MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Ingénierie des Systèmes d'Information**

THEME :

Indexation sémantique d'une base textuelle

Etudiante : TOUIHR Faiza Safia

Encadrante : BELKHEIR. L

Année Universitaire 2015/2016

Résumé

Indexation sémantique d'une base textuelle

La recherche d'information (RI) est apparue comme une discipline de recherche afin d'apporter une solution aux problèmes liés à l'accès aux informations contenues dans des grandes masses de documents. La RI avait proposé des techniques pour bien organiser et faciliter l'accès aux informations contenues dans ces documents dont le nombre ne cesse de croître. D'où le monde a assisté à une croissance importante en termes de nombre de ressources d'informations difficilement accessibles et de nombre d'utilisateurs qui souhaitent accéder aux informations contenues dans ces ressources.

Malgré le très grand nombre de recherches faites, les systèmes de recherche d'information présentent encore des lacunes au niveau sémantique et sont perfectibles à plus d'un sens. Pour cette raison, des recherches sont toujours en cours. Certaines de ces recherches tentent d'introduire des techniques pour améliorer les performances d'un SRI.

Nous nous intéressons dans le cadre de ce travail à une nouvelle orientation en RI : l'indexation sémantique d'une base textuelle, qui s'appuie sur les sens des mots, dans la représentation des documents et requêtes. Ces sens sont identifiés par des techniques de désambiguïsation des sens des mots.

La réalisation de notre travail se base sur un corpus Anglais, elle fait appel à l'ontologie linguistique Anglaise « *WordNet* » pour la désambiguïsation des sens des mots de corpus et requête à travers « *les synsets* », aussi pour proposer un classement sémantique des résultats retournés par le moteur de recherche.

Mots clé

Recherche d'Information (RI), Modèles de RI, Indexation Sémantique, Désambiguïsation des Sens des Mots, Corpus Anglais, l'Ontologie Linguistique WordNet.

Abstract

Semantic indexing of textual base

Information retrieval (IR) has emerged as a research discipline in order to provide a solution to problems related to access to information in the broad masses of documents. IR had proposed techniques to properly organize and facilitate access to information contained in these documents, the number continues to grow. Hence the world has witnessed significant growth in terms of number of information resources with difficult access and users who wish to access the information contained in these resources.

Despite the large number of research done, information retrieval systems still have shortcomings at the semantic level and can be improved to more than one meaning. For this reason, research is still ongoing. Some of these researches are trying to introduce techniques to improve the performance of SRI.

We are interested in the context of this memory on a new direction in IR: semantic indexing of textual base, based on the meaning of words, in the representation of documents and queries. These senses are identified by disambiguation techniques meaning of words.

The realization of our work is based on an English corpus, it uses the English language ontology "WordNet" for disambiguation of the meaning of words of corpus and query through the "synsets", also for proposing a semantic classification of results returned by the search engine.

Keywords

Information Retrieval (IR), IR Models, Semantic Indexing, Word Sense Disambiguation, English Corpus, the WordNet ontology language..

Sommaire

CHAPITRE IV : CONCEPTION ET IMPLÉMENTATION

Introduction générale

Contexte et Problématique

La Recherche d'Information (RI) s'intéresse principalement à sélectionner à partir d'un ensemble de documents existants, ceux qui sont pertinents à une requête utilisateur. Afin d'y parvenir, l'une des tâches principales d'un Système de Recherche d'Information (SRI) est l'indexation. *L'indexation* consiste à construire des représentations simplifiées décrivant le contenu informationnel des documents et requêtes en vue de faciliter la recherche. Ces représentations sont ensuite interprétées par un *modèle de recherche* dans un formalisme unifié, puis comparées dans le but d'évaluer les degrés de pertinence des documents pour les requêtes.

Dans les SRI classiques, les documents et les requêtes sont représentés (ou indexés) par des mots-clés, manuellement ou automatiquement extraits à partir de leurs textes. Dans de tels systèmes, l'appariement (ou mise en correspondance) document-requête est lexical basé sur la présence ou l'absence des mots de la requête dans le document. Un document est alors considéré d'autant plus pertinent pour la requête qu'il a de mots clés en commun avec cette requête. Or, les mots de la langue sont par nature ambigus. Un même mot utilisé dans le document et la requête peut définir des sens différents (cas de polysémie et d'homonymie), et plusieurs mots lexicalement différents utilisés dans le document et la requête peuvent refléter un même sens (cas de synonymie). De ce fait, des documents pourtant non pertinents, contenant des mots de la requête, sont retrouvés, tandis que des documents sémantiquement pertinents, ne contenant aucun mot de la requête, ne sont pas retrouvés. Pour pallier les problèmes de l'indexation basée mots-clés, *l'indexation sémantique* est apparue, elle s'appuie sur la représentation des documents et requêtes par des sens des mots (ou concepts). Ces sens, sont extraits, à partir du contenu des documents et requêtes, par des méthodes de *désambiguïsation des sens des mots*(WSD) permettant de retrouver le sens adéquat d'un mot ambigu dans son contexte d'utilisation dans le document ou la requête.

L'indexation sémantique à l'issue de la recherche d'information, permet de retrouver des documents sémantiquement pertinents à une requête utilisateur, bien que ne partageant pas de mots en commun avec cette dernière. La qualité d'une recherche d'information sémantique dépend de la précision des techniques de WSD utilisées pour sélectionner les concepts représentatifs des documents et requêtes.

Dans ce cadre, nous proposons une nouvelle approche d'indexation sémantique basée sur les *synsets* de l'ontologie linguistique *WordNet* qui permet de capturer le sens voulu par le besoin d'information qui ne s'exprime pas par les termes de la requête, et cela permet de couvrir tous les documents de la collection et donc récupérer tous les documents pertinents existants. Cette approche permet d'augmenter le rappel et la précision du SRI.

Objectif et Organisation du Mémoire

L'objectif de notre projet présenté dans ce mémoire est la conception et la réalisation d'un moteur de recherche sémantique d'un corpus anglais à base de l'ontologie linguistique WordNet.

Ce mémoire s'articule en quatre chapitres principaux :

- Le premier chapitre représente l'état de l'art de ce mémoire, dont la première section intitulée « La recherche d'information » : nous présentons les notions de base de notre domaine d'application et les modèles sur lesquels repose la « Recherche d'Information ». Et la deuxième section intitulée « Indexation sémantique » : nous présentons dans cette section le besoin de l'indexation sémantique et les différents travaux et méthodes de désambiguïsation des sens des mots dans cette approche.
- Le deuxième chapitre traite la langue anglaise puisque en va travailler sur un corpus anglais (section I), dans la section suivante nous présentons les principes de l'ontologie linguistique WordNet qui couvre la grande majorité des noms, verbes, adjectifs et adverbes de la langue anglaise, dans un premier lieu nous commençons a donné une définition de WordNet, et nous exposons par la suite les relations sémantiques et quelques données statistiques par la suite nous présentons les limites du WordNet.
- Le chapitre suivant est consacrée à la modélisation de notre projet par les différents diagrammes du langage UML avec l'outil StarUML.
- Le dernier chapitre représente l'essentiel de notre travaille, commençant par l'introduction et l'environnement de développement, puis les étapes de conception de notre application, ensuite l'illustration des interfaces de l'application le tout finalisé par une conclusion.

Introduction :

L'objectif du présent chapitre est de présenter un état de l'art sur deux domaines : la recherche d'information (Section 1) et l'indexation sémantique (Section 2).

Dans la première section, nous présentons les concepts de base de la RI. En particulier, nous décrivons les notions de document, de besoin d'information, de requête et de corpus; les

processus de recherche d'information et d'indexation ; ainsi que, les modèles de RI. La dernière partie de cette section est discutée l'évaluation des systèmes de recherche d'information.

La deuxième section du chapitre est consacrée à la prise en compte de la sémantique dans les SRI, à cet effet, nous présentons un état de l'art sur l'indexation sémantique. En premier temps, nous présentons la problématique de l'indexation classique basée mots-clés. Le reste de la section est dédiée à la présentation des approches d'indexation sémantique basées sur la désambiguïsation des sens des mots.

I. SECTION 1 : La Recherche d'Information : État de l'Art

I.1. Définition

La recherche d'information (RI) n'est pas un domaine récent, il date des années 40. Une des premières définitions de la RI a été donnée par SALTON : « la recherche d'information est un domaine qui a pour objectif, la représentation, l'analyse, l'organisation, le stockage et l'accès à l'information » [1].

Plusieurs tâches se regroupent sous le vocable de la RI, la plus ancienne est la recherche documentaire, l'extraction d'information, la recherche d'information multilingue, les questions réponses, la recherche d'information sur le web, etc.

I.2. Bref Historique de la RI

- **1940:** Apparition des SRI, focalisation des RI sur les applications des bibliothèques.
- **1950:** Apparition du modèle booléen et l'élaboration de petites expérimentations sur des petites collections de documents.
- **1960 et 1970:** Apparition du système SMART, Développement d'une méthodologie d'évaluation de système et conception de corpus de test (CACM).
- **1980:** Développement de l'intelligence artificielle (IA), ainsi l'intégration des techniques de l'IA en RI (système expert).
- **1990 et 1995:** L'apparition d'internet, la RI a été modifié et sa problématique plus élargie [2].

I.3. Concepts de base de la RI

Une synthèse des travaux de [3] et [4] nous a permis de dégager les concepts suivants :

I.3.1. Collection de documents

La collection de documents (Corpus de documents) constitue l'ensemble des informations exploitables et accessibles. Elle constitue des représentations simplifiées mais suffisantes.

I.3.2. Document

Le document constitue l'information élémentaire d'une collection de documents. L'information élémentaire (granule de document), peut représenter tout ou une partie d'un document. Un document peut être un texte, une page web, une image, une bande vidéo, etc.

I.3.3. Besoin d'information

La notion de besoin d'information en RI est assimilée au besoin de l'utilisateur. Trois types de besoin utilisateur ont été définis par [5]:

- **Besoin vérificatif :** L'utilisateur cherche à vérifier le texte avec les données connues qu'il possède déjà. Il recherche donc une donnée particulière, et sait même comment y accéder. Ce besoin est dit stable : il ne change pas au cours de la recherche.
- **Besoin thématique connu :** L'utilisateur cherche à clarifier, à revoir ou à trouver de nouvelles informations dans un sujet connus. Un besoin de ce type peut être stable ou variable : il est possible que le besoin s'affine au cours de la recherche.
- **Besoin thématique inconnu :** Cette fois, l'utilisateur cherche de nouveaux concepts ou de nouvelles relations en dehors des sujets ou des domaines qui lui sont familiers. Le besoin est variable et est toujours exprimé de façon incomplète.

I.3.4 Requête

Une requête constitue l'*expression* du *besoin* en informations de l'utilisateur. Plusieurs systèmes utilisent des langages différents pour décrire la requête :

- En langage naturel : cas des systèmes SMART et SPIRIT.
- En langage booléen : cas du système DIALOG.
- En langage graphique : cas du système NEURODOC.

I.3.5 Corpus

Nous employons le mot *corpus* dans une acception assez restreinte empruntée à [6] : « Un corpus est une collection de données langagières qui sont sélectionnées et organisées selon des critères linguistiques explicites pour servir d'échantillon du langage. » Nous précisons cette optique au chapitre VI. À cette aune, nombre de ressources textuelles perdent cette dénomination. Il s'agit souvent de collections ou de rassemblements de textes électroniques plutôt que de corpus à proprement parler.

I.4. Processus d'un système de recherche d'information

Un système de recherche d'information manipule un corpus de documents qu'il transpose à l'aide d'une fonction d'indexation en un corpus indexé. Ce corpus lui permet de résoudre des requêtes traduites à partir de besoins utilisateur. Un tel système repose sur la définition d'un modèle de recherche d'information qui fait correspondre les documents aux requêtes.
La figure I.1, présente les étapes de processus d'un système recherche d'information.

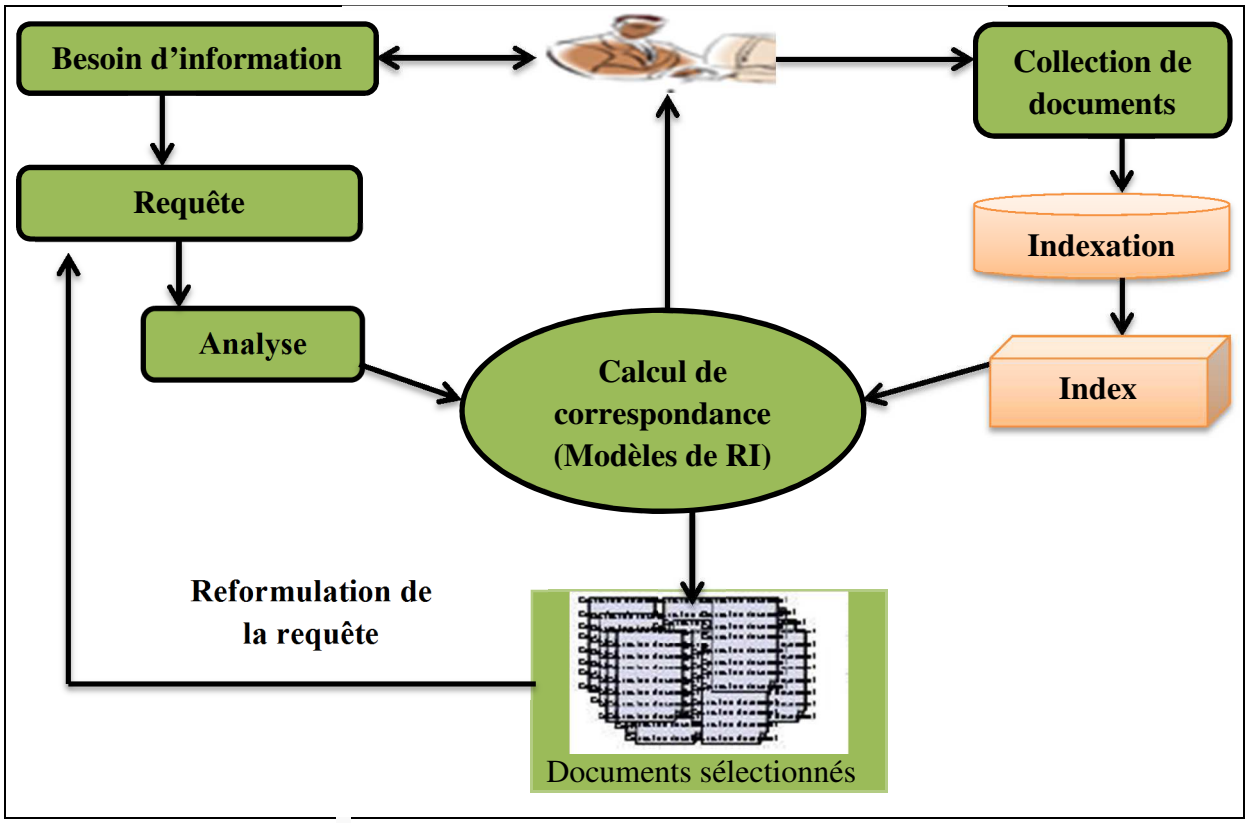

Figure I.1. Processus d'un système de recherche d'information.

I.4.1. L'indexation

Elle consiste à extraire des documents les mots les plus discriminants appelés *index* et les mettre dans un fichier appelé *fichier inverse*. Cette tâche est effectuée en marge du processus de recherche. **Les index** sont utilisés pour représenter le contenu des documents, ils ont un caractère réducteur car tous les termes d'un document ne sont pas importants à prendre en compte pour la recherche. Ils ont les caractéristiques suivantes :

- ils ne représentent qu'une partie du contenu des documents.
- ils peuvent prendre plusieurs formes (ex : mots simples, termes, syntagmes, entrées dans un thésaurus, etc.)[4].

Les fichiers inverses permettent d'associer des index aux documents qui les contiennent.

I.4.1.1. Les approches d'indexation [7]

A. Indexation manuelle (contrôlée)

C'est un spécialiste du domaine qui effectue l'analyse du document, pour identifier son contenu et construire une représentation de ce contenu.

B. Indexation semi-manuelle (contrôlée)

L'indexation semi manuelle se divise en 2 parties, une partie automatique permettant d'extraire une liste de descripteur, et une deuxième partie qui est manuelle réalisée par un spécialiste du domaine dont la tâche est de sélectionner des termes significatifs parmi les descripteurs retournés auparavant.

C. Indexation automatique (libre)

C'est le SRI qui génère les indexes des documents. L'indexation automatique a été créée afin de remédier aux problèmes liés aux approches précédentes, elle présente l'avantage d'une régularité du processus, car l'indexation automatique fournit toujours le même index pour le même document, ce qui constitue une qualité du système.

En ce qui nous concerne, c'est la troisième approche qui nous intéresse et nous pouvons la résumer comme suit.

I.4.1.2. Processus d'indexation automatique

Le processus d'indexation automatique (Figure I.2) passe par trois phases, chaque phase pouvant contenir une ou plusieurs étapes selon les usages des utilisateurs, c'est au programmeur de sélectionner les étapes qu'il souhaite intégrer au processus d'indexation.

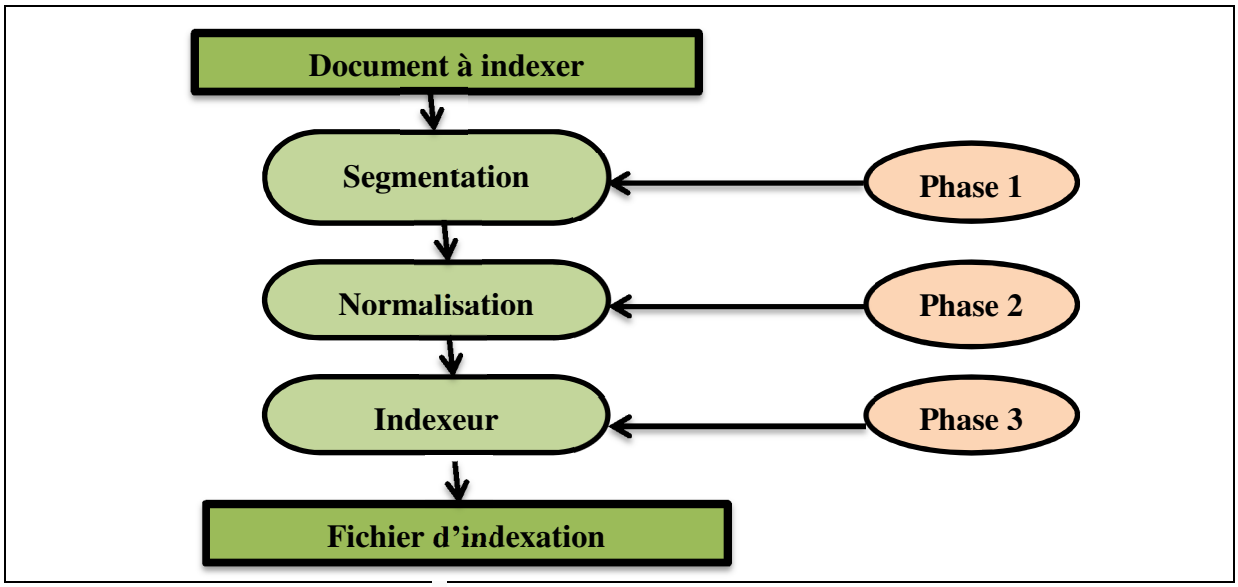

Figure I.2. Processus d'indexation automatique [8].

A. Phase de segmentation

Cette phase représente la fragmentation des documents en unités, elle est basée sur la ponctuation et sur une liste de séparateur, le résultat de cette phase est un ensemble de mots.

B. Phase de normalisation

Cette phase peut contenir plusieurs étapes, dans ce qui suit nous allons expliquer les étapes les plus importantes, elle traites plusieurs niveaux :

- **Niveaux lexical et morphologique :** Chaque mot de la langue lui correspond une catégorie morpho syntaxique :

•La lemmatisation : Le lemme s'obtient par une flexion. Les mots d'une langue peuvent être classés en 2 catégories: *Les lemmes: formes canoniques* (infinitif pour les verbes, singulier pour les noms, etc.) qui constituent les entrées dans un dictionnaire de cette langue ; *Les mots*

obtenus par flexion de ces lemmes: conjugaison d'un verbe, changement de genre ou de nombre, etc. Par exemple, le mot « devrait» est obtenu par flexion (conditionnel présent, 3em personne du singulier) du verbe « devoir» [9] [8].

•La racinisation: Consiste à rechercher la forme tronquée d'un mot à partir de laquelle peuvent être reconstruites ses différentes variantes morphologiques [9]. Elle peut être réalisée simplement, en utilisant un algorithme comme l'algorithme de Porter [10] [8].

- **Niveaux syntaxique** : Basée sur l'utilisation de règles dépendantes de la langue :

•L'élimination des mots vides : Les mots vides sont des mots qui permettent de lier entre eux les mots d'une phrase pour la structurer (les articles, les conjonctions de coordination, les verbes auxiliaires, etc.). Ces mots ne portent pas de sens, ils ne peuvent pas constituer des index il faut donc les éliminer [9] [8].

•La discrimination : Par "discrimination", on réfère au fait qu'un terme distingue bien un document des autres documents, un terme qui a une valeur de discrimination élevée doit apparaître seulement pour un petit nombre de documents. L'idée est de garder les termes discriminants, et éliminer ceux qui ne le sont pas [9] [8].

•L'extraction des entités nommées : Les entités nommées sont des mots ou des groupes de mots qui désignent des personnes, des organisations, des dates, des lieux [8].

- **Niveaux sémantique :** Ici on s'intéresse à déduire les sens des mots, leurs concepts représentatifs et les relations sémantiques entre les mots. A cette étape, une ontologie peut être utilisée. La section 2 sera dédiée à cette approche (l'indexation sémantique) car c'est celle qui nous intéresse dans le cadre de notre travail.
- **Niveau pragmatique :** Il s'agit de l'analyse du langage naturel par la connaissance du monde réel. Ce niveau n'a pas été automatisé pour le moment.

C. Phase d'indexeur

Dans cette phase on utilise une approche permettant de sélectionner les index et de leur associer *une pondération*, cette dernière permet d'assigner aux termes leur degrés d'importance dans les documents, il existe 3 approches pour le choix des index :

- **Approche basée sur la fréquence d'occurrences** : Cette approche consiste à choisir les mots représentants selon leur fréquence d'occurrence. La façon la plus simple consiste à définir un seuil sur la fréquence: si la fréquence d'occurrence d'un mot dépasse ce seuil, alors il est considéré important pour le document [7].
- **Approche basée sur la valeur de discrimination :** Le calcul de la valeur de discrimination a été développé dans la loi de Luhn [11].
- **Approche basée sur tf.idf :** Les schémas de pondération pour l'attribution d'un poids à un mot, prennent en considération trois facteurs : le facteur de *pondération local* (*tf* - term frequency-), qui mesure l'importance du terme dans le document ; un facteur de *pondération globale* (*idf* - inverted document frequency-), mesurant la représentativité globale du terme dans la collection et un facteur de *normalisation* qui prend en compte la longueur du document. Une formule *tf.idf* combine l'importance du terme pour un document (*tf*), et le pouvoir de discrimination de ce terme (*idf*). Un terme qui a une valeur de *tf.idf* élevée doit être important dans ce document [12] [13].

I.4.2. Appariement Document/ Requête (Fonction de correspondance)

Tout système de recherche d'information s'appuie sur un modèle de recherche d'information. Ce modèle se base sur une fonction de correspondance qui met en relation les termes d'un document avec ceux d'une requête en établissant une relation d'égalité entre ces termes. La correspondance s'effectue au niveau de l'appariement document/ requête. L'expression de la fonction d'appariement dépend du modèle de RI choisi [14] [4].

I.4.3. Les Modèles de la RI **[15]**

Un modèle de RI a pour rôle de fournir une formalisation du processus de RI. Il doit accomplir plusieurs rôles dont le plus important est de fournir un cadre théorique pour la modélisation de la mesure de pertinence. Il existe trois grandes classes : les modèles booléens les modèles vectoriels et les modèles probabilistes, nous les définissons comme suit :

I.4.3.1. Modèles Booléens (Ensemblistes)

Ce sont les plus anciens de tous les modèles de RI. Ils sont basés sur la théorie des ensembles. Le document « d » et la requête « q » sont représentés comme une conjonction logique reliés par des opérateurs Booléens {ET (∧), OU (∨) et NON (¬)} de termes (ti), par exemple : d = {(t1 ∧ t2 ∧ … ∧ tn)} et q = {(t1 ∧ t2) ∨ (t3 ∧ ¬t4)}.

I.4.3.2. Modèles Vectoriels (Algébriques)

Dans ces modèles les documents et les requêtes sont représentés par des vecteurs de poids des termes. Chaque poids interprète l'importance du terme correspondant dans le texte.

Les vecteurs sont décrits dans un espace vectoriel définis par l'ensemble des termes préalablement établi lors de l'indexation.

I.4.3.3. Modèles Probabilistes

Ils se basent sur un modèle mathématique fondé sur la théorie de la probabilité. Le processus de recherche s'interprète par calcul du proche en proche de probabilité de pertinence d'un document relativement à une requête.

Nous orientons l'attention du lecteur vers [15] pour de plus détails à propos de ces modèles.

I.4.4. La reformulation de la requête [7]

La possibilité de reformuler la requête initiale s'avère intéressante dans le processus de la RI. Cela fera en sorte que le résultat retourné soit plus pertinent. Il existe 3 méthodes.

- **La reformulation manuelle :** Elle consiste à présenter à l'utilisateur une liste de documents jugés pertinents en réponse à la requête initiale. C'est à l'utilisateur de sélectionner à partir des documents pertinents ceux dont lesquels le système va extraire les termes à rajouter à la requête initiale pour une nouvelle recherche.
- **La reformulation semi-automatique :** Elle nécessite l'intervention de l'utilisateur qui doit identifier les documents pertinents et les documents non pertinents.

- **La reformulation automatique :** L'extension de la requête est faite sans intervention de l'utilisateur grâce à l'utilisation d'un thésaurus contenant des informations linguistique (équivalence, association, hiérarchie).

I.5. Évaluation d'un système de recherche d'information

Depuis l'apparition des premiers modèles pour la RI, l'évaluation objective de leur efficacité représentait une pièce-maitresse dans le développement du domaine. Il était évident pour les chercheurs la nécessité de trouver des mesures standards pour estimer la qualité des résultats de recherche, la communauté de la RI a approuvé deux mesures de base qui sont la précision et le rappel [16] [15].

Pour l'évaluation de la précision et le rappel, la cardinalité des différents ensembles de documents est utilisée, Comme décrit dans le Tableau I.1.

	Documents pertinents	**Documents non-pertinents**
Documents sélectionnés	Documents trouvés	Documents trouvés Documents hors contexte : **bruit**
Document non-sélectionnés	Documents oubliés : **silence**	Documents non trouvés non pertinents

Tableau I.1. Les quatre ensembles de documents résultats en RI [17].

- **La précision :** Évalue la portion de *documents sélectionnés* par le SRI, et qui sont *pertinents* par rapport au besoin de l'utilisateur [17].
- **Le rappel :** Évalue la portion de *documents pertinents* qui sont sélectionnés, par rapport au besoin de l'utilisateur [17].

II. SECTION 2 : L'Indexation Sémantique : État de l'Art

II.1. La notion sémantique et les démarches d'indexation sémantique

La sémantique est un mot d'origine grecque qui signifie l'étude du sens. C'est une notion plutôt philosophique, qui a été employée à propos des systèmes d'information pour mieux rapprocher l'interprétation des choses par des machines avec celle des humains. L'idée de la sémantique est de construire des modèles qui permettent de comprendre, structurer et prédire certaines parties du monde [18]. Au fil du temps, cette idée a été mieux formalisée, et a commencé à s'intégrer dans différents domaines. En conséquence il existe deux démarches de l'indexation sémantique, la RI et le web sémantique[1].

[1] http://w3cwebsemantique.orgfree.com/upload/5.pdf [Date de dernière visite: Mai 2016]

II.2. la démarche d'indexation sémantique issue de la RI

II.2.1. De la RI classique à la RI sémantique

II.2.1.1. Problématique (Besoin de l'indexation sémantique)

Les modèles classiques de la RI, se basent sur l'hypothèse qu'il y a une correspondance stricte entre les mots et les sens, alors qu'un mot peut représenter plusieurs sens et un sens peut être représenté par plusieurs mots. En partant de cette hypothèse, la recherche d'information classique se trouve face à deux problèmes, *l'ambiguïté des mots* et leur *disparité*.

- *L'ambiguïté des mots*, dite ambiguïté lexicale, se rapporte à des mots lexicalement identiques et portant des sens différents. On parle ici du *bruit*.
- *La disparité des mots* se réfère à des mots lexicalement différents mais portant un même sens. Ceci implique que des documents, pourtant pertinents, ne partagent pas de mots avec la requête, ne sont pas retrouvés. On parle ici du *silence* [19] [20].

II.2.1.2. Solution

La solution globale permettant de répondre à ces deux problèmes consiste en l'indexation sémantique. L'indexation sémantique tente d'apporter des solutions au niveau de la représentation des documents et des requêtes, l'indexation sémantique ou conceptuelle est sensée améliorer les performances du SRI. D'où on distingue deux grandes approches : *l'indexation sémantique* et *l'indexation conceptuelle*.

L'indexation conceptuelle : Peut être vue comme une généralisation de l'indexation sémantique, dans la mesure où les concepts véhiculent des sens [3].

II.2.2. L'indexation sémantique

L'indexation sémantique s'intéresse principalement à la représentation des documents et requêtes par les sens des mots qu'ils contiennent plutôt que par les mots eux-mêmes. L'objectif sous-jacent est d'améliorer la représentation des entités indexées et de pallier aux problèmes de l'indexation classique basée mots [7].

II.2.3. Méthodes d'indexation sémantique en RI:

Nous détaillerons dans ce qui suit les travaux les plus représentatifs de l'utilisation du sens des mots dans la RI à travers les travaux de Voorhees, Krovetz & Croft, Sanderson, Mihalicea & Moldovan et Katz & Uzuner & Yuret.

A. La méthode de Voorhees

Voorhees [21] a construit un outil de désambiguïsation basé sur WordNet. Pour désambiguïser une occurrence d'un mot ambigu, les synsets (sens) de ce mot sont classés en se basant sur la valeur de cooccurrence calculée entre le contexte de ce mot et un voisinage contenant les mots du synset dans la hiérarchie de WordNet. Voorhees a expérimenté cette approche sur une collection de test désambiguïsée (les requêtes de la collection de test sont

aussi désambiguïsées manuellement) par rapport aux performances du même processus sur la même collection dans son état d'origine (ambigu).

Les résultats de ses expérimentations ont montré que pour chacune de ces collections, les performances du système de RI diminuent sensiblement dans le cas de l'utilisation des collections désambiguïsées.

B. La méthode de Krovetz & Croft

Krovetz et Croft [22] ont conduit une vaste étude sur certaines hypothèses ayant trait à la pertinence de la relation de correspondance du sens des mots dans la requête et les documents. En utilisant les collections de test CACM et Time, ils ont examiné les dix (10) premiers documents restitués pour chaque requête (pour les deux collections considérées).

Ils ont analysé la correspondance de sens entre chaque terme de la requête et ses occurrences dans chaque document restitué. Krovetz et Croft ont examiné l'amélioration de l'efficacité de la recherche en supprimant les documents sélectionnés avec des sens erronés. Sur la collection Time, une amélioration de 4% est constatée au niveau de la précision moyenne, mais sur la collection CACM, l'augmentation est de 33%. Ils concluent en suggérant des situations où la désambiguïsation peut s'avérer intéressante pour améliorer les performances des SRI.

C. La méthode de Sanderson

Les analyses de Sanderson [23] reprennent les travaux de Krovetz et Croft et détaillent l'impact des erreurs de désambiguïsation dans l'efficacité des SRI. Sanderson a utilisé une forme d'ambiguïté artificielle qu'il désigne par pseudo-mot (pseudoword).

Une concaténation de plusieurs mots choisis aléatoirement dans un corpus forme un pseudo-mot. Ces mots deviennent les pseudo-sens du pseudo-mot unique qu'ils forment, et toutes leurs occurrences dans ce corpus sont remplacées par ce pseudo-mot. En ajoutant des pseudo-mots dans un document de la collection de test, une quantité mesurable d'ambiguïté additionnelle est introduite et son impact sur l'efficacité de la recherche peut être déterminé.

Les résultats ont montré que l'ambiguïté introduite ne réduit pas les performances du système et que la désambiguïsation est utile dans le cas des requêtes courtes et avec un taux de performance (de l'outil de désambiguïsation) élevé (>90%).

D. La méthode de Katz & Uzuner & Yuret

La méthode de Katz & Uzuner & Yuret [24] est basée sur la notion de contexte pour désambiguïser les mots dans le texte. Le sens d'un mot est identifié à partir de son contexte local. Ils partent de l'hypothèse que les mots utilisés dans un même contexte local, appelés sélecteurs, ont souvent des sens proches. Les sélecteurs sont utilisés pour identifier le bon synset de WordNet (les synonymes d'un seul sens) correspondant à un mot dans son contexte.

L'algorithme de désambiguïsation de Katz est testé sur le corpus Semcor où chaque mot est étiqueté avec sa catégorie syntaxique (POS : nom, verbe, adjectif, adverbe) ainsi que le numéro de sens correspondant dans WordNet. Dans ce corpus, la précision pour le désambiguïseur est de 60% (les termes ayant un seul sens ne sont pas compris).

Katz et ses collègues ont intégré leur algorithme de désambiguïsation au processus de RI. Ils ont utilisé le système SMART. Leur conclusion est que leur algorithme n'améliore pas les performances du système de RI.

E. Mihalcea et Moldovan :

Mihalcea et Moldovan [25], ont observé une amélioration de 16% dans le rappel et de 4% dans la précision quand ils ont utilisé une combinaison de l'indexation basée sur les mots clés et de l'indexation basée sur les synsets de WordNet.

F. katz & Uzuner & Yuret :

Katz & Uzuner & Yuret, tout comme Voorhees et Sanderson ainsi Mihalicea & Moldovan, pensent qu'une désambiguïsation plus performante peut aider à améliorer les performances des systèmes de RI.

Pour retrouver les sens corrects des mots, l'indexation sémantique requiert des techniques de désambiguïsation des sens des mots (*WSD -Word Sense Disambiguation-*)**.**

Nous décrivons, dans ce qui suit, Les approches d'indexation qui sont basées sur les techniques de désambiguïsation des sens des mots, les ressources linguistiques externes (structurées et non structurées) les plus exploitées par WSD seront aussi présentées.

II.3. Les approches de désambiguïsation des sens des mots (WSD)

De nombreuses approches de désambiguïsation sémantique des mots existent. Ils peuvent être divisées en : *approches basées sur les corpus d'apprentissage (endogène)* et *approches basées sur les ressources linguistiques externes (exogène).*

II.3.1.Approche endogène

Ces approches se basent sur l'utilisation d'un corpus d'apprentissage composé d'un grand nombre de contextes de mots polysémiques, dans le but d'apprendre les connaissances utiles sur le sens d'usage des mots. Cette phase d'identification automatique des connaissances est appelée apprentissage. A l'issue de cette phase, l'algorithme de désambiguïsation est capable d'assigner le sens adéquat aux mots apparaissant dans une nouvelle phrase en s'appuyant sur les connaissances acquises durant la phase d'apprentissage. Les approches de désambiguïsation basées sur les corpus d'apprentissage se distinguent en approches supervisées et approches non supervisées [26] [27].

II.3.2. Approche exogène

La plupart des approches d'indexation sémantique basées sur la désambiguïsation exogène, s'appuient en général sur *des ontologies* pour déterminer les différents sens du mot et pour désambiguïser les sens des mots. Le principe de base de l'indexation consiste alors à

extraire dans un premier temps, l'ensemble des termes descripteurs (index) du document. Il s'agit ici d'une indexation classique. Ces termes sont ensuite désambiguïsés. Pour ce faire, les sens de chaque terme d'indexation sont d'abord retrouvés à partir de *la ressource externe.* Puis, des scores sont associés aux différents sens ainsi retrouvés. Le sens qui maximise le score est alors retenu comme le sens adéquat du terme d'indexation correspondant [28].

II.4. Les ressources linguistiques structurées

Elles jouent un rôle très important dans le domaine de RI conceptuelle, elles sont utilisées pour extraire les concepts à partir des documents et requêtes. Elles offrent une meilleure représentation des documents car elles permettent de définir les relations entre les concepts des documents. Différents types de ressources sémantiques peuvent être distingués parmi lesquels se trouvent les dictionnaires informatisés, les taxonomies, les thésaurus, et les ontologies.

II.4.1. Les dictionnaires informatisés (MRD)

Représentaient des sources de connaissances très populaires dans les années 80 pour les différentes disciplines du domaine du traitement automatique de la langue. Dans un dictionnaire informatisé, un mot de la langue possède un ou plusieurs sens qui sont définis par leurs glossaires (*gloss*). Le glossaire d'un sens décrit le sens du mot par une définition, des commentaires et/ou des exemples d'utilisation courante. Comme exemples de dictionnaires informatisés, on peut citer : le *Collins English Dictionary*, le *Oxford Dictionary of English* et le *Longman Dictionary of Contemporary English* [29].

II.4.2. Une taxonomie

C'est une structure qui permet de contrôler le vocabulaire par un seul type de relation donnant la possibilité de généraliser ou de préciser un sens. Elle se présente sous la forme d'une hiérarchie simple de terme [30]**.**

II.4.3. Les thésaurus

Selon la norme internationale ISO 15143-1 : 2010[2], les thésaurus sont : «vocabulaire contrôlé ordonné dans une disposition donnée dans lequel les relations entre les termes sont affichées et identifiées». Ces termes dénotent les concepts d'un domaine particulier.

Dans un thésaurus, les termes sont organisés dans une hiérarchie de concepts liés par des relations sémantiques. Les relations présentes dans un thésaurus sont des relations taxonomiques (spécialisation /généralisation), d'équivalence (synonymie), d'association (proximité sémantique, proche-de, relié-à). Les termes d'un thésaurus peuvent servir à indexer des documents comme c'est le cas dans les thésaurus médicaux MeSh[3] et UMLS[4].

[2] http://www.iso.org/iso/catalogue_detail.htm?csnumber=37406 [date de dernière visite : Février 2016]
[3] http://www.nlm.nih.gov/mesh/ [date de dernière visite: Février 2016]
[4] http://www.nlm.nih.gov/research/umls/ [date de dernière visite: Février 2016]

II.4.4. Une ontologie

Une ontologie est une collection de concepts bien définis qui décrivent un domaine spécifique [31]. Les relations entre ces concepts peuvent être différentes d'une ontologie à une autre. Par exemple, les relations existantes dans une ontologie du domaine juridique n'ont pas les mêmes significations que celles d'une ontologie de la génétique.

Avec le niveau élevé de modélisation fourni par les ontologies, des langages de représentation ont été proposés pour simplifier la manipulation des ressources. Les langages les plus connus sont issus du W3C, comme RDF, RDF Schema, OWL et SPARQL**.**

En ce qui nous concerne pour la désambiguïsation des sens des mots de notre travail c'est l'ontologie et nous pouvons la détailler comme suit :

Les ontologies sont connues comme des outils capables de manipuler les connaissances derrière les concepts, en peut dire aussi qu'une ontologie est un ensemble structuré de concepts organisés dans un graphe (ou réseau sémantique). Elles peuvent être utilisées à différents niveaux de SRI. Les objectives de notre étude est de voir les effets d'une ontologie anglaise WordNet dans la recherche des documents et la désambiguïsation des sens des requêtes.

II.5. Les ressources linguistiques non structurées

II.5.1. Les corpus d'apprentissage

Ce sont de longs textes utilisés dans les techniques d'apprentissage pour construire la connaissance nécessaire pour la WSD. Ces corpus peuvent être étiquetés manuellement avec les sens des mots. A titre d'exemple, le corpus *SemCor* [31] est la version étiquetée du corpus *Brown* [32] avec des sens issus de WordNet.

II.5.2. Les corpus de collocations

Ce sont des ensembles de collocations de mots qui ont une tendance de se produire ensemble régulièrement. Parmi ces ressources, nous citons : *The British National Corpus collocations* et le *Collins Cobuild CorpusConcordance* [8].

Conclusion

Nous avons consacré ce chapitre à l'état de l'art sur la recherche d'information et l'indexation sémantique, à travers ses différentes sections nous concluons que la recherche d'information, s'attache à définir des modèles et des systèmes afin de faciliter l'accès à un ensemble de documents se trouvant dans des bases documentaires. Le but est de permettre aux utilisateurs de retrouver les documents dont le contenu répond à leur besoin en information, il s'agit donc de retourner l'ensemble de documents pertinents.

Puis, nous avons passé en revue l'approche d'indexation sémantique proposée en RI. Cette approche a apporté la preuve que la représentation des documents et requêtes par les sens (ou concepts) de leurs mots est bénéfique dans un processus de RI, permet ainsi de résoudre les problèmes causés par les SRI classiques. Ces sens sont le plus souvent identifiés par des approches de désambiguïsation qui utilise des ressources ext

Introduction :

Un moteur de recherche sémantique peut être vu comme un outil qui répond à des requêtes (formulées avec les concepts d'une ontologie linguistique).

L'introduction d'une ontologie linguistique dans le processus d'indexation et de recherche requiert une analyse adéquate et spécifique pour chaque langue. Dans le cas de l'anglais, la tâche s'avère plus délicate vu la disponibilité des ressources lexicale sémantique comme WordNet : la base de connaissances générales la plus utilisée, elle a servi à mettre au point ou à tester de nombreuses expériences depuis le début des années 1990. Par ailleurs, WordNet est un exemple d'une ontologie lexicale conçue et pensée pour le support électronique.

Dans ce chapitre, nous décrivons la langue anglaise et ses caractéristiques linguistiques (Section 1). Les fonctionnalités de WordNet seront aussi discutées avant de passer au chapitre suivant (Section 2).

I. SECTION 1 : La langue anglaise.

I.1. introduction

Vu la disponibilité des corpus bien traités et qualifiés en anglais et l'existence des ressources sémantique pour le traitement de la langue anglaise comme WordNet, nous avons choisi de travailler sur un corpus en anglais et utiliser WordNet comme ressource linguistique sémantique pour la désambiguïsation des sens des mots.

I.2. Histoire de la langue

L'anglais fait partie de la famille des langues germaniques occidentales et est lié au néerlandais, à l'allemand et au luxembourgeois. L'anglais est né de la fusion de plusieurs dialectes rapportés en Grande-Bretagne par les colons germaniques appelés les Angles. Il a également été influencé par le vieux norrois et normand français pendant les invasions vikings et la conquête normande. Suite à l'influence de l'Empire britannique entre le 17e et le milieu du 20e siècle, l'anglais s'est considérablement propagé à travers le monde. Encore aujourd'hui, à travers les chaînes culturelles américaines (par exemple la musique, le cinéma et la télévision) l'anglais est une lingua franca de choix dans de nombreux contextes [33].

I.3. Géographie de la langue

765 millions de personnes parlent l'anglais à travers le monde, dont 360 millions comme première langue, les 430 millions restants l'utilisant en tant que deuxième langue. À cela s'ajoute le nombre de personnes parlant l'anglais comme langue étrangère, estimé à 750 millions, soit supérieur à ceux qui l'utilisent comme première langue. On considère que 1 personne sur 4 dans le monde parle anglais, selon plusieurs niveaux de compétence. L'anglais est la langue officielle de l'Australie, du Canada, de l'Irlande, de la Nouvelle-Zélande, du Royaume-Uni et des États-Unis, ainsi que de plus de 50 autres pays à travers le monde [33].

I.4. Diffusion dans les sciences et les techniques

L'emploi de mots anglais est notable dans des secteurs comme l'informatique, les télécommunications comme le fut (et l'est toujours, d'ailleurs). Mais les nouvelles technologies (DVD multi-langues, mondialisation de l'internet) et l'adaptation des entreprises à leurs clients (CNN diffusant en plusieurs langues, Microsoft fabriquant le logiciel Windows en plusieurs langues) ont porté un coup relatif à cette domination de l'anglais. L'anglais est depuis 1951 la langue utilisée dans l'aviation, sur décision de l'OACI. De plus en plus de travaux de recherches scientifiques (thèses, études, etc.) sont rédigés en anglais ou font l'objet d'une traduction dans cette langue [33].

I.5. Les propriétés morphologiques de la langue [34] [35][36] :

- L'anglais est basé sur l'alphabet latin qui comprend vingt-six lettres et se lit de gauche à droite. Le nombre de mots existants dans la langue anglaise est estimé à plus d'un million.
- Il existe pratiquement 100 000 familles de mots dans la langue anglaise.
- Une famille de mots est un groupage de mots dérivés de la même base. Par exemple : *active*, *actively*, et *activities* « actif, activement, activités et activité » sont tous de la même famille de mots.
- L'anglais est une langue respectant l'ordre SVO (sujet, verbe, objet) dans la phrase déclarative. Exemples : *Tom does his homework* : « Tom » (sujet) « fait » (verbe) « ses devoirs » (objet).
- En général, l'élément principal se trouve au début de la phrase. Exemple : *To run quickly* « courir vite » (phrase verbale) (mais on trouve également *to quickly run*, ce qu'on appelle *the split infinitive*, l'infinitif éclaté).
- Il existe cependant des exceptions dans la langue courante. Le génitif est en premier lieu le cas du complément de nom exprimant la possession. Il s'obtient en ajoutant, selon le cas, une apostrophe et la lettre *s* ou simplement une apostrophe. Exemples :
  - *The cat's ball* (« la balle du chat »)
  - *The teenagers' ball* (« le ballon des adolescents », *teenagers* est déjà un pluriel)
  - *The children's ball* (« le ballon des enfants », *child* donne le pluriel irrégulier *children*, qui ne prend pas de *s*)
- L'ordre des mots change également quand on passe d'une phrase affirmative à une phrase interrogative. Exemple : *Are you going to the beach?* (inversion de *you are*) « Est-ce que tu vas à la plage ? ».
- La voix passive existe en anglais : *That cake was eaten by Mary* « Ce gâteau-là a été mangé par Mary ».
- Les articles définis : the house « une maison ».
- Parfois la lettre *h* n'est pas prononcée. Lorsqu'un *h* n'est pas prononcé au début d'un mot, *an* est utilisé : a horse (un cheval) an hour (une heure).
- *An* est en général utilisée plutôt que *a* lorsqu'un nom commence par une voyelle : an apple « une pomme ».

- Les pronoms personnels sujets sont *I*, *you*, *he* / *she* / *it* au singulier et *we*, *you*, *they* au pluriel ; *I* est toujours une majuscule, même s'il n'est pas situé au début de la phrase. *You* est utilisé pour s'adresser à une seule personne aussi bien qu'à plusieurs. *It* est utilisé pour désigner un objet mais il est également utilisé pour un bébé quand le sexe du bébé est inconnu, ainsi que pour un animal quand le sexe n'est pas connu ou n'est pas important. *They* est utilisé à la fois pour des personnes ou des objets.
- L'anglais n'a pas de concept de genre grammatical pour les substantifs. La différence entre féminin et masculin est pertinente seulement pour les personnes (pronoms personnels *he* et *she*) ; tous les autres noms sont neutres de fait (pronom personnel *it*). Il y a de rares exceptions, par exemple, les navires ou les pays peuvent être traités comme des féminins : *The Titanic was a famous ship.* ***She*** *hit an iceberg and sank.* (L'utilisation de *it* pour des bateaux est acceptée toutefois dans la langue courante). Pour les pays, *she* est normalement réservé au langage littéraire ou poétique. Pour les animaux dont le sexe est inconnu, *it* suffit. Si le sexe d'un animal est connu, il est acceptable de remplacer *it* par *he* ou *she* selon le cas. Cette règle se trouve également avec des bébés : *Mary has just had a baby! — Is* ***it*** *a boy or a girl?* « Mary vient d'avoir un bébé ! — Est-ce un garçon ou une fille ? ».

II. SECTION : L'ontologie linguistique WordNet.

II.1. Introduction

La recherche d'information traite le problème de trouver tous les documents pertinents dans une collection de texte pour un compte tenu de la requête de l'utilisateur. Une base de données sémantique à grande échelle telles que WordNet [37] semble avoir un grand potentiel pour cette tâche. Il y a au moins trois évident les raisons:

- Il offre la possibilité de discriminer mot détecte dans les documents et les requêtes.
- WordNet fournit la chance de faire correspondre sémantiquement mots connexes. Par exemple : fontaine, écoulement, effusion, dans le lieu sens, peuvent être identifiés comme occurrences le même concept, « écoulement naturel des eaux souterraines ». Et au-delà de la la relation sémantique de synonymie,
- WordNet peut être utilisé pour mesurer la distance sémantique entre survenant termes pour obtenir des moyens plus sophistiqués de la comparaison des documents et des requêtes.

II.2. Domaines de WordNet

Des programmes issus du monde de l'Intelligence Artificielle ont également établi des passerelles avec WordNet. Le WordNet est utilisable librement, y compris pour un usage commercial, ce qui en a favorisé une diffusion très large. Plusieurs autres ressources linguistiques ont été constituées (manuellement ou automatiquement) à partir de, en extension à, ou en complément à WordNet. L'ensemble de ces ressources linguistiques constitue un

système complet couvrant des aspects lexicaux, syntaxiques et sémantiques. Combinées, ces ressources fournissent un point de départ intéressant pour des développements sémantiques dans le cadre du Web sémantique, tels que la recherche d'information, l'inférence pour la compréhension automatique de textes, la désambiguïsation lexicale ou la résolution d'anaphores [38].

II.3. Un projet ambitieux

Depuis 1985, un groupe de psycholinguistes et de linguistes de l'université de Princeton a développé une base de données lexicale selon des principes suggérés par des expériences et des recherches en psycholinguistique sur l'organisation de la mémoire humaine. Depuis cette date, ce projet a pris de l'ampleur ; il se poursuit encore de nos jours.

C'est un réseau sémantique de la langue anglaise, qui se fonde sur une théorie psychologique du langage. La première version diffusée remonte à juin 1991. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise [39]. Des versions de WordNet pour d'autres langues existent (Wolf pour le français et ArabicWordNet pour l'arabe et EuroWordNet[5]), mais la version anglaise est cependant la plus complète et riche à ce jour.

WordNet est distribué sous une licence libre, permettant de l'utiliser commercialement ou à des fins de recherche. La dernière version distribuée en avril 2013 est la 3.1. Cette version est par ailleurs consultable en ligne [40].

II.4. Principe

On peut considérer WordNet comme un graphe ou un réseau lexicale sémantique, souvent qu'on qualifie d'ontologie légère (Light Ontology), où :

- Les synsets sont les nœuds.
- Les relations sémantiques entre synsets sont les arcs.

II.4.1. Les synsets

La composante atomique sur laquelle repose le système entier est le *synset* c'est un groupe de mots interchangeables, dénotant un sens ou un usage particulier. La version 2.0 de WordNet définit ainsi le nom commun anglais « *car* » à l'aide de cinq synsets comme il est montré dans la figure II.1.

[5] ***EuroWordnet***, un projet de construction d'un ***WordNet*** multilingue a été lancé en mars 1996 (Vossen, 1996). Il concerne initialement l'allemand, l'italien et l'espagnol. La France accuse un certain retard.

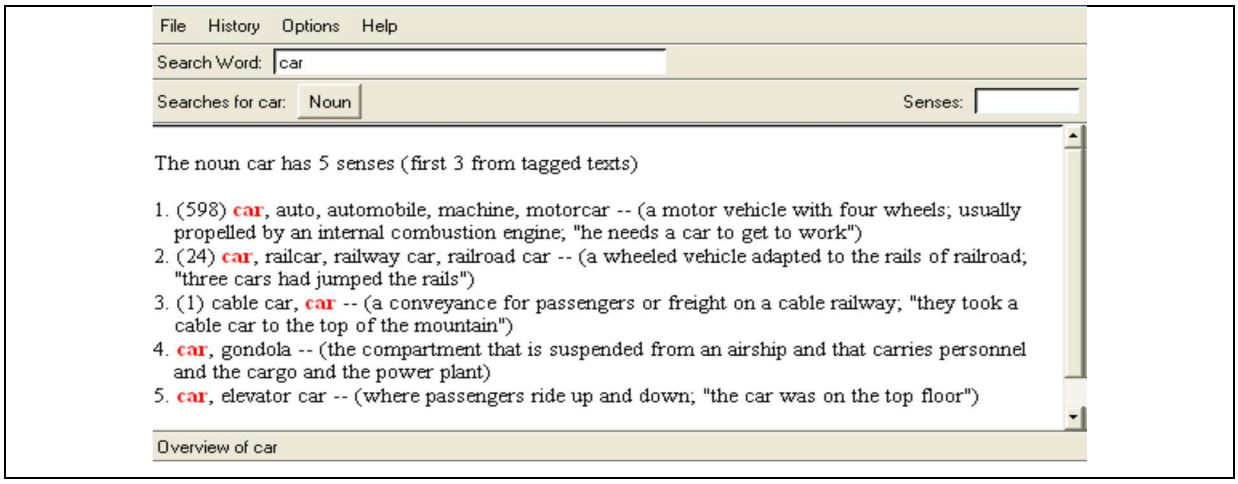

Figure II.1. Les différents sens du mot « car »

Chaque synset dénote une acception différente du mot « *car* », décrite par une courte définition. Une occurrence particulière de ce mot dénotant par exemple le premier sens (le plus courant), dans le contexte d'une phrase ou d'un énoncé, serait ainsi caractérisée par le fait qu'on pourrait remplacer le mot polysémique par l'un ou l'autre des mots du synset sans altérer la signification de l'ensemble [39].

II.4.2. Les relations sémantiques

Dans WordNet, les concepts sont reliés par des relations sémantiques. La relation de synonymie est la relation de base dans WordNet. Elle relie les termes d'un même noeud. Les noeuds (les concepts ou les synsets) sont reliés entre eux par des relations sémantiques telles que, la relation de composition (partie-tout) et la relation hyponymie-hyperonyme (est-un) [39]. comme représentées dans le schéma de la Figure II.2.

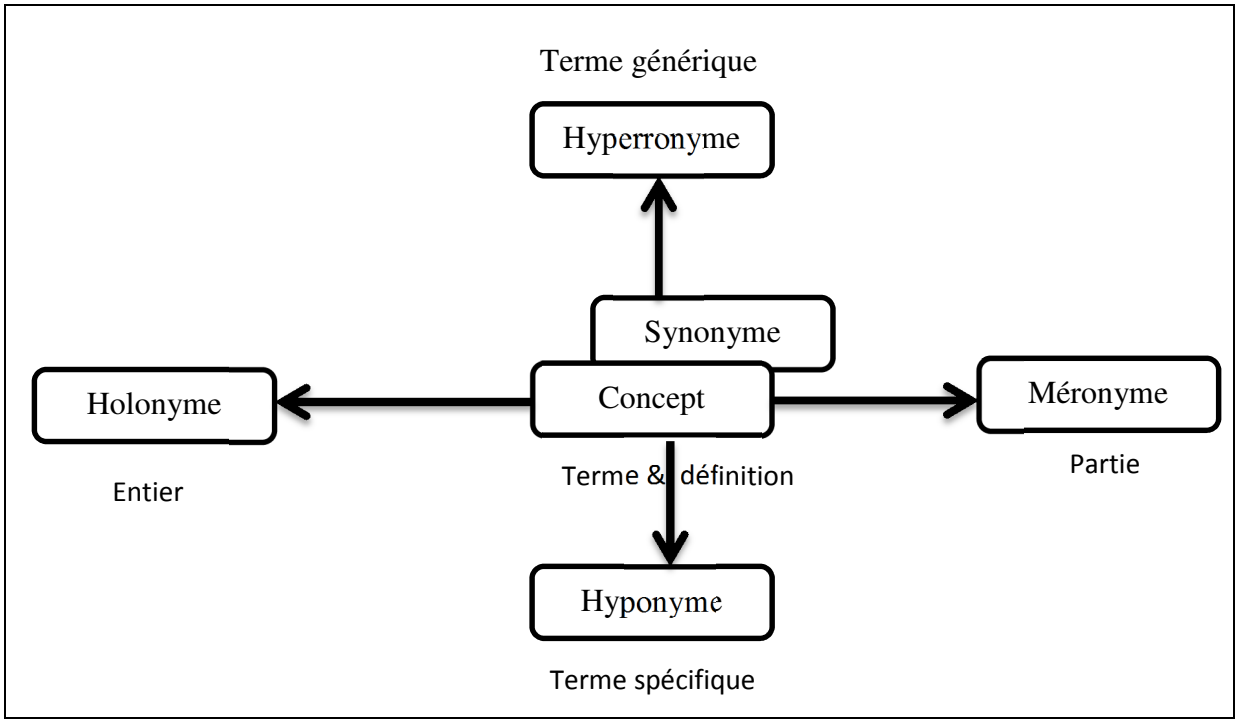

Figure II.2. Les relations entre les Synsets.

II.4.2.1. L'hyperonymie

L'hyperonymie est la relation sémantique hiérarchique d'un lexème à un autre selon laquelle l'extension du premier terme, plus général, englobe l'extension du second, plus spécifique. Le premier terme est dit hyperonyme de l'autre, ou super ordonné par rapport à l'autre. C'est le contraire de l'hyponymie [37].

II.4.2.2. L'hyponymie

L'hyponymie est une relation d'inclusion entre deux mots dont l'un est l'hyponyme de l'autre. La relation d'hyponymie est l'expression linguistique de la relation logique d'inclusion d'une classe dans une autre « on dit qu'un concept représenté par le synset {x, x',…} est l'hyponyme du concept représenté par le synset {y, y',…} si les locuteurs dont l'anglais est la langue maternelle acceptent les phrases du type *Un x est une sorte de y.* ».

On peut aussi définir les hyponymie comme la relation sémantique d'un lexème à un autre selon laquelle l'extension du premier est incluse dans l'extension du second. Le premier terme est dit hyponyme de l'autre. C'est le contraire de l'hyperonymie. [37] donne un exemple de chaîne hyponymique : *televangelist* < *evangelist*< *preacher* < *clergyman* < *spiritual leader* <*person*[6]

II.4.2.3. La méronymie

La méronymie est une relation sémantique entre mots d'une même langue. Des termes liés par méronymie sont des méronymes. La méronymie est une relation partitive hiérarchisée : une relation de partie à tout. Un méronyme X d'un mot Y est un mot dont le signifié désigne une sous-partie du signifié de Y. La relation inverse est l'holonymie. WordNet inclus trois types de méronymie :

- X est un composante de Y.
- X est un élément de Y.
- X est le matériau dont Y est constitué [41].

II.4.2.4. L'holonymie

L'Holonymie est une relation sémantique entre mots d'une même langue. Des termes liés par holonymie sont des holonomes. L'holonymie est une relation partitive hiérarchisée : un holonyme A d'un mot B est un mot dont le signifié désigne un ensemble comprenant le signifié de B. La relation inverse est la méronymie [41].

[6] Dans x < y, le mot *x* est donné comme l'hyponyme du mot *y*. On aurait pour le français la séquence suivante : *télé-évangéliste* < *évangéliste* < *prédicateur* < *ecclésiastique* < *chef spirituel* < *personne*.

II.4.2.5. La Synonymie

La synonymie est un rapport de similarité sémantique entre des mots ou des expressions d'une même langue. La similarité sémantique indique qu'ils ont des significations très semblables. Des termes liés par synonymie sont des synonymes.

Il existe des bases de données de synonymes, présentées comme des dictionnaires, librement téléchargeables. On en trouve aussi vendues ou consultables sous la forme de livres, de logiciels, ou de web, ou des jeux [42].

II.4.2.6. L'antonymies

Deux items lexicaux sont en relation d'antonymie si on peut exhiber une symétrie de leurs traits sémantiques par rapport à un axe. La symétrie peut se décliner de différentes manières, selon la nature de son support. On distingue plusieurs supports qui sont autant de type d'antonymie :

- Les antonymes complémentaires.
- Les antonymes scalaires.
- Les antonymes duals [42].

II.4.2.7. La Troponymie

La troponymie est une relation sémantique entre deux verbes, l'un décrivant de manière plus précise l'action de l'autre. Le premier verbe est dit troponyme du second [41].
La figure ci-dessous montre de manière simplifiée comment le premier sens de *credit* (*crédit*) se situe par rapport aux synsets voisins : c'est un hyponyme de *asset* (*avoir*), un hyperonyme de *credit-card* (*carte de crédit*), un antonyme de *cash* (*argent comptant*).

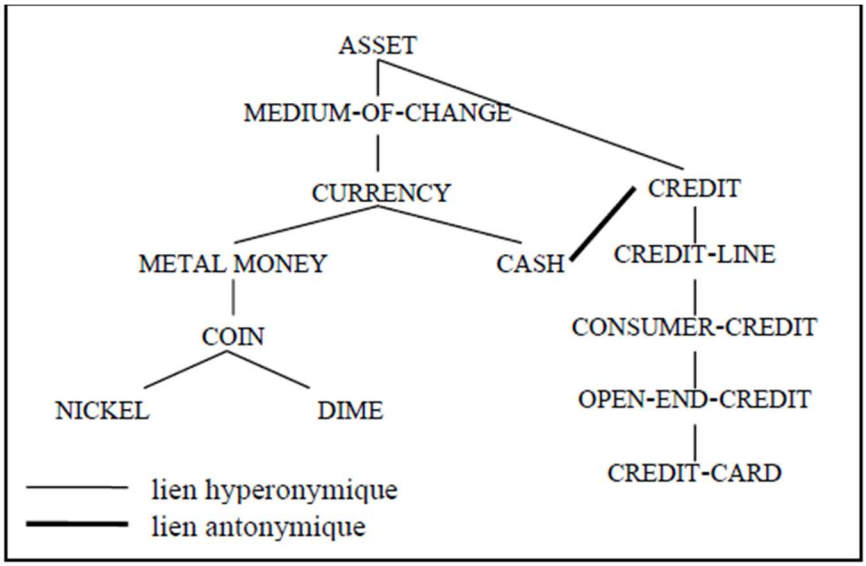

Figure II.4. Exemple de sous-hiérarchie de WordNet.

II.5. Une structure riche et différenciée :

WordNet décompose le lexique en cinq catégories : noms, verbes et adjectives, adverbes. Chacune de ces catégories a sa propre structure interne. « Ce sont des expériences

sur les associations de mots qui ont mis en évidence à l'origine que l'organisation varie d'une catégorie syntaxique à l'autre. ».

II.5.1 Des hiérarchies de noms

L'ensemble des noms, qui comporte des formes simples et des mots composés mais pas de noms propres, est organisé autour de la relation d'hyponymie. La structure induite est en fait un ensemble de 25 hiérarchies dominées par des catégories sémantiques générales

Cette structure hiérarchique peut être parcourue de haut en bas ou de bas en haut. À partir d'un sens donné, on peut ainsi retrouver ses ancêtres (hyperonymes directs et indirects), ses descendants (hyponymes directs ou indirects) mais aussi ses frères.

Outre leur place dans cette structure hiérarchique, les sens des noms se définissent par des propriétés : leurs attributs, leur composition et leurs fonctions. La composition est décrite par différents types de relations méronymiques dans WordNet : les relations de composant à objet composé (*branche / arbre*), d'élément à ensemble (*arbre / forêt*) et de matière (*arbre / bois*). En revanche, les attributs (un arbre peut être grand, vieux…) et les fonctions (*une hache sert à couper…*) ne sont pas représentés dans WordNet. Ce sont en effet des relations transcatégorielles qui devraient à terme relier les hiérarchies de noms aux réseaux des adjectifs ou des verbes [43].

II.5.2. Des classes d'adjectifs

Les synsets d'adjectifs comprennent essentiellement des adjectifs qualificatifs. Ces adjectifs ne s'organisent pas comme les noms. Pour les adjectifs, il n'existe pas de relation hiérarchique comme l'hyponymie. La relation fondamentale structurant l'espace des adjectifs est l'antonymie. Cette relation symétrique, mise en évidence par des tests psycholinguistiques sur les associations de mots, est difficile à formaliser. Les auteurs retiennent l'idée que les adjectifs antonymes expriment deux valeurs opposées d'un même attribut [44].
Partant cependant du constat que certains adjectifs proches par le sens par exemple : *heavy* et *weighty* « lourd/pesant » ont des antonymes différents *light* et *weigthless* « lumière/en état d'apesanteur » et que beaucoup d'adjectifs qualificatifs *ponderous* « lourd » n'ont pas d'antonymes directs, la structure retenue est celle de classes d'adjectifs similaires entre eux, ces classes étant organisées autour d'adjectifs pôles qui peuvent s'opposer à d'autres pôles par des liens d'antonymie. *heavy* et *light* sont donc considérés comme antonymes, mais *ponderous*, qui est similaire à *heavy* et qui n'a pas d'antonyme direct n'est qu'un antonyme indirect de *light* [39].

II.5.3 Des réseaux de verbes

Comme les noms et les adjectifs, les verbes sont regroupés en synsets. Ceux-ci comportent des formes simples mais aussi des tournures verbales, comme *look up*

« chercher », qui sont très fréquentes en anglais. Les synsets se répartissent eux-mêmes en 15 catégories générales.

La relation centrale pour le réseau des verbes n'est ni l'hyponymie, ni l'antonymie, mais l'implication. En distingue quatre types : la cause (*give* / *have* : *donner* / *avoir*), la présupposition (*succeed* / *try* : *réussir* / *essayer* ou *untie* / *tie* : *dénouer* / *nouer*), l'inclusion (*snore* / *sleep* : *ronfler* / *dormir* ou *buy* / *pay* : *acheter* / *payer*) et la troponymie (*limp* / *walk*, *boiter* / *marcher*).

Soulignant toutefois la complexité de la sémantique des verbes et la difficulté de définir une sémantique proprement différentielle, les auteurs de WordNet reconnaissent la moindre maturité du réseau des verbes.

Dans la pratique, les travaux qui exploitent ce réseau des verbes à des fins de désambiguïsation lexicale s'en tiennent souvent aux grandes catégories sémantiques [45].

II.6. Quelques données statistiques

Dans cette partie, nous présentons, de manière quantitative, le contenu de WordNet. La table II.1 montre la structure de WordNet en nombre de mots, nombre de synsets et nombre de sens globalement et par catégorie grammaticale. Du nombre total de formes décrites, la plupart sont des noms (74.6%), le reste étant constitué par des adjectif (14.6%),des verbes(7.6%) et des adverbes(3.2%). La polysémie (nombre de sens par mot) se manifeste dans WordNet par le fait qu'il y a des mots qui peuvent appartenir à plusieurs synsets (146350 formes traitées /111223 synsets) [46].

Partie de discours	**Nombre de mots**	**Nombre de synsets**	**Nombre de sens**
Noms	109195	75804	134716
Verbes	11088	13214	24169
Adjectifs	21460	18576	31184
Adverbes	4607	3629	5748

Tableau II.1. Nombre de mots, synsets et sens sans WordNet.

La taille du vocabulaire couvert suffit à donner la mesure de l'ambition qui a présidé à la construction de ce réseau. WordNet comporte 95 600 unités lexicales différentes : 51 500 mots simples et 44 100 expressions (*collocations*). À ces mots sont associés quelques 70 100 sens différents. Le tableau II.2 montre comment ces unités et sens se répartissent [46].

	Noms	Verbes	Adjectifs
Nombre d'unités lexicales	57000	21000	19500
Nombre de sens	48800	8400	10000
Nombre de catégories générales	25	14	

Tableau II.2. Exemple de sous-hiérarchie de WordNet.

II.7. Les points forts de WordNet :

L'utilisation de WordNet en recherche d'informations :

- Pour étendre la requête de l'utilisateur (ajout de synonymes, par exemple pour augmenter le rappel, c'est-à-dire la proportion de documents pertinents rapportés).
- Acquisition de relations sémantiques.
- Désambiguïsation sémantique.
- Pour l'étiquetage sémantique de corpus.
- Pour la structuration et catégorisation des documents.

En général WordNet est utilisé :

- Pour la recherche d'informations.
- Pour l'extraction d'informations.
- Pour les systèmes de questions/réponses.
- Pour enrichir la représentation avec des synonymes, hyperonymes, etc. [46].

Ceci nous amène à souligner l'absence de ressources similaires pour le français. Si la recherche sur les corpus en français peut sans doute tirer profit de l'expérience anglo-saxonne pour éviter certains tâtonnements, des problèmes spécifiques se posent pour chaque langue, qui imposent certains ajustements, voire la mise au point de méthodes particulières ou le développement d'outils spécifiques.

L'absence de ressources lexicales informatisée pour le français est déjà un frein pour tous les traitements sémantiques. Faute de moyens, la plupart des travaux français s'intéressent à l'acquisition de connaissances à partir de corpus.

II.8. Les limite du WordNet :

II.8.1. Informations manquantes :

WordNet ne précise pas l'étymologie, la prononciation, les formes de verbes irréguliers.

II.8.2. Profusion de sens pour un mot donné

La contrepartie de son importante couverture est que WordNet est très précis dans le sens des définitions. On a une granularité très fine des sens. Par exemple, le verbe *to give* (« donner ») n'a pas moins de 44 sens. Une telle profusion ne facilite pas une tâche de désambiguïsation lexicale.

II.8.3. Absence de relations pragmatiques

Conclusion

Dans ce chapitre nous avons décrit la langue anglaise et ses caractéristiques linguistiques, nous avons aussi présenté en détail WordNet et son principe de fonctionnement qui est basé sur la notion de synset et de relation sémantique. Â modélisation de notre travail avec le langage de modélisation *UML (UnifiedModellingLanguage).*

1. Introduction

Tout au long de ce chapitre, nous allons identifier les fonctionnalités du système à réaliser, ce qui nous conduira à la description des besoins de notre système ainsi que l'analyse et la conception objet de ces besoins. Ce qui nécessite des méthodes permettant de mettre en place un modèle, parmi lesquelles nous avons choisi le langage UML, nous décrivons, par les différents diagrammes UML, notre modélisation.

2. Qu'est-ce que UML ?

UML (Unified Modelling Language) le langage de modélisation unifié, est un langage qui s'impose à l'heure actuelle comme le standard de modélisation des applications informatiques : il est Utilisé dans le développement logiciel, dans la conception orientée objet et la modélisation. Il propose plusieurs types de diagrammes qui permettent de modéliser tous les aspects d'une application informatique [47].Nous sommes intéressés aux diagrammes suivants pour modéliser notre système :

- Le diagramme de cas d'utilisation utilisé pour l'expression des besoins.
- Le diagramme de séquence pour décrire les procédures les plus importantes.
- Le diagramme de classes pour la modélisation des données et des relations entre-elles.
- Le diagramme d'activité afin de montrer l'enchaînement des activités des acteurs du système.
- Le diagramme de composants dans le but de décrire l'organisation du système du point de vue des éléments logiciels.
- Le diagramme de déploiement utilisé pour la disposition physique des ressources matérielles qui composent le système.

Afin de vous accompagner dans nos modélisations UML, nous vous présentons la plateforme OpenSource **StarUML**[48].

3. L'outil StarUML

On a utilisé le logiciel StarUML comme outil de modélisation, StarUML est une plate-forme de génération des modèles basés sur le langage UML. L'avantage de cet outil UML est le fait que tous les diagrammes UML peuvent être générés, ainsi que l'exportation au format JPG afin d'intégrer les diagrammes au sein de documents [49].Les langages de programmation Com, C++, C# et Delphi sont pris en charge. StarUML supporte également l'architecture MDA[7] qui offre comme avantage la personnalisation des profils UML.

[7] MDA (*Model Driven Architecture*) est un processus de l'ingénierie dirigée par les modèles (ou MDE pour *Model Driven Engineering*). Proposée par l'OMG (*Object Management Group*) en 2000, l'approche MDA est basée sur la séparation des préoccupations. Elle permet prendre en compte, séparément, aspect métier et aspect technique d'une application, grâce à la modélisation. Le code source de l'application est obtenu par génération automatique à partir des modèles de l'application. Les modèles ne sont plus seulement un élément visuel ou de communication, mais sont, dans l'approche MDA, un élément productif et le pivot du processus MDA.

En somme, StarUML est complet, robuste et présente une pléthore d'outils et de paramètres. Il est toutefois dédié principalement aux utilisateurs chevronnés et aux projets complexes [48].

4. Diagramme de cas d'utilisation :

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur. Nous présentons dans (Figure III.1) le diagramme de cas d'utilisation global de notre application.

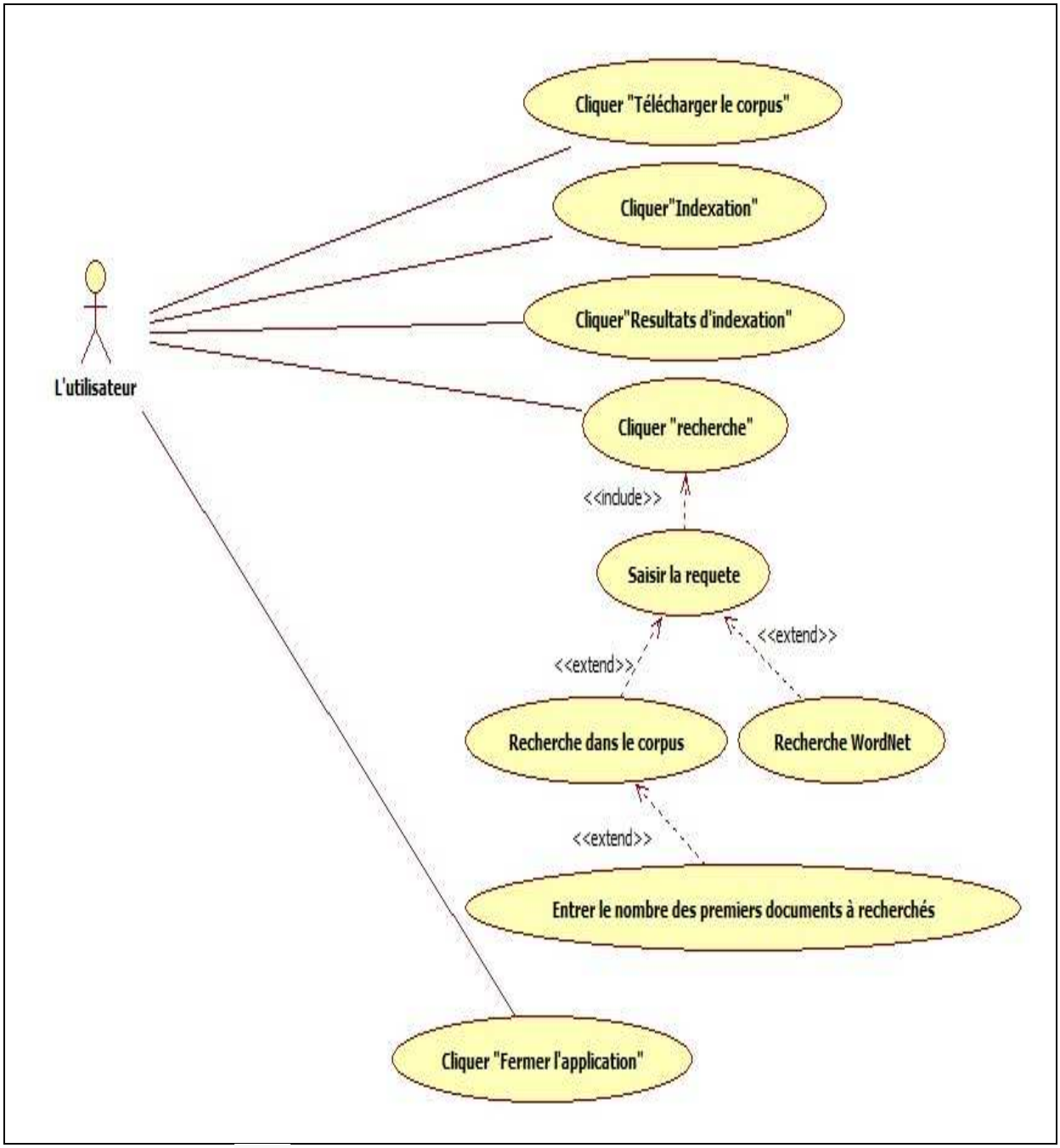

Figure III.1. Diagramme de cas d'utilisation du système en générale.

5. Diagramme de classe :

Le diagramme de classes UML décrit les structures d'objets et d'informations utilisées par notre application, il fournit une vue conceptuelle de l'architecture de notre application voir (Figure III.2).

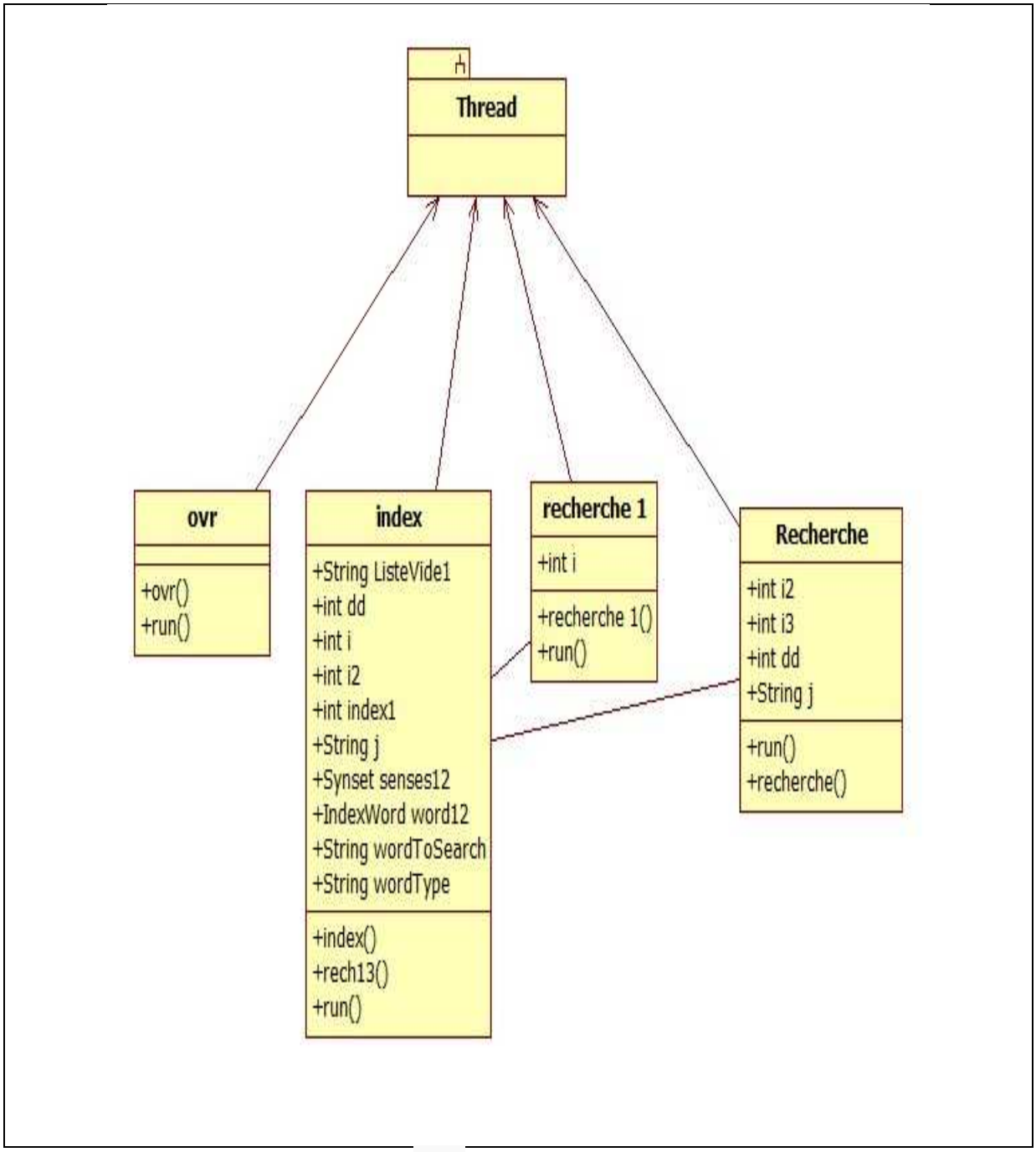

Figure III.2. Diagramme de classe.

6. Diagramme de séquence

Le séquencement de différentes tâches effectuées par les acteurs (Utilisateur et Système) est montré par le diagramme de séquence dans la Figure suivante :

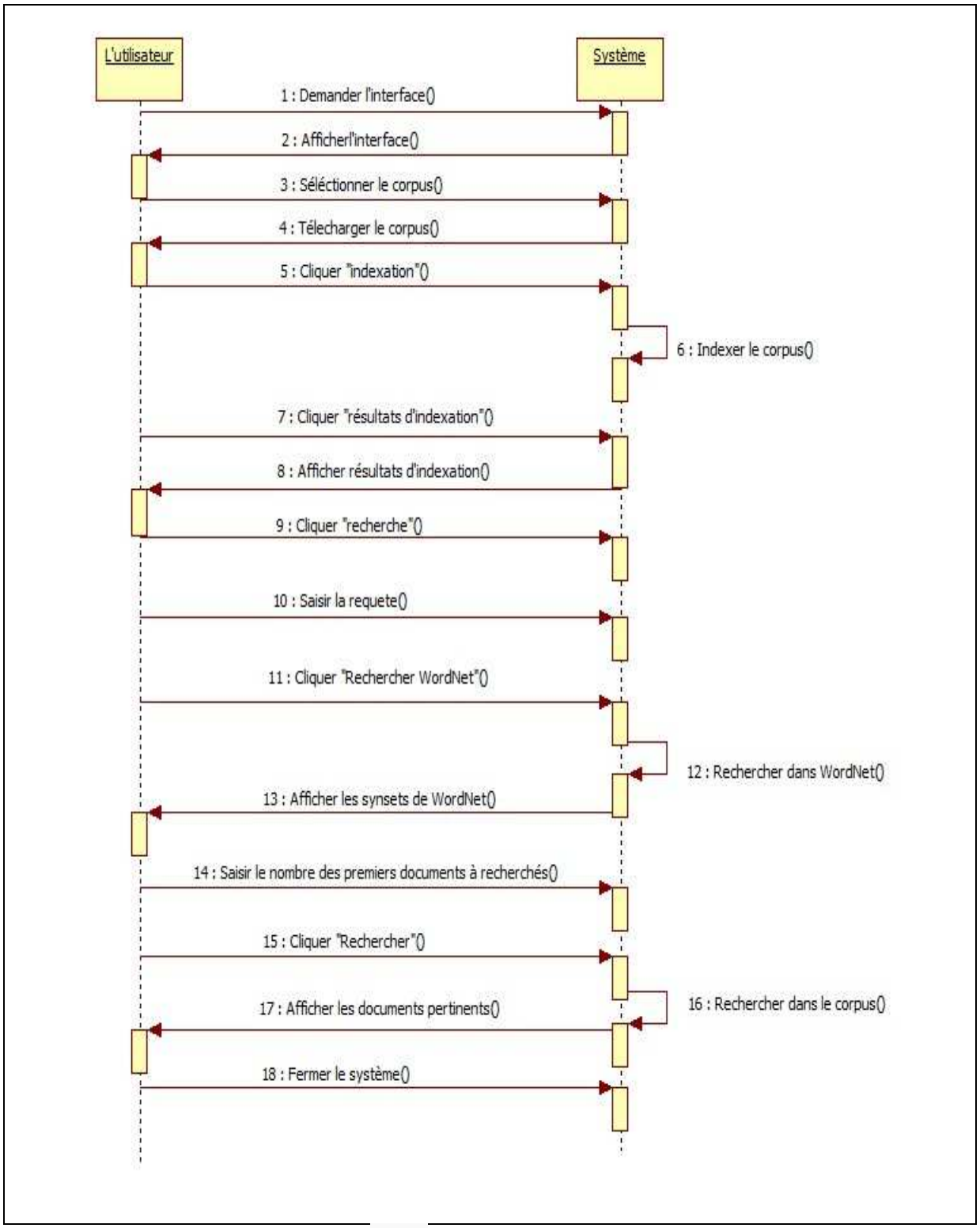

Figure III.3. Diagramme de séquence.

7. Diagramme d'activités

Nous montrons par la (Figure III.4) l'enchainement de toutes les activités qui peuvent être réalisés par notre système et l'utilisateur.

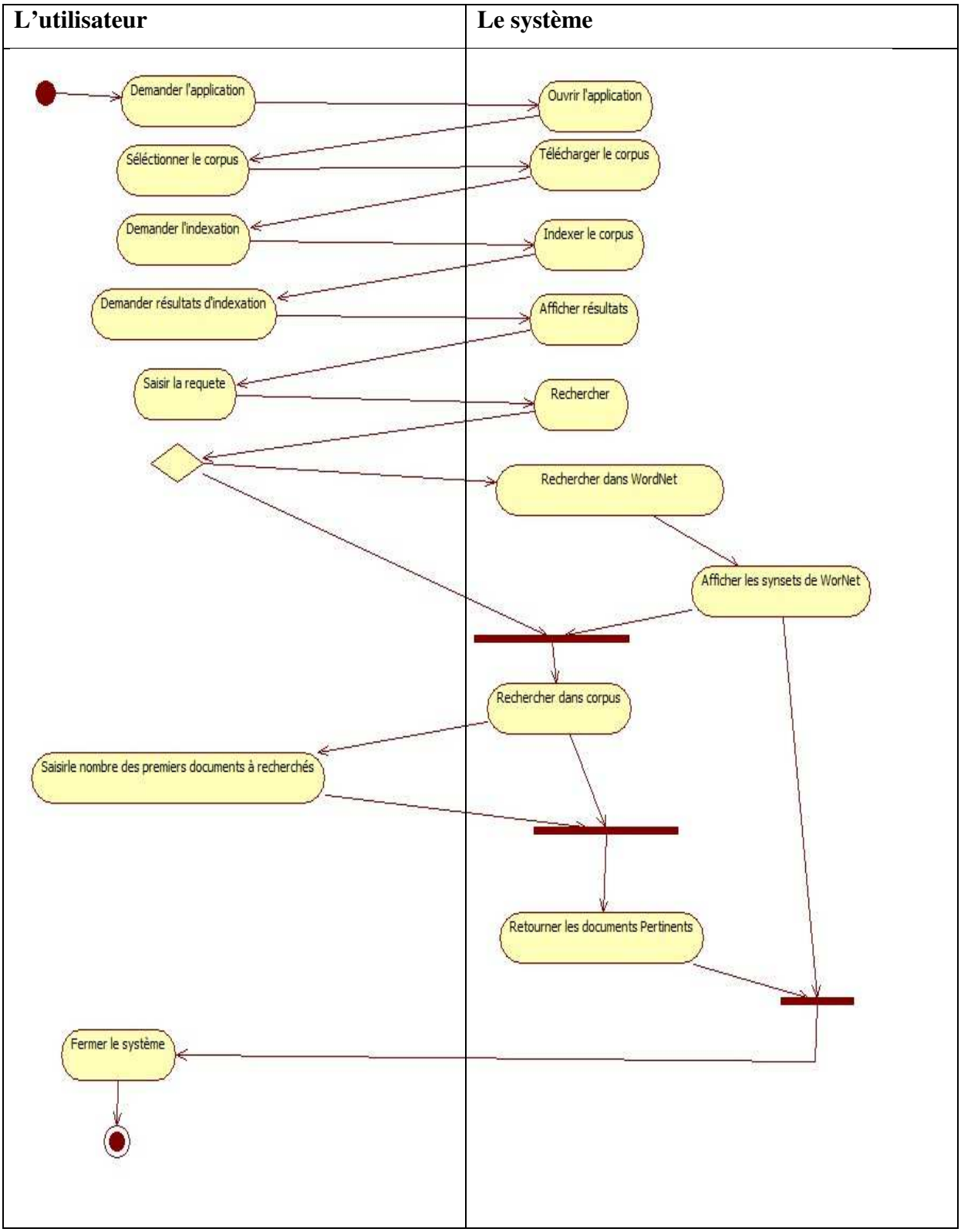

Figure III.4. Diagramme d'activités.

8. Diagramme de composant :

Nous décrivons par la Figure ci-dessus tous les éléments logiciels et composants de notre application.

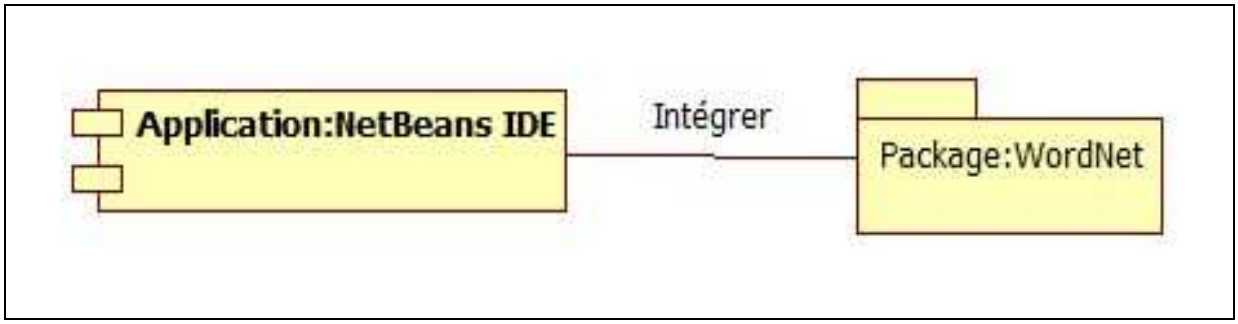

Figure III.5. Diagramme de composant.

9. Diagramme de déploiement :

La disposition physique des ressources matérielles qui composent notre système est monté par le diagramme de déploiement subséquent.

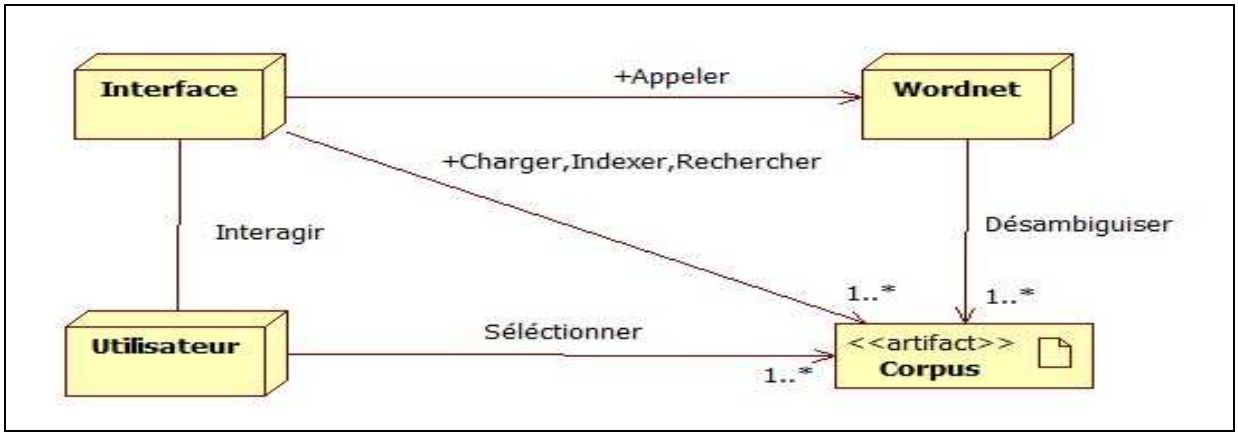

Figure III.6. Diagramme de déploiement.

10.Conclusion

Dans ce chapitre, nous avons fait une présentation générale de la modélisation avec le langage de modélisation *UML* ainsi que l'outil StarUML. Nous avons documenté et détaillé les tâches (par des diagrammes d'UML) que nous allons réaliser dans le chapitre suivant.

1. Introduction :

Il s'agit maintenant de mettre en œuvre les étapes explicitées dans les chapitres précédents pour concevoir et implémenter une interface d'indexation et de recherche d'informations sémantique d'un corpus Anglais.

Nous allons dans un premier temps présenter notre environnement d'implémentation : les ressources et les outils utilisées. Ensuite, en expliquant les étapes de prétraitements, il s'agit de la conception. Et terminera par la présentation de différentes interfaces et fonctionnalités de notre application

2. Le corpus utilisé :

Nous avons utilisé un corpus en anglais composé de 300.000 documents de domaine médical chimique de taille 324 MO, composé d'un ensemble d'articles de différents journaux d'Amérique couvrant la période 1988-1991.

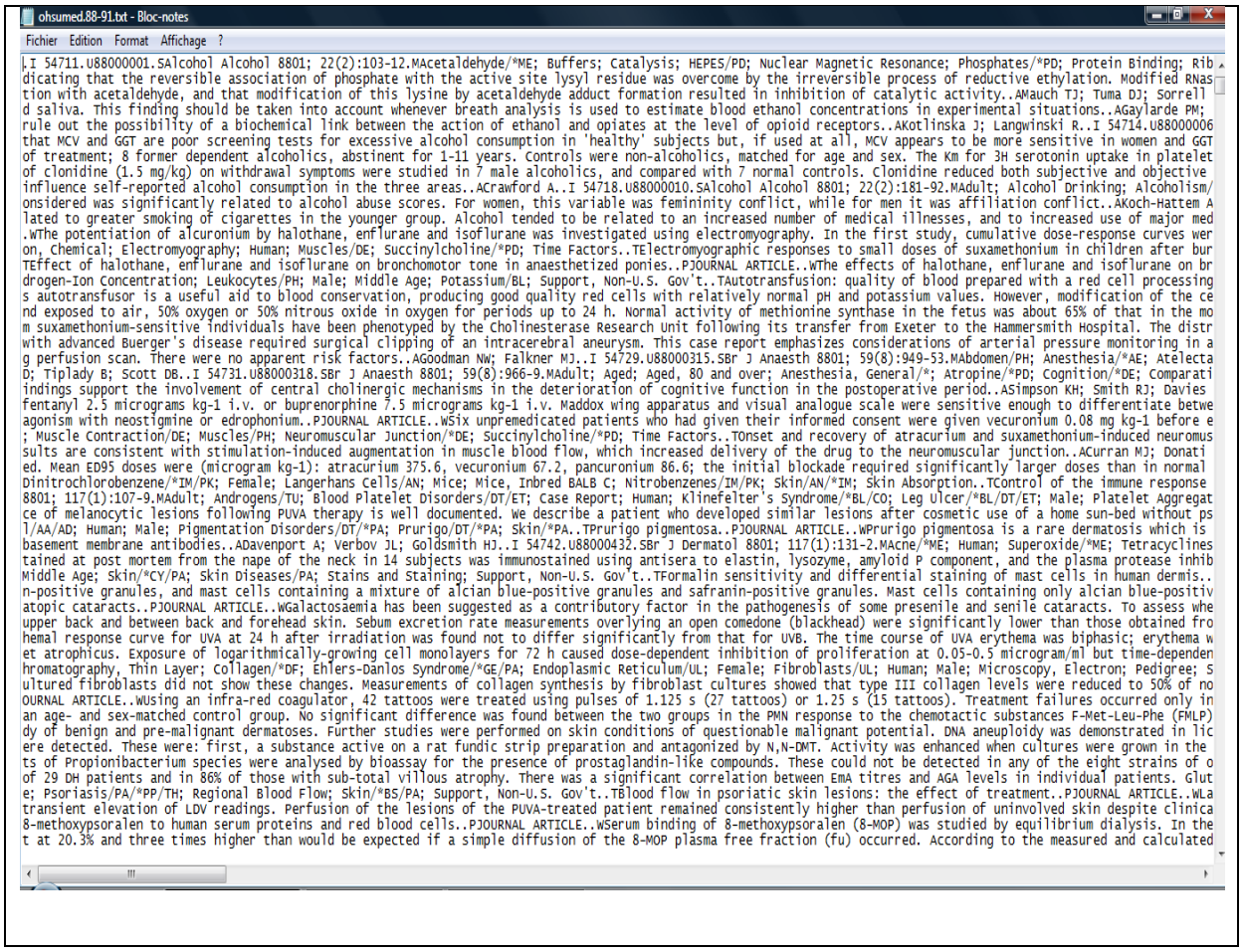

Figure IV.1. Le corpus du système.

3. L'environnement de l'application

L'implémentation et les tests de notre application ont été réalisés dans l'environnement matériel et logiciel suivant :

- Processeur : AMD Sempron ™ SI-42 2.10 GHz.
- Mémoire installée (RAM) : 2.00 GO.
- MS-Windows Vista™ Edition Familiale Basique, Type de système : système d'exploitation 32 bits.
- Java sous l'environnement NetBeans IDE 8.0.2.
- package de l'ontologie anglaise WordNet 3.0 pour la désambiguïsation des sens des mots.
- package de moteur de recherche lucene pour l'étape de racinisation pendant l'indexation.
- Corpus anglais de 300.000 documents du domaine médical chimique.

3.1. Langage d'application

Java est un langage de programmation et une plate-forme informatique créée par Sun Microsystems en 1995, racheté plus tard par Oracle. Il s'agit de la technologie sous-jacente qui permet l'exécution des applications modernes sur différentes plateformes. La portabilité, des programmes Java sur différents systèmes d'exploitation, représente son atout principal. Java est utilisée sur plus de 850 millions d'ordinateurs de bureau et un milliard de périphériques dans le monde, dont des périphériques mobiles et des systèmes de diffusion télévisuelle [50].

3.2. IDE NetBeans

NetBeans, créé à l'initiative de Sun Microsystems, présente toutes les caractéristiques indispensables à un IDE de qualité, que ce soit pour développer en Java, Ruby, C/C++ ou même PHP.

De licence Open Source, NetBeans permet de développer et déployer rapidement et gratuitement des applications graphiques Swing, des Applets, des JSP/Servlets, de l'architecture J2EE, dans un environnement fortement personnalisable [51].

3.3. Bibliothèques de WordNet utilisé

Nous avons utilisé les deux bibliothèques de WordNet afin de manipuler les méthodes de ses différents class java suivant :

- WordNetGlossTagLibrary : com.gtl.GlossTag;
- JWNL(Java WordNet Library) : net.didion.jwnl.JWNL;
  net.didion.jwnl.JWNLException;
  net.didion.jwnl.data.IndexWord;
  net.didion.jwnl.data.POS;

net.didion.jwnl.data.Synset;

net.didion.jwnl.data.Word;

net.didion.jwnl.dictionary.Dictionary;

3.4. Biblithèque de Lucene utilisé

Est un moteur de recherche et d'indexation développé dans le projet Apache. C'est un logiciel open source signifiant que son code source est libre et accessible gratuitement. Ce logiciel est une librairie de fonctions de recherche dans contenu textuel des documents. Il inclut une interface de programmation (API).

A la base, *Lucene* est écrit en Java mais il est maintenant disponible pour d'autres langages de programmation tels que Python, PHP, Delphi, Perl, C++, C# et Ruby. *Lucene* peut être utilisé avec de nombreux systèmes, c'est une multiplateforme pour : Windows, MacOS et Linux où il est plus précisément intégré à Ubuntu, Debian et Redhat [52].

Lucene est capable de traiter de grands volumes de documents grâce à sa puissance et à sa rapidité dues à l'indexation.

Nous avons utilisé la bibliothèque de lucene suivant dans le but d'effectuer l'étape de racinisation de la phase de normalisation d'indexation :

- org.apache.lucene : org.apache.lucene.analysis.EnglishAnalyzer;
  org.apache.lucene.queryParser.QueryParser;
  org.apache.lucene.util.Version;

4. Processus d'indexation

Avant de naviguer dans le corpus, il est utile d'appliquer certains prétraitements, ces prétraitements sont les étapes les plus importantes des phases du processus d'indexation (CHAPITRE I) suivant :

4.1. Segmentation

Nous avons réalisé cette phase selon 2 étapes :

Etape 1 :

Permet de segmenter le corpus en un ensemble de documents dans le cas de notre corpus il s'agit de construire des nouveaux documents en se basant sur le caractère « .I » cette étape se déroule comme suit :

Entrée: Corpus. Sortie: Corpus segmenté en documents.
Tant que (le corpus n'est pas fini) faire Si (trouver le caractère « .I ») Alors Créer un nouveau document Retourner document;

Fin; Fin;

Tableau IV.1. Algorithme de segmentation d'un corpus en un ensemble de documents.

Etape2 :

En séparant les mots entre eux en se basant sur le caractère de blanc, la procédure se déroule comme suit :

Entrée: document. Sortie: document segmenté en mots.
Tant que (le document n'est pas fini) faire Si (trouver un délimiteur d'espace) Alors Découper les mots de document Retourner sac de mots; Fin; Fin;

Tableau IV.2. Algorithme de segmentation d'un document en sac de mots.

4.2. Normalisation

Comme nous l'avons déjà précisé dans le premier chapitre, la normalisation traite plusieurs niveaux pour manipuler les variations du texte, nous avons exécuté plusieurs genres de normalisation sur le texte de corpus.

4.2.1. Niveau syntaxique

- **Elimination des caractères spéciaux :**

En désignant l'ensemble des caractères spéciaux (chiffres et symboles) comme :

[" "+"," + "." + "%" + "0" + "1" + "2" + "5" + "3" + "4" + "6" + "7" + "8" + "9" + ";"+ "!" + "?" + "~" + "/"+"+"+"-"+"*"+";"+""+")"+"("+"{"+"}"+"["+"]"+"\"+ """+":"+"§"]

Entrée: document segmenté en mots. Sortie: document normalisé
Pour chaque mot de document faire Si (mot est un caractère spécial) Alors Supprimer le mot de document Retourner document Fin; Fin.

Tableau IV.3. Algorithme d'élimination des caractères spéciaux.

- **Elimination des mots vides**

Un des problèmes majeurs de l'indexation consiste à extraire les termes significatifs et à éviter les mots vides. Nous distinguons deux techniques pour éliminer les mots vides :

➔L'utilisation d'une liste de mots vides (aussi appelée anti-dictionnaire).
➔L'élimination des mots dépassant un certain nombre d'occurrences dans la collection.

Nous avons utilisé la première technique. L'élimination des mots vides à l'avantage de réduire le nombre de termes d'indexation, elle peut cependant augmenter le taux de rappel ; c'est à dire la proportion de documents pertinents retournés par le système par rapport à l'ensemble des documents pertinents.

La liste des mots vides contient les pronoms personnels, les prépositions, les articles….etc. Nous avons utilisé la liste des mots vides suivante :

```
String Listvide[] ={ "\n",
"a","\"","«","»","a","about","above","after","again","against","all","am","an","and","any","are","aren","arent"
,"as","at","be","because","been","before","being","below","between","both","but","by","can","cannot","can't"
,"could","couldnt","did","didn't","do","does","doesn't","doing","don't","down","during","each","u","m","l","q"
,"j","few","for","from","further","had","has","hasn't","have","haven't","having","he","hed","hell","hes","her"
,"here","heres","hers","herself","him","himself","his","how","hows","i","id","if","i'll","im","in","into","is","isn't"
,"it","it'll","its","itself","i've","let","lets","me","more","most","mustnt","my","myself","no","nor","not","now"
,"of","off","on","once","only","or","other","others","ought","our","ours","ourselves","out","over","own","p"
,"page","pages","part","past","per","perhaps","placed","please","plus","poorly","possible","possibly","pp"
,"present","proud","put","q","que","quickly","quite","qv","r","ran","rather","rd","re","really","recent","ref"
,"refs","regards","related","relatively","research","resulted","resulting","results","right","run","s","said","same"
,"saw","say","says","sec","section","see","seem","seemed","seems","seen","self","sent","seven","she","shed"
,"she'll","shes","should","shouldn't","show","shows","since","six","so","some","sorry","such","sup","sure"
,"t","take","tell","tends","than","thank","that","that'll","thats","that've","the","their","them","then","there"
,"therd","there'll","thereof","lingspam","part10","part1","part2","part3","part4","part5","part6","part7","part8"
,"part9","public","these","they","theyd","they'll","theyre","they've","think","this","those","though","to","too"
,"two","under","unto","up","use","until","desctop","'ve","w","want","wants","was","wasn't","way","we","wed"
,"welcome","we'll","went","were","weren't","we've","what","whatever","what'll","whats","when","where"
,"wheres","which","while","whim","who","whod","whole","who'll","whom","whos","whose","why","why's"
,"with","without","yes","you","youd","you'll","your","youre","yours","yourself","yourselves","zero};
```

Tableau. IV.4. Liste des mots vides.

Si le mot apparu est un mot vide alors le système va le supprimer suivant l'algorithme de Tableau IV.6.

Entrée : document normalisé avec une liste des mots vides Sortie : document sans mots vides
Tant que (le document n'est pas fini) Faire Si (le mot appartient à la liste des mots vides) alors Supprimer le mot du document ; Retourner document ; Fin ; Fin.

Tableau IV.5. Algorithme d'élimination des mots vides

- **Transformation des majuscules en minuscules:**

En effet le mot “GIRL” et le mot “girl” vont être considérer différent alors qu’ils ont le même sens donc on transforme les majuscules en minuscule.

4.2.2. Niveau lexicale et morphologique

Ici, nous avons utilisé le procédé de *Racinisation* à l’aide des méthodes de la bibliothèque de lucene vue auparavant pour obtenir le « lexème » la forme élémentaire de chaque mot de corpus et requête, c'est en quelque sorte sa *racine linguistique*.

La racinisation est un procédé complexe et il est différent dans chaque langue, c'est pourquoi il n'existe pas d'algorithme informatique pour toutes les langues. De même, les algorithmes existants ne sont pas implémentés dans tous les langages informatiques. Ainsi, le langage informatique convenant le mieux à nos choix techniques (Java, .NET, PHP, *etc*.).

En effet, lucene se base sur l’algorithme de racinisation « Porter[8] » (ou «Porter stemmer») est un procédé pour éliminer les roturiers morphologiques et terminaisons flexionnelles de mots en anglais. Son utilisation principale est dans le cadre d'un processus de normalisation à long terme qui se fait habituellement lors de la mise en place de systèmes de recherche d'information.

4.2.3. Niveau sémantique

Notre traitement sémantique est réalisé par « Mapping des mots en sens » cette étape consiste a remplacé le mot par ces sens, ces sens représentent les synsets du l’ontologie WordNet, en effet chaque mot peut appartenir à plusieurs sens.
Exemple: le mot “Girl” devient :{ girl, miss, missy, young lady, young woman, fille, female child, little girl, daughter, girlfriend, lady friend}.

4.3. Indexeur :

Comme nous avons déjà expliqué dans CHAPITRE I, ici on utilise une approche permettant de sélectionner les mots normalisés et de leur associer *une pondération*, cette dernière permet d’assigner aux termes leur degrés d’importance dans les documents, il existe plusieurs approches pour le choix des index, nous avons utilisé l’approche suivante :

- **Approche basée sur la fréquence d'occurrences** : Cette approche consiste à choisir les mots représentants selon leur fréquence d'occurrence dans les documents.

Le système calcule le nombre de chaque mot ainsi que ses synsets de WordNet dans chaque document de corpus.

[8] http://tartarus.org/~martin/PorterStemmer/index.html [Date de dernière visite: 20/05/2016]

5. Mise en œuvre

5.1. Fenêtre principal

Notre application se compose d'une fenêtre principale à partir de laquelle l'utilisateur peut effectuer les traitements désirés selon le diagramme de cas d'utilisation décrit précédemment. La fenêtre principale est montrée dans la figure IV.2 :

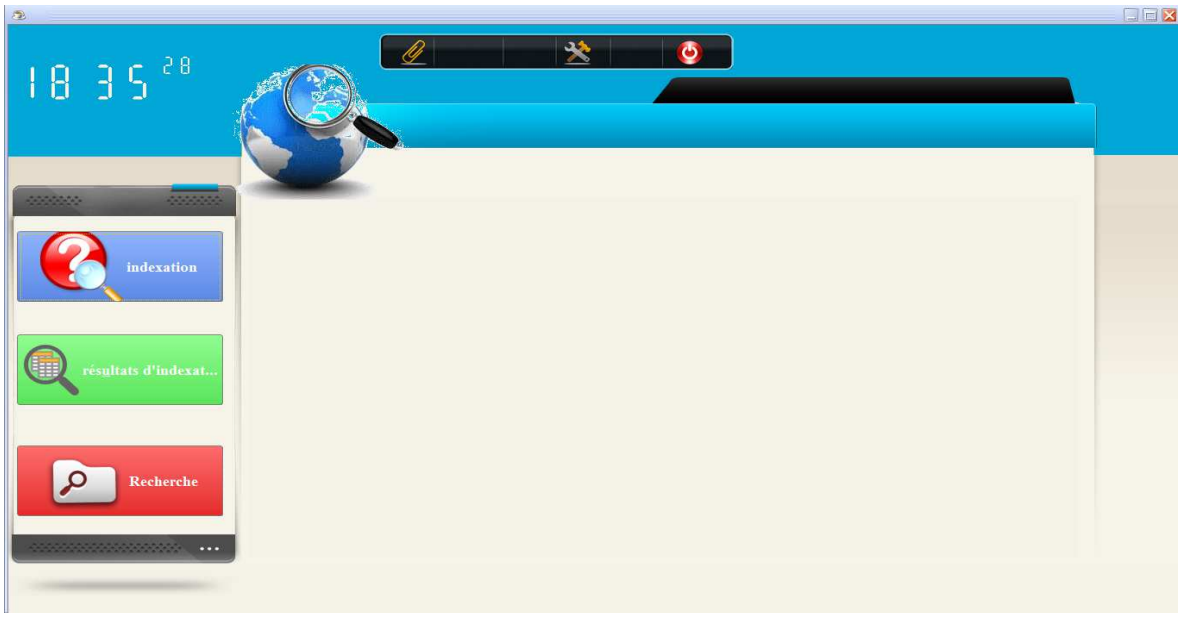

Figure IV.2. Fenêtre principal.

4.2.Chargement de corpus

La fenêtre principal montre 3 icônes supérieur : la première pour le téléchargement de corpus, la deuxième pour la saisie des premiers nombres de documents à rechercher et la troisième pour quitter l'application ; En cliquant sur la première une boite de dialogue s'ouvre, en sélectionne le corpus et le système va le charger comme la figure suivant montre.

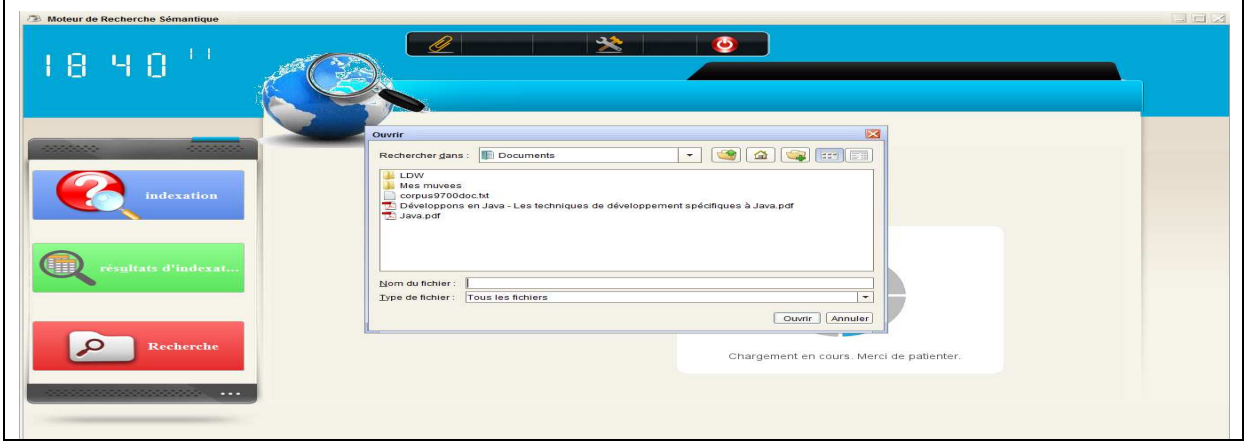

Figure IV.3.Téléchargement de corpus.

4.3.Indexation

Après le chargement de corpus, en passe à l'étape d'indexation en cliquant sur l'icône d'indexation, notre système va démarrer la réalisation des étapes d'indexation vu auparavant.

Figure IV.4. Indexation de corpus en cour.

4.4.Résultat d'indexation :

La deuxième icone à gauche nous permet de visualiser le résultat d'indexation, par le tableau au milieu de la Figure IV.5 en peut consulter le contenu de chaque document avant Figure IV.6 (la liste des documents à gauche) et après Figure IV.7 (la liste des documents à droite) l'indexation.

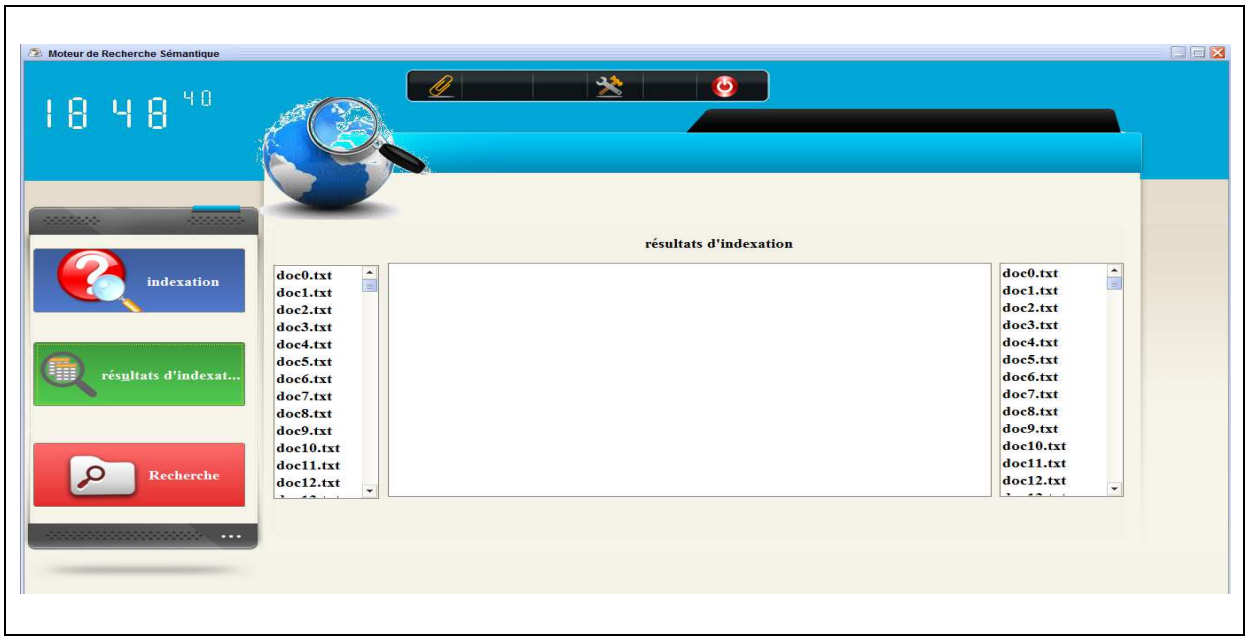

Figure IV.5.Résultat d'indexation.

Figure IV.5. Liste des documents avant indexation.

Figure IV.5. Liste des documents après indexation.

4.5. Recherche :

Le volet de recherche propose à l'utilisateur de saisir une requête avant d'effectuer une recherche selon les deux alternatives; la première pour une recherche sur le corpus tandis que la deuxième déploie la ressource *WordNet* pour inclure les synsets possibles des termes de la requête :

Recherche Sémantique : pour saisir le terme à rechercher
Rechercher : permet de lancer la recherche dans les documents indexés.
Résultats : Afficher la liste des documents pertinents avec la possibilité de les consulter.
Recherche WordNet : Afin d'offrir à l'utilisateur un moyen d'explorer le sens des termes et d'expliquer la requête, la ressource WordNet est intégré dans notre application.

- **Processus de recherche :**

Le processus de la recherche des documents permet non seulement de retournés les documents qui contient le terme de la requête mais aussi les synsets de cette dernière extrait à partir de WordNet. Exemple : en prenant la requête « estimate », la consultation des documents résultat montre les synsets de WordNet : (doc1.txt) contient le verbe du terme de la requête « to estimate » et le document suivant (doc73.txt) contient le synonyme du terme de la requête « idea » ainsi que (doc8060.txt) contient le synonyme de la requête « approximately » (Voir Figure IV.6 et Figure IV.7 et Figure IV.8) ; en cliquant sur le bouton (Recherche WorNet) en peut comprendre tous sens du terme « estimate » (Voir Figure IV.8).

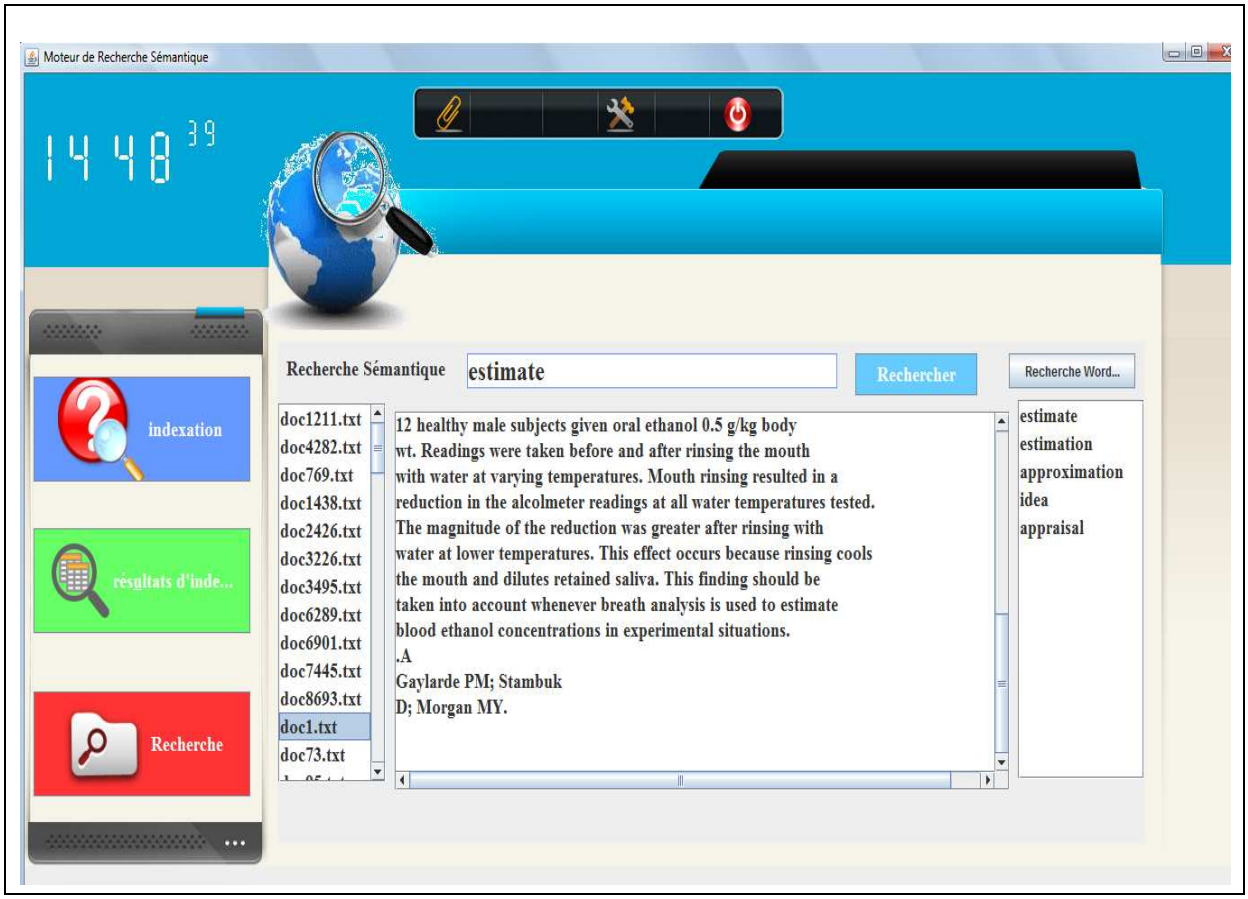

Figure IV.6. Afficher le contenu d'un document résultat sélectionné dans la liste de la requête « estimate ».

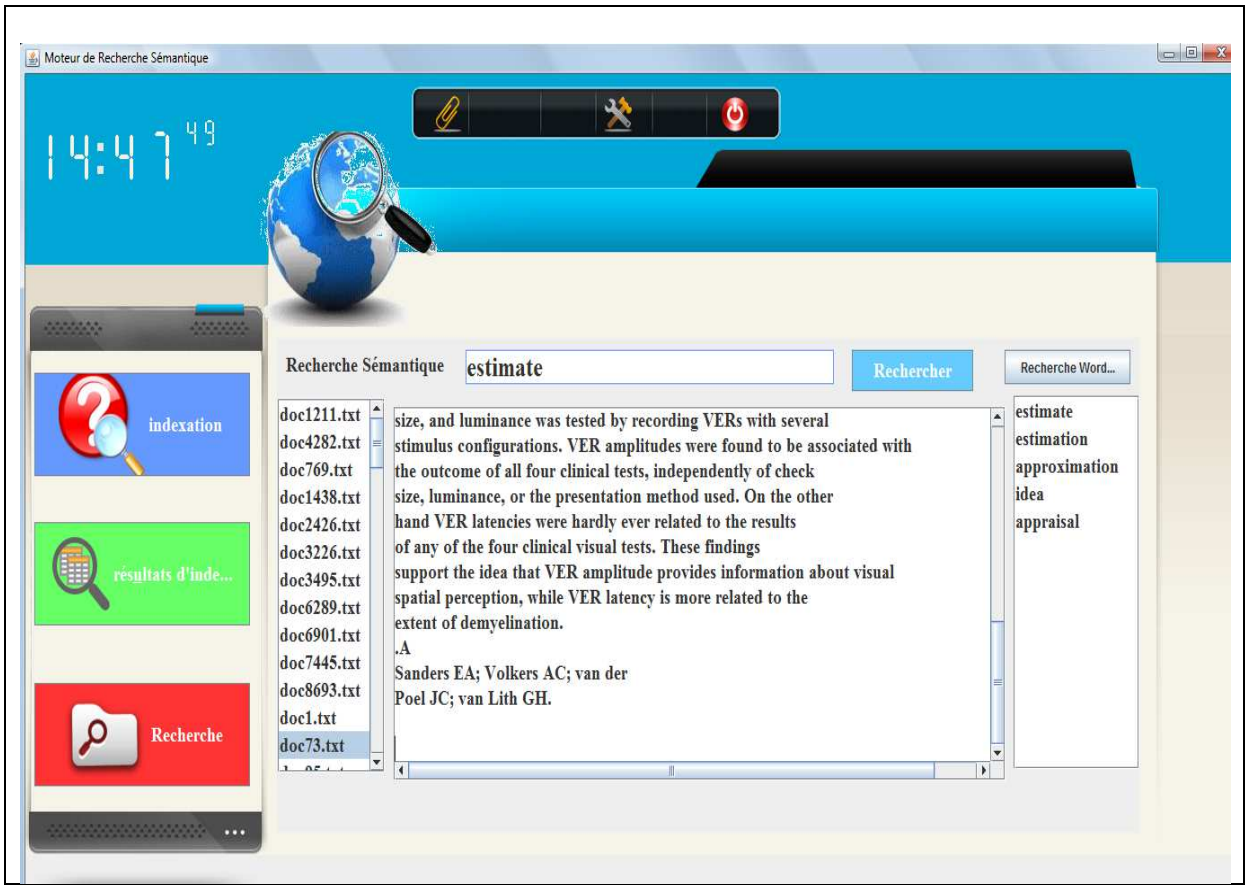

Figure IV.7. Afficher le contenu d'un document résultat sélectionné dans la liste qui contient le synonyme « idea »de la requête « estimate ».

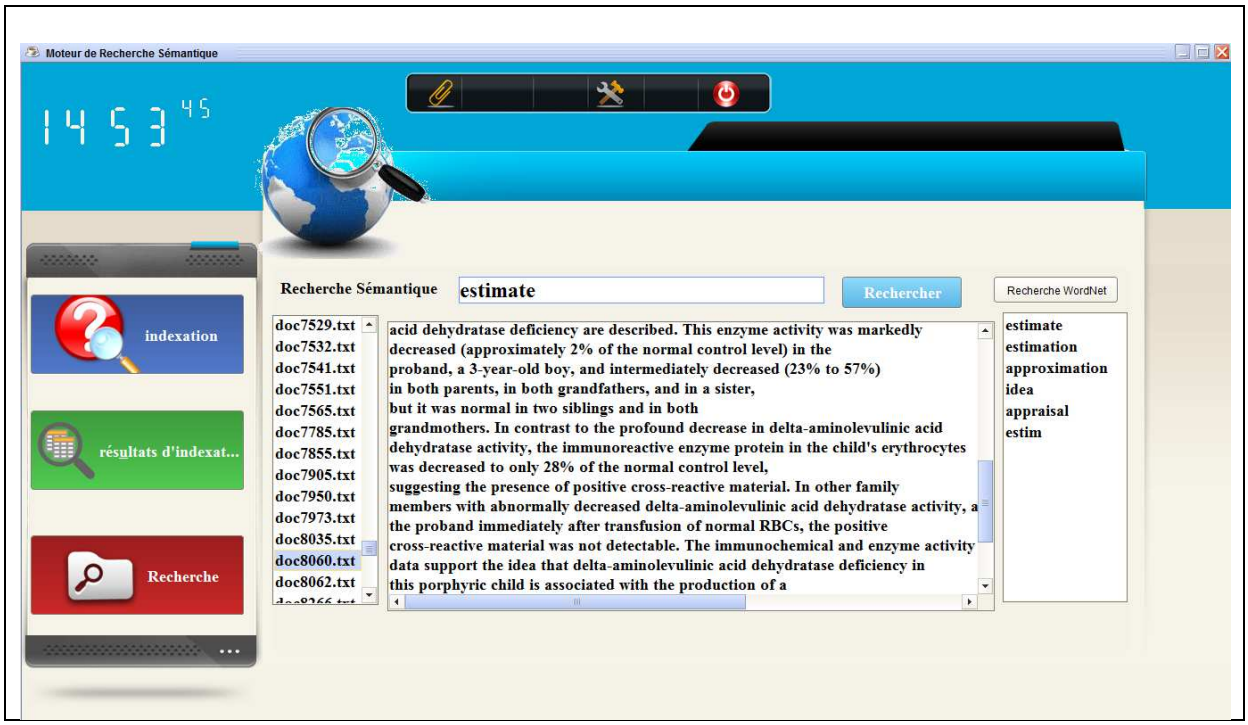

Figure IV.8. Afficher le contenu d'un document résultat sélectionné dans la liste qui contient le synonyme « approximately » de la requête « estimate ».

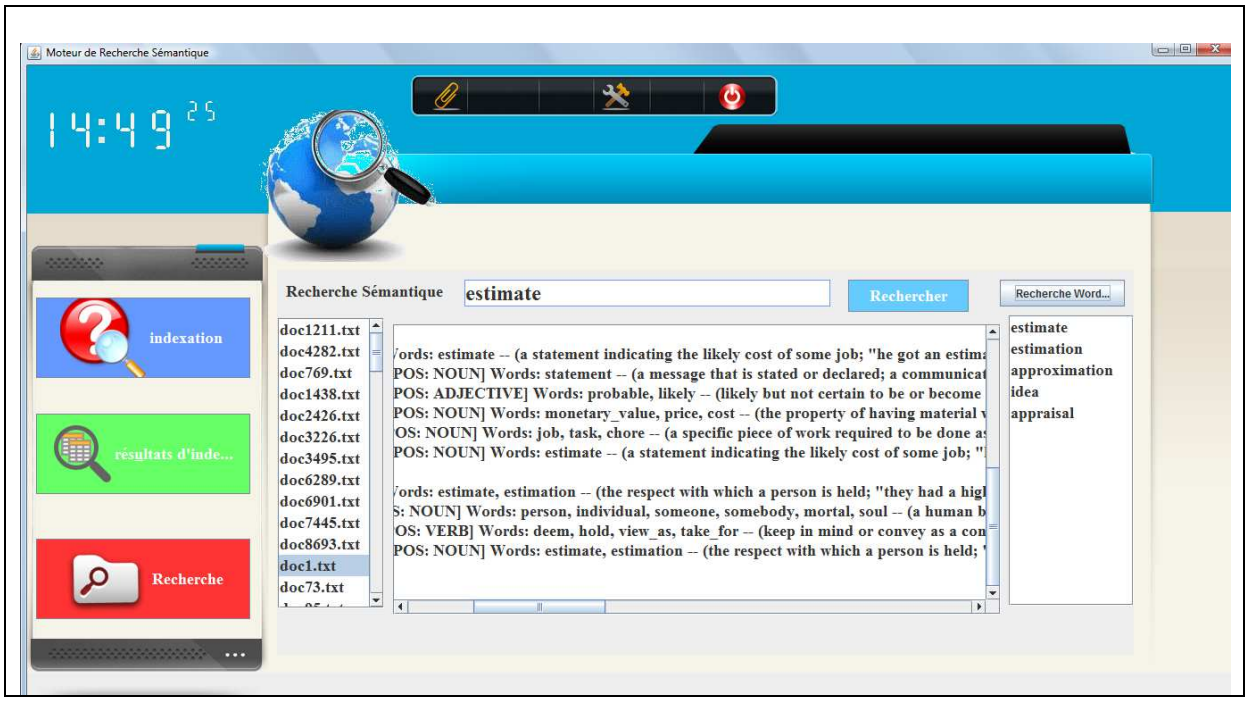

Figure IV.8. Explications sémantiques par WordNet du terme de la requête « estimate ».

Vu la durée d'indexation, de recherche et la quantité de documents retournés, l'utilisateur peut saisir le nombre des premiers documents à indexer ou à rechercher dans le corpus à travers la deuxième icône comme montre la Figure suivante.

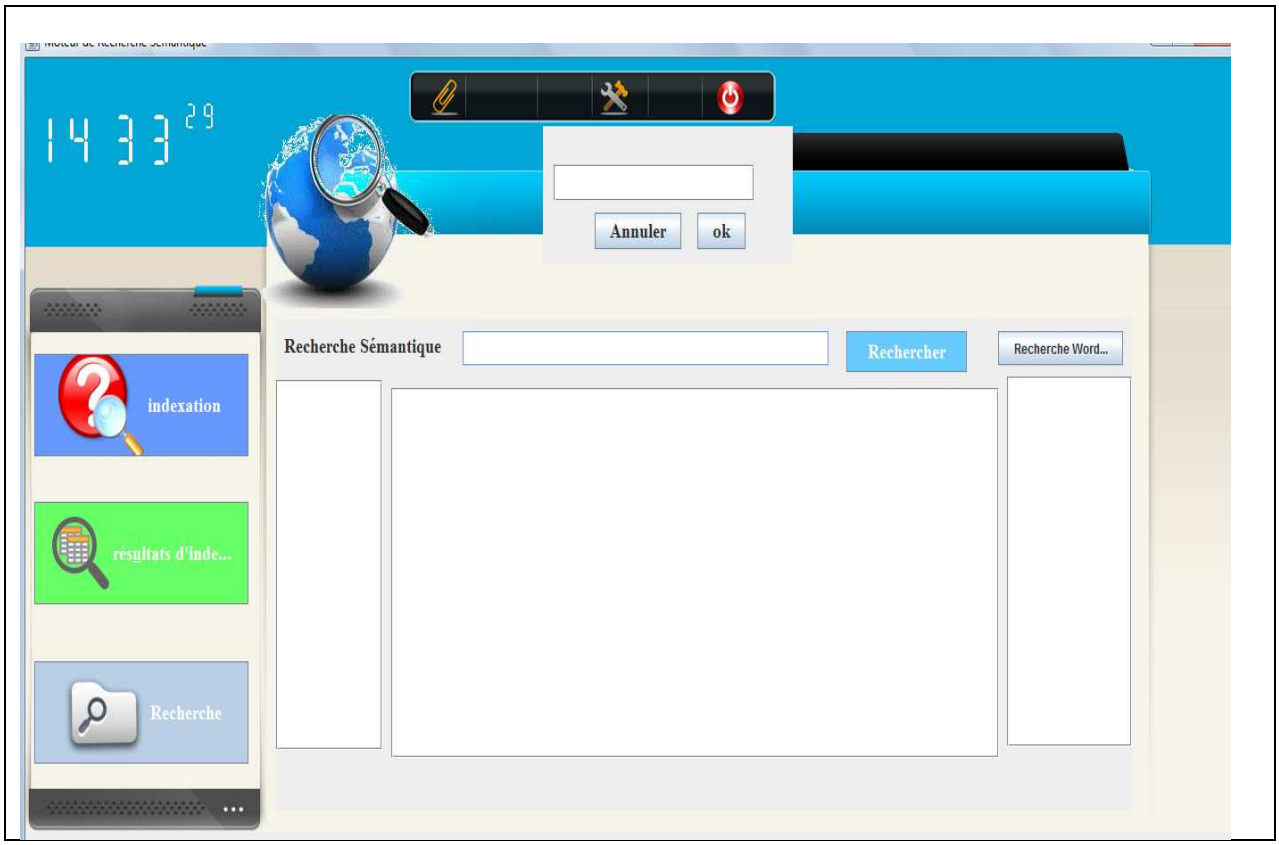

Figure IV.9. La saisie des premiers documents à indexer ou à rechercher.

Si le terme de la requête n'existe pas dans le corpus le système va afficher le message « Couldn't find ! »

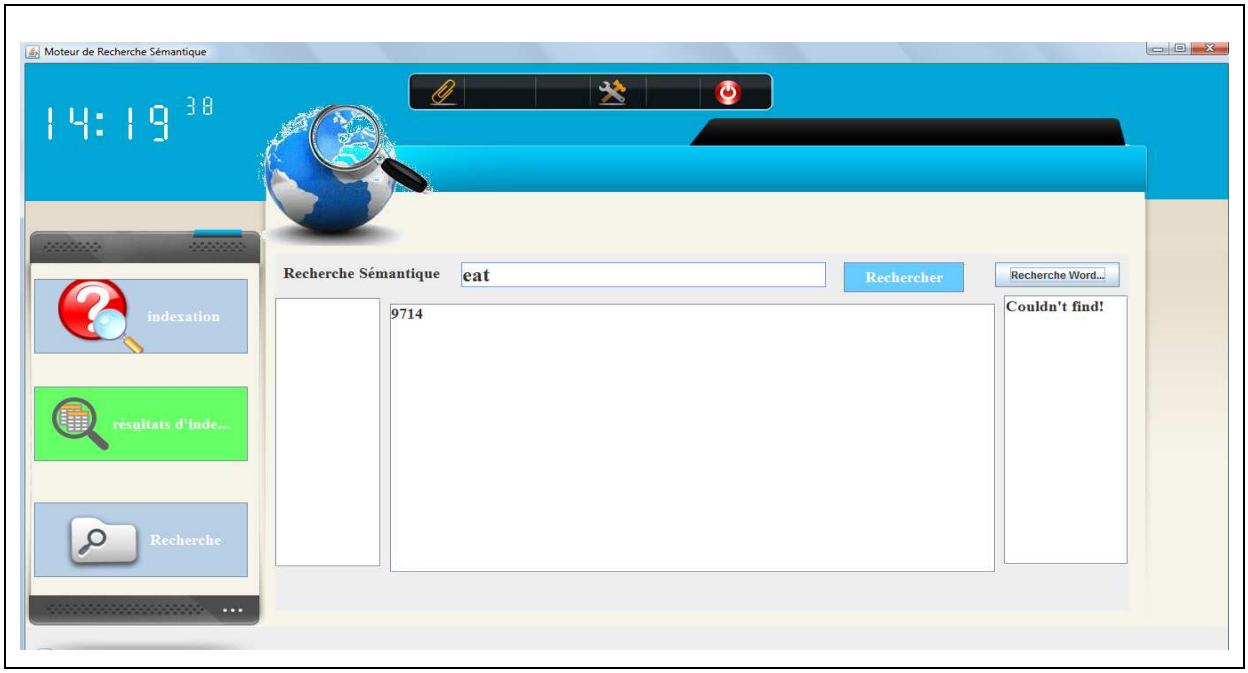

Figure IV.9. Le terme de la requête inexistant.

6. Conclusion

Ce chapitre a été consacré à la description conceptuelle de notre application. Différent outils logiciels et linguistiques ont été intégrés. Notre implémentation a pu mettre en œuvre plusieurs concepts et approches, qui paraissaient abstraits, dans une seule interface simplifiée. Les tests appliqués sur le corpus anglais sont encourageants et nous motivent à approfondir nos recherches dans ce domaine.

Conclusion et Perspectives

Conclusion

A travers les différents chapitres que nous avons présentés, nous concluons que la recherche d'information est un thème de recherche important en sciences de l'information. Elle peut porter sur plusieurs critères : le temps de réponse, la pertinence, la qualité et la présentation des résultats, etc. Le critère le plus important est celui qui mesure la capacité du système à satisfaire le besoin d'information d'un utilisateur, c'est-à-dire la pertinence des résultats.

Nos travaux développés dans ce mémoire s'inscrivent dans le cadre de l'indexation sémantique d'une collection de textes anglais.

Le problème de l'indexation basée sur l'approche statistique, en utilisant les termes simples et composés, est que le SRI ne prend en considération que les documents qu'ils partagent le maximum de mots-clés avec la requête. Cette méthode diminue la précision des SRI, il ne présente pas tous les documents pertinents de la collection.

C'est pourquoi, la représentation conceptuelle dans laquelle l'unité de vecteur serait un concept (groupe des synonymes appelé synsets), nous a permis de voir comment l'intégration d'une ressource externe WordNet a permis l'amélioration de la performance de notre SRI. Les éléments de cette représentation ne sont plus associés directement à de simples mots mais plutôt à des sysnsets.

Perspectives

Enfin, nous visons dans le cadre de nos travaux futures, d'enrichir notre système par plus de fonctionnalités pour la mission d'un moteur de recherche sémantique multilingue (Anglais, Arabe, Français et espagnol) qui prend en compte à la fois la sémantique et le contexte dans l'évaluation des SRI.

Il sera donc nécessaire de combiner plusieurs ressources à la fois ou d'intégrer « EuroWordNet » si ce dernier sera disponible. En tirant, pour chaque étape du processus d'évaluation, le principe de l'approche qui donne le meilleur résultat. L'évaluation sera par exemple faite en termes du temps de réponse et de la satisfaction des utilisateurs.

Somme toute, la combinaison des parties retenues de chaque langue donnera naissance à une nouvelle approche hybride qui utilise conjointement le contexte et la sémantique pour l'évaluation des SRI sur le web.

Bibliographies

[1] Salton.Gerard&McGill.Michael, "Introduction to modern information retrieval". McGraw Hill International Book Company, New York, 1983.

[2] Eric.Gaussier&Christian.Jacquemin&Pierre.Zweigenbaum, "Traitement automatique des langues et recherche d'information". Rapport de stage, Paris, 2003.

[3] Mustapha.Baziz, "Indexation conceptuelle guidée par ontologie pour la recherche d'information". Thèse de doctorat, Université Paul Sabatier de Toulouse, 2005.

[4] N.D.Y.Kompaoré, "Fusion de systèmes et analyse des caractéristiques linguistiques des requêtes: vers un processus de RI adaptatif". Thèse de doctorat, Université Paul Sabatier de Toulouse, 2008.

[5] P.Ingwersen, "Polyrepresentation of information needs and semantic entities: elements of a cognitive theory for information retrieval interaction". In proceedings of the ACM SIGIR international Seventeenth Annual Conference on research and development in information retrieval, pp. 101-110, 1994.

[6] J.Sinclair&R.Coulthard,"*Towards an Analysis of Discourse*".*The English used by Teachers and Pupils*,University Press, Oxford,1975.

[7] Souhila.Boucham, "Une approche basée Ontologies pour l'indexation automatique et la recherche d'information Multilingue". Mémoire de magister, Université M'hamed Bougara de Boumerdes, 2009.

[8] Nacira.Abbas, "Vers une Extension Sémantique de l'Analyse Formelle de Concepts : Application à la Recherche d'informations". Mémoire de magister, Université Mouloud Mammeri de Tizi-Ouzou, 03/07/2014.

[9] Alain.Berrendonner, "Grammaire pour un analyseur: aspects morphologiques". Document du travail du groupe de SYDO, Lyon, 1983.

[10] M.F.Porter, "An algorithm for suffix stripping". Program 14:130-137, 1980.

[11] P.Luhn, "The automatic creation of literature abstracts", pp. 159–165, April 1958.

[12] Karen.Spärck.Jones, "A statistical interpretation of term specificity and its application in retrieval". Journal of Documentation, Program 28:11–21, 1972.

[13] James.Callan&W.Croft&Stephen.Harding, "The inquery retrieval system". In Proceedings of the Third International Conference on Database and Expert Systems Applications, pp. 78–83, Springer-Verlag, 1992.

[14] C. Tambellini, "Un système de recherche d'information adapté aux données de incertaines: adaptation du modèle de langue". Thèse de doctorat, Université de Sophia Antipolis-UFR sciences, Nice, 2007.

[15] Abderrezak.Brahmi, "Contribution à la Recherche Intelligente sur le Web : Indexation Sémantique des Textes Non-Structurés". Thèse de doctorat, Oran, 2013.

[16] Singhal.A, "Modern Information Retrieval: A Brief Overview". Bulletin of the

IEEE Computer Society Technical Committee on Data Engineering, Vol. 24, pp. 35-43, 2001.

[17] Charhad, "Modèles de Documents Vidéo basés sur le Formalisme des Graphes Conceptuels pour l'Indexation et la Recherche par le Contenu Sémantique". Thèse de doctorat, pp. 24-25, Novembre 2005.

[18] Pascal.Hitzler&Markus.Krotzsch&Sebastien.Rudolph. "Foundations of semantic web technologies". CRC Press, pp. 8-11, 2009.

[19] Fatiha.Boubekeur-Amirouche, "Contribution à la définition de modèles de recherche d'information flexibles basés sur les CP-Nets". Thèse de doctorat, pp. 35-42, Université de Paul Sabatier de Toulouse, 2008.

[20] Fatiha.Boubekeur-Amirouche&Wassila.Azzoug, "Pondération des concepts en recherche d'information sémantique". Rapport de stage CORIA, pp. 441-450, 2013.

[21] E.Voorhees, "Using WordNet to Disambiguate Word Senses for Text Retrieval", Proceedings of the 16th Annual Conference on Research and Development in Information Retrieval, SIGIR'93, Pittsburgh, PA, 1993.

[22] R.Krovetz&W.B.Croft, "Lexical Ambiguity and Information Retrieval", in ACM Transactions on Information Systems, 10(1). 1992.

[23] M.Sanderson, "Word Sense Disambiguation and Information Retrieval", PhD Thesis, Technical Report (TR-1997-7) of the Department of Computing Science at the University of Glasgow, Glasgow G12 8QQ, UK, 1997.

[24] Boris.Katz&Özlem.Uzuner&Deniz.Yuret, "Word Sense Disambiguation for Information Retrieval", AAAI/IAAI 1999: 985. 1998.

[25] R.Mihalcea&D.Moldovan, "Semantic indexing using WordNet senses", In Proceedings of ACL Workshop on IR & NLP, Hong Kong, http://www.seas.smu.edu/~rada/papers/acl00.nlp_ir.ps.gz. October 2000.

[26] D.Yarowsky, "Hierarchical decision lists for word sense disambiguation. Journal Computers and the Humanities", Vol.34 (1-2), pp. 179-186, 2000.

[27] P.Pantel&D.Lin, "Discovering word senses from text". Proceedings of the 8th 619, ACMSIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 613, Canada 2002.

[28] R.Navigli, "Word Sense Disambiguation: A survey". ACM Computing Surveys, Vol. 41, No. 2, Article 10, 2009.

[29] Wassila.Azzoug, "Contribution à la définition d'une approche d'indexation sémantique de documents textuels". Mémoire de magister, 2012/2013.

[30] Bissan-Audeh, "Reformulation sémantique des requêtes pour la recherche d'information ad hoc sur le Web". Thèse de doctorat, 09 Septembre 2014.

[31] Reinout.Van-Rees, "Clarity in the usage of the terms ontology, taxonomy and classification". CIB REPORT, 2003.

[31] G.Miller&C.Leacock&R.Tengi&R.T.Bunnker, "A semantic concordance". In Proceedings of the ARPA Workshop on Human Language Technology, pp. 303-308, 1993.

[32] H.Kucera&W.N.Francis, "Computational Analysis of Present-Day American English". Brown University Press, 1967.

[37] G.A.Miller, "Nouns in WordNet : A lexical inheritance system", in : Five Papers onWordNet, http://www.cogsci.princeton.edu/ wn/ (sept. 1993), 10–25, revisedversion, 1997.

[38] Fatima.Ahmed-khaled&Rafik.Cherai, "Approche pour l'indexation conceptuelle basée sur les concepts et leurs concepts similaires pour la Recherche d'Information Arabe", Master, Médéa, 2015.

[39] C.Fellbaum&D.Gross&K.Miller, "Adjectives in WordNet", in : Five Papers on WordNet, http://www.cogsci.princeton.edu/ wn/, (sept. 1997), 26–39, revised version,1998.

[40] Hadjira.Hachemi&Nour-El-Houda.Rimouche, "Moteur de recherche sémantique", Master, Université Abou BakrBelkaid– Tlemcen, 2013.

[41] M.Silberztein, "Dictionnaires électroniques et analyse automatique de textes". Le système INTEX, Informatique linguistique, Masson, Paris, 1993.

[42] P.Srinivasan, "Thesaurus construction", in : Information Retrieval : Data Structures andAlgorithms, Frakes W. B., Baeza-Yates R., Prentice Hall, New Jersey, 1992.

[43] S.Johansson, "Some aspects of verb-adverbcombinations", in : The verb in contemporary English. Theory and description, Aarts B.,Meyer C. F., Cambridge University Press, Cambridge, 1995.

[44] J.S.Justeson&S.M.Katz, "Principled disambiguation : Discriminating adjective senseswith modified nouns", Computational Linguistics, 21, 1, 1995.

[45] R.Basili&M.Della-Rocca&M.T.Pazienza, "Contextual word sense tuning anddisambiguation", Applied Artificial Intelligence, 11, 1997.

[46] J.Gonzalo&F.Verdejo&I.Chugur&J.M.Cigarrán, "Indexing with WordNet synsets can improve Text Retrieval", CoRR, 1998.

[47] F.S.Touihr&I.Zitouni, "Conception et réalisation d'un site web dynamique pour la gestion des demandes d'assistances pour RTO (SONATRACH)", Licence, Université de Mostaganem, 2013-2014.

[49] Z.Graja,"Développement d'une application web pour la réservation de billet de train", Master, Université de Mostaganem, 2008-2009.

Cyber graphie

[33] http://www.axl.cefan.ulaval.ca/monde/anglais4.ModernE.htm, 12/03/2016.

[34] http://www.languageguide.org/english/grammar/fr/part1/nouns_r.jsp, 15/03/2016.

[35] http://www.languageguide.org/english/grammar/fr/part1/, 15/03/2016.

[36] https://fr.wikipedia.org/wiki/Grammaire_anglaise, 12/03/2016.

[48] http://www.clubic.com/telecharger-fiche384048-staruml.html, 25/04/2016.

[50] http://ipeti.forumpro.fr/t21-definition-de-langage-java-java-script, 18/05/2016.

[51] https://netbeans.org/index_fr.html, 18/05/2016.

[52] http://lucene.apache.org/,18/05/2016.

Introduction générale

Contexte et Problématique

La Recherche d'Information (RI) s'intéresse principalement à sélectionner à partir d'un ensemble de documents existants, ceux qui sont pertinents à une requête utilisateur. Afin d'y parvenir, l'une des tâches principales d'un Système de Recherche d'Information (SRI) est l'indexation. *L'indexation* consiste à construire des représentations simplifiées décrivant le contenu informationnel des documents et requêtes en vue de faciliter la recherche. Ces représentations sont ensuite interprétées par un *modèle de recherche* dans un formalisme unifié, puis comparées dans le but d'évaluer les degrés de pertinence des documents pour les requêtes.

Dans les SRI classiques, les documents et les requêtes sont représentés (ou indexés) par des mots-clés, manuellement ou automatiquement extraits à partir de leurs textes. Dans de tels systèmes, l'appariement (ou mise en correspondance) document-requête est lexical basé sur la présence ou l'absence des mots de la requête dans le document. Un document est alors considéré d'autant plus pertinent pour la requête qu'il a de mots clés en commun avec cette requête. Or, les mots de la langue sont par nature ambigus. Un même mot utilisé dans le document et la requête peut définir des sens différents (cas de polysémie et d'homonymie), et plusieurs mots lexicalement différents utilisés dans le document et la requête peuvent refléter un même sens (cas de synonymie). De ce fait, des documents pourtant non pertinents, contenant des mots de la requête, sont retrouvés, tandis que des documents sémantiquement pertinents, ne contenant aucun mot de la requête, ne sont pas retrouvés. Pour pallier les problèmes de l'indexation basée mots-clés, *l'indexation sémantique* est apparue, elle s'appuie sur la représentation des documents et requêtes par des sens des mots (ou concepts). Ces sens, sont extraits, à partir du contenu des documents et requêtes, par des méthodes de *désambiguïsation des sens des mots*(WSD) permettant de retrouver le sens adéquat d'un mot ambigu dans son contexte d'utilisation dans le document ou la requête.

L'indexation sémantique à l'issue de la recherche d'information, permet de retrouver des documents sémantiquement pertinents à une requête utilisateur, bien que ne partageant pas de mots en commun avec cette dernière. La qualité d'une recherche d'information sémantique dépend de la précision des techniques de WSD utilisées pour sélectionner les concepts représentatifs des documents et requêtes.

Dans ce cadre, nous proposons une nouvelle approche d'indexation sémantique basée sur les *synsets* de l'ontologie linguistique *WordNet* qui permet de capturer le sens voulu par le besoin d'information qui ne s'exprime pas par les termes de la requête, et cela permet de couvrir tous les documents de la collection et donc récupérer tous les documents pertinents existants. Cette approche permet d'augmenter le rappel et la précision du SRI.

Objectif et Organisation du Mémoire

L'objectif de notre projet présenté dans ce mémoire est la conception et la réalisation d'un moteur de recherche sémantique d'un corpus anglais à base de l'ontologie linguistique WordNet.

Ce mémoire s'articule en quatre chapitres principaux :

- Le premier chapitre représente l'état de l'art de ce mémoire, dont la première section intitulée « La recherche d'information » : nous présentons les notions de base de notre domaine d'application et les modèles sur lesquels repose la « Recherche d'Information ». Et la deuxième section intitulée « Indexation sémantique » : nous présentons dans cette section le besoin de l'indexation sémantique et les différents travaux et méthodes de désambiguïsation des sens des mots dans cette approche.
- Le deuxième chapitre traite la langue anglaise puisque en va travailler sur un corpus anglais (section I), dans la section suivante nous présentons les principes de l'ontologie linguistique WordNet qui couvre la grande majorité des noms, verbes, adjectifs et adverbes de la langue anglaise, dans un premier lieu nous commençons a donné une définition de WordNet, et nous exposons par la suite les relations sémantiques et quelques données statistiques par la suite nous présentons les limites du WordNet.
- Le chapitre suivant est consacrée à la modélisation de notre projet par les différents diagrammes du langage UML avec l'outil StarUML.
- Le dernier chapitre représente l'essentiel de notre travaille, commençant par l'introduction et l'environnement de développement, puis les étapes de conception de notre application, ensuite l'illustration des interfaces de l'application le tout finalisé par une conclusion.

Introduction

L'objectif du présent chapitre est de présenter un état de l'art sur deux domaines : la recherche d'information (Section I) et l'indexation sémantique (Section II).

Dans la première section, nous présentons les concepts de base de la RI. En particulier, nous décrivons les notions de document, de besoin d'information, de requête et de corpus; les processus de recherche d'information et d'indexation ; ainsi que, les modèles de RI. La dernière partie de cette section est discutée l'évaluation des systèmes de recherche d'information.

La deuxième section du chapitre est consacrée à la prise en compte de la sémantique dans les SRI, à cet effet, nous présentons un état de l'art sur l'indexation sémantique. En premier temps, nous présentons la problématique de l'indexation classique basée mots-clés. Le reste de la section est dédiée à la présentation des approches d'indexation sémantique basées sur la désambiguïsation des sens des mots.

I. SECTION 1 : La Recherche d'Information : État de l'Art

I.1. Définition

La recherche d'information (RI) n'est pas un domaine récent, il date des années 40. Une des premières définitions de la RI a été donnée par SALTON : « la recherche d'information est un domaine qui a pour objectif, la représentation, l'analyse, l'organisation, le stockage et l'accès à l'information » [1].

Plusieurs tâches se regroupent sous le vocable de la RI, la plus ancienne est la recherche documentaire, l'extraction d'information, la recherche d'information multilingue, les questions réponses, la recherche d'information sur le web, etc.

I.2. Bref Historique de la RI

- **1940:** Apparition des SRI, focalisation des RI sur les applications des bibliothèques.
- **1950:** Apparition du modèle booléen et l'élaboration de petites expérimentations sur des petites collections de documents.
- **1960 et 1970:** Apparition du système SMART et développement d'une méthodologie d'évaluation de système et conception de corpus de test (CACM).
- **1980:** Développement de l'intelligence artificielle (IA), ainsi l'intégration des techniques de l'IA en RI (système expert).
- **1990 et 1995:** L'apparition d'internet, la RI a été modifié et sa problématique plus élargie [2].

I.3. Concepts de base de la RI

Une synthèse des travaux de [3] et [4] nous a permis de dégager les concepts suivants :

I.3.1. Collection de documents

La collection de documents constitue l'ensemble des informations exploitables et accessibles. Elle constitue des représentations simplifiées mais suffisantes.

I.3.2. Document

Le document constitue l'information élémentaire d'une collection de documents. L'information élémentaire (granule de document), peut représenter tout ou une partie d'un document. Un document peut être un texte, une page web, une image, une bande vidéo, etc.

I.3.3. Besoin d'information

La notion de besoin d'information en RI est assimilée au besoin de l'utilisateur. Trois types de besoin utilisateur ont été définis par [5]:

- **Besoin vérificatif :** L'utilisateur cherche à vérifier le texte avec les données connues qu'il possède déjà. Il recherche donc une donnée particulière, et sait même comment y accéder. Ce besoin est dit stable : il ne change pas au cours de la recherche.
- **Besoin thématique connu :** L'utilisateur cherche à clarifier, à revoir ou à trouver de nouvelles informations dans un sujet connus. Un besoin de ce type peut être stable ou variable : il est possible que le besoin s'affine au cours de la recherche.
- **Besoin thématique inconnu :** Cette fois, l'utilisateur cherche de nouveaux concepts ou de nouvelles relations en dehors des sujets ou des domaines qui lui sont familiers. Le besoin est variable et est toujours exprimé de façon incomplète.

I.3.4 Requête

Une requête constitue l'*expression* du *besoin* en informations de l'utilisateur. Plusieurs systèmes utilisent des langages différents pour décrire la requête :

- En langage naturel : cas des systèmes SMART et SPIRIT.
- En langage booléen : cas du système DIALOG.
- En langage graphique : cas du système NEURODOC.

I.3.5 Corpus

Nous employons le mot *corpus* dans une acception assez restreinte empruntée à [6] : « Un corpus est une collection de données langagières qui sont sélectionnées et organisées selon des critères linguistiques explicites pour servir d'échantillon du langage. » Nous précisons cette optique au chapitre VI.

I.4. Processus d'un système de recherche d'information

Un système de recherche d'information manipule un corpus de documents qu'il transpose à l'aide d'une fonction d'indexation en un corpus indexé. Ce corpus lui permet de résoudre des requêtes traduites à partir de besoins utilisateur. Un tel système repose sur la définition d'un modèle de recherche d'information qui fait correspondre les documents aux requêtes.

La Figure I.1, présente les étapes de processus d'un système recherche d'information.

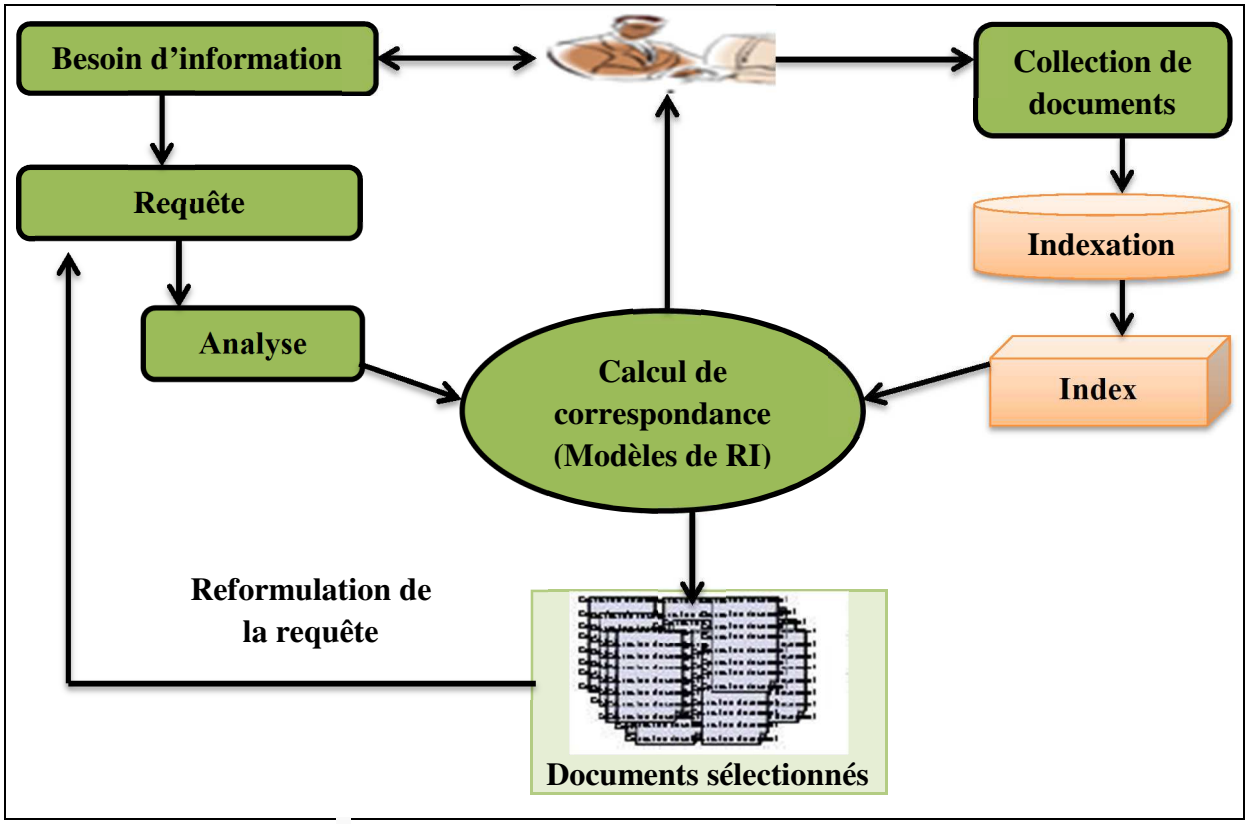

Figure I.1. Processus d'un système de recherche d'information.

I.4.1. L'indexation

Elle consiste à extraire des documents les mots les plus discriminants appelés *index* et les mettre dans un fichier appelé *fichier inverse*. Cette tâche est effectuée en marge du processus de recherche. **Les index** sont utilisés pour représenter le contenu des documents, ils ont un caractère réducteur car tous les termes d'un document ne sont pas importants à prendre en compte pour la recherche. Ils ont les caractéristiques suivantes :

- ils ne représentent qu'une partie du contenu des documents.
- ils peuvent prendre plusieurs formes (ex : mots simples, termes, syntagmes, entrées dans un thésaurus, etc.)[4].

Les fichiers inverses permettent d'associer des index aux documents qui les contiennent.

I.4.1.1. Les approches d'indexation [7]

A. Indexation manuelle (contrôlée)

C'est un spécialiste du domaine qui effectue l'analyse du document, pour identifier son contenu et construire une représentation de ce contenu.

B. Indexation semi-manuelle (contrôlée)

L'indexation semi manuelle se divise en 2 parties, une partie automatique permettant d'extraire une liste de descripteur (index), et une deuxième partie qui est manuelle réalisée par un spécialiste du domaine dont la tâche est de sélectionner des termes significatifs parmi les descripteurs retournés auparavant.

C. Indexation automatique (libre)

C'est le SRI qui génère les indexes des documents. L'indexation automatique a été créée afin de remédier aux problèmes liés aux approches précédentes, elle présente l'avantage d'une régularité du processus, car l'indexation automatique fournit toujours le même index pour le même document, ce qui constitue une qualité du système.

En ce qui nous concerne, c'est la troisième approche qui nous intéresse et nous pouvons la résumer comme suit.

I.4.1.2. Processus d'indexation automatique

Le processus d'indexation automatique (Figure I.2) passe par trois phases, chaque phase pouvant contenir une ou plusieurs étapes selon les usages des utilisateurs, c'est au programmeur de sélectionner les étapes qu'il souhaite intégrer au processus d'indexation.

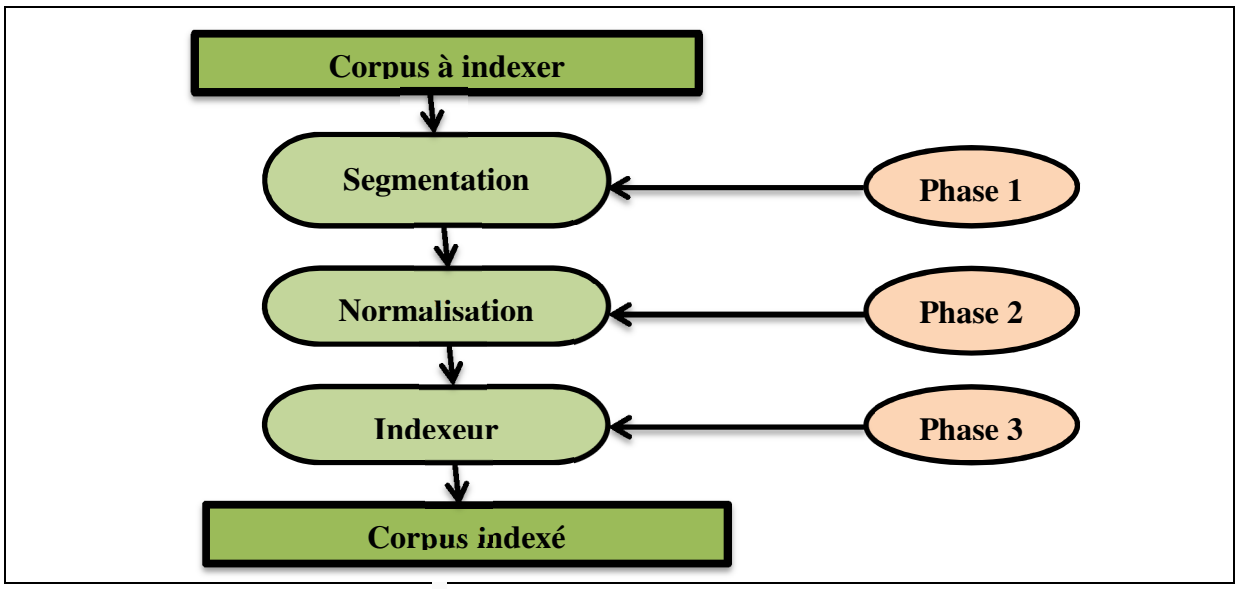

Figure I.2. Processus d'indexation automatique [8].

A. Phase de segmentation

Cette phase représente la fragmentation des documents en unités, elle est basée sur la ponctuation et sur une liste de séparateur, le résultat de cette phase est un ensemble de mots.

B. Phase de normalisation

Cette phase peut contenir plusieurs étapes, dans ce qui suit nous allons expliquer les étapes les plus importantes, elle traites plusieurs niveaux :

- **Niveaux lexical et morphologique :** Chaque mot de la langue lui correspond une catégorie morpho syntaxique :

•La lemmatisation : Le lemme s'obtient par une flexion. Les mots d'une langue peuvent être classés en 2 catégories: *Les lemmes: formes canoniques* (infinitif pour les verbes, singulier pour les noms, etc.) qui constituent les entrées dans un dictionnaire de cette langue.

Les mots obtenus par flexion de ces lemmes: conjugaison d'un verbe, changement de genre ou de nombre, etc. Par exemple, le mot « devrait» est obtenu par flexion (conditionnel présent, 3e personne du singulier) du verbe « devoir» [9] [8].

•La racinisation: Consiste à rechercher la forme tronquée d'un mot à partir de laquelle peuvent être reconstruites ses différentes variantes morphologiques [9]. Elle peut être réalisée simplement, en utilisant un algorithme comme l'algorithme de Porter [10] [8].

- **Niveaux syntaxique** : Basée sur l'utilisation de règles dépendantes de la langue :

•L'élimination des mots vides : Les mots vides sont des mots qui permettent de lier entre eux les mots d'une phrase pour la structurer (les articles, les conjonctions de coordination, les verbes auxiliaires, etc.). Ces mots ne portent pas de sens, ils ne peuvent pas constituer des index il faut donc les éliminer [9] [8].

•La discrimination : Par "discrimination", on réfère au fait qu'un terme distingue bien un document des autres documents, un terme qui a une valeur de discrimination élevée doit apparaître seulement pour un petit nombre de documents. L'idée est de garder les termes discriminants, et éliminer ceux qui ne le sont pas [9] [8].

•L'extraction des entités nommées : Les entités nommées sont des mots ou des groupes de mots qui désignent des personnes, des organisations, des dates, des lieux [8].

- **Niveaux sémantique :** Ici on s'intéresse à déduire les sens des mots, leurs concepts représentatifs et les relations sémantiques entre les mots. A cette étape, une ontologie peut être utilisée. La section II sera dédiée à cette approche (l'indexation sémantique) car c'est celle qui nous intéresse dans le cadre de notre travail.
- **Niveau pragmatique :** Il s'agit de l'analyse du langage naturel par la connaissance du monde réel. Ce niveau n'a pas été automatisé pour le moment.

C. Phase d'indexeur

Dans cette phase on utilise une approche permettant de sélectionner les index et de leur associer *une pondération*, cette dernière permet d'assigner aux termes leur degrés d'importance dans les documents, il existe plusieurs approches pour le choix des index :

- **Approche basée sur la fréquence d'occurrences** : Cette approche consiste à choisir les mots représentants selon leur fréquence d'occurrence. La façon la plus simple consiste à définir un seuil sur la fréquence: si la fréquence d'occurrence d'un mot dépasse ce seuil, alors il est considéré important pour le document [7].
- **Approche basée sur la valeur de discrimination :** Le calcul de la valeur de discrimination a été développé dans la loi de Luhn [11].
- **Approche basée sur tf.idf :** Les schémas de pondération pour l'attribution d'un poids à un mot, prennent en considération trois facteurs : le facteur de *pondération local* (*tf* - term frequency-), qui mesure l'importance du terme dans le document ; un facteur de *pondération globale* (*idf* - inverted document frequency-), mesurant la représentativité globale du terme dans la collection et un facteur de *normalisation* qui prend en compte la longueur du document. Une formule *tf.idf* combine l'importance du terme pour un document (*tf*), et le pouvoir de discrimination de ce terme (*idf*). Un terme qui a une valeur de *tf.idf* élevée doit être important dans ce document [12] [13].

I.4.2. Appariement Document/ Requête (Fonction de correspondance)

Tout système de recherche d'information s'appuie sur un modèle de recherche d'information. Ce modèle se base sur une fonction de correspondance qui met en relation les termes d'un document avec ceux d'une requête en établissant une relation d'égalité entre ces termes. La correspondance s'effectue au niveau de l'appariement document/ requête. L'expression de la fonction d'appariement dépend du modèle de RI choisi [14] [4].

I.4.3. Les Modèles de la RI [15]

Un modèle de RI a pour rôle de fournir une formalisation du processus de RI. Il doit accomplir plusieurs rôles dont le plus important est de fournir un cadre théorique pour la modélisation de la mesure de pertinence. Il existe trois grandes classes : les modèles booléens les modèles vectoriels et les modèles probabilistes, nous les définissons comme suit :

I.4.3.1. Modèles Booléens (Ensemblistes)

Ce sont les plus anciens de tous les modèles de RI. Ils sont basés sur la théorie des ensembles. Le document « d » et la requête « q » sont représentés comme une conjonction logique reliés par des opérateurs Booléens {ET (∧), OU (∨) et NON (¬)} de termes (ti), par exemple : $d = \{(t1 \wedge t2 \wedge \dots \wedge tn)\}$ et $q = \{(t1 \wedge t2) \vee (t3 \wedge \neg t4)\}$.

I.4.3.2. Modèles Vectoriels (Algébriques)

Dans ces modèles les documents et les requêtes sont représentés par des vecteurs de poids des termes. Chaque poids interprète l'importance du terme correspondant dans le texte.

Les vecteurs sont décrits dans un espace vectoriel définis par l'ensemble des termes préalablement établi lors de l'indexation.

I.4.3.3. Modèles Probabilistes

Ils se basent sur un modèle mathématique fondé sur la théorie de la probabilité. Le processus de recherche s'interprète par calcul du proche en proche de probabilité de pertinence d'un document relativement à une requête.

Nous orientons l'attention du lecteur vers [15] pour de plus détails à propos de ces modèles.

I.4.4. La reformulation de la requête [7]

La possibilité de reformuler la requête initiale s'avère intéressante dans le processus de la RI. Cela fera en sorte que le résultat retourné soit plus pertinent. Il existe 3 méthodes.

- **La reformulation manuelle :** Elle consiste à présenter à l'utilisateur une liste de documents jugés pertinents en réponse à la requête initiale. C'est à l'utilisateur de sélectionner à partir des documents pertinents ceux dont lesquels le système va extraire les termes à rajouter à la requête initiale pour une nouvelle recherche.
- **La reformulation semi-automatique :** Elle nécessite l'intervention de l'utilisateur qui doit identifier les documents pertinents et les documents non pertinents.

- **La reformulation automatique :** L'extension de la requête est faite sans intervention de l'utilisateur grâce à l'utilisation d'un thésaurus contenant des informations linguistique (équivalence, association, hiérarchie).

I.5. Évaluation d'un système de recherche d'information

Depuis l'apparition des premiers modèles pour la RI, l'évaluation objective de leur efficacité représentait une pièce-maitresse dans le développement du domaine. Il était évident pour les chercheurs la nécessité de trouver des mesures standards pour estimer la qualité des résultats de recherche, la communauté de la RI a approuvé deux mesures de base qui sont la précision et le rappel [16] [15].

Pour l'évaluation de la précision et le rappel, la cardinalité des différents ensembles de documents est utilisée, Comme décrit dans le Tableau I.1.

	Documents pertinents	**Documents non-pertinents**
Documents sélectionnés	Documents trouvés	Documents trouvés Documents hors contexte : **bruit**
Document non-sélectionnés	Documents oubliés : **silence**	Documents non trouvés non pertinents

Tableau I.1. Les quatre ensembles de documents résultats en RI [17].

- **La précision :** Évalue la portion de *documents sélectionnés* par le SRI, et qui sont *pertinents* par rapport au besoin de l'utilisateur [17].
- **Le rappel :** Évalue la portion de *documents pertinents* qui sont sélectionnés, par rapport au besoin de l'utilisateur [17].

II. SECTION 2 : L'Indexation Sémantique : État de l'Art

II.1. La notion sémantique et les démarches d'indexation sémantique

La sémantique est un mot d'origine grecque qui signifie l'étude du sens. C'est une notion plutôt philosophique, qui a été employée à propos des systèmes d'information pour mieux rapprocher l'interprétation des choses par des machines avec celle des humains. L'idée de la sémantique est de construire des modèles qui permettent de comprendre, structurer et prédire certaines parties du monde [18]. Au fil du temps, cette idée a été mieux formalisée, et a commencé à s'intégrer dans différents domaines. En conséquence il existe deux démarches de l'indexation sémantique, la RI et le web sémantique[1].

[1] http://w3cwebsemantique.orgfree.com/upload/5.pdf [Date de dernière visite: Mai 2016]

II.2. la démarche d'indexation sémantique issue de la RI

II.2.1. De la RI classique à la RI sémantique

II.2.1.1. Problématique (Besoin de l'indexation sémantique)

Les modèles classiques de la RI, se basent sur l'hypothèse qu'il y a une correspondance stricte entre les mots et les sens, alors qu'un mot peut représenter plusieurs sens et un sens peut être représenté par plusieurs mots. En partant de cette hypothèse, la recherche d'information classique se trouve face à deux problèmes, *l'ambiguïté des mots* et leur *disparité*.

- *L'ambiguïté des mots*, dite ambiguïté lexicale, se rapporte à des mots lexicalement identiques et portant des sens différents. On parle ici du *bruit*.
- *La disparité des mots* se réfère à des mots lexicalement différents mais portant un même sens. Ceci implique que des documents, pourtant pertinents, ne partagent pas de mots avec la requête, ne sont pas retrouvés. On parle ici du *silence* [19] [20].

II.2.1.2. Solution

La solution globale permettant de répondre à ces deux problèmes consiste en l'indexation sémantique. L'indexation sémantique tente d'apporter des solutions au niveau de la représentation des documents et des requêtes, l'indexation sémantique ou conceptuelle est sensée d'améliorer les performances du SRI. D'où on distingue deux grandes approches : *l'indexation sémantique* et *l'indexation conceptuelle*.

L'indexation conceptuelle : Peut être vue comme une généralisation de l'indexation sémantique, dans la mesure où les concepts véhiculent des sens [3].

II.2.2. L'indexation sémantique

L'indexation sémantique s'intéresse principalement à la représentation des documents et requêtes par les sens des mots qu'ils contiennent plutôt que par les mots eux-mêmes. L'objectif sous-jacent est d'améliorer la représentation des entités indexées et de pallier aux problèmes de l'indexation classique basée mots [7].

II.2.3. Méthodes d'indexation sémantique en RI

Nous détaillerons dans ce qui suit les travaux les plus représentatifs de l'utilisation du sens des mots dans la RI à travers les travaux de Voorhees, Krovetz & Croft, Sanderson, Mihalicea & Moldovan et Katz & Uzuner & Yuret.

A. La méthode de Voorhees

Voorhees [21] a construit un outil de désambiguïsation basé sur WordNet. Pour désambiguïser une occurrence d'un mot ambigu, les synsets (sens) de ce mot sont classés en se basant sur la valeur de cooccurrence calculée entre le contexte de ce mot et un voisinage contenant les mots du synset dans la hiérarchie de WordNet. Voorhees a expérimenté cette

approche sur une collection de test désambiguïsée (les requêtes de la collection de test sont aussi désambiguïsées manuellement) par rapport aux performances du même processus sur la même collection dans son état d'origine (ambigu).

Les résultats de ses expérimentations ont montré que pour chacune de ces collections, les performances du système de RI diminuent sensiblement dans le cas de l'utilisation des collections désambiguïsées.

B. La méthode de Krovetz & Croft

Krovetz et Croft [22] ont conduit une vaste étude sur certaines hypothèses ayant trait à la pertinence de la relation de correspondance du sens des mots dans la requête et les documents. En utilisant les collections de test CACM et Time, ils ont examiné les dix (10) premiers documents restitués pour chaque requête (pour les deux collections considérées).
Ils ont analysé la correspondance de sens entre chaque terme de la requête et ses occurrences dans chaque document restitué. Krovetz et Croft ont examiné l'amélioration de l'efficacité de la recherche en supprimant les documents sélectionnés avec des sens erronés. Sur la collection Time, une amélioration de 4% est constatée au niveau de la précision moyenne, mais sur la collection CACM, l'augmentation est de 33%. Ils concluent en suggérant des situations où la désambiguïsation peut s'avérer intéressante pour améliorer les performances des SRI.

C. La méthode de Sanderson

Les analyses de Sanderson [23] reprennent les travaux de Krovetz et Croft et détaillent l'impact des erreurs de désambiguïsation dans l'efficacité des SRI. Sanderson a utilisé une forme d'ambiguïté artificielle qu'il désigne par pseudo-mot (pseudoword).

Une concaténation de plusieurs mots choisis aléatoirement dans un corpus forme un pseudo-mot. Ces mots deviennent les pseudo-sens du pseudo-mot unique qu'ils forment, et toutes leurs occurrences dans ce corpus sont remplacées par ce pseudo-mot. En ajoutant des pseudo-mots dans un document de la collection de test, une quantité mesurable d'ambiguïté additionnelle est introduite et son impact sur l'efficacité de la recherche peut être déterminé.

Les résultats ont montré que l'ambiguïté introduite ne réduit pas les performances du système et que la désambiguïsation est utile dans le cas des requêtes courtes et avec un taux de performance (de l'outil de désambiguïsation) élevé (>90%).

D. La méthode de Katz & Uzuner & Yuret

La méthode de Katz & Uzuner & Yuret [24] est basée sur la notion de contexte pour désambiguïser les mots dans le texte. Le sens d'un mot est identifié à partir de son contexte local. Ils partent de l'hypothèse que les mots utilisés dans un même contexte local, appelés sélecteurs, ont souvent des sens proches. Les sélecteurs sont utilisés pour identifier le bon synset de WordNet (les synonymes d'un seul sens) correspondant à un mot dans son contexte.

L'algorithme de désambiguïsation de Katz est testé sur le corpus SemCor où chaque mot est étiqueté avec sa catégorie syntaxique (POS : nom, verbe, adjectif, adverbe) ainsi que le numéro de sens correspondant dans WordNet. Dans ce corpus, la précision pour le désambiguïseur est de 60% (les termes ayant un seul sens ne sont pas compris).

Katz et ses collègues ont intégré leur algorithme de désambiguïsation au processus de RI. Ils ont utilisé le système SMART. Leur conclusion est que leur algorithme n'améliore pas les performances du système de RI.

E. La méthode de Mihalcea et Moldovan

Mihalcea et Moldovan [25], ont observé une amélioration de 16% dans le rappel et de 4% dans la précision quand ils ont utilisé une combinaison de l'indexation basée sur les mots clés et de l'indexation basée sur les synsets de WordNet.

F. La méthode de katz & Uzuner & Yuret

Katz & Uzuner & Yuret, tout comme Voorhees et Sanderson ainsi Mihalicea & Moldovan, pensent qu'une désambiguïsation plus performante peut aider à améliorer les performances des systèmes de RI.

Pour retrouver les sens corrects des mots, l'indexation sémantique requiert des techniques de désambiguïsation des sens des mots (*WSD -Word Sense Disambiguation-*)**.**

Nous décrivons, dans ce qui suit, Les approches d'indexation qui sont basées sur les techniques de désambiguïsation des sens des mots, les ressources linguistiques externes (structurées et non structurées) les plus exploitées par WSD seront aussi présentées.

II.3. Les approches de désambiguïsation des sens des mots (WSD)

De nombreuses approches de désambiguïsation sémantique des mots existent. Ils peuvent être divisées en : *approches basées sur les corpus d'apprentissage (endogène)* et *approches basées sur les ressources linguistiques externes (exogène).*

II.3.1.Approche endogène

Ces approches se basent sur l'utilisation d'un corpus d'apprentissage composé d'un grand nombre de contextes de mots polysémiques, dans le but d'apprendre les connaissances utiles sur le sens d'usage des mots. Cette phase d'identification automatique des connaissances est appelée apprentissage. A l'issue de cette phase, l'algorithme de désambiguïsation est capable d'assigner le sens adéquat aux mots apparaissant dans une nouvelle phrase en s'appuyant sur les connaissances acquises durant la phase d'apprentissage. Les approches de désambiguïsation basées sur les corpus d'apprentissage se distinguent en approches supervisées et approches non supervisées [26] [27].

II.3.2. Approche exogène

La plupart des approches d'indexation sémantique basées sur la désambiguïsation exogène, s'appuient en général sur *des ontologies* pour déterminer les différents sens du mot et pour désambiguïser les sens des mots. Le principe de base de l'indexation consiste alors à extraire dans un premier temps, l'ensemble des termes descripteurs (index) du document. Il s'agit ici d'une indexation classique. Ces termes sont ensuite désambiguïsés. Pour ce faire, les sens de chaque terme d'indexation sont d'abord retrouvés à partir de *la ressource externe.*

Puis, des scores sont associés aux différents sens ainsi retrouvés. Le sens qui maximise le score est alors retenu comme le sens adéquat du terme d'indexation correspondant [28].

II.4. Les ressources linguistiques structurées

Elles jouent un rôle très important dans le domaine de RI conceptuelle, elles sont utilisées pour extraire les concepts à partir des documents et requêtes. Elles offrent une meilleure représentation des documents car elles permettent de définir les relations entre les concepts des documents. Différents types de ressources sémantiques peuvent être distingués parmi lesquels se trouvent les dictionnaires informatisés, les taxonomies, les thésaurus, et les ontologies.

II.4.1. Les dictionnaires informatisés (MRD)

Représentaient des sources de connaissances très populaires dans les années 80 pour les différentes disciplines du domaine du traitement automatique de la langue. Dans un dictionnaire informatisé, un mot de la langue possède un ou plusieurs sens qui sont définis par leurs glossaires (*gloss*). Le glossaire d'un sens décrit le sens du mot par une définition, des commentaires et/ou des exemples d'utilisation courante. Comme exemples de dictionnaires informatisés, on peut citer : le *Collins English Dictionary*, le *Oxford Dictionary of English* et le *Longman Dictionary of Contemporary English* [29].

II.4.2. Une taxonomie

C'est une structure qui permet de contrôler le vocabulaire par un seul type de relation donnant la possibilité de généraliser ou de préciser un sens. Elle se présente sous la forme d'une hiérarchie simple de terme [30]**.**

II.4.3. Les thésaurus

Selon la norme internationale ISO 15143-1 : 2010[2], les thésaurus sont : «vocabulaire contrôlé ordonné dans une disposition donnée dans lequel les relations entre les termes sont affichées et identifiées». Ces termes dénotent les concepts d'un domaine particulier.

Dans un thésaurus, les termes sont organisés dans une hiérarchie de concepts liés par des relations sémantiques. Les relations présentes dans un thésaurus sont des relations taxonomiques (spécialisation /généralisation), d'équivalence (synonymie), d'association (proximité sémantique, proche-de, relié-à). Les termes d'un thésaurus peuvent servir à indexer des documents comme c'est le cas dans les thésaurus médicaux MeSh[3] et UMLS[4].

II.4.4. Une ontologie

Une ontologie est une collection de concepts bien définis qui décrivent un domaine spécifique [31]. Les relations entre ces concepts peuvent être différentes d'une ontologie à une autre. Par exemple, les relations existantes dans une ontologie du domaine juridique n'ont pas les mêmes significations que celles d'une ontologie de la génétique.

[2] http://www.iso.org/iso/catalogue_detail.htm?csnumber=37406 [date de dernière visite : Février 2016]
[3] http://www.nlm.nih.gov/mesh/ [date de dernière visite: Février 2016]
[4] http://www.nlm.nih.gov/research/umls/ [date de dernière visite: Février 2016]

Avec le niveau élevé de modélisation fourni par les ontologies, des langages de représentation ont été proposés pour simplifier la manipulation des ressources. Les langages les plus connus sont issus du W3C, comme RDF, RDF Schema, OWL et SPARQL**.**

En ce qui nous concerne pour la désambiguïsation des sens des mots de notre travail c'est l'ontologie et nous pouvons la détailler comme suit :

Les ontologies sont connues comme des outils capables de manipuler les connaissances derrière les concepts, en peut dire aussi qu'une ontologie est un ensemble structuré de concepts organisés dans un graphe (ou réseau sémantique). Elles peuvent être utilisées à différents niveaux de SRI. Les objectives de notre étude est de voir les effets d'une ontologie linguistique anglaise *WordNet* dans la recherche des documents et la désambiguïsation des sens des requêtes.

II.5. Les ressources linguistiques non structurées

II.5.1. Les corpus d'apprentissage

Ce sont de longs textes utilisés dans les techniques d'apprentissage pour construire la connaissance nécessaire pour la WSD. Ces corpus peuvent être étiquetés manuellement avec les sens des mots. A titre d'exemple, le corpus *SemCor* [31] est la version étiquetée du corpus *Brown* [32] avec des sens issus de WordNet.

II.5.2. Les corpus de collocations

Ce sont des ensembles de collocations de mots qui ont une tendance de se produire ensemble régulièrement. Parmi ces ressources, nous citons : *The British National Corpus collocations* et le *Collins Cobuild CorpusConcordance* [8].

Conclusion

Nous avons consacré ce chapitre à l'état de l'art sur la recherche d'information et l'indexation sémantique, à travers ses différentes sections nous concluons que la recherche d'information, s'attache à définir des modèles et des systèmes afin de faciliter l'accès à un ensemble de documents se trouvant dans des bases documentaires. Le but est de permettre aux utilisateurs de retrouver les documents dont le contenu répond à leur besoin en information, il s'agit donc de retourner l'ensemble de documents pertinents.

Puis, nous avons passé en revue l'approche d'indexation sémantique proposée en RI. Cette approche a apporté la preuve que la représentation des documents et requêtes par les sens (ou concepts) de leurs mots est bénéfique dans un processus de RI, permet ainsi de résoudre les problèmes causés par les SRI classiques. Ces sens sont le plus souvent identifiés par des approches de désambiguïsation qui utilise des ressources externes, comme support à la modélisation des phases d'indexation et de recherche.

Introduction :

Un moteur de recherche sémantique peut être vu comme un outil qui répond à des requêtes (formulées avec les concepts d'une ontologie linguistique).

L'introduction d'une ontologie linguistique dans le processus d'indexation et de recherche requiert une analyse adéquate et spécifique pour chaque langue. Dans le cas de l'anglais, la tâche s'avère plus délicate vu la disponibilité des ressources lexicale sémantique comme WordNet : la base de connaissances générales la plus utilisée, elle a servi à mettre au point ou à tester de nombreuses expériences depuis le début des années 1990. Par ailleurs, WordNet est un exemple d'une ontologie lexicale conçue et pensée pour le support électronique.

Dans ce chapitre, nous décrivons la langue anglaise et ses caractéristiques linguistiques (Section I). Les fonctionnalités de WordNet seront aussi discutées avant de passer au chapitre suivant (Section II).

I. SECTION 1 : La langue anglaise.

I.1. introduction

Vu la disponibilité des corpus bien traités et qualifiés en anglais et l'existence des ressources sémantique pour le traitement de la langue anglaise comme WordNet, nous avons choisi de travailler sur un corpus en anglais et utiliser WordNet comme ressource linguistique sémantique pour la désambiguïsation des sens des mots.

I.2. Histoire de la langue

L'anglais fait partie de la famille des langues germaniques occidentales et est lié au néerlandais, à l'allemand et au luxembourgeois. L'anglais est né de la fusion de plusieurs dialectes rapportés en Grande-Bretagne par les colons germaniques appelés les Angles. Il a également été influencé par le vieux norrois et normand français pendant les invasions vikings et la conquête normande. Suite à l'influence de l'empire britannique entre le 17e et le milieu du 20e siècle, l'anglais s'est considérablement propagé à travers le monde. Encore aujourd'hui, à travers les chaînes culturelles américaines (par exemple la musique, le cinéma et la télévision) l'anglais est une lingua franca de choix dans de nombreux contextes [33].

I.3. Géographie de la langue

765 millions de personnes parlent l'anglais à travers le monde, dont 360 millions comme première langue, les 430 millions restants l'utilisant en tant que deuxième langue. À cela s'ajoute le nombre de personnes parlant l'anglais comme langue étrangère, estimé à 750 millions, soit supérieur à ceux qui l'utilisent comme première langue. On considère que 1 personne sur 4 dans le monde parle anglais, selon plusieurs niveaux de compétence. L'anglais est la langue officielle de l'Australie, du Canada, de l'Irlande, de la Nouvelle-Zélande, du Royaume-Uni et des États-Unis, ainsi que de plus de 50 autres pays à travers le monde [33].

I.4. Diffusion dans les sciences et les techniques

L'emploi de mots anglais est notable dans des secteurs comme l'informatique, les télécommunications comme le fut (et l'est toujours, d'ailleurs). Mais les nouvelles technologies (DVD multi-langues, mondialisation de l'internet) et l'adaptation des entreprises à leurs clients (CNN diffusant en plusieurs langues, Microsoft fabriquant le logiciel Windows en plusieurs langues) ont porté un coup relatif à cette domination de l'anglais. L'anglais est depuis 1951 la langue utilisée dans l'aviation, sur décision de l'OACI. De plus en plus de travaux de recherches scientifiques (thèses, études, etc.) sont rédigés en anglais ou font l'objet d'une traduction dans cette langue [33].

I.5. Les propriétés morphologiques de la langue [34] [35][36] :

- L'anglais est basé sur l'alphabet latin qui comprend vingt-six lettres et se lit de gauche à droite. Le nombre de mots existants dans la langue anglaise est estimé à plus d'un million.
- Il existe pratiquement 100 000 familles de mots dans la langue anglaise.
- Une famille de mots est un groupage de mots dérivés de la même base. Par exemple : *active*, *actively*, et *activities* « actif, activement, activités et activité » sont tous de la même famille de mots.
- L'anglais est une langue respectant l'ordre SVO (sujet, verbe, objet) dans la phrase déclarative. Exemples : *Tom does his homework* : « Tom » (sujet) « fait » (verbe) « ses devoirs » (objet).
- En général, l'élément principal se trouve au début de la phrase. Exemple : *To run quickly* « courir vite » (phrase verbale) (mais on trouve également *to quickly run*, ce qu'on appelle *the split infinitive*, l'infinitif éclaté).
- Il existe cependant des exceptions dans la langue courante. Le génitif est en premier lieu le cas du complément de nom exprimant la possession. Il s'obtient en ajoutant, selon le cas, une apostrophe et la lettre *s* ou simplement une apostrophe. Exemples :
  - *The cat's ball* (« la balle du chat »)
  - *The teenagers' ball* (« le ballon des adolescents », *teenagers* est déjà un pluriel)
  - *The children's ball* (« le ballon des enfants », *child*ren donne le pluriel irrégulier *child*, qui ne prend pas de *s*)
- L'ordre des mots change également quand on passe d'une phrase affirmative à une phrase interrogative. Exemple : *Are you going to the beach?* (inversion de *you are*) « Est-ce que tu vas à la plage ? ».
- La voix passive existe en anglais : *That cake was eaten by Mary* « Ce gâteau-là a été mangé par Mary ».
- Les articles définis : the house « une maison ».
- Parfois la lettre *h* n'est pas prononcée. Lorsqu'un *h* n'est pas prononcé au début d'un mot, *an* est utilisé : a horse (un cheval) an hour (une heure).
- *An* est en général utilisée plutôt que *a* lorsqu'un nom commence par une voyelle : an apple « une pomme ».
- Les pronoms personnels sujets sont *I*, *you*, *he* / *she* / *it* au singulier et *we*, *you*, *they* au pluriel ; *I* est toujours une majuscule, même s'il n'est pas situé au début de la phrase. *You* est utilisé

pour s'adresser à une seule personne aussi bien qu'à plusieurs. *It* est utilisé pour désigner un objet mais il est également utilisé pour un bébé quand le sexe du bébé est inconnu, ainsi que pour un animal quand le sexe n'est pas connu ou n'est pas important. *They* est utilisé à la fois pour des personnes ou des objets.

- L'anglais n'a pas de concept de genre grammatical pour les substantifs. La différence entre féminin et masculin est pertinente seulement pour les personnes (pronoms personnels *he* et *she*) ; tous les autres noms sont neutres de fait (pronom personnel *it*). Il y a de rares exceptions, par exemple, les navires ou les pays peuvent être traités comme des féminins : *The Titanic was a famous ship. **She** hit an iceberg and sank.* (L'utilisation de *it* pour des bateaux est acceptée toutefois dans la langue courante). Pour les animaux dont le sexe est inconnu, *it* suffit. Si le sexe d'un animal est connu, il est acceptable de remplacer *it* par *he* ou *she* selon le cas. Cette règle se trouve également avec des bébés : *Mary has just had a baby! — Is **it** a boy or a girl?* « Mary vient d'avoir un bébé ! — Est-ce un garçon ou une fille ? ».

II. SECTION : L'ontologie linguistique WordNet.

II.1. Introduction

La recherche d'information traite le problème de trouver tous les documents pertinents dans une collection de texte pour un compte tenu de la requête de l'utilisateur. Une base de données sémantique à grande échelle telles que WordNet [37] semble avoir un grand potentiel pour cette tâche. Il y a au moins trois évident raisons:

- Il offre la possibilité de discriminer mot détecte dans les documents et les requêtes.
- WordNet fournit la chance de faire correspondre sémantiquement mots connexes. Par exemple : fontaine, écoulement, effusion, dans le lieu sens, peuvent être identifiés comme occurrences le même concept, « écoulement naturel des eaux souterraines ». Et au-delà la relation sémantique de synonymie.
- WordNet peut être utilisé pour mesurer la distance sémantique entre survenant termes pour obtenir des moyens plus sophistiqués de la comparaison des documents et des requêtes.

II.2. Domaines de WordNet

Des programmes issus du monde de l'Intelligence Artificielle ont également établi des passerelles avec WordNet. Le WordNet est utilisable librement, y compris pour un usage commercial, ce qui en a favorisé une diffusion très large. Plusieurs autres ressources linguistiques ont été constituées (manuellement ou automatiquement) à partir de, en extension à, ou en complément à WordNet. L'ensemble de ces ressources linguistiques constitue un système complet couvrant des aspects lexicaux, syntaxiques et sémantiques. Combinées, ces ressources fournissent un point de départ intéressant pour des développements sémantiques dans le cadre du Web sémantique, tels que la recherche d'information, l'inférence pour la

compréhension automatique de textes, la désambiguïsation lexicale ou la résolution d'anaphores [38].

II.3. Un projet ambitieux

Depuis 1985, un groupe de psycholinguistes et de linguistes de l'université de Princeton a développé une base de données lexicale selon des principes suggérés par des expériences et des recherches en psycholinguistique sur l'organisation de la mémoire humaine. Depuis cette date, ce projet a pris de l'ampleur ; il se poursuit encore de nos jours.

C'est un réseau sémantique de la langue anglaise, qui se fonde sur une théorie psychologique du langage. La première version diffusée remonte à juin 1991. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise [39]. Des versions de WordNet pour d'autres langues existent (Wolf pour le français, ArabicWordNet pour l'arabe et EuroWordNet[1]), mais la version anglaise est cependant la plus complète et riche à ce jour.

WordNet est distribué sous une licence libre, permettant de l'utiliser commercialement ou à des fins de recherche. La dernière version distribuée en avril 2013 est la 3.1. Cette version est par ailleurs consultable en ligne [40].

II.4. Principe

On peut considérer WordNet comme un graphe ou un réseau lexicale sémantique, souvent qu'on qualifie d'ontologie légère (Light Ontology), où :

- Les synsets sont les nœuds.
- Les relations sémantiques entre synsets sont les arcs.

II.4.1. Les synsets

La composante atomique sur laquelle repose le système entier est le *synset,* c'est un groupe de mots interchangeables, dénotant un sens ou un usage particulier. La version 2.0 de WordNet définit ainsi le nom commun anglais « *car* » à l'aide de cinq synsets comme il est montré dans la figure II.1.

[1] ***EuroWordnet***, un projet de construction d'un ***WordNet*** multilingue a été lancé en mars 1996 (Vossen, 1996). Il concerne initialement l'allemand, l'italien et l'espagnol. La France accuse un certain retard.

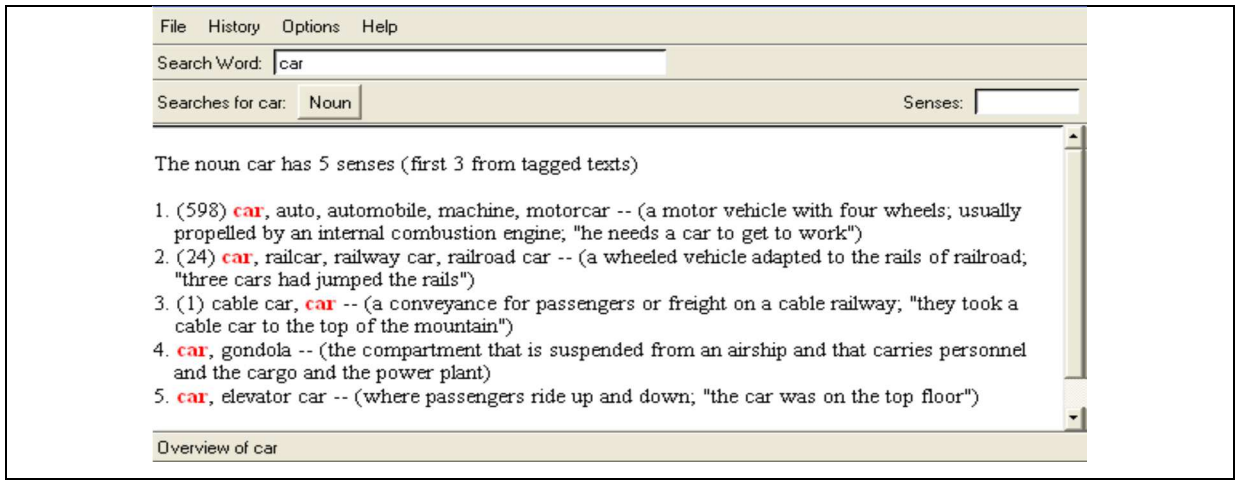

Figure II.1. Les différents sens du mot « car »

Chaque synset dénote une acception différente du mot « *car* », décrite par une courte définition. Une occurrence particulière de ce mot dénotant par exemple le premier sens (le plus courant), dans le contexte d'une phrase ou d'un énoncé, serait ainsi caractérisée par le fait qu'on pourrait remplacer le mot polysémique par l'un ou l'autre des mots du synset sans altérer la signification de l'ensemble [39].

II.4.2. Les relations sémantiques

Dans WordNet, les concepts sont reliés par des relations sémantiques. La relation de synonymie est la relation de base dans WordNet. Elle relie les termes d'un même noeud. Les noeuds (les concepts ou les synsets) sont reliés entre eux par des relations sémantiques telles que, la relation de composition (partie-tout) et la relation hyponymie-hyperonyme (est-un) [39]. comme représentées dans le schéma de la Figure II.2.

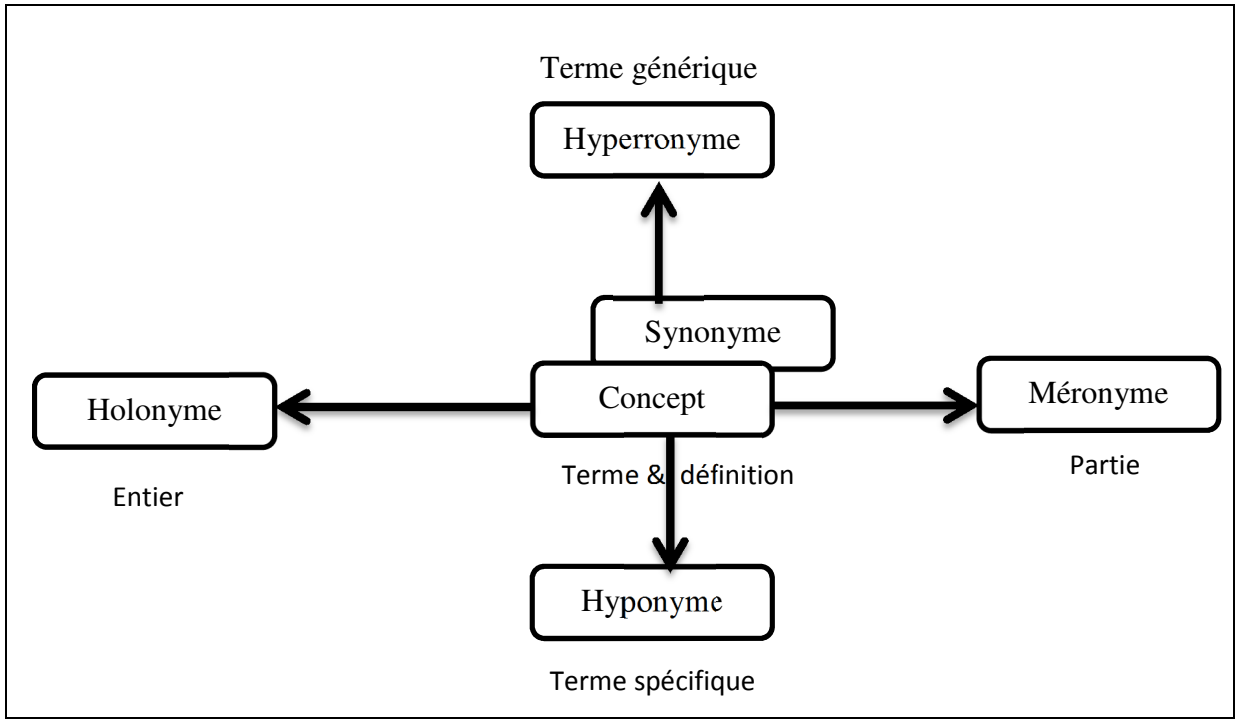

Figure II.2. Les relations entre les Synsets.

II.4.2.1. L'hyperonymie

L'hyperonymie est la relation sémantique hiérarchique d'un lexème à un autre selon laquelle l'extension du premier terme, plus général, englobe l'extension du second, plus spécifique. Le premier terme est dit hyperonyme de l'autre, ou super ordonné par rapport à l'autre. C'est le contraire de l'hyponymie [37].

II.4.2.2. L'hyponymie

L'hyponymie est une relation d'inclusion entre deux mots dont l'un est l'hyponyme de l'autre. La relation d'hyponymie est l'expression linguistique de la relation logique d'inclusion d'une classe dans une autre.

On peut aussi définir les hyponymie comme la relation sémantique d'un lexème à un autre selon laquelle l'extension du premier est incluse dans l'extension du second. Le premier terme est dit hyponyme de l'autre. C'est le contraire de l'hyperonymie. [37] donne un exemple de chaîne hyponymique : *televangelist* < *evangelist*< *preacher* < *clergyman* < *spiritual leader* <*person*[2]

II.4.2.3. La méronymie

La méronymie est une relation sémantique entre mots d'une même langue. Des termes liés par méronymie sont des méronymes. La méronymie est une relation partitive hiérarchisée : une relation de partie à tout. Un méronyme X d'un mot Y est un mot dont le signifié désigne une sous-partie du signifié de Y. La relation inverse est l'holonymie. WordNet inclus trois types de méronymie :

- X est un composante de Y.
- X est un élément de Y.
- X est le matériau dont Y est constitué [41].

II.4.2.4. L'holonymie

L'Holonymie est une relation sémantique entre mots d'une même langue. Des termes liés par holonymie sont des holonomes. L'holonymie est une relation partitive hiérarchisée : un holonyme A d'un mot B est un mot dont le signifié désigne un ensemble comprenant le signifié de B. La relation inverse est la méronymie [41].

II.4.2.5. La Synonymie

La synonymie est un rapport de similarité sémantique entre des mots ou des expressions d'une même langue. La similarité sémantique indique qu'ils ont des significations très semblables. Des termes liés par synonymie sont des synonymes.

Il existe des bases de données de synonymes, présentées comme des dictionnaires, librement téléchargeables. On en trouve aussi vendues ou consultables sous la forme de livres, de logiciels, ou de web, ou des jeux [42].

---

[2] Dans x < y, le mot *x* est donné comme l'hyponyme du mot *y*. On aurait pour le français la séquence suivante : *télé-évangéliste* < *évangéliste* < *prédicateur* < *ecclésiastique* < *chef spirituel* < *personne*.

II.4.2.6. L'antonymies

Deux items lexicaux sont en relation d'antonymie si on peut exhiber une symétrie de leurs traits sémantiques par rapport à un axe. La symétrie peut se décliner de différentes manières, selon la nature de son support. On distingue plusieurs supports qui sont autant de type d'antonymie :

- Les antonymes complémentaires.
- Les antonymes scalaires.
- Les antonymes duals [42].

II.4.2.7. La Troponymie

La troponymie est une relation sémantique entre deux verbes, l'un décrivant de manière plus précise l'action de l'autre. Le premier verbe est dit troponyme du second [41].
La figure ci-dessous montre de manière simplifiée comment le premier sens de *credit* (*crédit*) se situe par rapport aux synsets voisins : c'est un hyponyme de *asset* (*avoir*), un hyperonyme de *credit-card* (*carte de crédit*), un antonyme de *cash* (*argent comptant*).

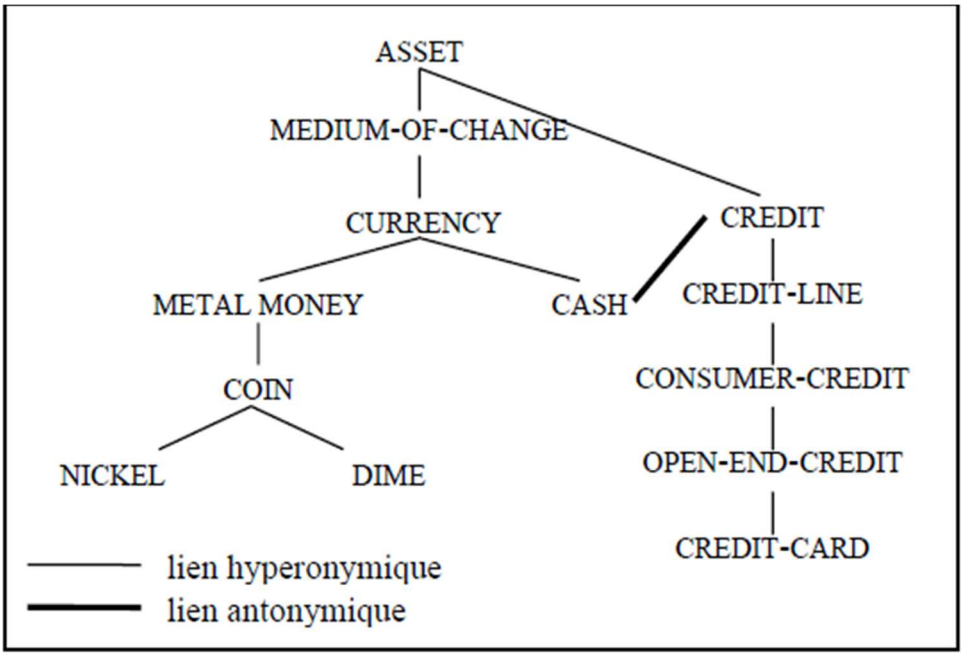

Figure II.4. Exemple de sous-hiérarchie de WordNet.

II.5. Une structure riche et différenciée :

WordNet décompose le lexique en cinq catégories : noms, verbes et adjectives, adverbes. Chacune de ces catégories a sa propre structure interne. « Ce sont des expériences sur les associations de mots qui ont mis en évidence à l'origine que l'organisation varie d'une catégorie syntaxique à l'autre. ».

II.5.1 Des hiérarchies de noms

L'ensemble des noms, qui comporte des formes simples et des mots composés mais pas de noms propres, est organisé autour de la relation d'hyponymie. La structure induite est en fait un ensemble de 25 hiérarchies dominées par des catégories sémantiques générales

Cette structure hiérarchique peut être parcourue de haut en bas ou de bas en haut. À partir d'un sens donné, on peut ainsi retrouver ses ancêtres (hyperonymes directs et indirects), ses descendants (hyponymes directs ou indirects) mais aussi ses frères.

Outre leur place dans cette structure hiérarchique, les sens des noms se définissent par des propriétés : leurs attributs, leur composition et leurs fonctions. La composition est décrite par différents types de relations méronymiques dans WordNet : les relations de composant à objet composé (*branche* / *arbre*), d'élément à ensemble (*arbre* / *forêt*) et de matière (*arbre* / *bois*). En revanche, les attributs (un arbre peut être grand, vieux…) et les fonctions (*une hache sert à couper…*) ne sont pas représentés dans WordNet. Ce sont en effet des relations transcatégorielles qui devraient à terme relier les hiérarchies de noms aux réseaux des adjectifs ou des verbes [43].

II.5.2. Des classes d'adjectifs

Les synsets d'adjectifs comprennent essentiellement des adjectifs qualificatifs. Ces adjectifs ne s'organisent pas comme les noms. Pour les adjectifs, il n'existe pas de relation hiérarchique comme l'hyponymie. La relation fondamentale structurant l'espace des adjectifs est l'antonymie. Cette relation symétrique, mise en évidence par des tests psycholinguistiques sur les associations de mots, est difficile à formaliser. Les auteurs retiennent l'idée que les adjectifs antonymes expriment deux valeurs opposées d'un même attribut [44].

Partant cependant du constat que certains adjectifs proches par le sens par exemple : *heavy* et *weighty* « lourd/pesant » ont des antonymes différents *light* et *weigthless* « lumière/en état d'apesanteur » et que beaucoup d'adjectifs qualificatifs *ponderous* « lourd » n'ont pas d'antonymes directs, la structure retenue est celle de classes d'adjectifs similaires entre eux, ces classes étant organisées autour d'adjectifs pôles qui peuvent s'opposer à d'autres pôles par des liens d'antonymie. *heavy* et *light* sont donc considérés comme antonymes, mais *ponderous*, qui est similaire à *heavy* et qui n'a pas d'antonyme direct n'est qu'un antonyme indirect de *light* [39].

II.5.3 Des réseaux de verbes

Comme les noms et les adjectifs, les verbes sont regroupés en synsets. Ceux-ci comportent des formes simples mais aussi des tournures verbales, comme *look up « chercher »*, qui sont très fréquentes en anglais. Les synsets se répartissent eux-mêmes en 15 catégories générales.

La relation centrale pour le réseau des verbes n'est ni l'hyponymie, ni l'antonymie, mais l'implication. En distingue quatre types : la cause (*give* / *have* : *donner* / *avoir*), la présupposition (*succeed* / *try* : *réussir* / *essayer* ou *untie* / *tie* : *dénouer* / *nouer*), l'inclusion (*snore* / *sleep* : *ronfler* / *dormir* ou *buy* / *pay* : *acheter* / *payer*) et la troponymie (*limp* / *walk*, *boiter* / *marcher*).

Soulignant toutefois la complexité de la sémantique des verbes et la difficulté de définir une sémantique proprement différentielle, les auteurs de WordNet reconnaissent la moindre maturité du réseau des verbes.

Dans la pratique, les travaux qui exploitent ce réseau des verbes à des fins de désambiguïsation lexicale s'en tiennent souvent aux grandes catégories sémantiques [45].

II.6. Quelques données statistiques

Dans cette partie, nous présentons, de manière quantitative, le contenu de WordNet. La table II.1 montre la structure de WordNet en nombre de mots, nombre de synsets et nombre de sens globalement et par catégorie grammaticale. Du nombre total de formes décrites, la plupart sont des noms (74.6%), le reste étant constitué par des adjectif (14.6%), des verbes(7.6%) et des adverbes(3.2%). La polysémie (nombre de sens par mot) se manifeste dans WordNet par le fait qu'il y a des mots qui peuvent appartenir à plusieurs synsets (146350 formes traitées /111223 synsets) [46].

Partie de discours	**Nombre de mots**	**Nombre de synsets**	**Nombre de sens**
Noms	109195	75804	134716
Verbes	11088	13214	24169
Adjectifs	21460	18576	31184
Adverbes	4607	3629	5748

Tableau II.1. Nombre de mots, synsets et sens sans WordNet.

La taille du vocabulaire couvert suffit à donner la mesure de l'ambition qui a présidé à la construction de ce réseau. WordNet comporte 95 600 unités lexicales différentes : 51 500 mots simples et 44 100 expressions (*collocations*). À ces mots sont associés quelques 70 100 sens différents. Le Tableau II.2 montre comment ces unités et sens se répartissent [46].

	Noms	Verbes	Adjectifs
Nombre d'unités lexicales	57000	21000	19500
Nombre de sens	48800	8400	10000
Nombre de catégories générales	25	14	

Tableau II.2. Exemple de sous-hiérarchie de WordNet.

II.7. Les points forts de WordNet :

L'utilisation de WordNet en recherche d'informations :

- Pour étendre la requête de l'utilisateur (ajout de synonymes, par exemple pour augmenter le rappel, c'est-à-dire la proportion de documents pertinents rapportés).
- Acquisition de relations sémantiques.
- Désambiguïsation sémantique.
- Pour l'étiquetage sémantique de corpus.
- Pour la structuration et catégorisation des documents.

En général WordNet est utilisé :

- Pour la recherche d'informations.

- Pour l'extraction d'informations.
- Pour les systèmes de questions/réponses.
- Pour enrichir la représentation avec des synonymes, hyperonymes, etc. [46].

Ceci nous amène à souligner l'absence de ressources similaires pour le français. Si la recherche sur les corpus en français peut sans doute tirer profit de l'expérience anglo-saxonne pour éviter certains tâtonnements, des problèmes spécifiques se posent pour chaque langue, qui imposent certains ajustements, voire la mise au point de méthodes particulières ou le développement d'outils spécifiques.

L'absence de ressources lexicales informatisée pour le français est déjà un frein pour tous les traitements sémantiques. Faute de moyens, la plupart des travaux français s'intéressent à l'acquisition de connaissances à partir de corpus.

II.8. Les limite du WordNet :

II.8.1. Informations manquantes :

WordNet ne précise pas l'étymologie, la prononciation, les formes de verbes irréguliers.

II.8.2. Profusion de sens pour un mot donné

La contrepartie de son importante couverture est que WordNet est très précis dans le sens des définitions. On a une granularité très fine des sens. Par exemple, le verbe *to give* (« donner ») n'a pas moins de 44 sens. Une telle profusion ne facilite pas une tâche de désambiguïsation lexicale.

II.8.3. Absence de relations pragmatiques

WordNet ne matérialise pas d'une façon formelle tout le sens contenu dans les définitions des termes. Par exemple, l'information qu'un chat ne rugit pas figure dans la définition, mais ne se retrouve formalisée dans aucune relation. De même, des relations pragmatiques telles que savon / bain (Soap / Bath) sont absentes de WordNet.

Conclusion

Dans ce chapitre nous avons décrit la langue anglaise et ses caractéristiques linguistiques, nous avons aussi présenté en détail WordNet et son principe de fonctionnement qui est basé sur la notion de synset et de relation sémantique. Â la fin du chapitre nous avons présenté quelques données statistiques de cette base de données lexicale, ainsi que ses avantages et ses limites. Le chapitre suivant est destinée à une présentation générale de la modélisation de notre travail avec le langage de modélisation *UML (Unified Modelling Language).*

1. Introduction

Tout au long de ce chapitre, nous allons identifier les fonctionnalités du système à réaliser, ce qui nous conduira à la description des besoins de notre système ainsi que l'analyse et la conception objet de ces besoins. Ce qui nécessite des méthodes permettant de mettre en place un modèle, parmi lesquelles nous avons choisi le langage UML, nous décrivons, par les différents diagrammes d'UML, notre modélisation.

2. Qu'est-ce que UML ?

UML (Unified Modelling Language) le langage de modélisation unifié, est un langage qui s'impose à l'heure actuelle comme le standard de modélisation des applications informatiques : il est utilisé dans le développement logiciel, dans la conception orientée objet et la modélisation. Il propose plusieurs types de diagrammes qui permettent de modéliser tous les aspects d'une application informatique [47]. Nous sommes intéressés aux diagrammes suivants pour modéliser notre système :

- Le diagramme de cas d'utilisation utilisé pour l'expression des besoins.
- Le diagramme de séquence pour décrire les procédures les plus importantes.
- Le diagramme de classes pour la modélisation des données et des relations entre-elles.
- Le diagramme d'activité afin de montrer l'enchaînement des activités des acteurs du système.
- Le diagramme de composants dans le but de décrire l'organisation du système du point de vue des éléments logiciels.
- Le diagramme de déploiement utilisé pour la disposition physique des ressources matérielles qui composent le système.

Afin de vous accompagner dans nos modélisations UML, nous vous présentons la plateforme OpenSource **StarUML**[48].

3. L'outil StarUML

On a utilisé le logiciel StarUML comme outil de modélisation, StarUML est une plate-forme de génération des modèles basés sur le langage UML. L'avantage de cet outil UML est le fait que tous les diagrammes UML peuvent être générés, ainsi que l'exportation au format JPG afin d'intégrer les diagrammes au sein de documents [49]. Les langages de programmation Com, C++, C# et Delphi sont pris en charge. StarUML supporte également l'architecture MDA[1] qui offre comme avantage la personnalisation des profils UML.

[1] MDA (*Model Driven Architecture*) est un processus de l'ingénierie dirigée par les modèles (ou MDE pour *Model Driven Engineering*). Proposée par l'OMG (*Object Management Group*) en 2000, l'approche MDA est basée sur la séparation des préoccupations. Elle permet prendre en compte, séparément, aspect métier et aspect technique d'une application, grâce à la modélisation. Le code source de l'application est obtenu par génération automatique à partir des modèles de l'application. Les modèles ne sont plus seulement un élément visuel ou de communication, mais sont, dans l'approche MDA, un élément productif et le pivot du processus MDA.

En somme, StarUML est complet, robuste et présente une pléthore d'outils et de paramètres. Il est toutefois dédié principalement aux utilisateurs chevronnés et aux projets complexes [48].

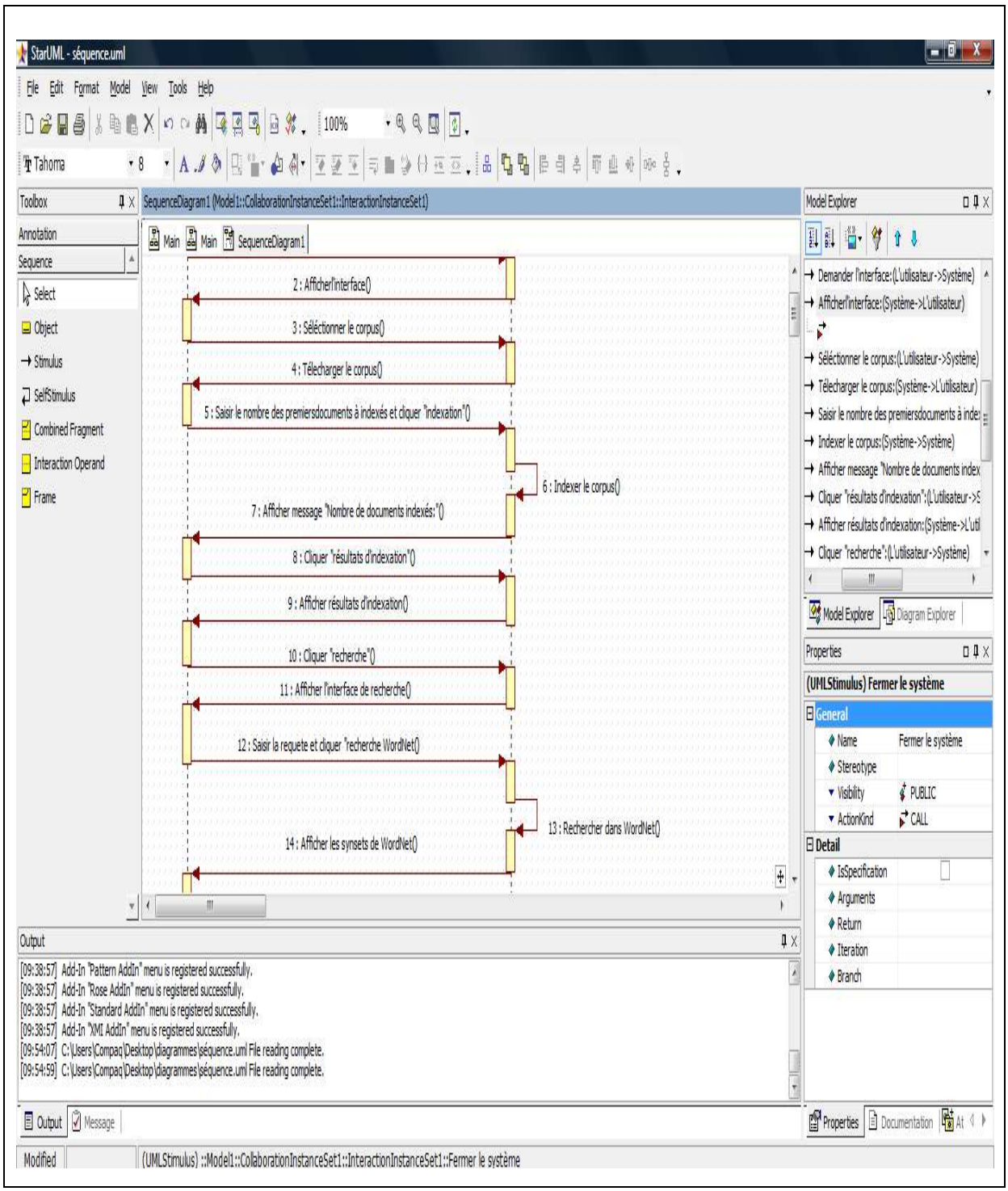

Figure III.1. Interface de l'outil StarUML.

4. Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur. Nous présentons dans (Figure III.2) le diagramme de cas d'utilisation global de notre application.

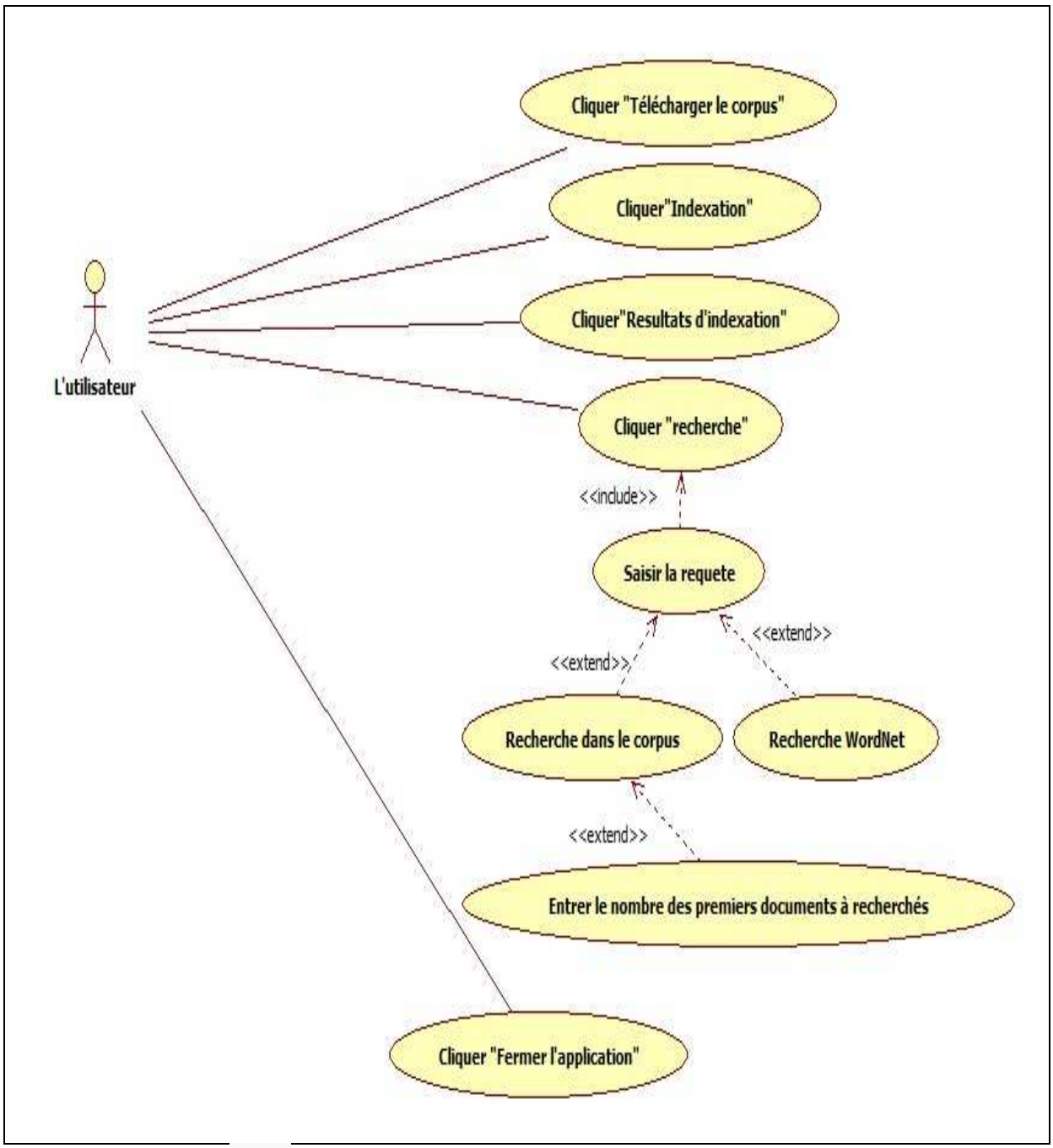

Figure III.2. Diagramme de cas d'utilisation du système en générale.

5. Diagramme de classe

Le diagramme de classes UML décrit les structures d'objets et d'informations utilisées par notre application, il fournit une vue conceptuelle de l'architecture de notre application voir (Figure III.3).

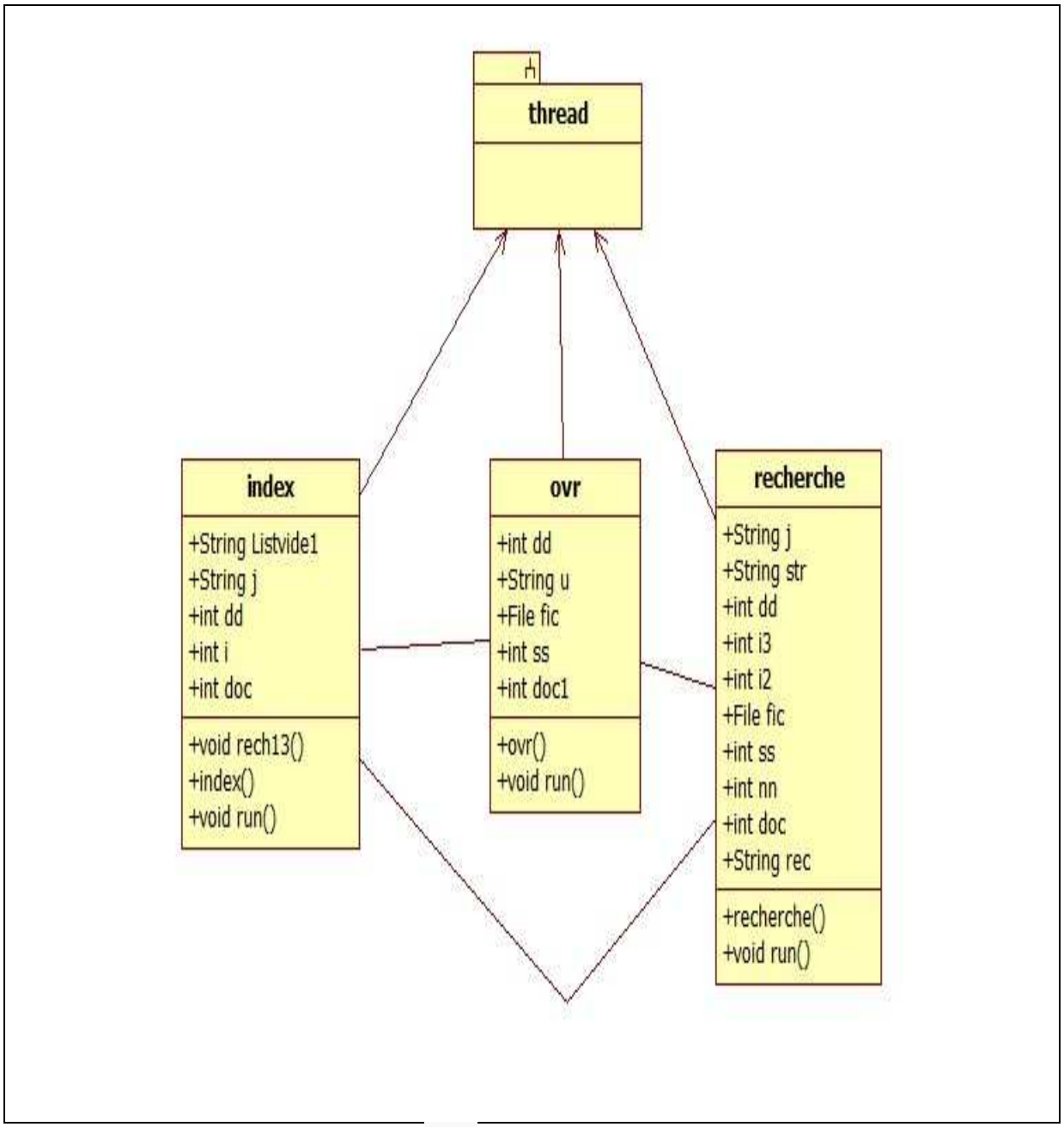

Figure III.3. Diagramme de classe.

6. Diagramme de séquence

Le séquencement de différentes tâches effectuées par les acteurs (Utilisateur et Système) est montré par le diagramme de séquence dans la Figure suivante :

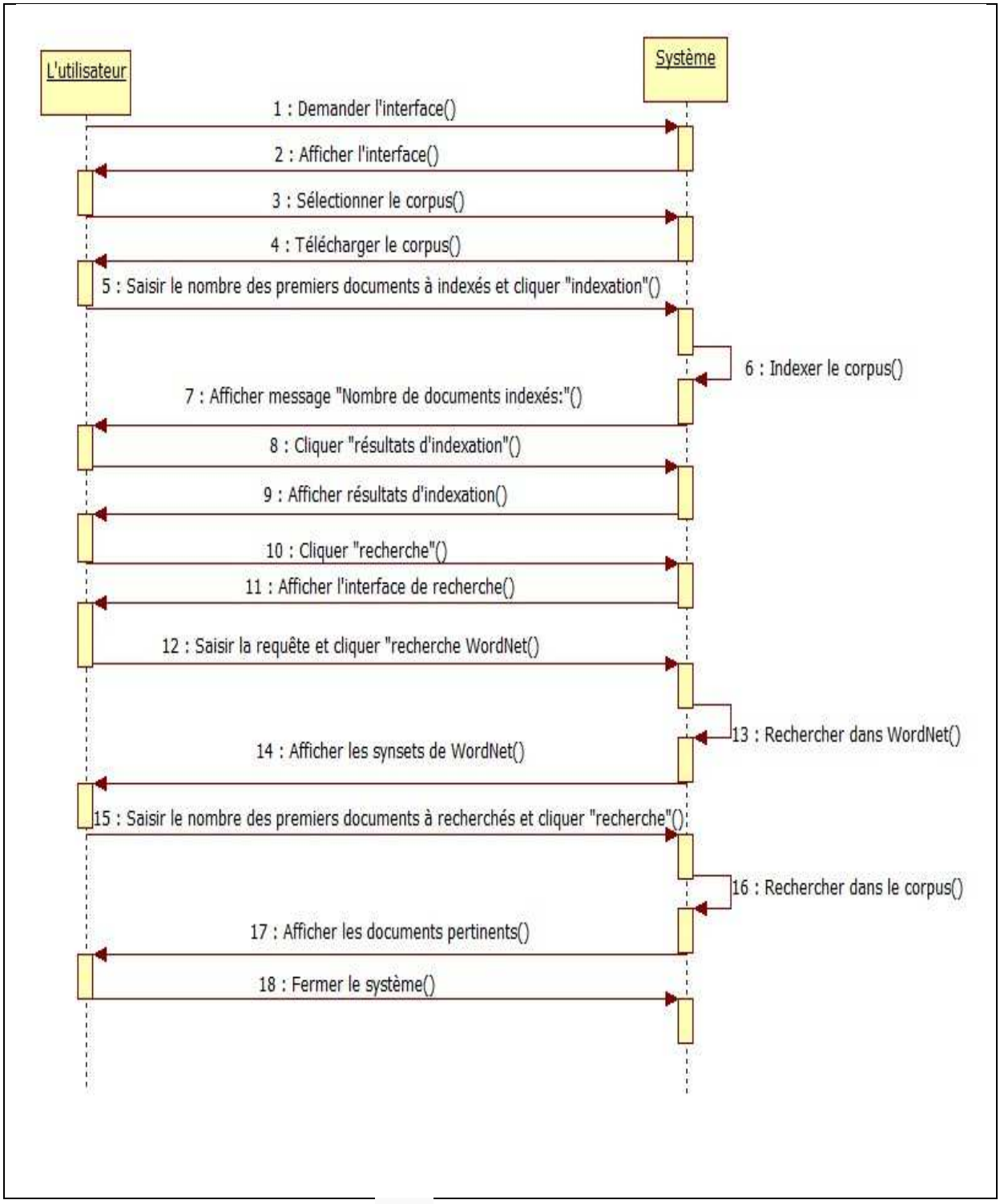

Figure III.4. Diagramme de séquence.

7. Diagramme d'activités

Nous montrons par la (Figure III.5) l'enchainement de toutes les activités qui peuvent être réalisés par notre système et l'utilisateur.

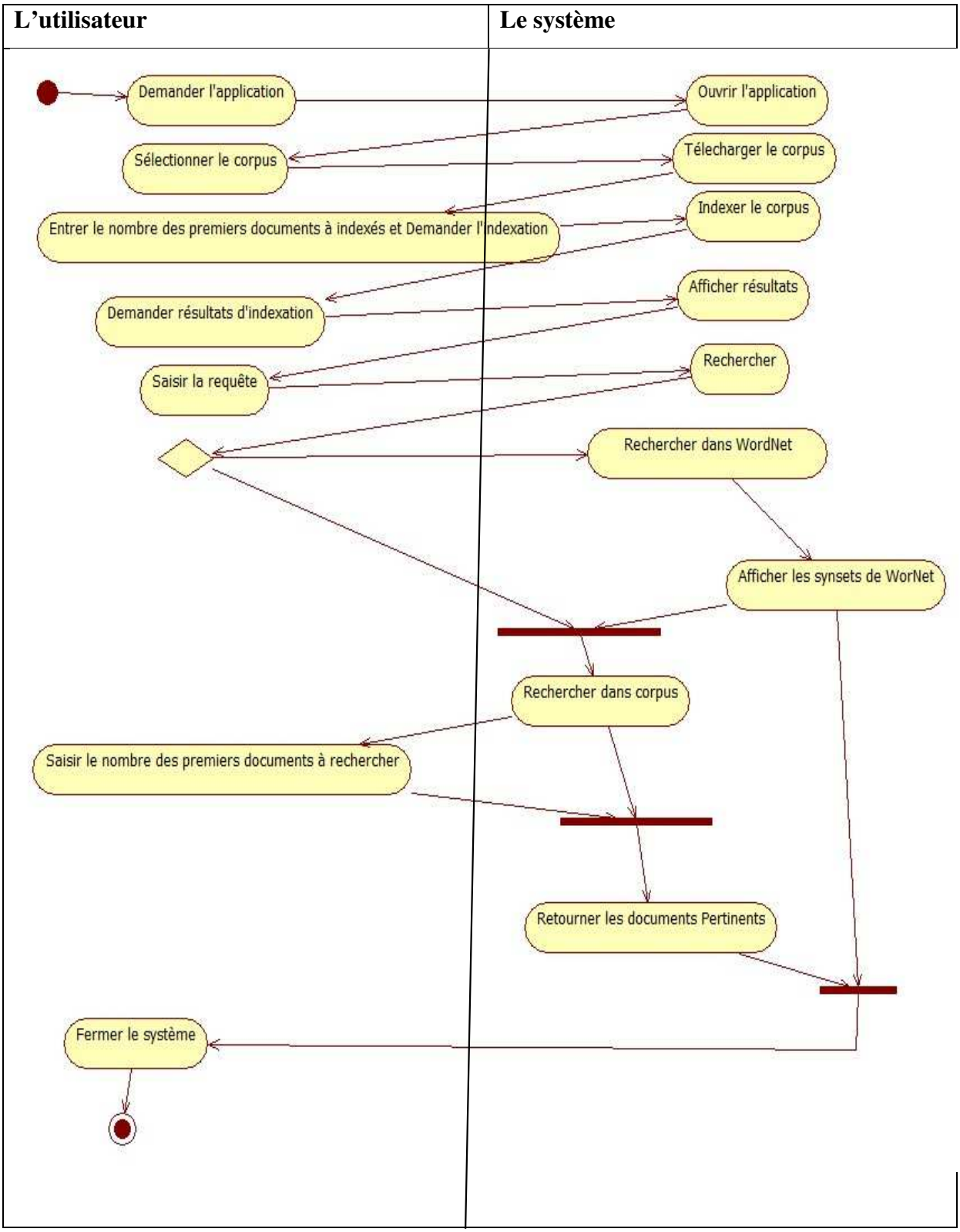

Figure III.5. Diagramme d'activités.

8. Diagramme de composant

Nous décrivons par la Figure ci-dessus tous les éléments logiciels et composants de notre application.

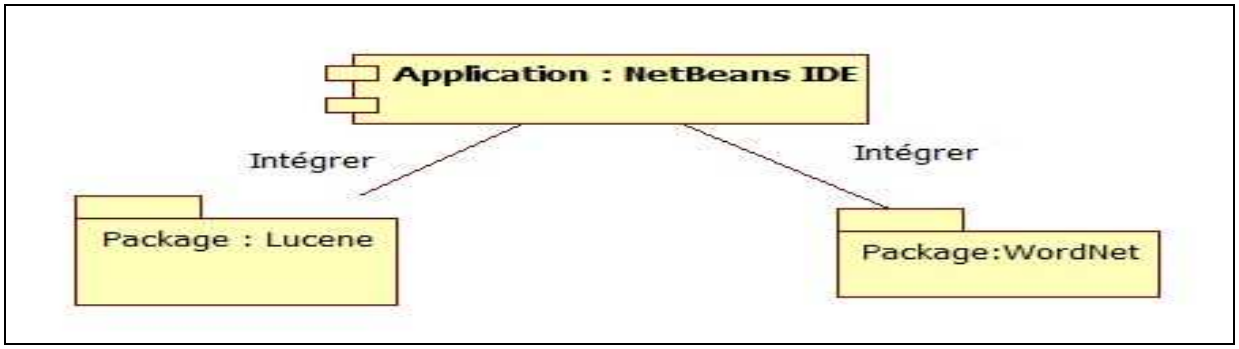

Figure III.6. Diagramme de composant.

9. Diagramme de déploiement

La disposition physique des ressources matérielles qui composent notre système est monté par le diagramme de déploiement subséquent.

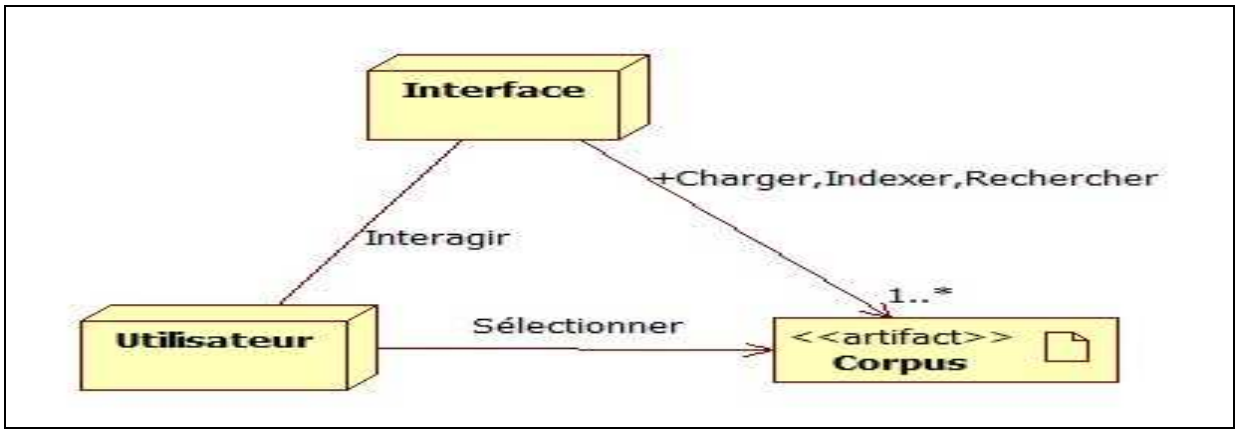

Figure III.7. Diagramme de déploiement.

10.Conclusion

Dans ce chapitre, nous avons fait une présentation générale de la modélisation avec le langage de modélisation *UML* ainsi que l'outil StarUML. Nous avons documenté et détaillé les tâches (par des diagrammes d'UML) que nous allons réaliser dans le chapitre suivant.

1. Introduction

Il s'agit maintenant de mettre en œuvre les étapes explicitées dans les chapitres précédents pour concevoir et implémenter une interface d'indexation et de recherche d'informations sémantique d'un corpus Anglais.

Nous allons dans un premier temps présenter notre environnement d'implémentation : les ressources et les outils utilisées. Ensuite, en expliquant les étapes d'indexation, il s'agit de la conception. Et en terminera par la présentation de différentes interfaces et fonctionnalités de notre application

2. Le corpus utilisé

Nous avons utilisé un corpus en anglais composé de 300.000 documents de domaine médical chimique de taille 324 MO, composé d'un ensemble d'articles de différents journaux d'Amérique couvrant la période 1988-1991.

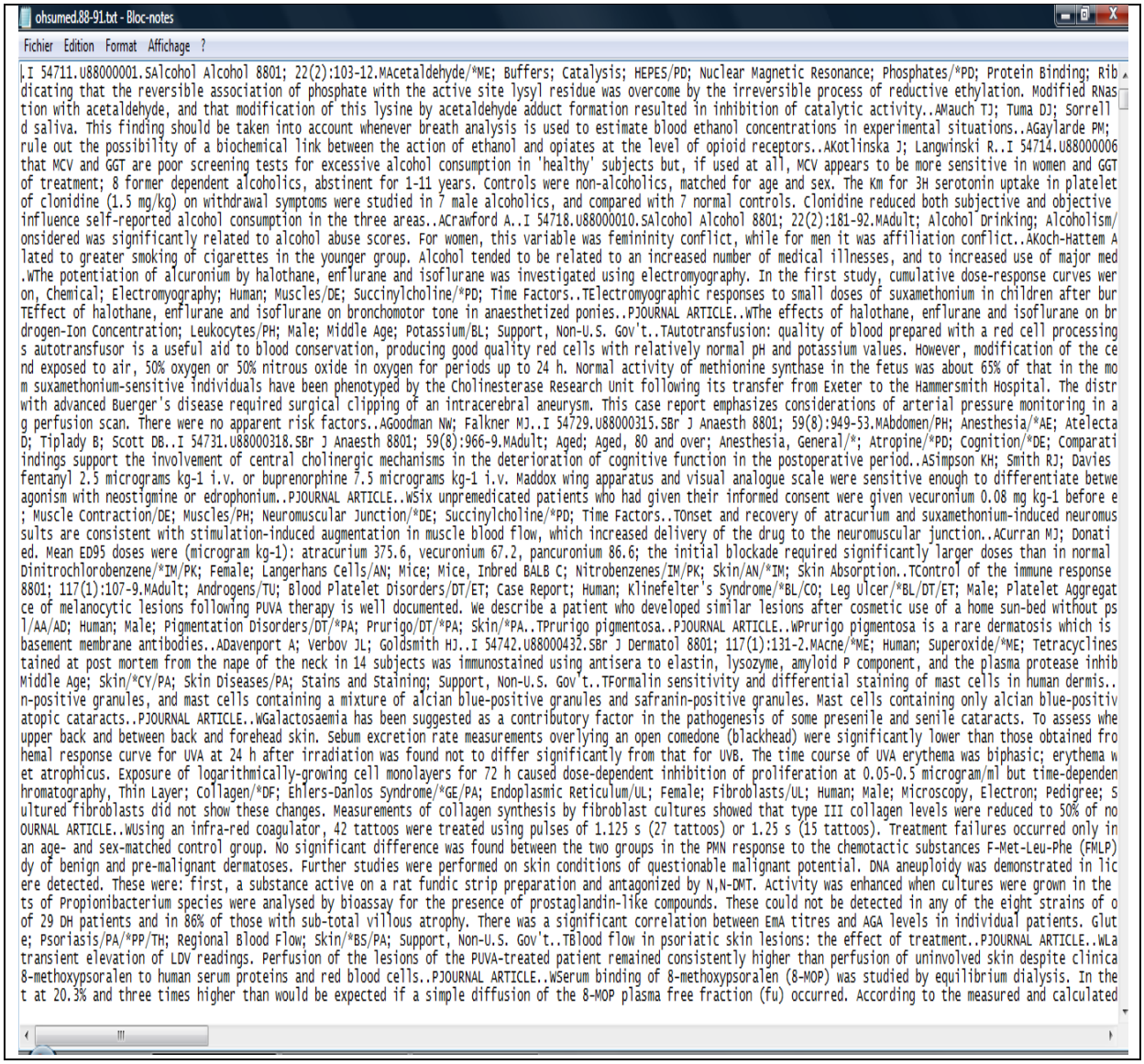

Figure IV.1. Le corpus du système.

3. L'environnement de l'application

L'implémentation et les tests de notre application ont été réalisés dans l'environnement matériel et logiciel suivant :

- Processeur : AMD Sempron ™ SI-42 2.10 GHz.
- Mémoire installée (RAM) : 2.00 GO.
- MS-Windows Vista™ Edition Familiale Basique, Type de système : système d'exploitation 32 bits.
- Java sous l'environnement NetBeans IDE 8.0.2.
- Package de l'ontologie anglaise WordNet 3.0 pour la désambiguïsation des sens des mots.
- Package de moteur de recherche lucene pour l'étape de racinisation pendant l'indexation.
- Corpus anglais de 300.000 documents du domaine médical chimique.

3.1. Langage d'application

Java est un langage de programmation et une plate-forme informatique créée par Sun Microsystems en 1995, racheté plus tard par Oracle. Il s'agit de la technologie sous-jacente qui permet l'exécution des applications modernes sur différentes plateformes. La portabilité, des programmes Java sur différents systèmes d'exploitation, représente son atout principal. Java est utilisée sur plus de 850 millions d'ordinateurs de bureau et un milliard de périphériques dans le monde, dont des périphériques mobiles et des systèmes de diffusion télévisuelle [50].

3.2. IDE NetBeans

NetBeans, créé à l'initiative de Sun Microsystems, présente toutes les caractéristiques indispensables à un IDE de qualité, que ce soit pour développer en Java, Ruby, C/C++ ou même PHP.

De licence Open Source, NetBeans permet de développer et déployer rapidement et gratuitement des applications graphiques Swing, des Applets, des JSP/Servlets, de l'architecture J2EE, dans un environnement fortement personnalisable [51].

3.3. Bibliothèques de WordNet utilisé

Nous avons utilisé les deux bibliothèques de WordNet afin de manipuler les méthodes de ses différents class suivant :

- WordNetGlossTagLibrary : com.gtl.GlossTag;
- JWNL(Java WordNet Library) : net.didion.jwnl.JWNL;
  net.didion.jwnl.JWNLException;
  net.didion.jwnl.data.IndexWord;
  net.didion.jwnl.data.POS;

net.didion.jwnl.data.Synset;

net.didion.jwnl.data.Word;

net.didion.jwnl.dictionary.Dictionary;

3.4. Biblithèque de Lucene utilisé

Est un moteur de recherche et d'indexation développé dans le projet Apache. C'est un logiciel open source signifiant que son code source est libre et accessible gratuitement. Ce logiciel est une librairie de fonctions de recherche dans contenu textuel des documents. Il inclut une interface de programmation (API).

A la base, *Lucene* est écrit en Java mais il est maintenant disponible pour d'autres langages de programmation tels que Python, PHP, Delphi, Perl, C++, C# et Ruby. *Lucene* peut être utilisé avec de nombreux systèmes, c'est une multiplateforme pour : Windows, MacOS et Linux où il est plus précisément intégré à Ubuntu, Debian et Redhat [52].

Lucene est capable de traiter de grands volumes de documents grâce à sa puissance et à sa rapidité dues à l'indexation.

Nous avons utilisé la bibliothèque de *lucene* suivant dans le but d'effectuer l'étape de racinisation de la phase de normalisation d'indexation :

- org.apache.lucene : org.apache.lucene.analysis.EnglishAnalyzer;
  org.apache.lucene.queryParser.QueryParser;
  org.apache.lucene.util.Version;

4. Processus d'indexation

Avant de naviguer dans le corpus, il est utile d'appliquer certains prétraitements, ces prétraitements sont les étapes les plus importantes des phases du processus d'indexation clarifiées en CHAPITRE I suivant :

4.1. Segmentation

Nous avons réalisé cette phase selon 2 étapes :

Etape 1 :

Permet de segmenter le corpus en un ensemble de documents dans le cas de notre corpus il s'agit de construire des nouveaux documents en se basant sur le caractère « .I » cette étape se déroule comme suit :

Entrée: Corpus. Sortie: Corpus segmenté en documents.
Tant que (le corpus n'est pas fini) faire Si (trouver le caractère « .I ») Alors Créer un nouveau document Retourner document; Fin; Fin;

Tableau IV.1. Algorithme de segmentation d'un corpus en un ensemble de documents.

Etape2 :

En séparant les mots entre eux en se basant sur le caractère de blanc, la procédure se déroule comme suit :

Entrée: document. Sortie: document segmenté en mots.
Tant que (le document n'est pas fini) faire Si (trouver un délimiteur d'espace) Alors Découper les mots de document Retourner sac de mots; Fin; Fin;

Tableau IV.2. Algorithme de segmentation d'un document en sac de mots.

4.2. Normalisation

Comme nous l'avons déjà précisé dans le premier chapitre, la normalisation traite plusieurs niveaux pour manipuler les variations du texte, nous avons exécuté plusieurs genres de normalisation sur le texte de corpus.

4.2.1. Niveau syntaxique

- **Elimination des caractères spéciaux**

En désignant l'ensemble des caractères spéciaux (chiffres et symboles) comme :

["," + "." + "%" + "0" + "1" + "2" + "5" + "3" + "4" + "6" + "7" + "8" + "9" + ";"+ "!" + "?" + "~" + "/"+"+"+"-"+"*"+";"+""+")"+"("+"{"+"}"+"["+"]"+"\"+ ""+":"+"§"]

Entrée: document segmenté en mots. Sortie: document normalisé
Pour chaque mot de document faire Si (mot est un caractère spécial) Alors Supprimer le mot de document Retourner document Fin; Fin.

Tableau IV.3. Algorithme d'élimination des caractères spéciaux.

- **Elimination des mots vides**

Un des problèmes majeurs de l'indexation consiste à extraire les termes significatifs et à éviter les mots vides. Nous distinguons deux techniques pour éliminer les mots vides :

➔L'utilisation d'une liste de mots vides (aussi appelée anti-dictionnaire).

➔L'élimination des mots dépassant un certain nombre d'occurrences dans la collection.

Nous avons utilisé la première technique. L'élimination des mots vides à l'avantage de réduire le nombre de termes d'indexation, elle peut cependant augmenter le taux de rappel ;

c'est à dire la proportion de documents pertinents retournés par le système par rapport à l'ensemble des documents pertinents.

La liste des mots vides contient les pronoms personnels, les prépositions, les articles….etc. Nous avons utilisé la liste des mots vides suivante :

```
String Listvide1[] ={ "\n",
"a","\"","«","»","a","about","above","after","again","against","all","am","an","and","any","are","aren","arent"
,"as","at","be","because","been","before","being","below","between","both","but","by","can","cannot","can't"
,"could","couldnt","did","didn't","do","does","doesn't","doing","don't","down","during","each","u","m","l","q"
,"j","few","for","from","further","had","has","hasn't","have","haven't","having","he","hed","hell","hes","her"
,"here","heres","hers","herself","him","himself","his","how","hows","i","id","if","i'll","im","in","into","is","isn't"
,"it","it'll","its","itself","i've","let","lets","me","more","most","mustnt","my","myself","no","nor","not","now"
,"of","off","on","once","only","or","other","others","ought","our","ours","ourselves","out","over","own","p"
,"page","pages","part","past","per","perhaps","placed","please","plus","poorly","possible","possibly","pp"
,"present","proud","put","q","que","quickly","quite","qv","r","ran","rather","rd","re","really","recent","ref"
,"refs","regards","related","relatively","research","resulted","resulting","results","right","run","s","said","same"
,"saw","say","says","sec","section","see","seem","seemed","seems","seen","self","sent","seven","she","shed"
,"she'll","shes","should","shouldn't","show","shows","since","six","so","some","sorry","such","sup","sure"
,"t","take","tell","tends","than","thank","that","that'll","thats","that've","the","their","them","then","there"
,"therd","there'll","thereof","lingspam","part10","part1","part2","part3","part4","part5","part6","part7","part8"
,"part9","public","these","they","theyd","they'll","theyre","they've","think","this","those","though","to","too"
,"two","under","unto","up","use","until","desctop","'ve","w","want","wants","was","wasn't","way","we","wed"
,"welcome","we'll","went","were","weren't","we've","what","whatever","what'll","whats","when","where"
,"wheres","which","while","whim","who","whod","whole","who'll","whom","whos","whose","why","why's"
,"with","without","yes","you","youd","you'll","your","youre","yours","yourself","yourselves","zero};
```

Tableau. IV.4. Liste des mots vides.

Si le mot apparu est un mot vide alors le système va le supprimer suivant l'algorithme de Tableau IV.5.

Entrée : document normalisé avec une liste des mots vides Sortie : document sans mots vides
Tant que (le document n'est pas fini) Faire Si (le mot appartient à la liste des mots vides) alors Supprimer le mot du document ; Retourner document ; Fin ; Fin.

Tableau IV.5. Algorithme d'élimination des mots vides

- **Transformation des majuscules en minuscules:**

En effet le mot "GIRL" et le mot "girl" vont être considérer différent alors qu'ils ont le même sens donc on transforme les majuscules en minuscule.

4.2.2. Niveau lexicale et morphologique

Ici, nous avons utilisé le procédé de *Racinisation* à l'aide des méthodes de la bibliothèque de lucene vue auparavant pour obtenir le « lexème » la forme élémentaire de chaque mot de corpus et requête, c'est en quelque sorte sa *racine linguistique*.

La racinisation est un procédé complexe et il est différent dans chaque langue, c'est pourquoi il n'existe pas d'algorithme informatique pour toutes les langues. De même, les algorithmes existants ne sont pas implémentés dans tous les langages informatiques.

4.2.3. Niveau sémantique

Notre traitement sémantique est réalisé par « Mapping des mots en sens » cette étape consiste a remplacé le mot par ces sens, ces sens représentent les synsets du l'ontologie WordNet, en effet chaque mot peut appartenir à plusieurs sens.

Exemple: le mot "Girl" devient :{ girl, miss, missy, young lady, young woman, fille, female child, little girl, daughter, girlfriend, lady friend}.

4.3. Indexeur

Comme nous avons déjà expliqué dans CHAPITRE I, ici on utilise une approche permettant de sélectionner les mots normalisés et de leur associer *une pondération*, cette dernière permet d'assigner aux termes leur degrés d'importance dans les documents, il existe plusieurs approches pour le choix des index, nous avons utilisé l'approche suivante :

- **Approche basée sur la fréquence d'occurrences** : Cette approche consiste à choisir les mots représentants selon leur fréquence d'occurrence dans les documents.

Le système calcule le nombre de chaque mot ainsi que ses synsets de WordNet dans chaque document de corpus.

5. Mise en œuvre

5.1. Fenêtre principale

La fenêtre principale à partir de laquelle l'utilisateur peut effectuer les traitements désirés selon le diagramme de cas d'utilisation décrit en CHAPITRE III est montrée dans la Figure IV.2.

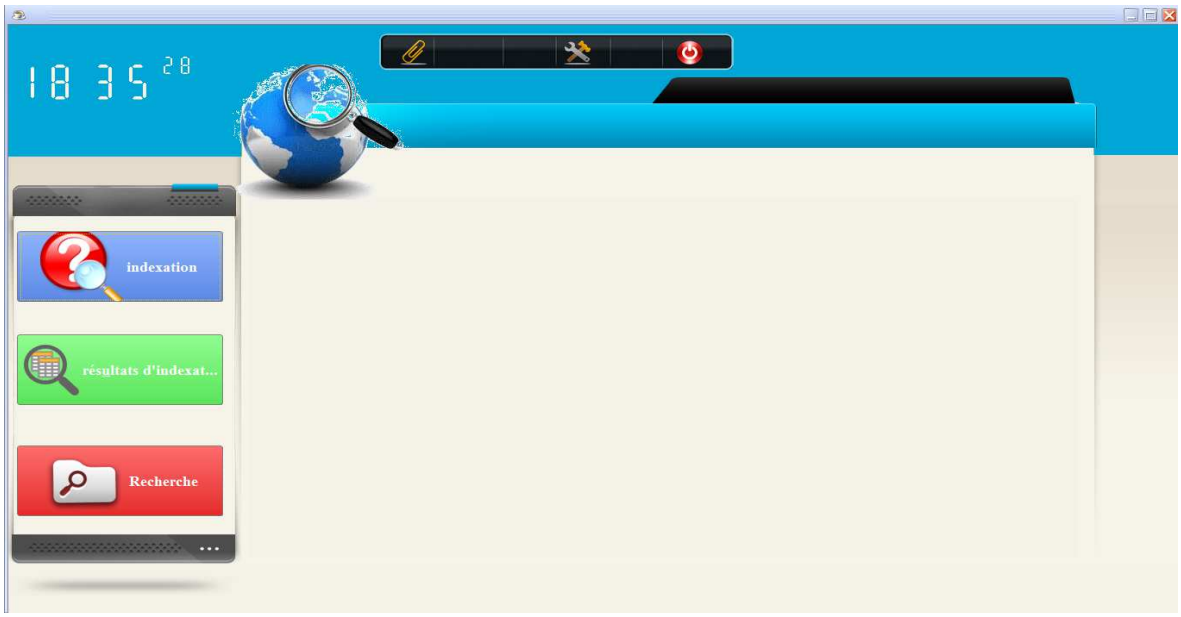

Figure IV.2. Fenêtre principale.

4.2.Chargement de corpus

La fenêtre principale montre 3 icônes supérieurs : la première pour le chargement de corpus, la deuxième pour la saisie des premiers nombres de documents à indexer ou à rechercher et la troisième pour quitter l'application. En cliquant sur la première, une boite de dialogue s'ouvre, en sélectionne le corpus et le système va le charger comme la figure suivante montre.

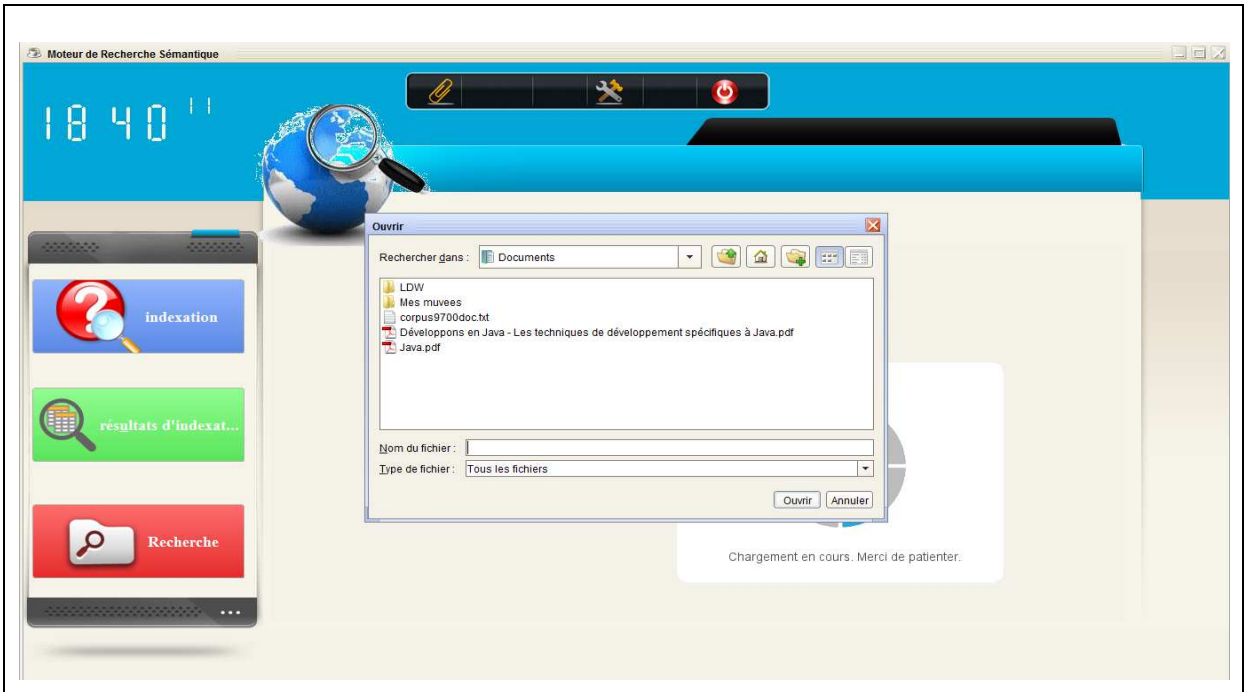

Figure IV.3. Chargement de corpus.

4.3. Indexation

Après le chargement de corpus, en passe à l'étape d'indexation en cliquant sur l'icône d'indexation, notre système va démarrer la réalisation des étapes d'indexation vu auparavant.

Figure IV.4. Indexation de corpus en cour.

4.4.Résultat d'indexation :

La deuxième icone à gauche nous permet de visualiser le résultat d'indexation, par le tableau au milieu de la Figure IV.5 en peut consulter le contenu de chaque document avant Figure IV.6 (la liste des documents à gauche) et après Figure IV.7 (la liste des documents à droite) l'indexation.

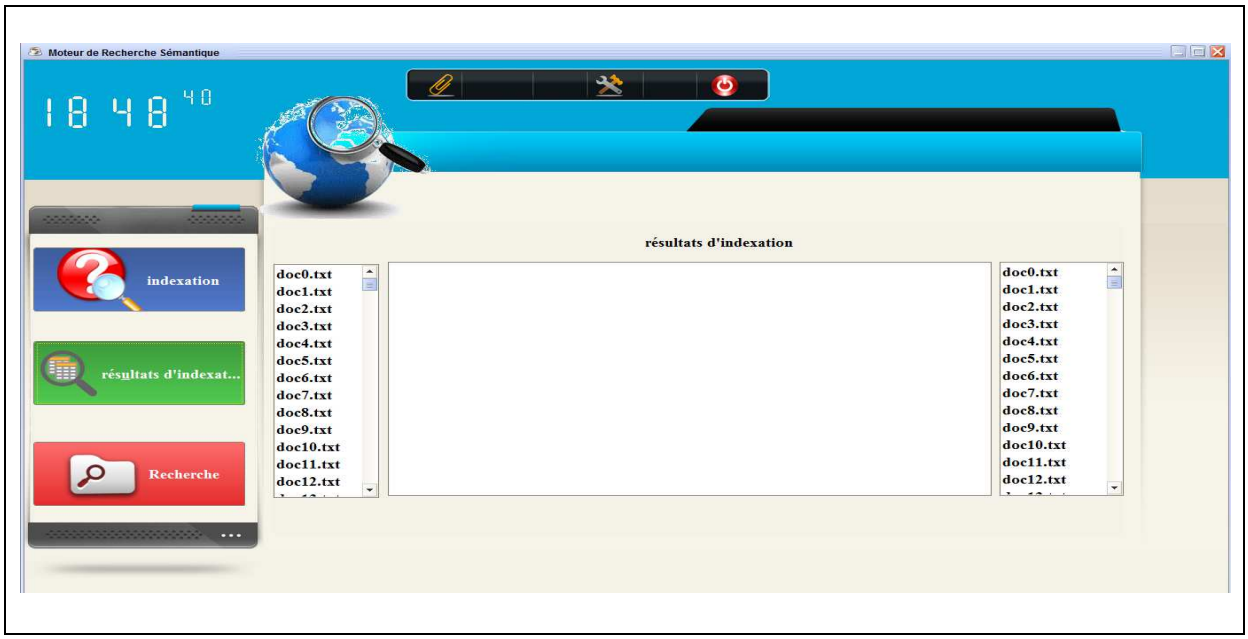

Figure IV.5. Résultat d'indexation.

Figure IV.6. Liste des documents avant indexation.

Figure IV.7. Liste des documents après indexation.

4.5. Recherche :

Le volet de recherche propose à l'utilisateur de saisir une requête avant d'effectuer une recherche selon les deux alternatives ; la première pour une recherche sur le corpus tandis que la deuxième déploie l'ontologie *WordNet* pour inclure les synsets possibles des termes de la requête, nous détaillons par la suite le fonctionnement de la recherche.

Recherche Sémantique : Pour saisir le terme à rechercher (requête d'utilisateur).
Rechercher : Permet de lancer la recherche dans les documents indexés.
Résultats : Afficher la liste des documents pertinents avec la possibilité de les consulter.
Recherche WordNet : Afin d'offrir à l'utilisateur un moyen d'explorer le sens des termes et d'expliquer la requête, la ressource WordNet est intégré dans notre application.

- **Processus de recherche :**

Le processus de la recherche des documents permet non seulement de retournés les documents qui contient le terme de la requête mais aussi les synsets de cette dernière extrait à partir de WordNet. Exemple : en prenant la requête « estimate », la consultation des documents résultats montre les synsets de WordNet : (doc1.txt) comporte le verbe du terme de la requête « to estimate » et le document suivant (doc73.txt) comporte le synset du terme de la requête « idea » ainsi que (doc8060.txt) contient le synset de la requête « approximately » (Voir Figure IV.8, Figure IV.9 et Figure IV.10) ; en cliquant sur le bouton (Recherche WorNet) en peut comprendre tous les sens et explication du terme « estimate » (Voir Figure IV.11).

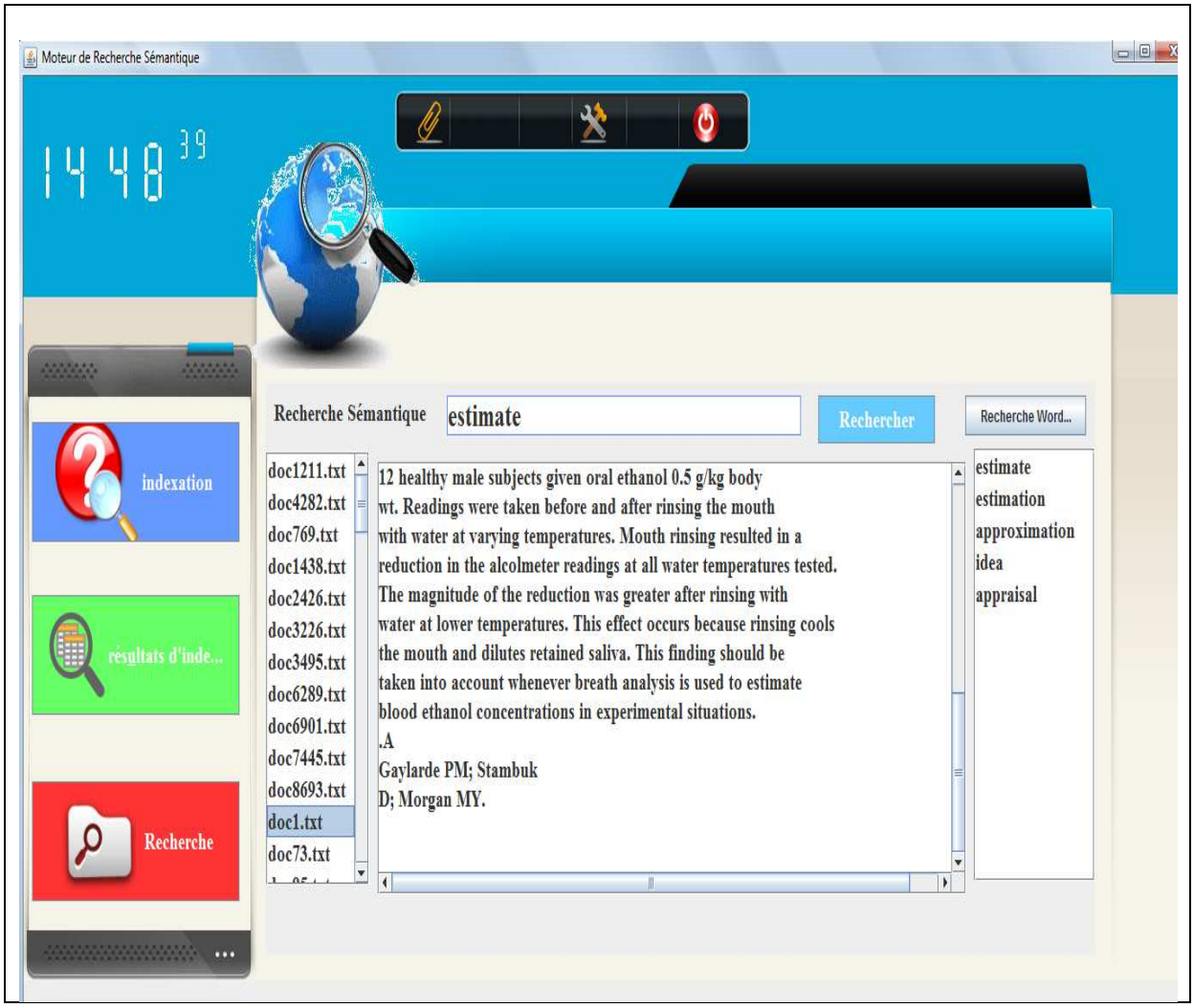

Figure IV.8. Afficher le contenu d'un document résultat sélectionné dans la liste de la requête « estimate ».

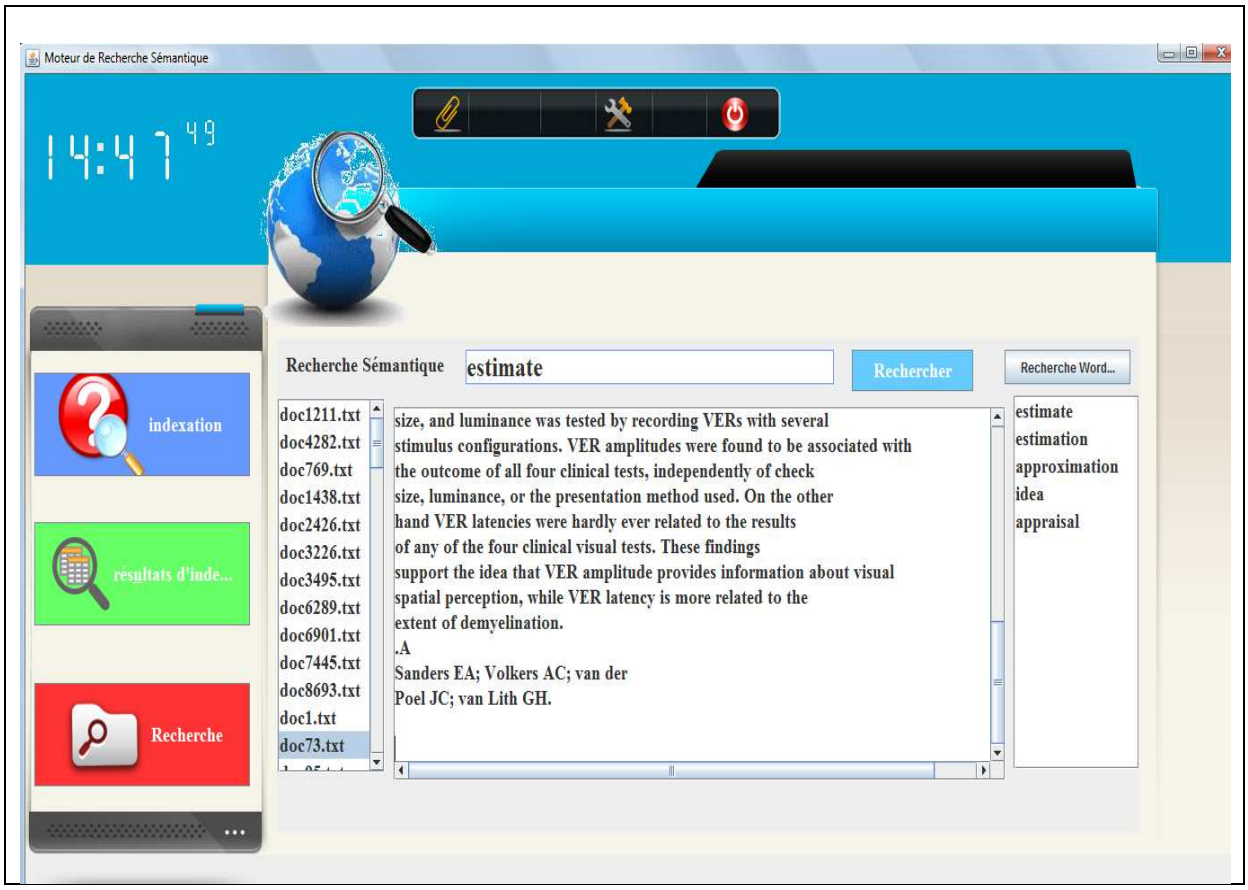

Figure IV.9. Afficher le contenu d'un document résultat sélectionné dans la liste qui contient le synset « idea » de la requête « estimate ».

Figure IV.10. Afficher le contenu d'un document résultat sélectionné dans la liste qui contient le synset « approximately » de la requête « estimate ».

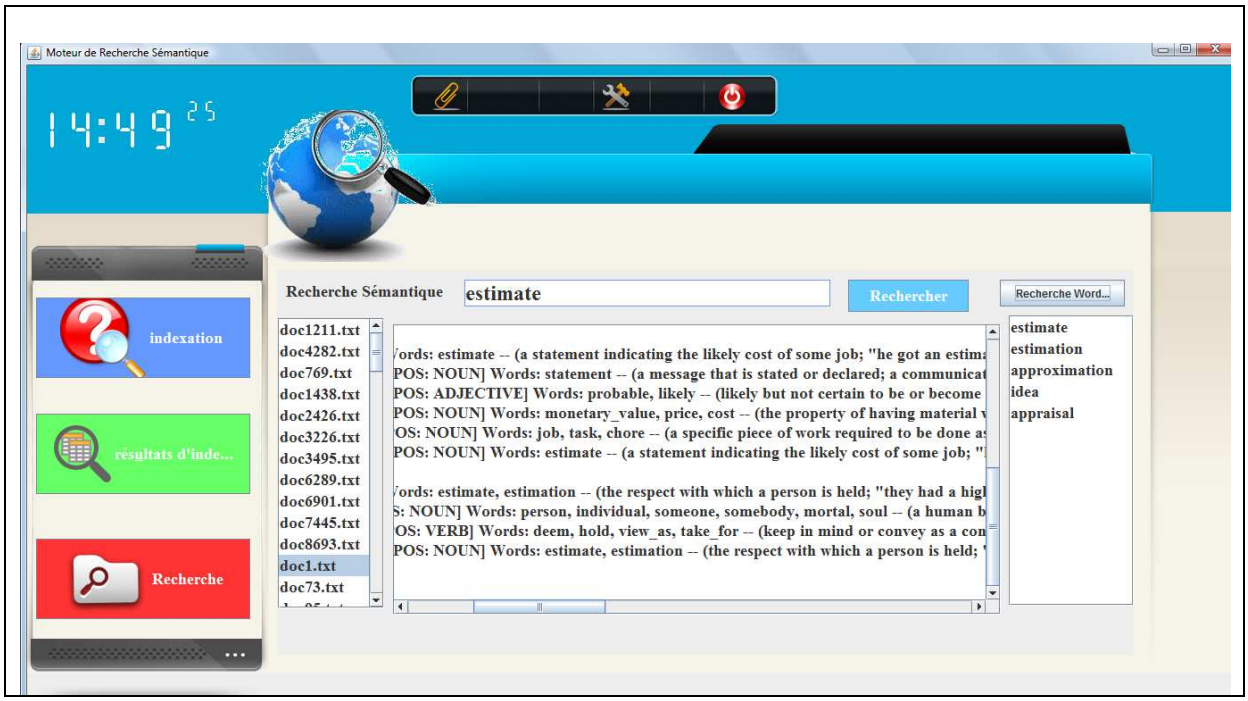

Figure IV.11. Explications sémantiques par WordNet du terme de la requête « estimate ».

Vu la durée d'indexation, de recherche et la quantité de documents retournés, l'utilisateur peut saisir le nombre des premiers documents à indexer ou à rechercher dans le corpus à travers la deuxième icône comme montre la Figure suivante.

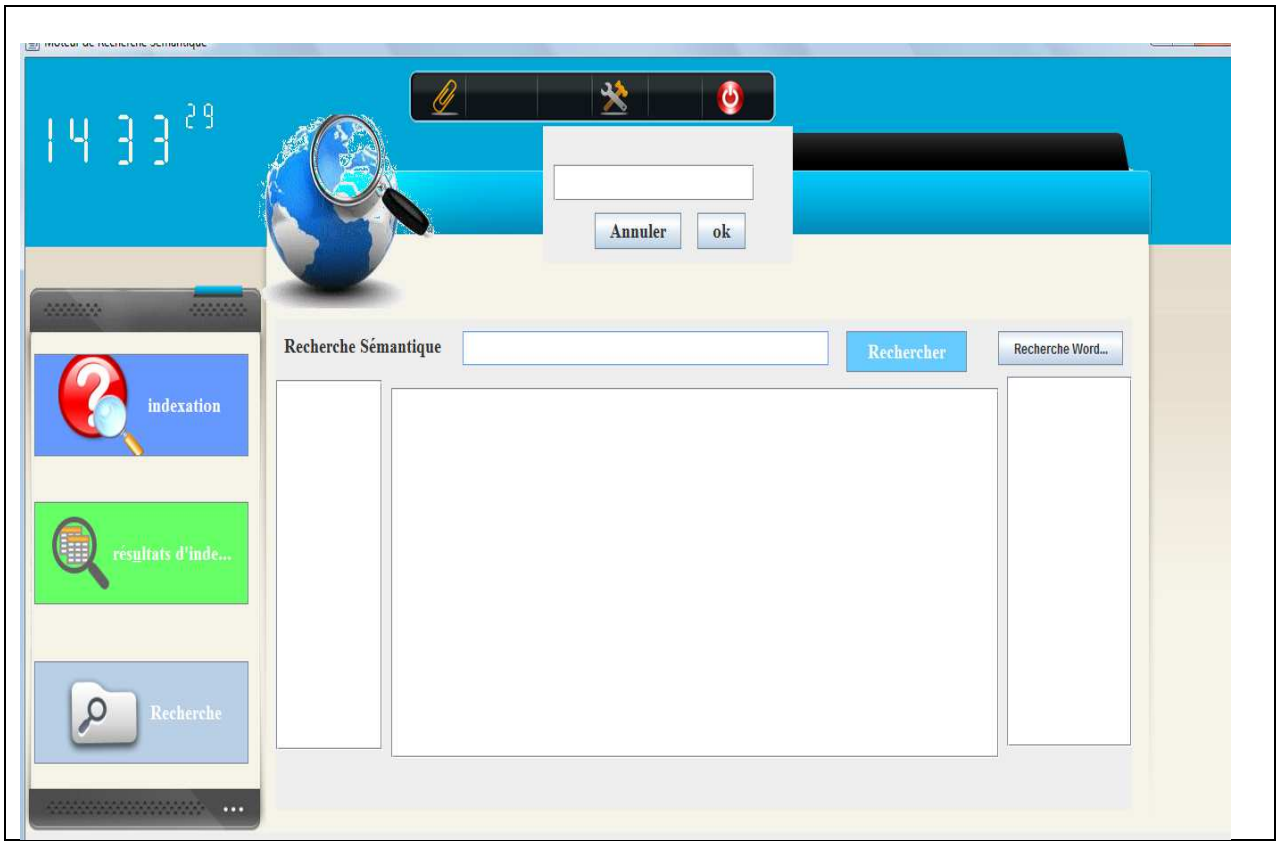

Figure IV.12. La saisie des premiers documents à indexer ou à rechercher.

Si le terme de la requête n'existe pas dans le corpus le système va afficher le message « Couldn't find ! »

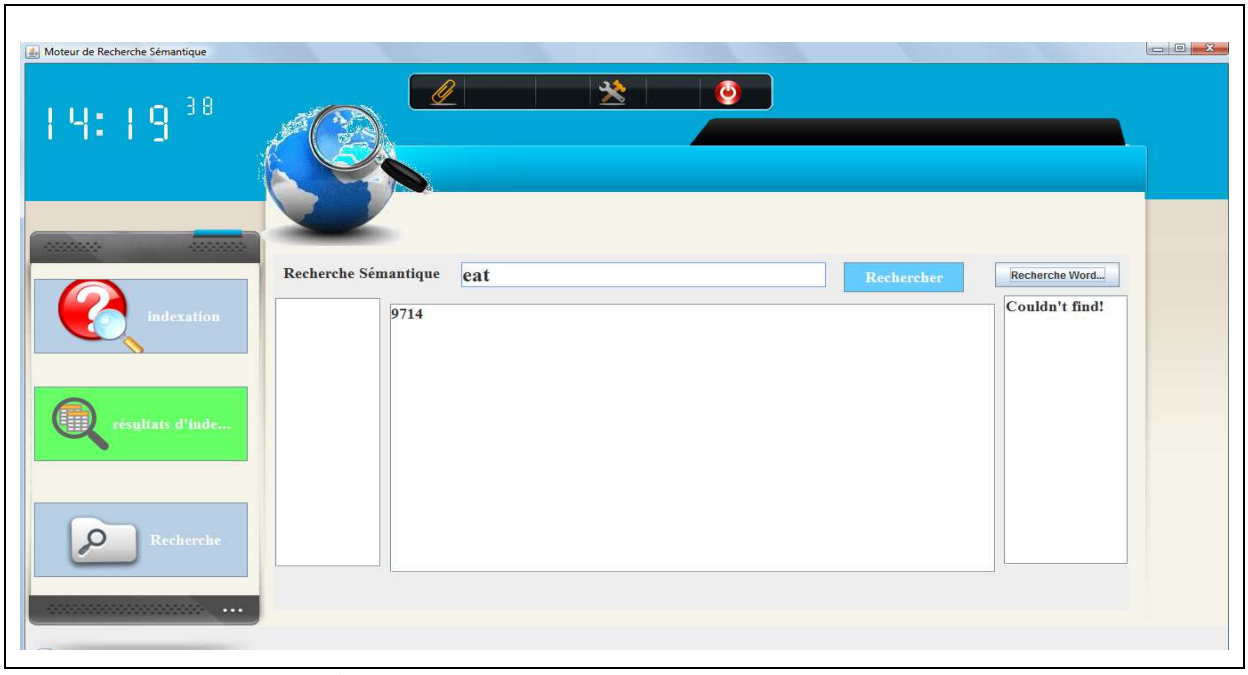

Figure IV.13. Le terme de la requête inexistant.

6. Conclusion

Ce chapitre a été consacré à la description conceptuelle de notre application. Différent outils logiciels et linguistiques ont été intégrés. Notre implémentation a pu mettre en œuvre plusieurs concepts et approches, qui paraissaient abstraits, dans une seule interface simplifiée. Les tests appliqués sur le corpus anglais sont encourageants et nous motivent à approfondir nos recherches dans ce domaine.

Cybergraphie

[33] http://www.axl.cefan.ulaval.ca/monde/anglais4.ModernE.htm, 12/03/2016.

[34] http://www.languageguide.org/english/grammar/fr/part1/nouns_r.jsp, 15/03/2016.

[35] http://www.languageguide.org/english/grammar/fr/part1/, 15/03/2016.

[36] https://fr.wikipedia.org/wiki/Grammaire_anglaise, 12/03/2016.

[48] http://www.clubic.com/telecharger-fiche384048-staruml.html, 25/04/2016.

[50] http://ipeti.forumpro.fr/t21-definition-de-langage-java-java-script, 18/05/2016.

[51] https://netbeans.org/index_fr.html, 18/05/2016.

[52] http://lucene.apache.org/, 18/05/2016.

Liste des Abréviations

RI Recherche d'information.

SRI Système de Recherche d'Information.

WSD Word Sens Disambiguation.

CACM Corpus from the Association of Computing Machinery journal.

SMART Self-Monitoring, Analysis, and Reporting Technology.

SPIRIT System Parametric Information Relational Intelligence Tool.

NEURODOC Plate-forme documentaire en neurosciences.

DIALOG Système destine à conserver avec un être humain, avec une structure cohérante.

MRD Machine Readable Dictionaries.

W3C World Wide Web Consortium.

RDF Resource Description Framework.

OWL Ontologie Web Language.

SPARQL Simple Protocol And RDF Query Language.

DVD Digital Versatile Disc.

CNN Cable News Network.

OACI Organisation de l'Aviation Civile Internationale.

UML Unified Modelling Language.

MDA Model Driven Architecture.

MO Mega Octet.

AMD Advanced Micro Devices.

GHZ Giga Hertz.

RAM Random Access Memory.

MS Micro Soft.

IDE Integrated Development Environment.

PHP	Hypertext PreProcessor.
JSP	Java Server Pages.
J2EE	Java 2 Enterprise Edition.
MacOS	Macintosh Operating System**.**
API	Application Programming Interfaces.

Liste des Tableaux

Liste des Figures

Liste des Figures

Liste des Tableaux

Liste des Abréviations

CACM	Corpus from the Association of Computing Machinery journal.
SMART	Self-Monitoring, Analysis, and Reporting Technology.
SPIRIT	System Parametric Information Relational Intelligence Tool.
NEURODOC	Plate-forme documentaire en neurosciences.
DIALOG	Système destine à conserver avec un être humain, avec une structure cohérante.
MRD	Machine Readable dictionaries.
W3C	Le World Wide Web Consortium.
RDF	Resource Description Framework.

OWL Ontologie Web Language.

SPARQL Simple Protocol And RDF Query Language.

YAGO Yet Another Great Ontology.

Conclusion et Perspectives

Conclusion

A travers les différents chapitres que nous avons présentés, nous concluons que la recherche d'information est un thème de recherche important en sciences de l'information. Elle peut porter sur plusieurs critères : le temps de réponse, la pertinence, la qualité et la présentation des résultats, etc. Le critère le plus important est celui qui mesure la capacité du système à satisfaire le besoin d'information d'un utilisateur, c'est-à-dire la pertinence des résultats.

Nos travaux développés dans ce mémoire s'inscrivent dans le cadre de l'indexation sémantique d'une collection de textes anglais.

Le problème de l'indexation basée sur l'approche statistique, en utilisant les termes simples et composés, est que le SRI ne prend en considération que les documents qu'ils partagent le maximum de mots-clés avec la requête. Cette méthode diminue la précision des SRI, il ne présente pas tous les documents pertinents de la collection.

C'est pourquoi, la représentation conceptuelle dans laquelle l'unité de vecteur serait un concept (groupe des synonymes appelé synsets), nous a permis de voir comment l'intégration d'une ressource externe WordNet a permis l'amélioration de la performance de notre SRI. Les éléments de cette représentation ne sont plus associés directement à de simples mots mais plutôt à des sysnsets.

Perspectives

Enfin, nous visons dans le cadre de nos travaux futures, d'enrichir notre système par plus de fonctionnalités pour la mission d'un moteur de recherche sémantique multilingue (Anglais, Arabe, Français et espagnol) qui prend en compte à la fois la sémantique et le contexte dans l'évaluation des SRI.

Il sera donc nécessaire de combiner plusieurs ressources à la fois ou d'intégrer « EuroWordNet » si ce dernier sera disponible. En tirant, pour chaque étape du processus d'évaluation, le principe de l'approche qui donne le meilleur résultat. L'évaluation sera par exemple faite en termes du temps de réponse et de la satisfaction des utilisateurs.

Somme toute, la combinaison des parties retenues de chaque langue donnera naissance à une nouvelle approche hybride qui utilise conjointement le contexte et la sémantique pour l'évaluation des SRI sur le web.

Bibliographie

[1] Salton.Gerard&McGill.Michael, "Introduction to modern information retrieval". McGraw Hill International Book Company, New York, 1983.

[2] Eric.Gaussier&Christian.Jacquemin&Pierre.Zweigenbaum, "Traitement automatique des langues et recherche d'information". Rapport de stage, Paris, 2003.

[3] Mustapha.Baziz, "Indexation conceptuelle guidée par ontologie pour la recherche d'information". Thèse de doctorat, Université Paul Sabatier de Toulouse, 2005.

[4] N.D.Y.Kompaoré, "Fusion de systèmes et analyse des caractéristiques linguistiques des requêtes: vers un processus de RI adaptatif". Thèse de doctorat, Université Paul Sabatier de Toulouse, 2008.

[5] P.Ingwersen, "Polyrepresentation of information needs and semantic entities: elements of a cognitive theory for information retrieval interaction". In proceedings of the ACM SIGIR international Seventeenth Annual Conference on research and development in information retrieval, pp. 101-110, 1994.

[6] J.Sinclair&R.Coulthard,"Towards an Analysis of Discourse".The English used by Teachers and Pupils,University Press, Oxford,1975.

[7] Souhila.Boucham, "Une approche basée Ontologies pour l'indexation automatique et la recherche d'information Multilingue". Mémoire de magister, Université M'hamed Bougara de Boumerdes, 2009.

[8] Nacira.Abbas, "Vers une Extension Sémantique de l'Analyse Formelle de Concepts : Application à la Recherche d'informations". Mémoire de magister, Université Mouloud Mammeri de Tizi-Ouzou, 03/07/2014.

[9] Alain.Berrendonner, "Grammaire pour un analyseur: aspects morphologiques". Document du travail du groupe de SYDO, Lyon, 1983.

[10] M.F.Porter, "An algorithm for suffix stripping". Program 14:130-137, 1980.

[11] P.Luhn, "The automatic creation of literature abstracts", pp. 159–165, April 1958.

[12] Karen.Spärck.Jones, "A statistical interpretation of term specificity and its application in retrieval". Journal of Documentation, Program 28:11–21, 1972.

[13] James.Callan&W.Croft&Stephen.Harding, "The inquery retrieval system". In Proceedings of the Third International Conference on Database and Expert Systems Applications, pp. 78–83, Springer-Verlag, 1992.

[14] C. Tambellini, "Un système de recherche d'information adapté aux données de incertaines: adaptation du modèle de langue". Thèse de doctorat, Université de Sophia Antipolis-UFR sciences, Nice, 2007.

[15] Abderrezak.Brahmi, "Contribution à la Recherche Intelligente sur le Web : Indexation Sémantique des Textes Non-Structurés". Thèse de doctorat, Oran, 2013.

[16] Singhal.A, "Modern Information Retrieval: A Brief Overview". Bulletin of the

IEEE Computer Society Technical Committee on Data Engineering, Vol. 24, pp. 35-43, 2001.

[17] Charhad, "Modèles de Documents Vidéo basés sur le Formalisme des Graphes Conceptuels pour l'Indexation et la Recherche par le Contenu Sémantique". Thèse de doctorat, pp. 24-25, Novembre 2005.

[18] Pascal.Hitzler&Markus.Krotzsch&Sebastien.Rudolph. "Foundations of semantic web technologies". CRC Press, pp. 8-11, 2009.

[19] Fatiha.Boubekeur-Amirouche, "Contribution à la définition de modèles de recherche d'information flexibles basés sur les CP-Nets". Thèse de doctorat, pp. 35-42, Université de Paul Sabatier de Toulouse, 2008.

[20] Fatiha.Boubekeur-Amirouche&Wassila.Azzoug, "Pondération des concepts en recherche d'information sémantique". Rapport de stage CORIA, pp. 441-450, 2013.

[21] E.Voorhees, "Using WordNet to Disambiguate Word Senses for Text Retrieval", Proceedings of the 16th Annual Conference on Research and Development in Information Retrieval, SIGIR'93, Pittsburgh, PA, 1993.

[22] R.Krovetz&W.B.Croft, "Lexical Ambiguity and Information Retrieval". In ACM Transactions on Information Systems, 10(1). 1992.

[23] M.Sanderson, "Word Sense Disambiguation and Information Retrieval", PhD Thesis, Technical Report (TR-1997-7) of the Department of Computing Science at the University of Glasgow, Glasgow G12 8QQ, UK, 1997.

[24] Boris.Katz&Özlem.Uzuner&Deniz.Yuret, "Word Sense Disambiguation for Information Retrieval", AAAI/IAAI 1999: 985. 1998.

[25] R.Mihalcea&D.Moldovan, "Semantic indexing using WordNet senses". In Proceedings of ACL Workshop on IR & NLP, Hong Kong, http://www.seas.smu.edu/~rada/papers/acl00.nlp_ir.ps.gz. October 2000.

[26] D.Yarowsky, "Hierarchical decision lists for word sense disambiguation. Journal Computers and the Humanities", Vol.34 (1-2), pp. 179-186, 2000.

[27] P.Pantel&D.Lin, "Discovering word senses from text". Proceedings of the 8th 619, ACMSIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 613, Canada 2002.

[28] R.Navigli, "Word Sense Disambiguation: A survey". ACM Computing Surveys, Vol. 41, No. 2, Article 10, 2009.

[29] Wassila.Azzoug, "Contribution à la définition d'une approche d'indexation sémantique de documents textuels". Mémoire de magister, 2012/2013.

[30] Bissan-Audeh, "Reformulation sémantique des requêtes pour la recherche d'information ad hoc sur le Web". Thèse de doctorat, 09 Septembre 2014.

[31] Reinout.Van-Rees, "Clarity in the usage of the terms ontology, taxonomy and classification". CIB REPORT, 2003.

[32] H.Kucera&W.N.Francis, "Computational Analysis of Present-Day American English". Brown University Press, 1967.

[37] G.A.Miller, "Nouns in WordNet : A lexical inheritance system". In : Five Papers onWordNet, http://www.cogsci.princeton.edu/ wn/ (sept. 1993), 10–25, revisedversion, 1997.

[38] Fatima.Ahmed-khaled&Rafik.Cherai, "Approche pour l'indexation conceptuelle basée sur les concepts et leurs concepts similaires pour la Recherche d'Information Arabe". Mémoire de master, Médéa, 2015.

[39] C.Fellbaum&D.Gross&K.Miller, "Adjectives in WordNet". In : Five Papers on WordNet, http://www.cogsci.princeton.edu/ wn/, (sept. 1997), 26–39, revised version,1998.

[40] Hadjira.Hachemi&Nour-El-Houda.Rimouche, "Moteur de recherche sémantique". Mémoire de master, Université Abou BakrBelkaid– Tlemcen, 2013.

[41] M.Silberztein, "Dictionnaires électroniques et analyse automatique de textes". Le système INTEX, Informatique linguistique, Masson, Paris, 1993.

[42] P.Srinivasan, "Thesaurus construction", in : Information Retrieval : Data Structures andAlgorithms, Frakes W. B., Baeza-Yates R., Prentice Hall, New Jersey, 1992.

[43] S.Johansson, "Some aspects of verb-adverbcombinations". In : The verb in contemporary English. Theory and description, Aarts B.,Meyer C. F., Cambridge University Press, Cambridge, 1995.

[44] J.S.Justeson&S.M.Katz, "Principled disambiguation : Discriminating adjective senseswith modified nouns". Computational Linguistics, 21, 1, 1995.

[45] R.Basili&M.Della-Rocca&M.T.Pazienza, "Contextual word sense tuning anddisambiguation". Applied Artificial Intelligence, 11, 1997.

[46] J.Gonzalo&F.Verdejo&I.Chugur&J.M.Cigarrán, "Indexing with WordNet synsets can improve Text Retrieval". CoRR, 1998.

[47] F.S.Touihr&I.Zitouni, "Conception et réalisation d'un site web dynamique pour la gestion des demandes d'assistances pour RTO (SONATRACH)". Mémoire de licence, Université de Mostaganem, 2013-2014.

[49] Z.Graja,"Développement d'une application web pour la réservation de billet de train". Mémoire de master, Université de Mostaganem, 2008-2009.