ACADEMIE DE MONTPELLIER
UNIVERSITE MONTPELLIER II
SCIENCES ET TECHNIQUES DU LANGUEDOC

Mémoire de Stage de Master

SPECIALITE : Recherche en Informatique
Mention : Informatique, Mathématiques, Statistiques

effectué au laboratoire LGI2P/EMA

Sous la direction de SYLVIE RANWEZ, GERARD DRAY, JEAN VILLERD

Distance Sémantique et Carte Conceptuelle

par

Makhtar MBAO

Soutenu le 31 Août 2007

Remerciements

Je tiens à remercier les personnes qui m'ont permis de mener à bien ce mémoire de recherche.

*Tous mes remerciements vont à mes encadrants Madame **Sylvie Ranwez**, Monsieur **Jean Villerd** et Monsieur **Gérard Dray** que je remercie tant pour leur soutien, leur patience, leur disponibilité, que pour leurs précieux conseils. Grâce à eux j'ai beaucoup appris dans le domaine de la recherche. Qu'ils soient ici assurés de mon profond respect et de la joie que j'ai eu à travailler avec eux.*

*Je remercie également les thésards du laboratoire pour leur accueil chaleureux. Je remercie en particulier **Nicolas** et **Fabien** pour les nombreuses discussions que nous avons eu durant le stage, discussions qui m'ont beaucoup apporté.*

Enfin, je remercie mes parents, pour leurs encouragements, leurs conseils et sacrifices. Je ne saurais jamais vous remercier assez pour les contributions inestimables que vous ne cessez de m'apporter.

Résumé

Le travail que j'ai dû effectuer au cours de ce stage de recherche s'inscrit dans le cadre de la distance sémantique et de la visualisation des ontologies sur une carte conceptuelle. En effet, dans le domaine des ontologies, le défi n'est plus de les projeter sur un « paysage », mais de donner un sens à ce paysage.

Dans un premier temps je me suis familiarisé avec les ontologies. J'ai étudié leur rôle, leurs principaux domaines d'utilisations et les langages qui leur sont dédiés en particulier le langage OWL.

Dans un deuxième temps, j'ai passé en revue les différentes définitions de distances sémantiques que j'ai trouvées dans la littérature. J'ai conclu de cette étude que la plupart ne respectent pas les qualités attendues d'une distance (positivité, symétrie, inégalité triangulaire) et on peut les classer en trois catégories d'approches : approche basée sur les arcs, approche basée sur les nœuds et celle basée sur les vecteurs de termes. J'ai présenté l'approche proposée par [Ranwez et al, 2006] afin de la valider.

Ensuite, j'ai étudié quelques techniques de regroupement (clustering hiérarchique ascendant et k-means). Ce qui m'a permis de faire une comparaison entre le paysage avec la distance proposée et le paysage avec application de l'algorithme de clustering hiérarchique ascendant et k-means.

En conclusion, j'ai dégagé quelques perspectives à savoir étendre la distance proposée sur les autres relations ontologiques ; et utiliser une visualisation en trois dimensions pour mieux répartir les concepts de l'ontologie sur le paysage.

Mots clés *: Ontologies, clustering, distances sémantiques, visualisation d'ontologies.*

Table des Matières

Résumé **3**

Abréviation **8**

Chapitre 1 : Introduction **9**

1.1. Contexte 9

1.2. Objectifs 9

1.3. Contributions 10

1.4. Organisation du document 10

Chapitre 2 : Distance Sémantique **11**

2.1. Ontologies 11

2.1.1. Définitions 11

2.1.2. Rôle des ontologies 12

2.1.3. Un langage standard pour les ontologies : OWL 12

2.2. Etat de l'art de distance sémantique 14

2.3. Distance proposée par [Ranwez et al, 2006] 17

2.3.1. Approche intuitive 17

2.3.2. Définition 17

Chapitre 3 : Clustering **19**

3.1. Rappels sur la classification 19

3.2. Application du clustering 19

3.2.1. Rapprochement de deux clusters 20

3.3. Clustering hiérarchique 21

3.4. Clustering K-Means 23

Chapitre 4 : Contributions **25**

4.1. Architecture fonctionnelle 25

4.2. Les applications utilisant des ontologies OWL 26

4.2.1. Protégé 2000 26

4.2.2. Jena 27

4.3. Visualisation d'ontologies (Molage) 28

4.4. Résultats et Discussion 29

4.4.1. Résultats avec une taxonomie simple 30

4.4.2. Résultats avec l'ontologie de TOXNUC-E 34

Chapitre 5 : Conclusion et Perspectives **38**
5.1. Conclusion 38
5.2. Perspectives 38
Bibliographie **40**

Table des Figures

3.1 Distance entre les deux clusters par lien simple. 20
3.2 Distance entre les deux clusters par lien complet. 20
3.3 Distance entre les deux clusters par lien moyen. 21
3.4 Distance entre les deux clusters par lien moyen de groupe. 21
3.5 Exemple de dendrogramme 23
4.1 Architecture simplifiée de l'application. 25
4.2 Une partie de l'ontologie de TOXNUC sous Protégé 2000 26
4.3 Molage. 25
4.4 Exemple de taxonomie. 30
4.5 Visualisation utilisant la distance. 30
4.6 Visualisation utilisant le clustering hiérarchique ascendant 31
4.7 Visualisation utilisant k-means avec (k = 3). 32
4.8 Visualisation utilisant k-meansavec (k = 5). 33
4.9 Visualisation utilisant des concepts de TOXNUC. 34
4.10 Visualisation utilisant le clustering hiérarchique. 35
4.11 Visualisation utilisant k-means (k = 3). 36
4.12 Visualisation utilisant k-means (k = 6). 37

Table des Algorithmes

3.1 Clustering hiérarchique ascendant. 22
3.2 Clustering k-means. 23

Abréviation

API : Application Programming Interface

DAG : Direct Acyclic Graph

DAML : DARPA Agent Markup Language

DARPA : Defense advanced Research Projects Agency

GO : Gene Ontology

LSA : Latent Semantic Analysis

MDS : MultiDimensionnal Scaling

OIL : Ontologie Inference Layer

OWL : Web Ontology Language

RDF : Ressource Description Framework

TLN : Traitement de Langage Naturel

XML : eXtensible Markup Language

Chapitre 1 : Introduction

1.1. Contexte

Dans tous les domaines de l'informatique dans lesquels on désire des outils de recherche d'information pertinents, il est nécessaire de disposer d'une mesure de similarité capable d'évaluer les ressemblances qui existent au sein des données. Cette notion de similarité est abordée dans différents domaines de l'informatique (Recherche d'Information, Intelligence Artificielle, Fouille de Données, Traitement de Langage Naturel, etc.).

Cependant, l'utilisation d'ontologies, c'est-à-dire de bases de connaissances, est devenue une voie incontournable pour représenter les connaissances en y associant de la sémantique. Elles permettent de montrer les liens (hypéronymie ou *is-a*, métonymie ou *part-whole*, etc.) entre les concepts. Le problème principal est de trouver une distance sémantique entre les concepts d'une ontologie afin de donner sens à la visualisation de l'ontologie en question.

C'est dans ce contexte que l'équipe de représentation des connaissances du centre de recherche LGI2P[1] a développé une application (Molage) pour manipuler une vaste collection de titres musicaux (navigation, indexation, fouille du paysage) de façon visuelle. Pour donner sens à ce paysage musical on projette l'ontologie du domaine. Or en utilisant une distance sémantique, cette projection entraîne des regroupements : quelles sont leurs significations ?

C'est ainsi que [Ranwez et al, 2006] a proposé une mesure de distance sémantique sur laquelle se base notre étude.

1.2. Objectifs

Notre principal objectif consiste à valider cette mesure, notamment en l'appliquant à la visualisation d'ontologies. Pour cela, il a fallu mettre en place un module permettant d'automatiser la visualisation des ontologies en format OWL sous Molage en utilisant la distance proposée. Ensuite, la comparaison des visualisations obtenues avec d'autres visualisations issues de techniques de classification devait nous permettre d'évaluer la pertinence de l'approche. L'utilisation du clustering hiérarchique et de l'algorithme k-means permet, en effet, d'avoir une idée sur le regroupement des concepts de l'ontologie, afin de valider la distance et l'outil de visualisation.

---

[1] Laboratoire de Génie Informatique et d'Ingénierie de Production de l'école des mines d'Ales

Un autre objectif est d'étendre la distance proposée sur les autres relations ontologiques.
Enfin, de dégager les perspectives qui découlent de cette étude.

1.3. Contributions

Après une analyse de la problématique et un état de l'art, j'ai mis en place un outil qui permet de visualiser des ontologies décrites en OWL sous Molage. L'outil est développé en java pour faciliter son intégration dans l'environnement Molage dans l'avenir. Ensuite, j'ai intégré dans cet outil deux modules. Le premier utilise l'algorithme de clustering hiérarchique ascendant et le second l'algorithme de clustering k-means. En comparant les résultats des différents paysages obtenus, j'ai trouvé des similitudes entre eux. Chaque regroupement de la carte est constitué de concepts de l'ontologie qui sont sémantiquement liés.
J'ai aussi entamé une étude sur les autres relations ontologiques (part-of) afin d'étendre la distance proposée sur les autre relations.

1.4. Organisation du document

Le **Chapitre 2** décrit brièvement l'état de l'art sur les distances sémantiques et plus particulièrement celle que nous devons valider. Il débute par une brève présentation de quelques définitions des ontologies, leurs rôles et le langage standard des ontologies (OWL).
Dans le **Chapitre 3** nous rappelons les méthodes traditionnelles de clustering (hiérarchique et k-means) que nous avons utilisées.
Le **Chapitre 4** détaille notre contribution. Il présente l'architecture que nous avons choisie et développée et les différents résultats de visualisation obtenus et leur interprétation. Une discussion est proposée à partir des limites de notre approche.
Enfin, le **Chapitre 5** présente la conclusion du mémoire et les perspectives qui en découlent.

Chapitre 2 : Distance Sémantique

Le présent chapitre décrit dans un premier temps un bref rappel sur les ontologies, ensuite présente un état de l'art de la distance sémantique, et enfin détaille l'approche proposée par [Ranwez et al, 2006].

2.1. Ontologies

Dans cette section, nous allons rappeler quelques définitions des ontologies, leurs applications, leurs rôles et le langage standard des ontologies (OWL).

2.1.1. Définitions

A l'origine, l'ontologie est une notion philosophique, dans laquelle les philosophes ont tenté de rendre compte de l'existence de façon formelle. L'intelligence artificielle et les chercheurs du Web sémantique ont adapté ce terme dans leur propre jargon et diverses définitions d'ontologie existent dans la littérature informatique.

Cependant, la définition la plus référencée et aussi la plus synthétique est celle de [Gruber, 1992]: « une ontologie est une spécification formelle explicite d'une conceptualisation ». [Mizoguchi, 2004] insiste sur la nature conceptuelle d'une ontologie, faite pour pouvoir partager des connaissances entre homme et systèmes, et entre systèmes. Il considère conceptualisation, partage et réutilisation comme les concepts-clés d'une ontologie, ce qui représente les attentes de la communauté d'Intelligence Artificielle sur les ontologies. Pour lui une ontologie est un système conceptuel qui permet de partager et de réutiliser des concepts grâce à une sémantique computationnelle. [Natalya et al, 2001] définit une ontologie comme une description formelle explicite des concepts dans un domaine du discours (classes appelées parfois concepts), des propriétés de chaque concept décrivant des caractéristiques et attributs du concept (attributs appelés parfois rôles ou propriétés) et des restrictions sur les attributs (facettes appelées parfois restrictions de rôles). Classe et concept définissent le même terme dans ce qui suit. Une méthodologie pour le développement d'une ontologie est aussi développée dans [Natalya et al, 2001].

En résumé, en informatique, une ontologie est comprise comme un système de concepts fondamentaux qui sont mis en relation les uns avec les autres et représentés sous une forme compréhensible par un ordinateur.

2.1.2. Rôle des ontologies

La communauté d'intelligence artificielle utilise les ontologies pour deux raisons principales : le partage et la réutilisation de connaissances, et l'amélioration de la communication. Le partage est une des raisons les plus courantes qui conduit à développer des ontologies. Supposons, par exemple, qu'un certain nombre de sites contiennent de l'information sur un domaine. Si ces sites partagent et publient tous la même ontologie, qui est constituée des termes qu'ils utilisent, alors les agents informatiques peuvent extraire et agréger l'information de ces différents sites. La réutilisation des données sur un domaine précis est une des raisons majeures qui ont poussé la recherche sur les ontologies. D'après [Ranwez, 2000] il existe trois types de communication dans un projet : communication homme-homme, homme-système ou entre les différents modules du système. Ces trois types de communication possèdent tous des caractéristiques particulières qui engendrent certains problèmes auxquels les ontologies peuvent apporter des solutions.

La communauté du *Web sémantique* utilise les ontologies pour diverses raisons. Elle l'utilise de façon simple pour améliorer la pertinence des recherches ; la requête exprimée peut ne faire référence qu'à un concept précis au lieu d'utiliser des mots-clés ambigus. Des applications plus avancées utilisent des ontologies pour associer l'information d'une page à des structures de connaissance et à des règles d'inférence.

Dans notre cas, nous allons visualiser les ontologies sur une carte conceptuelle. Cette carte doit servir de support à une recherche d'information dans de larges bases de connaissances. L'utilisateur doit cibler ses recherches sur les concepts proches de sa thématique (par exemple en biologie). Pour donner un sens à cette visualisation nous avons besoin d'une distance sémantique.

2.1.3. Un langage standard pour les ontologies : OWL

De nombreux langages informatiques sont apparus pour construire et manipuler des ontologies. Dans le but de mettre au point un langage standardisé, le W3C a crée le groupe Web Ont qui a mis en place le langage OWL. L'acronyme OWL regroupe les deux spécifications OIL développées par la Communauté Européenne et DAML langage développé par la DARPA. OIL est utilisé pour définir des ontologies et DAML ajoute un petit nombre de caractéristiques au RDF Schéma afin de rendre plus facile la définition de nouveaux langages permettant la communication entre agents intelligents. OWL définit donc une syntaxe RDF pour décrire et construire des vocabulaires pour créer des ontologies. Cependant, le langage

OWL offre trois sous langages d'expression croissante conçus pour des communautés de développeurs et d'utilisateurs spécifiques qui sont : OWL Lite, OWL DL et OWL Full. Les ontologies OWL se présentent sous forme de fichiers textes, de documents OWL. La création d'un document OWL fait l'objet de diverses recommandations [W3C 2004].

- **Structure d'une ontologie OWL**

Espaces de nommage : L'espace de noms, parfois appelé *espace de nommage* permet d'indiquer avec précision de quel vocabulaire les termes d'une ontologie proviennent. C'est la raison pour laquelle, comme tout autre document XML, une ontologie commence par une déclaration d'espace de noms contenue dans une balise *rdf:RDF*.

Exemple : le laboratoire LGI2P est partenaire du projet TOXNUC-E qui regroupe plus de 200 chercheurs autour de la toxicologie nucléaire (projet MNRT, CEA, INSERM, CNRS, INRA). Les exemples (ToxNucFev07.owl) choisis sont tirés de l'application de nos travaux à ce projet.

```
<rdf:RDF
        xmlns     = "http://www.owl-ontologies.com/ToxNucFev07#"
        xml:base  = "http://www.owl-ontologies.com/ToxNucFev07">
        xmlns:owl = "http://www.w3.org/2002/07/owl#"
        xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:rdfs= "http://www.w3.org/2000/01/rdf-schema#"
        xmlns:xsd = "http://www.w3.org/2001/XMLSchema#">
</ rdf:RDF>
```

La première déclaration identifie l'espace de nommage propre à l'ontologie que nous sommes en train d'écrire (ToxNucFev07). La deuxième déclaration identifie l'URI de base de l'ontologie. Les quatre dernières déclarations introduisent le vocabulaire d'OWL et les objets définis dans l'espace de nommage de RDF, du schéma RDF et des types de données du Schéma XML.

En-tête d'une ontologie : L'entête décrit le contenu de l'ontologie courante. La balise *owl:Ontology* permet d'indiquer les informations contenues dans l'ontologie.

- **Eléments de base**

Il existe divers éléments de base pour le langage OWL, mais nous allons décrire ceux qui nous semblent les plus importants pour mettre en œuvre notre distance. Pour des explications exhaustives du rôle de chacun des éléments composant le vocabulaire d'OWL, il est recommandé de se reporter à la Recommandation du W3C (OWL Web Ontology Language Reference) [W3C 2004].

Classes : Une classe ou concept est définie comme un groupe d'individus qui ont des caractéristiques similaires dans un domaine. Le langage OWL dispose d'une superclasse nommée *owl:Thing*. Chaque classe définie par l'utilisateur est donc implicitement une sous-classe de *owl:Thing*. Il existe également une classe nommée *noThing*, qui est sous-classe de toutes les classes OWL. Ces deux concepts permettent de construire le treillis des concepts.

Voici un exemple de déclaration de classe : *<owl:Class rdf:ID="Biologie">*

L'appel de la classe *Biologie* au sein du document se fait par *#Biologie,* par exemple *rdf:resource='#Biologie'*. La propriété *subClasseOf* permet de relier une classe spécifique *(BiologieStructurale*) à une classe plus générale (*Biologie*).

Exemple:

```
<owl:Class rdf : ID ="BiologieStructurale">
        <rdfs:subClassOf rdf:resource="# Biologie" />
</owl>
```

Propriétés : Dans le langage OWL, une propriété permet de définir des faits ou des relations entre des classes. Il existe en OWL deux types de propriétés : propriétés d'objet (*owl : ObjectProperty*) qui définissent une propriété entre deux individus d'une classe ou de plusieurs classes et les propriétés de type de données (*DataTypeProperty*) qui relient des instances à des valeurs de données).

Instances de classe : L'ensemble des individus d'une classe est désigné par le terme extension de classe, chacun de ces individus étant alors une instance de la classe. Les instances sont utilisées pour représenter les éléments spécifiques.

2.2. Etat de l'art de distance sémantique

Les différentes stratégies et méthodologies utilisées pour calculer la distance sémantique ont pour but de donner une valeur quantitative à la similarité entre les concepts. La détermination de la distance sémantique entre concepts dans une ontologie est un problème qui se pose dans

beaucoup de domaines de l'informatique : recherche d'information, intelligence artificielle, bases de données, classification, clustering de textes utilisant le Traitement du Langage Naturel, indexation automatique, etc. Diverses propositions ont été faites dans la littérature pour le calcul de distance sémantique mais la plupart ne respectent pas les qualités attendues d'une distance.

Cependant, la démarche la plus intuitive pour évaluer la distance sémantique dans une ontologie entre deux concepts, consiste à déterminer le nombre d'arcs séparant les concepts par le chemin le plus court dans la hiérarchie (treillis de types de concepts de Sowa) de concepts (*is-a*).

Dans la communauté d'*Intelligence artificielle*, la majorité des travaux de recherche sur les méthodes de calcul de distance sémantique a été réalisée dans le domaine des graphes conceptuels. Un état de l'art est présenté par [Ranwez, 2000] sur le calcul de distance sémantique sur les graphes conceptuels : distance entre types, distance entre concepts et distances entre graphes conceptuels.

En *recherche d'information*, diverses méthodes de calcul de similarité entre documents ont été proposées. Dans ce domaine, les documents sont représentés sous forme de vecteurs de termes. Ces méthodes basées sur la correspondance entre les termes présentent des inconvénients, puisqu'un concept peut être représenté par un ou plusieurs termes et un terme peut avoir plusieurs sens, ce qui est à l'origine du *silence*. Pour pallier ces problèmes [Deerwester & al, 1990] propose la méthode LSI qui a pour objectif de déterminer une nouvelle représentation des termes pour évaluer les affinités sémantiques entre les mots. LSI est issu des études de LSA en psycholinguistique. Ces deux méthodes de calcul de distance sémantique ont été utiles dans le domaine biomédical pour la recherche d'information et la classification des termes médicaux [Pedersen et al, 2006]. La vectorisation de documents est aussi utilisée en TLN dans des tâches telles que la vérification orthographique, désambiguïsation des sens de mots, etc.

Nous avons aussi des mesures de distance utilisant des ontologies comme WordNet [Budanitsky et Hirst, 2000]. WordNet peut être considéré comme un réseau sémantique où chaque nœud représente un concept du monde réel. Chaque nœud est composé d'un ensemble de synonymes qui représentent le même concept, cet ensemble s'appelle synset. Les synsets sont reliés par des arcs qui décrivent les relations entre les différents concepts. [Resnik, 1999] introduit la notion de contenu informationnel, qui traduit la pertinence d'un concept dans le corpus en tenant compte de sa spécificité ou généralité. Il définit la similarité sémantique entre deux concepts par la quantité d'information qu'ils partagent. Cette information partagée

est égale au contenu informationnel du plus petit généralisant commun. C'est dans ce contexte que [Zargayouna et Salotti, 2004] propose une mesure de similarité basée sur la notion du plus petit généralisant commun et la spécificité des concepts par rapport au plus bas concept de l'ontologie (bottom). Il utilise cette combinaison pour pallier les limites des mesures qui ne dépendent que du subsumeur le plus spécifique.

La *communauté Web sémantique*, en particulier les recherches intéressées par les ontologies, ont proposé différents algorithmes pour déterminer la distance sémantique entre deux concepts. [Shu et al, 2006] définit la distance sémantique entre deux concepts d'une ontologie comme étant la somme de la *distance de subsumption* et la *distance de définition*. La distance de subsumption est la distance entre deux concepts dans la hiérarchie de concepts et la distance de définition est la différence de la description sémantique entre les deux concepts. Il donne un algorithme pour le calcul de ce dernier. Mais, précise qu'en général, la fonction de distance n'est pas symétrique ce qui est contradictoire avec ce que nous attendons comme propriété de distance sémantique. D'autres algorithmes de calcul de distances ont été aussi proposés par [Cordi et al, 2005].

D'après l'état de l'art, nous pouvons classer les distances en trois catégories d'approches :

- Approche basée sur les arcs, c'est le cas où on utilise les graphes dont les nœuds sont des concepts.
- Approche basée sur le contenu en information d'un concept, cas où on utilise le contenu d'information.
- Approche basée sur les vecteurs de termes, cas utilisant la vectorisation des documents.

Cependant, la plupart de ces mesures de distance sont basées sur le plus court chemin entre concepts et ne sont pas satisfaisantes car ne tenant pas compte du degré de détail de l'ontologie. D'autres sont basées sur le plus petit généralisant commun or dans une ontologie les concepts ont souvent plusieurs parents.

Après une étude approfondie sur les mesures de distance sémantique, [Ranwez et al, 2006] propose une approche de mesure de distance basée sur la relation *is-a* des ontologies. Cette mesure utilise les liens existant dans l'ontologie et leur sémantique et respecte les trois propriétés d'une distance sémantique. Cette solution de calcul de distances sémantique est celle que nous utilisons, elle est donc détaillée dans la suite.

2.3. Distance proposée par [Ranwez et al, 2006]

Dans cette section, nous allons détailler la distance sémantique proposée par [Ranwez et al, 2006] sur laquelle se base notre étude.

2.3.1. Approche intuitive

D'une manière intuitive, deux concepts sont d'autant plus proches que le concept qui les subsume est légèrement plus général; pour estimer ce degré de généralisation nous considérons le nombre de concepts qui les entourent. Dans le cas simple où l'ontologie est un arbre et le concept *a* subsume le concept *b*, nous voulons que la distance de *a* à *b* soit le nombre de concepts entourés par *a* mais pas *b*. Ce nombre de concepts est ainsi utilisé pour estimer le degré de généralisation de *a* comparé à *b*.

Dans une hiérarchie de concepts supportant l'héritage multiple, on peut regarder les généralisants d'un concept en différents points de vue.

Ces deux notions intuitives de distance nous permettent de définir une mesure de distance entre deux concepts *a* et *b* qui prend en compte les différents points de vue d'un concept.

2.3.2. Définition

Différents types de relations lient les concepts dans une ontologie ; la similarité peut être estimée sans tenir compte de leur lien sémantique. Parmi ces relations la relation *is-a* définissant la généralisation de concepts, joue un rôle crucial dans les ontologies. Dans une ontologie un concept peut être une spécialisation de plusieurs concepts, la partie *is-a* de l'ontologie peut être représentée par un DAG dont les nœuds représentent les concepts de l'ontologie et les arcs orientés représentent la relation de spécialisation.

Nous allons donner quelques définitions importantes pour le calcul de la distance.

Soit un graphe (V,E) où V est l'ensemble des sommets et E l'ensemble des arcs, a est père de b si l'arc (a,b) appartient à E et a est ancêtre de b s'il existe un chemin entre a et b. L'ensemble des concepts qui ont comme ancêtre a est noté par $desc(a)$ et l'ensemble de ses ancêtres est noté par $ansc(a)$. Pour deux concepts a et b donnés, x est leur ancêtre exclusif s'il est exactement l'ancêtre de l'un d'entre eux c'est-à-dire si x est élément de $ansc(a) \cup ansc(b) - ansc(a) \cap ansc(b)$. L'ensemble des ancêtres exclusifs de a et b est noté par $anscEx(a,b) = anscEx(b,a)$.

Nous utilisons $d_{ISA}(a, b)$) pour noter la distance entre les concepts a et b basée sur la relation *is-a*. La distance est définie par la relation :

$$d_{ISA}(a, b) = | desc(ancExc(a, b)) \cup desc(a) \cup desc(b) - desc(a) \cap desc(b) |$$

Les trois propriétés de distance à savoir la positivité, la symétrie et l'inégalité triangulaire sont vérifiées par cette mesure de distance. La preuve est présentée dans [Ranwez et al, 2006].

Disposant d'une ontologie et de la distance sémantique qui respecte les propriétés d'une distance, nous allons projeter l'ontologie sur une carte conceptuelle (Molage) en calculant les distances entre les concepts. Pour être sûr des liens sémantiques entre les concepts de chaque groupe de la carte conceptuelle, nous allons utiliser les techniques de classification traditionnelles avec cette distance (Chapitre 3). Le but de ce travail consiste à évaluer la pertinence de cette distance sémantique afin de l'utiliser dans des projets industriels. Parmi ces derniers, nous pouvons noter l'environnement de cartographie Molage qui nous sera très utile pour visualiser des ontologies de domaines sur lesquelles le laboratoire travaille. C'est l'exemple de l'ontologie du projet TOXNUC-E et celle de la musique.

Chapitre 3 : Clustering

Nous rappelons que notre objectif est de visualiser une ontologie sur une carte conceptuelle. Pour donner sens à cette visualisation, nous avons utilisé la distance de [Ranwez, 2006] lors de la projection des concepts. Afin d'évaluer le paysage obtenu avec la distance sémantique, nous avons besoin des techniques de regroupement des concepts de l'ontologie en tenant compte de la sémantique, ce qui est possible avec les méthodes traditionnelles de clustering qui utilisent la plupart du temps la notion de distance sémantique.

3.1. Rappels sur la classification

De manière générale, les méthodes de classification ont pour but d'identifier les classes auxquelles appartiennent des objets à partir de certains traits descriptifs. Cependant, il existe deux variantes de classification : la classification dite *supervisée* et la classification *dite non supervisée*.

Dans l'approche de la classification *supervisée*, les classes existent a priori. On dispose en donnée du problème d'un ensemble de classes et d'objets, chacun d'eux étant déjà placé dans une classe qui lui convient au mieux. Le but de cette démarche est de pouvoir allouer une classe à un nouvel objet, en restant le plus cohérent possible avec la structure initiale en classes.

Dans une classification *non supervisée*, les classes sont encore inexistantes. Pour cette démarche, on dispose donc au départ d'un ensemble d'objets. L'idée consiste à découper l'ensemble des objets en groupes (clusters) de telle sorte que les caractéristiques des objets dans un même cluster soient similaires et les caractéristiques des objets dans des clusters différents soient distinctes. Dans la suite de ce chapitre nous allons nous intéresser exclusivement à l'approche *non supervisée*, encore nommée *clustering* ou *segmentation*. Cette approche est mieux adaptée à nos besoins, nous allons développer quelques une de ses caractéristiques dans ce qui suit.

3.2. Application du clustering

D'après la section précédente, nous venons de voir qu'appliquer une méthode de clustering revient à "mettre de l'ordre" dans un jeu de données. Ce qui est largement abordé par divers métiers. Par exemple, regrouper les clients d'une société par profil, rassembler des malades présentant les mêmes symptômes, classer des documents etc. Dans notre cas, nous allons

appliquer les méthodes de clustering sur une ontologie afin de regrouper les concepts similaires ensemble et de les projeter sur une carte conceptuelle (Molage).

3.2.1. Rapprochement de deux clusters

Un des problèmes rencontrés au niveau du clustering est le choix d'une technique de rapprochement des clusters. Le rapprochement entre deux clusters consiste à déterminer quels concepts prendre dans chacun des clusters pour définir la mesure de distance [Beck, 2006]. Cependant, ce rapprochement se fait sur la base de mesures de distances calculées à partir des valeurs prises par les concepts du cluster. Plusieurs techniques de rapprochement entre clusters existent, parmi celles-ci nous pouvons citer :

- **lien simple** (Single link) : il définit la distance entre deux clusters comme étant la plus petite distance entre toutes les paires de concepts des deux clusters.

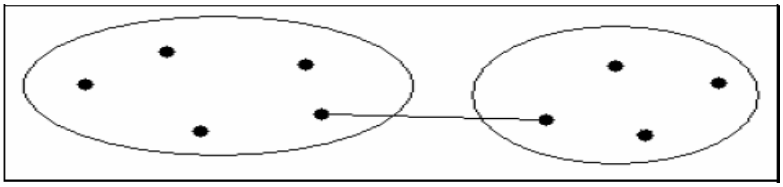


Figure 3.1 : Distance entre les deux clusters par lien simple.

- **Lien complet** (Complete link) : il définit la distance entre deux clusters comme étant la plus grande distance entre toutes les paires de concepts des deux clusters.

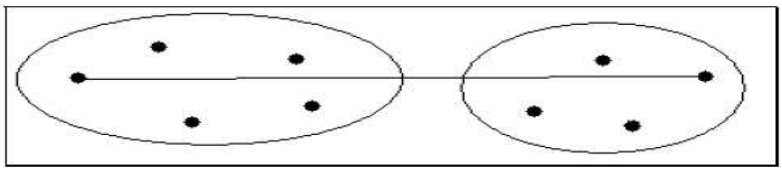


Figure 3.2 : Distance entre les deux clusters par lien complet.

- **Lien moyen** (Average Link) : il définit la distance entre deux clusters en faisant intervenir tous les objets présents dans ces clusters.

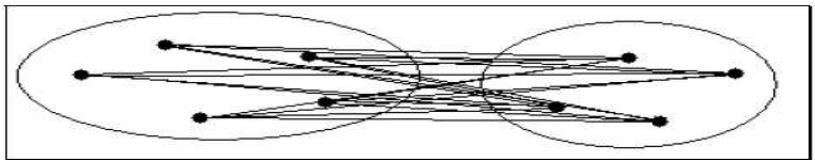


Figure 3.3 : Distance entre les deux clusters par lien moyen.

- **Lien moyen de groupe** (Group average link) : il définit la distance entre deux clusters comme étant la distance entre les centres des clusters.

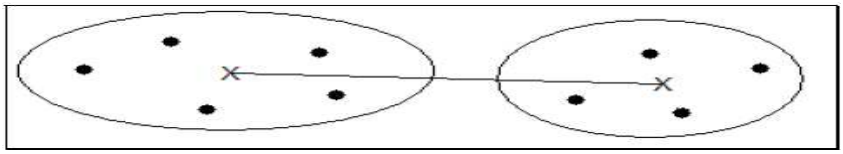


Figure 3.4 : Distance entre les deux clusters par lien moyen de groupe.

3.3. Clustering hiérarchique

La méthode hiérarchique génère une partition de l'espace des données, mais aussi une succession de partitions de l'espace des données. Celles-ci sont souvent représentées sous la forme d'un dendrogramme. Un dendrogramme est un arbre de partitions successives de l'espace des données (Fig. 3.5). Selon que l'on parcourt le dendrogramme de haut en bas ou de bas en haut, la méthode sera dite *descendante* (division) ou *ascendante* (agglomération).

Le principe de la méthode hiérarchique descendante consiste à commencer avec un groupe contenant tous les concepts de l'ontologie. Puis de procéder à une division du groupe en plus petits groupes jusqu'à ce que chaque groupe ne contienne plus qu'un exemple.

Dans la méthode hiérarchique ascendante, on commence avec autant de clusters qu'il y a de concepts. Ensuite, à chaque étape, on regroupe les deux clusters qui sont jugés les plus similaires pour terminer avec un seul grand cluster englobant toutes les données. Nous avons utilisé cette dernière, qui suit l'algorithme suivant :

- Chaque concept est placé dans son propre cluster
- Calcul de la matrice de distance M entre chaque couple de clusters
- Répéter
    - Sélectionner dans M les deux clusters les plus proches
    - Fusionner les clusters par un cluster plus général

- Mettre à jour M en calculant la distance entre le cluster créé et les autres clusters existants (techniques de rapprochements entre deux clusters)

- Jusqu' à k (avec k le nombre de regroupements à effectuer).

Algorithme 3.1 : Clustering hiérarchique ascendant.

Afin de déterminer le nombre optimal de clusters, on coupe l'arbre (dendrogramme) au moment où l'on commence à rassembler des clusters dits similaires. Comme la distance entre clusters varie en fonction de l'ontologie utilisée, dans le cas du dendrogramme suivant représentant celui de la taxonomie de la Fig. 4.4, nous avons fixé la *distance minimale* entre cluster, à trois afin d'obtenir le nombre optimal de clusters. La coupe est présentée en traits pointillés sur la Fig. 3.5. Dans notre cas, nous avons utilisé un paramètre en entrée qui représente le nombre de nouveaux clusters créés. Donc le *nombre de groupe de concepts* est égal à la différence entre le nombre de concepts de l'ontologie et le nombre de nouveaux clusters créés.

Pour le problème de technique de rapprochement entre les clusters, nous avons utilisé le principe du lien simple décrit dans la section précédente.

De manière générale, la méthode hiérarchique ascendante est très intuitive et facile à implémenter mais coûteuse à partir du moment où les clusters sont comparés deux à deux à chaque étape ($n\ (n-1)/2$ calculs effectués à chaque étape avec n le nombre de clusters).

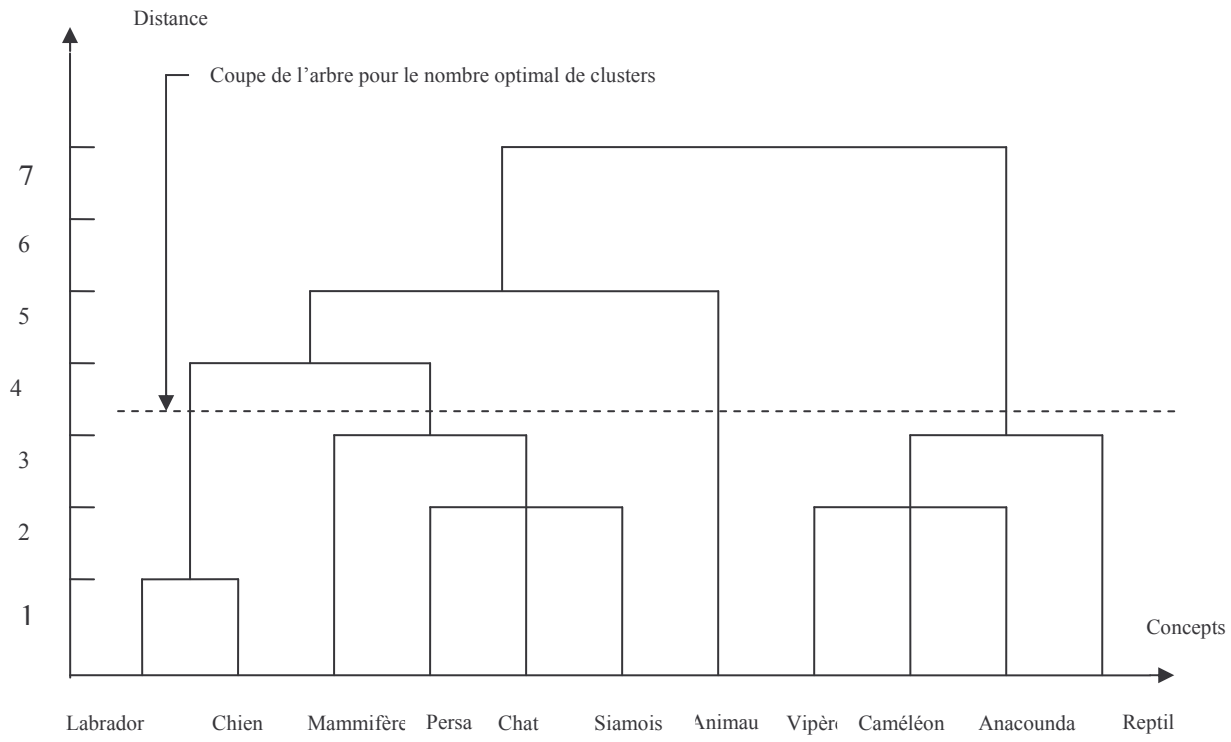


Figure 3.5 : Exemple de dendrogramme.

3.4. Clustering K-Means

La méthode de k- means ou algorithme de regroupement par centre mobile permet d'effectuer une classification d'un ensemble de données en k clusters. Un cluster regroupe plusieurs concepts similaires. Chaque cluster est décrit par son centre. Les centres des clusters sont mobiles au cours de l'exécution de l'algorithme.

L'algorithme peut se présenter comme suit:

- Le nombre de clusters, le paramètre k, est fourni au départ,
- Un ensemble de k centres est choisi aléatoirement dans l'ensemble des données,
- Les k clusters sont formés en regroupant dans chaque centre l'ensemble des données plus proches du centre courant que de tout autre centre,
- Le centre de chaque cluster est calculé et devient le nouveau centre,
- L'algorithme boucle alors sur l'étape précédente : les données sont réaffectées en fonction de ces nouveaux centres et la condition d'arrêt est que les centres deviennent immobiles.

Algorithme 3.2 : Clustering k-means.

Nous avons appliqué cet algorithme dans le contexte des ontologies (les données sont les concepts de l'ontologie). Cependant, le problème de calcul du centre d'un cluster se pose car dans les méthodes usuelles de k-means, le centre d'un cluster se calcule à partir des coordonnées des données (exemple des algorithmes sous Weka[2]). Or la notion de cordonnées n'existe pas dans la structure d'une ontologie. C'est ainsi que nous avons opté pour la démarche suivante pour le calcul du centre d'un cluster : pour un cluster, on calcule la somme des distances de chaque concept par rapport aux autres concepts. On prend le premier concept correspondant à la plus petite somme comme nouveau centre du cluster. Cette méthode converge et est adaptée aux ontologies.

De manière générale, l'algorithme de clustering du k-means est efficace, mais présente quelques faiblesses comme le nombre de clusters k qui doit être fixé à priori, ce qui oblige à initialiser les centres des clusters. L'initialisation des centres de clusters conditionne le résultat final (des initialisations différentes peuvent mener à des clusters différents, c'est l'exemple des figures 4.7a. et 4.7b. Il peut arriver qu'un cluster ne contienne que son centre (cluster *Mammifere* de la Fig. 4.8).

[2] http://www.cs.waikato.ac.nz/ml/weka/

Chapitre 4 : Contributions

Dans ce chapitre, nous exposerons l'architecture fonctionnelle de notre système, les différentes applications utilisant des ontologies que nous avons utilisées. Nous présenterons également notre outil de visualisation d'ontologies (Molage) et enfin, les différents résultats et une discussion sur ces derniers.

4.1. Architecture fonctionnelle

La figure ci-dessous représente l'architecture simplifiée de notre application de calcul de distances. Dans un premier temps le module java développé grâce à l'API Jena prend en entrée une ontologie OWL, calcule la matrice de distances entre concepts et la sauve sous la forme d'un fichier XML. Cette dernière opération est faite par une classe java appelée XMLWriter. Nous avons pris le format des fichiers utilisés par Molage pour sauver notre fichier XML. Nous avons sauvé les concepts en tant qu'entités graphiques élémentaires, les relations entre concepts en tant que liaisons et la distance entre deux concepts en tant que longueur des liaisons. Dans un second temps le fichier XML est utilisé par Molage afin de visualiser l'ontologie. Cette même matrice de distances est utilisée par les algorithmes de clustering (hiérarchique et k-means) avant de passer par XMLWriter.

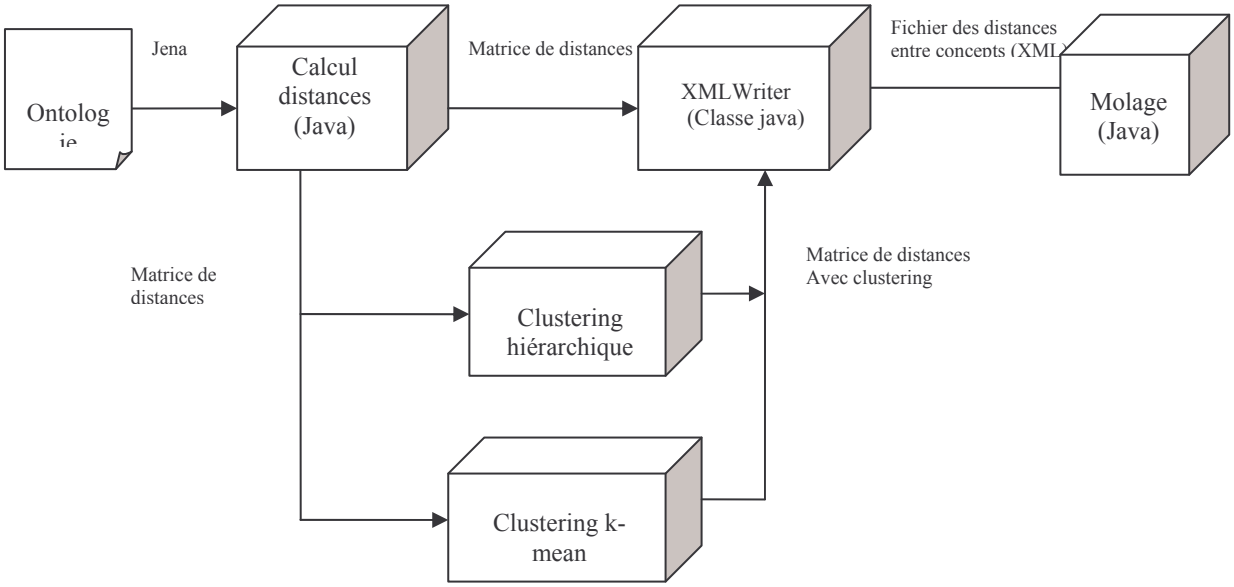


Figure 4.1 : Architecture simplifiée de l'application.

4.2. Mise en œuvre d'ontologies OWL

Les environnements logiciels permettent tout à la fois, la création, l'édition, le partage, la réutilisation et la mise en exploitation des ontologies. Beaucoup de ces logiciels sont open source, cependant les critères de choix d'un environnement particulier vont porter, par exemple, sur la méthodologie associée, la facilité de travailler en collaboration, les langages de représentation de connaissances afférents, etc.

Nous citons quelques environnements logiciels que nous avons utiliser pour la validation de notre distance sémantique.

4.2.1. Protégé 2000

Protégé 2000[3] est un éditeur d'ontologies distribué en open source par l'université d'informatique médicale de Stanford. Il permet de construire et d'éditer des ontologies sous divers formalismes RDF, DAML+OIL, OWL. De plus, Protégé est extensible en utilisant des plug-ins et des APIs java pour le développement d'applications particulières. L'interface de Protégé est assez simple, l'ensemble des fonctionnalités de l'éditeur étant regroupé en onglets. Le premier onglet présente les classes de l'ontologie. Le deuxième onglet permet de créer des instances et de leur affecter des propriétés. L'onglet des requêtes permet d'effectuer des requêtes sur l'ontologie en cours d'édition.

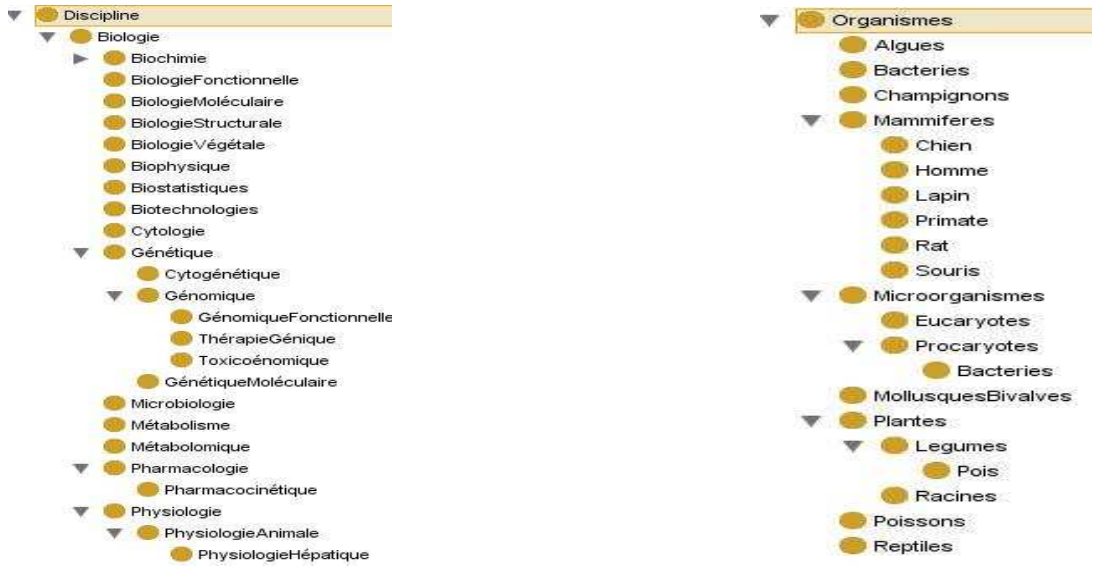


Figure 4.2 : Une partie de l'ontologie de TOXNUC sous Protégé 2000.

[3] http://protege.stanford.edu

4.2.2. Jena

Le framework Jena[4] est une API java permettant de lire et de manipuler des ontologies décrites en document OWL, RDF (S), DAML+OIL et d'y appliquer certains mécanismes d'inférences. Jena est diffusé en code source libre. Les primitives pour lire et parcourir des ontologies OWL constituent l'outil le plus utile pour le calcul de notre distance. Nous allons esquisser les éléments les plus importants de ces primitives.

Le package *com.hp.hpl.jena.ontology* fournit un ensemble de classes pour manipuler et accéder aux ontologies représentées en OWL.

Voici par exemple des primitives pour la lecture d'une ontologie OWL.

```
// Création d'un modèle d'ontologie m, dans notre cas OWL
OntModel m = ModelFactory.createOntologyModel( OntModelSpec.OWL_MEM, null
);

// Chargement et création d'une copie locale de notre ontologie ToxNuc07.owl
m.getDocumentManager().addAltEntry(http://www.w3.org/TR/2003/CR-owl-guide
20030818/ ToxNuc07 ,"file:./ontologieOWL/ ToxNuc07.owl ");

// Lecture de l'ontologie avec la méthode read
m.read( "http://www.w3.org/2001/sw/WebOnt/guide-src/ ToxNuc07" );
```

Ainsi nous venons de créer et de charger une copie de l'ontologie *ToxNuc07.owl*. Pour lister les classes d'une ontologie, la méthode *listClasses ()* retourne un itérateur des objets de type *OntoClasses*. Voici un exemple pour lister les classes.

```
// Liste des classes
for (Iterator i = m.listClasses();  i.hasNext(); ) { // Prendre toutes les classes de
l'ontologie
        OntoClass cls = (OntClass) i.next();
// Nom local de la classe par la méthode getLocalName())
System.out.println(cls.getLocalName());
}
```

[4] http://jena.sourceforge.net

Avec une *OntoClass* nous pouvons obtenir les super classes et les sous-classes avec les méthodes *listSuperClasses()* et *listSubClasses()*. Ces deux méthodes retournent des itérateurs contenant des *OntClass*, il est possible ainsi de parcourir la hiérarchie des classes d'une ontologie. Le bout de code suivant imprime pour chaque classe de l'ontologie ses super classes et ses sous-classes.

```
for (Iterator i = m.listClasses();   i.hasNext();) { // Prendre toutes les classes de
l'ontologie
        OntoClass cls = (OntClass) i.next();// Prendre une OntClass

 // Prendre les super classes de la classe cls
for(Interator i = cls.listSuper.Classes() ;i.hasNext() ;){
        Sysem.out.print(((OntClass) i.next()).getLocalName()+" ");
}

 // Prendre les sous- classes de la classe cls
for (Iterator i = cls.listSubClasses( ); i.hasNext(); ) {
        Sysem.out.print(((OntClass) i.next()).getLocalName()+" ");
 }
```

En utilisant ces méthodes pour trouver les *sous–classes* et les *super classes*, il est possible avec l'aide de la récursivité de trouver tous les descendants ou ancêtres d'un concept. Nous allons utiliser cette philosophie pour le calcul de notre distance sémantique qui est basée sur le calcul des descendants et ancêtres de deux concepts.

4.3. Visualisation d'ontologies (Molage)

Avec l'accroissement des ontologies en volumes et en complexité, il devient de plus en plus difficile pour l'utilisateur de s'y retrouver. La problématique consiste donc à visualiser l'ontologie sous le meilleur format possible qui montrera le plus fidèlement ses caractéristiques. Pour remédier à ce problème, nous avons fait appel à la discipline de la visualisation de l'information. La visualisation d'information est utilisée dans plusieurs domaines de l'informatique (recherche d'information, sciences cognitives, cartographie, etc.). Elle a pour objectif de trouver un sens parmi de grands volumes de données.

Ainsi, dans le cadre de projets industriels, le Centre de Recherche LGI2P a développé un environnement de cartographie de connaissance appelé Molage [Crampes et al, 2006]. L'environnement permet de manipuler une vaste collection de titres musicaux (navigation, indexation, fouille du paysage) de façon visuelle. Molage est écrit en java et les fichiers d'échange sont en XML (Fig. 4.3).

Dans le cadre de ce rapport, nous allons utiliser l'application pour projeter l'ontologie tout en utilisant la distance sémantique proposée à la section 2.3 du chapitre 2 afin de donner un sens au paysage obtenu.

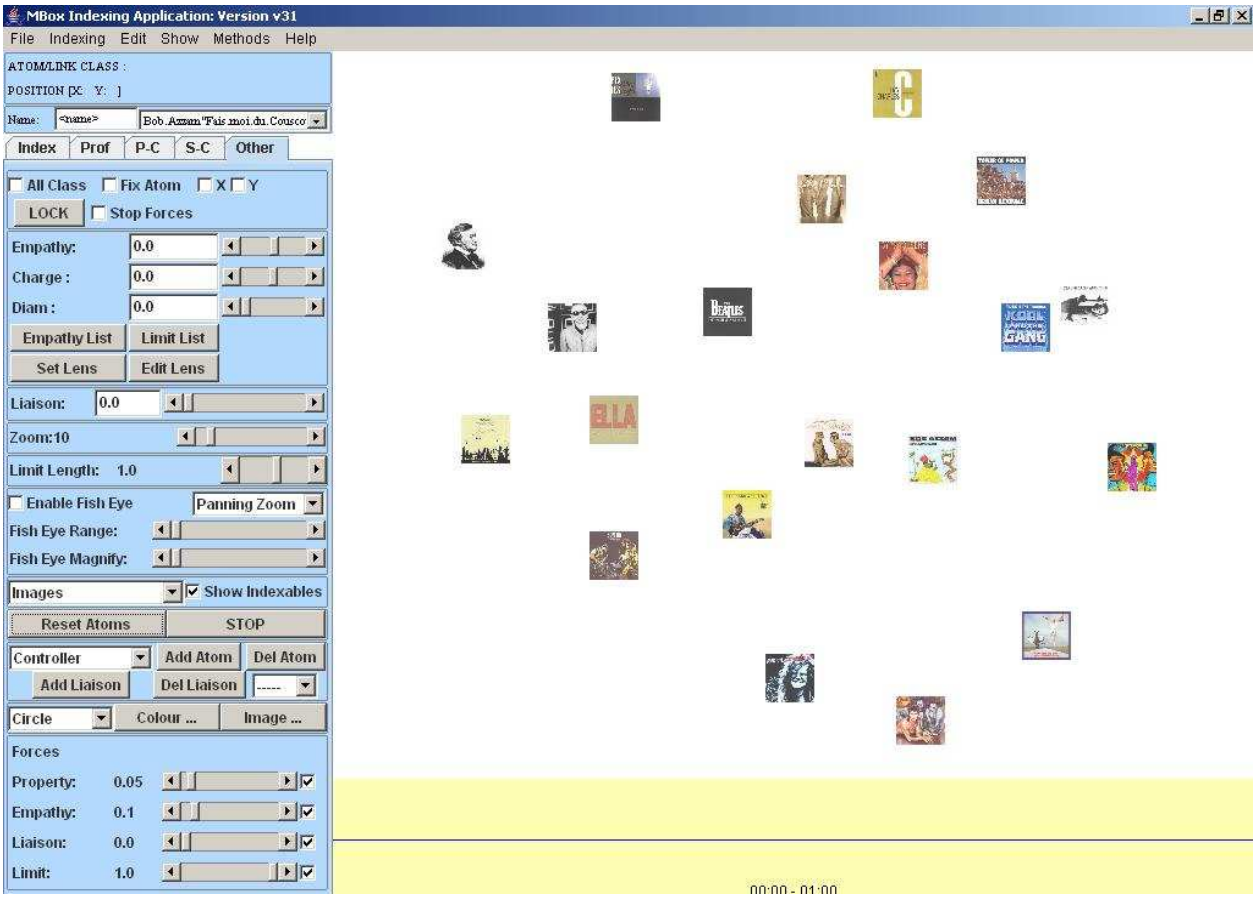


Figure 4.3 : Molage.

4.4. Résultats et Discussion

Pour évaluer l'approche de distance sémantique proposée dans [Ranwez et al, 2006] sur les ontologies, nous avons dans un premier temps utilisé l'exemple de taxonomie de la Fig. 4.4 et

dans un second temps, une partie de l'ontologie du projet TOXNUC-E constituée de 118 concepts.

4.4.1. Résultats avec une taxonomie simple

Dans cette section, nous allons nous appuyer sur l'exemple de taxonomie ci-dessous, basée sur la relation *is-a* afin d'illustrer nos propos [Hernandez, 2006]. Les concepts sont représentés par des rectangles et les flèches symbolisent la relation *is-a*.

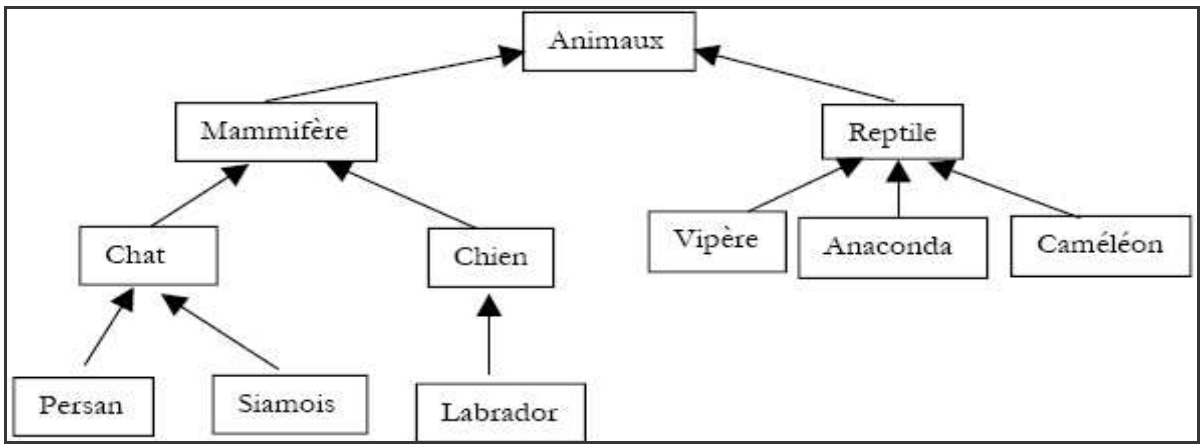


Figure 4.4 : Exemple de taxonomie.

- **Projection des concepts sous Molage**

Dans cette partie nous avons directement calculé les distances entre concepts et projeté l'ontologie sous Molage tout en respectant les distances entre concepts. Ce qui nous donne le résultat suivant :

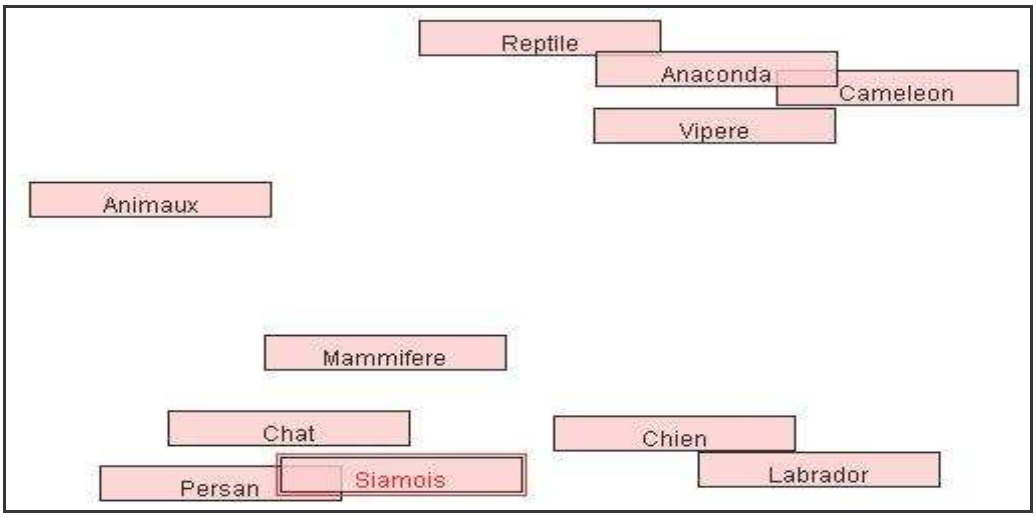


Figure 4.5 : Visualisation utilisant la distance.

Dans ce paysage, nous remarquons que le regroupement des concepts de l'ontologie est comparable à l'intuition humaine. Nous avons deux groupes, le groupe des *Reptiles*, celui des *Mammifères* et le groupe *Animaux* (subsument des deux principaux groupes). Le cluster des *Mammifères* peut être considéré comme deux clusters (cluster des *Chats* et celui des *Chiens*). D'après ces observations nous pouvons dire que la distance que nous avons respecte la sémantique des concepts, car dans le paysage les concepts qui ont des liens sémantiques forts sont groupés ensemble. Nous allons vérifier ces observations en appliquant les techniques de clustering dans la section suivante.

- **Visualisation utilisant le clustering hiérarchique**

Dans cette section, nous avons appliqué la méthode de clustering hiérarchique ascendante sur la matrice de distance obtenue avant projection de l'ontologie sous Molage afin de comparer le paysage obtenu avec celui utilisant la distance (Fig. 4.5). Après avoir effectué cette opération et projeté le résultat sous Molage, voici le paysage que nous avons :

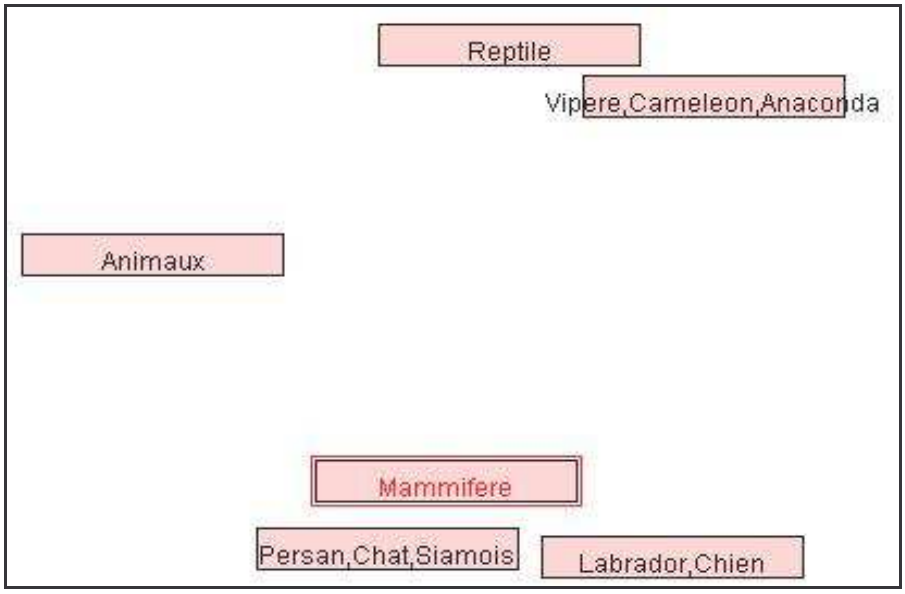


Figure 4.6 : Visualisation utilisant le clustering hiérarchique ascendant.

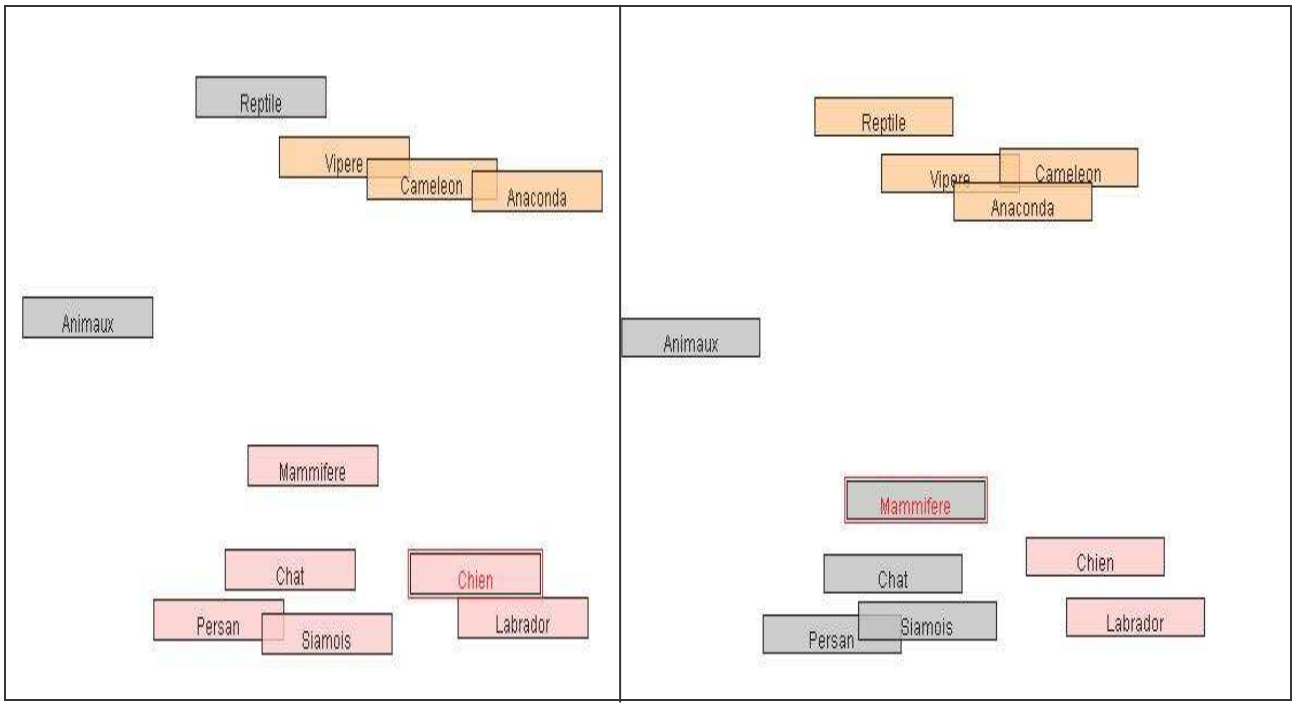
D'après ce paysage, nous remarquons que le principe de la méthode de clustering hiérarchique ascendante est bien appliqué c'est-à-dire de grouper les concepts les plus similaires comme nouveau concept. C'est le cas des concepts *Vipere*, *Cameleon* et *Anaconda* qui étaient très proche sur la Fig.4.5 et qui sont considérés comme nouveau concept après application de l'algorithme de clustering. Ce paysage comparé à celui obtenu sans application

de l'algorithme de clustering, ne présente pas de grandes différences. Nous retrouvons presque le même nombre de clusters dans les deux paysages.

- **Visualisation utilisant le clustering k-means**

Dans cette section, nous avons appliqué la méthode de clustering du k-means sur la même matrice de distances comme dans la section précédente.

Dans un premier temps, nous avons choisi k (nombre de clusters) égal à trois et les centres des trois clusters sont positionnés à *Mammifere*, *Reptile* et *Animaux* pour le paysage de la Fig. 4.7.a et à *Chien*, *Chat*, *Vipere* pour celui de la Fig. 4.7.b. L'algorithme boucle deux fois et nous donne respectivement comme nouveaux centres *Chat*, *Vipere*, *Animaux* pour le premier paysage et *Labrador*, *Chat*, *Vipere* pour le second.



4.7.a 4.7.b

Figure 4.7 : Visualisation utilisant k-means avec (k = 3).

Ces deux paysages comportant des clusters différents nous montrent que l'initialisation des centres conditionne le résultat final pour la méthode de clustering du k-means. Cependant, le paysage de la Fig. 4.7.a est comparable au paysage de la Fig.4.6 utilisant le clustering hiérarchique ascendante.

Pour vérifier si l'algorithme de clustering k-means tient compte de la sémantique des concepts, nous allons augmenter la valeur de k à cinq et positionner les centres à *Mammifere*, *Chat*, *Chien*, *Reptile* et *Animaux*. Ce qui nous donne le paysage suivant :

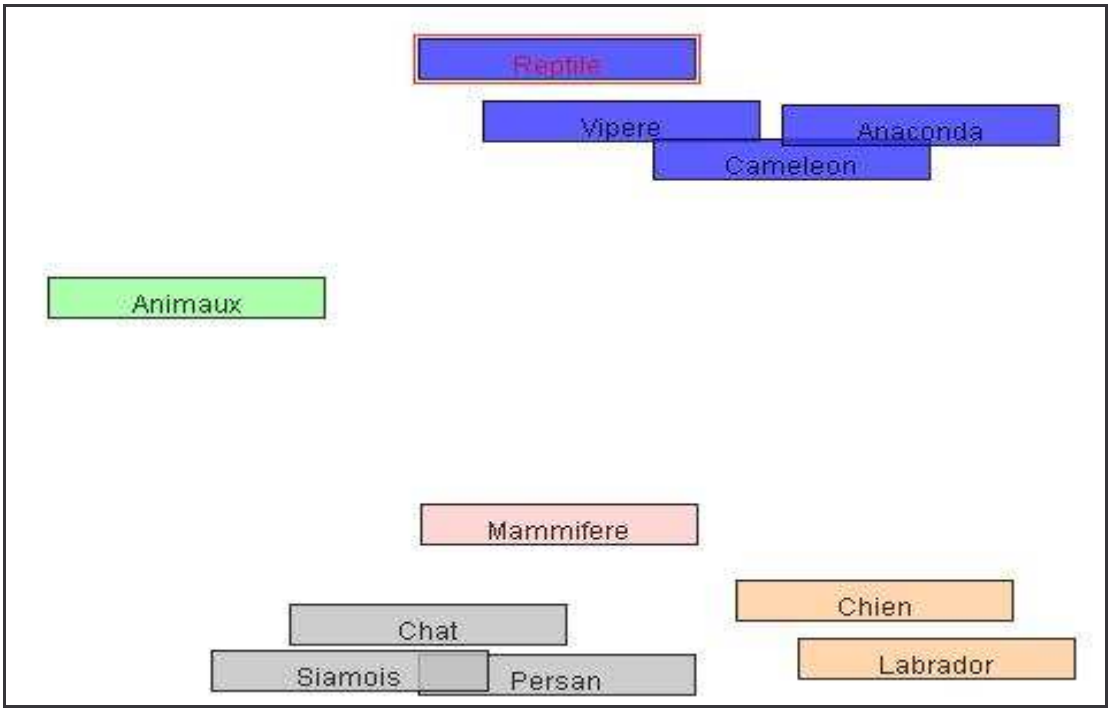


Figure 4.8 : Visualisation utilisant k-means avec (k = 5).

Ce paysage confirme l'intuition que nous avons sur le paysage de la Fig.4.5, c'est-à-dire les concepts qui ont des liens sémantiques forts sont dans un même cluster. C'est ainsi que nous avons *Reptile* et ses descendants (*Vipere*, *Anaconda* et *Caméléon*) dans un même cluster. *Mammifere* et ses descendants sont divisés en trois clusters à savoir le cluster *Mammifere* ne contenant que le centre, *Chat*, *Siamois*, *Persan*, et enfin le cluster *Chien*, *Labrador*.

Dans ce paysage, nous remarquons que chaque cluster est constitué d'un ascendant et ses descendants ce qui est comparable à l'intuition humaine, car il existe une forte similarité entre un concept et ses descendants.

Le paysage obtenu en projetant directement l'ontologie sous Molage comparé avec les paysages obtenus en appliquant les techniques de clustering (hiérarchique ascendante et k-means), montre que la distance que nous avons respecte la sémantique des concepts.

D'après ce qui précède, nous pouvons en déduire que, les résultats des ontologies de petite taille sont satisfaisants. Pour être sûr de la validité de notre distance et de notre outil de visualisation, nous allons appliquer le même processus avec une ontologie de grande taille. Ceci constitue l'objectif de la section suivante.

4.4.2. Résultats avec l'ontologie de TOXNUC-E

Nous rappelons que l'ontologie du projet TOXNUC-E est basée sur la relation *is-a*. La partie que nous avons utilisée, est constituée de trois sous ontologies qu'on peut nommer : *Discipline*, *Organismes* et *ModeleBiologiques*.

Nous remarquons des héritages multiples entre les concepts des trois sous ontologies, c'est le cas du concept *Algues* descendant de *Organismes* et *ModeleBiologiques*.

Dans ce qui suit, nous faisons l'interprétation sur les concepts visibles sur la carte car un défaut de notre outil de visualisation est qu'il superpose les concepts qui sont très proches.

- **Projection des concepts sous Molage**

Après projection de la matrice des distances de l'ontologie du projet TOXNUC-E sous Molage, nous avons le paysage de la figure ci-dessous.

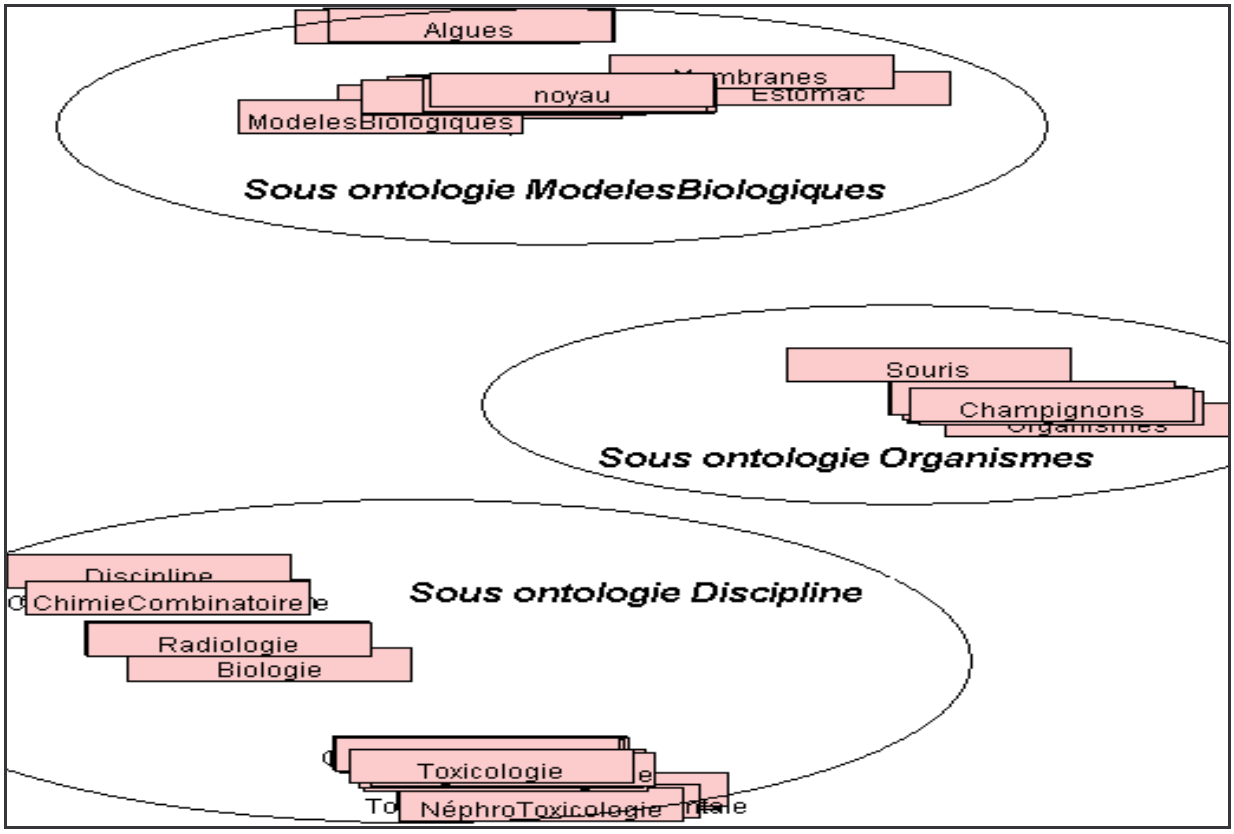


Figure 4.9 : Visualisation utilisant des concepts de TOXNUC.

Dans ce paysage, les concepts de chaque sous ontologie sont groupés en un cluster. Mais le concept *Algues* descendant de *Organisme* et *ModeleBiologiques* que nous attendons entre ces deux derniers, est plus proche de *ModeleBiologiques* que d'*Organismes*.

Nous allons voir les résultats des techniques de clustering.

- **Visualisation utilisant le clustering hiérarchique**

L'application du clustering hiérarchique nous permet non seulement de grouper les concepts plus proches de l'ontologie en un seul concept mais aussi d'avoir une idée sur les concepts superposés à la Fig. 4.9.

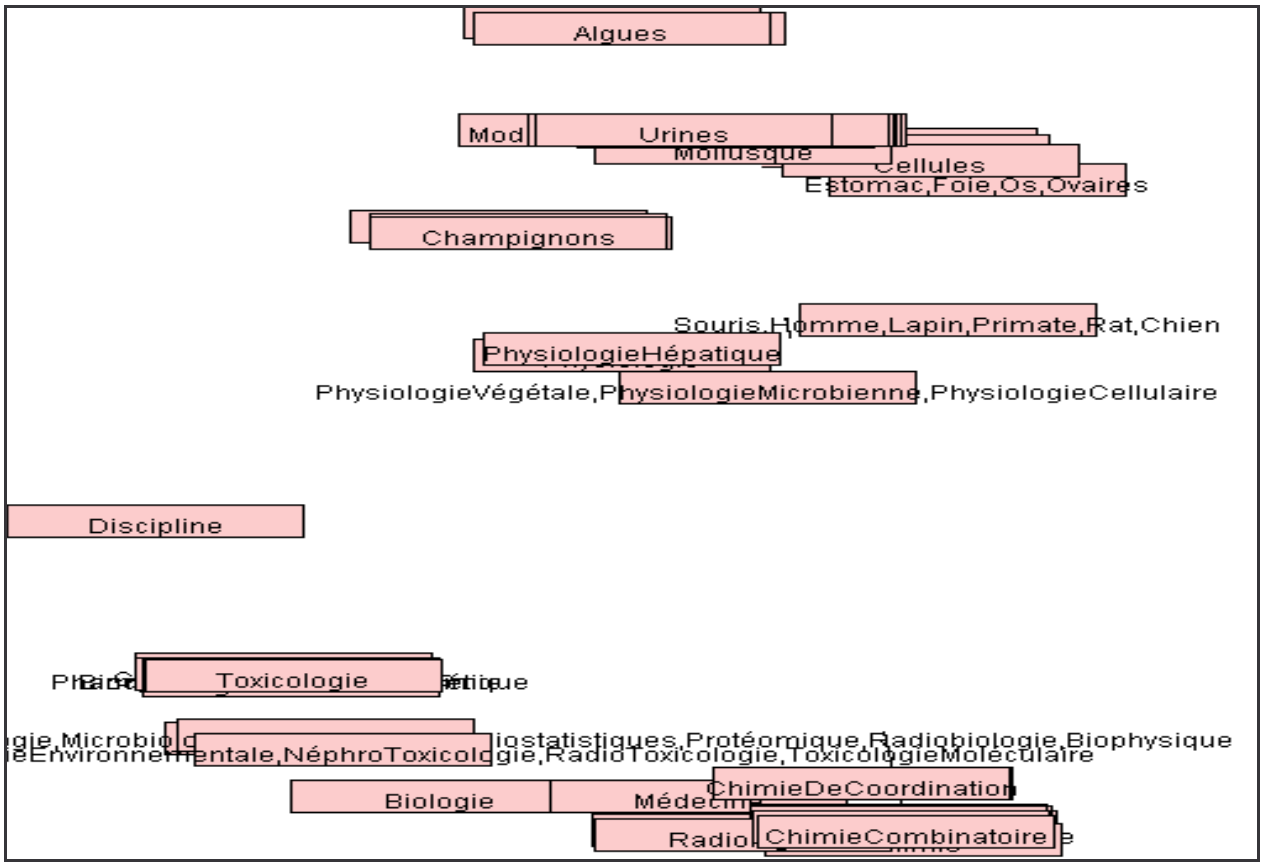


Figure 4.10 : Visualisation utilisant le clustering hiérarchique.

Nous remarquons que la majeure partie des concepts et leurs descendants forment un nouveau concept, ce qui est normal en appliquant l'algorithme de clustering hiérarchique. Nous pouvons toujours distinguer les trois sous ontologies.

- **Visualisation utilisant le clustering k-means**

Nous allons dans un premier temps choisir k égal à trois et positionner les centres des clusters à *Discipline*, *Organismes* et *ModeleBiologiques* (ascendants principaux des trois sous ontologies), puis comparer au résultat obtenu à la Fig.4.9.

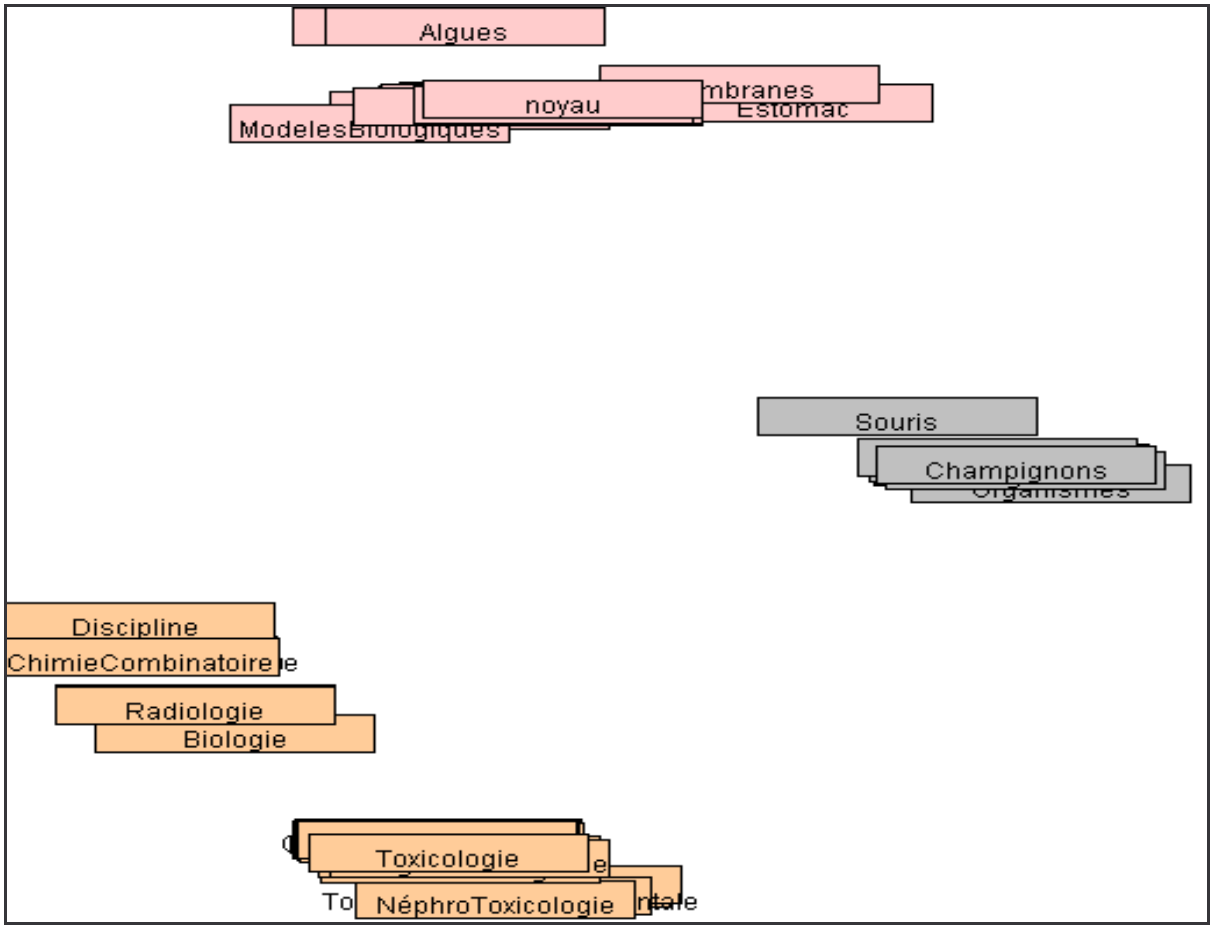


Figure 4.11 : Visualisation utilisant k-means (k = 3).

Nous remarquons que chaque sous ontologie est représentée par un cluster. Nous allons augmenter le nombre de cluster à six pour descendre à un niveau beaucoup plus bas de l'ontologie afin de voir si la sémantique des concepts est respectée à ce niveau. Pour cela nous allons conserver les trois centres de l'étape précédente et ajouter comme nouveaux centres : *Biologie* (descendant de *Discipline*), *Mammifere* (descendant de *Organismes*) et *Algues* (descendants de *ModelesBiologiques*).

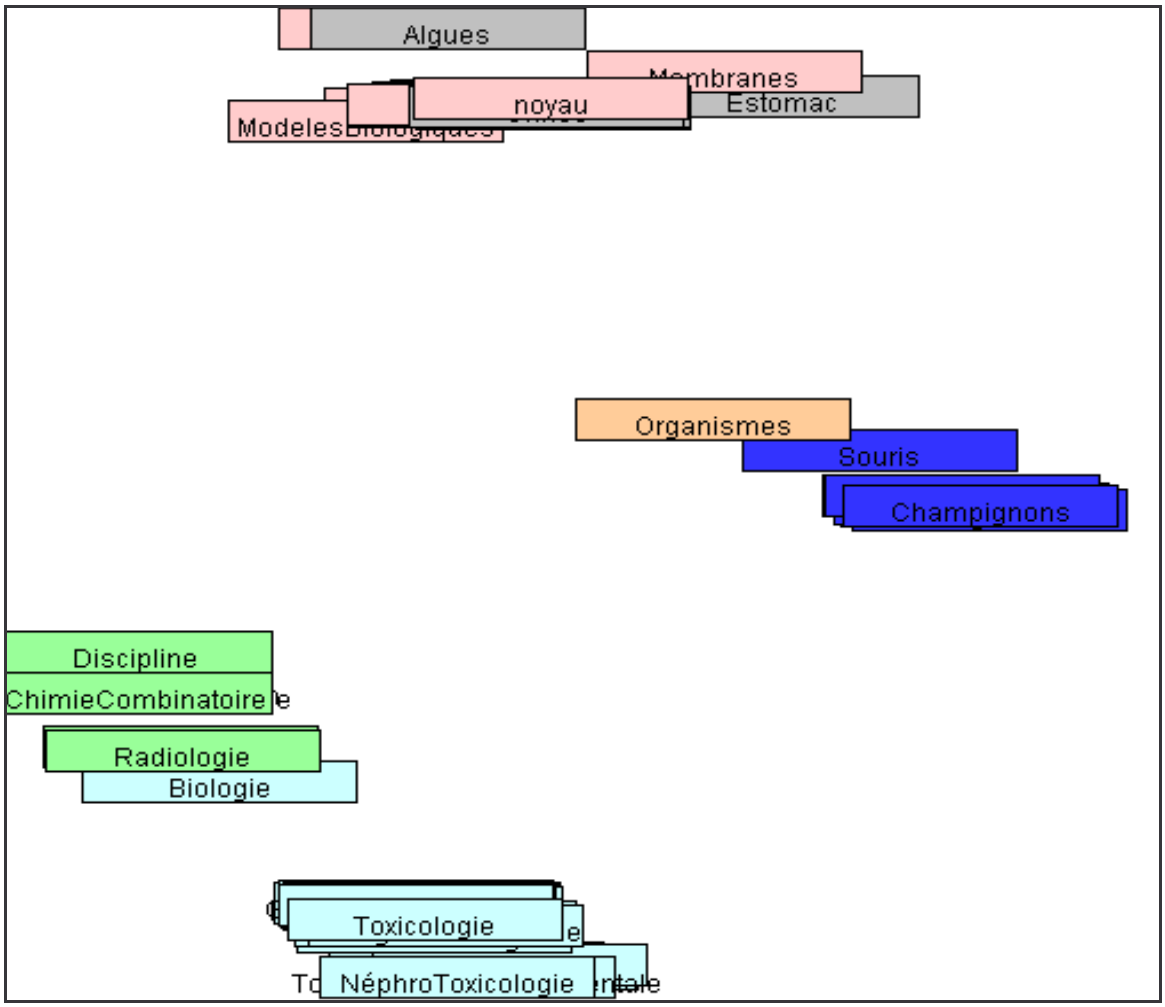


Figure 4.12 : Visualisation utilisant k-means (k = 6).

Avec les ontologies de grande taille, nous avons quelques surprises sur la visualisation, *Algues* descendant de *ModelesBiologiques* et *Organisme*, est dans le même cluster que *Estomac* qui n'est pas son frère.

D'une manière générale, les paysages obtenus en appliquant la distance proposée par [Ranwez et al, 2006] reflètent la sémantique de l'ontologie. Parce que chaque regroupent du paysage est constitué de concepts qui ont des liens sémantiques forts. Mais pour des ontologies de grande taille comportant des héritages multiples (ontologie du projet TOXNUC-E par exemple), nous avons quelques soucis. Car le concept héritant des autres concepts ne se trouve non seulement pas entre ses généralisants mais aussi loin de certains de ces derniers. C'est ainsi que nous jugeons nécessaire de revoir l'intuition que nous avons sur les ontologies supportant l'héritage multiple pour la définition de la distance sémantique détaillée à la section 2 du chapitre 2.

Chapitre 5 : Conclusion et Perspectives

Ce dernier chapitre clot le rapport et dégage les perspectives qui en découles.

5.1. Conclusion

Les travaux de recherche présentés dans ce mémoire concernent, la distance sémantique, la visualisation d'ontologie sur une carte conceptuelle et l'extension de la distance sémantique proposée sur les autres relations ontologiques (partie-tout, etc.).

En introduction de ce mémoire certains objectifs ont été présentés. Il convient ici de préciser ceux qui ont été atteints.

Le premier objectif consistait à la mise en place d'un module permettant de calculer la matrice de distances d'une ontologie au formant OWL et de projeter cette dernière sur notre outil de visualisation afin de voir la sémantique des concepts.

Le deuxième objectif consistait à utiliser quelques techniques de regroupement traditionnelles (clustering hiérarchique, k-means) dans le contexte des ontologies afin d'avoir une idée sur la sémantique des regroupements des concepts de l'ontologie (Fig. : 4.5 à 4.12).

Les comparaisons des paysages obtenus nous ont permis de conclure que le regroupement des concepts correspond à la sémantique de l'ontologie. Mais pour des ontologies de taille importante et comportant des héritages multiples, nous avons obtenus des résultats inattendus. Le concept héritant de deux autres concepts est loin d'un de ses ascendants.

Ces objectifs atteints, apportent deux contributions à la communauté d'ontologie. Le premier concerne la visualisation automatique des ontologies sous Molage. En utilisant cette distance, nous avons une carte conceptuelle où les concepts sont groupés selon leur sémantique. Le second concerne la validation des ontologies. Les développeurs d'ontologies peuvent valider facilement leur ontologie après l'avoir visualisée sous Molage.

5.2. Perspectives

Nos perspectives concernent notre outil de visualisation Molage et la distance sémantique proposée. Comme nous le remarquons sur les paysages, les concepts très proches se superposent. Afin de rendre la navigation sur le paysage beaucoup plus conviviale nous proposons d'utiliser une visualisation en 3D. Dans ce type de visualisation, la dimension supplémentaire permet de mieux repartir les concepts sur la carte conceptuelle.

En ce qui concerne la distance sémantique, il serait très intéressant de l'étendre sur les autres relations ontologiques comme la relation partie-tout. Dans la littérature, nous avons quelques propositions d'extension de distance sémantique sur les autres relations ontologiques comme :

- [Lord et al, 2003] qui étend les distances basées sur le plus petit subsumeur commun sur les relations de GO.
- [Bernstein et al, 2005] propose la représentation vectorielle des concepts.

Il serait intéressant de combiner ces distances sémantiques prenant en compte les autres relations ontologiques avec la distance de [Ranwez et al, 2006].

Du point de vue de la visualisation, une fois la mesure de distance établie, il serait intéressant de l'utiliser pour effectuer des paysages sélectifs :

- avoir une représentation qui respecte la sémantique et donne donc sens à un paysage projeté.
- Utiliser la distance pour effectuer des zooms ciblés.

Bibliographie

[Beck, 2006] Nicolas BECK : Application de Méthodes de Clustering Traditionnelles et Extension au cadre Multicritère. Mémoire de fin d'études, Université Libre de Bruxelles.

[Bernstein et al, 2005] A. Bernstein, E. Kaufmann, C. Buerki, M. Klein. How Similar Is It? Towards Personalized Similarity Measures in Ontologies, In Proceedings of the 7 Internationale Tagung Wirtschaftsinformatik, pp 1347-1366, 2005.

[Budanitsky et Hirst, 2000] Alexander Budanitsky, Graeme Hirst. Semantic Distance in WordNet: An Exprimental, Application-oriented Evaluation of Five Measures. Wordshop on WordNet and Other Lexical Resources, in the North American Chapter of the Association for Computational Linguistics, NAACL-2000, Pittsburgh, PA, June.

[Cordi et al, 2005] Valentina Cordi, Paolo Lombardi, Maurizio Martelli, Viviana Mascardi. An Ontologie-Based Similarity between Sts of Concepts. In Proceedings of WOA 2005, F. De Paoli, E. Merelli and A. Omicini eds. Pitagora Editrice Bologna, ISBN 88-371-1590-3 (2005) 16-21.

[Crampes et al, 2006]Michel Crampes, Sylvie Ranwez, Filip Velickovski, Christopher Mooney and Nicolas Mille : An Integrated Visual Approach for Music Indexing and Dynamic Playlist Composition. MMCN2006, Thirteenth Annual Multimedia Computing and Networking, January 18-19, 2006, San Jose, California.

[Deerwester et al, 1990] S.Deerwester et al. Indexing by latent Semantic analysis.
Journal of the American society for Information Science, vol 41, pp. 391-407, 1990.

[Gandon, 2002] Fabien Gandon. Distributed Artificial Intelligence and Knowledge Management : Ontologies and Multi-Agent Systems For a Corporate Semantic- Web.

[Gruber, 1992] Thomas R. Gruber. *Ontolingua: A mechanism to support portable ontologies*. Knowledge Systems Laboratory Technical Report KSL-91-66, Stanford University, version 3.0, CA, 1992.

[Hernandez, 2006] Nathalie Hernandez : Ontologies de domaines pour la modélisation du contexte en recherche d'information. Thèse en informatique Université Paul Sabatier de Toulouse.

[Lafourcade, 2002] Mathieu Lafourcade, Violaine Prince, Didier Schwab. Vecteurs conceptuels et structuration émergente de terminologie. Revue TAL.

[Lord et al, 2003] P.W. Lord, R.D. Stevens, A. Brass, C.A. Goble. Semantic similarity measures as tools for exploring the Gene Ontology, In Proceedings of the Pacific Symposium on Biocomputing, pp 601-612, 2003.

[Mizoguchi, 2004] Riichiro Mizoguchi. Le rôle de l'ingénierie ontologique dans le domaine des EIAH.

[Natalya et al, 2001] Natalya F. Noy et Deborah L. McGuinness. Développement d'une ontologie 101 : Guide pour la création de votre première ontologie.

[Pedersen et al, 2006] Ted Pedersen, Serguei V.S. Pakhomov, Siddharth Patwardhan, Christopher G. Chute. Measures of semantic similarity and relatedness in the biomedical domain.

[Ranwez et al, 2006] Sylvie Ranwez, Vincent Ranwez, Jean Villerd, Michel Crampes. Ontological Distance Mesures for Information Visualisation on Conceptual Maps.

[Ranwez, 2000] Sylvie Ranwez. Composition de documents Hypermédia Adaptatifs à partir d'Ontologies et de requêtes Intentionnelle de l'Utilisateur, PhD thesis in computer science, Montpellier II University.

[Resnik, 1999] Philip Resnik. An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence Research, Vol. 11 (1999) 95-130.

[Roddick et al, 2003] John F. Roddick, Kathleen Hornsby, Denise de Vries. A Unifying Semantic Distance Model for Determining the Similarity of Attribute Values. In Proceedings of th 26 th Autralian Computer Science Conference (ACSC2003), Adelaide, Australia.

[Shu et al, 2006] Gao Shu, Omer F. Rana, Nick J. Avis, Chen Dingfang. Ontology-based semantic matchmaking approach. Advances in Engineering Software, Volume 38, Issue 1, January 2007, Pages59-67.

[Zargayouna et Salotti, 2004] Haïfa Zargayouna et Sylvie Salotti. Mesure de similarité sémantique pour l'indexation de documents semi-structurés.Technique Report, mars 2001. 12ieme Atelier de raisonnement à partir de cas. Mars 2004.

[W3C 2004] Michael K. Smith, Chris Welty, Deborah L. McGuinness. OWL Web Ontology Language - Reference. http://www.w3.org/TR/2004/REC-owl-ref-20040210/ (en ligne au 16 juin 2005).