : *Annotea: An Open RDF Infrastructure for Shared Web Annotations*. In Proceedings of the WWW10 International Conference. Hong Kong.

[Kashyap, 2003] Kashyap V. et Borgida A. (2003) : *Representing the UMLS Semantic Network using OWL: (Or "What's in a Semantic Web link?")*. In ISWC'2003. Heidelberg: Springer-Verlag; 1-16.

- [Kazama et al., 2002] Kazama J., Makino T., Ohta Y., et Tsujii J. (2002) : *Tuning svm for biomedical named entity recognition*. In Proceedings of the workshop on NLP in the biomedical domain
- [Khelif et Dieng-Kuntz, 2004] Khelif K. et Dieng-Kuntz R. (2004) : *Ontology-Based Semantic Annotations for Biochip Domain*, Proceeding of EKAW 2004 Workshop on the Application of Language and Semantic Technologies to support KM Processes, U.K., 2004, <http://CEUR-WS.org/Vol-121/>.
- [Khelif et al., 2005] Khelif K., R. Dieng-Kuntz, P. Barbry (2005) : *Semantic web technologies for interpreting DNA microarray analyses: the MEAT system*. Proc. of WISE'05, 20-22/11 New York.
- [Khelif et al., 2006] Khelif K., R. Dieng-Kuntz, P. Barbry (2006) : *Web sémantique pour la mémoire d'expériences d'une communauté scientifique : le projet MEAT*. Proc. of EGC'06, lille.
- [Kiffer et al., 1995] Kiffer M., Laussen G. et Wu J. (1995) : *Logic Foundations of Object-Oriented and Frame-based Languages*, J. ACM, 42(4), pp. 741-843.
- [Kim et al., 2003] Kim, J.D., Ohta, T., Tateisi, Y., and Tsujii, J. (2003) : *GENIA corpus - semantically annotated corpus for bio-textmining*. Bioinformatics 19(Suppl. 1), i180-182.
- [Knublauch et al., 2004] Knublauch H., Ferguson R., Noy N., et Mark A. (2004) : *The Protégé OWL Plugin: An open development environment for semantic web applications*. In McIlraith et al., editor, Proc. ISWC '04, volume 3298 of Lect. Notes Comput. Sci., pages 229–243.
- [Krauthammer et al., 2000] Krauthammer M., Rzhetsky A., Morozov P. et Friedman C. (2000) : *Using BLAST for identifying gene and protein names in journal articles*. Gene 259(1-2):245-52.
- [Lassila et Swick, 2001] Lassila O. et Swick R. (2001) : *W3C Resource Description framework (RDF) Model and Syntax Specification*, <http://www.w3.org/TR/REC-rdf-syntax/>.
- [Le moigno et al., 2002] Le moigno S., Charlet J., Bourigault D. & Jaulent M.-C. (2002) : *Construction d'une ontologie à partir de corpus : expérimentation et validation dans le domaine de la réanimation chirurgicale*. Actes des 13èmes Journées Ingénierie des Connaissances, p. 229–38, Rouen, France.
- [Lomax et McCray, 2004] Lomax J. et McCray A. (2004) : *Mapping the Gene Ontology into the Unified Medical Language System*. Comparative and Functional Genomics, 5:354–361.
- [Martin et al., 2004] Martin D., Burstein M., Denker G., Hobbs J. et al. (2004) : *OWL-S: Semantic Markup for Web Services*. <http://www.daml-s.org/owl-s/1.0/>.
- [Matta et al., 1999] Matta N., Ribière M. et Corby O. (1999) : *Définition d'un modèle de mémoire de projet*. Rapport de recherche INRIA n°3720.
- [McBride, 2004] McBride B. (2004), *RDF Vocabulary Description Language 1.0: RDF Schema*, W3C Recommendation, <http://www.w3.org/TR/rdf-schema/>.
- [McCray, 2003] McCray A. (2003) : *An upper level ontology for the biomedical domain*. Comp Functional Genomics; 4: 80-84.
- [McCray, 1995] McCray AT. et Nelson SJ. (1995) : *The semantics of the UMLS knowledge sources*. Methods Inf Med ;34(1/2).

Bibliographie

- [McGuinness et Van Harmelen, 2004] McGuinness D.L. et Van Harmelen F. (2004) : *OWL Web Ontology Language Overview*, <http://www.w3.org/TR/owl-features/>.
- [Mizoguchi et Vanwelkenhuysen, 1995] Mizoguchi, R. et Vanwelkenhuysen J. (1995): *Task ontology for reuse of problem solving knowledge*, Proc. of KB&KS'95, pp.46-59.
- [Muller et al., 2004] Muller H.M., Kenny E.E. et Sternberg P.W. (2004) : *Textpresso: an ontology-based information retrieval and extraction system for biological literature*. PLoS Biologie, E309.
- [Napoli, 1997] Napoli, A. (1997) : *Une introduction aux logiques de description*. Rapport de recherche INRIA n°3314.
- [Neches et al., 1991] Neches R., Fikes R.E., Finin T., Gruber T.R., Patil R., Senator T., et Swartout W. R. (1991) : *Enabling technology for knowledge sharing*. AI Magazine, 12(3), 16-36.
- [Nédellec et Nazarenko, 2001] Nédellec C. et Nazarenko A. (2001) : *Application de l'apprentissage à la recherche et à l'extraction d'information - Un exemple, le projet Caderige : identification d'interactions géniques*. In Actes de la Journée thématique Exploration de données issues d'Internet, Bennani Y., et al. (Eds).
- [Nédellec et al., 2004] Nédellec C., et al. (2004) : *Machine learning for information extraction in genomics state of the art and perspectives*. In: Sirmakessis, S. (ed.) : Text Mining and its Applications. Studies in Fuzzi. and Soft Comp. 138. Springer Verlag, Berlin Heidelberg New York 99-118.
- [Niles et Pease, 2001] Niles I., et Pease A. (2001) : *Towards a Standard Upper Ontology*. Proceedings of the 2nd International Conference on Formal Ontology in Information Systems FOIS'2001.
- [Noy et al., 2001] Noy N., Sintek F., Decker M. et al. (2001) : *Creating semantic web contents with Protege-2000*. IEEE Intelligent Systems, 16(2): 60-71.
- [Noy et al., 2004] Noy N.F., Rubin L.D. et Musen A.M. (2004) : *Making Biomedical Ontologies and Ontology Repositories Work*, in IEEE Intelligent Systems, vol. 19, no. 6, 78-81.
- [Ohlbach et Schaffert, 2004] Ohlbach H.J. et Schaffert S. (2004) : *eds Workshop on Principles and Practice of Semantic Web Reasoning* at the 20th ICLP, St Malo, France.
- [Ouesleti et al., 1996] Ouesleti R., Frath P., Rousselot F. (1996) : *A corpus-based method for acquisition and exploitation of terms* , CLIM-96, Student Conference in Computational Linguistics in Montreal, Université de Montreal, Canada.
- [Pomian, 1996] Pomian J., (1996) : *Mémoire d'entreprise, techniques et outils de la gestion du savoir*. p. 11, Ed Sapientia.
- [Popov et al., 2004] Popov B., Kiryakov A., Ognyanoff D., Manov D. et A. Kirilov (2004) : *KIM – a semantic annotation platform for information extraction and retrieval*. Natural Language Engineering, 10, Issues 3-4, 375-392.
- [Price et Spackman, 2000] Price, C. et Spackman, K.,A. (2000) : *Snomed Clinical Terms*, British Journal of Healthcare Computing and Information Management 17(2):27-31

- [Rector et al., 1996] Rector A., Rogers J.E. et Pole P. (1996) : *The GALEN High Level Ontology*. Fourteenth International Congress of the European Federation for Medical Informatics, MIE96, Copenhagen, Denmark.
- [Rector et al., 1997] Rector A., Bechhofer S., Goble C., Horrocks I., Nowlan W. et Solomon W. (1997) : *The GRAIL concept modelling language for medical terminology*. Journal of AI in Medicine 9(2):139–171.
- [Rindflecsh et al., 2000] Rindflecsh T.C., Tanabe L., Weinstein J.N., Hunter L. (2000) : *EDGAR : extraction of drugs, genes and relations from the biomedical literature*. Proceedings of the Pac Symposium of Biocomputers, p. 517-528.
- [Séguéla et Aussenac-Gilles, 2000] Séguéla P. et Aussenac-Gilles N. (1999) : *Extraction de relations sémantiques entre termes et enrichissement de modèles du domaine*. In IC'99, Paris, 79-88.
- [Shatkay et al., 2002] Shatkay H., Edwards S. et Boguski M: *Information retrieval meets gene analysis*. IEEE Intelligent System (Special Issue on Intelligent Systems in Biology). 17:45-53.
- [Shatkay et Feldman, 2003] Shatkay H. et Feldman R. (2003) : *Mining the biomedical literature in the genomic era: an overview*. Journal of Computational Biology, 10, 821–855.
- [Schmid, 1994] Schmid H. (1994) : *Probabilistic part-of-speech tagging using decision trees*. In proceedings of International Conference on New Methods in Language Processing. Manchester
- [Seifoddini et Wolfe, 1986] Seifoddini H., Wolfe P.M. (1986) : Application of the similarity coefficient method in group technology, IIE Trans. 18(3) 271–277.
- [Smadja, 1993] Smadja F. (1993) : *Retrieving collocations from text: Xtract*. Computational Linguistics, 19(1):143-177.
- [Sowa, 1984] Sowa J.F. (1984) : *Conceptual Graphs: Information Processing*. in Mind and Machine. Reading, Addison Wesley.
- [Staab, 2002] Staab S, editor (2002) : *Mining information for functional genomics*. IEEE Intelligent System 17-66.
- [Toussaint et al., 1997] Toussaint Y., Royaute J., Muller C., Polanco X., (1997) : *Analyse linguistique et infométrie pour l'acquisition et la structuration des connaissances*. Actes des deuxièmes rencontres Terminologie et Intelligence Artificielle (TIA'97), pp 27-46. Toulouse.
- [Uren et al., 2006] Uren V., Cimiano P., Iria J., Handschuh S., Vargas-Vera M., Motta E. et Ciravegna F. (2006) : *Semantic annotation for knowledge management: Requirements and a survey of the state of the art*. In Web Semantics, Volume 4, Issue 1, pp 14-28
- [Uschold et Grüninger, 1996] Uschold M. and Grüninger M. (1996): *Ontologies: Principles, Methods and Applications*, Knowledge Engineering Review, 11(2).
- [Van Harmelen et al., 2001] Van Harmelen F., Horrocks I., Peter F. (2001) : *A model theoretic semantics for DAML+OIL*. W3C Note, 18 December 2001. <http://www.w3.org/TR/2001/NOTE-daml+oil-model-20011218>
- [Van Heijst et al., 1996] Van Heijst G., Van der Spek R., et Kruizinga E. (1996) : *Organizing Corporate Memories*. In B. Gaines, M. Musen eds, Proc. of KAW'96, Banff, Canada, November, pp. 42-1 42-17.

Bibliographie

[Van Heijst et al., 1997] Van Heijst G., Schreiber A., et Wielinga B. (1997): *Using explicit ontologies in KBS development*. Int. J. of Human-Computer Studies, 46(2/3):183–292.

[Vargas-Vera et al., 2002] Vargas-Vera M., Motta E., Domingue J., Lanzoni M., Stutt A. et Ciravegna F. (2002) : *MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup*, In Proc. of the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002, Springer Verlag LNAI 2473, 379-391.

[Zweigenbaum, 1994] Zweigenbaum P. (1994) : *MENELAS: an access system for medical records using natural language*. Comput Methods Programs Biomed. Oct;45(1-2):117-20.

[Zweigenbaum et al., 2003] Zweigenbaum P, Baud R, Burgun A, Namer F, Jarrousse E, Grabar N, Ruch P, Le Duff F, Thirion B, Darmoni S. (2003) : *UMLF: A Unified Medical Lexicon for French*. Proc AMIA Symp, Washington DC, USA.

[Zweigenbaum, 2004] Zweigenbaum P. (2004) : *L'umls entre langue et ontologie : une approche pragmatique dans le domaine médical*. Revue d'Intelligence Artificielle, 111-137.

Annexe I : Les grammaires de détection des relations

1. Description des relations sémantiques dans UMLS

Nous présentons ici la description des relations sémantiques que nous avons choisi d'extraire pour la génération des annotations.

Semantic Relation: affects

TUI: T151

Definition: Produces a direct effect on. Implied here is the altering or influencing of an existing condition, state, situation, or entity. This includes has a role in, alters, influences, predisposes, catalyzes, stimulates, regulates, depresses, impedes, enhances, contributes to, leads to, and modifies.

Inverse: affected_by

Abbreviation: AF

Semantic Relation: interacts_with

TUI: T142

Definition: Acts, functions, or operates together with.

Inverse: interacts_with

Abbreviation: IW

Semantic Relation: disrupts

TUI: T146

Definition: Alters or influences an already existing condition, state, or situation. Produces a negative effect on.

Inverse: disrupted_by

Abbreviation: DS

Semantic Relation: prevents

TUI: T148

Definition: Stops, hinders or eliminates an action or condition.

Inverse: prevented_by

Abbreviation: PV

Semantic Relation: complicates

TUI: T149

Definition: Causes to become more severe or complex or results in adverse effects.

Inverse: complicated_by

Abbreviation: CM

Semantic Relation: manages

TUI: T153

Definition: Administers, or contributes to the care of an individual or group of individuals.

Inverse: managed_by

Abbreviation: MN

Semantic Relation: treats

TUI: T154

Definition: Applies a remedy with the object of effecting a cure or managing a condition.

Inverse: treated_by

Abbreviation: TS

Semantic Relation: occurs_in

TUI: T152

Definition: Takes place in or happens under given conditions, circumstances, or time periods, or in a given location or population. This includes appears in, transpires, comes about, is present in, and exists in.

Inverse: has_occurrence

Abbreviation: OC

Semantic Relation: process_of

TUI: T140

Definition: Action, function, or state of.

Inverse: has_process

Abbreviation: PO

Semantic Relation: uses

TUI: T155

Definition: Employs in the carrying out of some activity. This includes applies, utilizes, employs, and avails.

Inverse: used_by

Abbreviation: US

Semantic Relation: indicates

TUI: T156

Definition: Gives evidence for the presence at some time of an entity or process.

Inverse: indicated_by

Abbreviation: IN

Semantic Relation: brings_about

TUI: T187

Definition: Acts on or influences an entity.

Inverse: brought_about_by

Abbreviation: BA

Semantic Relation: produces

TUI: T144

Definition: Brings forth, generates or creates. This includes yields, secretes, emits, biosynthesizes, generates, releases, discharges, and creates.

Inverse: produced_by

Abbreviation: PS

Semantic Relation: causes

TUI: T147

Definition: Brings about a condition or an effect. Implied here is that an agent, such as for example, a pharmacologic substance or an organism, has brought about the effect. This includes induces, effects, evokes, and etiology.

Inverse: caused_by

Abbreviation: CA

Semantic Relation: performs

TUI: T188

Definition: Executes, accomplishes, or achieves an activity.

Inverse: performed_by

Abbreviation: PE

Semantic Relation: measures

TUI: T162

Definition: Ascertaines or marks the dimensions, quantity, degree, or capacity of.

Inverse: measured_by

Abbreviation: MS

Semantic Relation: diagnoses

TUI: T163

Definition: Distinguishes or identifies the nature or characteristics of.

Inverse: diagnosed_by

Abbreviation: DI

Semantic Relation: analyzes

TUI: T193

Definition: Studies or examines using established quantitative or qualitative methods.

Inverse: analyzed_by

Abbreviation: AN

Semantic Relation: assesses_effect_of

TUI: T164

Definition: Analyzes the influence or consequences of the function or action of.

Inverse: assessed_for_effect_by

Abbreviation: AE

Semantic Relation: carries_out

TUI: T141

Definition: Executes a function or performs a procedure or activity. This includes transacts, operates on, handles, and executes.

Inverse: carried_out_by

Abbreviation: CO

Semantic Relation: practices

TUI: T143

Definition: Performs habitually or customarily.

Inverse: practiced_by

Abbreviation: PA

Semantic Relation: exhibits

TUI: T145

Definition: Shows or demonstrates.

Inverse: exhibited_by

Abbreviation: EX

Semantic Relation: contains

TUI: T134

Definition: Holds or is the receptacle for fluids or other substances. This includes is filled with, holds, and is occupied by.

Inverse: contained_in

Abbreviation: CT

Semantic Relation: connected_to

TUI: T174

Definition: Directly attached to another physical unit as tendons are connected to muscles. This includes attached to and anchored to.

Inverse: connected_to

Abbreviation: CN

Semantic Relation: interconnects

TUI: T175

Definition: Serves to link or join together two or more other physical units. This includes joins, links, conjoins, articulates, separates, and bridges.

Inverse: interconnected_by

Abbreviation: IC

2. Exemples de grammaires de détection des relations

Grammaire de la relation Has_role_in :

```
Phase: first
Options: control = appelt

Rule:Has_role_in

(
  ({Tag.lemme == "play"} | {Tag.lemme == "have"} )
  {SpaceToken}
  ({Tag.lemme == "a"} |
  {Tag.lemme == "an"})?
  ({SpaceToken})?
  ({Tag.cat == JJ})? //adjectif qualificatif optionnel: important,
vital, positive...
  ({SpaceToken})?
  ({Tag.lemme == "role"})

):hasrole -->
:hasrole.Relationship = {kind = "hasrole", rule=Has_role_in}
```

Grammaire de la relation Affects :

```
Phase: first
Options: control = appelt

Rule:Affect

(
  ({Tag.lemme == "affect"} | {Tag.lemme == "influence"} )

):affect -->
:affect.Relationship = {kind = "affect", rule=Affect}
```

Grammaire de la relation Interacts_with :

```
Phase: first
Options: control = appelt

Rule:Interact_with

(
  ({Tag.lemme == "interact" | Tag.lemme == "operate"} )
  {SpaceToken}
  ({Tag.cat== RB} // adverbe optionnel: positively, negatively...
  {SpaceToken})?
  {Tag.lemme == "with"}

):interact -->
:interact.Relationship = {kind = "interact", rule=Interact_with}
```

Grammaire de la relation Prevents :

```

Phase: first
Options: control = appelt

Rule:Prevent

(
  ( {Tag.lemme == "prevent"} ) |
  (
    {Tag.lemme == "have"}
    {SpaceToken}
    ( {Tag.lemme == "a"}
      {SpaceToken} ) ?
    {Tag.lemme == "preventive"}
    {SpaceToken}
    {Tag.lemme == "effect"}
  )
)

):prevent -->
:prevent.Relationship = {kind = "prevent", rule=Prevent}

```

Grammaire de la relation Disrupts :

```

Phase: first
Options: control = appelt

Rule:Disrupt

(
  ( {Tag.lemme == "disrupt"} |
    {Tag.lemme == "have"} | {Tag.lemme == "produce"} )
    {SpaceToken}
    ( {Tag.lemme == "a"}
      {SpaceToken} ) ?
    {Tag.lemme == "negative"}
    {SpaceToken}
    {Tag.lemme == "effect"}
  )
)

):disrupt -->
:disrupt.Relationship = {kind = "disrupt", rule=Disrupt}

```

Grammaire de la relation Connected_to :

```
Phase: first
Options: control = appelt

Rule:Connectedto

(
  ({Tag.mot == "connected"} | {Tag.mot == "attached"} |
   {Tag.mot == "anchored"})
  {SpaceToken}
  {Tag.mot == "to"}
):connect -->
:connect.Relationship = {kind = "connect", rule=Connectedto}
```

Grammaire de la relation Interconnects :

```
Phase: first
Options: control = appelt

Rule:Interconnect

(
  ({Tag.lemme == "interconnet"} | {Tag.lemme == "join"} |
   {Tag.lemme == "bridge"})
):interconnect -->
:interconnect.Relationship = {kind = "interconnect", rule=Interconnect}
```

Grammaire de la relation Exhibits :

```
Phase: first
Options: control = appelt

Rule:Exhibit

(
  ({Tag.lemme == "exhibit"} | {Tag.lemme == "show"} |
   {Tag.lemme == "demonstrate"})
):exhibit -->
:exhibit.Relationship = {kind = "exhibit", rule=Exhibit}
```

Grammaire de la relation Analyzes :

```
Phase: first
Options: control = appelt

Rule:Analyze

(
  {Tag.lemme == "analyze"} | {Tag.lemme == "analyse"}
):analyze -->
:analyze.RelationShip = {kind = "analyze", rule=Analyze}
```

Grammaire de la relation Alters :

```
Phase: first
Options: control = appelt

Rule:Alter

(
  {Tag.lemme == "alter"}
):alter -->
:alter.RelationShip = {kind = "alter", rule=Alter}
```

Grammaire de la relation Catalyzes :

```
Phase: first
Options: control = appelt

Rule:Catalyze

(
  {Tag.lemme == "catalyze"}
):catalyze -->
:catalyze.RelationShip = {kind = "catalyze", rule=Catalyze}
```

Grammaire de la relation Predisposes :

```
Phase: first
Options: control = appelt

Rule:Predispose

(
  {Tag.lemme == "predispose"}

):predispose -->
:predispose.RelationShip = {kind = "predispose", rule=Predispose}
```

Grammaire de la relation Stimulates :

```
Phase: first
Options: control = appelt

Rule:Stimulate

(
  {Tag.lemme == "stimulate"}

):stimulate -->
:stimulate.RelationShip = {kind = "stimulate", rule=Stimulate}
```

Grammaire de la relation Regulates :

```
Phase: first
Options: control = appelt

Rule:Regulate

(
  {Tag.lemme == "regulate"}

):regulate -->
:regulate.RelationShip = {kind = "regulate", rule=Regulate}
```