Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 2 - Jean Jaurès (UT2 **Jean-Jaurès**)

Cotutelle internationale : Université Mouloud Mammeri

Présentée et soutenue par

Samia ILTACHE

Le 18 novembre 2018

Modélisation ontologique pour la recherche d'informations: évaluation de la similarité sémantique de textes et application à la détection de plagiats.

École doctorale

EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse

Spécialité :

Informatique et Télécommunications

Unité de recherche

IRIT : Institut de Recherche en Informatique de Toulouse

Thèse dirigée par

Pierre-Jean CHARREL et Malik SI-MOHAMMED

Jury

M. Rachid AHMED OUAMER, Professeur, Université Mouloud Mammeri, Tizi-ouzou (Président du jury)
Mme Karima BENATCHBA, Professeur, Ecole Nationale Supérieure d'Informatique, Alger (Rapporteure)
M. Pierre-Jean CHARREL, Professeur, Université Toulouse - Jean Jaurès (Co-directeur de thèse)
Mme Catherine COMPAROT, Maître de Conférences, Université Toulouse - Jean Jaurès (Encadrante)
M. Christian SALLABERRY, Maître de Conférences, HDR, Université de Pau et des pays de l'Adour (Rapporteur)
M. Malik SI-MOHAMMED, Professeur, Université Mouloud Mammeri, Tizi-ouzou (Co-directeur de thèse)

Remerciements

Mes remerciements vont d'abord à mes deux directeurs de thèse, les Professeurs Malik Si Mohammed et Pierre-Jean Charrel, pour m'avoir accueillie dans leurs équipes respectives et pour m'avoir donné la chance de réaliser ce projet de thèse. Je les remercie particulièrement pour m'avoir fait confiance en m'accordant une liberté et une autonomie dans mon travail. Ils ont su orienter mes travaux et enrichir mes propositions grâce à leurs suggestions et leurs remarques pertinentes. Je remercie M. Malik Si Mohammed, qui, souvent, a su trouver les mots pour me redonner confiance dans les moments de doute et de découragement. Je n'oublie pas le soutien de M. Pierre-Jean Charrel pour résoudre tous les problèmes administratifs.

Je remercie Mme Catherine Comparot, Maître de Conférences à l'UT2J, d'avoir accepté de co-encadrer ce travail, pour les échanges que nous avons eu durant ces années de thèse malgré l'éloignement, ses remarques et suggestions et pour ses relectures attentives qui ont permis d'améliorer la qualité de ce mémoire.

Je remercie Mme Karima Benatchba, Professeur à l'ESI, et M. Christian Sallaberry, Maître de Conférences, HDR à L'UPPA, pour la lecture et la correction de mon manuscrit et pour l'intérêt qu'ils ont porté à mon travail. Je remercie également M. Rachid Ahmed Ouamer, Professeur à l'UMMTO, qui me fait l'honneur de présider mon jury.

Un grand merci à M. Mohamed Ibazizen, pour sa gentillesse et pour m'avoir donné toutes les explications nécessaires pour réaliser mes tests statistiques.

Merci à M. Hocine Fellag pour avoir répondu à mes questions sur les tests statistiques, à Mme Fatiha Amirouche pour ses éclaircissements relatifs à certains outils et au père Vincent Kyererezi pour les corrections de l'anglais.

Merci à mes amis, Samia Ait Adda, Ikram Allam, Massinissa Sekhi, Said Talbi, Nabil Amirouche, Mahieddine Mokhtari, pour leur soutien, leur aide et leur disponibilité. Chacun de vous, à sa manière, m'a permis de mieux vivre ce projet de thèse en m'apportant du réconfort pendant les moments de doute.

Merci à mes "anges gardiens", toutes ces personnes, qui n'ont pas forcément un lien avec cette thèse, parfois inconnues, que je croise sur mon chemin et qui permettent de surmonter certains obstacles, rendant la vie plus facile et plus agréable.

Merci à ma famille pour leur soutien, particulièrement ma sœur Wahiba qui est toujours là pour moi. Merci à ma mère, ma compagne des bons et mauvais moments. Sa présence à mes côtés m'apporte force et réconfort et me permet de sentir que tout est surmontable.

Résumé

L'expansion du web et le développement des technologies de l'information ont contribué à la prolifération des documents numériques en ligne. Cette disponibilité de l'information présente l'avantage de rendre la connaissance accessible à tous mais soulève de nombreux problèmes quant à l'accès à l'information pertinente, répondant à un besoin utilisateur. Un premier problème est lié à l'extraction de l'information utile parmi celle qui est disponible. Un second problème concerne l'appropriation de ces connaissances qui parfois, se traduit par du plagiat.

L'objectif de cette thèse est le développement d'un modèle permettant de mieux caractériser les documents afin d'en faciliter l'accès mais aussi de détecter ceux présentant un risque de plagiat. Ce modèle s'appuie sur des ontologies de domaine pour la classification des documents et pour le calcul de la similarité des documents appartenant à un même domaine. Nous nous intéressons plus spécifiquement aux articles scientifiques, et notamment à leurs résumés, textes courts, relativement structurés, et en principe suffisamment complets pour permettre à une communauté de lecteurs de juger de leur contenu. Il s'agit dès lors de déterminer comment évaluer la proximité/similarité sémantique de deux articles à travers l'examen de leurs résumés respectifs. Considérant qu'une ontologie de domaine regroupe les connaissances relatives à un domaine scientifique donné, notre processus est basé sur deux actions :

(i) Une classification automatique des documents dans un domaine choisi parmi plusieurs domaines candidats. Cette classification est sémantique. Elle détermine le sens d'un document à partir du contexte global (contexte défini par le contenu de tout le document) dans lequel s'inscrit son contenu. La classification telle que nous l'avons définie, permet de représenter le résumé sous forme d'un graphe conceptuel construit à partir de l'ontologie du domaine à laquelle il est rattaché.

(ii) Une comparaison des textes réalisée sur la base d'un enrichissement des graphes des résumés. L'enrichissement se fait en deux étapes : Un premier enrichissement est réalisé lors de la construction de ce que nous appelons le périmètre sémantique de chaque résumé sur la base des concepts initiaux de son graphe construit à l'étape de classification. Un enrichissement mutuel est ensuite effectué lors de la comparaison des graphes des résumés. L'évaluation de la similarité entre deux résumés est réalisée sur la base de l'enrichissement et l'appariement de leurs graphes. Plus les graphes sont proches sémantiquement, plus le risque de plagiat est important. La comparaison sémantique des résumés s'appuie sur une segmentation de leur contenu respectif en zones, unités documentaires, reflétant leur structure logique. C'est sur la comparaison des graphes conceptuels des zones analogues que le calcul de la similarité des résumés s'appuie.

Notre approche a été évaluée et comparée aux approches conventionnelles grâce aux applications que nous avons développées. Les résultats obtenus montrent l'intérêt de nos propositions pour la classification sémantique des documents d'une part et pour le calcul de la similarité des textes d'autre part.

Abstract

The expansion of the web and the development of different information technologies have contributed to the proliferation of digital documents online. This availability of information has the advantage of making knowledge accessible to all. However, many problems emerged regarding access to relevant information that meets a user's need. The first problem is related to the extraction of the useful available information. A second problem concerns the use of this knowledge which sometimes results in plagiarism.

The aim of this thesis is the development of a model that better characterizes documents to facilitate their access and also to detect those with a risk of plagiarism. This model is based on domain ontologies for the classification of documents and for calculating the similarity of documents belonging to the same domain as well. We are particularly interested in scientific papers, specifically their abstracts, short texts that are relatively well structured and normally provide enough knowledge to allow a community of readers to assess the content of the associated scientific papers. The problem is, therefore, to determine how to assess the semantic proximity/similarity of two papers by examining their respective abstracts. Forasmuch as the domain ontology provides a useful way to represent knowledge relative to a given domain, our process is based on two actions:

(i) An automatic classification of documents in a domain selected from several candidate domains. This classification is semantic. It determines the meaning of a document from the global context (context defined by the content of the whole document) in which its content is used. The classification as we defined it makes it possible to represent the abstract in the form of a conceptual graph constructed from the ontology of the domain to which it is attached.

(ii) A comparison of the texts performed on the basis of an enrichment of the graphs of the abstracts. The enrichment is done in two steps: A first one is realized during the construction of the semantic perimeter of each abstract based on the initial concepts of its graph which is built in the classification step. Mutual enrichment is then performed when comparing the graphs of the abstracts. The evaluation of the similarity between two abstracts is performed on the basis of the enrichment and the matching of their graphs. The more semantically close the graphs, the greater the risk of plagiarism. The semantic comparison of the abstracts is based on a segmentation of their respective content into zones, documentary units, reflecting their logical structure. It is on the comparison of the conceptual graphs of the zones playing the same role that the calculation of the similarity of the abstracts relies.

Our approach has been evaluated and compared to conventional approaches using the applications we have developed. The results obtained show the interest of our proposals for the semantic classification of documents on the one hand and for the calculation of the similarity of texts on the other hand.

Table des matières

Liste des Figures

Chapitre 1. Représentation des connaissances

Chapitre 2. Similarité des textes

Chapitre 3. Classification sémantique basée sur des ontologies

Chapitre 4. Similarité des textes : application aux résumés des articles scientifiques

Chapitre 5. Architecture des processus

Liste des Tables

Introduction générale

Chapitre 1. Représentation des connaissances

Chapitre 3. Classification sémantique basée sur des ontologies

Chapitre 4. Similarité des textes : application aux résumés des articles scientifiques

Introduction générale

Durant toute son existence, l'être humain n'a cessé de réfléchir et d'innover. Des recherches dans des domaines touchant diverses spécialités ont proliféré produisant de nouvelles connaissances. La masse des informations qui, pendant longtemps étaient conservées dans des encyclopédies, a considérablement augmenté. L'apparition de l'ordinateur et des nouvelles technologies associées au Web ont largement contribué à leur diffusion sous forme de livres, d'articles etc. Cela a soulevé de nouvelles problématiques liées au stockage, à la mise en ligne et à l'échange d'information. Plusieurs domaines de réflexion sur la façon d'organiser et d'extraire des informations pertinentes relativement aux besoins des utilisateurs ont vu le jour, tels que la classification automatique des documents et la recherche d'information.

La majorité des travaux existant dans la littérature propose une représentation vectorielle de contenu des documents. Les descripteurs des documents sont des mots indépendants les uns des autres. Pour comparer et faire des rapprochements entre documents, des comparaisons morphologiques entre ces mots sont réalisées. Cette représentation suppose que deux documents sont "proches" s'ils partagent les mêmes mots ou qu'un grand nombre de mots sont identiques. Cependant, des auteurs peuvent évoquer des sujets similaires sans pour autant utiliser les mêmes mots, ni le même niveau de détails. De plus, la présence des mots synonymes et polysémiques dans les langues naturelles engendre souvent des erreurs de rapprochement.

Contexte de travail et motivation

Surmonter les limites des approches représentant les documents par des mots est l'objet de nombreux travaux visant à donner une représentation sémantique de leur contenu. C'est dans ce contexte que s'inscrit notre travail. La disponibilité des informations, si elle présente l'avantage de rendre la connaissance accessible à tous, soulève de nombreux problèmes, notamment l'appropriation de ces connaissances conduisant dans certains cas au plagiat. Nous nous plaçons donc dans le contexte de l'évaluation de la similarité des textes basée sur l'exploitation de ressources sémantiques telles que des thésaurus et des ontologies et nous l'appliquons à la détection de plagiat dans les articles scientifiques.

Une ontologie est représentée par des concepts permettant d'associer un sens commun à des mots ayant des formes différentes. Notre approche vise à donner une représentation sémantique des documents en représentant le "sens" du document par un graphe conceptuel issu d'une ontologie. Ainsi, le rapprochement entre documents ne se fait plus au niveau des mots mais au niveau des thèmes décrits dans leur contenu.

Problématique

L'utilisateur ne connaissant pas a priori le contenu des documents disponibles sur le web, effectue des recherches en spécifiant quelques mots ou quelques phrases en fonction de ses connaissances sur ce qu'il recherche. De nombreux documents peuvent aborder des thématiques proches du sujet recherché en décrivant diverses connaissances méconnues de l'utilisateur et donc utiliser des mots non spécifiés explicitement par son besoin. Avec les représentations des documents basées sur les mots, ces documents ne seront pas retournés à l'utilisateur. L'utilisation des mots pour représenter le contenu d'un document pose plusieurs problèmes :

- Un premier problème concerne la polysémie présente dans la langue. Un même mot peut avoir plusieurs sens selon le domaine de connaissance où il est défini. De plus, au sein d'un même domaine, un mot peut avoir plusieurs sens. Prenons par exemple un résumé d'un article scientifique extrait du domaine *music* de notre corpus.

Similarity is an important concept in music cognition research since the similarity between (parts of) musical pieces determines perception of stylistic categories and structural relationships between parts of musical works. The purpose of the present research is to develop and test models of musical similarity perception inspired by a transformational approach which conceives of similarity between two perceptual objects in terms of the complexity of the cognitive operations required to transform the representation of the first object into that of the second, a process which has been formulated in information-theoretic terms. Specifically, computational simulations are developed based on compression distance in which a probabilistic model is trained on one piece of music and then used to predict, or compress, the notes in a second piece. The more predictable the second piece according to the model, the more efficiently it can be encoded and the greater the similarity between the two pieces. The present research extends an existing information-theoretic model of auditory expectation (IDyOM) to compute compression distances varying in symmetry and normalisation using high-level symbolic features representing aspects of pitch and rhythmic structure. Comparing these compression distances with listeners' similarity ratings between pairs of melodies collected in three experiments demonstrates that the compression-based model provides a good fit to the data and allows the identification of representations, model parameters and compression-based metrics that best account for musical similarity perception. The compression-based model also shows comparable performance to the best-performing algorithms on the MIREX 2005 melodic similarity task.

De ce résumé, nous prenons trois mots en exemple et donnons les sens qu'ils peuvent avoir dans quelques domaines. Ces sens sont extraits de la ressource WordNet et représentés dans la Table 1.

Terme	Domaine	Sens (WordNet)
Music	Music	06591368-n 00515842-n 05387226-n
Melody	Music	06598312-n 05381203-n
Operation	Computer_science Mathematics Enterprise	12761088-n 00817792-n 01033016-n
Pitch	Music Chemistry Buildings	04724487-n 14063488-n 14063488-n

Table 1 Sens des termes dans différents domaines

Le terme *music* possède trois sens dans le domaine *music*. Le mot pitch possède un sens dans les domaines *music*, *chemistry* et *buildings* Cette ambiguïté nécessite un processus de sélection du sens approprié des mots.

- Un second problème concerne les mots à retenir pour représenter un document. En effet, tous les termes d'un document n'ont pas un sens pour un domaine donné. C'est le cas, par exemple pour le terme *operation* qui ne possède aucun sens pour le domaine *music*. La question est alors comment déterminer les mots à retenir pour représenter un document.

- Un troisième problème concerne la comparaison morphologique des mots. Prenons par exemple les phrases suivantes extraites de quatre résumés d'articles scientifiques relatifs au domaine de la *classification des documents* :

Ph1 : *We cluster the documents by a standard partitional algorithm.*

Ph2 : *we compute multiple clustering results using KMeans.*

Ph3 : *Our experimental evaluation on Reuters newsfeeds.*

Ph4 : *We evaluate our methods based on F-Score on standard datasets…*

Si nous comparons les phrases *ph1* et *ph2* sur la base des mots, il n'est pas possible de retrouver un lien entre *standard partitional algorithm* et *Kmeans*. Il en est de même pour les mots *Reuters newsfeeds* et *standard datasets.* Ces mots ne sont pas synonymes mais un lien sémantique existe entre ces mots, décelable par un lecteur ayant des connaissances dans le domaine de la *classification des documents.* Extraire la sémantique des textes ne doit pas alors se limiter à l'attribution des sens aux mots.

- Au-delà des sens des termes, nous nous intéressons au sens du document. Dans un texte, un auteur aborde différentes notions dans une structure implicite comme c'est le cas pour les résumés d'articles scientifiques. Considérons le texte d'un résumé scientifique extrait de notre corpus :

"Text document clustering plays an important role in providing intuitive navigation and browsing mechanisms by organizing large amounts of information into a small number of meaningful clusters. The bag of words representation used for these clustering methods is often unsatisfactory as it ignores relationships between important terms that do not co-occur literally. In order to deal with the problem, we integrate background knowledge, in our application WordNet, into the process of clustering text documents. We cluster the documents by a standard partitional algorithm. Our experimental evaluation on Reuters newsfeeds compares clustering results with pre-categorizations of news. In the experiments, improvements of results by background knowledge compared to the baseline can be shown for many interesting tasks."

Dans ce résumé, l'auteur définit plusieurs parties :

- Le domaine de recherche dans lequel s'inscrit son travail : " *Text document clustering plays an important role in providing .... "*

- *S*es principales contributions : "*…, we integrate background knowledge, in our application WordNet, into the process of clustering text documents. We cluster the documents by a standard partitional algorithm* "

- Ses domaines d'application : ". *Our experimental evaluation on Reuters newsfeeds."*

- Sa conclusion sur les résultats obtenus par ses expérimentations : " *In the experiments, improvements of results by background knowledge compared to the baseline can be shown for many interesting tasks."*

Si pour un lecteur humain, il est facile de distinguer les différentes parties composant un résumé, il n'en est pas de même pour un automate. Il est alors nécessaire d'utiliser des ressources permettant d'extraire, de façon automatique, la sémantique contenue dans un texte.

Une ontologie permet de regrouper les mots décrivant une même idée autour d'un même concept. De plus les ontologies ont pour objectif de représenter toute la connaissance d'un domaine spécifique. Utiliser des informations extraites de ces ressources sémantiques externes pour donner une représentation sémantique des documents pose plusieurs problématiques :

- Comment extraire la sémantique décrite dans un texte ? Ce point soulève plusieurs questions :
  - Comment retrouver des liens entre les mots au-delà de la synonymie ?
  - Quels concepts et quelles relations extraire d'une ressource sémantique pour représenter le contenu des documents?
  - Comment extraire la fonction de chaque partie du contenu d'un document?
- Comment utiliser cette représentation sémantique pour faire des rapprochements entre des documents qui traitent de sujets similaires sans pour autant utiliser les mêmes mots ?

Contribution

L'objectif de cette thèse est d'utiliser les ontologies pour mettre en œuvre et exploiter une représentation sémantique des documents. Notre contribution dans le contexte de l'évaluation de la similarité des documents nous a amené à proposer une représentation d'un document par une information non explicitement citée dans son contenu. Nous citons ci-dessous les différentes contributions abordées dans cette thèse :

- Nous proposons d'utiliser les ontologies dans une approche multi-niveaux pour représenter la sémantique des documents.

- Nous proposons un processus de désambigüisation permettant de retrouver le sens des mots ambigus basée sur le voisinage le plus proche.

- Nous proposons de mettre en évidence une similarité entre les documents en mettant en avant le contexte général dans lequel ils s'inscrivent. Cette similarité est calculée par un processus de classification réalisé à partir des ontologies de domaine. Le classifieur permet de regrouper les documents partageant des connaissances similaires ou proches et de les rattacher aux ontologies de domaine appropriées.

- Nous utilisons le résultat de notre classifieur sémantique pour calculer une similarité "locale" entre les documents rattachés à une même ontologie de domaine. Nos principales contributions à ce niveau sont l'introduction de la notion de "périmètre sémantique" d'un document et l'enrichissement de son graphe que nous appliquons pour représenter les informations non explicitement citées dans le texte du document. L'enrichissement se fait en deux étapes :

    - Nous introduisons la notion de périmètre sémantique. A partir du processus de classification, un document est représenté par un graphe construit à partir de l'ontologie de domaine qui représente le mieux son contenu. Ce graphe contient les concepts initiaux correspondant aux termes explicitement cités dans le document. A partir de ce graphe, nous extrayons de l'ontologie de domaine des concepts liés sémantiquement aux concepts initiaux. L'ensemble de ces concepts forment le périmètre sémantique.

  - Le calcul de la similarité entre deux documents est basé sur la comparaison de leurs graphes. Pour comparer deux documents, nous appliquons un enrichissement mutuel en ajoutant un ensemble de concepts et de relations entre concepts à leurs graphes.

- Nous appliquons notre processus de calcul de la similarité des textes pour détecter des risques de plagiat. Nous nous intéressons particulièrement aux articles scientifiques et plus spécifiquement à leurs résumés. Alors que la plupart des approches de détection de plagiat recherche la similarité au niveau mots et au niveau phrase, nous proposons de diviser le texte plusieurs parties distinctes, que nous appelons zones, dont les rôles descriptifs sont différents. La comparaison de deux résumés d'articles scientifique se fait en comparant les zones de même type.
- Nous proposons plusieurs mesures :
  - Une mesure permettant de calculer le poids des mots explicitement et implicitement cités dans le contenu des documents.
  - Une mesure de similarité permettant de calculer la similarité entre deux textes. Cette mesure est utilisée pour calculer la similarité partielle entre deux unités documentaires appartenant à la même zone lors du calcul de la similarité des résumés scientifiques.
  - Une mesure combinant les similarités partielles pour calculer une similarité globale entre deux résumés.

Afin de vérifier l'apport de nos propositions par rapport aux approches existantes dans la littérature, nous avons évalué notre approche sur deux collections de résumés d'articles scientifiques et avons comparé les résultats obtenus avec ceux des approches conventionnelles. Le corpus utilisé pour évaluer notre classifieur est composé de résumés d'articles scientifiques répartis sur 10 domaines. Le deuxième corpus, utilisé pour évaluer la similarité des résumés appartenant à un domaine de connaissance, est composé de résumés d'articles scientifiques relatifs au domaine de la *classification des documents*.

Organisation de la thèse

Cette thèse est organisée en deux parties. La première partie est composée de deux chapitres sur l'état de l'art en relation avec le contexte dans lequel s'inscrivent nos travaux. La deuxième partie regroupe trois chapitres décrivant les différents traitements composant notre processus de calcul de similarité des textes.

Le chapitre 1 couvre un ensemble de notions relatives à la représentation des connaissances, notamment celles contenues dans les documents. Nous commençons par définir les notions de donnée, d'information et de connaissance en spécifiant la frontière qui les séparent et donnons un aperçu des différentes ressources constituant un support des connaissances. Nous nous attardons sur les ressources sémantiques sur lesquelles s'appuie notre travail, à savoir WordNet, WordNet Domains et les ontologies de domaine. Nous

décrivons les différentes formes des descripteurs de documents et nous détaillons les différentes approches d'extraction des termes à partir des documents. Nous abordons le problème d'ambiguïté des mots qui, jusqu'à aujourd'hui, suscite l'intérêt de nombreux travaux. Nous décrivons ensuite plusieurs approches de désambiguïsation basées sur un corpus et sur des ressources sémantiques.

Le chapitre 2 est dédié à l'état de l'art sur le calcul de la similarité des textes. La similarité peut être évaluée à plusieurs niveaux du document. Nous présentons d'abord la similarité entre mots et entre documents et nous citons ensuite des domaines de recherche exploitant ces différentes similarités. Pour chaque niveau de similarité nous détaillons un ensemble d'approches. Ces approchent se divisent en deux classes : des approches conventionnelles basées sur le contenu de document et sur un corpus d'une part et des approches exploitant des ressources sémantiques d'autre part.

La deuxième partie de la thèse concerne notre contribution à l'évaluation de la similarité sémantique des documents.

Dans le chapitre 3, nous présentons notre classifieur sémantique des documents basé sur des ontologies de domaines. Nous décrivons les différentes étapes composant le processus. Ce chapitre offre aussi une synthèse des résultats obtenus avec notre classifieur et des comparaisons effectuées avec des classifieurs conventionnels.

Le chapitre 4 donne une description détaillée du processus de calcul de la similarité locale entre documents. Les différentes étapes allant de la construction du graphe initial du document jusqu'à son enrichissement sont détaillées. Nous présentons la notion de périmètre sémantique et expliquons comment diviser le texte en plusieurs parties distinctes et comment affiner le calcul de similarité des textes basé sur cette décomposition. Ce chapitre se termine par une présentation des résultats de l'évaluation de notre approche et une comparaison de nos résultats avec d'autres approches menées sur un corpus de résumés d'articles scientifiques et sur une ontologie de domaine que nous avons construite.

Dans le chapitre 5, nous présentons des éléments d'architecture pour la mise en œuvre des processus décrits dans le chapitre 3 et le chapitre 4.

Pour finir, nous présentons un bilan des travaux réalisés et des résultats obtenus. Nous concluons sur l'intérêt de nos différentes propositions et nous donnons les perspectives envisagées pour la suite de nos travaux.

Partie 1

Etat de l’art

Chapitre 1
Représentation des connaissances

1.1 Introduction

La représentation classique des documents s'appuie sur une représentation vectorielle où les dimensions correspondent aux mots les plus représentatifs extraits de leur contenu. Cette représentation suppose l'indépendance de mots et seule la morphologie des mots est mise en avant. Les mots au sein des vecteurs sont dépourvus de sens.

Cette représentation soulève plusieurs problématiques relatives à la prise en compte du sens de mots et à la gestion de la synonymie et de la polysémie, présentes dans les langages. Pour surmonter ces problèmes, une représentation sémantique des documents s'avère nécessaire. Plusieurs approches se sont intéressées à cette problématique.

Pour comprendre comment est représentée la sémantique dans le contenu des documents et comment l'extraire pour l'exploiter de façon automatique, il est important de définir la notion de connaissance.

1.2 Donnée, information, connaissance

Les connaissances sont étudiées par plusieurs disciplines notamment la philosophie, la psychologie et les sciences cognitives. Dans le domaine de l'informatique, domaine du traitement de l'information, nous retrouvons souvent les mots "données" et "information". Dans le domaine de l'intelligence artificielle, on parle beaucoup plus de "connaissances".

- Une **donnée** est une suite de symboles (lettre, chiffre, bit). Une donnée est dépourvue de sens et elle constitue une description élémentaire d'une réalité. Transmise à un système, elle est traitée et modifiée en acquérant un sens, produisant ainsi une information.

- L'**information** désigne à la fois le message à communiquer et les symboles utilisés pour le représenter. L'information est immatérielle. Elle peut être consignée directement ou pas sur

un support matériel qui prend alors la valeur de document[1]. Une information correspond à une donnée contextualisée[2].

- **Connaissance** : Plusieurs auteurs différencient les connaissances tacites des connaissances explicites.

Les connaissances tacites regroupent les compétences innées ou acquises, le savoir-faire et l'expérience, et sont généralement difficiles à formaliser. Pour **[Nonaka et al, 1997]**, les connaissances tacites dépendent de l'expérience d'un individu, de ses idéaux, de ses valeurs et de ses émotions. Elles englobent ses aptitudes, ses croyances et ses perceptions.

Les connaissances explicites sont les connaissances transcrites sur un document ou dans un système informatique. Les connaissances explicites codifiées sont des connaissances énoncées dans un langage formel, inscrites sur un support permettant leur stockage, leur transmission et leur vérification, **[Partha et al., 1994].**

En gestion des connaissances, une connaissance correspond à l'appropriation et l'interprétation des informations par des êtres humains[3].

Pour Bachimont **[Bachimont, 2004]**, une connaissance est la capacité d'exercer une action pour atteindre un but. Nous pouvons dire alors que la connaissance est l'utilisation de l'information pour effectuer des raisonnements et des traitements.

Nous donnons un exemple pour chacune des notions définies ci dessus.

Camille pèse 80 kg est une donnée.

Camille est en surpoids est une information.

Camille doit faire du sport pour perdre du poids est une connaissance.

Les systèmes informatiques doivent intégrer les connaissances pour répondre aux divers besoins des utilisateurs. Cette intégration implique des processus de modélisation et d'acquisition de ces connaissances.

Acquérir les connaissances explicites n'est pas une tâche facile. Il s'agit de définir des modèles et supports pour expliciter et représenter ces connaissances dans un premier temps et de définir des méthodes permettant de les capturer pour les utiliser par la suite dans diverses applications.

[1] https://fr.wikipedia.org/wiki/Information
[2] https://fr.wikipedia.org/wiki/Connaissance
[3] https://fr.wikipedia.org/wiki/Connaissance

1.3 Représentation de la connaissance

Les connaissances peuvent être explicitées dans différents supports et structures. Nous considérons que la première ressource concentrant une quantité importante de connaissance est le document. Nous nous intéressons dans ce qui suit au document textuel auquel nous faisons référence indifféremment par les termes "document" ou "texte".

Les documents sont utilisés pour extraire les termes de leur contenu afin de construire des structures élaborées telles que des thésaurus, des ressources terminologiques et des ontologies. Les documents sont exploités également par des applications pour extraire les termes représentant des connaissances par le biais des ressources sémantiques afin de donner une représentation de ces documents exploitable par la machine. Ces deux tâches ont soulevé des problématiques différentes sur lesquelles plusieurs auteurs se sont penchés.

1.3.1 Corpus de textes

Un texte contient généralement la connaissance et le savoir de l'homme. Cette connaissance est sous forme brute et non structurée. Les textes sont souvent regroupés au sein d'un corpus selon certains critères afin d'extraire des connaissances pouvant répondre à des besoins particuliers.

François Rastier définit le texte comme "une suite linguistique autonome (orale ou écrite) constituant une unité empirique, et produite par un ou plusieurs énonciateurs dans une pratique sociale attestée" **[Rastier, 2001].**

Un corpus de textes est une collection de documents textuels. "La sélection d'un corpus de textes peut être faite pour une application ou une tâche particulière" **[Feldman et al, 2007].**

McEnery **[McEnery et al, 1996]** définit un corpus comme " une collection finie de textes exploitable par une machine, sélectionnée pour représenter de façon exhaustive une langue ou une thématique".

Bowker **[Bowker et al, 2002]** donne la définition suivante : " Un corpus peut être décrit comme une large collection de textes authentiques, regroupés sous une forme électronique selon un ensemble spécifique de critères ".

Les textes constituant un corpus devraient en effet être représentatifs de la langue et doivent être sélectionnés selon des critères explicites pour constituer un échantillon couvrant les thèmes à décrire. La taille des échantillons est la préoccupation de plusieurs auteurs. Certains préconisent d'utiliser des textes entiers car les termes importants peuvent apparaître à n'importe quel niveau du texte **[Sinclair, 1991][Pearson, 1998]**, alors que d'autres préfèrent utiliser seulement des parties de textes, pour éviter que certaines sections ne soient surreprésentées (telle que l'introduction) **[Habert et al., 1997]**.

La construction d'un corpus de textes peut être réalisée en utilisant des moteurs de recherches permettant de retrouver les documents pertinents du web répondant aux besoins de l'utilisateur ou en collectant des informations auprès d'experts de domaines.

L'information contenue dans un texte peut être représentée de différentes manières. La forme la plus basique est le texte brut où l'information est structurée par une suite de mots agencés dans des phrases et des paragraphes.

Pour une représentation plus structurée des connaissances, des représentations plus élaborées ont été proposées. Les premières formes sont représentées par des dictionnaires et des encyclopédies.

1.3.2 Dictionnaire

Un dictionnaire est un ouvrage de référence contenant l'ensemble des mots d'une langue ou d'un domaine d'activité généralement présentés par ordre alphabétique et fournissant pour chacun une définition, une explication ou une correspondance (synonyme, antonyme, cooccurrence, traduction, étymologie)[1].

1.3.3 Encyclopédie

Une encyclopédie constitue une ressource plus riche que le dictionnaire. C'est un ouvrage ou un ensemble d'ouvrages de référence visant à synthétiser des connaissances et à en montrer l'organisation de façon à les rendre accessibles au public en privilégiant un style concis et favorisant la consultation par des tables et des index[2].

Les encyclopédies et les dictionnaires ont évolué pour passer de leur forme classique à des formes numériques accessibles en ligne. Wikipédia[3], une encyclopédie universelle et multilingue représente la plus grande encyclopédie en ligne.

1.3.4 Hiérarchie informelle

Les hiérarchies informelles sont des hiérarchies explicites organisant des catégories à partir de la notion de généralisation / spécification.

- Taxonomie

La taxonomie est la forme la plus simple des vocabulaires contrôlés. Les liens hiérarchiques caractérisant une taxonomie sont des liens de spécialisation /généralisation.

- Thésaurus

Un thésaurus structure les termes normalisés d'un vocabulaire de manière conceptuelle. Un thésaurus est un ensemble de termes organisés suivant trois relations **[Foskett, 1980]**.

[1] https://fr.wikipedia.org/wiki/Dictionnaire
[2] https://fr.wikipedia.org/wiki/Encyclop%C3%A9die
[3] https://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal

(i) les relations hiérarchiques entre descripteurs. Elles permettent de définir si un terme est plus général ou plus spécifique qu'un autre terme, (ii) les relations d'équivalence (entre descripteurs et non-descripteurs). Cette relation est utilisée pour lier un descripteur avec ses synonymes et, (iii) les relations d'association (entre descripteurs). Cette relation n'est pas définie, elle permet de dire simplement que deux descripteurs sont reliés par une relation.

Les thésaurus sont des structures informelles car elles ne définissent pas les relations entre descripteurs. Néanmoins elles permettent d'extraire des informations telles que des définitions et des synonymes. Elles définissent un vocabulaire standardisé permettant leur exploitation dans des applications d'indexation et de recherche documentaire.

WordNet

WordNet[1] **[Miller, 1995]** est un thésaurus représentant une ressource linguistique très utilisée. C'est est une base de données lexicale développée par des linguistes du laboratoire des sciences cognitives de l'université de Princeton. Son but est de répertorier, classifier et mettre en relation de diverses manières le contenu sémantique et lexical de la langue anglaise.

WordNet est décrit par quatre arbres distincts. Chaque arbre couvre respectivement la majorité des noms, verbes, adjectifs et adverbes de la langue Anglaise. Un arbre de WordNet représente des nœuds et les relations qui les relient.

Dans WordNet, un nœud représente un concept appelé "synset" et désigne un terme et l'ensemble de ses termes synonymes. Parmi les différents éléments qui définissent un synset nous citons :

- Le numéro du synset : permet d'identifier de façon unique un synset.
- Le terme représentant le Synset : c'est le terme pour lequel le synset est identifié. Dans WordNet, les termes les plus utilisés sont placés en premier.
- Les termes synonymes.
- Le glossaire : Il contient une définition du synset avec éventuellement un ou plusieurs exemples du monde réel.

Les synsets sont reliés entre eux par plusieurs relations sémantiques nommées : (1) relation de synonymie, (2) relation *is-a* (plus spécifique-générique ou hyponyme-hyperonyme), (3) relation de composition (meronymie-holonymie ou partie-tout, (4) relation antonymie pour exprimer les sens opposés pour les synsets. La Figure 1.1 donne un exemple de relations relatives au synset *Tree*.

[1] https://fr.wikipedia.org/wiki/WordNet

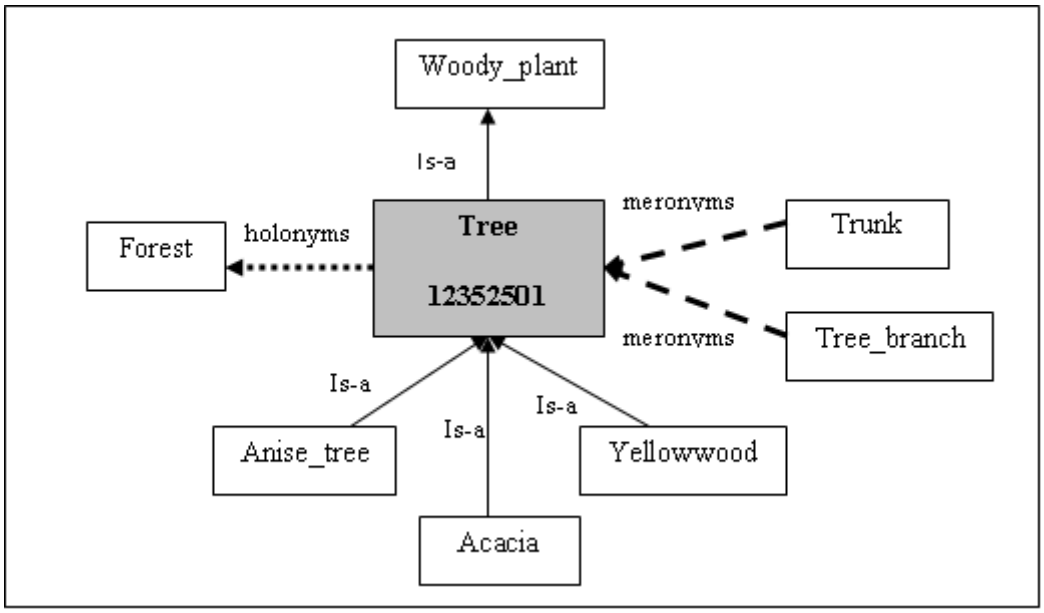

Figure 1.1 Relation extraites de WordNet pour le synset Tree

WordNet Domains

WordNet Domains **[Magnini et al, 2000]** est une ressource lexicale construite de manière semi-automatique en intégrant à WordNet des étiquettes de domaine. Les synsets de WordNet ont été annotés avec une ou plusieurs étiquettes de domaine, pour lesquels ces synsets possèdent un sens. Ces étiquettes sont sélectionnées parmi un ensemble d'environ deux cents étiquettes structurées selon la hiérarchie des domaines WordNet. Les domaines utilisent une terminologie spécifique et fournissent un moyen naturel d'établir des relations sémantiques entre les sens des mots permettant ainsi leur exploitation dans des applications de classification et de désambigüisation. Les domaines définis, sont par exemple, la médecine, l'informatique, l'histoire, le sport etc. Nous donnons dans la Figure 1.2 un extrait de la hiérarchie de domaines définie dans Wordet Domains. La Table 1.1 donne un exemple d'annotation de synsets dans WordNet Domains.

Labels des synsets	N° Synsets	Domaines
exoergic_reaction	12715186	Physics
garnet	13836381	chemistry / geology / jewellery
isometry	14414692	biology / economy / metrology
console	02980721	computer_science / physics

Table 1.1 Annotation de synsets par les domaines où ils possèdent un sens.

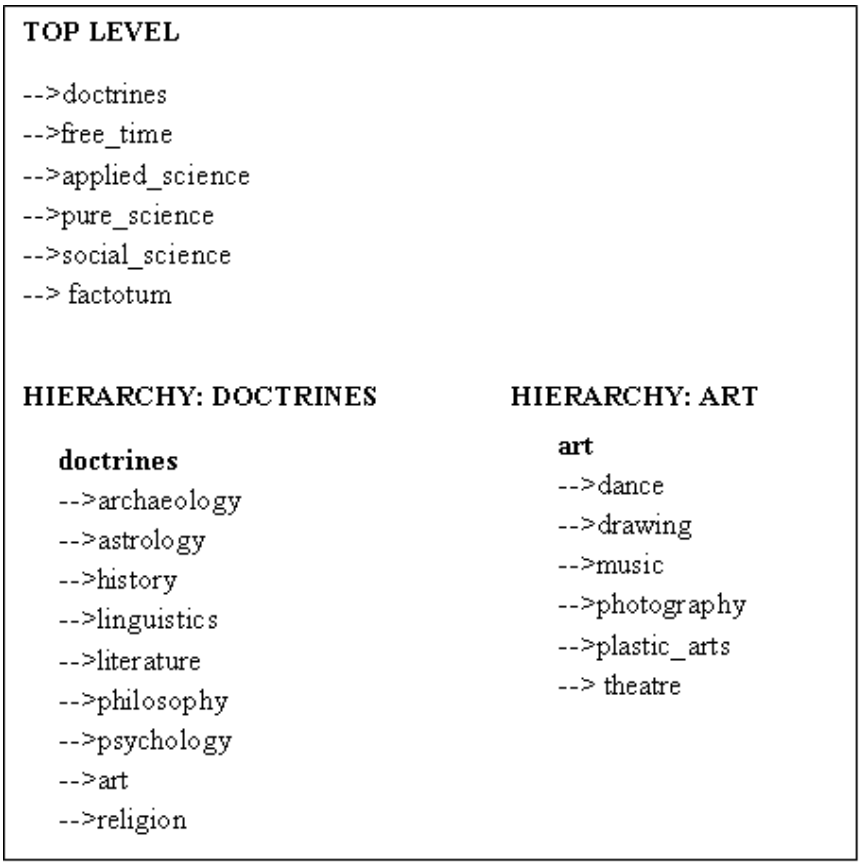

Figure 1.2 Extrait de domaines définis dans WordNet Domains.

1.3.5 Ontologies

Les ontologies sont des ressources conceptuelles présentant des structures formelles. Issues du domaine de la philosophie, elles ont été ensuite introduites en informatique et leur principal objectif est de modéliser les connaissances d'un domaine donné. La représentation d'une ontologie est fondée sur la notion d'abstraction des entités du monde. L'ontologie constitue un modèle de données représentant un ensemble de concepts dans un domaine ainsi que des relations entre ces concepts.

Selon Gruber, **[Gruber, 1995]** "une ontologie est une spécification explicite et formelle d'une conceptualisation partagée". Cela signifie que c'est une représentation qui permet de spécifier dans un langage formel les concepts d'un domaine et leurs relations et que cette représentation est validée par un groupe d'individus.

1.3.5.1. Eléments composant une ontologie

Les ontologies se composent d'un ensemble d'éléments que nous définissons ci-dessous.

- Les **concepts**. Le dictionnaire Larousse donne la définition suivante : "Idée générale et abstraite que se fait l'esprit humain d'un objet de pensée concret ou abstrait, et qui lui permet

de rattacher à ce même objet les diverses perceptions qu'il en a et d'en organiser les connaissances"[1].

Un concept désigne un sens, une idée, de façon non ambiguë. Il représente l'idée véhiculée par les propriétés communes à un ensemble d'objets (un objet peut désigner une entité physique ou abstraite).

Les concepts, appelés également classes, correspondent aux objets à organiser, souvent représentés par un terme ou un groupe de termes. Uschold **[Uschold et al., 1995]** caractérise un concept par trois éléments : (i) un ou plusieurs termes permettant d'identifier le concept, (ii) une notion qui désigne le sens ou la sémantique relative au concept et (iii) un ensemble d'instances du concept (appelé également extension du concept).

- Les **attributs** sont les propriétés, fonctionnalités, caractéristiques ou paramètres que les objets peuvent posséder et partager.

- Les **relations** reliant deux objets. Les relations peuvent être des relations de spécialisation et composition (relations is-a, partie de) ou des relations sémantiques (transversales) propres au domaine considéré ; la sémantique qui leur est associée est définie par un label.

- Les **instances** correspondent aux données réelles dans le domaine représenté (telles que, *Obama* pour le concept *Président*, *tulipe* pour le concept *Fleur*).

- Les **axiomes** sont des assertions qui sont toujours vraies. Ils combinent des concepts, des relations pour définir des règles d'inférence et permettent de vérifier la validité des informations spécifiées ou de déduire de nouvelles informations **[Hernandez, 2005]**.

1.3.5.2. Classification des ontologies

Les ontologies peuvent être distinguées en fonction de l'objet de conceptualisation. Guarino **[Guarino, 1998]** propose une classification à quatre niveaux : ontologie de haut niveau, ontologie de domaine, ontologie de tâches et ontologie d'application. En plus de ces quatre niveaux, Gomez-Perez **[Gomez-Perez, 1999]** propose, deux autres niveaux : ontologie générique, ontologie de représentation. Plusieurs auteurs **[Psyché et al., 2003][Mizoguchi, 2003]** proposent d'autres critères de classification tels que le niveau de granularité et le niveau d'expressivité.

1.3.5.2.1 Classification selon l'objet de conceptualisation

Cette classification fait référence au type de connaissance représentée par une ontologie.

- Ontologies de haut niveau : Les ontologies de haut niveau modélisent des concepts abstraits généraux et des objets du monde tels que les processus, les actions, le temps, l'espace, les évènements etc. *SUMO* (Suggested Upper Merged Ontology)[2] **[Niles et al., 2001],**

[1] http://www.larousse.fr/dictionnaires/francais/concept/17875
[2] http://www.adampease.org/OP/

BFO (Basic Formal Ontology)[1], GFO (General Formal Ontology)[2] et *DOLCE* (Descriptive ontology for Linguistic and Cognitive Engineering)[3] sont des exemples d'ontologies de haut niveau.

- Ontologies Génériques : appelées également méta-ontologies ou core ontologies, ces ontologies représentent des connaissances moins abstraites que celles représentées par l'ontologie de haut niveau et peuvent être réutilisées par différents domaines. Elles permettent par exemple de décrire plusieurs sous-domaines d'un domaine composite (par exemple, une déclinaison du droit en droits public, privé, européen, etc.).

Les ontologies *Mikrokosmos* (ontologie des matériaux) **[Beale et al., 1995]**, *Méréologique* **[Borst, 1997]** et *DC* (Dublin Core Meta data Initiative)[4] qui permet la représentation de documents, sont des ontologies génériques. WordNet est utilisé comme une ontologie générique pour construire des ontologies de domaine **[Hernandez, 2005]**.

- Ontologies de représentation : Elles Permettent de décrire les formalismes de représentation utilisés dans toutes les ontologies. L'ontologie *Frame-Ontology* en est un exemple. Elle est utilisée dans *Ontolingua* **[Gruber, 1993]**. Elle définit de manière formelle les concepts utilisés dans les langages à base de frames : classes, sous-classes, attributs, valeurs, relations et axiomes.

- Ontologies des tâches : Ces ontologies modélisent les tâches d'une activité donnée indépendamment d'un domaine.

- Ontologies d'application : Elles sont construites pour une application spécifique. Elles définissent les connaissances nécessaires pour la réalisation de cette application. Les concepts représentant cette ontologie sont propres à un domaine et une application particuliers. De ce fait, ces ontologies ne sont pas réutilisables par d'autres applications.

- Ontologies de domaine : Elles permettent de modéliser une conceptualisation et une structuration des connaissances d'un domaine. Elles représentent le vocabulaire, activités et théories d'un domaine spécifique par des concepts et des relations entre concepts.

1.3.5.2.2 Classification selon le niveau de granularité

Cette classification met en avant le niveau de description des connaissances représentées par l'ontologie **[Psyché, 2003]** :

- **Granularité fine** : Ce sont des ontologies qui utilisent un vocabulaire très riche permettant une description détaillée des concepts d'un domaine. Ainsi, plusieurs niveaux hiérarchiques sont définis.
- **Granularité large** : ce niveau correspond à des vocabulaires moins détaillés comme c'est le cas pour les ontologies de haut niveau.

[1] http://www.ifomis.org/bfo
[2] http://www.onto-med.de/ontologies/gfo/
[3] http://www.loa.istc.cnr.it/old/DOLCE.html
[4] http://dublincore.org/

1.3.5.1.3 Classification selon le niveau d'expressivité

Mizoguchi **[Mizoguchi, 2003]** distingue les ontologies "légères" et les ontologies "lourdes".

Selon le type de composant mis en œuvre pour représenter une ontologie, nous pouvons avoir deux types d'ontologies. Les ontologies "légères" contiennent seulement les concepts et les relations entre concepts **[Ding et al., 2001]**. Ce sont ces ontologies qui sont les plus utilisées notamment dans le domaine de la recherche d'information. Deux des plus répandues sont *Gene Ontology* (GO)[1] **[Ashburner et al., 2000]**, *UMLS* ((Unified Medical Language System) [**Lindeberg et al., 1993]**. Les ontologies "lourdes" quant à elles regroupent en plus des concepts et des relations entre concepts, des axiomes permettant de réaliser des raisonnements sur les concepts.

Les ontologies lourdes demandent d'importants efforts de conception et le raisonnement avec les axiomes devient complexe lorsqu'il est appliqué sur une ontologie de grande taille. *TOVE* **[Gruninger et al., 1995]** (qui a pour objectif de modéliser les connaissances génériques de l'entreprise) et *PIF* **[Lee et al., 1996]** sont des ontologies lourdes.

1.3.6 Bilan

Dans cette section, nous avons présenté les différentes structures qui peuvent supporter les connaissances. Les connaissances prennent une forme brute lorsqu'elles sont présentées dans un document et une forme plus structurée lorsqu'elles sont regroupées dans des structures plus élaborées telles que des thésaurus et des ontologies. Nous nous intéressons particulièrement aux connaissances contenues dans des documents textuels et dans des ontologies de domaine ainsi que dans WordNet et WordNet Domains. Les textes appartenant à nos corpus sont des résumés d'articles scientifiques. Ces textes appartiennent à des domaines de recherche différents. Notre objectif est d'utiliser les connaissances contenues dans des ontologies de domaine pour annoter le contenu textuel de ces documents. Il s'agit de déterminer les descripteurs qui décrivent le mieux leur contenu. Plusieurs approches se sont intéressées à la définition et à l'extraction des descripteurs de documents. Dans la section suivante, nous allons présenter des approches traitant cette problématique. Nous commençons d'abord par définir les différentes formes que peut prendre un descripteur de document.

1.4 Indexation automatique de documents

1.4.1 Descripteurs de documents

Les applications exploitant des collections de documents pour en extraire ceux pouvant répondre aux besoins des utilisateurs ou pour représenter un domaine spécifique visent à représenter l'information contenue dans ces documents par le biais de descripteurs. Ces derniers sont extraits à partir des corpus de documents **[Zweigenbaum et al., 2003]**. Plusieurs

[1] https://www.uniprot.org/help/gene_ontology

approches se sont intéressées à cette problématique. L'extraction des descripteurs peut être réalisée d'une manière automatique, semi-automatique ou manuelle.

L'extraction manuelle est effectuée par des experts de domaines. Le volume important et sans cesse croissant des documents rendu disponible par le développement d'internet et des différentes technologies de l'information rend cette tâche difficile et coûteuse **[Rastier et al, 1994]**. Des approches ayant recours à une extraction automatique des descripteurs sont apparues.

Pour exploiter les documents, les applications appliquent un prétraitement sur le contenu brut des documents pour les représenter par un ensemble de descripteurs appelés mots clés. Un processus de segmentation est alors appliqué pour diviser le texte en plusieurs unités. Il s'agit d'extraire à partir d'un document l'ensemble des descripteurs qui représente le mieux son contenu textuel. Cet ensemble doit être le plus exhaustif possible. Un document peut être représenté non pas par les termes explicitement cités dans son contenu mais par les liens sémantiques reliant les termes co-occurrants dans un corpus de référence comme c'est le cas dans l'approche Latent Semantic Indexing **[Deerwester et al., 90].** Les descripteurs peuvent être des tokens, des stemmes, des lemmes, des N-grammes, des groupes de mots ou des concepts généralement extraits à partir d'un thésaurus ou d'une ontologie.

- Token

Les tokens **[Gaussier et al., 03]** sont des unités élémentaires qui peuvent avoir des présentations différentes : variation dans le genre (féminin, masculin), forme conjuguée des verbes, forme pluriel ou singulière. Un token est alors représenté par sa base à travers un processus de normalisation. La normalisation appliquée est soit la racinisation (stemming) ou la lemmatisation.

- La **racinisation** (désuffixation, ou stemming en anglais) est un procédé qui vise à transformer les flexions en leur radical (racine) ou stemme. La racine d'un mot correspond à la partie du mot restant une fois que l'on a supprimé son préfixe et son suffixe. Les algorithmes les plus connus sont l'algorithme *Lovins* développé par Julie Beth Lovins **[Lovins, 1968]** et l'algorithme *Porter* développé par Martin Porter[1] **[Porter, 1980]**.

Les algorithmes produisant des stemmes sont difficiles à comprendre et donnent des mots qui n'ont pas de sens et peuvent conduire dans certains cas à une normalisation agressive (production d'un mot dont le sens est différent du mot original).

Par exemple, le mot *organization* donne le stemme *organ*, le mot *university* donne le stemme *univers* avec l'algorithme Porter.

- La **lemmatisation** est une analyse lexicale du contenu d'un texte regroupant les mots d'une même famille et les transformant en leur forme canonique appelée lemme. Un lemme correspond à un mot réel de la langue, alors que la racine ou stemme ne correspond généralement pas à un mot réel. Deux exemples d'algorithmes de lemmatisation sont

[1] https://tartarus.org/martin/PorterStemmer/index.html

TreeTagger[1] **[Schmid, 1994]**, développé par l'Université de Stuttgart et *Stanford Pos Tagger*[2] **[Toutanova et al., 2003]** développé à l'université de Stanford. Ces deux algorithmes traitent de multiples langues dont le français et l'anglais. La Table 1.2 donne les stemmes et les lemmes extraits pour quelques mots pris en exemple.

Mots	Stemmes (Porter)	lemmes
General	Gener	General
Chemical	Chemic	chemical
Carried	Carri	carry
Iteration	Iter	Iteration
companies	compani	company

Table 1.2 Extraction des lemmes et des stemmes

- Groupe de mots

Un groupe de mots, appelé également syntagme ou phrase en anglais, représentent plusieurs mots contigus ayant une sémantique. Les syntagmes sont souvent utilisés parce qu'ils reflètent mieux le sens des mots pris dans le contexte de leur apparition. Par exemple, les mots, (*memory, device) et (secretary, of, state, for, the, home, department)* pris ensemble respectivement constituent les groupes de mots *memory_device* et *secretary_of_state_for_the_home_department*. Ces groupes de mots possèdent un sens précis.

- N-gramme

Un n-gramme désigne une suite de *n* éléments qui peuvent être des caractères ou des mots. Les n-grammes ne possèdent pas une sémantique mais sont utilisés dans des traitements statistiques notamment en traitement automatique du langage naturel. Ces traitements permettent d'apprendre les n-grammes à partir d'un corpus en se basant sur l'hypothèse que, étant donnée une séquence de k éléments, ($k >= n$) la probabilité de l'apparition d'un élément en position *i* ne dépend que des *n-1* éléments précédents. Les tailles des n-grammes les plus utilisées sont les uni-grammes (n=1), les bi-grammes (n=2) et les tri-grammes (n=3). Par exemple, les tri-grammes associés à la suite de caractères ABCDEF sont : ABC, BCD, CDE, DEF.

1.4.2 Pondération des termes

Un descripteur dans un document est souvent représenté par son poids. Le poids d'un terme dans un document évalue l'importance de ce terme dans le document et son pouvoir discriminant vis-à-vis du document qui le contient. Le calcul du poids est généralement basé sur des considérations statistiques. **[Robertson et al., 1997] [Singhal et al., 1997] [Spark Jone, 1971] [Spak Jones, 1979].**

[1] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
[2] https://nlp.stanford.edu/software/tagger.shtml

La loi de Luhn **[Luhn, 1958]** élimine les descripteurs très fréquents, très utilisés par la majorité des documents et les descripteurs peu fréquents, considérant leur pouvoir discriminant très faible.

La pondération communément utilisée*, tf-idf* **[Salton et al., 1983]**, combine deux facteurs. Un facteur de pondération locale, appelé *tf* (*term frequency*) mesurant l'importance d'un terme dans un document, et un facteur de pondération globale *idf* (*Inverse of Document Frequency*), mesurant la représentativité globale du terme vis-à-vis du corpus considéré. Un poids plus important doit être donné aux termes qui apparaissent moins fréquemment dans la collection.

- Term frequency, *tf*

Plusieurs formules de pondération locale ont été proposées basées sur le nombre d'occurrence du terme dans le document.

$$fr(t,d) \tag{1.1}$$

$$1+\log(fr(t,d)) \tag{1.2}$$

$$\frac{fr(t,d)}{\max_{t'} fr(t',d)} \tag{1.3}$$

- Inverse document frequency, *Idf*

$$idf = \log\left(\frac{N}{df}\right) \tag{1.4}$$

où *df* représente le nombre de documents contenant le terme considéré et *N* représente le nombre total de documents de la collection.

Les formules ainsi définies favorisent les documents longs puisque le nombre d'occurrences des termes appartenant aux documents longs est plus élevé que dans les documents courts et, par conséquent, un poids plus important leur est affecté. Une normalisation des valeurs pour prendre en compte la taille des documents a été introduite. **[Singhal et al, 1996][Robertson et al, 1997].** Nous citons les pondérations locales normalisées suivantes :

$$\frac{fr(t,d)}{longueur(d)} \tag{1.5}$$

$$\frac{fr(t,d)}{fr(t,d)+0{,}5+1{,}5\times\frac{longueur(d)}{longuer\_moyenne\_doc}} \tag{1.6}$$

Robertson **[Robertson et al, 1997]** a proposé la formule suivante :

$$Wij = \frac{tf_{ij} \times (K_1 + 1)}{K_1 \times ((1-b) + b \times \frac{dl_j}{\Delta l}) + tf_{ij}} \qquad (1.7)$$

Où *Wij* est le poids du terme *ti* dans le document *dj*,

tf_{ij} est la fréquence du terme *ti* dans le document *dj*

K_1 contrôle l'influence de la fréquence du terme *ti* dans le document *dj*. Sa valeur dépend de la longueur des documents dans la collection de documents.

b est une constante qui contrôle l'effet de la longueur du document

dl_j est la longueur du document *dj*

Δl est la longueur moyenne des documents dans la collection.

Pour mieux représenter un document, la segmentation de son contenu pour extraire les différents descripteurs ne se fait pas toujours au niveau "mot simple". En effet, des mots contigus, pris ensemble, peuvent représenter une sémantique qui ne peut être retrouvée lorsque ces mots sont pris séparément. Dans la section suivante, nous décrivons quelques approches qui traitent de l'extraction d'une suite de mots représentant une unité sémantique.

1.4.3 Approches d'extraction des termes

Les premiers travaux définissaient le processus d'extraction des termes par l'identification de collocation. Leur point commun est d'identifier des segments de texte qui se répètent à l'intérieur d'un corpus. Choueka, **[Choueka, 1988]** définit une collocation comme étant une suite de deux ou plusieurs mots formant une unité syntaxique ou sémantique dont la signification ne peut être déduite directement à partir de ses constituants.

Les applications permettant d'extraire des termes à partir de documents exploitent un corpus spécialisé couvrant le domaine à représenter et définissent deux modes d'analyse différents. Une catégorie d'approches applique une analyse statistique et une autre catégorie s'appuie sur une analyse linguistique du corpus **[Claveau, 2003**]. De nouvelles approches combinent les deux catégories et proposent des méthodes hybrides.

1.4.3.1 Méthodes statistiques

Les méthodes statistiques sont très utilisées lorsqu'on veut traiter un corpus de taille volumineuse. Elles permettent un traitement rapide puisqu'elles n'utilisent aucun traitement linguistique et ne reposent que sur l'exploitation d'un corpus. L'analyse statistique s'appuie sur la distribution et le contexte d'apparition des termes dans les documents. Elles exploitent uniquement un corpus et aucune ressource externe n'est utilisée. Les descripteurs peuvent être des stemmes ou des lemmes représentés souvent par leurs fréquences.

Des critères d'associations sont également utilisés pour mesurer la liaison de deux lemmes ou lexèmes (Unité de sens et de son figée dans une langue, sans distinction flexionnelle ou dérivationnelle. Par exemple, *forment* et *formeront* sont des formes du même lexème *former*[1]). L'information mutuelle permet de comparer la probabilité d'apparition des deux lexèmes ensemble avec la probabilité de les observer séparément **[Brown et al., 1990][Lebart et al., 1988][Church et al., 1989].**

Travaux de Church

Les premiers travaux statistiques traitant des données linguistiques sont ceux de Church **[Church et al., 1989]** qui identifie automatiquement l'ensemble des collocations contenues dans un ensemble de documents textuels. Il définit l'information mutuelle qui reflète la liaison entre deux lexèmes. Il compare alors la probabilité d'observer ces deux lexèmes ensemble avec leur probabilité d'apparition seuls. La probabilité d'apparition d'un lexème seul est donnée par le rapport entre sa fréquence totale dans un corpus et le nombre total de lexèmes dans le corpus. La probabilité d'apparition des deux lexèmes ensemble est donnée par le nombre de fois où les deux lexèmes apparaissent ensemble dans une fenêtre de taille *t*, où *t* représente le nombre variable de lexèmes formant cette fenêtre.

Travaux de Fagan

La méthode statistique utilisée par Fagan **[Fagan, 1987]** permet d'extraire des descripteurs de deux types : uniterme et bi-termes. Les bi-termes sont des groupes de deux mots adjacents. Son processus détermine, à partir de leurs co-occurrences, les termes acceptables pour former des groupes de mots candidats *p*. Les groupes de mots sont sélectionnés en fonction de leur fréquence *df-phrase* dont la valeur doit être comprise entre un seuil minimal et un seuil maximal définis par l'auteur. A Chaque token du document est associé un statut déterminant s'il représente la tête de *p* (*phrase-head*) ou un composant de *p* (*phrase-comp*). Pour cela des seuils de fréquences *df-head* et *df-comp* sont définis. Un token *t* est acceptable pour être *phrase-head* si sa fréquence est supérieure à *df-head*. Si la fréquence de *t* est supérieure à *df-comp*, *t* est alors un *phrase-comp*. Fagan construit deux vecteurs, l'un contenant les descripteurs uniterme et l'autre les groupes de deux mots.

Dans son exemple, le document 71 extrait de la collection *CISI* possède les tokens (*word, word, associ, docu, retrief, system*). Le token *docu* est acceptable pour être un *phrase-head*. Il est alors combiné avec les tokens, *associ* et *retrief* qui lui sont adjacents pour former les groupes "*docu associ*" et "*docu retrief* ".

Associ ne peut être un *phras-head* car sa fréquence est inférieure au seuil retenu.

Fagan extrait pour le document 71, les descripteurs unitermes et bi–termes suivants : (*word, associ, docu, retrief, system, retrief system, docu retrief, word associ, docu associ*).

[1] https://fr.wiktionary.org/wiki/lex%C3%A8me

Pour l'extraction des termes, Fagan propose également une méthode syntaxique. Les résultats de ses expérimentations montrent que la méthode statistique donne de meilleurs résultats que la méthode syntaxique.

Travaux de Lebart

Les travaux de Lebart **[Lebart et al., 1988][Lebart et al., 1994**] ont pour objectif d'extraire des termes composés à partir d'un corpus de textes lemmatisés. Ils recherchent dans un corpus des séquences de mots contigus (segments) qui se répètent plusieurs fois dans un texte. Un seuil est fixé par expérimentation et permet de décider qu'une séquence définit un terme composé.

Travaux d'Ahmad

Les travaux d'Ahmad **[Ahmad, 1996]** consistent à repérer des formes, appelés "étranges", couvrant un domaine donné. L'objectif est de regrouper ces formes dans des listes pour leur exploitation par des terminologues. Deux corpus sont utilisés, l'un est technique et l'autre non technique. Il définit un coefficient d'étrangeté (*co-efficient of weirdness*) comme étant le rapport entre la fréquence relative d'une forme dans un corpus non spécialisé et la fréquence relative de la même forme dans un corpus technique. La liste des formes obtenue est triée en fonction du coefficient d'étrangeté et place les formes liées à la thématique du corpus technique en début de liste.

Travaux d'Alvarez

Alvarez **[Alvarez et al., 2004]** montre dans ses travaux que l'hypothèse de l'indépendance des termes sur laquelle se base la plupart des approches classiques dans le domaine de la recherche d'information n'est pas toujours justifiée. Si un utilisateur spécifie dans sa requête les mots clés "recherche d'information", un document qui traite le thème *moteurs de recherche* et contient les termes "*recherche d'information*" est intuitivement plus pertinent qu'un document n'ayant pas de rapport avec le thème recherché et qui contient les termes "recherche" et "information" dans des contextes indépendants.

Alvarez exploite les termes composés de deux mots qui apparaissent l'un à côté de l'autre sans contrainte sur l'ordre des mots avec le modèle de langue. Alvaez part de l'hypothèse que l'ordre des mots n'est pas toujours important en recherche d'information. En prenant par exemple la requête "*apartment rent*". Un document contenant l'expression "*rent an apartment*" ne doit pas être considéré moins pertinent qu'un document contenant l'expression "*apartments for rent"*. Il sélectionne les paires de mots en déterminant des relations statistiques, ou des affinités lexicales, entre les mots qui co-occurrent dans une fenêtre de 5 mots (à gauche et à droite d'un mot donné).

1.4.3.2 Méthodes linguistiques

L'analyse linguistique nécessite la connaissance de la langue du corpus et exploite les structures morphologiques et syntaxiques des termes. Les expressions (syntagmes) sont extraites syntaxiquement par l'exploitation des relations linguistiques entre les mots. Les approches linguistiques se basent sur une analyse syntaxique partielle ou l'utilisation de patrons syntaxiques pour détecter les termes composés. Par exemple les patrons (*nom nom*) ou (*nom prep nom*) sont souvent utilisés.

Plusieurs analyseurs syntaxiques existent dans la littérature Nous décrivons ci-dessous quelques outils d'acquisition automatique des termes et nous commençons par *TERMINO* qui est considéré comme le premier outil à répondre à cette problématique.

TERMINO

L'outil *TERMINO* **[David et al., 1990],** appelé *NOMINO* **[Perron, 1996]** dans sa version récente, est un outil conçu par une équipe du Centre d'ATO de l'Université du Québec et l'Office de la langue française du Québec. Les termes extraits par *NOMINO* sont des syntagmes nominaux appelés "*synapsies*". Le processus englobe plusieurs étapes.

Un prétraitement du texte est réalisé permettant son découpage en phrases et en lexèmes. Il permet également d'identifier des lexèmes particuliers comme les noms propres et les abréviations. Les lexèmes sont ensuite lemmatisés par analyse morphosyntaxique. Une catégorie grammaticale leur est attribuée. A cette étape, un lexème peut avoir plusieurs catégories grammaticales.

Une étape de désambiguïsation en contexte détermine pour chaque lexème une catégorie grammaticale unique par l'application de règles exploitant la morphologie des unités lexicales et la syntaxe d'une langue. Ces règles rendent cet outil dépendant des langues. La dernière étape consiste à extraire les unités nominales à partir des textes.

LEXTER

LEXTER est un outil conçu par Bourigault **[Bourigault, 1992]**. A sa création, il permettait d'extraire des syntagmes nominaux et était utilisé lors de la construction de ressources terminologiques ou d'ontologies spécialisées. Sa version actuelle, appelée *SYNTEX* **[Bourigault et al., 2000]**, permet l'extraction à partir d'un corpus textuel d'un ensemble de termes simples ou composés appelés syntagmes nominaux, verbaux et adjectivaux. SYNTEX est un analyseur de corpus car le réseau de dépendance est construit pour l'ensemble du corpus.

Un étiqueteur morpho-syntaxique est d'abord utilisé pour définir le rôle des mots dans un document. Pour un corpus en langue anglaise, *SYNTEX* prend en entrée les résultats de *TreeTagger* pour déterminer si un mot est un nom, un verbe, un adjectif etc. *SYNTEX* se base sur la grammaire traditionnelle pour réaliser une analyse syntaxique qui définit pour chaque phrase des relations de dépendance syntaxique entre les mots (sujet, complément d'objet,

épithète etc.). Pour chaque phrase, il extrait des syntagmes nominaux, verbaux et adjectivaux et pour tout le corpus, il construit un réseau de syntagmes où chaque syntagme est décomposé en deux parties : la partie tête (*T*) et la partie expansion (*E*). Dans ce réseau, chaque syntagme est relié à sa tête et à son expansion, et chaque tête et chaque expansion sont reliées aux syntagmes dont ils font partie.

FASTR

FASTR **[Jacquemin, 1997]** est un analyseur syntaxique qui utilise une liste contrôlée et un ensemble de métarègles pour repérer des termes et leurs variantes dans un corpus. Les variantes peuvent être de types syntaxiques, morpho-syntaxiques et sémantico-syntaxiques.

SYMONTOS

SYMONTOS **[Velardi et al., 2001]** est un outil qui prend en compte les termes simples et complexes.

1.4.3.3 Méthodes hybrides

Système ACABIT

Daille **[Daille, 1993]** se base sur l'idée qu'une analyse statistique doit succéder et compléter l'analyse linguistique. De ce fait, dans son outil *ACABIT,* il s'intéresse à l'acquisition automatique des termes composés à partir d'un corpus en langue française, préalablement étiqueté, en deux étapes. Une première étape consiste à exploiter les procédés linguistiques tels qu'ils sont définis dans *NOMINO* et *LEXTER*. Des syntagmes nominaux retrouvés par un automate sont lemmatisés et représentent l'entrée pour la deuxième étape où une analyse statistique est appliquée pour décider quels syntagmes retenir. Plusieurs mesures statistiques sont testées. Une liste de termes validés par des terminologues est comparée aux résultats de chaque test et permet de déterminer la mesure qui donne les meilleurs résultats. Dans d'autres travaux, Daille utilise un corpus bilingues (anglais-français) **[Daille, 1994a] [Daille, 1994b]**.

Système TERMS

Justeson **[Justeson et al., 1995]** s'intéresse également à l'acquisition automatique des termes composés à partir d'un corpus en langue anglaise. Ses travaux ont abouti à la réalisation du logiciel *TERMS*. Les termes composés sont sélectionnés d'abord en fonction de leur fréquence. Celle-ci doit être au moins supérieure ou égale à 2. A partir de termes composés sélectionnés, certains sont éliminés en utilisant une grammaire qui décrit les structures possibles pour les termes.

Système XTRACT

Le système XTRACT, réalisé par Smadja **[Smadja, 1993]**, permet de définir des collocations dans un corpus en langue anglaise.

Dans une première étape, l'auteur exploite des mesures statistiques pour déterminer les mots fortement corrélés. Une fenêtre de 5 mots est utilisée pour extraire des couples de mots : bi-grammes) ayant une information mutuelle élevée. Une analyse contextuelle des bi-grammes permet de déterminer des séquences plus longues : n-grammes.

A ces n-grammes est appliqué un filtrage en appliquant une analyse syntaxique qui permet de définir les catégories grammaticales ou syntaxique des différents mots des bi-grammmes au sein des collocations. Cela permet de retrouver la relation qui unit les deux mots d'un bi-gramme (nom-nom, sujet-verbe, verbe-objet, adjectif-nom, etc.).

1.4.4 Analyse morphosyntaxique et étiqueteurs grammaticaux

L'analyse morphosyntaxique permet de déterminer la forme et la syntaxe d'un mot qui correspond à son rôle dans une phrase. Elle est utilisée dans plusieurs applications faisant appel au traitement de la langue comme les applications de traduction automatique des langues, et la recherche d'information. Son objectif est d'assigner à un mot sa catégorie grammaticale qui peut être un nom, un verbe, un adjectif, un adverbe etc.

Des étiqueteurs morphosyntaxiques ont été développés sur la base d'algorithmes différents : (1) Approches à base de règles, *TAGGIT* **[Greene et al., 1971]**, (2) Approches statistiques qui englobent les approches qui utilisent les Modèles de Markov Cachés, ou le Maximum d'entropie (*Stanford Pos Tagger*), ou encore les arbres binaires de décision (*TreeTagger*) et (3) Approches non supervisées **[Brill, 1995].**

Nous donnons dans ce qui suit une description de trois étiqueteurs morphosyntaxiques qui sont accessibles en ligne, distribués librement et prêts à l'emploi. Ces algorithmes ont l'avantage de prendre en charge plusieurs langues, notamment l'anglais et le français.

TREETAGGER [Schmid, 1994] est un outil permettant l'étiquetage morphosyntaxique et la lemmatisation. Il permet d'annoter des textes en assignant une étiquette aux mots en indiquant s'ils représentent un nom, un verbe, un adjectif etc. Il a été développé par Helmut Schmid dans le cadre du projet TC dans l'ICLUS (l'Institut for Computational Linguistics of the University of Stuttgart). Il prend en charge plusieurs langues telles que le français, l'anglais et l'allemand et utilise un corpus d'apprentissage manuellement étiqueté. *TreeTagger* utilise un arbre de décision binaire pour calculer la taille du contexte à utiliser afin d'estimer les probabilités de transition. *TreeTagger* fonctionne en deux phases. Une première phase d'apprentissage génère un fichier à partir d'un corpus d'apprentissage et un lexique. Ce fichier est exploité dans la seconde phase pour réaliser l'étiquetage proprement dit.

STANFORD TAGGER [Toutanova et al., 2000][Toutanova et al., 2003] est un étiqueteur qui prend en charge plusieurs langues (anglais, français, chinois etc.). Il se base sur le contexte d'apparition d'un mot pour calculer les probabilités associées aux différentes étiquettes. Il s'appuie sur le modèle bidirectionnel de Markov et sur le maximum d'entropie.

Le processus génère un ensemble de caractéristiques pondérées. Un corpus d'apprentissage est utilisé et des règles sont définies permettant de faire des correspondances avec les données d'entrainement. La probabilité d'obtenir une étiquette à partir d'un contexte est donnée par l'équation (1.8).

$$P(t / C) = \frac{1}{Z(C)} \exp\left(\sum_{i=1}^{n} \lambda_i f_i(C,t)\right) \qquad (1.8)$$

Où *t* désigne une étiquette morphosyntaxique, *C* représente le contexte, *fi* et λ_i représentent respectivement les caractéristiques et leur poids. *Z(C)* est une constante.

TALISMANE[1] [Urieli et al., 2013] est un analyseur syntaxique développé au sein du laboratoire CLLE-ERSS. Il s'appuie sur un classifieur probabiliste et utilise quatre étapes : le découpage en phrases, la segmentation en mots, l'étiquetage (attribution d'une catégorie morphosyntaxique) et le repérage des dépendances syntaxiques entre les mots. Les modules sont définis par apprentissage sur un corpus annoté et ils sont configurables au niveau des traits et au niveau des règles. Les traits, définis durant l'étape d'apprentissage, sont des informations sur les configurations rencontrées dont dispose l'algorithme pour prendre chacune des décisions. Par exemple, pour l'étiquetage, des traits classiques sont calculées pour chaque mot, tels que des traits liés à sa forme, aux étiquettes indiquées dans un lexique de référence, aux catégories des mots qui l'entourent, etc. Des traits plus complexes peuvent être définis, comme par exemple indiquer que le mot précédent est placé entre parenthèses.

Les règles sont appliquées lors de l'analyse pour contraindre les réponses fournies par le classifieur probabiliste quand un critère est rempli. Ces règles permettent d'éviter des résultats non cohérents comme par exemple l'attribution de deux sujets à un verbe ou de respecter des contraintes spécifique à un corpus en attribuant par exemple une catégorie fixe à un mot donné.

1.4.5 Annotation sémantique de documents

L'annotation sémantique est un processus qui permet d'associer à une donnée de nouvelles données sur la base de liens sémantiques.

Cette annotation peut se diviser en deux classes : Une annotation qui vise à ajouter des données complémentaires appelées métadonnées au contenu ou à une partie du contenu textuel d'un document et une annotation qui permet d'associer un sens ou un concept à un mot pour indexer un document. Cette dernière est appelée indexation sémantique.

[1] http://redac.univ-tlse2.fr/applications/talismane.html

Les approches d'annotation qui permettent d'attacher des métadonnées à des fragments textuels au sein d'un document se basent sur l'utilisation de patrons ou sur l'exploitation des techniques d'apprentissage automatique. Un patron représente une forme syntaxique définie manuellement ou automatiquement. Il permet d'associer une annotation à une entité lorsque ce patron est retrouvé dans le texte. Par exemple avec le patron <*A est né à B*>, on peut annoter *A* par *personne* et *B* par *lieu*.

Pour définir des patrons de façon automatique, une liste initiale d'entités est constituée. Ces entités sont recherchées dans un ensemble de documents. Les formes syntaxiques relatives à ces entités vont constituer des patrons. Si on recherche par exemple l'entité *virus* dans un document, nous pouvons retrouver dans son contexte le mot bronchite. <*A* provoque *B*> peut constituer un patron qui permet de lier un *virus* à une *maladie*.

Des modèles probabilistes sont utilisés par les techniques d'apprentissage automatique pour prédire l'annotation d'une entité en fonction du contenu des documents. C'est le cas par exemple pour les approches visant à retrouver la catégorie grammaticale des éléments d'une phrase.

Pour annoter des fragments de textes, des bases de connaissances sont utilisées comme Wikipédia **[Alemzadeh et al.,2010]** et DBpedia. DBpedia **[Auer et al. 2008]** est un projet permettant l'exploration et l'extraction automatique des informations structurées dérivées de Wikipédia dans plusieurs langues en les rendant accessible sur le Web. C'est une ontologie inter-domaines créée sur la base des infoboxes les plus utilisées dans Wikipédia. Cependant, DBpedia est une ontologie générique et, par conséquent, peu adaptée pour des documents techniques.

Des annotateurs sémantiques sont utilisés tels que DBpedia Spotlight **[Mendes et al., 2011]** et Wikimeta[1] pour enrichir le contenu textuel d'un document avec des annotations sémantiques en utilisant les documents DBpedia. DBpedia Spotlight permet d'associer les pages Web (page de DBpedia) aux entités nommées reconnues.

Une entité nommée est une unité textuelle qui fait référence par exemple à un nom de personne, un pays, un lieu, une entreprise, une date etc. Pour l'identification des autres entités propres à un domaine donné, une liste appelée *gazetteer* doit être établie pour être exploitée.

Plusieurs plates formes d'annotation linguistique permettent d'intégrer un certain nombre d'outils de Traitement Automatique des Langues existants. Nous pouvons citer par exemple GATE **[Cunningham, 2002]**, et Ogmios **[Hamon et al., 2007]**. Pour l'annotation des textes basée sur une ontologie externe, le logiciel GATE **[Cunningham et al. 2011]** peut être utilisé. Gate est un logiciel open source qui contient un système d'extraction d'information, ANNIE (A Nearly-New Information Extraction System) composé de plusieurs modules comme un analyseur lexical, un analyseur syntaxique permettant la segmentation de phrases, une base de toponymes (gazetteer) etc. GATE utilise également le langage JAPE (Java Annotation Patterns Engine) pour construire des règles d'annotation de documents. Gate est capable de réaliser des appariements entre les documents et les éléments d'une ontologie donnée en entrée. Les noms (labels par exemple) des ressources ontologiques telles que des classes et

[1] https://www.programmableweb.com/api/wikimeta

des instances sont extraits de l'ontologie sur lesquels des prétraitements sont effectués afin de déterminer les racines des ressources ontologiques. Les racines des mots des documents sont également extraites. Les racines obtenues à partir de l'ontologie et les racines des mots appartenant aux documents sont ensuite comparées.

Nous nous plaçons dans le contexte d'une indexation sémantique et plus précisément dans une indexation conceptuelle basée sur des ressources externes.

Dans une indexation conceptuelle basée sur WordNet, des API sont utilisées pour faire une recherche dans la ressource. Nous citons par exemple l'API *JAWS* (Java API for WordNet Searching) et l'API J*WNL* (Java Word Library). Le choix d'une API se fait en fonction de plusieurs critères comme la compatibilité avec la version de WordNet utilisée et la disponibilité des mesures de similarité. Une comparaison entre les différentes API est donnée dans **[Finlayson, 2014]**. Dans notre travail, nous utilisons l'API JWNL. Nous montrerons dans le chapitre 5 comment représenter l'ontologie de domaine que nous avons construite afin d'exploiter cette API.

1.4.6 Bilan

Dans cette section, nous avons présenté les différentes formes que peut prendre un descripteur. Un descripteur peut être un token, un terme, un n-gramme ou un groupe de mots. Dans notre cas, nous visons à extraire du texte les groupes de mots ayant une entrée dans les ressources sémantiques que nous allons utiliser. Nous montrerons dans le chapitre 3, l'intérêt de ce choix.

Aux descripteurs de documents est souvent associé un poids représentant son pouvoir discriminant au sein du document. Ce poids est calculé sur la base d'une fréquence locale (*tf*) et d'une fréquence globale (*idf*). Avec ces formules, des mots peu fréquents dans le document se voient attribuer un poids très faibles et par conséquent, ils ne seront pas retenus pour représenter un document même si ces descripteur en un sens pour son contenu.

Nous avons également abordé la notion d'annotation sémantique des documents. Cette annotation permet d'associer aux fragments de textes des données complémentaires (métadonnées) ou, dans une indexation sémantique, des sens ou des concepts extrait à partir d'une ressource sémantique. Dans le cadre de nos travaux, nous ne nous plaçons pas dans une approche d'annotation par métadonnées mais plutôt dans un processus d'indexation conceptuelle du contenu des documents.

Les descripteurs de documents peuvent avoir plusieurs sens en fonction du contexte dans lequel ils apparaissent. Un processus de désambiguïsation est alors nécessaire. Nous présentons dans la section suivante le problème lié à l'ambiguïté des mots.

1.5 Le problème de l'ambiguïté des mots

Dans une indexation classique, un document est représenté par un ensemble de descripteurs appelés mots clés qui ne sont considérés que comme une suite de caractères dépourvue de sens. Les mots considérés seulement par leur morphologie génèrent une

ambigüité et par conséquent engendrent des erreurs d'interprétation par les applications traitant le contenu textuel des documents.

Krovetz **[Krovetz, 1997]** dissocie l'ambigüité des mots en deux types : l'ambiguïté syntaxique et l'ambiguïté sémantique.

- L'**ambiguïté syntaxique** concerne des mots ayant des morphologies similaires et des catégories syntaxiques différentes (nom, verbe etc.). Par exemple *je ficelle* (verbe) et *une ficelle* (nom).

- Quant à **l'ambiguïté sémantique**, elle fait référence au sens des mots ayant une même morphologie. Nous avons dans cette catégorie :

  - la **polysémie** qui désigne un mot ayant plusieurs sens. Par exemple, le mot *avocat* désigne le *fruit* et le *juriste*.
  - l'**homonymie** qui désigne des mots ayant des morphologies similaires et des sens différents tel que c'est le cas pour le mot *fils* dans la phrase, *Il lègue à son fils toute sa fortune* et dans la phrase, *le fils à coudre*.

Weiss **[Weiss, 73]** divise l'ambigüité en trois classes : vraie, contextuelle et syntaxique.

L'ambigüité contextuelle est basée sur la notion de fonction sémantique. Pris seuls, les mots n'ont pas un sens bien défini. Leur signification exacte est déduite à partir de leur contexte d'apparition. L'influence qu'un mot exerce sur son contexte est appelée sa fonction sémantique. Par exemple, dans la phrase *bottom of the bottle*, l'utilisation du mot *bottom* (fonction sémantique) dans le contexte " *of the bottle* " donne la valeur sémantique : le point bas dans un récipient en verre. Sur la base de la fonction sémantique, il distingue deux types d'ambigüité : une ambiguïté vraie et une ambigüité contextuelle.

- Une **ambiguïté vraie** est un mot ayant au moins deux fonctions sémantiques distinctes. Par exemple le mot *degree* en anglais peut désigner une *unité de mesure* ou un *prix académiqu*e.

- **Ambigüité contextuelle**. Certains mots ayant une seule fonction sémantique peuvent sembler ambigus. Cela arrive lorsqu'une seule fonction sémantique, agissant sur une variété de contextes, produit des significations différentes. De tels mots sont appelés contextuellement ambigus parce que leurs multiples significations proviennent de leur contexte et non pas de leur fonction sémantique.

Dans l'exemple ci-dessous, le mot *base* possède une seule fonction sémantique mais produit des sens différents en fonction du contexte où il est utilisé.

FIRST BASE (baseball)

MILITARY BASE

LAMP BASE

BASE 2 (radix of number system)

BASE REGISTER

- **Ambigüité syntaxique** : Ce sont des mots dont le sens varie en fonction de leur rôle syntaxique dans la phrase.

1.6 Indexation sémantique des documents

La représentation d'un document par des mots présente l'inconvénient d'ignorer la sémantique décrite dans le contenu textuel du document. Cette représentation ne met pas en évidence les relations reliant les mots dans le contexte de leur apparition et les mots sont dénués de sens. Les documents sont alors représentés comme des sacs de mots. Le problème récurrent dans le processus de représentation d'un document par un ensemble de descripteur est la présence d'ambigüités inhérentes à certains termes du langage. Comme la polysémie, la synonymie génère également des problèmes d'interprétation lorsqu'elle n'est pas prise en charge. En effet des mots représentant un même objet peuvent avoir des morphologies différentes. Retrouver alors le sens des mots du texte pour reproduire fidèlement la sémantique véhiculée par son contenu est une problématique sur laquelle plusieurs approches se sont penchées.

Pour construire une représentation sémantique des documents, certaines approches utilisent seulement un corpus pour déterminer le sens des mots alors que d'autres exploitent des ressources sémantiques pour désambigüiser les termes ambigus présents dans le texte.

Une représentation sémantique des documents repose sur l'acquisition du sens des mots décrivant leur contenu et fait appel généralement à un processus de désambigüisation. On parle alors d'indexation sémantique quand le sens d'un mot est déterminé en fonction de son contexte d'apparition, ou d'indexation conceptuelle quand les descripteurs d'un document sont des concepts correspondants aux nœuds définis dans des structures conceptuelles telles des thésaurus et des ontologies. Un concept est représenté par un ou plusieurs termes. Un terme peut dénoter plusieurs concepts dans des domaines différents.

1.6.1 Représenter par le sens

Dans cette représentation, le contexte d'apparition des mots dans les documents d'un corpus choisi est utilisé. Une phase d'apprentissage permet de collecter, à partir d'un corpus d'apprentissage, les contextes où les mots ambigus occurrent. Des règles ou des regroupements de contextes similaires sont définis et sont utilisées dans une deuxième phase pour retrouver le sens approprié des mots ambigus apparaissant dans d'autres contextes.

D'autres approches utilisent des dictionnaires (Machine Readable Dictionary) pour utiliser les définitions des mots ambigus et les définitions des mots qui apparaissent dans le même contexte que les mots ambigus pour retrouver leur sens adéquat.

Un mot cible est alors représenté soit par les mots apparaissant dans son contexte, soit par un numéro correspondant à un sens lorsque qu'une ressource sémantique est utilisée.

1.6.1.1 Utilisation d'un corpus comme seule source de connaissance

Travaux de Weiss

Weiss **[Weiss, 1973]** s'appuie sur les approches définies dans Stone **[Stone, 1969]** et Coyaud **[Coyaud, 1968]** basées sur l'exploitation du contexte pour retrouver le sens approprié d'un mot ambigu. L'approche de Weiss consiste à analyser dans un premier temps diverses ambiguïtés et à déterminer les mots clés qui permettent de les résoudre. Ces mots clés sont utilisés pour construire un ensemble de règles et des *motifs de mots* permettant, lorsqu'ils apparaissent dans le contexte d'un mot ambigu, de retrouver le sens approprié à lui attribuer. Le processus de désambiguïsation consiste alors à faire correspondre une phrase contenant un mot ambigu avec l'ensemble de règles. Si une correspondance est trouvée, l'interprétation associée au motif correspondant est considérée comme le sens approprié.

Deux règles sont définies. La première règle appelée *context rules*, nécessite uniquement la cooccurrence du mot ambigu avec les mots clés et la deuxième règle, appelée *template*, tient compte de la structure de la phrase lors de l'appariement. Cette règle prend en compte non seulement la cooccurrence des mots (mot ambigus et mots clés), mais également leur ordre et leur contiguïté.

Considérons le mot *type* qui possède deux sens (*printing* ou synonyme de *variety*).

En appliquant la première règle, si un mot clé comme *print* ou *pica* occurre dans le contexte du mot *type* le sens retenu est celui de *printing*.

En appliquant la deuxième règle, si le mot clé *of* apparaît immédiatement après le mot *type*, le sens retenu est le synonyme de *variety*.

Les règles *template* sont considérés comme plus fiables que les règles *context rules*. Le processus de désambigüisation commence par rechercher les règles *template*. Si aucune correspondance n'est trouvée, les règles de contexte sont recherchées. Au sein de chaque groupe de règles, les meilleures règles sont placées en premières positions. Les ensembles de règles sont alors examinés de haut en bas dans le groupe de règles.

Pour son évaluation, Weiss utilise 5 mots ambigus. Pour chaque mot, il crée un corpus de 50 phrases extraites de la collection ADI[1]. Le corpus est divisé en un ensemble *S1* contenant 20 phrases utilisées pour créer et modifier les règles ainsi que pour trouver le meilleur ordre pour ces règles. Le deuxième ensemble contient le reste des phrases utilisées pour le test. Les résultats ont été concluants montrant une valeur de précision égale à 0,90 et une valeur de rappel égale à 0,96.

[1] Ensemble d'articles courts sur l'automatisation et la communication scientifique publiés par l'Institut Américain de Documentation, (American Documentation Institute), 1963.

Travaux de Schütz

Schütz **[Schütz, 1992]** propose de représenter la sémantique, les mots et les contextes d'un document par des vecteurs. Les dimensions de l'espace sont des mots et les vecteurs initiaux sont déterminés par les mots qui co-occurrent avec le mot à représenter dans une fenêtre de 50 mots. L'espace possède alors plusieurs milliers de dimensions (mots). Pour éviter cette représentation dense, une réduction des dimensions par décomposition en valeur singulière est réalisée. Schütz **[Schütz, 1998]** utilise un corpus non étiqueté dans le processus de désambigüisation. Tous les contextes du mot *m* sont collectés à partir du corpus.

Chaque sens d'un mot ambigu *m* est interprété comme un cluster regroupant des contextes du mot *m* qui sont similaires. Les mots, les contextes et les sens sont représentés dans un espace vectoriel respectivement par des *vecteurs mots*, *vecteurs contextes* et *vecteurs sens*. Les vecteurs mots de *m* sont construits par les mots voisins de *m* dans le corpus. La Figure 1.3 montre les vecteurs mots *juge* et *robe* représentés dans un espace à deux dimensions "legal" et "clothes". La fréquence d'apparition des mots dans le corpus est comme suit : (*legal, juge*)=300, (*legal, robe*)=133, (*clothes, juge*)=75, (*clothes, robe*)=200.

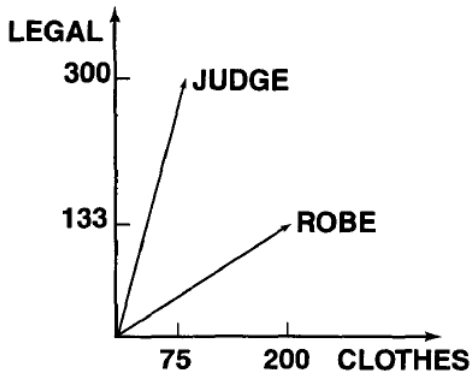

Figure 1.3 Vecteurs mot *juge* et *robe* **[Schütz, 98]**

Les vecteurs contextes de *m* sont calculées sur la base de la somme des vecteurs des mots qui co-occurrent avec *m* dans le contexte et les vecteurs sens sont construits par regroupement des vecteurs contextes similaires en cluster : tous les contextes du mot ambigu sont collectés à partir du corpus. Pour chaque contexte, un vecteur de contexte est calculé. Cet ensemble de vecteurs de contexte est ensuite découpé en un nombre de groupes de clusters (groupe de contextes) en utilisant Buckshot **[Cutting et al., 1992]**, une combinaison de l'algorithme EM et du clustering agglomérative. Les vecteurs sens représentent les *centroïdes* des clusters. La Figure 1.4 montre un exemple de vecteurs contextes et de vecteurs sens.

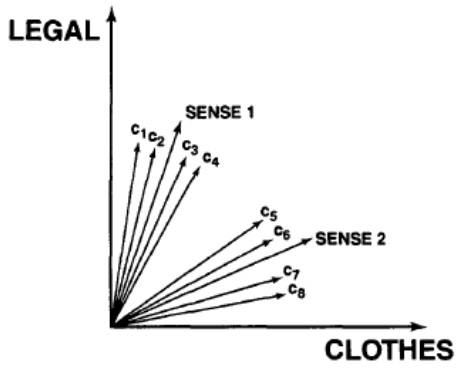

Figure 1.4 Vecteurs contexte et vecteurs sens **[Schütz, 1998]**

Les vecteurs de sens sont construits en regroupant les vecteurs de contexte d'un mot ambigu (*cl, c2, c3, c4, c5, c6, c7 et c8*) et en calculant les vecteurs de sens comme les centroïdes des groupes résultants. Les vecteurs *SENSE1* et *SENSE2* sont les vecteurs sens des clusters {*cl, c2, c3, c4*} et {*c5, c6, c7, c8*}, respectivement.

La désambiguïsation d'une occurrence *Om* d'un mot ambigu *m* se fait en trois étapes :

- Assigner *Om* au vecteur contexte *Vc* lui correspondant en utilisant les vecteurs des mots apparaissant dans son contexte.
- Retrouver tous les vecteurs sens *Vs* du mot *m*.
- Assigner à *Om* le sens dont le vecteur sens *Vs* est le plus proche de *Vc*.

Travaux de Deerwester

Le modèle *Latent Semantic Indexing* (LSI) **[Deerwester et al, 1990]** exploite une matrice dont les colonnes représentent les documents d'une collection et les lignes sont les termes extraits des documents. Les cellules de la matrice contiennent le poids des différents termes dans chaque document. Dans ce modèle, la matrice terme-document est projetée dans un espace de dimension plus faible. Le but de ce modèle est de donner une représentation conceptuelle des documents. Les documents qui partagent des termes co-occurrents ont des représentations proches dans l'espace défini par le modèle. Ainsi cette approche permet de trouver des documents pertinents pour une requête même s'ils ne partagent aucun mot avec cette requête.

LSI décompose la matrice terme-document *W* en valeurs singulières (En mathématiques, le procédé d'algèbre linéaire de décomposition en valeurs singulières d'une matrice est un outil important de factorisation des matrices rectangulaires réelles ou complexes)[1]. La décomposition en valeurs singulières repose sur un théorème qui stipule qu'une matrice rectangulaire *A* peut être exprimée sous la forme du produit de trois matrices : une matrice orthogonale *U*, une matrice diagonale *S* et la transposée d'une matrice *V* tel que :

[1] https://fr.wikipedia.org/wiki/D%C3%A9composition_en_valeurs_singuli%C3%A8res

$$A = U \times S \times V^T \qquad (1.9)$$

Où $UU^T = U^TU = I$, les colonnes de U sont les vecteurs propres orthogonaux de AA^T, les colonnes de V sont les vecteurs propres orthogonaux de A^TA et S est une matrice diagonale qui contient les racines carrées des valeurs propres de V dans l'ordre décroissant.

LSI réduit ainsi l'espace des termes d'indexation. Les documents et les requêtes représentés dans ce nouvel espace ne dépendent plus des termes d'indexation mais des concepts contenus dans les documents.

$$W = Tm \times S \times D^T \qquad (1.10)$$

Où Tm est la matrice termes, D la matrice document et S la matrice des valeurs singulières.

La matrice W est ensuite réduite par une matrice contenant les k plus grandes valeurs singulières. Les documents seront représentés dans un nouvel espace de dimension k.

Le paramètre k est important à définir. Une réduction à un espace de trop grande dimension ne ferait pas émerger suffisamment les liaisons sémantiques entre mots et un trop petit nombre de dimensions conduirait à une perte d'informations.

Le modèle LSI présente de bonnes performances pour un corpus de petite et moyenne taille mais diminuent quand la taille du corpus augmente.

Travaux basés sur les réseaux de neurones

L'utilisation des réseaux de neurones dans le processus de désambiguïsation est très récente. les systèmes supervisés, généralement entraînés sur le SemCor[1] **[Miller et al., 1993]**, mettent en œuvre un réseau de neurones à base de vecteurs de mots et de cellules récurrentes de type LSTM pour prédire le sens d'un mot cible.

Ces cellules dites "à mémoire" sont utilisées pour l'apprentissage automatique sur des séquences de texte et permettent de calculer une sortie en considérant l'élément courant de la séquence, et l'historique passé des cellules précédentes.

Dans le modèle de **[Yuan et al. (2016],** un réseau neuronal à base de cellules LSTM est utilisé comme modèle de langue pour prédire un mot d'une séquence en fonction de son contexte. Leur système est entrainé sur des corpus annotés en sens pour qu'il soit capable de prédire le sens d'un mot en fonction des mots prédits par le modèle de langue. Ensuite, une similarité entre les représentations vectorielles des phrases est calculée entre les phrases annotées et des phrases appartenant à des corpus non annotés pour déterminer les phrases proches. Les annotations en sens sont ensuite propagées de la phrase initialement annotée vers

[1] SemCor est un corpus extrait du corps Brown. Tous les mots dans SemCor sont annotés par un étiqueteur morpho-syntaxique pour définir leur rôle dans les phrases (nom, verbe, adjectif, etc.). Les mots sont désambiguïsés manuellement sur la base des sens définis dans WordNet.

la phrase non annotée. Le processus mis en ouvre ne permet pas d'annoter à la fois tous les mots en entrée. L'annotation des mots se fait indépendamment les uns des autres.

Raganato **[Raganato et al., 2017]** et **[Vial et al., 2018]** proposent d'annoter tous les mots en entrée et abordent le processus de désambiguïsation lexicale comme un processus de classification dans lequel un label est assigné à chaque mot. Ils proposent un modèle à base de LSTM qui apprend directement à prédire un label pour chacun des mots donnés en entrée. Le label à prédire fait partie d'un ensemble de tous les sens possible d'un mot pris soit dans un dictionnaire ainsi que tous les mots observés pendant l'entraînement. Contrairement à l'approche proposée par Raganato, le modèle proposé par Vial peut apprendre non seulement sur des données entièrement annotées, mais également sur des données partiellement annotées.

1.6.1.2. Utilisation des définitions des mots

Travaux de lesk

Lesk **[Lesk, 1986]** est l'un des premiers auteurs à s'intéresser au problème de désambigüisation des mots basée sur les dictionnaires informatisés. De ces dictionnaires, il extrait la définition du mot ambigu et celles de ses mots voisins. Un mot *m* ambigu peut avoir plusieurs sens. A chaque sens correspond une définition d_i^m dans le dictionnaire. Pour chaque définition d_i^m, il calcul le taux de chevauchement des mots appartenant à d_i^m avec les mots appartenant aux définitions des mots voisins de *m*. La définition ayant obtenu le plus grand score est retenue pour désigner le sens correcte du mot *m*.

L'approche de Lesk suppose que des mots ne sont proches sémantiquement que s'ils partagent dans leur définition des mots communs. Dans le cas où les définitions du mot ambigu *m* et de ses voisins *Vm* utilisent des mots différents, aucun rapprochement ne peut se faire entre *m* et *Vm*. Ils sont alors considérés comme sémantiquement différents.

Travaux de Fragos

Fragos **[Fragos et al., 2003]** dans son approche "*chevauchement pondéré*" étend le travail de Lesk. Il s'appuie sur la ressource WordNet pour retrouver, à partir du glossaire de chaque synset, les définitions du mot ambigu et celles des mots apparaissant dans son contexte. Il utilise également les relations d'hyperonymie définies dans la structure de la ressource.

Pour chaque sens *S* d'un mot ambigu *m*, Fragos construit un sac de sens constitué à partir de la définition d_i^m correspondant à *S* et des définitions des hyperonymes des noms et des verbes appartenant à d_i^m. Le même procédé est appliqué pour construire un sac de contexte pour les mots apparaissant dans le contexte de *m*.

Un poids est appliqué à chaque mot en fonction de la profondeur de sa position dans la hiérarchie de WordNet. Ce poids est inversement proportionnel à la profondeur de la hiérarchie à laquelle le synset, correspondant au mot considéré, appartient. Les hyperonymes auront ainsi un poids décroissant à chaque fois que l'on remonte dans la hiérarchie. La

désambigüisation d'un mot est réalisée par un calcul de similarité entre les mots appartenant au sac de sens et au sac de contexte.

1.6.2 Représenter par des concepts

La disponibilité des ressources telles que des thésaurus et des ontologies a fait évoluer la représentation d'un document par des mots vers une représentation conceptuelle. Les descripteurs d'un document sont alors les concepts correspondant aux mots appartenant à son contenu textuel, extraits à partir de ces ressources. Etant donné qu'un concept est souvent représenté par plusieurs mots, il s'agit alors de retrouver les groupes de mots correspondant à ces concepts. L'utilisation des ressources sémantiques présente l'avantage d'extraire du texte des syntagmes qui ont non seulement un sens mais ce dernier est précis puisque les relations sémantiques décrites dans ses ressources permettent de retrouver le sens des mots du texte en fonction de leur contexte d'apparition grâce à un processus de désambiguïsation. Pour que la représentation d'un document par des concepts soit exhaustive, il est nécessaire que les ressources utilisées couvrent entièrement le vocabulaire du domaine représenté par le corpus.

La représentation conceptuelle permet d'améliorer la représentation classique des documents et surmonter les limites d'une représentation par des mots où uniquement la morphologie est mise en avant. Les concepts véhiculent une richesse définie dans la structure qu'ils représentent et permettent ainsi de réaliser un processus de désambigüisation en exploitant les relations reliant les concepts. WordNet est une ressource largement utilisée pour représenter des documents de contenu "général". Pour les documents de contenu plus spécifique tels que des documents du domaine médical, la ressource lexicale MeSch (Médical Subject Heading)[1] est utilisée. Des ontologies de domaine sont utilisées dans le processus de désambigüisation comme l'ontologie construite pour le domaine du sport par Khan **[Khan, 2000**]. Dans ces ressources, les relations de subsomptions sont exploitées pour déterminer la similarité sémantique entre mots. Cela se traduit par le calcul d'une distance sémantique reliant les concepts correspondants aux mots à comparer. Nous citons dans ce qui suit quelques approches d'indexation conceptuelle.

Travaux de Khan

Khan **[Khan, 2000]** définit un processus de désambigüisation en utilisant une ontologie couvrant le domaine du sport. Il applique son approche à des paragraphes décrivant des passages audio dans le domaine du sport. Khan introduit plusieurs définitions et plusieurs formules de calcul. Le processus de désambigüisation se fait en deux étapes : une désambigüisation par la région pour ne retenir qu'une région et les concepts qu'elle contient et une désambigüisation à l'intérieur de la région sélectionnée. Nous décrivons ci-dessous les différentes étapes du processus.

a. Sélection d'une région. Khan commence par définir la notion de région dans l'ontologie comme étant un ensemble de concepts appartenant à une même zone (voisinage)

[1] http://www.nlm.nih.gov/mesh/MBrowser.html

de l'ontologie. Une région, déterminée par la ligue, son équipe et ses joueurs contient un ensemble de concepts. L'idée sous jacente est que des mots clés qui occurrent dans un même contexte déterminent le contexte approprié pour un autre mot clé. Pour l'auteur, le contexte est défini par une région. La région qui obtient le plus grand score sera retenue et utilisée pour annoter le document. Si une région est sélectionnée, tous les autres concepts sélectionnés qui appartiennent à d'autres régions sont éliminés. Pour chaque région, l'auteur calcule un score selon l'équation (1.11).

$$\text{Score (région)} = \sum score(ci) \qquad (1.11)$$

Les *ci*, sont les concepts sélectionnés qui apparaissent dans la région considérée.

La Figure 1.5 donne un exemple de régions issues de l'ontologie définie par Khan. Les ensembles de concepts représentant chaque région sont disjoints.

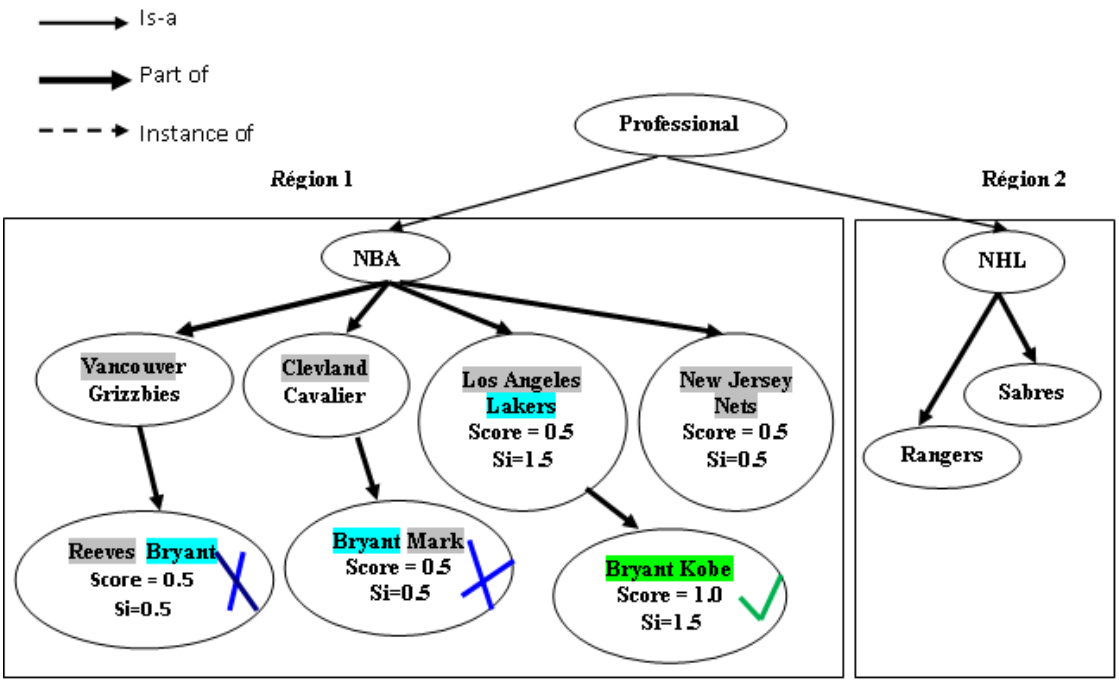

Figure 1.5 Différentes régions de l'ontologie et désambiguïsation des concepts dans une région [Khan, 2000]

Considérons le paragraphe audio donné en exemple par Khan :

Lakers *keep grooving with 8th straight win.* ***Kobe Bryant*** *scores 21 points as the* ***Lakers*** *remain perfect on their* ***eastern*** *road trip with a 97-89 triumph over the* ***Nets****.* ***Bryant*** *discussed the eight game win streak and his performance in the All Star game.*

La région *NBA* est retenue car elle contient un plus grand nombre de concepts relativement au contenu du paragraphe. Les concepts sélectionnés sont ceux qui apparaissent dans la région *NBA* retenue. Le score d'un concept c*i* est calculé comme suit.

b. Score d'un concept. Le score et le score propagé permettent de définir la région à retenir pour un paragraphe et de sélectionner et désambiguïser les mots ambigus.

Un synonyme *lj*, appelé élément, est représenté par un ensemble de mots clés et un document, appelé *objet,* est représenté par un ensemble de mots clés.

Pour un concept c*i* dont les synonymes sont *(l1,l2,…….lj,..ln)* et un document *d*, un appariement (rapprochement) est calculé entre *d* et *ci* et un score est calculé pour chaque élément *li* appartenant à *ci.*

- **Extraction des termes de *d***. Les termes sont extraits par appariement des mots clés de *d* avec les mots clés de chaque élément *lj*, (j=1,n), correspondant au concept *ci.*

- **Calcul de EScore** de l'élément *lj* appartenant au concept *ci* selon l'équation (1.12).

$$\text{EScore}_{ij} = \frac{\#\, mots\ clés\ \ de\ l_j\ \ appariés}{\#\, mots\ clés\ \ \in\ \ l_j} \qquad (1.12)$$

$EScore_{ij}$ permet de retrouver parmi les mots clés de *lj*, le nombre de mots clés appartenant à l'élément *lj* correspondant aux mots clés de *d*.

- **Calcul du score $Score_i$ du concept *Ci***. $Score_i$ est le score du concept *ci*. Il est donné par le plus grand $Escore_{ij}$ obtenu par l'un de ses éléments *lj* selon l'équation (1.13).

$$Score_i = \text{Max EScore}_{ij} \quad avec\ 1 \le j \le n \qquad (1.13)$$

Par exemple, pour le document *d* (*Lakers, Bryant, Kobe*), le calcul du score des concepts de la Figure 1.5 selon l'équation (1.12) est comme suit :

Score concept (*los Angeles lakers*) =1/2=0.5

Score concept (*Bryant Kobe*) =2/2=1.0

Score concept (*Bryant Mark*) =1/2=0.5

c. Concepts corrélés. L'auteur définit la notion de corrélation (association) entre deux concepts *ci* et *cj* et l'exploite dans la propagation du score de façon à ce que les concepts corrélés obtiennent un plus grand score *Si* que les concepts non corrélés.

ci et *cj* sont dits disjoints (pas d'association, pas de corrélation) en raison de leur association au concept parent à travers les liens Is-a. Ils sont dits corrélés s'ils sont reliés par une relation de type "*instance of*" ou "*part of*". La Figure 1.5 et la Figure 1.6 montrent les relations liant les différents concepts.

(*Los Angeles Lakers*) est corrélé avec (*Bryant Kobe*) car relié par la relation "*part of* ".

(*Los Angeles Lakers*) n'est pas corrélé avec (*Bryant Mark*) car aucune relation de type "*part of* " ou "*instance of* " ne les lie.

(*Los Angeles Lakers*) n'est pas corrélé avec (*Reeves Bryant*) car aucune relation de type "*part of* " ou "*instance of* " ne les lie.

d. Propagation du score Si. Le score d'un concept est mis à jour grâce aux concepts auxquels il est corrélé. Ce nouveau score est appelé *Si* et il est calculé par l'équation (1.14). La Figure 1.5 et la Figure 1.6 montrent les valeurs des scores et des scores propagés d'un ensemble de concepts corrélés.

$$\mathrm{Si} = \mathrm{Score}_{\,i} + \sum_{j=1}^{n} \frac{Score_{\,j}}{SD(ci,cj)} \qquad (1.14)$$

c*i* et *cj* sont deux concepts corrélés, $1 \leq j \leq n$). *SD(ci,cj)* représente la distance sémantique entre *ci* et *cj*.

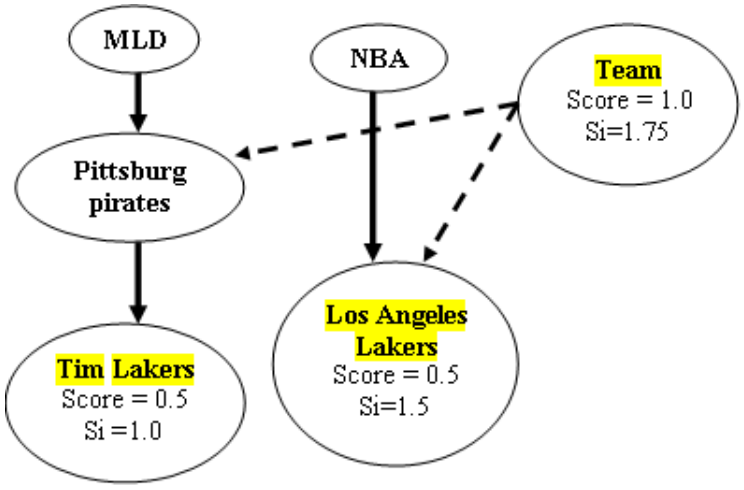

Figure 1.6 Score et score propagé des concepts [Khan, 2000].

SD(team, los Angeles Lakers)=1

SD(team, TimLaker)=2

Si(team)=Score(team)+Score(LosAngelesLakers)/1+score(Timlakers)/2
=1+0.5+0.5/2=1.75.

Si (LosAngelesLakers) = score(LosAngelesLakers)+Score(team)/1 = 0.5+1.0/1=1.5

Si (TimLaker) = score(TimLaker)+Score(team)/2 = 0.5+1.0/2=1.0

Khan considère que deux concepts *ci* et *cj* sont "plus corrélés" (plus proche sémantiquement) lorsque la distance reliant *ci* et *cj* est petite. Cela se traduit dans le calcul de la propagation du score. Deux concepts corrélés ayant une distance sémantique *SD* supérieure à 1 auront un score de propagation *Si* inférieur à ceux ayant les mêmes scores mais avec une distance sémantique égale à 1.

e. Désambiguïsation

- *Désambiguïser par la région*. Si un mot clé du document correspond à deux concepts appartenant à deux régions différentes, le concept retenu est celui qui appartient à la région retenue par le calcul du score des régions.

- *Désambiguïser à l'intérieur d'une même région*. Dans une même région, un mot clé du document peut correspondre à plusieurs concepts. Un processus de désambiguïsation est alors nécessaire. Par exemple, dans la région *NBA* sélectionnée, le mot-clé *Bryant* est associé à plusieurs concepts sélectionnés. Il faut donc le désambiguïser.

Les concepts sélectionnés et corrélés entre eux vont avoir un plus grand score et une plus forte probabilité d'être retenus que des concepts non corrélés. Les concepts (correspondant au mot ambigu) non corrélés sont éliminés.

Exemple 1

Nous reprenons la Figure 1.5 et nous considérons un document *d(Bryant, Lakers)* contenant les mots clés *Bryant* et *Lakers*.

- *Lakers* n'est pas ambigu, il correspond au concept (*Los Angeles Lakers*). Ce concept est sélectionné et retenu.
- *Bryant* est ambigu, il correspond aux concepts (*Bryant Mark*), (*Bryant Kobe*) et (*Reever Bryant*).
- (*Bryant Mark*), (*Reever Bryant*) ne sont pas corrélés avec le concept (Los *Angeles Lakers*). Ils sont alors éliminés
- (*Bryant Kobe*) est corrélé avec le concept (*Los Angeles Lakers*). Il sera alors retenu.

Exemple 2

Considérons la requête "*Please tell me about* ***team Lakers***" et la Figure 1.6. Les concepts *team*, *Los Angeles Laker* et T*im Laker* (joueur de baseball de l'équipe *Pittsburgh Pirates*) sont sélectionnés. Pour savoir quel concept retenir parmi les concepts L*os Angeles Laker* et T*im Laker,* un calcul du score est effectué. Nous obtenons ***Si*** (*LosAngelesLakers*) =1.5 et ***Si*** (*TimLaker*) =1.0. Le concept *LosAngelesLakers* obtient ainsi un score plus élevé lui permettant d'être retenu.

Dans cette approche, pour un mot clé *t*, l'auteur considère tous les concepts contenant *t* comme des concepts candidats et les sélectionne tous. Pour retenir des concepts, il s'appuie sur un seuil. Les concepts qui ont un score propagé *Si* inférieur à ce seuil ne sont pas retenus. Dans ses expérimentations, il montre que la valeur du seuil influe sur les résultats obtenus : soit des concepts pertinents sont rejetés, soit des concepts non pertinents sont retenus.

Travaux de Baziz

Baziz **[Baziz, 2005a]** définit un processus de désambiguïsation qui met en avant les liens entre les différents termes d'un document. Le sens approprié d'un terme ambigu est déterminé en fonction des liens qui le relient aux autres termes du document.

Le modèle *DocCore* proposé par baziz est un modèle de représentation des documents s'inspirant des réseaux sémantiques et utilisant l'ontologie générique WordNet. L'auteur définit pour chaque document du corpus un noyau sémantique du document. Les concepts des documents sont construits de façon automatique et les liens entre ces concepts sont pondérés en fonction de la proximité sémantique (similarité sémantique) existant entre ces concepts.

Le contenu sémantique d'un document est obtenu en projetant les termes du document sur WordNet afin d'extraire les concepts les plus représentatifs. Le choix de ces concepts est fait sur la base de deux critères. Un premier critère concerne la co-occurrence appelé *cf*×*idf* utilisé pour extraire les concepts importants et un deuxième critère représente la similarité sémantique qui permet de désambiguïser les termes. Nous détaillons les différentes étapes de construction du réseau sémantique d'un document.

a. Extraction des concepts candidats. Il s'agit de projeter les termes d'un document sur WordNet. Un concept candidat est constitué par des mots adjacents dans le texte. Ce concept candidat est recherché dans WordNet. S'il ne correspond à aucune entrée de WordNet, sa forme de base est utilisée. Pour la combinaison des mots, le terme le plus long qui correspond à un concept est retenu.

b. Pondération des termes. La mesure *cf*×*idf*, une variante de *tf*×*idf*, est utilisée pour pondérer les termes (concepts) extraits des documents. Pour chaque terme *t* composé de *n* mots, sa fréquence dans un document dépend du nombre d'occurrences du terme lui-même et de celui de ses sous-termes dérivés. Les termes importants extraits du document sont utilisés pour construire le noyau sémantique de ce document.

c. Calcul de similarité entre concepts. Une phase de désambiguïsation est nécessaire puisqu'un terme peut avoir plusieurs sens (être associé à plusieurs concepts) dans WordNet. Des mesures de similarité entre les différents sens des termes sont calculées pour sélectionner le meilleur concept. L'auteur utilise des mesures de proximités sémantiques existant dans la littérature **[Resnik, 1995][Leacock et al., 1998]**. Il obtient des valeurs de proximité sémantique entre tous les concepts candidats. Ces concepts candidats sont les différents sens possibles pour lesquels un terme extrait peut être affecté.

d. Construction du noyau sémantique. Soit *DT* l'ensemble des termes *Ti* extraits à l'étape (a).

$$D_T = \{T_1, T_2, ..., T_m\}$$

Chaque terme T_i appartenant à l'ensemble D_T peut avoir plusieurs sens S_i représentés par des synsets de WordNet.

$$S_i = \left\{C_1^i, C_2^i, ..., C_n^i\right\}$$

Le terme *Ti* a $|Si| = n$ sens. Le nombre total de réseaux sémantiques possibles *Nb_SN* dépend du nombre de sens que chaque terme de DT peut avoir :

Pour choisir, parmi les *Nb_SN* configurations possibles de réseaux sémantiques, celle qui représente au mieux le contenu d'un document, un processus de désambiguïsation des termes extraits à l'étape (a) est réalisé.

Pour réaliser cette désambiguïsation, l'auteur se base sur l'hypothèse suivante : le concept (sens) le plus adéquat pour un terme donné est celui qui a le plus de liens avec les autres concepts du même document auquel il appartient. Cette règle est appliquée à tous les termes de *DT*. On obtient ainsi des termes qui se désambiguïsent mutuellement par rapport au contexte du document.

L'auteur affecte à chaque concept candidat (ou sens d'un terme) un score (*C_score*). Ce score est égal à la somme des valeurs de similarité qu'il a obtenu avec les autres concepts candidats. Pour un terme *Ti*, le score de son k^{ème} sens est calculé par l'équation (1.15).

$$C\_score(C_k^i) = \sum_{\substack{l \in [1,m], l \neq i \\ j \in [1,n]}} P_{i,l}(C_k^i, C_j^l) \qquad (1.15)$$

$P_{i,l,}$ désigne la proximité sémantique. *m* représente le nombre de termes de *DT* et *n* le nombre de sens qui est propre à chaque *Ti.*

Le concept *Ci* qui représente le mieux le sens du terme *Ti* est celui qui maximise *C_score*. Il est noté *Best_score*(*Ti*). Les concepts (sens) sélectionnés représenteront les nœuds du réseau sémantique du document. Le poids des nœuds est donné par *Best_score(Ti)* et les valeurs des liens sont représentées par les valeurs de proximité sémantique calculée à l'étape (c).

Travaux de Kolt

Kolt **[Kolt et al., 2009a]** exploite WordNet pour retrouver le sens des mots et propose d'améliorer le processus défini par Lesk **[Lesk, 1986]** en utilisant des définitions supplémentaires basées sur la relation *hyperonyme / Hyponyme* pour enrichir davantage les sacs de mots. Les définitions des sens du mot à désambigüiser et les définitions liées aux *hyperonymes* des noms et verbes trouvés pour les mots appartenant au contexte du mot à

désambigüiser sont utilisés. IL retrouve le sens d'un mot ambigu dans une phrase à travers les verbes se trouvant dans son contexte et par l'utilisation des relations sémantiques supplémentaires comme *ability* (habilité), *capability* (capacité, aptitude) et *function* (fonction). Ces relations existent dans WordNet en langue Hindi, *Hindi WordNet*. Kolt les a rajoutés manuellement dans la version anglaise de WordNet afin de les utiliser dans son approche. Nous donnons ci-dessous quelques exemples de désambigüisation utilisés par Kolt.

- Utilisation de la relation d'hyperonymie (fils de)

Phrase : *He ate* ***dates***

Mot : *Date*

Analyse : Le mot *date* est annoté comme nom dans de la phrase et il possède 8 sens avec la relation d'hyperonymie. De plus, dans WordNet, date apparait comme fils du synset *fruit* et le verbe *ate* apparaît dans le contexte de date. Le sens retenu est celui du fruit.

Sens retenu : Date-- (sweet edible fruit of the date palm with a single long woody seed).

- Utilisation des relations de meronymie et d'holonymie (Partie-tout)

Phrase : *The* ***trunk*** *is the main structural member of a tree that supports the branches*.

Mot : ***Trunk.***

Analyse : Le mot *trunk* annoté comme nom dans la phrase : possède 5 sens avec la relation d'holonymie. Avec l'apparition du nom *tree* dans le contexte de *trunk,* le sens retenu est celui de tronc.

Sens retenu : PART OF : {12934526} <noun.plant> tree#1 -- (a tall perennial woody plant having a main trunk and branches forming a distinct elevated crown; includes both gymnosperms and angiosperms).

- Utilisation de la relation Ability

Ce lien spécifie les caractéristiques héritées d'un concept nominal (transmises par un concept nominal).

Phrase : *A* ***crane*** *was flying across the river*.

Mot : *Crane*

Analyse : Le mot *crane* annoté comme nom dans la phrase possède 5 sens. Grâce à la présence du verbe *flying* dans le contexte de *crane*, le sens retenu est celui de oiseau.

Sens retenu : Crane -- (large long-necked wading bird of marshes and plains in many parts of the world).

- Utilisation de la relation Capability

Ce lien spécifie les caractéristiques acquises d'un concept nominal.

Phrase : *The **chair** asked members about their progress*.

Mot : *Chair*

Analyse : Le mot *chair* annoté comme nom dans la phrase possède 4 sens. Puisque une personne à la capacité de demander, le sens retenu est :

Sens retenu : president, chairman, chairwoman, chair, chairperson -- (the officer who presides at the meetings of an organization; "address your remarks to the chairperson").

- Utilisation du lien Function

Phrase : *Please keep the papers in the **file***.

Mot : F*iles*

Analyse : Le mot *files,* annoté comme nom dans la phrase possède 4 sens et le sens retenu est :

Sens retenu : file, file cabinet, filing cabinet -- (office furniture consisting of a container for keeping papers in order).

Travaux de Wang

Wang **[Wang et al., 2012]** définit une approche permettant de représenter un document sous forme d'une structure composée de concepts et de relations entre concepts extraits à partir de WordNet. Pour retrouver le concept approprié pour un terme *t*, il construit le modèle *W-C* (words-concepts), représenté par une matrice symétrique *UsC*. Sachant qu'un terme *t* (mot ou groupes de mots) du document peut correspondre aux mots constituant le label de plusieurs concepts, la matrice permet de décider quel concept retentir pour représenter *t*.

$$\mathbf{U_sC} = \begin{pmatrix} u_sc_{11} & u_sc_{12} & \dots & u_sc_{1n} \\ u_sc_{21} & u_sc_{22} & \dots & u_sc_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_sc_{n1} & u_sc_{n2} & \dots & u_sc_{nn} \end{pmatrix}$$

Wang construit une matrice *UsC* pour chaque concept *c* candidat pour *t*. Les lignes et les colonnes de *UsC* représentent les mots *ti* (i=1,n) constituant *c*. UsC_{ij} correspond à la fréquence d'apparition simultanée des mots *ti* et *tj* dans le même paragraphe. La matrice est ensuite mise à jour en parcourant tout le document. La ligne *i* de *UsC* indique la probabilité d'apparition simultanée des mots *ti* et *tj*, (j=1,n) dans le même document. La matrice *UsC* dénote la pertinence de *t* vis-à-vis du concept *c* et permet de décider du concept qui sera retenu pour représenter *t*.

Wang construit ensuite une matrice *C-D* dont les éléments représentent les relations entre les mots appartenant au même paragraphe. Ces relations sont pondérées en fonction des liens qui les relient dans WordNet. Ces valeurs sont exploitées pour déterminer la pertinence d'un document vis-à-vis d'une requête.

Travaux de kolt

Plusieurs approches se sont intéressées au rôle que peut jouer l'utilisation des domaines dans le processus de désambigüisation. L'hypothèse de base est que la notion de domaine constitue un moyen naturel de retrouver les relations sémantiques entre les sens des mots et permet de réduire le nombre de sens de ces derniers aux seuls sens définis dans le domaine de connaissance considéré.

Kolt **[Kolte et al., 2009b]** s'appuie sur les domaines définis dans WordNet domains afin de déterminer le sens approprié d'un mot ambigu *m*. Il se base sur l'hypothèse que les mots d'une phrase permettent de déterminer le domaine de la phrase. Son processus détermine le domaine du mot ambigu et le sens correspondant à ce domaine est retenu comme étant le sens approprié pour ce mot. Il utilise le contexte local du mot ambigu *m,* représenté par une phrase, et WordNet Domains. Pour chaque phrase, Il applique un prétraitement pour déterminer le rôle syntaxique des termes dans la phrase. Les termes d'une phrase sont retenus pour former un premier ensemble *b1*. A partir de *b1*, il crée un ensemble *b2* qui contient tous les domaines, extraits de WordNet Domains, où un terme de la phrase à un sens. Dans une dernière étape, les domaines du mot ambigu *m*, figurant dans *b2,* sont insérés dans un troisième ensemble *b3*. Chaque domaine de l'ensemble *b3* aura un score indiquant le nombre de fois que ce domaine est attribué aux termes de la phrase considérée. Le domaine de *b3* qui obtient le plus grand score (à l'exception de factotum) est retenu et le sens défini dans ce domaine est retenu comme étant le sens correct de *m*.

Par exemple, dans la phrase "*The virus infected all files on the hard disk*.", pour désambigüiser le nom *virus*, les ensemble *b1, b2* et *b3* obtenus sont représentés par la Figure 1.7.

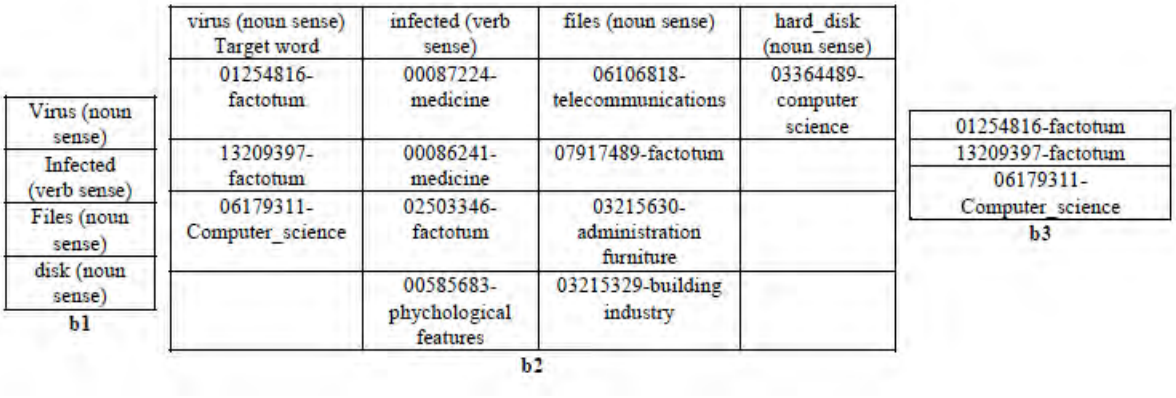

Figure 1.7 Contenu des ensemble *b1*, *b2* et *b3* **[Kolte et al., 2009b]**

Le domaine *computer_science* obtient le plus grand score pour la phrase considéré. Il est alors retenu pour représenter le domaine de la phrase et le sens 06179311 est retenu pour le mot *virus*.

1.6.3 Bilan

Nous avons présenté dans cette section, différents travaux visant à donner une représentation sémantique des documents. Ces approches se divisent en deux catégories : une première catégorie retrouve le sens d'un mot cible en fonction des mots qui apparaissent dans son contexte. La seconde catégorie recherche une correspondance mots-concept en exploitant les relations définies dans une ressource sémantique externe.

Pour ces approches, seuls les mots explicitement cités dan le texte représentent le contenu des documents. Même si ces représentations prennent en charge le problème de la synonymie et de la polysémie, elles ne permettent pas de retrouver des liens sémantiques entre documents utilisant des mots différents mais proches dans le sens. Par exemple, un lecteur recherchant un document sur les *carrés* pourrait être intéressé par un document abordant le sujet des *losanges* ou le sujet des *rectangles*.

1.7 Conclusion

Dans ce chapitre, nous avons présenté la notion de connaissance et ses différentes représentations à travers des collections de documents et des ressources sémantiques. Nous avons décrit les différentes formes que peut prendre un descripteur de document et avons cité différentes méthodes d'identification des descripteurs dans les documents. Nous avons ensuite abordé les approches d'indexation sémantique des documents.

Nous nous intéressons particulièrement à la représentation des connaissances contenus dans les documents textuels. Le problème récurrent dans le processus de représentation d'un document par un ensemble de mots est la présence d'ambigüités inhérentes à certains termes du langage. Retrouver alors le sens des mots du texte pour reproduire fidèlement la sémantique véhiculée par son contenu est une problématique sur laquelle plusieurs approches se sont penchées. Pour construire une représentation sémantique des documents, certaines approches utilisent seulement un corpus pour déterminer le sens des mots alors que d'autres

exploitent des ressources sémantiques pour désambigüiser les termes ambigus présents dans le texte.

La représentation d'un document basée sur le sens des mots, exploite la notion de co-occurrence. Le sens du mot cible est déterminée en fonction de son contexte d'apparition. Ce contexte prend en compte le voisinage du mot cible défini dans une fenêtre de quelques mots en ignorant son appartenance au reste du document. D'autres approches retrouvent le sens des mots en utilisant la définition du mot cible et celle de ses mots voisin à travers des dictionnaires informatisées ou des ressources sémantiques externes. Les approches utilisant les définitions des mots présentent des limites, puisque l'hypothèse sur laquelle ces approches se basent stipule que des mots ne sont proches sémantiquement que s'ils partagent dans leur définitions les mêmes mots. Les approches plus récentes, exploitant les réseaux de neurones, nécessitent l'utilisation de corpus annotés pour entrainer leur système. L'inconvénient de ces approches c'est le manque de disponibilité de ces corpus annotés quand les documents traités sont des documents techniques.

L'indexation conceptuelle permet de trouver une correspondance entre les mots d'un document et les concepts d'un thésaurus ou d'une ontologie. Pour Baziz, l'extraction des concepts pour un document ne tient pas compte de l'emplacement des mots dans le texte. Les sens des mots, et donc les concepts retenus, sont déduits uniquement en fonction des relations qui les relient dans WordNet. Cela signifie que deux documents utilisant les même mots auront une même représentation conceptuelle, quelque soit leur emplacement dans le texte. Pourtant deux documents composés des mêmes mots n'ont pas toujours le même sens.

Les différentes approches décrites représentent un document par le sens des mots ou par les concepts correspondant aux mots explicitement citées dans leur contenu. Ces représentations posent le même problème qu'une représentation classique basée sur des mots puisque seuls les mots explicitement cités dans les documents sont retenus. Aucun rapprochement ne peut alors être déduit pour deux documents utilisant des mots différents mais proches sémantiquement. Les liens entre termes à travers le thème dominant d'un document ne sont pas mis en avant. La majorité des mots non vides participent à la représentation d'un document alors que tous ces mots n'ont pas forcément un sens pour le thème principal du document.

Notre objectif est de donner une représentation sémantique des documents en nous basant sur les termes significatifs pour le sujet du document. Ces derniers incluent ceux qui sont explicitement cités dans son contenu et ceux déduits en exploitant les informations implicites qui y sont décrites. Nous étendons la notion de contexte introduite par Wang **[Wang et al., 2012]** à tout le document. Nous combinons pour cela, le contexte d'apparition des termes dans le document entier avec la notion de domaine. Nous utilisons un ensemble d'ontologies de domaine sur lesquelles le contenu des documents est projeté et nous exploitons simultanément les liens reliant les termes dans le document et dans les ontologies.

La pondération *tf-idf* mesure le pouvoir discriminant des mots. Cette pondération ne met pas l'accent sur leur pouvoir représentatif vis-à-vis d'un domaine. Dans nos travaux, nous définissons un poids qui permet d'extraire les termes en fonction du thème dominant abordé dans le contenu des documents.

L'objectif de nos travaux est de calculer une similarité sémantique entre documents. Après avoir donné une représentation sémantique du contenu sémantique des documents, nous comparons leurs représentations. Dans le chapitre suivant, nous présentons un certain nombre d'approches qui existent dans la littérature dont l'objectif est de calculer la similarité entre documents pouvant apporter des réponses à divers applications.

Chapitre 2
Similarité des textes

2.1 Introduction

Calculer la similarité sémantique entre entités linguistiques est au cœur de nombreuses applications de traitement du langage naturel. Les entités linguistiques à comparer peuvent être des mots, des phrases ou des documents entiers. Au niveau des mots, la similarité peut être exploitée par des applications telles que la désambiguïsation des sens du mot (WSD) **[Schütz, 1998] [Yaworsky, 1993] [Navigli, 2009]**, l'expansion des requêtes **[Voorhees, 1994],** ou encore l'alignement des ressources lexicales ou ontologiques **[Pilehvar et al., 2014]**. Au niveau des textes (phrases ou documents entiers), elle est notamment utilisé par les systèmes de traduction automatique **[Lavie et al., 2009]**, la reconnaissance de paraphrases **[Glickman et al., 2003][Pilehvar et al., 2015][Mohhebi et al., 2016]**, les résumées automatiques **[Wang et al., 2008][Rusu et al., 2009][Ferreira et al., 2016]**, la recherche d'information **[Baziz et al., 2005b][Otegi et al., 2015]** et la classification des documents **[Hotho et al., 2002] [Jaillet et al., 2003]**. En recherche d'information, un score de pertinence est calculé pour un document relativement à une requête représentant un besoin d'un utilisateur. Dans le domaine de la classification des documents, la similarité entre documents est calculée pour permettre leur regroupement au sein de classes prédéfinies ou de clusters. Les diverses approches s'appuient sur un corpus de textes et/ou sur une ressource sémantique externe où WordNet et Wikipédia tiennent une place prépondérante.

Nous présentons dans ce chapitre un certain nombre d'approches liées à notre problématique. Ces approches appartiennent à des domaines différents et la similarité calculée dépend de la granularité du texte considéré définissant deux niveaux : niveau mot et niveau texte.

2.2 Similarité des mots

Le calcul de la similarité des mots est nécessaire pour plusieurs applications ayant recours au traitement du langage naturel comme c'est le cas pour le processus de désambiguïsation (WSD) **[Resnik, 1999]** et l'extraction automatique de concepts de thésaurus **[Curran, 2002]**. La similarité des mots est également utilisée dans les tâches relatives au Web telle que l'annotation de pages Web **[Cimano et al., 2004]**. Nous distinguons deux classes d'approches qui définissent des mesures de similarités des mots : les approches basées sur le contexte des mots et les approches basées sur une ressource externe.

2.2.1 Les approches contextuelles

Les approches contextuelles ou distributionnelles considèrent un mot relativement à son contexte d'apparition. La seule ressource utilisée est un corpus de textes. Ces approches se basent sur l'hypothèse que l'information contextuelle donne une bonne approximation du sens des mots puisque des mots similaires ont souvent les mêmes distributions contextuelles **[Miller et al., 1991] [Harris, 1954]**. Pour Rubenstein **[Rubenstein et al., 1965]** la proportion du nombre de mots communs aux contextes de deux mots *m1* et *m2* indique à quel point *m1* et *m2* sont similaire dans le sens.

Le contexte d'un mot "cible" est déterminé par la "portion" de texte où ce mot apparaît **[Sahlgren, 2006].** Il est défini selon plusieurs critères.

- Le contexte peut est défini par une fenêtre graphique de mots (mots apparaissant dans le voisinage du mot cible) **[Lund et al., 1996].** La taille de la fenêtre a un impact sur les performances des systèmes. Par conséquent, la fenêtre ne doit être ni trop étroite, ni trop large de façon à trouver le meilleur compromis entre la spécificité et la dispersion des données **[Rapp, 2003]**.

Dans l'approche *LSI* **(***Latent Semantic Indexing*) **[Deerwester et al., 1990]]** (cf. chapitre1, section 1.6.1) Deerwester exploite la co-occurrence des mots pour retrouver les liens sémantiques entre ces mots. *LSI* construit une matrice de co-occurrence des mots où chaque colonne représente un contexte définit par le contenu d'un document. Pour mesurer la similarité des mots, le document entier ne peut constituer la taille idéale pour représenter le contexte d'un mot. Une variante de l'analyse *LSI,* appelée *HAL* (*Hyperspace Analog to Language*) **[Burgess et al., 1998]** est utilisée pour construire une matrice *mots*×*mots* (au lieu d'une matrice mots × documents). Un mot cible est alors représenté par un vecteur dont les éléments sont extraits de son contexte. Les lignes de la matrice correspondent aux mots cibles et les colonnes aux contextes de ces mots cibles.

- Le contexte peut contenir uniquement les mots liés syntaxiquement au mot cible. Les relations de dépendances syntaxiques sont construites sous la forme de triplets (mot cible, relation syntaxique, contexte) **[Lin, 1998b]**[**Curran, 2003]**.

Lin **[Lin, 1998b]** extrait, à partir d'un corpus, des triplets de dépendance (*w1,r,w2*) représentant deux mots *w1* et *w2* reliés par la relation syntaxique *r* dans une phrase. Dans la phrase "*I have a brown dog*", les triplets de dépendance sont par exemple (*have subj I*), (*I subj-of have*), (*dog obj-of have*) etc. Un mot *w* est décrit par tous les triplets de dépendance qui correspondent au patron (*w,*,**) ((*w,*,**) représente l'ensemble des contextes pour un mot cible *w*) et leur fréquence dans le corpus. Les triplets de dépendance donnés ci-après sont extraits de l'ensemble décrivant le mot *cell* :

(*cell, subj-of, absorb*) (fréquence =1)

(*cell, subj-of, adapt*) (fréquence =1)

(*cell, obj-of, attack*) (fréquence =6)

La similarité de deux mots est calculée sur la base des triplets de dépendance apparaissant dans leurs descriptions.

- Les approches "*topic-modèl*" modélisent un mot comme une distribution de probabilités sur un ensemble de sujets. *LDA (Latent Dirichlet Allocation)* **[Steyvers et al., 2007]** en est un exemple. *LDA* est un modèle Bayésien qui se base sur l'hypothèse qu'un document *d* couvre un ensemble de topics (sujets, thème) *t* et que chaque mot *w* de *d* doit être assigné à l'un des thèmes *t* du document. Cette approche permet d'apprendre les thèmes représentés dans chaque document et les mots associés à ces thèmes.

Initialement, un thème est attribué aléatoirement à chaque mot de chaque document selon une distribution de Dirichlet sur un ensemble de thèmes dont la taille est fixée au préalable. Ensuite, pour tout document *d,* le thème assigné à chaque mot *w* de *d* est mis à jour en calculant la probabilité *P(w)* que le thème *t* génère le mot *w* dans le document *d*, selon l'équation (2.1). Le nouveau thème est celui qui aura la plus forte probabilité de générer le mot *w* dans le document *d*.

$$P(w) = P(t\backslash d) \times P(w\backslash t) \qquad (2.1)$$

où :
P(t \ d) représente la probabilité que le document *d* soit assigné au thème *t*,
P(w\ t) représente la probabilité que le thème *t* dans le corpus soit assigné au mot *w*.

- Le contenu textuel structuré des ressources lexicales spécifiques telles que Wikipedia a également été utilisé pour la similarité distributionnelle des mots. Gabrilovich **[Gabrilovich et al., 2007]** propose une approche appelée *ESA (Explicit Semantic Analysis)* pour représenter le contenu du texte. Contrairement à l'approche *LSI* qui utilise des concepts "latents", Gabrilovich définit des concepts manifestes ancrés dans la cognition humaine. Ces concepts sont les articles définis dans Wikipédia. Il s'appuie sur l'hypothèse que l'être humain ne juge pas un texte uniquement sur la base des mots qui le composent. Ces mots déclenchent des raisonnements qui impliquent l'exploitation d'un contexte défini par les connaissances et les

expériences des individus. Il exploite ainsi la richesse définie dans les articles de Wikipédia pour représenter les mots appartenant à un texte.

Selon cette approche, chaque texte *T* est représenté par les concepts qui représentent le mieux son contenu. Pour chaque concept de Wikipédia (article), un mot *wi* de *T* est représenté par un poids calculé sur la base de *tf-idf* **[Salton et al., 1983]** déterminant l'importance de ce mot relativement au concept considéré. Un score de pertinence est calculé pour chaque concept de Wikipédia relativement à *T.* Ce score est basé sur la somme des poids de tous les mots *wi* appartenant au texte *T*. Le vecteur d'interprétation sémantique *V* pour le texte *T* est un vecteur de taille *N* (*N* étant le nombre de concepts de Wikipédia retenus pour *T*) dont les composantes représentent le score de pertinence de chaque concept relativement à *T*.

2.2.2 Les approches basées sur une ressource structurée

Plusieurs méthodes permettant de calculer la proximité des mots, à travers la distance sémantique reliant les nœuds correspondant à ces mots, sont proposées dans la littérature. Elles sont réparties en deux catégories. Une première catégorie se base uniquement sur la structure hiérarchique de la ressource utilisée **[Leacock et al., 1998][Rada et al., 1989][Wu et al., 1994][Howe, 2009]**, une autre catégorie utilise en plus de la structure hiérarchique, le contenu informationnel (des statistiques relatives au nœuds de cette structure) **[Lin, 1998a][Resnik, 1995]**. Les ressources utilisées sont principalement WordNet, Wikipédia **[Ponzetto et al., 2007][Milne et al., 2008]** et les wiktionnaires **[Pilehvar et al., 2014]**.

2.2.2.1 Les approches basées sur la structure hiérarchique

Les approches s'appuyant sur la structure hiérarchique calculent des distances séparant les différents nœuds en utilisant les relations " is-a", sachant qu'un nœud dans une taxonomie correspond généralement à une sémantique.

- Rada [Rada et al., 1989] propose une approche qui s'appuie sur le comptage d'arcs. Il définit la distance entre deux nœuds *N1* et *N2* comme le nombre minimum d'arcs reliant *N1* et *N2*. Cela représente le chemin le plus court entre *N1* et *N2*.

$$dist(N1, N2) = Min_{arcs}\,(N1, N2) \qquad (2.2)$$

- Wu [Wu et al., 1994] Propose une mesure basée sur la notion "*the least common super-concept"* c'est à dire le concept commun le plus éloigné de la racine. Cette mesure reflète que plus on descend dans la hiérarchie, plus les nœuds sont proches sémantiquement.

$$Sim(N1, N2) = \frac{2 \times depth(N)}{depth(N1) + depth(N2)} \qquad (2.3)$$

où *N* est le concept commun à *N1* et *N2* le plus éloigné de la racine, et *depth(Ni)* représente le nombre d'arcs reliant *Ni* et la racine.

- Leacock [Leacock et al., 1998] se base sur la mesure de Rada et propose une mesure qui normalise la distance entre deux nœuds en utilisant la profondeur de la structure hiérarchique.

$$sim\ (N1, N2) = -log\frac{dist(N1,N2)}{2\times D} \qquad (2.4)$$

où *D* est la profondeur maximum de la hiérarchie et *dist(N1,N2)* est le nombre minimum d'arcs reliant *N1* et *N2*.

- Howe [Howe, 2009] propose une mesure appelée *Rita*. Cette mesure tient compte de la distance minimale reliant deux nœuds *N1* et *N2* à leur parent commun le plus spécifique (*Cp*) et la distance entre *Cp* et la racine.

$$dist(N1, N2) = \frac{mindist\ ((Cp,N1),(Cp,N2))}{dist(Cp,Root)+mindist((Cp,N1),(Cp,N2))} \qquad (2.5)$$

2.2.2.2 Les approches basées sur le contenu informatif des nœuds

Ces approches incluent le contenu informationnel des nœuds dans les mesures de similarité des nœuds.

- Resnik [Resnik, 1995] propose une mesure qui détermine la similarité entre deux nœuds *N1* et *N2* basée sur le contenu informationnel (*IC*) de leur nœud commun le plus spécifique *N*. Cette mesure utilise un corpus et une ressource sémantique. Le contenu informationnel d'un nœud *N* exprime le degré d'importance de ce nœud dans le corpus. Il est mesuré par la fréquence d'apparition de *N* et par celle de ses nœuds descendants dans le corpus. Le contenu informationnel de *N* est calculé comme suit :

$$IC(N) = -\log(P(N)) \qquad (2.6)$$

Où *P(N)* est la probabilité de trouver *N* ou un de ses descendants dans le corpus. *P(N)* est calculée par l'équation (2.7).

$$P(N) = \frac{fr(N)}{frtot(N)} = \frac{\sum_{Ci\in Des(N)} count(Ci)}{frtot\ (N)} \qquad (2.7)$$

où :

Des(N) est l'ensemble des concepts *Ci* subsumés par le nœud *N*,
count(Ci) est le nombre d'occurrences d'un concept *Ci* dans le corpus,
frtot(N) est le nombre total d'occurrences de *N* dans le corpus.

La similarité sémantique entre deux nœuds *N1* et *N2* est donnée par le contenu informationnel de leur nœud commun le plus spécifique *N* selon l'équation (2.8). Ce contenu informationnel traduit l'information partagée par *N1* et *N2*.

$$Sim(N1, N2) = IC(N) \qquad (2.8)$$

La similarité de deux nœuds ne dépend que de leur nœud commun. Ce qui signifie que la similarité de n'importe quels nœuds (*N1,N2*) ayant le même nœuds commun ont la même similarité, ce qui dans la réalité n'est pas toujours vraie **[Blanchard et al., 2008]**.

- Lin [Lin, 1998a] s'appuie sur la mesure de Resnik **[Resnik, 1995]** et propose une mesure qui évalue la similarité de deux nœuds *N1* et *N2*. Cette mesure tient compte à la fois de l'information partagée par *N1* et *N2* et de l'information qui les différencient selon l'équation (2.9).

$$Sim(N1, N2) = \frac{2 * log(P(N))}{log(P(N1)) + log(P(N2))} \qquad (2.9)$$

2.3 Similarité des textes

Le calcul de la similarité de textes a pour objectif d'identifier des documents ayant un contenu similaire ou différent. Les représentations les plus utilisées sont des représentations vectorielles ou sous forme de graphe.

Dans notre travail, nous considérons un corpus de résumés des articles scientifiques relatifs à des domaines de recherche différents. Notre objectif est d'évaluer une similarité entre documents et de rechercher des documents présentant une forte similarité pouvant indiquer un risque de plagiat. Notre travail est lié à trois domaines : recherche d'informations, classification et détection de plagiat. Nous présentons dans ce qui suit des approches relatives à ces domaines.

2.3.1 Similarité basée sur le contenu des documents

2.3.1.1 Recherche d'information

La recherche d'information a pour objectif d'acquérir, d'organiser, de stocker, de rechercher et de restituer des informations répondant à un besoin utilisateur. Ces différentes tâches sont réalisées par des systèmes de recherche d'information (SRI) qui reçoivent en entrées des requêtes exprimées le plus souvent par des mots clés. Un système de recherche d'information met en œuvre un processus permettant de retrouver les documents pertinents relativement au besoin d'un utilisateur.

Dans le modèle vectoriel proposé par Salton dans le système SMART **[Salton, 1971]**, un texte (document ou requête) est projeté dans un espace vectoriel où chaque dimension est représentée par un terme d'indexation. Chaque élément d'un vecteur consiste en

un poids associé à un terme d'indexation. Ce poids représente l'importance d'un terme dans un document et il est calculé sur la base de *tf-idf* **[Salton et al., 1983]** ou ses variantes. La similarité vectorielle est calculée par plusieurs métriques telles que la mesure du cosinus qui mesure le cosinus de l'angle formé par les vecteurs correspondant aux textes (document-requête). Deux textes sont similaires si leurs vecteurs sont proches dans l'espace vectoriel dans lequel ils sont représentés.

La RI classique ne considère les documents que par leur contenu textuel. L'évolution du type des documents vers une représentation structurée et plus précisément vers le format XML a soulevé de nouvelles problématiques. Dans les documents structurés, la structure logique d'un document est séparée de son contenu. Un document est ainsi caractérisé par un contenu informationnel (du texte) et des contraintes structurelles (des balises comme le titre, section, paragraphes) formant une hiérarchie d'éléments. De nouvelles approches **[Schlieder et al., 2002][Carmel et al., 2003][Sauvagnat, 2005]** visent à exploiter cette structure pour développer de nouveaux concepts pour l'indexation et l'interrogation du corpus XML. L'information structurelle des documents peut en effet servir à affiner le concept de granule documentaire. La réponse fournie à l'utilisateur ne se résume plus à un document entier mais à des parties de document apportant une information pertinente à un besoin utilisateur.

2.3.1.2 Classification des documents

La classification automatique de textes permet de regrouper, dans une même classe, des documents traitant des thèmes similaires. La classification permet d'organiser de grandes quantités d'information et de faciliter leur recherche.

Les approches traitant de la classification supervisée affectent des documents à des classes prédéfinies **[Joachims, 1997][Joachims, 1998][Yang et al., 1999][Soucy et al., 2001] [Jaillet et al., 2006]** alors que les approches de classification non supervisée définissent les classes, appelées clusters, de façon automatique [**Hotho et al., 2002**][**Tar et al., 2011**]. Dans la classification supervisée, les classifieurs utilisent deux collections de documents : une collection contenant les documents d'apprentissage permettant de déterminer les caractéristiques de chaque catégorie et une collection contenant les nouveaux documents à classifier automatiquement. La classification d'un nouveau document dépend des caractéristiques retenues pour chaque catégorie. Nous nous intéressons plus particulièrement dans ce qui suit aux approches traitant de la classification supervisée des documents, objet d'une partie de nos travaux.

La classification automatique de textes a pour objectif d'attribuer une ou plusieurs étiquettes (classes) à un document en fonction de son contenu textuel. Ces classes correspondent aux thèmes abordés dans le document et elles sont déterminées par une fonction *f* définie comme suit :

$$f : (d, c) \rightarrow \{\text{vrai, faux}\} \ \forall \ (\text{d, c}) \in \text{D} \times \text{C}$$

Etant donnés un document *d* appartenant à l'ensemble des documents ***D*** et ***c*** une catégorie parmi les *n* catégories de l'ensemble *C={c1,c2,.....,cn}*, La fonction *f* détermine l'appartenance ou non du document *d* à la catégorie *c*. Autrement dit, la fonction *f* estime une similarité entre un document et chacune des différentes catégories. Nous décrivons dans les sections suivantes les classifieurs les plus connus dans la littérature.

2.3.1.2.1 Rocchio

Certains classifieurs créent une classe "prototype" **[Joachims, 1997]** à partir de la collection d'apprentissage. Cette classe est représentée par le vecteur moyen de tous les vecteurs des documents de la collection, par conséquent, seules certaines caractéristiques sont retenues. Un nouveau document *d*, représenté par un vecteur est affecté à la classe dont le vecteur prototype est le plus proche de celui de *d*.

2.3.1.2.2 Support Vector Machine

Avec les Machines à Vecteur de Support (SVM) **[Joachims, 1998]**, les documents sont représentés dans un espace vectoriel par les termes d'indexation qui les composent. Par apprentissage, cette méthode permet de définir une surface de séparation appelée hyperplan entre les documents de deux classes, minimisant le risque d'erreur de catégorisation et maximisant la marge entre ces deux classes. La marge est la distance entre la frontière de séparation et les échantillons les plus proches. Ces derniers sont appelés vecteurs de supports. Une catégorie *c* est attribuée à un nouveau document *d* en fonction de la position de *d* par rapport à la surface de séparation.

2.3.1.2.3 Arbre de décision

Les classifieurs de type arbre de décision **[Quinlan, 1986][Lewis et al., 1994][Goller et al., 2000]** utilisent un arbre construit par apprentissage dont les nœuds et les feuilles représentent respectivement les termes des document d'apprentissage et les différentes classes. Un arbre de décision représente un ensemble de choix sous la forme d'un arbre. Les différentes décisions sont représentées par les feuilles de l'arbre.

A chaque nœud est associée une condition relative au terme qu'il représente. Pour affecter un document *d* à une classe, l'arbre est parcouru de la racine jusqu'aux feuilles en passant par les nœuds internes. Le test associé à un nœud détermine le nœud suivant à atteindre. En fin de parcours, la feuille atteinte détermine la classe à retenir pour le document *d*.

2.3.1.2.4 K plus proches voisins

La méthode basée sur les *k* plus proches voisins (KNN) **[Soucy et al., 2001][Yang et al., 1999]** suppose que si les représentations vectorielles de deux documents sont proches dans l'espace vectoriel, ils ont une forte chance d'appartenir à une même catégorie. Un nouveau document *d* est comparé aux textes appartenant au jeu d'apprentissage. Pour déterminer la catégorie à affecter au document *d*, la classe la plus affectée au *k* voisins les plus proches de *d* est retenue ou bien un poids est affecté aux différentes classes des *k* voisins les plus proches

en fonction du classement de ces derniers. Ainsi la classe ayant le plus grand poids sera retenue.

2.3.1.2.5 Classifieurs probabilistes

Les classifieurs probabilistes se basent sur le calcul de la probabilité qu'un document, représenté par un vecteur de termes pondérés, appartienne à une classe donnée *P(Ci/d)*, (*i=1,…n, n* étant le nombre classes considérées). *d* sera affecté à la classe qui obtient la probabilité maximale. La plupart des approches probabilistes définies dans le domaine de la classification sont des classifieurs naïfs bayésiens. Ces derniers sont basés sur le théorème de Bayes avec des hypothèses fortement indépendantes. Cela signifie que l'attribution d'une caractéristique à une classe ne dépend pas des autres caractéristiques **[Cheeseman et al, 1996][Schneider, 2005]**.

2.3.1.3 Détection de plagiat

Le plagiat consiste à copier un travail d'un auteur et le présenter comme étant le sien sans faire référence à la source. Les systèmes de détection de plagiat possèdent en général en entrée le document original et le document suspect et se focalisent sur les points suivants : copie exacte du texte (copier/coller), insertion ou suppression de mots, substitution de mots (utilisation de synonymes), reformulation et modification de la structure des phrases.

L'approche commune des méthodes statistiques est la construction du vecteur de document à partir de valeurs décrivant le document telles que la fréquence des termes. La comparaison du document source et du document suspect revient à calculer leur degré de similarité en se basant sur différentes mesures (BM25 **[Robetson et al., 1994]**, modèle de langue **[Zhai et al., 2001]** etc.). Certaines approches calculent le pourcentage de recouvrement des mots, des n-grammes ou des phrases en appliquant un alignement entre le document original et le document suspect.

Les méthodes statistiques **[Lukashenko et al., 2007]** ne nécessitent pas la compréhension du sens des documents et ne permettent pas de détecter les cas de plagiat où la synonymie est utilisée pour remplacer des mots dans la reformulation des phrases.

Les descripteurs des documents ne sont pas toujours des mots. Les approches n-grammes caractérisent le contenu textuel d'un document par des séquences de *n* caractères consécutifs **[Brin et al., 1995][Basile et al., 2008][Basile et al., 2009]**. Basés sur des mesures statistiques, chaque document peut être décrit par un ensemble d' "empreintes digitales", où des n-grammes sont extraits et sélectionnés comme étant des "empreintes digitales" **[Stein et al., 2005]**. Le recouvrement entre deux empreintes digitales appartenant au document source et au document suspect indique des passages pouvant constituer un risque de plagiat. Nous décrivons ci-après trois approches relatives à la détection de plagiat.

2.3.1.3.1 Approche de Lewis

Dans **[Lewis et al., 2006]**, les auteurs exploitent l'algorithme d'alignement de textes défini dans **[Yamamoto et al., 2003]** pour faire aligner un texte avec les différents documents d'un corpus. L'alignement présente l'avantage de préserver l'ordre des mots et l'ordre des phrases. Cet algorithme utilise une matrice où la suppression ou l'ajout d'un mot est représenté par -1, le décalage d'un mot par un 0 et si un mot apparait à la même place, il est représenté par un poids. Les auteurs utilisent un alignement de texte intégral où le score le plus élevé à partir de toute cellule de la matrice d'alignement représente le score de similarité des deux textes. Le processus proposé est implémenté dans le système eTBLAST, qui est considéré comme un moteur de recherche permettant d'identifier entre autres, les citations similaires dans *Medline*. Cette dernière est une base de données regroupant la littérature relative aux sciences biologiques et biomédicales.

Dans ses travaux, Errami **[Errami et al., 2009]** a exploité et paramétré *eTBLAST* afin d'identifier les citations qui ont une similarité inhabituellement élevée, qui sont ensuite enregistrées dans la base de données *Déjà vu* en attente d'une vérification manuelle.

2.3.1.3.2 Approche Vani

Vani **[Vani et al., 2015]** segmente le document source et le document suspect en phrases. Chaque phrase est ensuite représentée par un vecteur de termes qui la composent. Ces termes sont pondérés. Chaque phrase du document source est comparée à toutes les phrases du document suspect en calculant une similarité basée sur leur vecteur en utilisant séparément plusieurs métriques (cosinus, dice etc). Vani étudie l'importance de la combinaison de ces différentes métriques sur la détection du plagiat. Il explore également l'impact de l'utilisation du rôle syntaxique des mots dans les phrases sur le calcul de la similarité des phrases. Les phrases étiquetés par un analyseur syntaxique **[Toutanova et al., 2003]** sont ainsi comparées en mettant en correspondance les termes appartenant à la même classe (les noms avec les noms, les verbes avec les verbes, les adjectifs avec les adjectifs et les adverbes avec les adverbes). Attribuer aux mots leur fonction syntaxique rend la comparaison plus significative et élimine des détections de similarité erronées.

2.3.1.3.3 Approche de Basile

Basile **[Basile et al., 2009]** propose un algorithme en trois étapes pour la détection de plagiat. La première étape consiste à sélectionner, pour chaque document suspect, un sous ensemble de documents source à partir d'un corpus. Le contenu des documents est découpé en un ensemble de 8-grammes puis représenté sous forme vectorielle. Une distance est calculée entre chaque document source et chaque document suspect puis un classement par ordre décroissant de leur similarité est effectué. Pour chaque document suspect, les dix premiers documents source sont retenus. Une similarité entre deux documents x et y est calculée par l'équation (2.10).

$$Sim(x,y) = \frac{1}{|Dn(x)| + |Dn(y)|} \times \sum_{w \in Dn(x) \cup Dn(y)} \frac{(f_y(w) - f_x(w))^2}{(f_y(w) + f_x(w))^2} \qquad (2.10)$$

Où *w* dénote un n-gramme, *fx(w)* dénote la fréquence relative de *w* dans le texte x et *Dn(x)* représente l'ensemble des n-grammes définissant le dictionnaire de *x*.

Dans une deuxième étape, une analyse approfondie des documents source retenus pour chaque document suspect est réalisée dans le but de retrouver les passages plagiés. Il s'agit de retrouver des correspondances entre séquences dont la longueur dépasse un seuil fixé. Un codage de type T9 est adopté pour représenter le contenu textuel des documents. L'idée est de remplacer 3 ou 4 lettres différentes par un même caractère. Par exemple {a,b,c} sont remplacés par 2, {d,e,f} par 3, une nouvelle ligne et un espace sont représenté par 0 etc. le nouvel alphabet est constitué des symbols {0,1,…,9}. La "compression" T9, indique qu'une séquence T9 de 10 à 15 caractères correspond dans la plupart des cas à une phrase unique qui possède un sens dans le texte d'origine.

Les passages suspects sont ensuite recherchés. En commençant de n'importe quelle position du document suspect, des correspondances de séquences avec le document source, les plus longues possibles dépassant un seuil seront retenues. Cette étape donne une longue liste de correspondances pour chaque paire de documents (source- suspect). Le plagiat étant souvent masqué, les passages sélectionnés à la deuxième étape ne correspondent pas toujours à des passages consécutifs dans le document source. La dernière étape consiste donc à retrouver exactement l'emplacement des passages plagiés au niveau des documents source. La paire de documents est représenté sur un plan à deux dimensions où le document source est sur l'axe des (y) et le document suspect est sur l'axe des (x). Toute correspondance de taille *l* commençant à partir de la position x dans le document suspect et à la position y du document source, dessine une ligne de *(x,y)* jusqu'à *(x+l, y+l)*. Sur le plan, des lignes ou des formes ressemblant à des carrés sont formés. Le plagiat "non caché" est représenté par des lignes. Ce sont des correspondances de passages successives à la fois dans les documents source et suspect, alors que le plagiat "caché" correspond aux carrés. Ils représentent des correspondances retrouvées dans un ordre différent dans les documents sources et suspect.

2.3.2 Similarité sémantique des documents

Les approches conventionnelles, comme celles présentées dans les sections précédentes, représentent un document par un ensemble de descripteurs (mots, n-grammes) sans mettre en avant le sens véhiculés par ces derniers. Par conséquent la compréhension du sens des documents est ignorée engendrant des erreurs de rapprochement. Les nouvelles approches visent à extraire et à représenter la sémantique décrite par le contenu des documents en exploitant des ressources sémantiques telles que des thésaurus et ontologies de domaine. Nous présentons dans les sections suivantes un certains nombre de ces approches.

2.3.2.1 Similarité vectorielle

Plusieurs approches dans la littérature exploitent des ressources sémantiques pour représenter les documents par des concepts, extraits de ces ressources, correspondant au contenu textuel des documents. L'objectif de ces approches est de donner un sens aux mots et de réduire la taille des vecteurs. Un document est représenté par un vecteur dont la dimension est égale au nombre de caractéristiques retenues pour représenter les différentes catégories dans le cas de la classification des documents. Dans le cas d'une indexation basée sur un corpus, comme c'est le cas pour la recherche d'information, le nombre de dimension est égal au nombre de termes représentant la collection. Les grandes dimensions des vecteurs réduit les performances des différentes applications. Dans **[Beyer et al., 1999]**, les auteurs ont étudié l'impact du nombre de dimensions sur le problème du "plus proche voisin". Leur analyse a montré que quand ce nombre augmente, la distance du *point donnée* (data point) le plus proche se rapproche de la distance *du point donnée* le plus éloigné.

Pour les approches adoptant une représentation vectorielle des documents, utiliser des ressources sémantiques pour représenter un document par un vecteur de concepts permet de réduire la taille des vecteurs. Etant donné que les termes synonymes correspondent à un même concept, le nombre de descripteurs représentant un document est plus petit que le nombre de termes le composant. Pour réduire la taille des vecteurs, une sélection des termes les plus importants en fonction de leur poids est également effectuée. Nous détaillons dans ce qui suit quelques approches du domaine de la classification exploitant des ressources sémantiques.

2.3.2.1.1 Approche de Hotho

Dans le domaine de la classification non supervisée Hotho, **[Hotho et al., 2002]** propose une approche appelée *Concept Selection and Agregation* (*COSA*) pour la sélection des caractéristiques représentant le contenu des documents à partir d'une ontologie. L'objectif de cette approche est de réduire les dimensions des vecteurs des documents et de donner une explication évidente sur la façon dont les clusters sont construits. L'utilisation d'une ontologie permet de générer plusieurs représentations de l'ensemble de documents sur la base de l'algorithme standard *K-Means*. L'utilisateur peut alors décider de préférer l'une des représentations en fonction des concepts utilisés pour la classification.

Une ontologie de domaine est utilisée pour le prétraitement et la sélection des vues pertinentes (i.e. agrégations) sur l'ensemble des documents. Un prétraitement est effectué pour réduire la taille des vecteurs et ne sélectionner que les termes importants en calculant leur poids basé sur *tf-idf*. L'ontologie est une hétérarchie constituée d'un ensemble de concepts C^* reliés entre eux par différentes relations (directe, acycliques, transitives et réflexives).

COSA associe les termes retenus aux concepts de l'ontologie et exploite l'hétérarchie de concepts pour former les clusters. Une heuristique est recherchée pour générer de bonnes agrégations et pour réduire davantage la taille des vecteurs. L'algorithme *Generate Concept Views* est alors proposé. Il consiste à parcourir l'hétérarchie de haut en bas en décomposant les concepts ayant un *support* élevé (équation 2.12) en leurs sous-concepts et en abandonnant les concepts ayant des *supports* faibles. Ainsi, l'algorithme génère des listes de concepts qui

n'apparaissent ni trop souvent ni trop rarement car ils ne sont pas appropriés pour le clustering. Les valeurs des *supports* sont calculées par les équations (2.11) et (2.12).

$$\text{Support}(i,C) = \sum_{\substack{B \in C^* \\ H(B,C)}}' cf(i,B) \qquad (2.11)$$

$$\text{Support}(C) = \sum_{i=1}^{N} \text{Support}(i,C) \qquad (2.12)$$

Le support direct d'un concept *c* dans un document *di* est défini par la fréquence de *c*, *cf (i, c),* basée sur l'apparition de l'un des termes *ti* de *c* dans *di.* Un support complet d'un concept *c* prend en compte tous les sous-concepts de *c* selon l'équation (2.11).

Nous donnons ci-après un exemple de construction d'une agrégation. La variable *Agenda* est définie pour décrire la liste courante de concepts utilisés pour générer une représentation particulière à partir d'un ensemble de documents.

Agenda au départ est constitué des concepts [*Accomodation*, *Vacation*, *Sight-seeing*]. Cette agrégation est modifiée en remplaçant les concepts de *Agenda* qui ont une valeur de *support* la plus élevée par leurs sous-concepts. Le concept *Accomodation* possède les sous-concepts [*Hotel, Guesthouse*, *Youth-hostel*]. Le sous concept *Hotel* ayant le *support* le plus élevé est retenu et les autres sous concepts sont agrégés en un seul concept [*Guesthouse*, *Youth-hostel*]. *Agenda* contiendra alors les concepts suivants : [*Vacation*, [*Guest-house*, *Youth-hostel]*, *Hotel*, *Sight-seeing*].

L'utilisateur peut s'appuyer sur l'hétérarchie proposée par l'approche *COSA* pour contrôler et éventuellement interpréter les résultats de regroupement en clusters.

2.3.2.1.2 Approche de Gabrilovich

Dans l'approche proposée dans **[Gabrilovich et al., 2005],** la collection d'apprentissage est remplacée par des "connaissances du monde" extraites à partir d'ontologies publiquement disponibles contenant des centaines de milliers de concepts telles que *Open Directory Project* (ODP). Le contenu des documents est associé aux concepts appropriés de l'ontologie qui vont constituer des caractéristiques qui vont enrichir le vecteur document initial, souvent comparé à un "*sac de mots*". La classification des documents à l'aide de caractéristiques basées sur ces connaissances exploite des informations qui ne peuvent pas être déduites directement des documents. Les URLs répertoriées dans ODP sont explorées pour rassembler une grande quantité d'informations. Le processus génère de nouvelles caractéristiques et les rajoute au *sac de mot*.

ODP contient un ensemble de concepts organisés dans une hiérarchie (une hiérarchie d'environ 600 000 catégories avec plus de 4 000 000 sites Web, chacun représenté par une URL, un titre, et un bref résumé de son contenu). Chaque concept contient un ensemble de textes qui sont exploités par le processus pour apprendre les définitions des concepts afin de les assigner aux documents appropriés.

Un classifieur hiérarchique appelé "*générateur de caractéristiques*" est construit et le texte est découpé en plusieurs segments appelés contexte. Le générateur de caractéristiques permet d'associer chaque contexte à des concepts ODP pertinents. Ces concepts représentent les sous-thèmes abordés dans le document. A partir des textes ODP, le processus extrait les mots les plus importants pour représenter le contenu de chaque concept. Ces mots, appelés attributs, vont constituer les vecteurs concepts. Les caractéristiques correspondent aux concepts ainsi que leurs ancêtres dans la hiérarchie. Les descriptions textuelles des nœuds ODP et de leurs URLs sont utilisées comme exemples d'apprentissage pour entrainer le *générateur de caractéristiques*.

Pour illustrer l'utilité du *générateur de caractéristiques*, Gabrilovich cite en exemple le document # 15264 de Reuters-21578. Ce document appartient à la catégorie "*cuivre*" et traite d'une entreprise minière rassemblant plusieurs sociétés. Le document ne mentionne que brièvement les participations des sociétés impliquées (*Teck Corporation, Cominco et Lornex Mining*) et leur activités minières. La catégorie "*cuivre" est une catégorie relativement petite, et aucune de ces compagnies n'est mentionnée dan*s l'ensemble d'apprentissage de cette catégorie. Par conséquent, les trois classifieurs de texte (SVM, KNN et C4.5) n'ont pas réussi à classer le document correctement.

La Figure 2.1 illustre le processus de génération de caractéristiques pour le document # 15264.

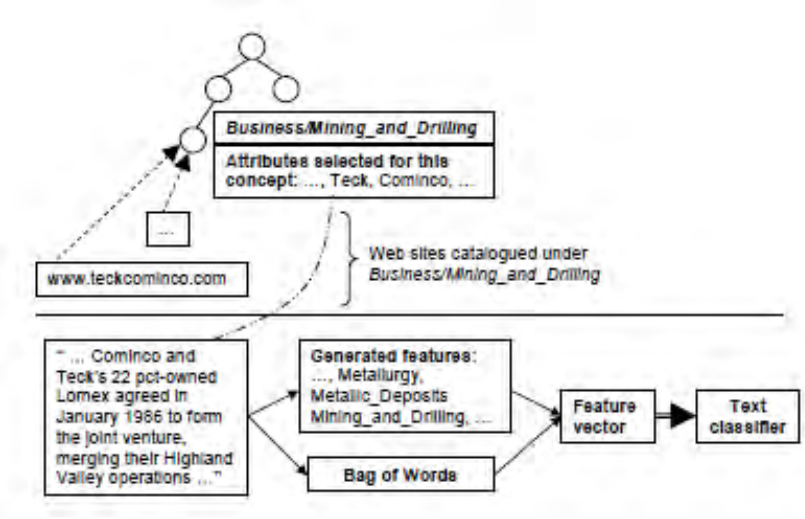

Figure 2.1 Exemple de génération de caractéristiques [Gabrilovich et al., 2005].

Le *générateur de caractéristiques* explore les sites Web catalogués sous des concepts ODP liés à l'exploitation minière tels que *Business/Mining_and_Drilling*, *Science/Technology/Mining* et *Business/Industrial_Goods_and_Services/Materials/Metals*. Ces concepts incluent *www.teckcominco.com* et *www.miningsurplus.com* qui appartiennent à la société fusionnée *Teck Cominco*. En raison de l'importance de l'entreprise, elle est fréquemment mentionnée dans les sites Web explorés. Par conséquent, les mots *Teck* et *Cominco* sont inclus dans l'ensemble des attributs sélectionnés pour représenter les concepts ci dessus.

Le processus effectue implicitement et dans une certaine mesure une désambiguïsation des mots polysémiques puisqu'un contexte contenant des mots ambigus est associé aux concepts qui correspondent au sens partagé par tous les mots du contexte. Le sens approprié des mots ambigus est déterminé par celui des mots qui co-occurrent avec lui dans le même contexte. De plus, l'enrichissement des représentations des documents par des concepts de plus haut niveau permet des rapprochements entre documents abordant des sujets similaires même s'ils utilisent des vocabulaires différents. Cette approche, cependant, présente l'inconvénient de demander un grand effort d'ingénierie pour déterminer les nouvelles caractéristiques (environ 425 Go de fichiers html à explorer).

2.3.2.1.3 Approche de Tar

L'approche proposée dans **[Tar et al., 2011]** s'appuie sur une ontologie de domaine pour extraire et pondérer des concepts de l'ontologie correspondants aux termes décrivant le contenu des documents. L'objectif étant de réduire les dimensions des vecteurs, le système développé pour supporter cette approche propose trois modules majeurs : un module pour les prétraitements des documents permettant d'extraire les mots-clés de leur contenu textuel, un module pour calculer les poids des concepts et un module pour la classification des documents représentés par des vecteurs de concepts pondérés.

Le système extrait des documents les mots-clés en supprimant les mots vides. Le poids habituellement calculé par la formule *tf-idf* n'est pas adopté. Pour calculer le poids *W* d'un mot clé *m*, le processus propose l'équation (2.13).

$$W = \lg \times Freq \times Coefficient\ de\ corréaltion + P(C) \quad (2.13)$$

Où *W* est le poids d'un mots-clé *m*, *lg* représente la longueur de *m*, *Freq* désigne la fréquence d'apparition de *m*. Le *coefficient de corrélation* est égal à 1 si le concept existe dans l'ontologie, 0 dans le cas contraire. *P(C)* est basée sur la probabilité de *C* dans le document. Elle est estimée par l'équation (2.14).

$$P(C) = \frac{NbOc\ (C)}{TotOc} \quad (2.14)$$

où *NbOc* et *TotOc* représentent respectivement le nombre d'occurrences de *C* et le nombre d'occurrences de tous les concepts dans le document.

Le système classe ensuite les poids et sélectionne les mots-clés qui ont les plus grands poids pour le processus de classification.

2.3.2.1.4 Approche de Qazi

Dans le but de réduire les dimensions des vecteurs représentant les documents, Qazi **[Qazi et al., 2018]** propose également d'utiliser une ontologie de domaine pour extraire les concepts correspondants aux termes des documents. Son processus est appliqué à la classification des documents. Il définit des catégories au sein de l'ontologie (Cricket, Football, Hockey et Basketball) comme étant des classes. Pour chaque document, il ne retient que les termes ayant une correspondance dans les catégories retenues. Ces termes pondérés formeront les représentations vectorielles des documents.

2.3.2.2 Similarité de graphes

Les approches utilisant la représentation vectorielle des documents possèdent plusieurs limites telles que rapportées dans **[Pincemin, 2000]** : leurs performances diminuent dès qu'elles s'appliquent à des textes relativement longs. Avec les formules de pondération utilisées, les mots qui n'apparaissent qu'une seule fois dans le document ou au contraire qui sont très répétés sont ignorés bien qu'ils aient un sens pour le contenu du document. La représentation vectorielle telle qu'elle est définie ne met pas en avant les relations entre les mots d'un document engendrant ainsi des rapprochements erronés.

Pour surmonter ces problèmes, une représentation sémantique des documents sous forme de graphe est proposée dans plusieurs travaux. Les documents sont souvent représentés non pas par les mots composant leur contenu mais par des concepts correspondants à ces mots, extraits à partir de ressources sémantiques. La similarité entre deux documents est calculée par la similarité des graphes associés aux documents.

2.3.2.2.1 Approche de Baziz

Baziz propose deux approches pour représenter les documents. Une première représentation est donnée sous forme de réseaux sémantique et la seconde sous forme d'arbre.

● Baziz **[Baziz, 2005a]** propose un premier modèle de représentation des documents *DocCore* inspiré des réseaux sémantiques **[Quillian, 1968]**. Pour chaque document de la collection, un noyau sémantique de document *DocCore* est construit en utilisant WordNet. Les nœuds et les arcs du réseau sémantique d'un document *d* sont extraits de WordNet. Les nœuds correspondent aux termes composant *d* et les arcs reliant ces nœuds sont pondérés. Ces poids calculés par des mesures de similarité sémantique indiquent la proximité sémantique de deux nœuds mais également mesurent l'importance des termes dans le document. L'approche est composée des étapes suivantes :

- Extraction des termes les plus fréquents qui possèdent une entrée dans WordNet.

- Listage des différents sens correspondants aux concepts retenus. Plusieurs réseaux sémantiques sont alors construits. Pour chaque terme de *d*, un score est calculé pour chacun de ses concepts candidats. Chaque concept candidat possède un score égal à la somme des valeurs de similarité qu'il a obtenu avec les autres concepts candidats correspondant à tous les termes représentant le document. Les concepts candidats ayant les plus grands scores sont alors sélectionnés pour représenter les nœuds du meilleur noyau sémantique de *d*. Des détails sur le calcul les différents scores sont donnés dans le chapitre précédent. (cf. chapitre 1, section 1.6.3).

● Baziz **[Baziz, 2005b]**, dans son deuxième modèle *Doctree*, propose de construire un graphe, pour chaque document et pour chaque requête, à partir de concepts extraits de WordNet. Une mise en correspondance du graphe d'un document et celui de la requête amène l'auteur à représenter les deux graphes par rapport à un même référentiel constitué des nœuds appartenant au document et à la requête. Chaque graphe est ensuite élargi en rajoutant des nœuds du référentiel.

Les poids des nœuds rajoutés au niveau de la requête valent zéro alors que dans le sous arbre du document où un nœud est rajouté, le poids d'un nœud de niveau *s* est mis à jour récursivement en multipliant le poids du nœud de niveau *s+1* (le nœud de niveau *s* subsume le nœud de niveau *s+1*) par un facteur qui dépend du niveau de la hiérarchie. Le poids des nœuds est mis à jour selon l'équation (2.15).

$$w^{s}_{i,new=max}\left(w^{s}_{i}, (max_{i} w^{s+1}_{i,new}) \times fact(s)\right) \qquad (2.15)$$

Baziz applique ensuite une sélection sur les nœuds pour ne retenir que les nœuds les plus spécifiques de la hiérarchie. La Figure 2.2 donne un exemple de construction des graphes relatifs à un document et une requête.

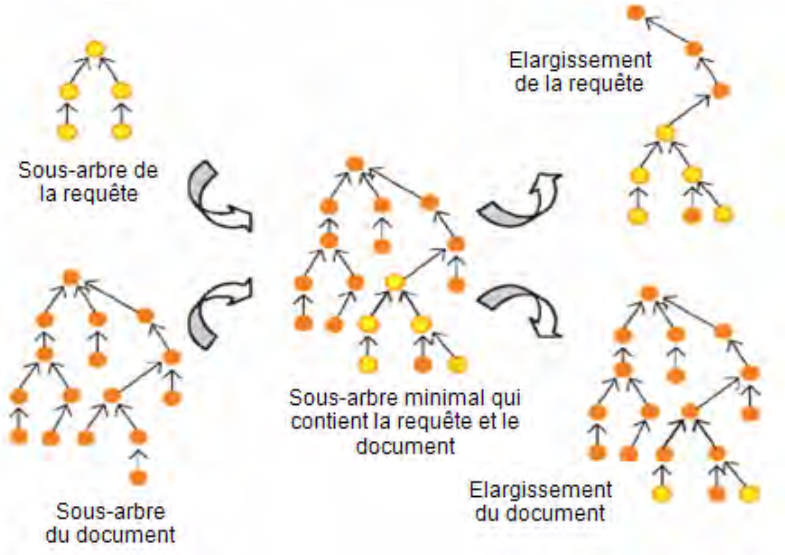

Figure 2.2 Construction des graphes correspondant au document et à la requête [Baziz et al., 2005b]

Les représentations du document et de la requête sont ensuite comparées en utilisant des opérateurs flous et une valeur de pertinence est calculée. Cette valeur exprime jusqu'à quel point le document couvre le sujet exprimé dans la requête.

2.3.2.2.2 Approche de Dudognon

Dans le domaine de la recherche d'information sémantique, Dudognon **[Dudognon et al., 2010]** représente les documents par des annotations où chaque annotation est constituée d'un ou plusieurs graphes de concepts extraits de WordNet. Le calcul de similarité entre annotations se fait sur la base de trois similarités.

- ***Similarité des concepts***. Dudognon fait une analogie entre les relations reliant les termes d'un vocabulaire dans une structure (ressource) sémantique avec les relations généalogiques entre membres d'une famille. Il propose de calculer la similarité entre deux concepts *C1* et *C2* par une mesure qui repose sur leur nombre d'ancêtre communs et sur la distance séparant ces concepts de leur ancêtre. Cette similarité est mesurée par l'équation (2.16).

$$Sim(C1,C2) = \frac{\left|ancêtres(C1,C2)\right|^2}{\left|Gen(C1)\right| \times \left|Gen(C2)\right|} \qquad (2.16)$$

où *Gen(Ci)* représente l'ensemble des concepts qui entrent dans la généalogie du concept *Ci*, depuis la racine jusqu'à *Ci* et *Ancêtres (C1, C2)* représente l'ensemble des ancêtres communs des concepts *C1* et *C2*.

- ***Similarité des graphes de concepts***. Les concepts possèdent des importances différentes en fonction des applications. Le degré d'importance des Top-concepts, représentant les concepts le plus génériques de l'ontologie, est calculé arbitrairement ou par apprentissage. Le degré d'importance d'un concept correspond à celui du *Top-concept* dont il est le descendant. La similarité entre deux graphes est définie comme la moyenne pondérée des similarités entre les concepts qui les composent selon l'équation (2.17).

$$Sim\,(G1,G2) = \frac{\sum_{i=1}^{\left|noeuds(G1)\right|} Coef\,(G1_i) \times \underset{j=1}{\overset{\left|noeuds(G2)\right|}{Max}} (Simconcept\,s\,(G1_i, G2_j))}{\sum_{i=1}^{\left|noeuds(G1)\right|} Coef\,(G1_i)} \qquad (2.17)$$

- ***Similarité des annotations***. La similarité entre deux annotations est déterminée par la moyenne des similarités des graphes de concepts qui les constituent.

2.3.2.2.3 Approche de Zhang

Dans **[Zhang et al., 2011]**, les auteurs proposent de calculer la similarité entre deux textes basée sur une représentation des documents sous formes de graphes construits en utilisant Wikipédia comme base de connaissances. Cette approche repose sur l'hypothèse selon laquelle deux objets d'un même type sont similaires s'ils sont liés à des objets similaires d'un autre type. Chaque article de Wikipédia est appelé concept et pour chaque document, les termes les plus représentatifs sont extraits et mappés (mis en relation) avec les concepts de Wikipédia. Les concepts forment les nœuds se trouvant en haut de la hiérarchie d'un graphe bipartite. Un arc relie chaque document *d* avec les concepts *c* qui correspondent à son contenu. Le poids d'un arc *(d,c)* est déterminé par le nombre d'occurrences du concept *c* dans le document *d.* Un exemple de graph est donné par la Figure 2.3.

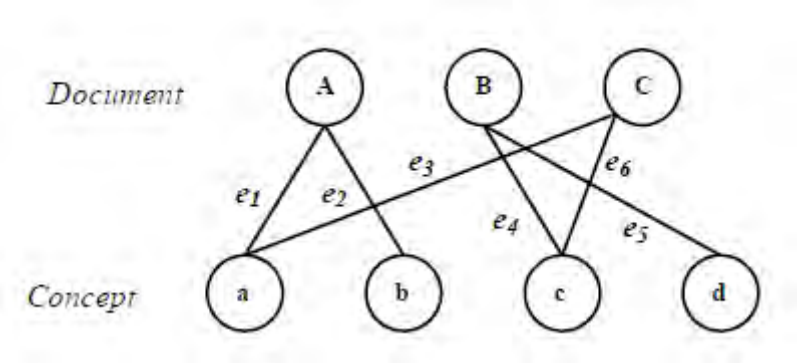

Figure 2.3 Graphe bipartite documents-concepts [Zhang et al., 2011]

La similarité de deux documents est déterminée par la similarité des concepts qu'ils contiennent. Deux documents n'ont pas besoin de partager des concepts communs pour être similaires tant que leurs concepts correspondants sont corrélés.

Par exemple, comme illustré par la Figure 2.3, les documents *A* et *B* n'ont aucun concept en commun. Une représentation vectorielle de ces documents et un calcul de leur similarité par la mesure du cosinus donnera une valeur égale à zéro.

Dans l'approche de Zhang, la relation sémantique entre les concepts peut être déduite en calculant par itération leur similarité à travers le graphe, ce qui permet de retrouver une similarité entre les documents. L'apparition conjointe des concepts *a* et *c* dans le document *C* indique une sorte de corrélation entre eux. Cette relation sémantique permet de déterminer un score de similarité entre *A* et *B*. Cette approche a été évaluée dans un processus de classification supervisée et non supervisée et a montré de meilleurs résultats que les approches basées sur une représentation vectorielle des documents.

2.3.2.2.4 Approche de Osman

Osman **[Osman et al., 2011]** décrit une approche permettant de détecter le plagiat en représentant les documents (original et suspect) par un graphe déduit de WordNet. Cette approche permet de détecter les formes de plagiat où la synonymie est exploitée pour reformuler les phrases. Le document est divisé en phrases. Chaque document est représenté sous forme de graphe où chaque nœud représente les termes d'une phrase. Ces derniers sont projetés sur WordNet pour extraire les concepts leur correspondant. La valeur de chaque arc reliant deux nœuds est donnée par l'imbrication des concepts appartenant aux deux nœuds. Ces concepts permettent de détecter les parties suspectes d'un document. La similarité entre deux phrases *S1* et *S2* est calculée sur la base de l'imbrication de leurs nœuds selon l'équation (2.18).

$$Sim(S1,S2)=$$

$$Imbrication\ (Sl,S2)=\frac{\left|CSl_i\right|\cap\left|CS2_j\right|}{\left|CSl_i\right|\cup\left|CS2_j\right|} \qquad (2.18)$$

Où $CS1_i$ et $CS2_j$ représentent respectivement les concepts de *S1* et les concepts de *S2*.

Tous les nœuds des différentes phrases sont connectés à un nœud unique appelé "*Topic Signature*". Ce nœud contient les concepts correspondant aux termes des différentes phrases. En utilisant WordNet, les hyperonymes et les synonymes des termes sont extraits. Le regroupement des concepts d'une phrase dans un nœud détermine le contenu de ces nœuds. Le nœud "*Topic Signature*" gère un index pour chaque nœud afin de déterminer pour chaque concept à quelle phrase il appartient. Le nœud "*Topic Signature*" permet ainsi de capturer rapidement les parties suspectes des documents lors de la comparaison du document source *D1* et le document suspect *D2.* La similarité entre ces deux documents est calculée sur la base de la similarité entre le nœud "Topic Signature" de *D1* et le nœud "*Topic Signature*" de *D2*. Le processus détermine ensuite la quantité de phrases plagiées.

2.3.2.2.5 Approche de Shenoy

Shenoy **[Shenoy et al., 2012**] propose une représentation d'un document par un graphe et l'exploite dans le domaine de la détection du plagiat. La Figure 2.4 donne l'architecture du processus.

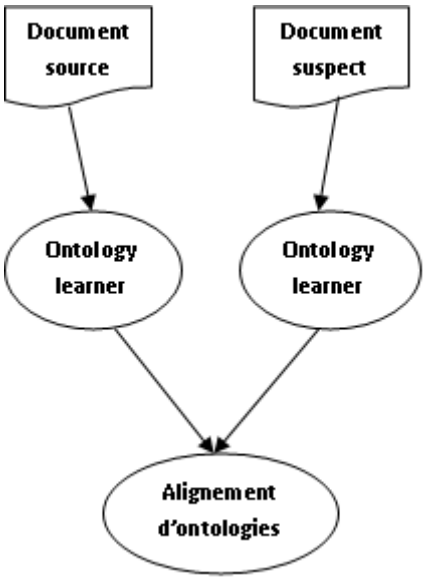

Figure 2.4 Architecture du système de détection [Shenoy et al., 2012]

Shenoy représente le contenu d'un document par une "sous ontologie" construite en utilisant la version démo de OntoGen[1] Ontology Learner **[Fortuna et al., 2007]**. La comparaison du document source avec le document suspect revient à aligner leurs "sous ontologies" et à calculer le nombre de concepts, propriétés et relations qu'ils partagent. L'alignement est exprimé comme une fraction de l'ensemble. Si cette fraction est au dessus d'un seuil donné, le système conclut que les deux documents sont similaires dans le sens.

2.3.2.2.6 Approche de Galdos

Dans l'approche définie dans **[Galdos et al., 2017]**, le calcul du poids des termes est similaire au calcul que nous avons proposé dans **[Iltache et al., 2016]**. Ce poids est déterminé en fonction des termes qui apparaissent conjointement avec lui dans le texte.

Les auteurs représentent les termes et les relations entre ces termes par un graphe. Les termes *ti* correspondent aux nœuds du graphe. Le poids d'un arc du graphe reliant deux termes représente le nombre de fois où ces deux termes apparaissent ensemble dans le document et le poids d'un nœud est donné par la somme des poids des arcs auquel il est relié. Les nœuds communs à deux graphes, ayant les plus grands poids sont retenus pour former une liste de mot clé, L_{Kw}. Pour la comparaison de deux graphes, l'algorithme Dijkstra est appliqué afin de définir un voisinage d'un nœud v appartenant à L_{kw} sur un rayon ρ. Plusieurs rayons peuvent être définis come le montre la Figure 2.5.

[1] OntoGen est un éditeur d'ontologie semi-automatique. Il permet l'édition d'ontologies thématiques qui représentent un ensemble de sujets liés par différents types de relations.

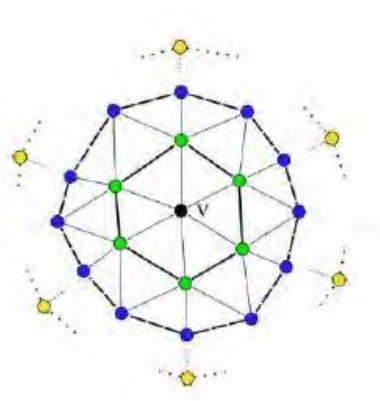

Figure 2.5 Voisinages de v

Chaque couleur représente un anneau constituant un voisinage pour *v*. Le voisinage de chaque nœud *v* appartenant à L_{kw} est déterminé en calculant la distance minimale à partir de *v* et le reste des nœuds appartenant aux deux graphes. Les nœuds communs aux deux graphes appartenant au voisinage de *v* sont retenus pour constituer une liste *R*. Sur la base de la liste *R* et la liste $L_{kw,}$, une similarité entre les graphes est finalement calculée pour regrouper des documents similaires.

2.4 Conclusion

Dans ce chapitre, nous avons commencé par présenter des travaux traitant de la similarité des mots sur laquelle se basent plusieurs applications ayant recours au traitement du langage naturel. Nous avons ensuite présenté plusieurs travaux traitant de la similarité des textes. Nous avons commencé par présenter dans une première partie, des approches relatives au domaine de la classification, de la détection de plagiat et la recherche d'informations. Dans ces domaines, une similarité entre documents est évaluée. Dans le cas de la classification et de la recherche d'informations, une similarité entre les documents entiers est calculée alors que pour le domaine de plagiat, ce sont des portions de textes (phrases) plagiées qui sont recherchées. La similarité est alors évaluée en fonction du nombre de mots ou de n-grammes communs aux deux documents. Ces différentes approches ne considèrent les documents que par les mots (ou n-grammes) qui les composent et ne mettent pas l'accent sur le sens que ces derniers peuvent véhiculer. Par conséquent, aucun rapprochement ne peut se faire si les documents utilisent des mots différents pour exprimer les mêmes notions ou des notions similaires. De plus, la présence de la polysémie inhérente au langage naturel engendre des rapprochements erronés.

La détection de plagiat, telle qu'elle est réalisée par différentes approches, vise à retrouver les phrases plagiées, c'est-à-dire des phrases extraites à partir d'un document source et insérées dans un autre document en faisant une opération de type copier/coller. Ces approches ne détectent pas les cas de plagiat lorsque la synonymie est exploitée pour remplacer des mots des phrases sources.

Pour pallier les limites des approches conventionnelles, de nouvelles approches se penchent sur la prise en compte de la sémantique dans le calcul de la similarité des textes. La

deuxième partie de la section 2.3 a permis de présenter différentes approches permettant de calculer la similarité sémantique entre documents. Nous les avons classées en deux catégories. Une catégorie donne une représentation vectorielle du contenu textuel et une autre catégorie le représente sous forme de graphe.

Les classifieurs sémantiques adoptent pour la plupart une représentation vectorielle où les dimensions sont représentées soit par les concepts, extraient d'une ressource sémantique, correspondant aux mots du document, soit par un vecteur de mots augmenté de ces concepts. Cette représentation n'exploite pas les liens reliant les mots dans le document. Des réflexions sont toujours nécessaires pour choisir les caractéristiques à retenir afin de réduire la taille des vecteurs. La similarité entre documents est basée sur le nombre de concepts et/ou de mots partagés par leurs vecteurs.

Pour les approches relatives à la détection de plagiat, les points abordés sont la copie/coller du contenu, la modification de la structure des phrases ou le remplacement des mots par leurs synonymes. La similarité entre deux phrases est souvent calculée par le nombre de mots communs rapporté au nombre de mots différents puis un pourcentage de phrases plagiées est calculée pour le document entier.

Pour notre corpus de résumés d'articles scientifiques, nous abordons le plagiat d'un autre point de vue. Le plagiat n'est pas déterminé au niveau phrase. Il ne s'agit pas de vérifier si les mots des phrases sont identiques ou modifiés. Nous proposons d'extraire des notions du contenu d'un résumé en leur attribuant une fonction relativement à son contexte globale. C'est sur la base de ces notions que la similarité entre deux résumés est évaluée.

La représentation des documents sous forme de graphe permet de s'affranchir des limites induites par la taille des vecteurs. Combinée à une ressource sémantique, une représentation sémantique des documents peut être mise en avant. C'est cette représentation que nous adoptons dans nos travaux.

Baziz [Baziz, 2005b] présente une approche pour l'enrichissement des graphes pour faire ressortir des informations non explicitement citées dans le document. Cependant, Baziz considère un document comme un tout. L'extraction des concepts pour représenter les mots explicitement cités dans le document ne dépendent ni du contexte d'apparition des mots, ni de la thématique abordée dans son contenu. Tous les termes du document qui possèdent une entrée dans WordNet sont retenus. La comparaison entre une requête et un document vise à calculer non pas une similarité entre leur représentation mais plutôt le degré d'inclusion d'une requête dans un document.

Pour notre cas, nous ne considérons par un document comme un tout. Nous montrons que dans les résumés d'articles scientifiques, il existe une structure implicite. Nous exploitons cette structure pour caractériser leur contenu textuel des résumés et pour évaluer leur similarité. Notre approche est décrite dans les chapitres suivants.

Partie 2

Contribution à l'utilisation des ontologies pour l'évaluation de la similarité des textes

Chapitre 3

Classification sémantique basée sur des ontologies

3.1 Introduction

La majorité des classifieurs existant dans la littérature représentent un document par un ensemble de mots composant le contenu textuel. Lorsque des ressources externes sont utilisées, les dimensions de vecteurs correspondent soit à des concepts soit à un ensemble de mots et de concepts. Ces approches considèrent les mots du document indépendants les uns des autres. De plus elles sont confrontées à un dilemme relatif aux choix des caractéristiques à retenir pour le document : représenter les documents par toutes les caractéristiques extraites de leur contenu induit des dimensions très grandes qui diminuent l'efficacité des classifieurs, et réduire le nombre de caractéristiques conduit à une perte d'information.

Nous proposons un processus qui construit un classifieur qui représente un document indépendamment des autres documents de la collection. Contrairement aux approches classiques, nous ne représentons pas un document par un vecteur dont la taille est égale au nombre de caractéristiques retenues pour représenter les différentes classes. Notre approche consiste à exploiter un ensemble d'ontologies de domaine pour construire une représentation conceptuelle d'un texte sous forme d'un graphe sémantique dans lequel les nœuds correspondent à des concepts extraits de l'ontologie de domaine qui représente le mieux son contenu. Seuls les termes ayant un sens dans l'ontologie candidate sont retenus pour constituer les nœuds du graphe d'un document. La formule *tf-idf*, largement utilisée, représente le pouvoir discriminant des mots d'un document relativement aux autres documents du corpus. Dans notre cas, le poids des concepts retenus détermine leur pouvoir représentatif vis-à-vis du domaine de connaissance dans lequel s'inscrit le contenu textuel du document. Nous décrivons dans les sections suivantes les différentes étapes de construction de notre classifieur.

3.2 Des ontologies pour représenter le contenu des documents

Notre processus que nous appelons CBO (classification basée ontologies) se base sur une classification sémantique des textes en exploitant des ontologies de domaine **[Iltache et al., 2016]**. La Figure 3.1 résume le processus de classification que nous mettons en œuvre.

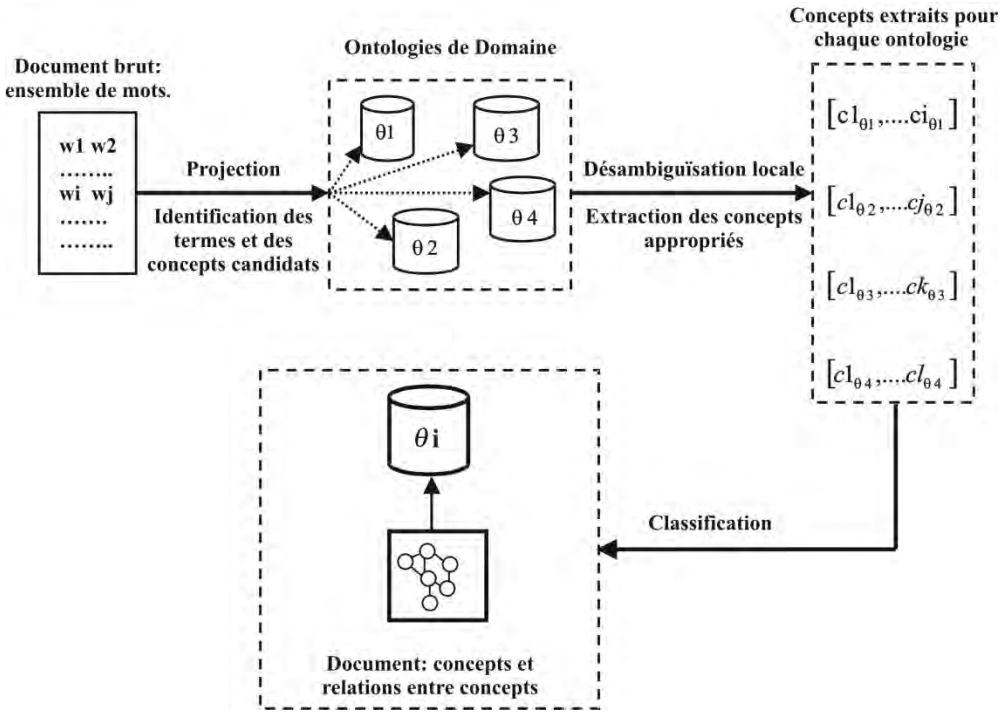

Figure 3.1 Classification d'un document.

Notre approche vise à construire, pour un document, un graphe dont les nœuds et les arcs sont représentés respectivement par des concepts et des relations entre concepts.

La classification des documents permet de regrouper des documents en fonction du domaine de connaissances défini par leur contenu. Ce regroupement identifie une similarité globale exprimée par le contexte dans lequel s'inscrivent les documents. A cette étape, les concepts correspondant à leur contenu sont sélectionnés. La classification que nous mettons en œuvre est sémantique, car contrairement aux approches classiques nous prenons en compte le lien entre les termes grâce à leur contexte d'apparition dans le document et à l'extraction des concepts correspondants à ces termes à partir des ontologies de domaine.

Notre processus comporte plusieurs étapes. Une première étape consiste à projeter le contenu d'un document sur plusieurs ontologies de domaine pour déterminer celle qui représente le mieux son contenu. Sachant que certains documents peuvent aborder plusieurs thèmes (sujets), notre processus doit retrouver le thème dominant qui permet de décider à quelle ontologie rattacher chaque document. Une deuxième étape permet de réaliser une désambiguïsation locale au sein de chaque ontologie. Une troisième étape classifie les documents en calculant un score pour chaque ontologie relativement à chaque document de la

collection. Ces trois étapes sont décrites dans les sections suivantes. Nous nous basons sur les faits suivants :

Un utilisateur peut utiliser les mêmes termes pour décrire des connaissances différentes. Ainsi, un terme peut avoir plusieurs sens selon le contexte dans lequel il est utilisé. Un même terme *ti* extrait d'un document *d* peut être alors affecté à plusieurs concepts appartenant à différentes ontologies comme montré par (3.1).

$$t_i^d = \{C_{\theta 1}, C_{\theta 2}, \dots \dots, C_{\theta n}\} \qquad (3.1)$$

θi représente la ième ontologies, $C_{\theta i}$ représente le concept extrait de l'ontologie θi pour le terme *ti*.

- Un terme extrait d'un document peut faire référence à plusieurs concepts au sein d'une même ontologie.
- Le thème abordé dans un document dépend des termes utilisés pour construire son contenu et la façon dont ces termes sont regroupés au sein d'une phrase et d'un paragraphe.

3.2.1 Projection, extraction des termes et des concepts candidats

Nous commençons par projeter un document sur les différentes ontologies afin d'identifier les termes représentant son contenu relativement aux concepts appartenant à ces ontologies et d'extraire les concepts adéquats correspondants à ces termes.

La "projection" d'un document sur les différentes ontologies permet d'associer le sens des termes composant le document avec les concepts appartenant à ces ontologies et de sélectionner les concepts candidats. La notion de concept donne un sens à un terme relativement au domaine où ce concept est défini.

Nous segmentons tout le document en phrases. Chaque phrase est parcourue de gauche à droite depuis le premier mot. Les mots de chaque phrase sont projetés, sans suppression des mots vides, sur un ensemble d'ontologies de domaines. Les concepts sont souvent représentés par plusieurs mots, nous extrayons alors les termes (groupes de mots adjacents dans une phrase) qui correspondent aux concepts les plus longs. Nous choisissons d'extraire les termes longs d'une phrase car les termes long sont moins ambigus que les mots les composant pris individuellement et déterminent mieux le sens véhiculé par le document.

Nous montrons dans l'exemple suivant l'intérêt d'extraire les termes les plus longs à partir d'une phrase. Considérons la phrase "*The Secretary of State for the Home Department had clearly indicated that evidence obtained by torture was inadmissible in any legal proceedings*". Les synsets représentés dans la Table 3.1 sont extraits de WordNet.

Mots dans la phrase	Labels des synset dans WordNet	N° des synset dans WordNet
Secretary of State for the home Department	secretary_of_state_for_the_home_department	**09526473**
	secretary_of_state	09883412 09455599 00569400
	Secretary	09880743 09880504 09836400 04007053
	State	07682724 08125703 07673557 00024568 07646257 08023668 13192180 13656873
	Home	08037383 03141215 07973910 13687178 03398332 07974113 07587703 03399133 08060597
	department	07623945 08027411 05514261

Table 3.1 Extraction des termes et leurs synsets correspondants.

Comme le montre la Table 3.1, il existe plusieurs synsets dans WordNet qui correspondent aux mots : *secretary of state for the home department* de la phrase. Ces synsets sont composés d'un ou plusieurs mots. Le terme le plus long "*secretary of state for the home departme*nt" est extrait de la phrase. Il correspond au synset *secretary_of_state_for_the_home_departmen*t, numéro 09526473 qui représente le sens utilisé dans la phrase.

Nous montrons dans la Table 3.2 comment se fait la projection sur différentes ontologies de domaine. Nous reprenons l'exemple précédent et choisissons trois domaines définis dans WordNet Domains, *administration, politics et enterprise*.

Termes commençant par le mot *Secretary*	Label des synsets dans WordNet	N° synset dans WordNet		
		Domain administration	Domain politics	Domain enterprise
"Secretary of state for the home department" "Secretary of state" "Secretary"	secretary_of_state_for_the_home_department	**09526473**	**09526473**	
	secretary_of_state	09883412 00569400	09455599	00569400
	secretary	09880743		

Table 3.2 Projection de *The Secretary of State for the Home Department* sur trois domaines.

Comme le montre la Table 3.2, nous avons pour les trois domaines plusieurs synsets commençant par le mot *secretary*. Ces synsets possèdent un ou plusieurs mots. Nous choisissons d'extraire de la phrase le terme le plus long secretary *of state for the home department* correspondant au synset *secretary_of_state_for_the_home_department* (09526473), qui représente le correct sens dans la phrase. Les deux domaines administration et politics repésentent alors le terme de la phrase par le synsets 09526473.

Dans l'étape projection, plusieurs concepts appartenant à une même ontologie de domaine peuvent être candidats pour un terme donné de la phrase. Un processus de désambiguïsation locale est nécessaire pour déterminer quel concept retenir pour chaque terme.

3.2.2 Désambiguïsation locale

Lors de la rédaction d'un texte, la structure habituellement adoptée est de séparer des idées en phrases et en paragraphes. Considérons le texte suivant :

Le terme générique définissant les virus est malware, le terme virus est utilisé couramment de manière abusive pour désigner l'ensemble des malwares. Il existe différents types de malwares : les virus, les vers, les chevaux de Troie ou les bombes logiques, certains se chargent en mémoire, d'autres infectent directement le disque dur. Les virus se répliquent et se propagent en s'insérant dans d'autres logiciels.

Les antivirus sont des logiciels informatiques ayant pour objectif de détecter et supprimer les virus et malwares du poste ou du flux analysé. Il existe plus de 50 antivirus commerciaux actuellement et quelques antivirus open-source. La qualité d'un antivirus dépend en grande partie de sa rapidité à identifier les nouveaux virus et à mettre à jour sa base de signatures antivirus.

Cet exemple est composé de deux paragraphes. Les phrases du premier paragraphe abordent la même notion de "virus" mais chacune d'elle traite une idée différente : la première phrase aborde l'autre nom avec lequel un virus est désigné. La deuxième phrase cite les différents types de virus et la troisième explique la manière dont le virus se propage. Le second paragraphe aborde la notion d'"anti-virus" qui est différente de celle du virus.

Cette structure nous conduit à émettre l'hypothèse que les termes se trouvant dans une même phrase ont plus de chance d'être proche sémantiquement, qu'avec les termes d'une autre phrase ou d'un autre paragraphe.

Prenons en exemple le texte suivant appartenant au domaine *music* et calculons les distances entre certains de ses termes à travers la ressource WordNet :

Similarity is an important concept in music cognition research since the similarity between (parts of) musical pieces determines perception of stylistic categories and structural relationships between parts of musical works. The purpose of the present research is to develop and test models of musical similarity perception inspired by a transformational approach which conceives of similarity between two perceptual objects in terms of the complexity of the cognitive operations required to transform the representation of the first object into that of the second, a process which has been formulated in information-theoretic terms. Specifically, computational simulations are developed based on compression distance in which a probabilistic model is trained on one piece of music and then used to predict, or compress, the notes in a second piece. The more predictable the second piece according to the model, the more efficiently it can be encoded and the greater the similarity between the two pieces. The present research extends an existing information-theoretic model of auditory expectation (IDyOM) to compute compression distances varying in symmetry and

normalisation using high-level symbolic features representing aspects of pitch and rhythmic structure. Comparing these compression distances with listeners' similarity ratings between pairs of melodies collected in three experiments demonstrates that the compression-based model provides a good fit to the data and allows the identification of representations, model parameters and compression-based metrics that best account for musical similarity perception. The compression-based model also shows comparable performance to the best-performing algorithms on the MIREX 2005 melodic similarity task.

Termes	N°synset	Distance
Music – part	06591368-n - 06600579-n	0,25
Music – piece	06591368-n - 06607179-n	0.14285715
Music - note	06591368-n - 06442795-n	0.5555556
Music -pitch	06591368-n - 04724487-n	0.8333333
Music- melody	06591368-n - 06598312-n	0.14285715
Part -piece	06600579-n - 06607179-n	0.25
Part-pitch	06600579-n - 04724487-n	0.875
Part-melody	06600579-n - 06598312-n	0.125

Table 3.3 Distances sémantiques entre synsets

La plupart des distances calculées et résumées dans la Table 3.3 montrent que les termes proches dans le texte ont de petites distances sémantiques. Ce qui signifie qu'ils sont proches sémantiquement. Néanmoins si nous regardons la distance reliant le terme *melody* aux termes *part* et *music*, nous remarquons que ces distances sont petites malgré l éloignement des termes dans le document. Ce qui montre que les termes appartenant à un document sont liés les uns aux autres relativement au sujet abordé dans le document et peuvent participer au processus de désambiguïsation.

Le processus de désambigüisation permet de sélectionner parmi plusieurs concepts candidats d'une même ontologie, le concept le plus approprié pour un terme *t* appartenant à un document. Pour ce faire, nous prenons en considération le contexte d'apparition du terme *t* dans le document et nous posons les hypothèses suivantes :

- Nous admettons que le lien sémantique reliant des termes dépend de la distance séparant ces termes au sein du document. Plus cette distance est courte, plus le lien sémantique est grand. Le lien sémantique décroit en passant de la phrase au paragraphe puis d'un paragraphe à un autre.
- Nous choisissons le concept approprié pour le terme *t*, en prenant en considération à la fois la distance sémantique reliant le terme *t* avec les termes voisins relativement à son contexte d'apparition et la distance sémantique reliant les

concepts associés au terme *t* aux concepts correspondants aux termes voisins dans l'ontologie considérée.

- Le sens d'un terme *t* dans un document est déterminé par ses termes voisins, non ambigus, les plus proches. *t* sera alors désambiguïsé soit par son voisin de gauche le plus proche, soit par son voisin de droite le plus proche. Dans le cas où les voisin de droite et de gauche existent simultanément, ils seront pris tous les deux en considération.

Le processus de désambiguïsation se fait alors selon trois niveaux : la désambiguïsation se fait d'abord au niveau phrase. Pour chaque phrase, les termes ambigus sont désambiguïsés en considérant leur voisin de gauche et leur voisin de droite, non ambigus, au niveau de la phrase. Un terme désambiguïsé permettra à son tour de désambiguïser un autre terme ambigu. Ce processus est réitéré dans le cas où il reste des termes ambigus en considérant dans une deuxième étape le niveau paragraphe puis, le cas échéant, le niveau document.

Le processus de désambiguïsation au niveau phrase se déroule comme suit ; nous considérons les termes voisins, non ambigus, qui possèdent des concepts associés dans l'ontologie considérée, qui entourent *t* : nous prenons le voisin *nl*, le plus proche qui se trouve à gauche de *t*, et le voisin *nr*, le plus proche qui se trouve à droite de *t*. Nous recherchons ensuite dans l'ontologie les concepts *Cnl* et *Cnr*, associés à *nl* et *nr* respectivement.

Le concept adéquat pour le terme *t* parmi les concepts candidats est le concept le plus proche sémantiquement de *Cnl* ou de *Cnr*. Cela revient à parcourir l'ontologie et à calculer la distance minimale entre chaque concept candidat associé à *t* et les concepts candidats *Cnl*, *Cnr*. Plusieurs métriques existant dans la littérature permettent de calculer cette distance minimale. Nous donnons, dans la Figure 3.2, l'algorithme de désambigüisation locale au niveau phrase. Un exemple de désambiguïsation local au niveau phrase est donné par la Figure 3.3.

```
Input
    Ec = {extracted concepts for S}  {S, current sentence}
    Et = {terms belonging to S}
    E = {Unambiguous terms of S}
Output
   Ec = {retained concepts for S}

Procedure disambiguation (i:integer)
  var
    j:integer
Begin
  t ←S[i]
  nl (t) ← S[i-1]
  nr (t) ← S[i+1]
  if (nl (t) in E ) and (nr (t) in E)  then
    compute  Min-dist ((Ci,Cnl), (Ci,Cnr))  {Ci, The concepts associated with t}
    E ← E ∪ t
    Ec ← Ec ∪ C          {C, retained concept for t}
  else
    if (nl (t) in E)  then
      compute  Min-dist (Ci,Cnl)    {Cnl: The concepts associated with nl}
    E ← E ∪ t
    Ec ← Ec ∪ C
   else
      if (nr (t) in E) then
        compute  Min-dist (Ci,Cnr)    {Cnr: The concepts associated with nr}
        E ← E ∪ t
        Ec ← Ec ∪ C
      else
        j ← i + 1
        disambiguation(j)
        pos ← pos + 1
        t ← S[j-1]
        compute  Min-dist (Cj-1,Cnr)
        E ← E ∪ t
        Ec ← Ec ∪ C
       End if
     End if
  End if
End
Begin
  Ec←∅
  pos ← 1
  k ← 1
  t ← S[k]
  while ( not end (S) ) do
    if ( t not in E)  then
      disambiguation (k)
       k ← pos + 1
       pos ← pos + 1
    else
      Ec ← Ec ∪ C
      pos← k+1
      k ←k + 1
    end if
    t ← S[k]
  end while
 end.
```

Figure 3.2 Désambiguïsation locale au niveau phrase.

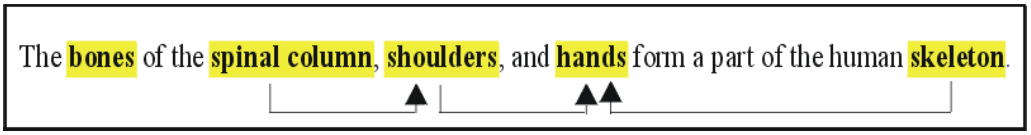

Figure 3.3 Désambiguïsation de *shoulder* et *hand*.

La Table 3.44 montre les termes et leur sens (synsets) dans le domaine *anatomy* de WordNet Domains. Les différentes distances calculées permettent de choisir le synset le plus approprié pour chaque terme ambigu.

Le terme *shoulder* dans la phrase est ambigu. Pour le désambiguïser, *spinal column*, son voisin gauche non ambigu le plus proche est considéré. Le synset retenu est 05231159.

Le terme *hand* dans la phrase est ambigu. Sa désambiguïsation, est réalisée en considérant *shoulder* et *skeleton*, ses deux voisins non ambigus de gauche et de droite les plus proches. Le synset retenu est 05246212.

Mots de la phrase	**Labels des synsets (domaine Anatomy)**	**N° synset**	**Distance entre synsets**	**Termes extraits**
Bones	bone	04966339		bone Spinal column shoulder hand skeleton
Spinal	Spinal_column	05268544		
Column	shoulder	**05231159** 05231380	Dist(05268544,05231159)= 0.42857143 Dist(05268544, 05231380)= 0.5	
Shoulders (ambigu)	hand	**05246212** 02352577	Dist(05246212,05231159)= 0.42857143 Dist(02352577,05231159)= 0.6363636 Dist(05246212,05265883)= 0.42857143 Dist(02352577,05265883)= 0.6363636	
Hands (ambigu)				
skeleton	skeleton	05265883		

Table 3.4 Désambiguïsation des termes ambigus

A l'issue des étapes précédentes, un document *d* est représenté par plusieurs ensembles de concepts extraits des ontologies de domaine θi sur lesquelles il a été projeté. Ces ensembles de concepts sont représentés par (3.2).

$$d = \begin{cases} \theta_1^d = \{c_{11}, c_{21}, \ldots, c_{n1}\} \\ \theta_i^d = \{c_{1i}, c_{2i}, \ldots, c_{ni}\} \\ \ldots\ldots\ldots \\ \ldots\ldots\ldots \end{cases} \qquad (3.2)$$

3.2.3 Classification : Désambiguïsation globale

Le classifieur doit pouvoir conclure de la pertinence d'un document relativement à un contexte donné et choisir parmi les différentes représentations ontologiques celle qui correspond le mieux à son contexte. Pour ce faire, en associant les différentes ontologies de domaine à des classes, le classifieur procédera à la classification d'un document relativement à une ontologie de domaine unique.

Les mots utilisés par un utilisateur pour décrire une idée donnée ne sont pas choisis arbitrairement, mais ils sont choisis en ayant un sens commun guidé par cette idée. Ils sont donc liés sémantiquement. Cependant, dans un document, il est rare que tous ces mots fassent référence à un seul domaine.

Rappelons que les étapes précédentes permettent d'extraire les concepts correspondant aux termes présents dans le document. Les concepts extraits peuvent appartenir à plusieurs ontologies.

La classification que nous définissons a pour objectif de déterminer, pour chaque ontologie θi, le poids sémantique de chaque concept extrait pour le document *d*. Ce poids détermine l'importance d'un concept relativement à un document. Il détermine ainsi son pouvoir représentatif vis-vis d'une ontologie. L'évaluation de ce poids ce fait à deux niveaux : niveaux paragraphe et niveau document.

Niveau paragraphe : Nous calculons le poids de chaque concept *Ci* en fonction des autres concepts apparaissant avec lui dans le paragraphe.

Niveau document : Nous calculons le poids total de chaque concept *Ci* dans tout le document. Ce poids est obtenu en additionnant les poids obtenus pour le concept *Ci* dans les différents paragraphes du document *d*.

Les différents termes composant un document, pris ensemble en tenant compte des relations contextuelles les reliant permettent une évaluation sémantique du contenu textuel. Un score est calculé pour chaque ontologie relativement à chaque document de la collection. Le plus grand score détermine l'ontologie candidate qui sera retenue pour représenter le document *d*. Pour chaque ontologie et pour chaque document nous associons une matrice définie par (3.3).

$$M_{\theta_i}^{d} = \begin{pmatrix} lc_1c_1 & lc_1c_2 \ldots\ldots & lc_1c_n \\ & & \\ lc_nc_1 & lc_nc_2 \ldots\ldots & lc_nc_n \end{pmatrix} \qquad (3.3)$$

Les lignes et colonnes de cette matrice représentent tous les concepts extraits de l'ontologie θi pour le document *d*. *Ci*, est tout concept extrait de l'ontologie θi après projection du document *d* sur θi et *lcicj* représente le poids du lien entre le concept *Ci* et le concept *Cj* (i≠j). Ce poids est calculé comme suit :

- La matrice est initialisée à zéro
- Si un terme *ti* et un terme *tj* apparaissent ensemble dans un même paragraphe du document *d* et les concepts *Ci* et *Cj* correspondent aux termes *ti* et *tj* respectivement, alors le poids *lcicj* =1.
- Le poids *lcicj* est mis à jour à chaque fois que les termes *ti* et *tj* apparaissent ensemble dans un même paragraphe.
- Le poids *lcici* correspond à l'apparition du terme *ti* dans le document *d*. Il est égal à 1.
- Le poids *lcicj* est mis à jour pour tous les paragraphes du document *d*.

Chaque ligne de la matrice représente le poids total d'un concept extrait de l'ontologie θi relativement au document *d*. Ce poids évalue l'importance du concept *Ci* dans le document *d*.

La somme des poids de tous les concepts, extraits d'une ontologie relativement au document *d*, mesure à quel point chaque ontologie représente ce document. Le plus grand score déterminera l'ontologie candidate qui sera retenue pour représenter le document *d*. L'algorithme de la Figure 3.4 résume le processus de construction d'une matrice θ pour un document *d*.

```
Input
    d={Pi}   {Pi=paragraphe du document d}
    Cd={Ci} {Ci=concepts extrait de l'ontologie θ pour d}
Output
    Matrix θ  { matrice correspondant à l'ontologie θ pour le document d }
Begin
{Initialisation}
    For i=1 to |Cd| do
       For j=1 to |Cd| do
          If (i=j) then
            θ[i,j] ← 1
          else
            θ[i,j] ← 0
          End If
       End for
    End for
{Mise à jour de la matrice}
    For each Pi ∈ d do
      For i =1 to |Cd-1| do
        For j = i+1 to |Cd| do
          If (Ci ∈ Pi) and (Cj ∈ Pi) then
             θ[i,j] ← θ[i,j] +Fr(Ci)×Fr(Cj)    { θ est une matrice symétrique}
             θ[j,i] ← θ[j,i] +Fr(Ci)×Fr(Cj)    {Fr(Ci) est le nombre d'occurrence
                                                  du concept Ci dans Pi}
          End if
        End for
      End for
    End for
End.
```

Figure 3.4 Construction de la matrice θ pour le document *d*

Nous illustrons le calcul du score d'une ontologie avec le texte *T* suivant :

A computer virus is a malware program that, when executed, replicates by inserting copies of itself (possibly modified) into other computer programs, data files, or the boot sector of the hard drive; when this replication succeeds, the affected areas are then said to be infected. Viruses often perform some type of harmful activity on infected hosts, such as stealing hard disk space or CPU time.

Virus writers use social engineering and exploit detailed knowledge of security vulnerabilities to gain access to their hosts' computing resources. The vast majority of viruses target systems running Microsoft Windows, employing a variety of mechanisms to infect new hosts.

Considérons le domaine *computer_science* appartenant à WordNet Domains et le texte *T*. nous extrayons les synsets ci dessous et nous construisons la matrice correspondante donnée par la Table 3.5. Les couleurs noire, bleue et rouge représentent respectivement les poids initiaux des concepts, leur poids après avoir parcouru le premier paragraphe et le deuxième paragraphe.

Synset	**Computer _virus / virus**	**Program / Computer_program**	**Sector**	**Hard_drive**	**host**	**Hard_disk**	**Cpu**	**Access**	**Computing**	**windows**	**Poids concept**
Computer _virus / virus 06179311	1	0+4	0+2	0+2	0+2+4	0+2	0+2	0+2	0+2	0+2	**25**
Program / Computer_program 06165318	0+4	1	0+2	0+2	0+2	0+2	0+2	0	0	0	**15**
Sector 12859914	0+2	0+2	1	0+1	0+1	0+1	0+1	0	0	0	**9**
Hard_drive 03093124	0+2	0+2	0+1	1	0+1	0+1	0+1	0	0	0	**9**
Host 04016750	0+2+4	0+2	0+1	0+1	1	0+1	0+1	0+2	0+2	0+2	**19**
Hard_disk 03364489	0+2	0+2	0+1	0+1	0+1	1	0+1	0	0	0	**9**
Cpu 02888449	0+2	0+2	0+1	0+1	0+1	0+1	1	0	0	0	**9**
Access 02579745	0+2	0	0	0	0+2	0	0	1	0+1	0+1	**7**
Computing 05762229	0+2	0	0	0	0+2	0	0	0+1	1	0+1	**7**
window 04410964	0+2	0	0	0	0+2	0	0	0+1	0+1	1	**7**
Score											**116**

Table 3.5 Calcul du score du document *T* relativement au domaine *computer_science.*

3.3 Evaluation du processus CBO

3.3.1 Les données

Nous avons implémenté notre processus de classification sémantique en utilisant simultanément les ressources WordNet **[Miller et al., 1995]** et WordNet Domains [**Magnini et al, 2000]**. Dans WordNet Domains plusieurs domaines de connaissances sont utilisés et chaque synset est annoté avec le ou les domaines dans lequel il possède un sens. Nous avons assimilé ces différents domaines à des ontologies de domaine. Nous avons utilisé la mesure de similarité Rita **[Howe, 2009]** pour mesurer la distance sémantique entre deux synsets dans WordNet. Les termes au sein des phrases sont annotés avec leur type (nom, verbe, adverbe, adjectif) par Stanford POS Tagger **[Toutanova et al., 2003]**.

Pour évaluer les classifieurs classiques sur notre collection, nous avons effectué un prétraitement sur les documents. Nous avons retenus les noms, les verbes et les adjectifs utilisés dans chaque document. Nous avons extrait les lemmes relatifs à ces termes puis calculé leur poids basé sur *tf-Idf*. Ce sont donc ces lemmes qui constitueront la représentation vectorielle des documents. Pour comparer notre approche, nous avons retenu trois types de classifieurs conventionnels : SVM **[Joachims, 1998]**, Naïve bayes **[Cheeseman et al, 1996]** et arbre de décision **[Quinlan, 1986].** Nous avons utilisé les algorithmes correspondant à ces classifieurs implémentés dans Weka **[Hall et al., 2009].**

Pour les classifieurs, SVM et arbre de décision, nous avons testé plusieurs paramètres et nous avons retenu ceux avec lesquels nous avons obtenu les meilleurs résultats. Les différents paramètres testés sont comme suit :

- Classifieur SVM

SMO (Sequential Minimal Optimization) définit la méthode des SVM implémentée dans Weka. Les paramètres considérés concernent le *noyau*, la *complexité*, l'*exposant* et *gamma.* Quand il est difficile de séparer n'importe quel jeu de données par un simple hyperplan (par exemple les données des deux classes se chevauchent sévèrement), une fonction non-linéaire appellée fonction noyau est utilisée pour projeter les points d'apprentissage dans un espace de dimension plus élevée. Au total, nous avons réalisé 16 combinaisons des différents paramètres correspondant à 16 modèles SVM. Nous gardons le modèle qui donne la meilleure exactitude.

Pour le noyau, deux fonctions sont testées. Un noyau polynomial et un noyau RBF. Pour le noyau polynomial, nous testons les valeurs 1, 2 et 3 de l'exposant. Un noyau polynomial avec un exposant égal à 1 est un noyau linéaire qui construit un hyperplan séparateur sous la forme d'une droite dans un espace à deux dimensions. Un noyau polynomial utilisé avec un exposant dont la valeur est supérieure à 1 construit un hyperplan séparateur sous la forme d'une courbe. Pour le noyau RBF (Gaussian Radial Basis Function) qui permet de définir d'autre type d'hyperplan séparateur, nous testons cinq valeurs de gamma (0.01, 0.03, 0.04, 0.25, 0.1). Le paramètre gamma est utilisé comme mesure de similarité entre deux points. Une petite valeur de gamma définit une fonction gaussienne avec une grande variance. Dans ce cas, deux points peuvent être considérés comme similaires même s'ils sont éloignés les uns des autres. Une grande valeur de gamma définit une fonction gaussienne avec une petite

variance et dans ce cas, deux points sont considérés similaires seulement s'ils sont proches les uns des autres.

Pour chaque fonction noyau, nous testons deux valeurs pour le paramètre *complexité* (1 et 10). Sa valeur permet de contrôler l'hyperplan séparateur en indiquant le nombre d'instances qui seront utilisées comme "vecteurs de support" pour tracer la frontière de séparation linéaire dans l'espace euclidien. *c* détermine le coût d'une mauvaise classification. Une petite valeur pour *c* augmente le nombre d'erreurs d'entraînement. Une valeur de *c* plus élevée augmente la pénalisation des erreurs de classification et réduit ainsi le taux d'erreur de classification sur les données d'entraînement. On augmente cette valeur dont le but de séparer au maximum les données de classes différentes.

- Classifieur arbre de décision

J48 est l'algorithme à base d'arbre de décision implémenté dans Weka. Le paramètre considéré pour les arbres de décision est le facteur de confiance *c* (confidence factor). Nous avons testé plusieurs valeurs de *c* (0.01, 0.02, 0.03, 0.04, 0.1, 0.2, 0.25, 0.3). La valeur de *c* influe sur la qualité de l'arbre. Des valeurs de *c* plus petite, donnent des arbres plus élagués. Au total huit arbres sont construits et nous avons gardé l'arbre qui donne la meilleure exactitude.

Notre évaluation couvre 10 domaines définis dans WordNet Domains et une collection composée de 976 résumés d'articles scientifiques. Quelques résumés du domaine médecine ont été extraits du corpus *Muchmore* qui est un corpus contenant des résumés d'articles scientifiques du domaine médical extraits à partir du site web de Springer. *Muchmore* est disponible en anglais et en allemand. Le reste des résumés scientifiques de notre corpus sont extraits à partir de plusieurs journaux scientifiques spécialisés dans les domaines que nous avons retenus, en parcourant leur site web. La Table 3.6 donne la répartition des différents résumés relativement aux domaines sélectionnés.

Domaines	**Nombre de résumés**
Music	106
Law	83
Computer_science	101
Politics	76
Physics	101
Chemistry	83
Economy	104
Buildings	104
Medicine	117
Mathematics	101
Total	**976**

Table 3.6 Répartition des résumés scientifiques par domaine

3.3.2 Résultats et discussion

Les mesures traditionnellement utilisées en catégorisation sont considérées dans nos travaux : *précision*, *rappel*, *F-mesure* et *accuracy*. Nous avons comparé les résultats de notre processus à ceux des classifieurs classiques. Les résultats obtenus sont résumés dans la Table 3.7.

Le rappel (*Rp*) détermine le nombre de documents bien classés dans une classe rapporté au nombre total de documents appartenant à cette classe. La précision (*Pr*) définit le nombre de documents bien classés dans une classe rapporté au nombre de documents affectés a cette classe. Une mesure qui combine la précision et le rappel est leur moyenne harmonique, nommée *F-mesure* (*F*). *Accuracy* (*Ac*) donne le pourcentage de documents bien classés sur le nombre total de documents de la collection. Les différentes équations sont données ci-dessous.

$$Rp = \frac{nombre\ de\ documents\ correctement\ attribués\ à\ la\ classe\ i}{nombre\ de\ documents\ appartenant\ à\ la\ classe\ i} \quad (3.4)$$

$$\mathrm{Pr} = \frac{nombre\ de\ documents\ correctement\ attribués\ à\ la\ classe\ i}{nombre\ de\ documents\ attribués\ à\ la\ classe\ i} \quad (3.5)$$

$$F = \frac{2 \times précision \times rappel}{précision + rappel} \quad (3.6)$$

$$Accuracy = \frac{nombre\ total\ de\ documents\ bien\ classés}{nombre\ total\ de\ documents\ du\ corpus} \quad (3.7)$$

Classes	CBO			Naive Bayes			SVM (SMO)			Tree C4.5 (J48)		
	Pr	**Rc**	**F**	**Pr**	**Rc**	**F**	**Pr**	**Rc**	**F**	**Pr**	**Rc**	**F**
Music	0,962	0,943	0,952	0,835	0,906	0,869	0,963	0,981	0,972	0,913	0,887	0,900
Law	0,952	0,964	0,958	0,777	0,880	0,825	0,947	0,867	0,906	0,766	0,711	0,737
Computer_science	0,970	0,950	0,960	0,845	0,861	0,853	0,872	0,941	0,905	0,474	0,644	0,546
Politics	0,949	0,974	0,961	0,788	0,829	0,808	0,944	0,882	0,912	0,754	0,645	0,695
Physics	0,960	0,960	0,960	0,833	0,842	0,837	0,887	0,931	0,908	0,513	0,386	0,441
Chemistry	0,940	0,952	0,946	0,947	0,867	0,906	0,986	0,880	0,930	0,848	0,807	0,827
Economy	0,980	0,962	0,971	0,820	0,788	0,804	0,855	0,904	0,879	0,541	0,442	0,487
Buildings	0,980	0,962	0,971	0,950	0,913	0,931	0,925	0,952	0,938	0,757	0,750	0,754
Medicine	1,000	0,983	0,991	0,982	0,940	0,961	0,991	0,991	0,991	0,894	0,863	0,878
Mathematics	0,925	0,980	0,952	0,904	0,842	0,872	0,898	0,871	0,884	0,493	0,673	0,569
Average	**0,964**	**0,963**	**0,963**	**0,872**	**0,869**	**0,870**	**0,926**	**0,924**	**0,924**	**0,694**	**0,682**	**0,683**
Accuracy (Ac)	**0,963**			**0,869**			**0,924**			**0,682**		

Table 3.7 Comparaison des résultats des différents classifieurs

Pour calculer les différentes valeurs pour SVM, Naïve Bayes et arbre de décision C4.5, nous avons réalisé une validation croisée (*cross-validation*) et nous avons retenu les résultats obtenus avec les meilleurs paramètres.

Dans la Table 3.7, nous pouvons voir que pour notre méthode, les valeurs du *rappel* et de la *précision* sont proches. Ces valeurs sont égales ou proches de 1, ce qui représente un bon indicateur de la bonne performance de notre classifieur. En considérant les moyennes des *précisions*, des *rappels*, des *F-mesure* ainsi que la valeur de *accuracy*, notre processus obtient de meilleurs résultats que les trois classifieurs classiques considérés. Le meilleur pourcentage de documents bien classés relativement à l'ensemble des documents du corpus est obtenu par notre processus de classification sémantique CBO.

Pour étudier la signification statistique de l'amélioration obtenue par notre processus, nous avons utilisé le test des rangs signés de Wilcoxon Nous avons calculé les P-value entre notre processus CBO et les autres classifieurs conventionnels. Ce test est basé sur les valeurs de F-mesures obtenus par CBO, Naïve bayes, SVM et C4.5. L'amélioration est considérée statistiquement significative si P-value est inférieure à 0.05 et très significative si P-value est inférieure à 0.01. Les différentes valeurs calculées sont résumées dans la Table 3.8.

	CBO - SVM	**CBO – Naive Bayes**	**CBO -Tree C4.5**
P-value (F-measure)	0.00885858	0.00294464	0.000976562

Table 3.8 Résultats du test de Wilcoxon

Les P-values obtenues avec le test de Wilcoxon sont toutes inférieures à 0.01. Cela nous permet de conclure que notre système de classification sémantique CBO améliore significativement le processus de classification des documents par comparaison aux classifieurs conventionnels au seuil $\alpha = 0.01$.

Les trois classifieurs conventionnels ont en commun la représentation des documents par des termes indépendants les uns des autres ainsi qu'une comparaison morphologique des termes contenus dans les documents. La comparaison est réalisée au niveau mot, alors que dans notre processus, la comparaison est réalisée au niveau contexte global du document. Un document est représenté par le domaine décrit dans son contenu. Ce domaine est déduit par les mots du document pris dans leur ensemble en considérant leurs relations dans le contexte dans lequel ils apparaissent. De plus, notre processus est construit à partir d'ontologies de domaine, ce qui représente une base plus stable qu'une collection de documents d'apprentissage. En effet, une modification dans le choix des documents constituant cette collection d'apprentissage entraine une modification des résultats des classifieurs conventionnels.

3.4 Conclusion

Dans ce chapitre, nous avons présenté notre processus de classification sémantique des documents. Le processus englobe plusieurs étapes permettant de passer du texte brut du document vers une représentation sémantique sous forme de graphe. L'étape de projection permet d'extraire les termes d'un document et les concepts candidats à partir des ontologies de domaine considérées. L'étape de désambiguïsation locale a pour objectif de retenir pour chaque terme ambigu, le concept approprié relativement à chaque ontologie de domaine. La dernière étape calcule un score pour chaque ontologie relativement à chaque document. Le score le plus élevé détermine l'ontologie pertinente pour représenter un document.

Des expérimentations sur un corpus de résumés d'articles scientifiques ont montré que les résultats obtenus par notre processus dépassent ceux des classifieurs conventionnels. Notre classifieur doit sa performance à l'utilisation conjointe des ontologies de domaines et des relations entre les mots dans le texte. Les mots du document pris dans leur ensemble ont permis de retrouver le domaine dans lequel s'inscrit le document.

La classification sémantique que nous avons définie constitue la première étape de notre processus de calcul de la similarité sémantique entre documents. La deuxième étape de notre approche permet de calculer une similarité entre documents appartenant à une même ontologie de domaine. Nous décrivons ce processus dans le chapitre suivant.

Chapitre 4

Similarité des textes : application aux résumés des articles scientifiques

4.1 Introduction

Le processus décrit dans le chapitre 3 nous a permis de classifier des documents en fonction des ontologies de domaines qui décrivent le mieux leur contenu. Notre intérêt porte à présent sur la sélection, à partir d'un corpus d'articles scientifiques, des articles susceptibles de représenter un risque de plagiat. Nous proposons une approche **[Iltache et al., 2018]** qui répond à cet objectif par l'examen des résumés des articles scientifiques rattachés à un même domaine de connaissance et pour lesquels nous calculons une "similarité locale".

Dans la section 4.2, nous définissons notre similarité textuelle. Cette similarité est basée sur la notion de périmètre sémantique que nous introduisons et sur l'enrichissement de graphes décrits dans la section 4.2.2. L'enrichissement appliqué en deux étapes à travers la construction du périmètre sémantique et la comparaison des graphes des documents a pour objectif de retrouver une similarité entre documents même si ces derniers n'utilisent pas les mêmes mots. Nous donnons ensuite dans la section 4.2.3 les différentes formules pour calculer la similarité entre deux textes. Ces formules mettent en avant les notions communes abordées par deux textes et mesurent à quel point leur contenu est similaire.

Dans la section 4.3, nous expliquons comment affiner le processus de calcul de similarité des textes pour l'appliquer à des textes scientifiques représentés par leurs résumés. La section 4.4 donne les résultats de nos expérimentations et enfin nous concluons sur l'intérêt de notre approche.

4.2 Similarité textuelle et périmètre sémantique

Un auteur, pour décrire le sujet de son document, évoque une ou plusieurs notions différentes. Il peut décrire ces notions en abordant plusieurs aspects appelés sous notions. Ces notions et/ou sous notions peuvent être décrites de façon générale ou précise selon le niveau de détail qu'il choisit de mettre en évidence.

Dans une ontologie, il existe une certaine structure définissant le sens des informations représentant un domaine de connaissances donné et la façon dont ces informations sont reliées entre elles. Cette structure est définie par plusieurs branches représentant des hiérarchies différentes. Chaque hiérarchie possède des ramifications pour séparer des données ayant des caractéristiques communes mais également des caractéristiques différentes. L'arbre de la Figure 4.1 inspirée par l'ontologie des figures géométriques décrite dans **[Bendaoud, 2009]** montre deux branches *Br1* (*figure)* et *Br2* (*angle*) représentant deux informations différentes. La branche *Br2* possède deux sous branches 2.1 et 2.2 correspondant respectivement à *angle droit* et *angle aigu*. *Angle droit* et *angle aigu* sont deux concepts ayant des caractéristiques différentes mais des caractéristiques communes définies par leur parent commun *angle*.

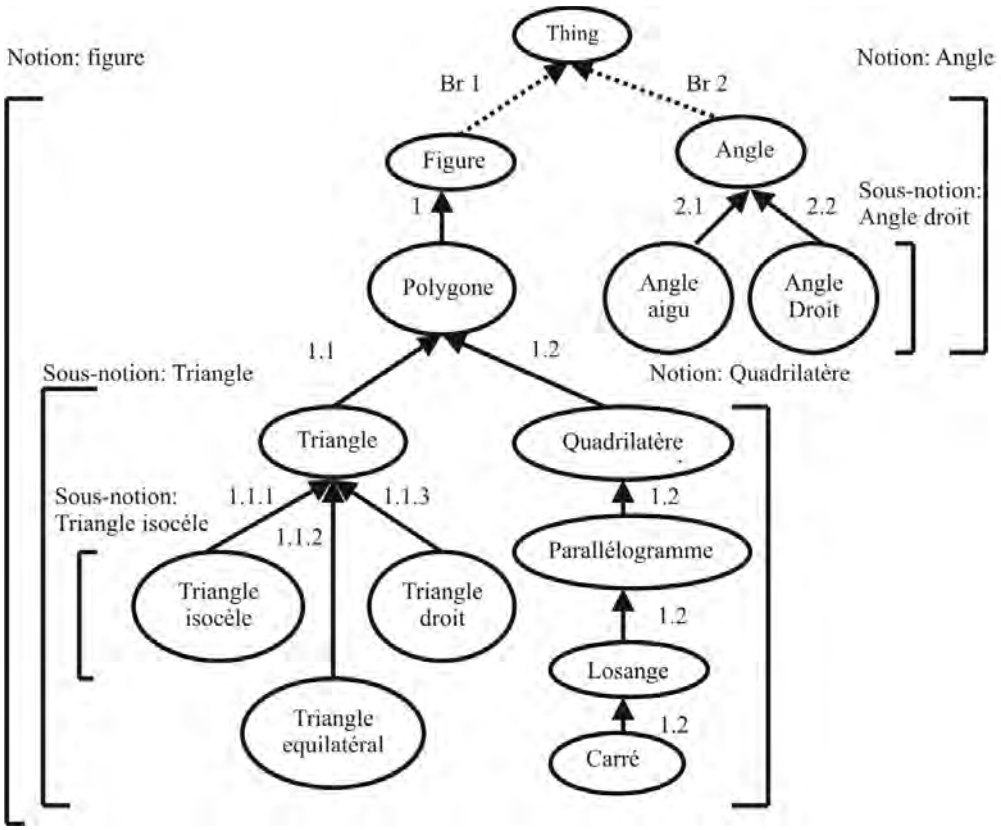

Figure 4.1 Extrait de l'ontologie des figures géométriques.

4.2.1 Objectif de l'approche

Considérons deux textes *T1* et *T2* dont il s'agit d'évaluer la similarité *Sim (T1, T2),* classés préalablement dans un même domaine de connaissance représenté par une ontologie de domaine. Nous posons les hypothèses suivantes :

1. Chaque branche/sous-branche de l'ontologie est associée à une notion/sous-notion décrite dans un document.
2. Des concepts reliés entre eux par des relations is-a forment une branche.
3. Une branche peut avoir plusieurs sous branches.
4. Deux branches n'ayant que pour seul parent commun la racine de l'ontologie (Thing) représentent deux notions différentes.
5. Deux sous branches ayant un parent commun représentent deux sous notions différentes partageant des caractéristiques communes définies par leur parent commun.
6. Le poids d'un concept initial est égal à 1.
7. Le poids d'un concept ajouté représentant une information implicite est inférieur à 1.
8. La similarité de deux textes varie entre 0 et 1.

Notre approche se base sur l'identification des branches auxquelles les concepts des documents appartiennent et sur l'enrichissement des graphes de ces documents. Associer une notion à une branche permet d'identifier les notions différentes et les notions identiques. Nous pouvons dire par exemple que la notion "angle" est différente de la notion "figure" ou que la notion "triangle" est différente de la notion "quadrilatère" car ils appartiennent à des branches ou à des sous branches différentes. Les concepts *quadrilatère*, *parallélogramme*, *losange* et *carré* appartiennent à une même sous branche décrivant une même notion. Chacun d'eux apporte un degré de précision sachant que cette précision est de plus en plus grande en allant vers le bas de la hiérarchie.

Les notions abordées dans un document sont souvent explicitées par des mots choisis par l'auteur. Ces notions peuvent faire référence à d'autres notions similaires mais non évoqué dans le texte. Prenons par exemple un document évoquant la notion de polygone en donnant des informations telles qu'une définition, des caractéristiques du polygone et un document abordant la notion de triangle. Ces deux documents présentent une certaine similarité puisque un triangle est un polygone. Il s'agit alors de trouver un moyen de faire ressortir cette information non explicitement citées dans le contenu textuel des documents. Cette information est présente de façon implicite dans leur contenu et peut être retrouvé par un processus d'enrichissement des graphes correspondant aux documents.

L'enrichissement des graphes permet de faire ressortir des notions communes à deux documents sans que celles-ci ne soient explicitement citées dans leur contenu et de déduire des similarités entre notions à travers l'examen des branches auxquelles leurs concepts appartiennent.

4.2.2 Enrichissement des graphes

Pour décrire un sujet donné, les auteurs, en fonction de l'importance que chacun d'eux souhaite donner à une notion qu'il aborde dans le texte, peuvent choisir des mots différents et

des niveaux de description différents. Ainsi, l'enrichissement des graphes par l'ajout de concepts permet de déduire une information implicite qui peut être partagée par ces deux textes.

A l'instar de Baziz [**Baziz, 2005b**], nous enrichissons les graphes de textes par l'ajout de concepts. L'enrichissement appliqué diffère de celui réalisé par Baziz dans le choix des concepts à rajouter et le poids affectés à ces derniers. Pour notre cas le poids affecté aux concepts permet de définir la présence implicite ou explicite d'un concept.

L'enrichissement d'un graphe est réalisé par la construction du périmètre sémantique du texte lui correspondant et lors de sa comparaison avec un autre graphe correspondant à un autre texte.

4.2.2.1 Construction du périmètre sémantique d'un texte

Définition1. Nous définissons le périmètre sémantique d'un document comme étant un graphe dont les nœuds sont des *concepts initiaux* et des *concepts liaison* correspondant à son contenu. Le périmètre sémantique ainsi construit pour chaque document permet d'évaluer leur similarité sémantique même si ces derniers expriment les mêmes idées avec des termes différents.

Les *concepts initiaux* sont extraits de l'ontologie de domaine à laquelle le document est rattaché. Ces concepts représentent l'information explicitement décrite dans son contenu. Avec ces concepts, nous construisons un graphe conceptuel que nous enrichissons par des *concepts liaison* représentant l'information implicite du texte déduite à partir des concepts initiaux et à travers le parcours des relations *is-a* et des relations transversales définies dans l'ontologie de domaine. Les relations transversales sont des relations permettant de représenter un lien sémantique entre concepts. Ce lien traduit une sémantique définie dans un domaine de connaissance et ne peut être représenté par la relation is-a. Ces relations possèdent un label. Par exemple, la relation possède reliant le concept *figure* au concept *segment* (une figure possède des segments) est une relation transversale.

- Construction du graphe de concepts initiaux

Lors du processus de classification, un texte est projeté sur un ensemble d'ontologies de domaine. A l'issue ce cette étape, ce texte est représenté par un graphe dont les nœuds constituent les *concepts initiaux*, extraits de l'ontologie à laquelle le document a été rattaché. Ces concepts correspondent aux termes explicitement cités dans le document.

- Construction du périmètre sémantique

Nous ajoutons au graphe d'un document, les *concepts liaison*, extraits de l'ontologie, se trouvant sur le chemin reliant les *concepts initiaux Ci* et *Cj* par des relations *is-a* ou par des relations transversales. Les concepts liaison sont extraits selon les algorithmes de la Figure 4.2 et la Figure 4.3.

```
Input
    ECinit={Cinit}   { concepts initiaux extrait pour un document d}
Output
    ECtransv ={Ctransv}  { concepts extraits par les relations transversales}
    EC=ECinit ∪ ECtransv
Begin
   for i= 1 to  |ECinit| -1 do
      CI ← ECinit[i]
      For  j= I+1 to  |ECinit| do
         CJ ← ECinit[j]
         If (chemintransv (CI,CJ)= true ) then  {CI est relié à CJ par des relations transversales}
           for each C ∈  chemintransv (CI,CJ) do   {relations transversales}
                add  (C, ECtransv)
           endfor
         else
           If  (ancetre(CI)  exists ) then
             ACI ← ancêtre (CI)
             If (chemintransv (CJ,ACI) =true ) then  {ancêtre(CI) est relié à CJ par des relations transversales}
               For each C ∈ chemin (CI, ACI) do  {relation is-a}
                 add (C, ECtransv)
               End for
               Add (ACI, ECtransv)
               For each C ∈  chemintransv (ACI,CJ) do
                 add (C, ECtransv)
               End For
             else
               If (ancetre(CJ)  exists ) then   {ancêtre(CI) reliés à ancêtre de (CJ)par des relations transversales}
                   ACJ ← ancetre(CJ)
                   If (chemintransv (ACI, ACJ) =true ) then
                       For each C ∈   chemin (CI, ACI) do
                           add (C, ECtransv)
                       End for
                       add( ACI, ECtransv)
                       For each C  ∈  chemin (CJ,ACJ)   do
                           Add (C, ECtransv)
                       End for
                       Add ( ACJ, ECtransv)
                       For each  C ∈  chemintransv (ACI,ACJ) do
                           Add  (C, ECtransv)
                       End for
                   End if
               End if
             End if
           Else
               If  (ancetre(CJ)  exists ) alors  {CI est relié a ancêtre (CJ)par des relations transversales}
                   ACJ ← ancetre(CJ)
                   If (chemintransv (CI,ACJ) =true ) then
                       For each C ∈  chemin (CJ,ACJ)  do
                           add (C, ECtransv)
                       End for
                       Add ( ACJ, ECliaison)
                       For each C ∈  chemintransv (ACJ,CI)  do
                         add (C, ECtransv)
                       End for
                   End if
               End if
           End if
         End if
       End for
   End for
   EC ←ECinit ∪ ECtrans
End.
```

Figure 4.2 Extraction des concepts liaison à travers les relations transversales

```
Input
 EC=ECinit ∪ ECtransv ) {concepts initiaux extrait pour un document d
                        et concepts extraits par les relations transversales}
Output
   ECL ={Cliaison}  {concepts liaison extrais par la relation is-a}
   EC Mis jour
Begin
   ECL ←Ø
   For i=1 to |EC| -1 do
     Ci ← EC[i]
     For j= i+1 to | EC| do
        Cj ← EC[j]
        For each C ∈ chemin (Ci,Cj) do {relations is -a}
              Add(C,ECL)
        End for
      End for
   End for
```

Figure 4.3 Extraction des concepts liaison à travers les relations is-a

Une sélection des *concepts liaison* est réalisée et nous ne retenons que les concepts ayant un sens relativement au domaine de connaissance représenté par l'ontologie. En effet, une ontologie peut contenir des concepts permettant de construire la structure de l'ontologie mais sans représenter un sens pour le domaine considéré. C'est le cas par exemple pour WordNet. La sélection des concepts liaison se fait selon l'algorithme de la Figure 4.4.

```
Procedure add(C, var E)
    Input
        D =domaine retenu pour le document d
        E= ensemble de concepts
    Output
        E = ensemble de concepts mis à jour

    Begin
        Dom ← {domaines où C possède un sens}
        If (D in Dom ) then
            If (C not in E) then
                E ←E ∪ C
            End if
        End if
    End.
```

Figure 4.4 Sélection des concepts liaison

Exemple : Considérons le texte *T* que nous projetons sur WordNet. Ce texte est classé dans le domaine *computer_science.* La Figure 4.5 montre les *synsets liaison* reliant, dans WordNet, les deux *synsets initiaux, host* et *hard_disk,* extraits pour *T*.

T : " Infection of **hosts** can cause unusual and harmful activities, such as stealing **hard disk** space."

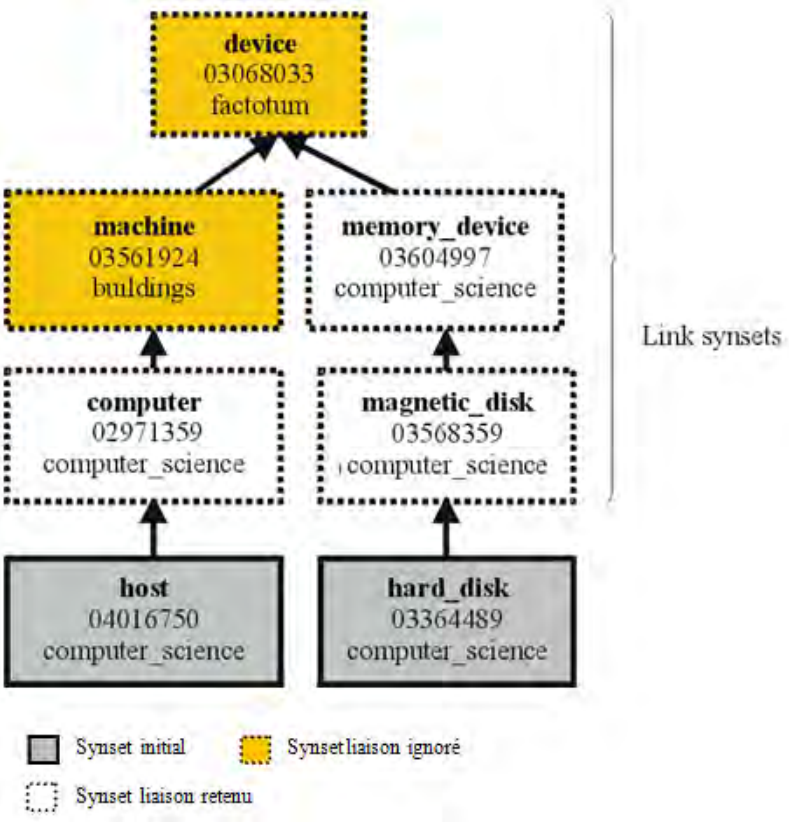

Figure 4.5 Synsets liaison reliant host à hard_disk.

Les synsets liaison sont : {computer 02971359, machine 02971359, device 03068033, memory_device 03604997, magnetic_disk 03568359}. Nous ne retenons pas les synsets, machine 02971359 et device 03068033 car ils appartiennent respectivement au domaine buildings et factotum. Rappelons que chaque synset dans WordNet est annoté par les domaines où il possède un sens.

4.2.2.2 Comparaison des graphes

Une comparaison de deux textes *T1* et *T2* est effectuée à partir de leur périmètre sémantique *G1* et *G2*. Un enrichissement mutuel de ces deux graphes est réalisé en comparant les concepts appartenant à *G1* avec les concepts appartenant à *G2*. Chaque graphe enrichi l'autre et des concepts sont ajoutés à *G1* et/ou à *G2*. Ceci est réalisé en parcourant les graphes de bas en haut comme suit :

- Si le graphe *G1* (le graphe *G2*) contient un concept *C1* et le graphe *G2* (le graphe *G1*) contient un concept *C2* tel que *C2* est un ancêtre de *C1*, alors le concept *C2* est ajouté au graphe *G1* (au graphe *G2*).
- Les graphes sont également enrichis par l'ajout des concepts parents communs aux concepts figurant dans les graphes *G1* et *G2*. Cet enrichissement se fait en deux étapes :
  - En mettant en correspondance les concepts appartenant seulement au graphe *G1* (au graphe *G2*).
  - En mettant en correspondance les concepts appartenant aux graphes *G1* et *G2*.

L'enrichissement par les concepts parents communs permet de déterminer les branches et les sous-branches communes à *G1* et *G2* et ainsi déduire une similarité implicite entre *T1* et *T2*.

A titre illustratif, dans le domaine *figures géométriques* représenté par la Figure 4.1, nous considérons trois textes *T1*, *T2* et *T3* dont le contenu est donné ci-dessous :

> T1 : *Un carré est un polygone régulier qui a quatre côtés. Il possède quatre angles droits et ses côtés ont la même mesure.*
>
> T2 : *Un losange est un parallélogramme. Certains losanges possèdent quatre angles de même mesure.*
>
> T3 : *Un triangle possède 3 côtés. S'il possède un angle droit, c'est un triangle rectangle.*

- Comparons les deux textes *T1* et *T2*.

Les périmètres sémantiques de *T1* et *T2* et la comparaison de leur graphe respectif *G1* et *G2* sont donnés par la Figure 4.6.

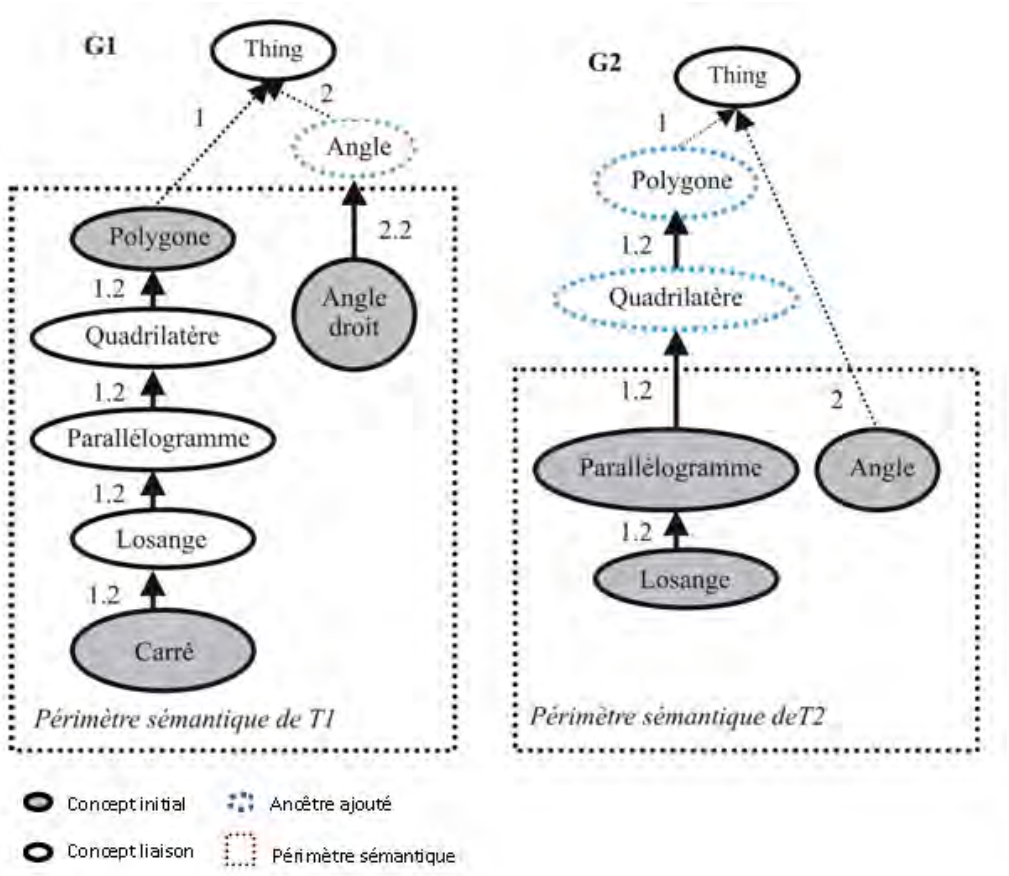

Figure 4.6 Comparaison et enrichissement des graphes correspondants à T1 et T2.

La projection des textes *T1* et *T2* sur l'ontologie, représentée par la Figure 4.1, nous permet de retrouver les concepts initiaux pour construire les graphes *G1* et *G2*.

G1 est représenté par les concepts (*carré*, *polygone*, *angle droit*) et *G2* est représenté par les concepts (*losange*, *parallélogramme*, *angle*). A cette étape, les graphes ne présentent aucun concept en commun. L'enrichissement de ces deux graphes a permis d'ajouter des concepts sémantiquement liés aux concepts initiaux et fait ressortir des concepts communs aux deux textes, non explicitement cités dans leur contenu. Les concepts communs sont *losange*, *parallélogramme*, *quadrilatère*, *polygone* et *angle*.

- Comparons les deux Textes *T2* et *T3*.

Les périmètres sémantiques de *T2* et *T3* et la comparaison de leur graphe respectif *G2* et *G3* sont donnés par la Figure 4.7.

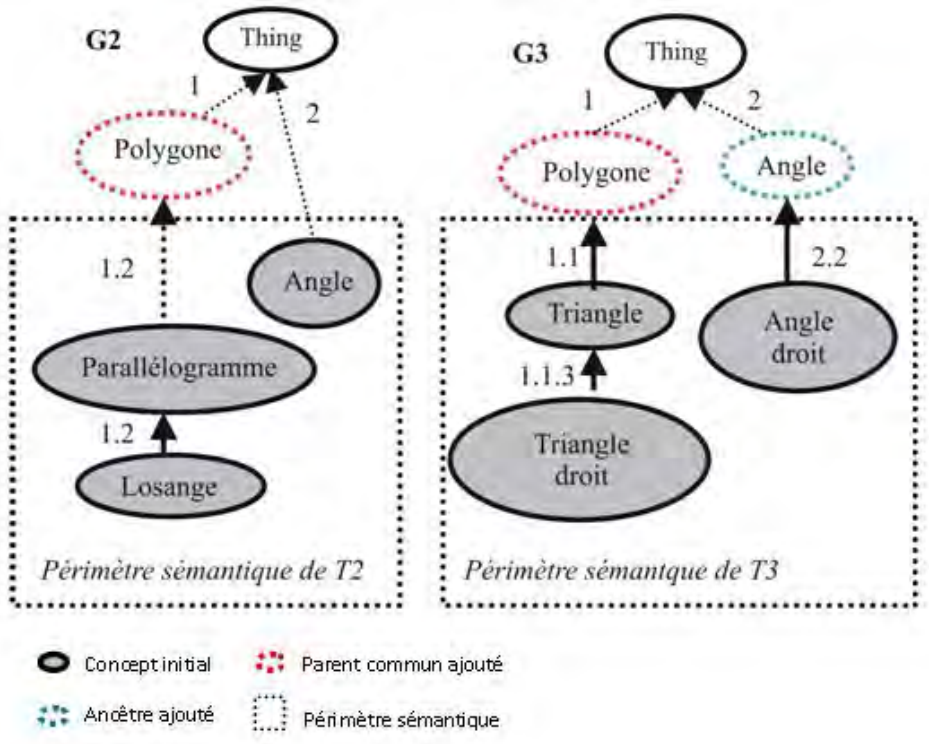

Figure 4.7 Comparaison et enrichissement des graphes correspondants à T2 et T3.

La projection des textes *T2* et *T3* sur l'ontologie, représentée par la Figure 4.1, nous permet de retrouver les concepts initiaux pour construire les graphes *G2* et *G3*.

G2 est représenté par les concepts (*losange*, *parallélogramme*, *angle*) et *G3* est représenté par les concepts (*triangle*, *triangle droit*, *angle droit*). L'enrichissement des deux graphes a permis de retrouver des concepts communs (*angle* et *polygone*).

4.2.3 Calcul de la similarité de deux textes

Définition 2. Nous définissons la similarité textuelle comme étant l'ensemble des notions et sous notions communes abordées par deux textes. Elle est fonction des concepts correspondant aux deux textes, de leur poids et des branches auxquelles ces concepts appartiennent. La similarité de deux textes (*T1*, *T2*) est donnée par la similarité de leur graphe respectif selon l'équation (4.1).

$$Sim(T1, T2) = Sim(G_{T1}, G_{T2}) \qquad (4.1)$$

4.2.3.1 Poids des concepts

Nous attribuons à un *concept initial* un poids égal à 1. Ce poids définit la présence explicite du concept dans le document. Les concepts appartenant à une même branche n'ont pas le même poids sémantique : les concepts se trouvant en haut de la hiérarchie ont un sens plus général que les concepts se trouvant en bas de la hiérarchie qui représentent un sens plus précis. Plus on descend vers le bas de la hiérarchie, plus le sens des concepts est précis. Ainsi,

à un concept ajouté au graphe *G1,* lors du processus d'enrichissement, nous attribuons un poids dont la valeur est inférieure à 1. Ce poids représente la valeur d'une information implicite et il est calculé sur la base du paramètre *g*. *g* exprime le degré de généralisation d'un concept père vis-à-vis de son concept fils.

Dans plusieurs approches en recherche d'information **[Fuhr et al., 2001] [Baziz, 2005b]**, **[sauvagnat, 2005]** le poids des nœuds d'un arbre est calculé par propagation en parcourant la hiérarchie de bas en haut, par multiplication par un facteur dont la valeur est comprise entre 0 et 1. Dans notre cas, nous proposons de calculer uniquement le poids des concepts ajoutés. Le poids des concepts initiaux étant égal à 1. Le poids d'un concept ajouté est calculé en utilisant le paramètre *g* dont la valeur est comprise entre 0 et 0.1 selon l'équation (4.2).

$$P(C_j) = 1 - (g \times (length(C_i, C_j)) \qquad (4.2)$$

Cj est le concept ajouté et *Ci* est le concept initial appartenant à *G1* et/ou à *G2*, le plus bas de la branche à laquelle *Cj* est ajouté et *length(ci,cj)* indique le nombre d'arcs reliant *Cj* à *Ci* dans la branche.

4.2.3.2 Similarité sémantique entre deux graphes G1 et G2

Nous introduisons un facteur indiquant le pourcentage de notions communes décrites par deux textes. Sa valeur est calculée par le nombre de branches communes rapporté au nombre de branches total appartenant aux deux graphes. La similarité de deux graphes *G1* et *G2* est calculée par l'équation (4.3).

$$Sim(G1, G2) = \frac{nbBc_{(G1,G2)}}{nbB_{(G1,G2)}} \times \frac{\sum_{Bc} \sum_{Ccom \in Bc} P(C_{com})}{\sum_{B} \sum_{C \in B} P(C)} \qquad (4.3)$$

B représente toute branche appartenant aux graphes *G1*,*G2* et *Bc* une branche commune aux deux graphes. *C* est un concept appartenant aux graphes *G1*, *G2* et *Ccom* est un concept commun aux deux graphes. *nbBc(G1,G2)* et *nbB(G1,G2)* représentent respectivement le nombre de branches communes et le nombre de branches total appartenant aux deux graphes.

4.2.3.3 Exemple

Nous reprenons les exemples représentés par la Figure 4.6 et la Figure 4.7 et récapitulons les différents résultats obtenus dans Table 4.1 et Table 4.2. Pour le paramètre *g*, nous utilisons la valeur 0.05. Rappelons que *g* exprime le degré de généralisation d'un concept père vis-à-vis de son concept fils. Nous avons testé plusieurs valeurs pour *g* et la valeur 0.05 est celle qui a donné les meilleurs résultats.

Textes	Concepts	Type	Poids
T1	carré	initial	1
	losange	liaison	0,95
	parallélogramme	liaison	0,90
	quadrilatère	liaison	0,85
	polygone	initial	1
	angle	ancêtre	0,95
	Angle droit	initial	1
T2	losange	initial	1
	parallélogramme	initial	1
	quadrilatère	ancêtre	0,85
	polygone	ancêtre	0,80
	Angle	initial	1
Branches communes		1 1.2 2	
Toutes les branches		1 1.2 2 2.2	

Table 4.1 Concepts de *T1* et *T2* après enrichissement de leur graphe respectif.

Texts	Concepts	Type	Poids
T2	losange	initial	1
	parallélogramme	initial	1
	polygone	Parent commun	0,85
	angle	initial	1
T3	angle droit	initial	1
	triangle droit	initial	1
	triangle	initial	1
	polygone	Parent commun	0,85
	angle	ancêtre	0,95
Branches communes		1 2	
Toutes les branches		1 1.1 1.2 1.1.3 2 2.2	

Table 4.2 Concepts de *T2* et *T3* après enrichissement de leur graphe respectif.

$$Sim(T1,T2) = \frac{3}{4} \times \frac{(0,80) + (0,85+0,90+0,95) + (0,95)}{(1) + (0,85+1+1+1) + (1) + (1)} = 0,49$$

$$\mathrm{Sim}(\mathrm{T2},\mathrm{T3}) = \frac{2}{6} \times \frac{(0,85) + (0,95)}{(0,85) + (1+1) + (1) + (1) + (1) + (1)} = 0,09$$

Initialement, *G1* et *G2* ne présentaient aucun concept en commun et donc a priori aucune similarité. L'enrichissement de ces deux graphes a permis de faire ressortir une similarité entre les deux textes non explicitement décrite dans leur contenu. Les résultats montrent également que le texte *T2* est plus proche sémantiquement de *T1* que de *T3*.

4.3 Application aux résumés scientifiques

4.3.1 Caractérisation du contenu textuel des résumés

Plusieurs travaux se sont intéressés à l'annotation de la structure discursive des articles scientifiques (text zoning) **[Omodei et al., 2014][Guo et al., 2011].** Leur objectif est de mieux caractériser le contenu des articles en définissant plusieurs classes (objectif, méthode et résultats, conclusion etc.), sachant que l'existence de ces classes dépend du corpus étudié. La catégorisation s'effectue au niveau phrase. Pour chaque phrase d'un résumé, les auteurs associent une classe choisie parmi les classes définies.

Notre approche traite de la décomposition des résumés des articles scientifiques en zones à des fins de détection de plagiat. A partir de la structuration généralement reproduite par les auteurs d'articles scientifiques, nous proposons de décomposer le contenu d'un résumé scientifique en trois parties distinctes, que nous nommons zones, définissant respectivement le *contexte*, la *contribution* et le *domaine d'application*. Nous considérons en effet que ce découpage se retrouve dans la plupart des articles scientifiques destinés en principe à faire part d'une contribution scientifique dans un domaine donné. Ce découpage a pour objectif d'extraire les notions relatives à chaque zone et permet ainsi de faire une comparaison entre les parties de même type. Nous pouvons alors évaluer, dans une approche progressive, si deux résumés traitent du même contexte, si leurs contributions sont similaires et s'ils appliquent leur approche à un même domaine d'application, le risque de plagiat évoluant évidemment à la hausse à chaque comparaison concluante.

La catégorisation au niveau phrase pose un problème lorsque des informations d'une classe se trouvent citées dans une autre classe. En analysant plusieurs résumés, nous avons constaté qu'il n'y a pas une uniformité stricte lors de la rédaction du résumé : toutes les phrases appartenant à une zone ne contiennent pas uniquement les termes décrivant cette zone mais peuvent contenir des termes représentant une autre zone. Par exemple, une phrase attribuée à la zone "domaine d'application" peut contenir des termes définissant un algorithme ou une méthode (des termes qui définissent plutôt la zone "contribution"). Cette imbrication de plusieurs zones dans une même phrase génère alors des erreurs d'étiquetage.

Pour la suite de cette section et pour illustrer notre approche, nous considérons deux résumés *A1* et *A2* extraits de deux articles scientifiques. Ces articles traitent de l'enrichissement des ontologies. Publiés en français, nous les avons traduits pour le besoin de notre travail.

***A1** : Ontology enrichment based on sequential pattern.*

The mass of information now available via the web, in constant evolution, requires structuring in order to facilitate access and knowledge management. In the context of the Semantic Web, ontologies aim at improving the exploitation of informational resources, positioning themselves as a model of representation. However, the relevance of the information they contain requires regular updating, and in particular the addition of new knowledge. In this paper, we propose an ontologies enrichment approach based on data mining techniques and more specifically on the search for sequential patterns in textual documents. The presented approach has been tested and evaluated on an ontology of the water domain, which we have enriched from documents extracted from the Web.

Keywords : ontology, enrichment, semantic web, data mining, sequential pattern

***A2** : Web usage mining for ontology enrichment.*

Recently, new approaches have integrated the use of data mining techniques in the ontologies enrichment process. Indeed, the two fields, data mining and ontological meta-data are extremely linked : on one hand data mining techniques help in the construction of the semantic Web, and on the other hand the semantic Web assists in the extraction of new knowledge. Thus, many works use ontologies as a guide for the extraction of rules or patterns, allow to discriminate the data by their semantic value and thus to extract more relevant knowledge. It turns out, however, that few works aimed at updating the ontology are concerned with data mining techniques. In this paper, we present an approach to support the ontologies management of websites based on the use of Web Usage Mining techniques. The presented approach has been tested and evaluated on a website ontology, which we have constructed and then enriched based on the sequential patterns extracted on the log.

Keywords: Semantic Web, ontology, Web Usage Mining, enrichment, data mining, sequential pattern

Pour illustrer la catégorisation au niveau phrase telle que c'est réalisé dans **[Guo et al., 2011],** considérons le résumé *A2*. Chaque phrase de *A2* est associée à l'une des trois zones que nous avons définies.

<Contexte> *Recently, new approaches have integrated the use of data mining techniques in the ontology enrichment process.* </Contexte>

<Contexte> *Indeed, the two fields, data mining and ontological meta-data are extremely linked : on one hand data mining techniques help in the construction of the semantic Web, and on the other hand the semantic Web assists in the extraction of new knowledge.* </Contexte>

<Contexte> *Thus, many works use ontologies as a guide for the extraction of rules or patterns, allow to discriminate the data by their semantic value and thus to extract more relevant knowledge.* </Contexte>

<Contexte> *It turns out, however, that few works aimed at updating the ontology are concerned with data mining techniques*. </Contexte>

<Contribution> *In this paper, we present an approach to support the ontologies management of websites based on the use of Web Usage Mining techniques.* </Contribution>

<Domaine d'application>*The presented approach has been tested and evaluated on a website ontology , which we have constructed and then enriched based on the sequential patterns extracted on the log*. </Domaine d'application>

En analysant le résumé ci-dessus, nous constatons les incohérences suivantes :

- Le terme *sequential pattern* se retrouve affecté à la zone *domaine d'application* alors qu'il représente l'algorithme et la méthode utilisée. Il définit donc la *contribution*.

- Le terme *data mining technique* est assigné à la zone *contexte* alors qu'il représente la contribution.

- Le terme *ontologies management* est attribué à la zone *contribution* alors qu'il définit le contexte.

Pour évaluer la similarité sémantique des deux résumés *A1* et *A2*, nous les avons découpés préalablement comme illustré ci-dessus. Pour chaque résumé, nous avons créé et enrichi trois graphes : un graphe pour chaque zone que nous avons définie. Trois similarités ont été calculées en mettant en correspondance les graphes représentant la même zone. Les valeurs de similarités obtenues sont très faibles. Cela se justifie par l'attribution des termes à une zone alors qu'ils définissent sémantiquement une autre zone, conséquence de la décomposition basée sur la catégorisation au niveau phrase.

Pour pallier ce problème, nous associons une ou plusieurs zones à une même phrase. Nous attribuons les termes de chaque phrase d'un résumé aux zones appropriées en fonction du sens global véhiculé par son contenu. A partir du sens global d'un résumé, on peut déduire le sens et la fonction des termes le composant. Un terme peut décrire le contexte dans lequel s'inscrit l'article (classification supervisée des documents, classification non supervisée des documents, classification supervisée d'images, enrichissement des ontologies, recherche d'information etc.), ou la contribution (les méthodes et les algorithmes ainsi que les notions permettant de les décrire) ou le domaine d'application (classification appliquée à une collection de documents donnée, fouille de données appliquée au textes, fouille de données appliquée au web, fouille de données appliqué aux images etc.).

Par exemple, pour la phrase *In this paper, we present an approach to support the ontologies management of websites based on the use of Web Usage Mining techniques*, nous associons trois zones : *contexte*, *contribution* et *domaine d'application*, car le terme *ontology management* définit le *contexte,* le terme *website* définit le domaine d'application et les termes *web usage mining* et *technique* définissent la *contribution*. La décomposition de cette phrase en zones est comme suit :

<Contexte>In this paper, we present an approach to support the ontologies management</contexte> <domaine d'application>of websites </domaine d'application> <Contribution>based on the use of Web Usage Mining techniques</Contribution>

La fonction de chaque terme est définie en fonction du domaine de connaissance où il est utilisé. Par exemple, un terme peut définir le contexte dans une ontologie et le domaine d'application dans une autre ontologie.

Nous utilisons également les termes contenus dans les titres ainsi que les mots clés car ces termes peuvent contenir des informations non citées dans les résumés.

L'annotation sémantique des concepts a été réalisée notamment dans WordNet Domains. Dans WordNet Domains **[Magnini et al, 2000]**, différents domaines sont définis tels que médecine, architecture, informatique, sport. Chaque synset de WordNet **[Miller et al., 1995]** est annoté par le ou les domaines où ce synset possède un sens. Nous adoptons également l'annotation des concepts d'une ontologie de domaine par la zone où ce concept possède un sens. Cette annotation est réalisée manuellement dans la perspective de rendre cette tâche automatique.

4.3.2 Similarité des résumés

L'extraction, à partir du contenu d'un résumé, des concepts correspondant à chaque zone se fait par la projection des termes du résumé sur l'ontologie. La comparaison de deux résumés revient à comparer les zones jouant le même rôle. Trois similarités partielles sont alors calculées par la mise en correspondance des concepts des deux résumés appartenant à la même zone. Nous pouvons ainsi comparer deux résumés à trois niveaux. Une similarité globale de deux résumés scientifiques *A1* et *A2* est obtenue par combinaison des trois similarités partielles selon l'équation (4.4).

La similarité globale permet de classer les résumés scientifiques par ordre décroissant de leur similarité comme illustré dans les tables Table 4.5, Table 4.6 et Table 4.7.

$$Sim(A1, A2) = \alpha\, sim_{contexte}(A1, A2) + \beta\, sim_{contribution}(A1, A2) + \gamma\, sim_{domaine\ application}(A1, A2) \quad (4.4)$$

α, β, γ sont des paramètres, dont les valeurs sont comprise entre 0 et 1, permettant de définir l'importance accordée au contexte, à la contribution et au domaine d'application. α+ β+ γ=1. Les valeurs des paramètres α, β et γ dépendent de l'ontologie et du corpus utilisés. Elles sont fixées par expérimentation.

Les documents traités ne sont pas forcément suspects puisqu'il est possible de mettre en œuvre cette approche pour comparer un document en cours de revue, par exemple, à tout un fonds documentaire, sans a priori quant à son respect de l'éthique scientifique. Un seuil de

similarité, déterminé par expérimentation et en fonction de l'ontologie et de la collection de résumés scientifiques utilisés, permet de faire le rapprochement de deux articles scientifiques et de déterminer si un risque de plagiat existe. Des résumés présentant une similarité élevée nécessiteront alors un examen complet de tout le document.

4.3.3 Mise en œuvre de notre approche

Pour illustrer notre approche, nous avons construit une ontologie associée au domaine de l'*enrichissement des ontologies* représentée par la Figure 4.8. Les concepts de l'ontologie sont annotés par les différentes zones que nous avons retenues pour caractériser le contenu d'un résumé scientifique. Nous reprenons les résumés *A1* et *A2* sur lesquels nous appliquons les différentes étapes de notre approche.

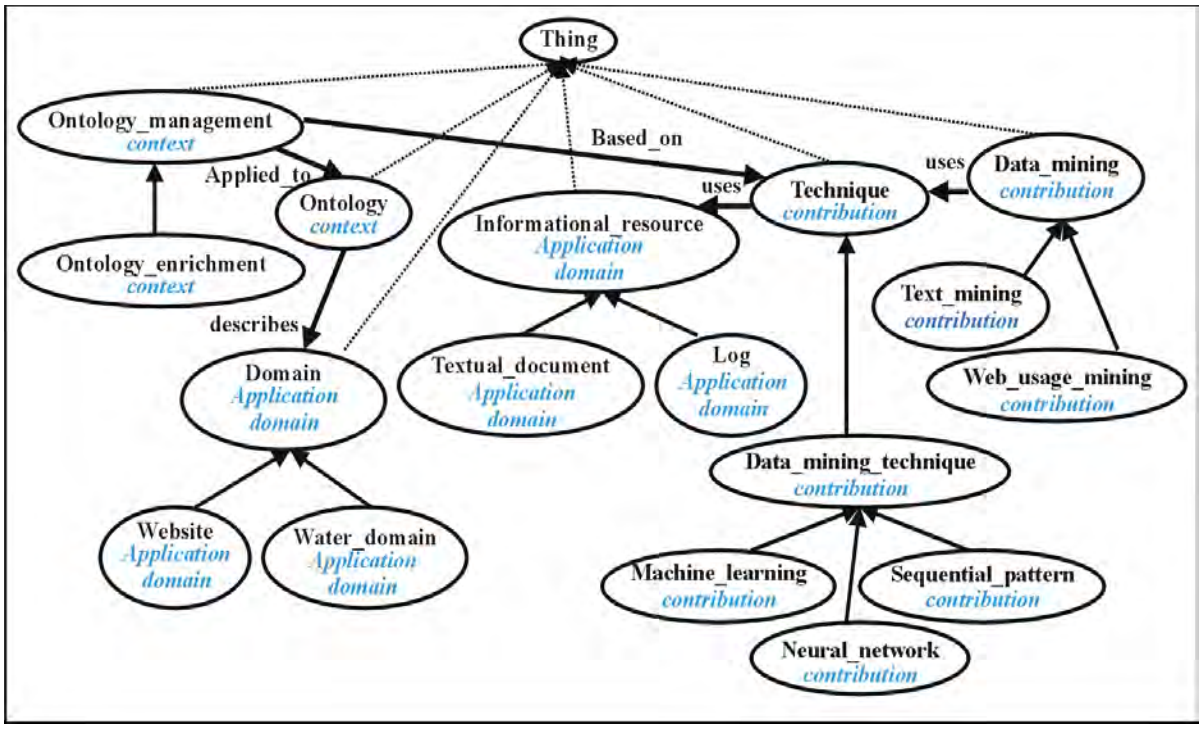

Figure 4.8 Extrait de l'ontologie du domaine enrichissement des ontologies, annotation des concepts par leur zone.

4.3.3.1 Extraction des concepts initiaux pour chaque résumé

La projection et l'extraction des *concepts initiaux* se fait à l'étape classification. Les deux résumés sont rattachés à l'ontologie représentée par la Figure 4.8. A chaque concept correspond une zone.

4.3.3.2 Enrichissement des graphes correspondant aux deux résumés

A partir des *concepts initiaux* nous enrichissons les graphes des deux résumés par la construction de leur périmètre sémantique et par comparaison de leur graphe. Les graphes enrichis des deux résumés *A1* et *A2* sont représentés par la Figure 4.9 et la Figure 4.10 et la répartition par zone des *concepts initiaux* et des concepts ajoutés par enrichissement de chaque résumé est donnée dans la Table 4.3.

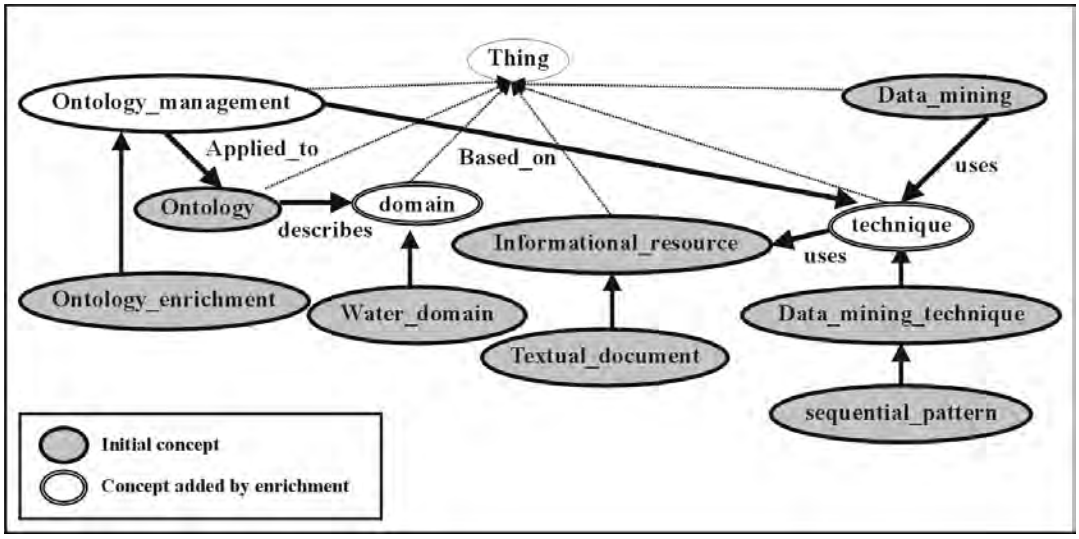

Figure 4.9 Graphe enrichi de A1.

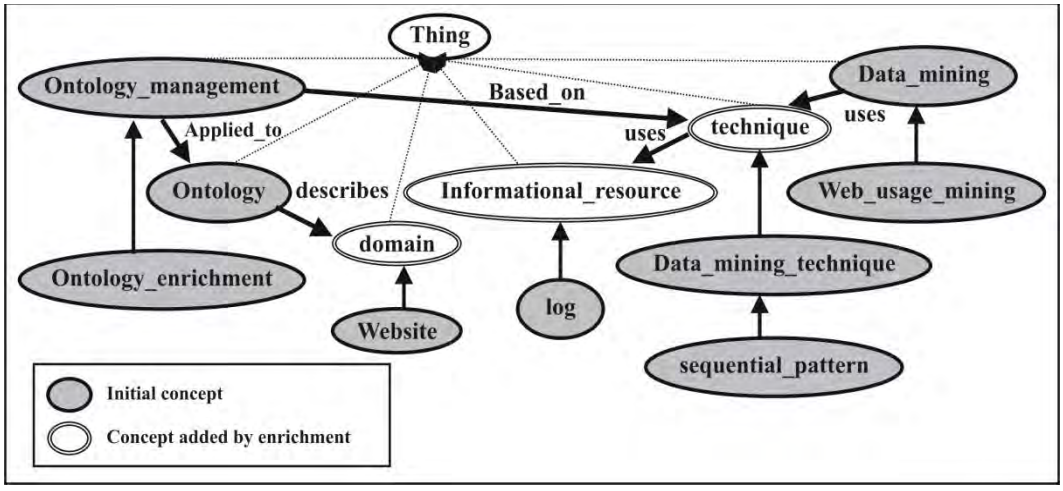

Figure 4.10 Graphe enrichi de A2.

	A1		A2	
Zones	**Concepts de A1**	**Concept type**	**Concepts de A2**	**Concept type**
contexte	Ontology_management	Ajouté	Ontology_management	Initial
	Ontology_enrichment	Initial	Ontology_enrichment	Initial
	Ontology	Initial	Ontology	Initial
contribution	Data_mining	Initial	Data_mining	Initial
	Technique	Ajouté	Technique	Ajouté
	Data mining_technique	Initial	Data_mining_technique	Initial
	Sequential_pattern	Initial	Sequential_pattern	Initial
			Web_usage_mining	Initial
Domaine d'application	Informational_resource	Initial	Informational_resource	Ajouté
	Textual_document	Initial	log	Initial
	Domain	Ajouté	Domain	Ajouté
	Water_domain	Initial	Website	Initial

Table 4.3 Distribution par zone des concepts de A1 et de A2.

4.3.3.3 Calcul de la similarité entre les résumés A1 et A2

Nous récapitulons dans la Table 4.4 les valeurs des similarités partielles et la valeur de la similarité globale obtenues entre A1 et A2. (valeurs obtenues avec $\alpha = 0.35$, $\beta = 0.63$, $\gamma = 0.02$, g =0,05. Ici, nous avons gardé les mêmes valeurs pour les paramètres que celles utilisées lors de nos expérimentations pour le domaine de la *classification des documents*. (cf. section 4.4).

$Sim_{contexte}$(A1,A2)	0,98
$Sim_{contribution}$ (A1,A2)	0,59
$Sim_{domainapplication}$(A1,A2)	0,10
Sim (A1,A2)	0,72

Table 4.4 Similarités entre Abstract1 et Abstract2.

4.3.3.4 Evaluation du résultat de la comparaison entre A1 et A2

Les résultats obtenus pour les résumés *A1* et *A2* indiquent que ces deux résumés traitent le même contexte (sim contexte = 0.98) avec des approches similaires. La similarité obtenue pour la contribution est élevée (sim contribution = 0.59). Ces deux résumés différent au niveau du domaine d'application puisque la valeur obtenue pour la similarité au niveau de cette zone est très faible (sim domaine application= 0.10). La similarité globale obtenue est

élevée (sim globale = 0.70). Cette valeur indique que les articles associés à ces deux résumés devraient faire l'objet d'une analyse plus approfondie qui pourrait révéler ou non un cas de plagiat.

4.4 Expérimentations

4.4.1 Les données

Nous avons étendu notre implémentation en ajoutant les parties nécessaires pour la construction du périmètre sémantique, pour la division du contenu des résumés scientifiques en trois zones, pour la comparaison des graphes et pour le calcul des différentes similarités.

Pour évaluer notre approche définissant la similarité sémantique des résumés scientifiques, nous avons construit une ontologie représentant le domaine de la *classification automatique des documents*. Pour construire notre corpus, un ensemble de résumés scientifiques relatifs à ce domaine ont été extraits du web. Nous avons retenu deux contextes pour ce domaine : *classification supervisée* et *classification non supervisée*. Les documents sont choisis de façon à ce que les notions abordées dans leur contenu présentent des similitudes et des différences que se soit au niveau de la contribution qu'au niveau du domaine d'application. Dans nos différents tests, nous avons pris en compte le résumé, le titre de l'article et les mots clés. Nous avons comparé les documents deux à deux. Nous donnons comme exemple les résultats obtenus en comparant 20 résumés scientifiques que nous nommons A1, A2,…A20, pour lesquels 190 comparaisons ont été réalisées. La construction du graphe initial, du périmètre sémantique de chaque résumé et la comparaison des graphes se fait selon le processus défini dans les sections précédentes.

Nous avons annoté chaque concept de notre ontologie par les zones caractérisant le contenu des résumés scientifiques où il possède un sens : *contexte*, *contribution* et *domaine d'application*. Cette annotation est réalisée en fonction du rôle que joue chaque concept relativement au domaine retenu. Par exemple, les concepts *clustering*, *classification* et *document* sont annotés par la zone *contexte*, les concepts représentant les différents algorithmes et les différentes techniques utilisés par les auteurs ainsi que toutes les notions décrivant ces algorithmes sont annotés par la zone *contribution* et les concepts représentant le type du document (*Texte*, *Web*) et le corpus utilisé sont annotés par la zone *domaine d'application*. Nous avons comparé notre approche à deux approches existantes dans la littérature.

- La première approche est basée sur une représentation vectorielle du contenu des documents appelée communément *sac-de-mots* car elle suppose l'indépendance des mots composant le document. Pour représenter les résumés scientifiques par des vecteurs, nous avons extrait les termes des résumés scientifiques par le même procédé que nous avons défini pour le processus de classification (cf. chapitre 3). Le vecteur résumé contient les lemmes correspondants aux noms, verbes et adjectifs extraits du texte. Les lemmes sont représentés par leur poids calculé sur la base de *tf-idf*. La similarité entre deux résumés est calculée en mesurant le cosinus de l'angle formé par leur vecteur respectif.

- La deuxième approche que nous avons utilisée est l'approche *n-grammes* qui représente le contenu textuel d'un résumé par un ensemble de mots appelés n-grammes. Le texte est divisé en un ensemble de n-grammes. La taille d'un n-gramme est déterminée par un nombre choisi de caractères consécutifs *n.* Nous avons testé plusieurs valeurs de *n* (n=2, 4 et 8) et pour chaque valeur de *n*, nous avons calculé la similarité entre deux résumés en utilisant l'équation (4.5) **[Basile et al., 2008][Basile et al., 2009]** et l'équation (4.6) **[Stein et al., 2005]**. Pour chaque paire de deux résumés *x* et *y*, la similarité *Sim(x,y)* est calculée comme suit :

$$Sim(x,y) = \frac{1}{|Dn(x)| + |Dn(y)|} \times \sum_{w \in Dn(x) \cup Dn(y)} \frac{(f_y(w) - f_x(w))^2}{(f_y(w) + f_x(w))^2} \qquad (4.5)$$

$$Sim(x,y) = \frac{|x \cap y|}{|x \cup y|} \qquad (4.6)$$

w représente un n-gramme donné, *fx(w)* dénote la fréquence relative de *w* dans le résumé *x* et *Dn(x)* représente le dictionnaire de n-grammes de *x*.

Nous avons retenu les meilleurs résultats obtenus avec n=8 et l'équation (4.6) pour lesquels nous avons observé le moins d'erreurs de rapprochement entre résumés.

4.4.2 Résultats

Nous avons testé plusieurs valeurs pour les différents paramètres utilisés. Notre objectif est de donner plus d'importance aux zones contexte et contribution parce que nous cherchons des rapprochements indiquant d'abord des documents traitant le même contexte avec des contributions similaires. Nous avons retenu les valeurs suivantes : $\alpha = 0.35$, $\beta = 0.63$, $\gamma = 0.02$, g =0,05). Ces valeurs ont permis de regrouper les résumés en fonction de leur contexte. Nous donnons en exemple, dans la Table 4.5 et la Table 4.6, les résultats obtenus respectivement lors de la comparaison des résumés *A1* et *A12* avec les autres résumés de notre collection.

Table 4.5 et Table 4.6 montrent les trois similarités partielles calculées pour chaque paire de résumés comparés ainsi que leur similarité globale. Les résultats, triés par ordre décroissant de la similarité globale, montrent un regroupement des résumés par contexte. Le résumé *A1* traite le contexte de la classification non supervisée (clustering). Les résumés qui ont la plus grande similarité avec *A1* correspondent à ce contexte. De même, le résumé *A12* traite le contexte de la classification supervisée des documents (classification). Les résumés qui ont la plus grande similarité avec *A12* correspondent également à ce contexte.

Texte1	Texte2	Similarités			
		Contexte	Contribution	Domaine application	Globale
A1.clustering	A3.clustering	1,000	0,401	1,000	0,622
A1.clustering	A10.clustering	1,000	0,295	0,157	0,539
A1.clustering	A2.clustering	0,982	0,306	0,065	0,538
A1.clustering	A9.clustering	1,000	0,227	0,153	0,496
A1.clustering	A16.clustering	1,000	0,169	0,065	0,458
A1.clustering	A15.clustering	1,000	0,103	0,237	0,419
A1.clustering	A17.clustering	1,000	0,092	0,345	0,415
A1.clustering	A5.clustering	1,000	0,095	0,237	0,414
A1.clustering	A18.clustering	1,000	0,022	0,353	0,371
A1.clustering	A19. classif-clust	0,558	0,016	0,065	0,207
A1.clustering	A14.classification	0,244	0,125	0,431	0,172
A1.clustering	A6.classification	0,240	0,074	0,016	0,131
A1.clustering	A8.classification	0,225	0,060	0,541	0,127
A1.clustering	A7.classification	0,225	0,060	0,065	0,118
A1.clustering	A4.classification	0,244	0,036	0,065	0,109
A1.clustering	A11.classification	0,237	0,034	0,108	0,107
A1.clustering	A13.classification	0,230	0,014	0,065	0,090
A1.clustering	A12.classification	0,231	0,007	0,125	0,088
A1.clustering	A20.classification	0,237	0,005	0,031	0,087

Table 4.5 Similarités entre A1 et les autres résumés.

Texte1	Texte2	Similarités			
		Contexte	Contribution	Domaine application	Globale
A12.classification	A13.classification	1,000	0,015	0,000	0,360
A12.classification	A4.classification	0,966	0,032	0,000	0,358
A12.classification	A20.classification	0,964	0,012	0,483	0,355
A12.classification	A6.classification	0,965	0,015	0,193	0,351
A12.classification	A14.classification	0,966	0,012	0,066	0,347
A12.classification	A11.classification	0,964	0,007	0,023	0,342
A12.classification	A8.classification	0,900	0,005	0,185	0,322
A12.classification	A7.classification	0,900	0,005	0,000	0,318
A12.classification	A19.classif-clust	0,541	0,107	0,000	0,257
A12.classification	A5.clustering	0,234	0,027	0,329	0,105
A12.classification	A3.clustering	0,234	0,032	0,125	0,105
A12.classification	A18.clustering	0,234	0,026	0,125	0,101
A12.classification	A17.clustering	0,234	0,024	0,123	0,100
A12.classification	A9.clustering	0,227	0,019	0,189	0,095
A12.classification	A15.clustering	0,231	0,004	0,329	0,090
A12.classification	A1.clustering	0,231	0,007	0,125	0,088
A12.classification	A2.clustering	0,233	0,005	0,000	0,085
A12.classification	A16.clustering	0,231	0,006	0,000	0,085
A12.classification	A10.clustering	0,227	0,004	0,032	0,083

Table 4.6 Similarités entre A12 et les autres résumés.

Dans la Table 4.5, nous pouvons comparer la similarité de *A1* avec les autres résumés à trois niveaux. Nous pouvons comparer leur similarité au niveau contexte, au niveau *contribution* et au niveau *domaine d'application*. Les valeurs obtenues pour les résumés *A1* et *A3* indiquent que ces deux résumés traitent du même contexte (sim contexte =1), présentent des contributions proches (Sim contribution= 0.401) et appliquent leur approche au même domaine (sim domaine d'application =1). La valeur de leur similarité globale est très élevée. Ces valeurs permettent de retenir ces deux résumés comme étant des documents suspects et nécessitent une lecture et analyse de leur contenu dans sa globalité.

Dans la Table 4.6, nous pouvons comparer la similarité de *A12* avec les autres résumés à trois niveaux. Pour les dix dernières lignes de la Table 4.6, nous avons de très faibles similarités partielles et globales. La Table 4.6 montre également que les résumés correspondant aux huit premières lignes traitent le même contexte que *A12* (sim contexte ≥ 0.900) mais utilisent des approches différentes (sim contribution ≤ 0.032). Leur similarité globale est faible (≤ 0.360). Cela nous permet de conclure que le résumé *A12* ne présente aucun risque de plagiat avec les autres résumés du corpus.

L'objectif de notre approche est d'être capable de retrouver les documents suspects, c'est à dire des documents avec des similarités élevées. Pour retrouver ces documents, un seuil pour les valeurs de similarités calculées est déterminé par expérimentation.

Pour comparer les résultats obtenus avec notre approche et les approches *sac-de-mots* (*Bag-of-words*) et *n-grammes*, des similarités entre les différents résumés de notre collection ont été calculées utilisant ces deux approches. Les résumés ont été ensuite triés par ordre décroissant de leur similarité. Pour ces deux approches, plusieurs erreurs de rapprochement entre résumés ont été constatées. La Table 4.7 donne un exemple de comparaison des similarités entre *A4* et les autres résumés obtenues avec notre approche et les approches *sac-de-mots* et *n-grammes*. A4 traite du contexte classification. Avec les approches sac-de-mots et n-grammes, la majorité des résumés sémantiquement proches de *A4* traite du contexte clustering.

- Pour l'approche *sac-de-mots*, les résumés appartenant au contexte clustering (*A10, A3, A2, A5, A15, A1*) obtiennent un score de similarité meilleur que celui des résumés (*A11, A8, A12, A20, A7, A14*) qui traitent le même contexte que *A4*. Il en est de même pour l'approche *n-grammes*. Les résumés appartenant au contexte clustering (*A18, A3, A10, A1*) obtiennent un score de similarité meilleur que celui des résumés (*A7, A13, A14, A20*) qui traitent le même contexte que *A4*.

- Pour toutes les comparaisons effectuées entre les résumés du corpus, notre approche est capable de regrouper les résumés par contexte comme le montrent Table 4.5, Table 4.6, Table 4.7 et Table 4.8. Le clustering et la classification (classification supervisée) sont deux contextes différents. Pour cette raison, les similarités entre deux résumés appartenant à ces deux contextes doivent être faibles (similarité de contexte faible et similarité de contribution faible) et par conséquent, le risque de plagiat doit être très faible ou inexistant.

Texte1	Texte2	Notre Approche	Sac-de-mots		N-gramme	
A4.classification	A6.classification	0.417272	A06.classification	0.125685	A11.classification	0,042080
A 4.classification	A11.classification	0.401363	**A10.clustering**	0.108323	**A18.clustering**	0,038287
A 4.classification	A13.classification	0.373563	A13.classification	0.097182	**A03.clustering**	0,036313
A 4.classification	A12.classification	0.358287	A19.classif-clust	0.095763	A06.classification	0,035757
A 4.classification	A14.classification	0.358132	**A03.clustering**	0.092988	**A10.clustering**	0,035634
A 4.classification	A7.classification	0.353878	**A02.clustering**	0.092751	A08.classification	0,035602
A 4.classification	A20.classification	0.353120	**A05.clustering**	0.089178	A12.classification	0,034261
A 4.classification	A8.classification	0.330633	**A15.clustering**	0.073636	**A01.clustering**	0,033475
A 4.classification	A19.classif-clust	0.257688	**A01.clustering**	0.066826	A19.classif-clust	0,033400
A 4.classification	A5.clustering	0.191517	**A11.classification**	0.061259	**A07.classification**	0,033071
A 4.classification	A3.clustering	0.180843	**A08.classification**	0.045829	A17.clustering	0,032417
A 4.classification	A9.clustering	0.176679	A18.clustering	0.043951	A09.clustering	0,029097
R4.classification	A2.clustering	0.175801	**A12.classification**	0.042752	A15.clustering	0,026786
R4.classification	A15.clustering	0.147094	A16.clustering	0.041947	A05.clustering	0,025901
A4.classification	A10.clustering	0.135412	**A20.classififcation**	0.033817	**A13.classification**	0,025269
A4.classification	A18.clustering	0.129238	**A07.classification**	0.031982	**A14.classification**	0,023015
A4.classification	A17.clustering	0.119075	A17.clustering	0.028876	A02.clustering	0,020426
A4.classification	A16.clustering	0.114507	**A14.classification**	0.026670	A16.clustering	0,018511
A4.classification	A1.clustering	0.109055	A09.clustering	0.023351	**A20.classification**	0,015968

Table 4.7 Similarités entre A4 et les autres résumés calculées par notre approche, sac-de-mots, et N-gramme.

Pour déterminer quelle approche réalise des rapprochements corrects entre les résumés du corpus, la précision P5 et la R-précision pour chaque approche et pour chaque résumé ont été calculées.

Nous admettons qu'un résumé *A1* est pertinent pour un résumé *A2*, si *A1* traite du même contexte que *A2*. La précision Px au point x (x=5, R) représente le ratio de résumés pertinents parmi les x premiers résumés retournés par le processus. R dans la R-précision représente le nombre de résumés pertinents pour un résumé donné dans le corpus. La Table 4.8 résume les différentes valeurs.

Résumés	P5			R-precision		
	Sac-de-mots	N-gramme	Notre approche	Sac-de-mots	N-gramme	Notre approche
A1	1,000	1,000	1,000	0,800	1,000	1,000
A2	0,800	1,000	1,000	0,800	1,000	1,000
A3	0,800	1,000	1,000	0,800	0,900	1,000
A4	0,600	0,400	1,000	0,333	0,556	1,000
A5	0,800	0,600	1,000	0,900	0,800	1,000
A6	1,000	1,000	1,000	0,667	0,778	1,000
A7	0,800	0,800	1,000	0,778	0,667	1,000
A8	0,800	0,800	1,000	0,778	0,556	1,000
A9	0,800	1,000	1,000	0,900	0,900	1,000
A10	0,800	1,000	1,000	0,800	0,900	1,000
A11	1,000	1,000	1,000	0,778	0,889	1,000
A12	0,800	0,800	1,000	0,778	0,667	1,000
A13	0,800	0,800	1,000	0,667	0,667	1,000
A14	1,000	1,000	1,000	0,778	0,667	1,000
A15	0,800	1,000	1,000	0,800	1,000	1,000
A16	1,000	1,000	1,000	0,800	0,900	1,000
A17	0,600	1,000	1,000	0,700	0,900	1,000
A18	0,800	1,000	1,000	0,600	0,800	1,000
A19	1,000	1,000	1,000	1,000	1,000	1,000
A20	0,800	0,800	1,000	0,778	0,667	1,000
Average	**0,840**	**0,900**	**1,000**	**0,762**	**0,811**	**1,000**

Table 4.8 Valeurs de précision pour notre approche et les approches sac-de-mots et n-gramme.

Notre processus obtient de meilleurs résultats que les approches *sac-de-mots* et *n-grammes*. Notre processus est capable de faire des rapprochements corrects entre les résumés traitant du même contexte. Il est par conséquent plus précis que les autres approches.

Nous avons utilisé le test des rangs signés de Wilcoxon dans le but d'étudier la signification statistique de l'amélioration des valeurs de précision apportée par notre processus. Nous avons calculé les p-values entre notre système et les autres approches et les résultats sont résumés dans la Table 4.9.

Les p-values obtenues avec le test Wilcoxon sont inférieures à 0.01. Ce sont des p-values très significatives qui nous permettent de conclure que notre processus est capable de regrouper les résumés par contexte plus correctement que les approches *sac-de-mots* et *n-grammes*. D'autres résultats sont résumés dans la Table 4.10.

	Notre approche / sac-de-mots	Notre approche / n-gramme
P-value à P5	0.000213431	0.0089409
P-value à R-precision	0.0000638361	0.000219794

Table 4.9 résultats du test de Wilcoxon.

Text1	Text2	Notre approche				Sac-de-mots	N-gramme
		contexte	contribution	domaine d'application	globale		
A1.clustering	A3.clustering	1.000000	0.400673	1.000000	0.622424	0.724688	0,352187
A2.clustering	A10.clustering	0.982456	0.486622	0.112994	0.652692	0.198869	0,050761
A15.clustering	A16.clustering	1.000000	0.188889	0.000000	0.469000	0.470623	0,108580

Table 4.10 Comparaison entre notre approche et les approches sac-de-mots et n-gramme.

- Le contenu des résumés *A1*, *A2*, *A3* et *A10* indique une grande similarité entre les résumés (*A1-A3*) et les résumés (*A2-A10*). Ces deux paires de résumés traitent du même contexte, utilisent les mêmes algorithmes et exploitent des ontologies pour résoudre des problématiques similaires a priori. Comme le montre la Table 4.10, notre approche permet de sélectionner ces résumés comme étant suspects, alors que les approches sac-de-mots et n-grammes sélectionnent uniquement les résumés (*A1-A3*). *A1* et *A3* utilisent globalement les mêmes mots dans leur contenu Alors que pour les résumés *A2* et *A10*, leur contenu est décrit avec différents mots et différentes phrases, mais les deux résumés se focalisent sur la sélection des caractéristiques pour représenter les clusters en exploitant une ontologie et utilisent le même algorithme de classification. Notre approche est capable de capturer le sens des résumés et retient donc ces deux résumés pour un examen complet de leurs articles correspondants.

- L'approche *sac-de-mots* indique un rapprochement entre les résumés *A15* et *A16*. Ces deux résumés ont une similarité élevée alors que les auteurs de ces deux résumés utilisent différentes méthodes dans leurs contributions. Notre approche a l'avantage de comparer les résumés à trois niveaux. Avec notre processus, la similarité au niveau contribution entre *A15* et *A16* indique une valeur très faible, ce qui signifie que les méthodes utilisées par les auteurs pour résoudre leur problématique sont différentes. Nous pouvons alors conclure que même si ces deux résumés présentent des contextes similaires, le risque de plagiat est faible.

4.5 Conclusion

Dans ce chapitre, nous avons décrit notre approche permettant de calculer la similarité des documents. Nous nous intéressons particulièrement aux résumés des articles scientifiques. Le calcul de la similarité des textes se fait sur la base de la comparaison des résumés rattachés à une même ontologie de domaine. Cette similarité est basée sur les graphes correspondant au contenu des textes.

Notre approche met en œuvre un processus d'enrichissement des graphes à travers la construction de périmètres sémantiques des textes et par la comparaison de leurs graphes. Une similarité non explicitement exprimée dans les textes est déduite.

Nous avons proposé de décomposer le texte des résumés scientifiques en trois zones : *contexte*, *contribution* et *domaine d'application*. Le processus calcule alors des similarités partielles permettant de faire ressortir les notions communes à deux résumés. Les résumés

sont regroupés en fonction de leur contexte et triés par ordre décroissant de leur similarité globale.

L'objectif de notre approche est de retrouver des résumés suspects. Elle présente l'avantage de comparer le contenu des résumés à trois niveaux. L'examen de chaque similarité partielle obtenue nous permet de conclure sur l'existence d'un risque de plagiat.

Les résultats obtenus par les expérimentations menées sur un ensemble de résumés d'articles scientifiques montrent que notre processus est plus performant que les processus conventionnels et souligne les avantages apportés par l'enrichissement des graphes et par la décomposition des résumés en parties distinctes.

Dans le dernier chapitre, nous allons expliciter les différentes applications que nous avons développées pour mettre en œuvre les différentes étapes de notre approche.

Chapitre 5
Architecture des processus

5.1 Introduction

Les chapitres précédents ont permis d'expliquer la conception des processus composant notre approche. Pour expérimenter notre approche et pour la comparer aux approches existantes, nous avons développé cinq applications. Les différentes étapes de réalisation ainsi que les outils utilisés sont présentés dans ce qui suit.

5.2 Mise en œuvre des différents processus

Notre implémentation inclut cinq applications. Deux d'entre elles correspondent à notre approche qui prend en charge la classification des documents (section 5.2.1) et le calcul de la similarité des textes (section 5.2.3) et les trois autres correspondent à la prise en charge des approches conventionnelles (sections 5.2.2, 5.2.4 et 5.2.5). L'objectif étant d'évaluer les approches individuellement et de comparer ensuite leurs résultats. Ces applications sont implémentées en langage Java (version 7) en utilisant des API telles que Stanford Pos Tagging (version 3.7.0), Weka **[Hall et al. 2009]** (version 3.6.0), *JDOM*[1] (version 2.0.4) et Rita **[Howe, 2009]** (version 1.0.64) qui inclue un ensemble de bibliothèques dont JWNL. Les ressources externes utilisées sont WordNet **[Miller et al., 1995]** (version 2.0), WordNet Domains **[Magnini et al, 2000]** (version 3.2) et une ontologie du domaine de la *classification des document*s que nous avons construite. Les applications comportent un ensemble de modules permettant d'effectuer des traitements spécifiques. Le contenu de chaque document des deux collections que nous avons utilisées est placé dans un fichier texte. L'ensemble des documents est placé dans un dossier. Les différentes applications et leur architecture sont données ci-dessous :

1. Classification sémantique des documents (CBO).
2. Classification en utilisant les classifieurs conventionnels.
3. Similarité sémantique des documents.
4. Sac-de-mots.
5. N-grammes.

[1] http://www.jdom.org/

5.2.1 Classification sémantique des documents (CBO)

Cette application implémente la classification des documents que nous avons définie (CBO). Les différents modules de l'application sont résumés dans la Figure 5.4. A l'entrée de l'application, nous avons :

- Le fichier texte *listedomaines.txt* contenant les différents domaines définis dans WordNet Domains. Nous avons construit ce fichier en parcourant le fichier wn-domains-3.2.txt. La Figure 5.1 donne un extrait des fichiers *listedomaines.txt.*

- Le fichier texte *listOntologies.txt* qui contient les domaines choisis par l'utilisateur pour classifier les documents. La Figure 5.2 montre un exemple du fichier *listOntologies.txt.*

- La collection de documents à classifier. Les documents de la collection sont choisis par l'utilisateur et placés dans un dossier que l'application va exploiter. Un extrait de la collection utilisée par cette application est donné par la Figure 5.3.

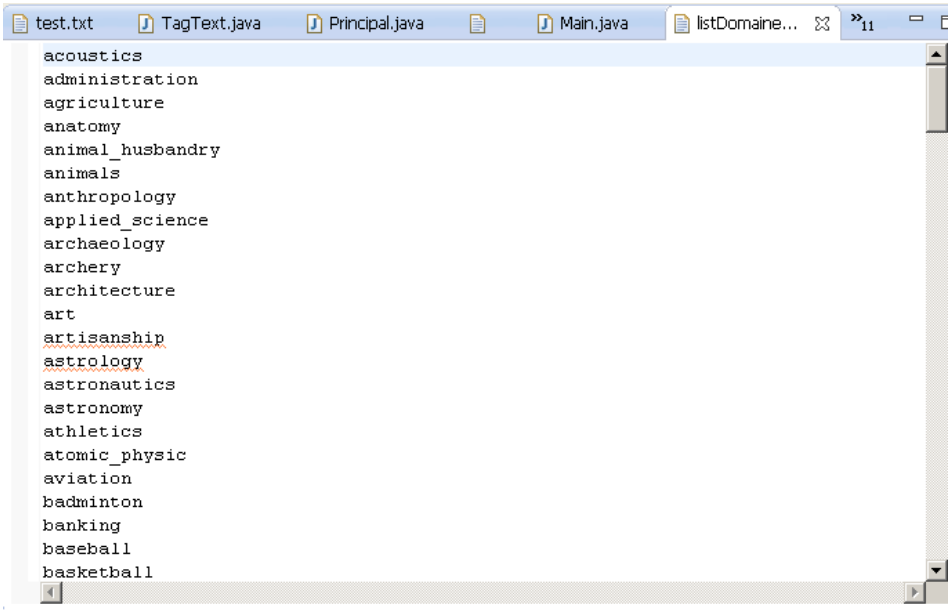

Figure 5.1 Extrait de *listedomaines.txt*.

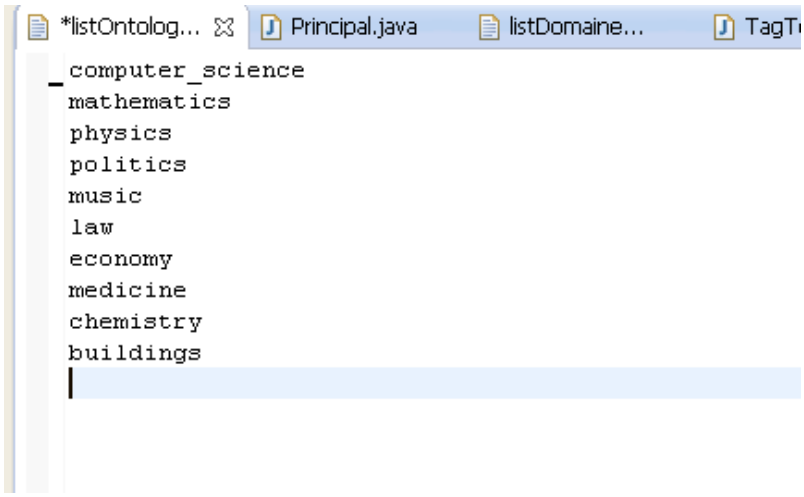

Figure 5.2 Exemple du contenu de listOntologies.txt.

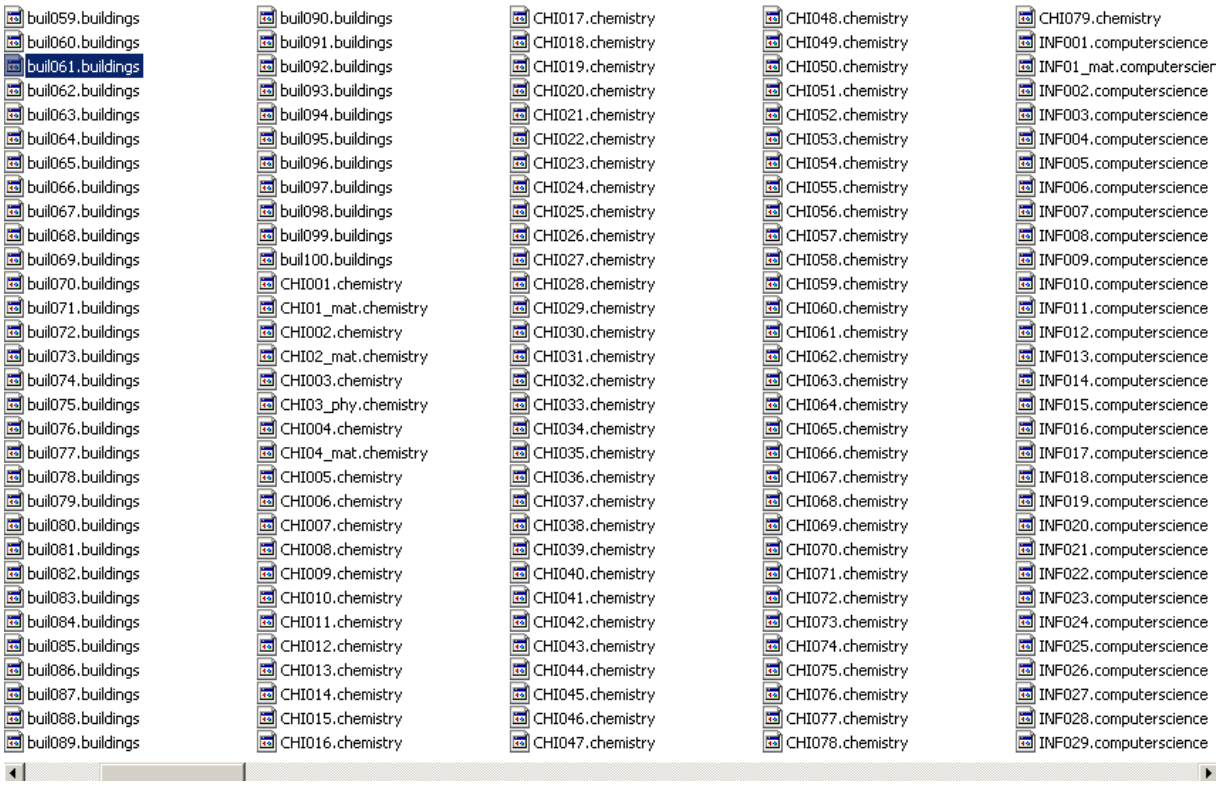

Figure 5.3 Extrait de la collection de documents utilisée par le processus CBO.

Le déroulement du programme s'effectue de la façon suivante :

Division du texte en paragraphes, phrases, mots.

Le document est divisé en paragraphes, les paragraphes en phrases et les phrases en mots. La position des paragraphes dans le document, la position des phrases dans les paragraphes et la position des mots dans les phrases sont mémorisées.

Annotation des mots par leur type.

Les mots des phrases sont annotés par leur type (nom, verbe, adjectif, etc.) en utilisant l'API Stanford Pos Tagging.

Projection sur WordNet : extraction des termes et des synsets.

Les mots annotés de chaque phrase sont ensuite projetées sur WordNet pour extraire les termes et les synsets correspondant à ces termes. L'API JWNL est utilisée. Sachant que WordNet est divisé en plusieurs fichiers regroupant les synsets par type : (data.noun, data.verb, data.adj et data.adv), seuls les noms sont pris en considération.

Projection sur WodNet Domains : selection des synsets pour les domaines séléctionnés.

Les synsets extraits à l'étape précédente sont filtrés pour ne retenir que les synsets relatifs aux domaines retenus pour la classification. Pour cela le fichier WordNet Domains est utilisé. Ce fichier présente dans chaque ligne, un numéro de synset et les domaines où ce synset possède un sens.

Désambiguïsation locale.

- Une désambiguïsation locale est réalisée au niveau de chaque domaine. Elle permet de choisir parmi les synsets correspondant à un terme donné de la phrase, le synset le plus approprié. La distance entre synsets est calculée en utilisant la mesure *Rita*.

Classification.

La classification est réalisée en dernière étape. Elle calcule un score pour chaque domaine et pour chaque document de la collection. Le résultat est ensuite donné dans un fichier résultat qui résume pour chaque document, le score obtenu par chaque domaine de la liste retenue.

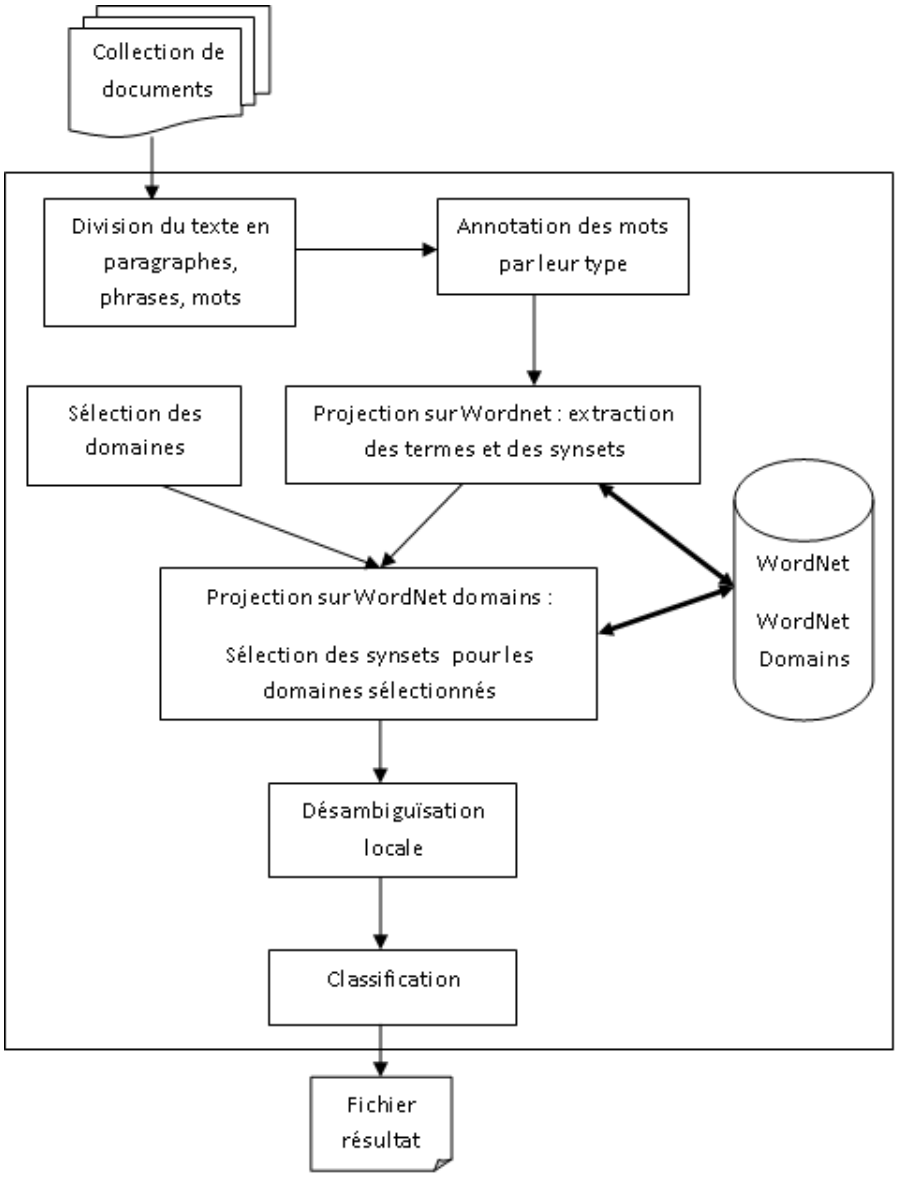

Figure 5.4 Classification des documents avec CBO.

5.2.2 Classification en utilisant les classifieurs conventionnels

Nous avons développé cette application pour représenter les documents par des vecteurs de termes qui seront utilisés par les classifieurs conventionnels (SMO, Naïve bayes et J48) définis dans Weka. L'architecture de l'application est donnée dans la Figure 5.5. Cette application est composée des étapes suivantes :

- Le texte des documents est divisé en phrases puis en mots.

- Les mots sont annotés par leur type.

Ces deux premières étapes sont identiques à celles utilisées dans l'application classification sémantique (cf. section 5.2.1).

- Pour l'indexation des documents, les lemmes des noms, des verbes et des adjectifs extraits des documents sont retenus. De ce fait, les mots vides ne sont pas pris en considération et sont éliminés automatiquement. Les poids des mots basé sur *tf-ifd* sont calculés. Les documents sont représentés par des vecteurs dont les dimensions sont les poids des lemmes des mots retenus.

- Ensuite, les documents indexés à l'étape précédente sont présentés avec le format *Arff* de Weka pour être utilisés par les classifieurs conventionnels. L'API Weka est utilisée. Le fichier résultat constitue l'entrée des différents classifieurs conventionnels.

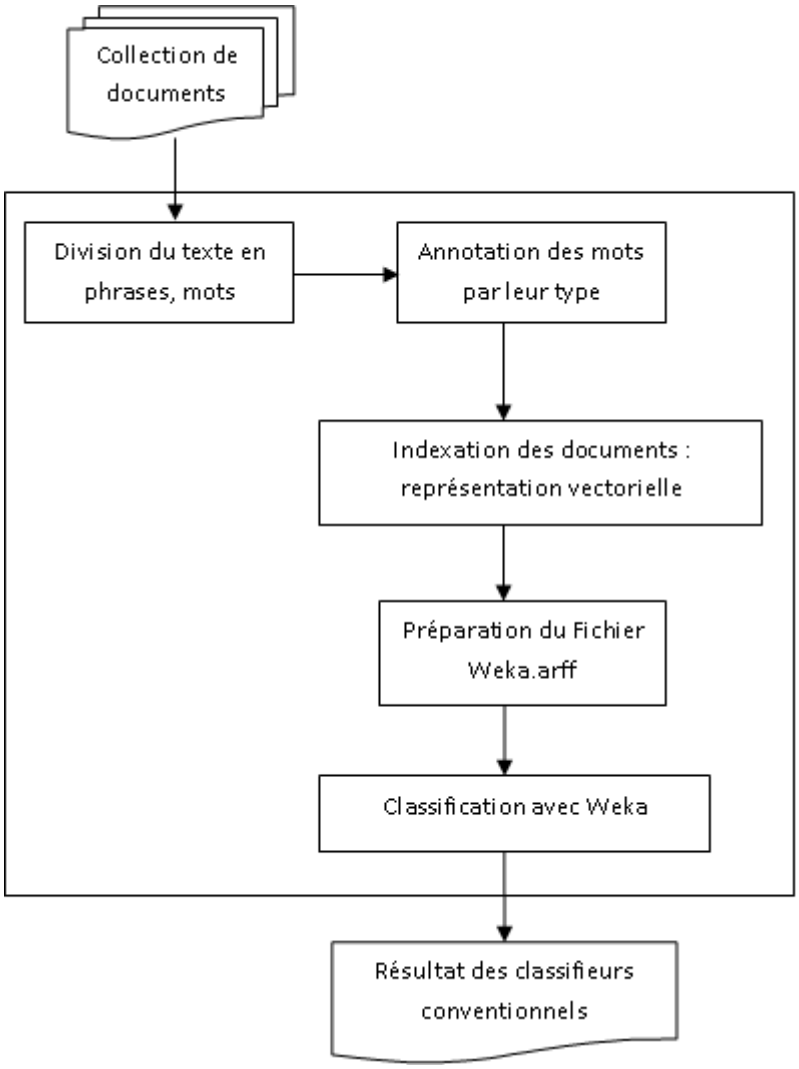

Figure 5.5 Classification avec les classifieurs conventionnels.

Dans Weka, l'algorithme correspondant au classifieur SVM est implémenté sous le nom de SMO **[Platt, 1998][Keerthi et al., 2001][Hastie et al., 1998]**, le classifieur arbre de décision est implémenté sous le nom de J48 **[Quinlan 1993]** et le classifieur probabiliste sous le nom de Naive bayes **[John et al., 1995]**.

5.2.3 Similarité sémantique des documents

Cette application met en œuvre notre processus de calcul de la similarité des résumés scientifique. Nous avons construit un corpus contenant les résumés des articles scientifiques traitant de la classification des documents (supervisée et non supervisée). Ce corpus est extrait du Web. Nous avons également construit une ontologie relative au domaine de la classification des documents. Cette ontologie a une profondeur égale à 6 et contient 109 concepts reliés entre eux par des relations *is-a* et des relations transversales. Chaque concept de l'ontologie est annoté avec les zones que nous avons définies. Les différents modules de cette application sont résumés dans la Figure 5.11.

5.2.3.1 Représentation des informations relative à l'ontologie "classification des documents"

L'application Similarité sémantique des documents exploite les algorithmes définissant les processus de projection et de désambiguïsation utilisés dans l'application CBO (cf. section 5.2.1). Pour cela, nous nous somme inspirés de WordNet pour présenter les informations relatives à notre ontologie. Rappelons que les informations relatives à WordNet sont représentées dans des fichiers textes. WordNet crée un fichier texte par type de synset : data.noun pour représenter les synsets de type nom et leurs relations, data.verb pour représenter les synsets de type verbe et leurs relations etc. WordNet possède également des fichiers index pour chaque type de synset pour représenter les synsets et leurs numéros (index.noun, index.verb etc.).

Les fichiers textes créés sont comme suit :

- Un fichier texte *data.noun* pour représenter les concepts et les relations *is-a* reliant les concepts. Chaque ligne du fichier correspond à un concept. Un concept est représenté par un numéro, un label, le nombre de ses synonymes, ses synonymes, son concept parent (représenté par le symbole @), ses concepts fils (représentés par le symbole ~) et un glossaire. La Figure 5.6 donne un extrait de ce fichier.

```
00001949 03 n 02 automatic_classification 0 automatic_categorization 0 004 @ 00001740 n 0000 ~ 00002596 n 0000 ~ 00002130 n 0000 ~ 00002308 n 0000
00002130 03 n 03 unsupervized_classification 0 unsupervized_categorization 0 clustering 0 003 @ 00001949 n 0000 ~ 00002722 n 0000 ~ 00002856 n 0000
00002308 03 n 05 supervized_classification 0 supervized_categorization 0 supervized_machine_learning 0 classification 0 categorization 0 005 @ 0000
00002596 03 n 01 dataless_classification 0 001 @ 00001949 n 0000 |
00002722 03 n 02 hierarchical_document_clustering 0 hierarchical_clustering 0 001 @ 00002130 n 0000 |
00002856 03 n 01 standard_partitional_algorithm 0 003 @ 00002130 n 0000 ~ 00002998 n 0000 ~ 00003221 n 0000 |
00002998 03 n 06 Bi_section_kmeans 0 bi_section_k_means 0 bisecting_kmeans 0 bisecting_k_means 0 bi_k_means 0 bi_kmeans 0 001 @ 00002856 n 0000 | a
00003221 03 n 02 Kmeans 0 k_means 0 001 @ 00002856 n 0000 |
00003336 03 n 02 hierarchical_classification 0 hierarchical 0 001 @ 00002308 n 0000 |
00003477 03 n 03 multi_label_classification 0 multi_label 0 mlc 0 003 @ 00002308 n 0000 ~ 00003739 n 0000 ~ 00003850 n 0000 |
00003635 03 n 03 label_ranking 0 ranking 0 lr 0 001 @ 00002308 n 0000 |
00003739 03 n 02 bp_mll 0 multi_label_neural_network 0 001 @ 00003477 n 0000 |
00003850 03 n 02 ml_knn 0 multi_label_knn 0 001 @ 00003477 n 0000 |
00003950 03 n 01 single_class 0 004 @ 00002308 n 0000 ~ 00004092 n 0000 ~ 00004214 n 0000 ~ 00004492 n 0000 |
00004092 03 n 01 semantic_classification 0 001 @ 00003950 n 0000 |
00004214 03 n 02 statistical_classification 0 classification_algorithm 0 008 @ 00003950 n 0000 ~ 00004633 n 0000 ~ 00004750 n 0000 ~ 00004856 n 000
00004492 03 n 01 syntactic_classification 0 002 @ 00003950 n 0000 ~ 00005445 n 0000 |
00004633 03 n 02 dt 0 decision_tree 0 001 @ 00004214 n 0000 |
00004750 03 n 01 rocchio 0 001 @ 00004214 n 0000 |
```

Figure 5.6 Extrait du fichier data.noun

- Un fichier texte *index.noun* contenant les labels des concepts avec leurs numéros. Un extrait de ce fichier est donné par la Figure 5.7.

```
automatic_classification n 1 00001949
automatic_categorization n 1 00001949
unsupervized_classification n 1 00002130
unsupervized_categorization n 1 00002130
clustering n 1 00002130
supervized_classification n 1 00002308
supervized_categorization n 1 00002308
supervized_machine_learning n 1 00002308
classification n 1 00002308
categorization n 1 00002308
dataless_classification n 1 00002596
hierarchical_document_clustering n 1 00002722
hierarchical_clustering n 1 00002722
standard_partitional_algorithm n 1 00002856
Bi_section_kmeans n 1 00002998
bi_section_k_means n 1 00002998
bisecting_kmeans n 1 00002998
bisecting_k_means n 1 00002998
bi_kmeans n 1 00002998
bi_k_means n 1 00002998
k_means n 1 00003221
kmeans n 1 00003221
hierarchical_classification n 1 00003336
hierarchical n 1 00003336
multi_label_classification n 1 00003477
multi_label n 1 00003477
mlc n 1 00003477
```

Figure 5.7 Extrait du fichier *index.noun*

- Un fichier texte *transversale.noun* contenant les relations transversales reliant les concepts. Un extrait de ce fichier est donné dans la Figure 5.8.

```
00012487 00012804 00013233 00015030
weight feature concept external_resource
00012487 00012804 00014155
weight feature class
00015030 00013233 00012804 00014155
external_resource concept feature class
00007998 00009978 00010861
corpus document document_type
00007998 00009978 00011501 00012804 00013233 00015030
corpus document representation feature concept external_resource
00007998 00009978 00011501 00012804 00012487
corpus document representation feature weight
00007998 00009978 00011501 00012804 00014155
corpus document representation feature class
00010861 00009978 00011501 00012804 00013233 00015030
document_type document representation feature concept external_resource
00010861 00009978 00011501 00012804 00012487
document_type document representation feature weight
00010861 00009978 00011501 00012804 00014155
document_type document representation feature class
00001949 00009978 00011501 00012804 00013233 00015030
automatic_classification document representation feature concept external_resource
00001949 00009978 00011501 00012804 00012487
automatic_classification document representation feature weight
00001949 00009978 00011501 00012804 00014155
automatic_classification document representation feature class
00001949 00009978 00007998
```

Figure 5.8 Extrait de *transversale.noun*

- Un fichier texte *fichzone.noun* pour annoter chaque concept avec la zone appropriée. Un extrait de ce fichier est donné dans la Figure 5.9.

```
00001949 n automatic_classification 0 automatic_categorization z contexte
00002130 n unsupervized_classification 0 unsupervized_categorization 0 clustering z contexte
00002308 n supervized_classification 0 supervized_categorization 0 classification 0 supervized_machine_learning 0 categorization z contexte
00002596 n dataless_classification z contexte
00002722 n hierarchical_document_clustering 0 hierarchical_clustering z contribution
00002856 n standard_partitional_algorithm z contribution
00002998 n Bi_section_kmeans 0 bi_section_k_means 0 bisecting_kmeans 0 bisecting_k_means 0 bi_k_means 0 bi_kmeans z contribution
00003221 n Kmeans 0 k_means z contribution
00003336 n hierarchical_classification 0 hierarchical z contribution
00003477 n multi_label_classification 0 multi_label 0 mlc z contribution
00003635 n label_ranking 0 ranking 0 lr z contribution
00003739 n bp_mll 0 multi_label_neural_network z contribution
00003850 n ml_knn 0 multi_label_knn z contribution
00003950 n single_class z contribution
00004092 n semantic_classification z contribution
00004214 n statistical_classification 0 classification_algorithm z contribution
00004492 n syntactic_classification z contribution
00004633 n dt 0 decision_tree z contribution
00004750 n rocchio z contribution
00004856 n vote z contribution
00004959 n hyperlink_vector_voting z contribution
00005081 n svm 0 support_vector_machine z contribution
00005208 n knn 0 k_nearest_neighbor z contribution
00005331 n naive_baye 0 nb z contribution
00005445 n coefficient_des_deux_ecarts z contribution
```

Figure 5.9 Extrait de *fichzone.noun*

- Un fichier *domaine.txt* jouant le rôle du fichier WordNet Domains.

- Un fichier Xml *arbre.xml* définissant l'arborescence de notre ontologie. Cet arbre est utilisé pour définir les numéros des branches et des sous-branches (cf. chapitre 4, Figure 4.1).

5.2.3.2 Différentes modules de l'application

Division du texte en paragraphes, phrases, mots.

Le document est divisé en paragraphes, les paragraphes en phrases et les phrases en mots. La position des paragraphes dans le document, la position des phrases dans les paragraphes et la position des mots dans les phrases sont mémorisées.

Annotation des mots par leur type.

Les mots des phrases sont annotés par leur type (nom, verbe, adjectif, etc.) en utilisant l'API Stanford Pos Tagging.

Projection sur l'ontologie : extraction des termes et des concepts.

Les mots annotés de chaque phrase sont ensuite projetées sur l'ontologie pour extraire les termes et les concepts correspondant à ces termes. L'API JWNL est utilisée ainsi que le fichier *index.noun*.

Désambiguïsation locale.

Une désambiguïsation locale est réalisée. Elle permet de choisir parmi les concepts correspondant à un terme donné de la phrase, le concept le plus approprié. La distance entre concepts est calculée en utilisant la mesure *Rita*.

Construction du périmètre sémantique.

Le périmètre sémantique est construit pour les documents à comparer. A cet effet, des concepts sont extraits de l'ontologie pour l'enrichissement des graphes en exploitant les relations définis dans l'ontologie. Pour retrouver le chemin reliant deux concepts, la bibliothèque Rita est utilisée ainsi que le fichier *transversale.noun*.

Comparaison des graphes : enrichissement.

Les graphes des documents sont comparés deux à deux : Un enrichissement des graphes est réalisé en ajoutant et en calculant les poids des concepts représentant les graphes. La bibliothèque Rita est utilisée pour retrouver le parent commun à deux concepts ainsi que le concept parent d'un concept.

Construction de l'arbre XML.

Un arbre Xml est créé pour numéroter tous les concepts de l'ontologie en fonction des branches et des sous-branches auxquelles ces concepts appartiennent. Cette numérotation est exploitée dans l'équation qui calcule la similarité des graphes en déterminant le nombre de branches communes et le nombre de branches total composant les graphes à comparer (cf. chapitre 4, section 4.2.3.2). Le fichier Xml est créé avec l'API *JDOM2*. Un extrait de ce fichier est donné dans la Figure 5.10.

```
<noeud synset="00001949" nom="automatic_classification" branche="1.2" niveau="2">
  <noeud synset="00002596" nom="dataless_classification" branche="1.2.1" niveau="3" />
  <noeud synset="00002130" nom="unsupervized_classification" branche="1.2.2" niveau="3">
    <noeud synset="00002722" nom="hierarchical_document_clustering" branche="1.2.2.1" niveau="4" />
    <noeud synset="00002856" nom="standard_partitional_algorithm" branche="1.2.2.2" niveau="4">
      <noeud synset="00002998" nom="Bi_section_kmeans" branche="1.2.2.2.1" niveau="5" />
      <noeud synset="00003221" nom="Kmeans" branche="1.2.2.2.2" niveau="5" />
    </noeud>
  </noeud>
  <noeud synset="00002308" nom="supervized_classification" branche="1.2.3" niveau="3">
    <noeud synset="00003336" nom="hierarchical_classification" branche="1.2.3.1" niveau="4" />
    <noeud synset="00003477" nom="multi_label_classification" branche="1.2.3.2" niveau="4">
      <noeud synset="00003739" nom="bp_mll" branche="1.2.3.2.1" niveau="5" />
      <noeud synset="00003850" nom="ml_knn" branche="1.2.3.2.2" niveau="5" />
    </noeud>
    <noeud synset="00003635" nom="label_ranking" branche="1.2.3.3" niveau="4" />
    <noeud synset="00003950" nom="single_class" branche="1.2.3.4" niveau="4">
      <noeud synset="00004092" nom="semantic_classification" branche="1.2.3.4.1" niveau="5" />
      <noeud synset="00004214" nom="statistical_classification" branche="1.2.3.4.2" niveau="5">
        <noeud synset="00004633" nom="dt" branche="1.2.3.4.2.1" niveau="6" />
        <noeud synset="00004750" nom="rocchio" branche="1.2.3.4.2.2" niveau="6" />
        <noeud synset="00004856" nom="vote" branche="1.2.3.4.2.3" niveau="6" />
        <noeud synset="00004959" nom="hyperlink_vector_voting" branche="1.2.3.4.2.4" niveau="6" />
        <noeud synset="00005081" nom="svm" branche="1.2.3.4.2.5" niveau="6" />
        <noeud synset="00005208" nom="knn" branche="1.2.3.4.2.6" niveau="6" />
        <noeud synset="00005331" nom="naive_baye" branche="1.2.3.4.2.7" niveau="6" />
      </noeud>
```

Figure 5.10 Extrait de l'arbre XML relatif à l'ontologie *classification des documents*.

Calcul de la similarité.

La similarité des graphes est ensuite calculée. Le processus parcourt le fichier *fichzone.noun* pour déterminer à quelle zone est associé chaque concept appartenant aux graphes. Le calcul de la similarité met en correspondance les concepts appartenant à la même zone. Le processus retourne deux fichiers résultats : un premier fichier contient les différentes similarités entre documents comparés deux à deux. Un deuxième fichier contient les paires de documents suspects.

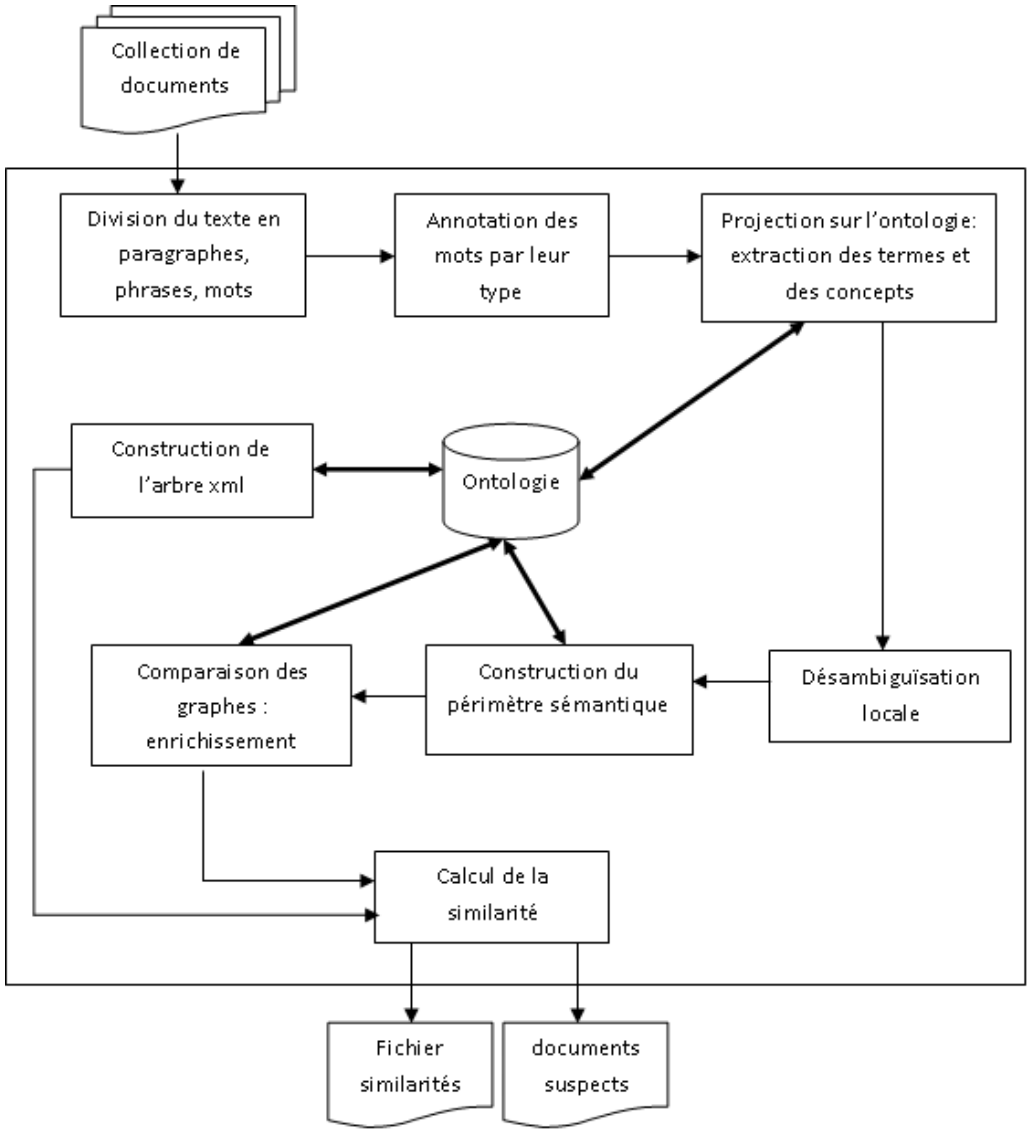

Figure 5.11 Calcul de la similarité sémantique des documents.

5.2.4 Sac-de-mots

Cette application permet de calculer la similarité des textes deux à deux en représentant le contenu des documents par des vecteurs de mots. L'architecture de l'application est donnée par la Figure 5.12.

- Les étapes permettant la division du texte en phrases et en mots, l'annotation des mots par leur type et la représentation vectorielle des documents sont identiques à celles de l'application classification avec Weka (cf. section 5.2.2).

- La similarité entre chaque paire de documents *Di* et *Dj* de la collection est calculée par la mesure du cosinus. Les valeurs des différentes similarités sont résumées dans un fichier résultat. Les résultats sont présentés sous forme d'un tableau où les colonnes représentent respectivement les documents *Di*, *Dj* et leurs Valeurs de similarité.

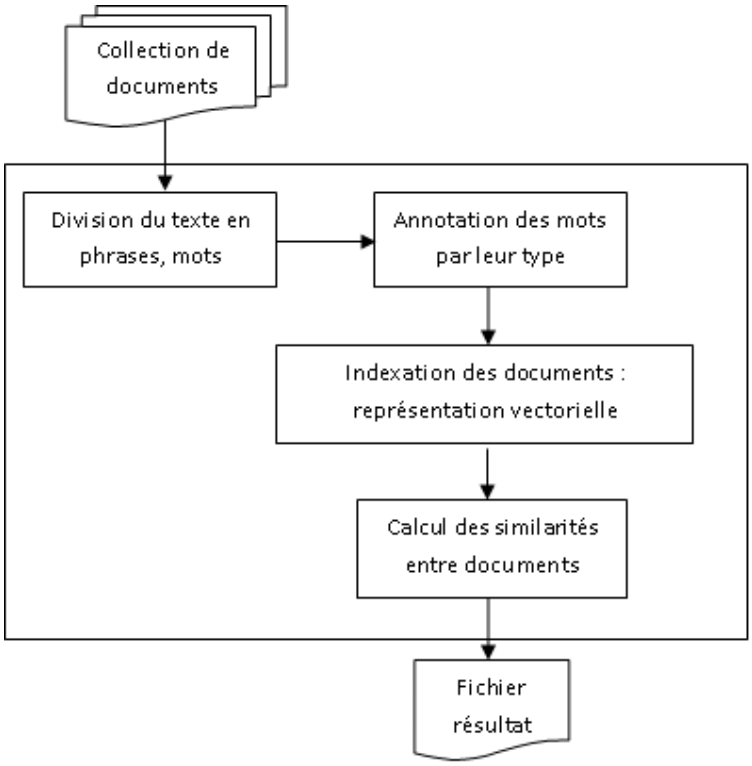

Figure 5.12 Similarité des textes basée sur la représentation sac-de-mots.

5.2.5 N-grammes

Dans cette application, le découpage du texte d'un document ne se fait pas de la même manière que pour les applications précédentes. L'architecture de l'application est donnée dans la Figure 5.13.

- Le texte est découpé en un ensemble de n-grammes. La taille des n-grammes est choisie par l'utilisateur.

- L'indexation des documents est basée sur le calcul du nombre d'occurrence et des fréquences relatives des n-grammes.

- Des similarités entre chaque paire *Di* et *Dj* de document sont calculées en faisant varier la valeur de *n* (donnée en entrée) et en utilisant deux équations (cf. chapitre 4, section 4.4.1). Le but étant de choisir la combinaison (*n, équation*) qui donne le minimum de rapprochements erronés entre documents. Les résultats sont ensuite résumés dans un fichier résultat sous forme d'un tableau de 4 colonnes : *Di*, *Di*, les valeurs de leurs similarités (cf. chapitre 4).

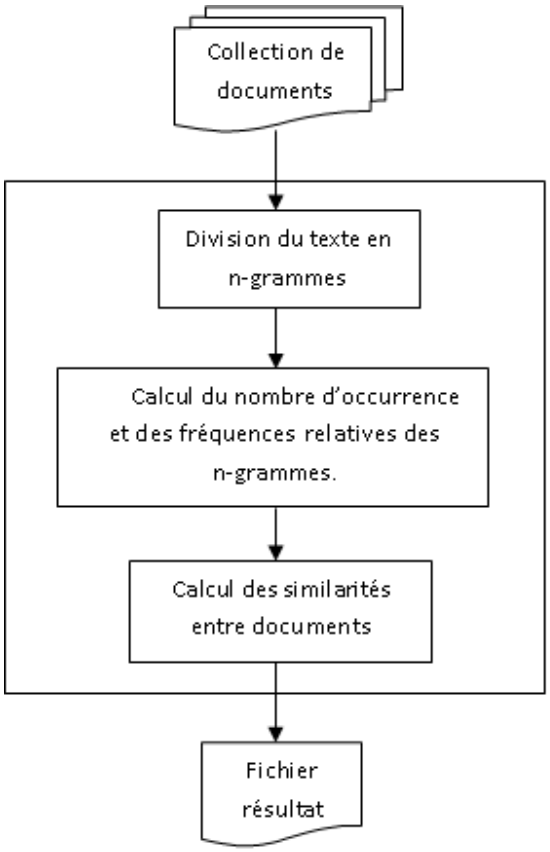

Figure 5.13 Similarité des textes basée sur la représentation n-grammes

5.3 Conclusion

Notre implémentation inclut cinq applications. Leur indépendance permet leur utilisation par d'autres applications.

- Classification sémantique des documents (CBO)

Dans cette application, nous avons implémenté la projection des mots d'un document sur WordNet pour l'extraction des termes les plus longs et les synsets leur correspondant. Nous avons également implémenté le processus de désambiguïsation en utilisant la bibliothèque Rita. Ces deux processus sont largement utilisés par des applications visant à donner une représentation sémantique des documents en exploitant WordNet. Les parties du programme associées à ces processus peuvent être exploités par d'autres utilisateurs. Il suffit de les définir en tant que modules.

- Similarité sémantique des documents

Cette application peut être réutilisée pour calculer la similarité entre documents appartenant à une autre ontologie. Il suffit d'introduire les informations de l'ontologie dans les différents fichiers textes que nous avons crées.

- Classification en utilisant les classifieurs conventionnels

Ce programme peut être exploité entièrement par toute application visant à classifier des documents par les classifieurs implémentés dans Weka.

- Sac-de-mots et N-grammes

Ces programmes peuvent être exploités par différentes applications nécessitant une représentation vectorielle des documents soit par des mots, soit par des n-grammes. La similarité entre documents peut être calculée en utilisant d'autres équations, il suffit de les intégrer aux programmes.

Pour l'application n-gramme, l'utilisateur peur choisir la taille des n-grammes. Le programme l'invite à introduire la valeur de *n* de son choix.

Conclusion générale

Synthèse

Les travaux présentés dans cette thèse se situent dans le contexte de l'évaluation de la similarité des textes basée sur l'exploitation de ressources sémantiques. Une ressource sémantique externe peut être principalement un thésaurus ou une ontologie. Ces ressources représentent des connaissances dans une structure reliant des concepts par un ensemble de relations. D'importants travaux se focalisent sur la construction de ressources sémantiques et de nombreuses ressources externes sont disponibles dans la littérature.

Le calcul de la similarité des textes est nécessaire pour plusieurs applications ayant recours au traitement du langage naturel. La similarité peut être évaluée au niveau mot, au niveau phrase et au niveau document. Les approches contextuelles considèrent un mot relativement à son contexte d'apparition. Ces approches se basent sur l'hypothèse qui stipule que l'information contextuelle donne une bonne approximation du sens des mots. Les approches récentes visent alors à donner une représentation conceptuelle des documents basée sur des ressources externes.

Dans notre travail, nous nous sommes intéressés à l'introduction des ontologies à différents niveaux de notre approche. Nous avons présenté plusieurs contributions permettant de donner une représentation sémantique des documents sous forme de graphes basée à la fois sur le contexte d'apparition des termes du document et sur l'exploitation des connaissances représentées par des ontologies.

Une première contribution de ce travail est la détermination de la similarité globale des textes déduite à partir du contexte dans lequel s'inscrit leur contenu. Cette similarité repose sur une classification sémantique des documents. Le classifieur proposé utilise en entrée un ensemble d'ontologies de domaine. Plusieurs étapes sont alors nécessaires pour déterminer quelle ontologie représente le mieux le contenu d'un document.

Une première étape consiste à projeter les termes du document sur les différentes ontologies pour extraire de chaque ontologie, les concepts qui leur correspondent. Pour un terme donné, plusieurs concepts d'une même ontologie peuvent être candidats. Une étape de désambiguïsation locale est alors nécessaire.

Le processus de désambiguïsation locale constitue notre deuxième contribution. Ce processus tient compte de la distance sémantique séparant le terme ambigu de ses voisins non ambigus, les plus proches dans un contexte donné. Ce processus est répété de façon récursive et sur trois niveaux en considérant d'abord le niveau phrase, puis le niveau paragraphe et enfin le niveau document. A l'issue de cette étape, un document possède plusieurs représentations conceptuelles extraites à partir des ontologies considérées. Une dernière étape calcule pour chaque document, le score obtenu par les différentes ontologies. Le classifieur associe alors un document à l'ontologie qui obtient le meilleur score.

Pour évaluer notre classifieur sémantique, nous avons utilisé deux ressources sémantiques : WordNet et WordNet Domains et nous avons sélectionné un ensemble de domaines à partir de WordNet Domains que nous avons associé à des ontologies de domaine. Nous avons construit une collection de documents contenant des résumés d'articles scientifiques extraits à partir du corpus Muchmore et des sites Web des journaux spécialisés dans les domaines retenus. Nous avons comparé les résultats de notre classifieur à ceux obtenus par trois classifieurs conventionnels. Les résultats ont montré les performances de notre classifieur qui a permis d'améliorer les résultats de la classification des documents.

Notre troisième contribution concerne le calcul de la similarité des textes. Les ontologies sont également utilisées dans ce traitement. La similarité des textes telle que nous l'avons définie repose sur l'idée que le sens d'un document est décrit par les termes explicitement cités dans son contenu mais peut être complété par une information implicite déduite à travers les liens sémantiques reliant les termes dans leur contexte et à travers le parcours de l'ontologie. Nous avons introduit la notion de périmètre sémantique qui a pour objectif d'enrichir les graphes initiaux des documents construits par le classifieur sémantique. Cette notion de périmètre sémantique nous permet de mettre en valeur des informations implicites qui seront utilisées lors de la comparaison des graphes des documents. Un enrichissement mutuel des graphes est réalisé à cette étape permettant ainsi de retrouver une similarité entre textes abordant des sujets similaires en utilisant des mots différents. Nous avons défini un poids pour les concepts. Ce poids indique la présence explicite ou implicite des concepts dans le document. Nous avons également proposé une mesure pour mesurer la similarité des documents

Parmi les nombreux domaines d'application possibles de notre approche, nous nous sommes intéressés à la détection de plagiat dans les articles scientifiques. Notre objectif est de retrouver des similarités entre deux articles à travers l'examen de leurs résumés. Une dernière contribution de ce travail est représentée alors par la structuration des résumés en trois parties distinctes appelées zones. Le but étant d'extraire les notions relatives à chaque zone et de comparer ensuite le contenu des zones de même type. Les zones définies sont le *contexte,* la *contribution* et le *domaine d'application*. Nous considérons que ces trois parties se retrouvent dans la plupart des résumés scientifiques. Des similarités partielles relatives à chaque zone sont calculées puis combinées pour calculer une similarité globale. Les résumés sont triés par ordre décroissant de leur similarité globale. Des seuils déterminés par expérimentation pour les différentes similarités calculées permettent de retenir les documents suspects représentant un risque de plagiat.

Pour évaluer notre approche, nous avons conçu une structure ontologique du domaine de la classification des documents et un corpus contenant des résumés d'articles scientifiques relatifs à ce domaine. Les résultats sont ensuite comparés à ceux de deux approches conventionnelles. La première approche *sac-de-mots* représente le contenu des documents par un ensemble de mots indépendants les un de autres. La deuxième approche *n-grammes* divise le contenu des résumés en un ensemble de n-grammes où *n* défini le nombre de caractères consécutifs formant un n-gramme. Ces deux approches ne considèrent évidemment ni le sens des mots, ni le sens du document dans leur représentation et dans le processus d'appariement des documents. Les résultats obtenus montrent que notre processus obtient de meilleurs résultats et permettent de conclure sur l'intérêt d'une telle approche.

Perspectives

Les perspectives envisageables pour nos travaux portent sur les points suivants.

Un premier point concerne l'annotation sémantique des concepts au sein d'une ontologie de domaine. Pour le calcul de la similarité des résumés, nous avons annoté les concepts représentant une ontologie par les trois zones que nous avons définies. Cette annotation a été réalisée manuellement pour l'ontologie du domaine de la classification des documents que nous avons construite. Pour compléter notre approche, une annotation automatique ou semi automatique des concepts est nécessaire. Nous pouvons nous inspirer des travaux menés sur WordNet Domains où certains concepts sont annotés manuellement. A partir de ces concepts et en parcourant les différentes relations entre concepts dans la structure de l'ontologie, l'annotation des autres concepts peut être définie.

Un deuxième point concerne l'extraction des termes d'un document. La projection telle qu'elle est réalisée extrait les termes qui possèdent un sens dans l'ontologie sur laquelle le document est projeté. Le sens retenu pour ce terme ne correspond pas toujours au sens réel du terme dans la phrase. La combinaison de ce terme avec d'autres mots contigus dans la phrase peut correspondre à un concept d'un autre domaine non considéré dans le processus de projection. Pour limiter l'extraction erronée des termes dans le processus de classification impliquant différentes ontologies de domaine, il serait intéressant d'utiliser dans un premier temps, la ressource WordNet pour extraire une liste de groupes de mots qui sera ensuite exploitée pour ne retenir que les termes ayant un sens pour les domaines concernés par le processus de classification.

Un troisième point consiste à sélectionner les concepts liaison à retenir, notamment lors de l'exploitation des documents de taille plus importante que celle les résumés. Le périmètre sémantique des documents pourrait éventuellement nécessiter une adaptation puisque les notions abordées dans ces documents peuvent être variées et les termes pourraient être très éloignés dans la structure de l'ontologie. Les concepts liaison à rajouter seraient alors en grand nombre et pourraient faire ressortir trop de notions différentes. Le calcul de la similarité entre documents pourrait alors être affecté. Une sélection des concepts liaison doit être réalisée pour ne garder que les concepts liaisons dont le poids ne doit pas être inférieur à un certain seuil. Ce dernier sera défini par expérimentation.

Un quatrième point concerne l'évaluation du processus de désambiguïsation locale dans des tâches de désambiguïsation (WSD) en utilisant différentes ressources telles que WordNet et MeSh. La désambiguïsation locale constitue l'un des traitements composant le processus de la classification sémantique des documents. Elle permet de sélectionner pour un terme ambigu, le concept approprié parmi les concepts candidats au sein d'une même ontologie.

Une cinquième perspective porte sur l'utilisation de notre approche de calcul de la similarité des textes dans d'autres tâches. Une application possible pourrait être envisagée dans le domaine de la recherche d'information. Notre processus fonctionne en deux phases. La première phase permet de faire un regroupement en fonction du contexte global des documents autour d'une même ontologie de domaine et une deuxième phase calcule la similarité des textes en réalisant un rapprochement plus affiné des contenus des documents. Notre approche serait capable de retourner les documents pertinents relativement aux requêtes des utilisateurs comme par exemple constituer un fond documentaire pour un thème donné où la requête peut être un document entier. Le poids des concepts, calculé dans le processus de classification, peut être exploité afin de trier les documents par ordre décroissant de similarité. Une comparaison avec la pondération *tf-idf* peut être réalisée afin de déterminer l'apport de notre pondération sur l'évaluation de la pertinence des documents.

Références

[Ahmad, 1996] K. Ahmad. *Language engineering and the processing of specialist terminology*. http://www.computing.surrey.ac.uk/ai/pointer/paris.html, 1996.

[Alemzadeh et al.,2010] M. Alemzadeh and F. Karray. *An Efficient Method for Tagging a Query with Category Labels Using Wikipedia towards Enhancing Search Engine Results*. In: 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Toronto, Canada, pp. 192-195, 2010.

[Alvarez et al., 2004] C. Alvarez, P. Langlais, J.Y. Nie. *Word Pairs in Language Modeling for Information Retrieval*. In Proceeding RIAO '04 Coupling approaches, coupling media and coupling languages for information retrieval, pp. 686-705, Vaucluse, France, 2004.

[Ashburner et al., 2000] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill4, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock. Gene ontology : Tool for the unification of biology. Journal of Nature genetics Vol. 25, issue 1, pp. 25-29, 2000.

[Auer et al. 2008] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak and Z. Ives. *DBpedia : A Nucleus for a Web of Open Data*. In Proceedings of the 6th International Semantic Web Conference (ISWC), Lecture Notes in Computer Science, vol. 4825, pp. 722–735, Springer, 2008.

[Bachimont, 2004] B. Bachimont, *Arts et sciences du numérique : Ingénierie des connaissances et critique de la raison computationnelle*. Mémoire d'Habilitation à Diriger des Recherches, Université de Technologie de Compiègne, 2004.

[Basile et al., 2008] C. Basile, D. Benedetto, E. Caglioti, and M. D. Esposti. *An example of mathematical authorship attribution*. Journal of Mathematical Physics, Vol. 49, Issue 12, pp. 125211-1–125211-20, 2008.

[Basile et al., 2009] C. Basile, D. Benedetto, E. Caglioti, G. Cristadoro and M. D. Esposti. *A plagiarism detection procedure in three steps : selection, matches and squares*. 3rd Workshop on Uncovering Plagiarism, Authorship and Social Software Misuse, PAN 2009.

[Baziz et al., 2005a] M. Baziz, M. Boughanem and N. Aussenac-Gilles. *Conceptual Indexing Based on Document Content Representation*. In Proceeding of the 5th international conference on Context : conceptions of Library and Information Sciences, pp. 171-186, Glasgow, UK, 2005.

[Baziz et al., 2005b] M. Baziz, M. Boughanem, H. Prade and G. Pasi. *A Fuzzy Set Approach to Concept-based Information Retrieval*. In Proceedings of the 4th Conference of the European Society for Fuzzy Logic and Technology and the 11ème Eleventh Rencontres Francophones sur la Logique Floue et ses Applications (Eusflat-LFA 2005 joint Conference), pp. 1287–1292, Barcelona, Spain, 2005.

[Beale et al., 1995] S. Beale, S. Nirenburg and K. Mahesh. *Semantic Analysis in the Mikrokosmos Machine Translation Project*. In Proceedings of the Second Symposium on Natural Language Processing, pp. 297-307, Bangkok, Thailand, 1995.

[Bendaoud, 2009] R. Bendaoud. *Analyses formelle et relationnelle de concepts pour la construction d'ontologies de domaines à partir de ressources textuelles hétérogènes*. PhD thesis, Henri Poincaré University, Nancy 1, 2009.

[Beyer et al., 1999] K. Beyer, J. Goldstein, R. Ramakrishnan and U. Shaft. *When is `nearest neighbor' meaningful*. In Proceedings of ICDT, International Conference on Database Theory, pp. 217-235, 1999.

[Blanchard et al., 2008] E. Blanchard, M. Harzallah, P. Kuntz and H. Briand. *Sur l'évaluation de la quantité d'information d'un concept dans une taxonomie et la proposition de nouvelles mesures*. In MC 2008, Revue des Nouvelles Technologies de l'Information, vol. RNTI-E-12, pp.127-146, 2008.

[Borst, 1997] W. N Borst. *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD thesis, Centre for Telematics and Information Technology (CTIT), University of Twente, Enschede, 1997.

[Bourigault, 1992] D. Bourigault *Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases*. In Proceedings of the Fourteenth International Conference on Computational Linguistics-COLING 92, pp. 977-981, Nantes, 1992.

[Bourigault et al., 2000] D. Bourigault and C. Fabre. *Approche linguistique pour l'analyse syntaxique de corpus*. Cahiers de Grammaire, Université Toulouse Le Mirail, n° 25, pp. 131-151, 2000.

[Bowker et al., 2002] L. Bowker and J. Pearson. *Working with specialized language : A practical guide to using corpora*. Routledge Editor, 2002.

[Brill, 1995] E. Brill. *Unsupervised learning of disambiguation rules for part of speech tagging*. In Natural Language Processing Using Very Large Corpora, pp. 1-13. Kluwer Academic Press, 1995.

[Brin et al., 1995] S. Brin, J. Davis and H. Garcia-Molina. *Copy detection mechanisms for digital documents*. In Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data, pp. 398–409, San Jose, California, 1995.

[Brown et al., 1990] P. F. Brown, J. C. Cocke, A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer and P. S. Roossin. *A statistical approach to machine translation*. Computational linguistics, Vol. 16, Issue 2, pp. 79-85, 1990.

[Burgess et al., 1998] C. Burgess, K. Livesay, and K. Lund. 1998. *Explorations in context space : Words, sentences, discourse*. Journal of Discourse Processes, Vol. 25, issue 2-3, pp. 211–257.

[Carmel et al., 2003] D. Carmel, Y. Maarek, M. Mandelbrod, Y. Mass and A. Soffer. *Searching xml documents via xml fragments*. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 151– 158, Toronto, Canada, 2003.

[Cheeseman et al., 1996] P. Cheeseman and J. Stutz. *Bayesian classification (autoclass) : theory and results*. In Advances in Knowledge Discovery and Data Mining, pp. 153–180, 1996.

[Choueka, 1988] Y. Choueka. *Looking for Needles in a Haystack or Locating Interesting Collocationnal Expressions in a Large Textual Databases*. Dans Actes de colloque du RIAO, pp. 609-624, 1988.

[Church et al., 1989] K. W. Church and P. Hanks. *Word Association Norms, Mutual Information, and Lexicography*. Journal of Computational Linguistics, vol. 16, Issue 1, pp. 22-29, 1989.

[Cimano et al., 2004] P. Cimano, S. Handschuh, and S. Staab. *Towards the self-annotating web*. In Proceedings of the 13th international conference on World Wide Web, pp. 462-471, New York, USA, 2004.

[Claveau, 2003] V. Claveau. *Acquisition automatique de lexiques sémantiques pour la recherche d'information*. Thèse de doctorat, Université de Rennes 1, 2003.

[Coyaud, 1968] M. *Coyaud. Resolution of lexical ambiguities in ophthalmology*. Cornell University, Ithaca, New York, 1968.

[Cunningham, 2002] H. Cunningham(2002). *Gate, a general architecture for text engineering*. Journal of Computers and the Humanities, Vol. 36, Issue 2, pp. 223-254.

[Cunningham et al. 2011] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, N. Aswani, I. Roberts, G. Gorrell, A. Funk, A. Roberts, D. Damljanovic, T. Heitz, M. A. Greenwood, H. Saggion, J. Petrak, Y. Li and W. Peters. *Text Processing with GATE*. University of Sheffield Department of Computer Science, 2011.

[Curran, 2002] J. Curran. *Ensemble methods for automatic thesaurus extraction*. In Proceedings of the conference on Empirical methods in natural language processing (EMNLP), Vol.10, pp. 222-229, Philadelphia, 2002.

[Curran, 2003] J. Curran. *From distributional to semantic similarity*. PhD thesis, Institute for Communicating and Collaborative Systems, School of Informatic, University of Edinburgh, 2003.

[Cutting et al., 1992] D. Cutting, D. Karger, J. Pedersen, and J. Tukey. *Scatter/Gather : a cluster-based approach to browsing large document collections*. In Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 318-329, Copenhagen, Denmark, 1992.

[Daille, 1993] B. Daille. *Extraction automatique de terminologie monolingue*. Dans Actes du colloque Informatique et langue naturelle, Nantes, 1993.

[Daille 1994a] B. Daille. *Approche mixte pour l'extraction de terminologie : statistique lexicale et filtres linguistiques*. Thèse de doctorat en Informatique Fondamentale, Université de Paris 7, Paris, 1994.

[Daille 1994b] B. Daille. *Extraction de noms composés terminologiques du domaine des Télécommunications*. Dans 5èmes Journées ERLA-GLAT (Études et Recherches Lexicales appliquées), Brest, 1994.

[David et al., 1990] S. David and P. Plante. *Termino version 1.0*, Rapport du centre Ato Analyse de Textes par Ordinateur, Université du Québec, 1990.

[Deerwester et al., 1990] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. *Indexing by latent semantic analysis*. Journal of the American Society of Information Science, Vol. 41, Issue 6, pp. 391–407, 1990.

[Ding et al., 2001] Y. Ding and R. Engels. *IR and AI : Using co-occurrence Theory to Generate Lightweight Ontologies*. DEXA Workshop, pp. 961-965, 2001.

[Dudognon et al., 2010] D. Dudognon, G. Hubert and B. Ralalason. *Proxigénéa : Une mesure de similarité conceptuelle*. In Proceedings of the Colloque Veille Stratégique Scientifique et Technologique (VSST 2010), 2010.

[Errami et al., 2009] M. Errami, Z. Sun, T. C. Long, A. C. George and H. R. Garner. *Déjà vu: a database of highly similar citations in the scientific literature*. Nucleic Acids Research, Vol. 37, Issue suppl_1, pp. 921–924, 2009.

[Fagan, 1987] J. L. Fagan. *Experiments in Automatic Phrase Indexing for Document Retrieval : A Comparison of Syntactic and Non-syntactic methods,* PhD thesis, Department of Computer Science, Cornell University, 1987.

[Feldman et al., 2007] R. Feldman and J. Sanger. The Text mining handbook : Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, 2007.

[Ferreira et al., 2016] R. Ferreira, R. D. Lins, S. Simske, F. Freitas, and M. Riss. *Assessing sentence similarity through lexical, syntactic and semantic analysis*. Journal of Computer Speech and Language, vol. 39, pp. 1–28, 2016.

[Finlayson, 2014] M. A. Finlayson. *Java Libraries for Accessing the Princeton WordNet : Comparison and Evaluation*. In the 7th Conference on Global WordNet (GWC), Tartu, Estonia, 2014.

[Fortuna et al., 2007] B. Fortuna, M. Grobelnik and D. Mladenic. *OntoGen Semi-automatic Ontology Editor*. In symposium of Human Interface and the Management of Information. Interacting in Information Environments, Lecture Notes in Computer Science, Vol. 4558, pp. 309-318, Springer, Berlin, Heidelberg, 2007.

[Foskett, 1980] D. J. Foskett. *Thesaurus*. In Encyclopedia of Library and Information Science, eds. A. Kent, H. Lancour and J.E. Daily, Vol. 30, pp. 416-463, New York : Marcel Dekker , 1980.

[Fragos et al., 2003] K. Fragos, I. Maistros, and C. Skourlas. *Word sense disambiguation using WordNet relations*. In Proceedings of the 1st Balkan Conference in Informatics, Thessaloniki Greece, 2003.

[Fuhr et al., 2001] N. Fuhr and K. Grossjohann. *XIRQL : a query language for information retrieval in XML documents*. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 172-180, New Orleans, Louisiana, USA, 2001.

[Gabrilovich et al., 2005] E. Gabrilovich and S. Markovitch. *Feature Generation for Text categorization Using World Knowledge*. In Proceedings of IJCAI 2005 : the Nineteenth International Joint Conference on Artificial Intelligence, pp. 1048-1053, Edinburgh, Scotland, UK, 2005.

[Gabrilovich et al., 2007] E. Gabrilovich and S. Markovitch. *Computing semantic relatedness using Wikipedia based explicit semantic analysis*. In Proceedings of the 20th International Joint Conference on Artificial Intelligence, pp. 1606-1611, Hyderabad, India, 2007.

[Galdos et al., 2017] L. Galdos, G. Guillen and C. Del Alamo. *A New Graph-Based Approach for Document Similarity Using Concepts of Non-Rigid Shapes*. In proceeding of the IMMM 2017: The Seventh International Conference on Advances in Information Mining and Management, pp. 41-46, Venice, Italy, 2017.

[Gaussier et al., 2003] E. Gaussier, C. Jacquemin and P. Zweigenbaum. *Traitement automatique des langues et recherche d'information*. In Éric Gaussier and Marie-Hélène Stefanini, editors, Assistance intelligente à la recherche d'informations, chapter 2, pages 71-96. Hermès-Lavoisier, Paris, 2003.

[Glickman et al., 2003] O. Glickman and I. Dagan. *Acquiring lexical paraphrases from a single corpus : A case study for verb*. In Proceedings of Recent Advances in Natural Language Processing, pp.81–90, Borovets, Bulgaria, 2003.

[Goller et al., 2000] C. Goller, J. Löning, T. Will, and W. Wolff. *Automatic document* classification - a thorough evaluation of various methods. In Internationales Symposium für informationswissenschaft (ISI), pp. 145–162, 2000.

[Gomez-Perez, 1999] A. Gomez-Perez. *Ontological Engineering : a State of the Art.* Expert Update. Vol. 2, Issue 3, pp. 33 – 43, 1999.

[Greene et al., 1971] B. B. Greene and G. M. Rubin. *Automatic grammatical tagging of English*. Department of Linguistics, Brown University, Providence, Rhode Island, 1971.

[Gruber, 1993] T.R. Gruber, A translation approach to portable ontology specifications. Journal of Knowledge Acquisition, Vol. 5, issue 2, pp. 199-220, 1993.

[Gruber, 1995] T.R. Gruber. *Towards principles for the design of ontologies used for knowledge sharing*. International Journal of Human and Computer Studies, Vol. 43, issue 5-6, pp. 907-928, 1995.

[Gruninger et al., 1995] M. Grunninger and M. Fox, *Methodology for the design and evaluation of ontologies*. In Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI'95, 1995.

[Guarino, 1998] N. Guarino. *Formal Ontology and Information Systems.* In Proceedings of the First International Conference FOIS'98, IOS Press, pp. 3-15, Trento, Italy, 1998.

[Guo et al., 2011] Y. Guo, A. Korhonen and T. Poibeau. *A Weakly-supervised Approach to Argumentative Zoning of Scientific Documents*. In Proceedings of the 2011 conference on Empirical Methods in Natural Language Processing, pp. 273–283, Edinburgh, UK, 2011.

[Habert et al., 1997] B. Habert, A. Nazarenko and A. Salem. *Les linguistiques de corpus*. Armand Colin, 1997.

[Hall et al., 2009] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. *The weka data mining software : an update*. SIGKDD Exploration, vol. 11, issue 1, pp. 10-18, 2009.

[Hamon et al., 2007] T. Hamon, J. Derivière J. and A. Nazarenko. *Ogmios : a scalable nlp platform for annotating large web document collections*. In Proceedings of Corpus Linguistics, Birmingham, UK, 2007.

[Harris, 1954] Z.S. Harris. *Distributional structure*. Word, Vol. 10, Issue 2-3, pp. 146-162, 1954.

[Hastie et al., 1998] T. Hastie and R. Tibshirani. *Classification by Pairwise Coupling*. The Annals of Statistics, Vol. 26, Issue 2, pp. 451–471, 1998.

[Hernandez, 2005] N. Hernandez. *Ontologies de domaine pour la modélisation du contexte en recherche d'information*. Thèse de doctorat, Université Paul Sabatier de Toulouse, 2005

[Hotho et al., 2002] A. Hotho, A. Maedche and S. Staab. *Ontology-based Text Document Clustering*. KI, Vol. 16, Issue 4, pp. 48-54, 2002.

[Howe, 2009] D. C. Howe. *RiTa : creativity support for computational literature*. In Proceedings of the seventh ACM conference on Creativity and cognition (C&C '09), pp. 205-210, Berkeley, California, USA, 2009.

[Iltache et al., 2016] S. Iltache, C. Comparot, M. Si Mohammed and P. J. Charrel. *Using domain ontologies for classification and semantic interpretation of documents.* In Proceedings of ALLDATA 2016: 2nd International Conference on Big Data, Small Data, Linked Data and Open Data, pp. 76-81, 2016.

[Iltache et al., 2018] S. Iltache, C. Comparot, M. Si Mohammed and P. J. Charrel. *Using semantic perimeters with ontologies to evaluate the semantic similarity of scientific papers*. In Informatica: an International Journal of Computing and Informatics, Vol. 42, Issue 3, 2018.

[Jacquemin, 1997] C. Jacquemin. *Variation terminologique : Reconnaissance et acquisition automatiques de termes et de leurs variantes en corpus*. Mémoire d'habilitation à diriger des recherches en informatique fondamentale, Université de Nantes, 1997.

[Jaillet et al., 2003] S. Jaillet, J. Chauché, V. Prince, and M. Teisseire. *Classification automatique de documents: La mesure des deux écarts*. Actes du XXIème Congrès INFORSID, pp. 87–102, Nancy, France, 2003.

[Jaillet et al., 2006] S. Jaillet, A. Laurent and M. Teisseire. *Sequential patterns for text categorization*. Journal of Intelligent Data Analysis, IOS Press, Vol.10, issue 3, pp.199–214, 2006.

[Joachims, 1997] T. Joachims. *A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization*. In Proceedings of the Fourteenth International Conference on Machine Learning, pp.143-151, Tennessee, 1997.

[Joachims, 1998] T. Joachims. *Text categorization with support vector machines: learning with many relevant features.* In Proceedings of ECML-98, 10th European Conference on Machine Learning, Chemnitz, pp. 137–142, Germany, 1998.

[John et al., 1995] G. H. John and P. Langley. *Estimating Continuous Distributions in Bayesian Classifiers.* In Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338-345, San Mateo, 1995.

[Justeson et al., 1995] J. Justeson and S. Katz. *Technical terminology: some linguistic properties and an algorithm for identification in text*. Journa of Natural Language Engineering, Vol. 1, Issue 1, pp. 9-27, 1995.

[Keerthi et al. 2001] S.S. Keerthi, S.K. Shevade, C. Bhattacharyya and K.R.K. Murthy. *Improvements to Platt's SMO Algorithm for SVM Classifier Design*. Neural Computation, vol. 13, issue. 3, pp. 637–649, 2001.

[Khan, 2000] L. R. Khan. *Ontology-based Information Selection*. PhD Thesis, Faculty of the Graduate School, University of Southern California, 2000.

[Kolt et al., 2009a] S. G. Kolte and S. G. Bhirud. *Exploiting links in WordNet hierarchy for word sense disambiguation of nouns*. In Proceedings of the International Conference on Advances in Computing, Communication and Control, (ICAC3'09), pp. 20-25, Mumbai, India, 2009.

[Kolte et al., 2009b] S. G. Kolte and S. G. Bhirud. *WordNet: A Knowledge Source for Word Sense Disambiguation*. International Journal of Recent Trends in Engineering, Vol. 2, Issue 4, pp. 213-217, 2009.

[Krovetz, 1997] R. Krovetz. *Homonymy and polysemy in information retrieval*. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, pp. 72-79, 1997.

[Lavie et al., 2009] A. Lavie and M. Denkowski, *The Meteor metric for automatic evaluation of Machine Translation*. Journal of Machine Translation, Vol. 23, Issue 2–3, pp. 105–115, 2009.

[Leacock et al., 1998] C. Leacock, G. A. Miller, and M. Chodorow. *Using corpus statistics and WordNet relations for sense identification*. Journal of Computational Linguistics, Vol. 24, Issue 1, pp. 147-165, 1998.

[Lebart et al, 1988] L. Lebart and A. Salem. *Analyse statistique des données textuelles : questions ouvertes et lexicométrie*. Dunod, Paris, 1988.

[Lebart et al., 1994] L. Lebart and A. Salem. *Statistique textuelle*. Dunod, Paris, 1994.

[Lee et al., 1996] J. Lee, M. Gruninger and the PIF Working group. *The PIF process interchange format and framework*. Working Paper Series 194, MIT Center for Coordination Science, 1996.

[Lesk, 1986] M. Lesk. *Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone*. In Proceedings of the Fifth Annual International Conference on Systems Documentation, pp.24-26, Toronto, Canada, 1986.

[Lewis et al., 1994] D. D. Lewis and M. Ringuette. *A comparison of two learning algorithms for text categorization*. In Third Annual Symposium on Document Analysis and Information Retrieval, pp. 81–93, 1994.

[Lewis et al., 2006] J. Lewis, S. Ossowski, J. Hicks, M. Errami and H. R. Garner. *Text similarity: an alternative way to search MEDLINE*. Journal of Bioinformatics Vol. 22, Issue 18, pp. 2298–2304, 2006.

[Lin, 1998a] D. Lin. *An information-theoric definition of similarity*. In Proceedings of the 15th international conference on Machine Learning, pp. 296-304, 1998.

[Lin, 1998b] D. Lin. *Automatic retrieval and clustering of similar words*. In Proceedings of the 17th International Conference on Computational Linguistics, Vol. 2, pp.768–774, Montreal, Quebec, Canada, 1998.

[Lindeberg et al., 1993] D.A. Lindberg, B.L. Humphreys and A.T. McCray. *The Unified Medical Language System*, journal of Methods of Information in Medicine, Vol. 32, issue 4, pp. 281-291, 1993.

[Lovins, 1968] J. B. Lovins. *Development of a stemming algorithm*. Mechanical Translation and Computational Linguistics, Vol. 11, pp. 22–31, 1968.

[Luhn, 1958] H. Luhn. *The automatic creation of literature abstracts*. IBM Journal of Research and Development, Vol. 2, Issue 2, pp. 159–165, 1958.

[Lukashenko et al., 2007] R. Lukashenko, V. Graudina and J. Grundspenkis. *Computer-Based Plagiarism Detection Methods and Tools: An Overview*. In Proceeding of the 2007 International Conference on Computer Systems and Technologies - CompSysTech'07, article N° 40, Bulgaria, 2007.

[Lund et al., 1996] K. Lund and C. Burgess. *Producing high-dimensional semantic spaces from lexical co-occurrence*. Journal of Behavior Research Methods, Instruments, and Computers, Vol. 28, issue 2, pp. 203- 208, 1996.

[McEnery et al., 1996] T. McEnery and A. Wilson. *Corpus linguistics*. Edinburgh University Press, 1996.

[Magnini et al, 2000] B. Magnini and G. Cavaglià. *Integrating Subject Field Codes into WordNet*. In Proceedings of LREC-2000, Second International Conference on Language Resources and Evaluation, pp. 1413-1418, Athens, Greece, 2000.

[Mendes et al. 2011] P. N. Mendes, M. Jakob, A. García-Silva and C. Bizer. *DBpedia Spotlight: Shedding Light on the Web of Documents*. In I-Semantics, 7th International Conference on Semantic Systems, Graz, Austria, 2011.

[Miller et al., 1991] G. Miller and W. Charles. *Contextual correlates of semantic similarity*. Journal of Language and Cognitive Processes, Vol. 6, Issue 1, pp. 1–28, 1991.

[Miller et al., 1993] G. Miller, C. Leacock, R. Tengi, and R. Bunker. *A semantic concordance*. In Proceedings of the workshop on Human Language Technology, pp. 303-308, 1993.

[Miller et al., 1995] G. Miller. *WordNet: A Lexical Database for English*. Communications of the ACM Vol. 38, issue 11, pp. 39-41, 1995.

[Milne et al., 2008] D. Milne and I. H. Witten. *Learning to link with Wikipedia*. In Proceedings of the 17th ACM Conference on Information and Knowledge Management, California, USA, pp. 509–518, 2008.

[Mizoguchi, 2003] R. Mizoguchi. *Tutorial on Ontological Engineering. Part 1: Introduction to Ontological Engineering*. Journal of New Generation Computing, Vol. 21, Issue 4, pp. 365-384, 2003.

[Mohhebi et al., 2016] M. Mohebbi and A. Talebpour. *Texts Semantic Similarity Detection Based Graph Approach*. The International Arab Journal of Information Technology Vol. 13, Issue 2, pp. 246-251, 2016.

[Navigli, 2009] R. Navigli. *Word Sense Disambiguation: a survey*. Journal of ACM Computing Surveys, Vol. 41, Issue 2, pp. 1–69, 2009.

[Niles et al., 2001] I. Niles and A. Pease. *Toward a Standard Upper Ontology.* In Proceedings of the 2nd International Conference on Formal Ontology in Information Systems, pp. 2-9, Ogunquit, Maine, USA, 2001.

[Nonaka et al., 1997] I. Nonaka and H. Takeuchi. *The knowledge creating company: how Japanese companies create the dynamics of innovation*. Oxford University Press. Traduction française de Marc Ingham, *La connaissance créatrice : dynamique de l'entreprise apprenante*, Management, DeBoeck Université, 1997.

[Omodei et al., 2014] E. Omodei, Y. Guo, J. P. Cointet and T. Poibeau. *Analyse discursive automatique du corpus ACL Anthology*. In Actes de la 21ème conférence Traitement Automatique des Langues Naturelles, Marseille, 2014.

[Osman et al., 2011] A. H. Osman, N. Salim, M. S. Binwahlan, H. Hentably and A. M. Ali. *Conceptual similarity and graph-based method for plagiarism detection*. Journal of Theoretical and Applied Information Technology, Vol. 32, Issue 2, pp. 135-145, 2011.

[Otegi et al., 2015] A. Otegi, X. Arregi, O. Ansa and E. Agirre, *Using knowledge-based relatedness for information retrieval*. Journal of Knowledge and Information Systems, Vol. 44, Issue 3, pp. 689–718, 2015.

[Partha et al., 1994] D. Partha and P. A. David. *Towards a new economics of science*. Journal of Research Policy, Vol. 23, Issue 5, pp. 487–521, 1994.

[Pearson, 1998] J. Pearson. *Terms in Context*. John Benjamins publishing company, 1998.

[Perron, 1996] J. Perron. *Adepte-NOMINO: un outil de veille terminologique*. Dans Terminologies nouvelles, Vol. 15, pp. 32-47, 1996.

[Pilehvar et al., 2014] M.T. Pilehvar and R. Navigli. *A robust approach to aligning heterogeneous lexical resources*. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pp. 468–478, Baltimore, USA, 2014.

[Pilehvar et al., 2015] M. T. Pilehvar and R. Navigli. *From senses to texts: An all-in-one graph-based approach for measuring semantic similarity*. Journal of Artificial Intelligence Vol. 228, pp. 95–128, 2015.

[Pincemin, 2000] B. Pincemin. *Similarites texte–texts expérience d'une application de diffusion ciblée et propositions*. In Matemáticas y Tratamiento de Corpus, Actes du 2ème séminaire de l'Ecole interlatine de linguistique appliquée,San Millán de la Cogolla, Logroño, Espagne, Logroño : Fundación San Millán de la Cogolla, 2002, pp 35-52, 2000.

[Platt, 1998] J. Platt. *Fast Training of Support Vector Machines using Sequential Minimal Optimization*. In B. Schoelkopf and C. Burges and A. Smola, editors, Advances in Kernel Methods - Support Vector Learning, 1998.

[Ponzetto et al., 2007] S. P. Ponzetto and M. Strube. *Knowledge derived from Wikipedia for computing semantic relatedness*. Journal of Artificial Intelligence Research, Vol. 30, Issue 1, pp. 181–212, 2007.

[Porter, 1980] M.F. Porter. *An algorithm for suffix stripping*. Journal of Program, Vol. 14 Issue 3, pp. 130-137, 1980.

[Psyché et al., 2003] V. Psyché, O. Mendes and J. Bourdeau. *Apport de l'ingénierie ontologique aux environnements de formations à distance*. Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, ATIEF, Vol. 10, pp. 89-126, 2003.

[Qazi et al., 2018] A. Qazi and R. Goudar. An Ontology-based Term Weighting Technique for Web Document Categorization. Journal of Procedia Computer Science, Vol. 133, pp 75-81, 2018.

[Quillian, 1968] M. Quillian. Semantic Memory. In M. Minsky (Ed.), Semantic information Processing, pp. 227-270, MIT Press, 1968.

[Quinlan, 1986] R. Quinlan. *Induction of decision trees. Journal of Machine Learning*. vol. 1, pp. 81–106, 1986.

[Quinlan, 1993] R. Quinlan. *C4.5: Programs for machine learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[Rada et al., 1989] R. Rada, H. Mili, E. Bicknell and M. Blettner. *Development and application of a metric on semantic nets.* IEEE Transactions on systems, Man and Cybernetics, Vol 19, Issue 1, pp.17-30, 1989.

[Rada et al., 1989] A. Raganato, C. Delli Bovi and R. Navigli. *Neural sequence learning models for word sense disambiguation*. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1156-1167, 2017.

[Rapp, 2003] R. Rapp. *Word sense discovery based on sense descriptor dissimilarity*. In Proceedings of the Ninth Machine Translation Summit, pp. 315–322, 2003.

[Rastier, 2001] F. Rastier. *Arts et sciences du texte*. Presses Universitaires de France, 2001.

[Rastier et al, 1994] F. Rastier and M. Cavazza, A. Abeille. *Sémantique pour l'analyse* De la linguistique à l'informatique. Masson, Paris, 1994.

[Resnik, 1995] P. Resnik. *Using information content to evaluate semantic similarity in a taxonomy*. In Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol 1, pp. 448-453, Montreal, Quebec, Canada, 1995.

[Resnik, 1999] P. Resnik. *Semantic similarity in a taxonomy: An information based measure and its application to problems of ambiguity in natural language*. Journal of Artificial Intelligence Research, Vol.11, Issue 1, pp. 95-130, 1999.

[Robetson et al., 1994] S. Robertson and S. Walker. *Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval*. In Proceedings of the 17th annual international ACM/SIGIR conference on research and development in information retrieval, pp. 232–241, Dublin, Ireland, 1994.

[Robertson et al., 1997] S. E. Robertson and S. Walker. *On relevance weights with little relevance information*. In Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 16–24, 1997.

[Rubenstein et al., 1965] H. Rubenstein and J. Goodenough. *Contextual correlates of synonymy*. Communications of the ACM, Vol. 8, issue 10, pp.627-633, 1965.

[Rusu et al., 2009] D. Rusu, B. Fortuna, M. Grobelnik and D. Mladenić. Semantic Graphs Derived from Triplets with Application in Document Summarization. Informatica Vol.33, Issue 3, pp. 357–362, 2009.

[Sahlgren, 2006] M. Sahlgren. *The Word-Space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations between Words in High-Dimensional Vector Spaces*. Thèse de doctorat, Stockholm University, Stockholm, Sweden, 2006.

[Salton, 1971] G. Salton. *The SMART Retrieval System – Experiments in Automatic Document Processing*. Prentice-Hall, 1971.

[Salton et al., 1983] G. Salton and M.J. McGill. *Introduction to modern information retrieval*. McGraw-Hill computer Science Series, 1983.

[Sauvagnat, 2005] K. Sauvagnat. *Modèle flexible pour la recherche d'Information dans des corpus de documents semi structurés*. Thèse de doctorat, IRIT, Université Paul Sabatier de Toulouse, 2005

[Schileder et al., 2002] T. Schileder and H. Meus. *Querying and ranking XML documents*. Journal of the American Society for Information Science and Technology, Vol. 53, Issue 6, pp. 489–503, 2002.

[Schmid, 1994] H. Schmid. *Probabilistic part-of-speech tagging using decision trees*. In *Proceedings of the* International Conference on New Methods in Language Processing, Manchester, UK, 1994.

[Schneider, 2005] K. M. Schneider. *Techniques for improving the performance of naive bayes for text classification*. In Proceedings of the 6th international conference on Computational Linguistics and Intelligent Text Processing, pp. 682-693, Mexico, 2005.

[Schütz, 1992] H. Schütze. *Dimensions of meaning.* In Proceedings of the 1992 ACM/IEEE conference on Supercomputing, pp. 787–796, Minneapolis, Minnesota, USA, 1992.

[Schütz, 1998] H. Schütze. *Automatic word sense discrimination*. Journal of Computational Linguistics: Special Issue on Word Sense Disambiguation, Vol. 24, Issue 1, pp. 97–123, 1998.

[Shenoy et al., 2012] K. M. Shenoy, K.C. Shet and U.D. Acharya. *Semantic plagiarism detection system using ontology mapping*. Advanced Computing: An International Journal (ACIJ), Vol.3, Issue 3, pp. 59–62, 2012.

[Sinclair, 1991] J. Sinclair. *Corpus, Concordance, Collocation*. Oxford University Press, 1991.

[Singhal et al., 1996] A. Singhal, C. Buckley and M. Mitra. *Pivoted document length normalization*. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 21-29, Zurich, Switzerland, 1996.

[Singhal et al., 1997] A. Singhal, M. Mitra and C. Buckley. *Learning routing queries in a query zone*. In Proceedings of the 20th Annual international ACM SIGIR Conference on Research and Development in information Retrieval, pp. 25-32, Philadelphia, Pennsylvania, USA, 1997.

[Smadja, 1993] F. Smadja. *Retrieving collocations from text: Xtract*. Journal of Computational Linguistics-Special issue on using large corpora, Vol. 19, Issue 1, pp. 143-177, 1993.

[Soucy et al., 2001] P. Soucy, G. W. Mineau. *A Simple k-NN Algorithm For Text Categorization*. In Proceedings of IEEE International Conference on Data Mining, pp.647–648, San Jose, USA, 2001.

[Spark Jone, 1971] K. Sparck Jones. *Automatic keywords classification for information retrieval*, Archon Books, 1971.

[Spak Jones, 1979] K. Sparck Jones. *Experiments in relevance weighting of search terms*. Journal of Information Processing and Management, Vol. 15, Issue 3, pp. 133-144, 1979.

[Stein et al., 2005] B. Stein and S.M. zu Eissen. *Near Similarity Search and Plagiarism Analysis*. In Proceeding of the 29th Annual Conference of the GfKl Springer, pp. 430-437, 2005.

[Steyvers et al., 2007] M. Steyvers and T. Griffiths. Probabilistic topic models. In T. Landauer, D.S. McNamara, S. Dennis, and W. Kintsch, editors, Handbook of Latent Semantic Analysis. Erlbaum, 2007.

[Stone, 1969] P. J. Stone. *Improved quality of content analysis categories: computerized disambiguation rules for high frequency English words*. In Analysis of Communication Content, pp. 199-233, New York, 1969.

[Tar et al., 2011] H. H. Tar and T.T. Soe.Nyunt. *Ontology-Based Concept Weighting for Text documents*. International Conference on Information Communication and Management IPCSIT vol.16, IACSIT Press, Singapore, 2011.

[Toutanova et al., 2000] K. Toutanova and C. D. Manning. *Enriching the knowledge sources used in a maximum entropy part-of-speech tagger*. In Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora : Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics,Vol. 13, pp. 63–70, Hong Kong, 2000.

[Toutanova et al., 2003] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. *Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network*. In Proceedings of HLT-NAACL2003, pp. 252-259, 2003.

[Urieli et al., 2013] A. Urieli and L. Tanguy. *L'apport du faisceau dans l'analyse syntaxique en dépendances par transitions : études de cas avec l'analyseur Talismane*. In Actes de la 20e conférence du Traitement Automatique du Langage Naturel (TALN), Sables d'Olonne, France, 2013.

[Uschold et al., 1995] M. Uschold and M. King. *Towards a Methodology for Building Ontologies*. In Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing at the International Joint Conference on Artificial Intelligence (IJCAI'1995), 1995.

[Vani et al., 2015] K. Vani and D. Gupta. *Investigating the Impact of Combined Similarity Metrics and POS tagging in Extrinsic Text Plagiarism Detection System*. In Proceeding of the International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1578-1584, Kochi, India, 2015.

[Velardi et al., 2001] P. Velardi, M. Missikoff and R. Basili. *Identification of relevant terms to support the construction of domain ontologies.* In Proceedings of the workshop on Human Language technologies and Knowledge Management, Vol. 2001, Toulouse, France, 2001.

[Vial et al., 2018] L. Vial, B. Lecouteux and D. Schwab. *Approche supervisée à base de cellules LSTM bidirectionnelles pour la désambiguïsation lexicale*. 25e conférence sur le Traitement Automatique des Langues Naturelles (TALN), Rennes, France, 2018

[Voorhees, 1994] E.M. Voorhees. *Query expansion using lexical-semantic relations*. In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'94, pp. 61–69, Dublin, Ireland, 1994.

[Wang et al., 2008] D. Wang, T. Li, S. Zhu and C. Ding. *Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization*. In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'08, pp.307–314, Singapore, Singapore, 2008.

[Wang et al., 2012] H. Wang, Y. Guo, J. Li and X. Shi. *Research of the conceptual representing of documents based on light ontology*. 9th International Conference on Fuzzy Systems and Knowledge Discovery, (FSKD), Sichuan, China, 2012.

[Weiss, 1973] S. F. Weiss. *Learning to disambiguate*. Journal of Information Storage and Retrieval, vol. 9, Issue 1, pp. 33-41, 1973.

[Wu et al., 1994] Z. Wu and M. Palmer. *Verb semantics and lexical selection*. In Proceedings of the 32nd Annual Meetings of the Associations for Computational Linguistics, pp. 133-138, Las Cruces, New Mexico, 1994.

[Yamamoto et al., 2003] E. Yamamoto, M. Kishida, Y. Takenami, Y. Takeda and K. Umemura. *Dynamic programming matching for large scale information retrieval*. In Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages, Vol.11, pp. 100–108, Sapporo, Japan, 2003.

[Yang et al., 1999] Y. Yang and X. Liu. *A re-examination of text categorization methods*. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 42–49, Berkley, 1999.

[Yaworsky, 1993] D. Yarowsky. *One sense per collocation*. In Proceedings of the workshop on Human Language Technology, pp. 266-271, Princeton, New Jersey, 1993.

[Yuan et al., 2016] D. Yuan, J. Richardson, R. Doherty, C. Evans and E. Altendorf. *Semi-supervised word sense disambiguation with neural models*. In Proceedings of COLING, pp. 1374–1385, 2016.

[Zhai et al., 2001] C. Zhai and J. Lafferty. *A study of smoothing methods for language models applied to ad-hoc information retrieval*. In Proceedings of the 24th annual international ACM/SIGIR conference on research and development in information retrieval, pp. 334–342, New Orleans, Louisiana, USA, 2001.

[Zhang et al., 2011] L. Zhang, C. Li, J. Liu and H. Wang. *Graph-Based Text Similarity Measurement by Exploiting Wikipedia as Background Knowledge*. International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol.5, Issue 11, pp. 1328–1333, 2011.

[Zweigenbaum et al, 2003] P. Zweigenbaum, R. Baud, A. Burgun, F. Namer, E. Jarrousse, N. Grabar, P. Ruch, F. Le Duff, B. Thirion and S. Darmoni. *UMLF: construction d'un lexique médical francophone unifié*. In Actes des 10 Journées Francophones d'Informatique Médicale, Tunis, 2003, France. 2003.