ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

RAPPORT DE PROJET PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA MAITRISE
EN GÉNIE LOGICIEL

PAR
OULAIDI M'hammed

DÉVELOPPEMENT D'UNE COUCHE INTELLIGENTE POUR UN MOTEUR DE RECHERCHE SÉMANTIQUE À L'AIDE DU TOPIC MODELING

MONTREAL, LE 7 AVRIL 2016

PRÉSENTATION DU JURY

CE RAPPORT DE PROJET OU MÉMOIRE OU THÈSE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

Professeur Alain April, directeur de projet
Département de génie logiciel & des technologies de l'information
École de technologie supérieure

M. Benoit Des Ligneris, codirecteur de projet
Chef de croissance à Subscribility.

Professeur Alain Abran, jury
Département de génie logiciel et des technologies de l'information
École de technologie supérieure

REMERCIEMENTS

Je tiens à remercier toutes les personnes qui ont contribué à la réussite de ce projet et particulièrement à l'entreprise BiblioMondo, pour sa collaboration lors des expérimentations. Grâce à leur confiance j'ai pu accomplir cette recherche dans les meilleures conditions.

Je tiens à exprimer ma reconnaissance et ma gratitude à mon professeur et directeur de projet, le professeur **Alain April** qui m'a fait confiance et qui m'a encouragé tout au long de ce projet. Grâce à ses directives et ses conseils, j'ai pu relever le défi. Doté de grande chaleur humaine ainsi qu'un grand savoir-faire, il m'a poussé à être au niveau des attentes du partenaire industriel. J'espère que ces mots seront témoin de mon grand respect et ma reconnaissance la plus profonde.

Mes remerciements sont aussi adressés à monsieur **Apollinaire Adembega**, directeur recherche et développement, et à monsieur **Benoit Des Ligneris**, chef de croissance chez Subscribility, pour leurs précieuses conseils, ainsi que leurs disponibilités tout au long de ce projet.

J'aimerais rendre hommage à ma famille dont l'encouragement, le soutien et l'amour immense m'ont été d'une grande aide tout au long de mes études.

DÉVELOPPEMENT D'UNE COUCHE INTELLIGENTE POUR UN MOTEUR DE RECHERCHE SÉMANTIQUE UTILISANT LE TOPIC MODELING

OULAIDI M'hammed

RÉSUMÉ

De nos jours le Web sémantique prend de l'essor et vise à structurer les données ainsi que de permettre la synchronisation avec les nouvelles données diffusées sur le Web. Cette structuration permettra une meilleure compréhension du contenu par les humains ainsi que par les machines. Les bibliothèques numériques centralisées gèrent une quantité croissante de données issues de centaines de sources et désirent maintenant intégrer des technologies intelligentes qui pourraient gérer les recherches, en langage naturel, et apporter des réponses complètes et immédiates aux requêtes de ses utilisateurs. Elles cherchent à favoriser le développement de nouvelles formes d'intelligence collective. Cette recherche vise à combiner deux technologies existantes de l'intelligence artificielle : 1) le forage de textes (qui vise à comprendre le sens des requêtes mises en langage naturel et les classer dans leurs dimensions sémantiques) et la classification automatique des documents d'une manière qui facilitera l'accès aux données en les hiérarchisant.

Ce rapport de projet de maîtrise appliquée, de 15 crédits, décrit une étude de spécification et des exigences de cette problématique, ainsi que la conception et l'expérimentation d'une solution qui utilise divers algorithmes. Cette proposition de solution vise la mise en place d'un thésaurus multilingue qui a pour objectif de se synchroniser avec différents modules existants d'une bibliothèque, dont un Backend, qui contient déjà plusieurs ontologies et thésaurus multilingues gérés par un logiciel open source Ginco. La solution est conçue en couches et vise à optimiser le traitement des données lors d'une requête en augmentant la pertinence des résultats, ainsi que la recommandation d'items et documents plus pertinents.

Mots clés : Thésaurus multilingues, Web sémantique, forage de textes, intelligence artificielle, Ginco.

DEVELOPMENT OF INTELLIGENT LAYER FOR A SEMANTIC SEARCH ENGINE USING THE TOPIC MODELING

OULAIDI M'hammed

ABSTRACT

Nowadays, the Semantic Web is getting more attention and aims to structure the data and allow synchronization with the new data published on the Web. This technology will enable a better understanding of the content by humans and also by machines.

Centralized electronic libraries manage an increasing amount of multimedia data from hundreds of sources and now wish to integrate smarter technologies that could enhance the precision of their search engines, using natural language. They seek to promote the development of new forms of collective intelligence by: 1) text mining (which aims at understanding the meaning of queries made in natural language and classify them in their semantic dimension); and 2) data mine using an automatic classification of the documents in a manner that will facilitate access to data.

This 15 credits applied research master degree report provides a description of the business requirements followed by the design, implementation and testing of a proposed solution (i.e. an intelligent layer/service between the existing front-end and back-end). It proposes a multilingual thesaurus that contains several ontologies implanted using Ginco. This solution will improve the accuracy of the query results of the electronic library and, consequently, recommend the right products and results to its users.

Keywords: Multilingual thesaurus semantic web, text mining, artificial intelligence, Ginco.

Table de matière

LISTE DES TABLEAUX

Page

Liste Des figures

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

CMS	Content Management System
GINCO	Gestion Informatisée de Nomenclatures Collaboratives et Ouvertes
SRCT	Méthode pour les relations sémantiques
CLOPE	Clustering Algorithm for Transactional Data
MeSH	Medical Subject Heading
LA COUCHE IA	Couche d'intelligence artificielle
MVC	Modèle-vue-contrôleur
TF-IDF	Term frequency-inverse document frequency
AI	Intelligence artificielle
NSG	Near synonym graph
API	Interface de programmation applicative
SOCOM	Semantic oriented cross lingual ontology mapping
LSA	Latent semantic analysis
RDF	Ressource Description Framework
SKOS	Simple Knowledge Organization System
XML	Extensible Markup Language

SOAP	Simple Object Access Protocol
VIAF	Virtual International Authority file
CGI	Interface de passerelle commune
NLTK	Natural language toolkit
NORP	Nationalities or religious or political groups
GPE	Geo-political Entity
NLP	Natural Linguistic Programming
LDA	Latent dirichlet allocation

INTRODUCTION

A. Mise en contexte

L'évolution de l'internet a incité les chercheurs à créer de nouvelles solutions qui pourront soutenir et utiliser intelligemment la grande quantité de données disponibles sur le web. Parmi les sujets émergents du domaine des TI's, en ce moment, le Big Data, l'infonuagique et le web sémantique sont très populaires [20].

Une problématique grandissante est de pouvoir manipuler et de faire du sens de cette grande quantité de données. Conséquemment, la recherche dans le domaine du Web sémantique, émerge avec de nouvelles propositions de solutions, des nouvelles techniques et de nouvelles propositions de normes. Un des axes de recherche, plus pertinents pour ce projet, vise l'extraction d'une grande quantité de données et la conception d'une structure de données qui s'assure de capturer le sens des données en s'appuyant sur la notion d'ontologie. Par définition, une ontologie est un modèle de données des connaissances, c'est-à-dire une représentation des diverses connaissances pour qu'elles soient utilisables par les ordinateurs, qui peut être utilisée afin de construire une dimension sémantique de ces connaissances (Trojahn, 2014). L'objectif de ce projet de recherche appliquée, vise à hiérarchiser les données existantes par thèmes, c'est-à-dire implanter la notion de « Topic Modeling » dans le but d'obtenir des résultats de requêtes plus précises. Pour ce faire, l'extraction des connaissances des textes (c.-à-d. le data mining) sera effectué en utilisant plusieurs algorithmes. De plus, des techniques de forage de données seront aussi utilisées afin de structurer les données, d'une manière optimale, et les hiérarchiser (c.-à-d. regrouper en « clusters ») afin d'accéder facilement aux résultats escomptés et ce très rapidement. Pour ce faire, il a été nécessaire d'étudier les propositions existantes, les problématiques, ainsi que de concevoir un prototype de solution, qui vise à intégrer le logiciel résultant facilement dans les logiciels de requêtes existants d'une bibliothèque électronique. BiblioMondo est le leader mondial du domaine des logiciels de bibliothèques numérique francophone.

L'expérimentation de la solution proposée sera effectuée sur son logiciel de bibliothèque électronique qui traite une très grandes quantités de données et de documents numériques.

B. Objectifs du projet et réalisations :

Bibliomondo, une société Québécoise, s'intéresse aux nouvelles techniques du web sémantique afin d'améliorer l'efficacité des requêtes de ses logiciels de gestion de bibliothèques. La croissance de l'entreprise et l'acquisition de nouveaux clients posent de nouveaux défis pour la société. D'une part elle devait gérer des documents de manière efficace, et ce sans dégrader la qualité de son service. D'autre part elle désire améliorer son moteur de recherche pour qu'il soit mieux adapté aux domaines d'affaires et aux langues choisis par les utilisateurs. Cette clientèle est localisée dans nombreux pays, c'est-à-dire au Québec, en Allemagne, en Espagne, en France, et même en Angleterre, etc. Le multilinguisme représente un défi de taille pour cette société Montréalaise. En effet afin de développer un système de requêtes plus performant, son logiciel doit effectuer les fonctionnalités et les traitements d'une requête en répliquant les requêtes dans toutes les langues. Actuellement, il n'est pas possible de procéder à la traduction automatique des termes d'une requête, en temps réel, car le sens des mots change d'une langue à une autre.

Une analyse approfondie de l'architecture du système de requête existant, de la société, a donc été nécessaire afin d'identifier les solutions possibles à cette problématique. L'architecture du système actuel, a fait ressortir que la solution utilise des technologies Big Data tels que l'indexation à l'aide du logiciel Solr [21] qui permet de structurer une grande quantité de données. Actuellement, les recherches sur les données, se font d'une manière connue, qui est la méthode syntaxique (Edward Gibson, 2013).

Au moment de débuter cette recherche appliquée, les résultats de l'exécution du moteur de recherche, dans la librairie électronique, ne sont pas semblables quand on utilise une langue ou une autre. Il serait donc intéressant d'ajouter une capacité sémantique multilingue au moteur de recherche existant. Pour ce faire il est nécessaire d'ajouter un thésaurus accompagné d'ontologies multilingues (Leyla Zhuhadar, 2015). Étant donné que l'entreprise

désire avoir une solution qui sera conçue à l'aide de logiciels libres, le logiciel Ginco [17] a été choisi par les experts en place et il sera hébergé sur son nuage privé.

Le plan de travail est donc, dans une première étape, d'effectuer une revue littéraire, suivie d'une analyse critique de diverses solutions déjà publiées (c.-à-d. état de l'art) portant sur la gestion des thésaurus multilingues ainsi que leur intégration dans un moteur de recherche. Par la suite, suite à l'identification d'une approche prometteuse, l'objectif principal de la prochaine étape est de développer une preuve de concept d'un thésaurus multilingue qui pourra s'insérer entre le front-end et le back-end du moteur de recherche existant de la société. L'objectif de cette approche est qu'il y ait le moins d'impact possible sur l'architecture logicielle du moteur de recherche existant. Une autre exigence de cette preuve de concept est que la solution proposée soit facilement accessible à l'aide d'un service web utilisant des technologies Java.

C. Organisation du rapport du projet :

Le chapitre 1, de ce rapport, présente la revue de huit publications qui ont été retenues et qui décrivent différentes approches de mise en œuvre des thésaurus multilingues, ainsi que les algorithmes utilisés pour les gérer. Le chapitre 2 présente les spécifications du prototype visé par la preuve de concept, le but et l'objectif du projet, l'architecture du prototype expérimental conçu ainsi que les parties développés, testées et expérimentées. Finalement, le chapitre 3 décrit les étapes de développement du prototype expérimental en expliquant : (1) l'analyse de la situation actuelle; et (2) le choix de l'architecture de la solution proposée.

CHAPITRE 1

Revue de la littérature

Ce chapitre présente la revue littéraire concernant les thesaurus multilingues ainsi que les différents algorithmes permettant la gestion du multilinguisme. Cette section fait la synthèse de huit articles retenus accompagnés d'une analyse critique de la méthode proposée. Il se termine par la présentation d'un tableau récapitulatif des approches présentées.

Un thesaurus est une liste de termes normalisée et contrôlée afin de représenter les concepts d'un domaine de connaissances, les termes, qu'il inclut, sont reliés par des relations sémantiques (c.-à-d. synonymies, homonymies et polysémies).

1.1 Article 1 : État de l'art des ontologies multilingue et monologue et leurs alignements

Les auteurs de cet article (Trojahn, 2014) présentent plusieurs approches d'indexation d'ontologies multilingues ainsi qu'une évaluation de chacune de ces approches. Avant de débuter l'analyse de cet article, il est utile de rappeler la signification d'ontologie et son rapport avec le domaine du web sémantique pour les lecteurs. L'ontologie est un ensemble structuré qui représente des données ou des éléments d'un domaine de connaissances précis [22]. C'est donc un modèle qui représente un ensemble de concepts ainsi que les relations entre ces concepts. Les concepts sont organisés, à l'aide de graphes de concepts, possédant des relations sémantiques (aussi nommé subsomption). Ces relations décrivent généralement, des individus, des classes, des attributs, des relations, des évènements et des métaclasses. Il y a plusieurs approches pour mettre en œuvre une ontologie multilingue. En voici quelques exemples:

- Alignement des ontologies monolingue : cette approche consiste à traduire manuellement une ontologie d'une langue à une autre langue. C'est le cas du

thesaurus Agrovoc (Liang, 2006) qui a été aligné avec le thesaurus d'agriculture chinois. La limite de cette approche est très coûteuse ainsi qu'il est difficile à l'adopter quand il s'agit d'ontologies complexes à traduire;

- Approche basée sur corpus : cette approche est utilisée pour aligner le thésaurus chinois HowNet à un thesaurus anglais WordNet [23] en se basant sur un corpus bilingue d'un domaine similaire aux deux ontologies. Autrement dit l'alignement est le fait qu'on va avoir un thesaurus qui supportera les deux langues sans qu'on change de sens. Cette méthode s'appuie sur le calcul de la fréquence de la similitude des relations entre les deux concepts avec le corpus bilingue. Pourtant, il est possible que le corpus ne soit pas disponible pour le domaine spécifique des deux ontologies impliquées;
- Corpus et base de connaissances : Cette approche utilise un corpus générique multilingue, à grande échelle, afin d'établir des relations sémantiques entre deux ontologies monolingues;
- Approche d`alignement par traduction automatique: Cette approche consiste une traduction automatique des ontologies. Le SOCOM (Semantic Oriented Cross Lingual Ontology Mapping) Framework permet à l'utilisateur de contribuer à une meilleure performance de traduction en permettant de choisir la traduction appropriée ainsi que d'améliorer la relation entre les deux ontologies. Pourtant, dans le cas d'une traduction erronée, cela peut générer de mauvaises relations sémantiques et de correspondance relationnelle dans les autres langues;
- Approche d`apprentissage machine : Pour utiliser cette approche, il est nécessaire de concevoir un alignement entre les concepts manuellement, de manière à obtenir une adaptation multilingue. L'approche propose la composition indirecte en se basant sur l'alignement entre les concepts déjà existants, c'est à dire pour aligner une ontologie française et portugaise, il suffit d'utiliser un alignement déjà existant entre l'une de ces deux avec une ontologie en anglais. Certes; il reste que les concepts alignés manuellement ne peuvent pas être disponible;

- Approche basée sur la similarité d`images: Afin d'évaluer la similitude, un calcul de la similitude des images associées à l'entité est calculé. Par exemple (river, Rio Grande). Mais cette approche est très coûteuse au niveau du calcul et de ressources informatiques requises.

1.2 Article 2: Une approche basée sur la construction automatique d'un thésaurus utilisant la « clusterisation » et l'analyse d'un dictionnaire

Problématique: La construction manuelle d'un thésaurus améliore sa qualité, mais cela est très coûteux. Pour se faire, un regroupement de plusieurs données, c'est-à-dire, des documents, des articles, des livres et bien d'autres documents par un expert du domaine (du thesaurus) est effectuée et, il analyse manuellement ces données. L'expert choisit, à chaque fois, les termes pertinents et les regroupent dans des « clusters ». Ensuite il définit les relations hiérarchique et associative entre ces données. Ceci demande beaucoup d'effort et il devient vite nécessaire de trouver une manière qui permet d'automatiser ce processus de construction de thésaurus afin d'être plus efficace et ce, sans qu'il y ait un impact sur la qualité résultante.

Solution proposée: Dans cet article (Lagutina, 2015), les auteurs présentent une approche semi-automatique pour la construction d'un thesaurus pour le domaine de la cardiologie en utilisant un dictionnaire monolingue de ce domaine à titre de corpus initial. Le processus débute par l'extraction du corpus (c.-à-d. dans le format d'un fichier texte), qui suit un format spécifique comportant trois items : (c.-à-d. Term, Headword et Description). Ensuite, une génération des termes candidats, qui vont être utilisées dans ce thésaurus, est faite à l'aide d'un algorithme automatique proposé par les auteurs. Cet algorithme prend, comme entrée, de 10 à 15 mots clés et morphèmes, puis calcule la fréquence des mots qui contiennent ce morphème ou les synonymes des mots clés. Par exemple : myocarduim, pericarduim, endocarditis.. (Morpheme : CARD). Suite à la génération des termes candidats, la prochaine étape vise la classification (c.-à-d. le « clustering ») à l'aide des liens sémantiques qui les regroupent. Les auteurs prétendent que la technique du calcul de la fréquence d'occurrence du mot est inappropriés (TF-IDF) pour ce cas. En revanche, les auteurs proposent un nouvel

algorithme nommé CLOPE. (C.-à-d. l'algorithme de construction de thesaurus dans des clusters).

L'algorithme de CLOPE est utilisé pour la classification automatique, il a été conçu afin de permettre la minimisation de similarité entre les classes extraites et maximiser la similarité entre les mots de chaque classes, alors pour un document **D** et un mot **W** et une classe **C**, le calcul arithmétique proposé par l'auteur permettra le calcul du Profit **P**. Ce dernier détermine la probabilité que le mot **W** du document **D** peut appartenir au cluster **C** , donc le maximum de profit de chaque mot pourra définir à quelle cluster le mot appartient

Suite à l'utilisation de cet algorithme, lors d'une étude de cas, les résultats ont été jugés très satisfaisants par les auteurs. Cependant, CLOPE vise à mettre les termes candidats dans un « cluster » ayant une dimension sémantique partagée avec un ***R*** comme variante et qui dépend de l'exactitude voulue du regroupement. Toutefois, le choix du ***R*** reste ambigu. Dans le cas qui nous concerne, c'est-à-dire la mise en place d'un thésaurus multilingue, un morphème comme CARD n'a pas le même sens dans les autres langues et l'algorithme de génération de termes candidats proposé ici ne permet pas sa prise en considération.

1.3 Article 3 : Améliorer l'interfonctionnement sémantique multilingue dans les systèmes d'entreprises grâce au concept de désambigüisation

Problématique : Les systèmes informatiques sont devenus de plus en plus nombreux ainsi, les échanges de données entre ces systèmes ont démontré qu'il y a des problèmes dans le sens des données lors des échanges entre eux.

Solution *:* Les auteurs de cet article (Jingzhi, 2012) proposent une approche nommée : ***«*** synonyme-dictionnary driven », afin de maintenir une bonne consistance des relations sémantiques entre des données de toutes sortes (c.-à-d. des articles, des texte…) et multilingues de manière automatique. Il ont utilisé un dictionnaire anglais-chinois à titre de corpus initial, et concernant le dictionnaire de la langue locale (c.-à-d. qui est anglaise), ils

ont utilisé une base de donnée lexicale (c.-à-d. une combinaison de dictionnaire et de thésaurus), WordNET, comme référence externe afin d'avoir plus de précision sur les mots anglais utilisés dans une requête. Le terme ***synset*** (c.-à-d. synonym set) est souvent utilisé dans cet article. Le ***synset*** permet de définir une dimension sémantique d'un terme, par exemple :

- car, auto, automobile, machine, motorcar -- (4-wheeled motor vehicle; usually propelled by an internal combustion engine; he needs a car to get to work) ;
- car, railcar, railway car, railroad car -- (a wheeled vehicle adapted to the rails of railroad; three cars had jumped the rails).

Chaque ***synset*** dénote une acception différente du mot car, décrite par une courte définition.

Description et processus de la solution : Cet article présente un cadriciel, nommé NSG (Near Synonym Graph), qui inclue trois processus et qui se base d'une part sur un dictionnaire anglais-chinois, pour le corpus initial, et d'autre part sur un dictionnaire de la langue locale, qui est l'anglais. Il utilise WordNet à titre de corpus initial parce que c'est l'un des plus grands dictionnaires multilingues au monde, ce qui assure une grande variété de données à savoir :

1. Utiliser WordNet comme un dictionnaire de synonymes en utilisant ses ***synsets***;
2. Mettre en place un TAG (par ex. : T1-Car, T2-Car … etc.) pour chaque ***synset***;
3. Utiliser un dictionnaire anglais-chinois (la base de données) comme un vocabulaire initial et qui permet de chercher des synonymes proches des ***synsets*** extraits de WordNet;
4. Mettre en place un TAG pour chaque mot dans la base de données (c.-à-d. il peut y avoir plusieurs TAG en vue d'avoir un ***synset***);
5. Produire un graphe qui relie sémantiquement les ***synsets*** à la base de données qui décrit une liste de synonyme potentiels du mot d'une requête;
6. Définir un et un seul ***synset*** pour chaque ***synset*** qui le relie sémantiquement entre la base de données et WordNet.

En somme, cette solution nécessite deux étapes :

- ✓ La 1ére étape vise l'extraction des ***synsets*** potentiels, et
- ✓ La 2éme étape vise l'identification d'un seul ***synset*** résultant.

Par la suite, un modèle probabiliste calcule l'ensemble le plus près qui peut contenir le synonyme du mot recherché. Par la suite il identifie le synonyme, à L'aide d'un dictionnaire multilingue (c.-à-d. une base de données) anglais–chinois ainsi que le dictionnaire de synonymes anglais WordNet. Cette méthode, qui permet de calculer des relations sémantiques, se nomme le SRCT (Sparse Sepresentation-based Classification Task)

- ✓ **La première étape : Extraction des *synsets* potentiels.**

Le premier défi de cette approche est de calculer le plus près ensemble potentiel qui contient le synonyme. Par exemple les mots qui sont apparus dans WordNet et dans la base de données anglais-chinois sont regroupés dans 22 catégories. Les catégories intéressantes ici sont celles où il y a une grande fréquence que le ***synsets*** s'y retrouve (c.-à-d. celles identifiées par l'algorithme SRCT). Pour relier les ***synsets*** entre WordNet et la base de données, soit le ***synset*** existe dans les deux dictionnaires, soit qu'un ***synset*** dans WordNet contient un mot qui a un synonyme dans la base de données et, ce mot existe dans un ***synset*** dans la base de donnée. Dans ce cas un calcul de similitude se fait à l'aide de l'algorithme SRCT.

- ✓ **La deuxième étape : Identification du *synset*.**

L'identification est effectuée en précisant le poids du mot dans le concept pour chaque « cluster » potentiel, et en choisissant le maximum de ces valeurs pour identifier le synonyme au cas où on a deux valeurs sont égales (c.-à-d. que le mot est polysémique).

Le score SRCT peut être calculé de deux manières :

a. Calculer la fréquence de la relation sémantique entre deux bouts de texte entre WN et BD en utilisant les hyperonymes (« église » possède deux hyperonymes, « bâtiment » et « lieu sacré ») et les hyponymes(« haut-de-forme » est un

hyponyme de « chapeau » et « chapeau » est un hyponyme de « coiffure » et de « couvre-chef ») de WordNet comme ressource initiale

b. Prendre un texte avec les mots figurant dans WordNet et un autre texte avec les mots figurant dans la BD Anglais-Chinois et calculer la similitude entre les deux et définir les synonymes.

1.4 Article 4: Une stratégie synergique pour combiner les approches fondées sur un corpus et à base de thésaurus dans la construction d'une ontologie pour les moteurs de recherche multilingues

Problématique : La conception d'un moteur de recherche, pour des données multilingues, est complexe car l'utilisateur doit avoir des résultats appropriés dans la langue utilisée pour sa recherche, sans avoir à lui demander sa préférence linguistique (c.-à-d. découvrir automatiquement la langue utilisée).

Solutions : Les auteurs de cet article (Zhuhadar, 2015) ont développé un moteur de recherche multilingue en se basant sur deux approches : **corpus-based & thesaurus based**, pour que les résultats d'une recherche soient plus précis à l'aide de plusieurs informations. Avant de pouvoir utiliser ces deux approches, les auteurs ont dû développer un système de recommandation de termes. Ce système utilise des techniques d'apprentissage machine ainsi que le calcul de la similitude des préférences d'un utilisateur (c.-à-d. comparer aux autres utilisateurs). Par exemple, les préférences d'un utilisateur sont souvent représentées par un vecteur de choix. Le calcul du cosinus de deux vecteurs sert à déduire l'angle de similitude :

- Approche basée sur le thésaurus : Pour cette approche, les auteurs ont choisi un thesaurus bilingue (anglais – espagnol) pour distinguer les concepts et les sous-concepts des deux langues. Dans le cas où un utilisateur effectue une recherche, le thesaurus vise à lui fournir d'autres résultats reliés, car à travers ce thesaurus il y a une détection des concepts reliés au terme recherché;
- Approche basée sur corpus : C'est l'utilisation de la traduction des requêtes (d'anglais vers espagnol ou l'inverse). En effet le modèle « espace vecteur » est utilisé pour

calculer le produit scalaire entre la requête traduite et les documents qui se trouvent dans la base de données. Pour effectuer ce calcul, l'algorithme TF-IDF été mis en place.

Le TF-IDF (de l'anglais Term Frequency-Inverse Document Frequency) est une méthode de pondération souvent utilisée en recherche d'information et en particulier dans la fouille de textes. Cette mesure statistique permet d'évaluer l'importance d'un terme contenu dans un document, relativement à une collection ou un corpus.

1.5 Article 5 : Gestion automatisée des thesaurus

Problématique : Il est souvent difficile de mettre en œuvre des relations sémantiques entre plusieurs thésaurus multilingues car le système doit être assez intelligent afin de comprendre chaque terme d'une requête, dans son contexte, et chercher son synonyme, ce qui peut devenir complexe dans de très grandes quantités de données.

Solutions proposées : Dans cet article (Petraki, 2015), les auteurs proposent un modèle intelligent basé sur trois algorithmes qui permettent d'indexer plusieurs thesaurus multilingues. Ils utilisent aussi une base de données centrale qui s'occupe d'un changement ou d'une modification à un terme. Cette approche, le FDB (Frame Object Database), ne permet pas d'impacter le schéma, les relations ou la couche logique du modèle :

Solutions proposées : Dans cet article (Petraki, 2015), les auteurs proposent un modèle intelligent basé sur trois algorithmes qui permettent d'indexer plusieurs thesaurus multilingues. Ils utilisent aussi une base de données centrale qui s'occupe d'un changement ou d'une modification à un terme. Cette approche, le FDB (Frame Object Database), ne permet pas d'impacter le schéma, les relations ou la couche logique du modèle :

- Le premier algorithme : commence à cherche le mot (**w**). S'il correspond à un terme dans un thésaurus, il ajoute une relation avec un « tag » qui le relie avec une description existante dans la base de données principale. Si le mot n'existe pas, c'est-à-dire qu'il n'est pas un terme dans le thesaurus, il cherche un synonyme de (**w**) dans le dictionnaire

puis il vérifie s'il existe dans le thésaurus. Donc si (**w**) est corrélé avec le terme connexe, il est ajouté dans le thésaurus;

- Deuxième algorithme : L'algorithme cherche si les termes dérivés du mot (**w**) d'un tag spécifique existent dans les termes du thesaurus. Si ce mot existe alors la recherche de tous les synonymes du mot (**w**) est effectuée puis ils sont ajoutés au thesaurus avec les mêmes relations sémantiques. Au cas où le mot (**w**) n'existe pas dans le thesaurus, une recherche d'un synonyme du mot dans le dictionnaire va être lancée jusqu'à ce qu'il soit trouvé, puis le mot (**w**) est ajouté dans le thesaurus avec les mêmes relations sémantiques du synonyme mais indiqué comme un terme apparenté;
- Troisième algorithme : Cet algorithme procède à créer un thésaurus d'un domaine spécifique de manière automatique dans la base de données stockée dans le modèle FDB (articles, documents). Cet algorithme n'est pas très performant et requiert l'intervention manuelle d'un expert pour être validé. Les auteurs ont utilisé l'approche corpus.

La problématique de cette proposition est que ces algorithmes sont complexes à calculer et il y a une intervention manuelle.

1.6 Article 6 : Programme shell basé sur l'approche d'ontologie visant à construire et gérer un thesaurus multilingue pour un domaine spécifique

Problématique : La construction d'un thésaurus multilingue, sa mise en coordination des termes et les relier avec des liens sémantiques (c.-à-d. synonyme, patronyme, hyperonymie) dans différentes langues et ce d'une manière automatique reste un grand défi actuellement.

Solutions proposées : Dans cet article (Zagorulko, 2013), les auteurs ont développé une plateforme de création de thésaurus multilingue pour la langue Russe. Cette plateforme vise la facilité d'utilisation par les experts du domaine, sans aide ou formation requise en TI. Cette proposition décrit en détail l'architecture de la solution expérimentée afin de supporter le multilinguisme lors de la construction d'un thesaurus. Nous avons vu qu'un thesaurus comprend des termes, la source de termes (c.-à-d. documents textes, articles, etc…) ainsi que

des connaissances reliées aux termes. Cette proposition de plateforme logicielle permet d'extraire tous les termes potentiels d'un corpus à l'aide de sa fréquence d'occurrence. Il peut aussi être modifié par un expert, manuellement, au cas où il y aurait une incohérence sémantique ou des problèmes de relations entre les concepts. Les chercheurs ont déployé une traduction automatique du terme recherché de la langue anglaise vers la langue russe lors d'une recherche multilingue. Les résultats de la requête sont affichés pour une langue spécifique ou pour les deux langues. La valeur ajoutée de cette approche est qu'ils ont implanté deux types de termes descripteurs (c.-à-d. le préféré) et non-descripteurs (c.-à-d. le non préféré). Tous les termes descripteurs sont reliés avec leurs équivalents dans l'autre langue et leurs synonymes non descripteur avec leurs définitions.

1.7 Article 7 : Représentation du savoir pour document classification transductive et multilingue

Suite à la surcharge des données multilingues sur internet, dans cet article (Salvatore, 2015), les auteurs exploitent BabelNET comme base de connaissances. BabelNET utilise les capacités ontologiques de WordNet jumelé avec le pouvoir encyclopédique de Wikipédia. Les chercheurs proposent d'utiliser l'approche d'apprentissage transductive. Cette approche se base sur la technique BOS (Bag Of Synsets (c.-à-d. synonym-set). Afin d'intégrer de nouveaux documents, utilisent deux phases : pour la première phase, il est nécessaire d'extraire tous les termes utilisés et, par la suite, ils sont regroupés dans des suites de « tags ». La technique de fréquence d'occurrence des mots est utilisée (c.-à-d. l'algorithme). Pour la deuxième phase, chaque paire de termes accompagnés de son « tag » est extraite à l'aide d'un algorithme qui les relie avec le sens le plus approprié recommandé par BabelNET. Pour résoudre le problème de la traduction en plusieurs langues, en minimisant les problèmes sémantiques, enfin ils traduisent tous les documents dans une langue commune afin qu'elle soit utilisée pour effectuer une corrélation des résultats.

1.8 Article 8: extension des requêtes multilingues dans la base de données bibliographiques SveMed (Une étude de cas)

Les auteurs de cet article (Gavel, 2014) ont développé une base de données bibliographique du domaine médical nommé SveMed+. Son interface utilisateur est basée sur un moteur de recherche qui utilise la technologie Big Data Solr [21] afin d'indexer les données, dans une base de données qui contient un thésaurus multilingue. Cette base de donnée se nomme MEsh, (c.-à-d. un thesaurus crée par la librairie nationale de médecine des États-Unis) et dans une base de données bibliographique du domaine, PUBMED/MEDLINE [28]

Lors d'une requête émise par un utilisateur en langue naturelle, l'algorithme cherche un terme similaire, dans le thésaurus, en utilisant l'algorithme TF–IDF. Par la suite, il associe les termes qui ont une relation avec la requête du client Solr, ce qui permet d'avoir accès à d'autres informations telles que : des auteurs qui ont écrit sur le même sujet, un synonyme potentiel, etc.... Ensuite, les résultats qui sont obtenus de Solr sont triés selon la date d'entrée, et les facettes sont regroupées dans des « clusters ». Ce regroupement est effectué à l'aide de l'algorithme « query expansion », c'est-à-dire il se concentre sur le « mapping » des données du langage naturel émis par un utilisateur avec les termes existant dans le thésaurus et ainsi permettre à l'utilisateur d'extraire les données. La reconnaissance de la langue naturelle émise par l'utilisateur et la traduction de la requête, dans la langue ou les langues que supporte le thesaurus, représentent des limites importantes de cette approche. En fait le soutient du multilinguisme et de l'indexation pour le multilingue nécessitent l'implication manuelle d'un expert du domaine ce qui est coûteux.

1.9 Récapitulatif des approches publiées:

Tableau 1.1 : Sommaire des approches utilisées pour chaque article

Articles	Corpus-based	Clusterization	Semi-automatique construction	synonym-dictionnary driven	Linked-Data	Apprentissage Transductive	SRCT	TF-IDF
2		X						
3				X			X	
4	X							X
5	X				X			
6					X			X
7						X		
8								X

Ce chapitre a présenté l'état de l'art du domaine. L'analyse des articles a permis de faire l'inventaire de différentes approches et algorithmes utilisés pour construire et gérer des thésaurus multilingues. Il a été possible de faire l'analyse critique et de comprendre les limites de chaque approche. Il y a donc, une opportunité pour améliorer la situation en proposant une approche qui influerait :

- la classification;
- l'automatisation des relations sémantiques; et
- La détection automatique de la langue naturelle et son analyse.

Un tableau récapitulatif des approches utilisées, par chaque article est présenté. Dans le prochain chapitre, la problématique du projet sera présentée ainsi que les objectifs du projet et ses limites. Finalement il sera possible de conclure et présenter la méthodologie suivie par le projet.

CHAPITRE 2

Problématique du projet

Ce chapitre décrit la problématique que le projet résous. La première section présente le but ainsi que l'objectif du projet. La deuxième section discute des limites du projet. La troisième section décrit le processus suivi pour réaliser ce projet de recherche appliquée et présente les résultats. Finalement, la quatrième section présentera une synthèse.

2.1 Le but du projet :

Le but de ce projet de recherche appliquée vise le développement d'un moteur de recherche sémantique multilingue en se basant sur divers thésaurus et ontologies à titre de base de connaissance ainsi que la structuration de documents par thème. Son objectif principal est de modéliser une architecture d'une couche middleware, qui s'insère entre le back-end et le front-end actuel du système existant de la société BiblioMondo. Cette nouvelle couche de service devra être développée sous forme d'un pipeline de modules qui permettent de traiter les requêtes, en langage naturel, avec une structuration de données par thèmes en utilisant l'algorithme LDA (Latent Dirichlet Allocation)

2.2 Limites du projet :

La couche intelligente qui sera développée ne pourra pas effectuer une recherche syntaxique au cas où la recherche sémantique ne parvient pas à remettre des résultats. De plus, les données utilisées pour tester la couche seront libre (c.-à-d. des données publiques). Ce qui veut dire que le prototype expérimental ne sera pas testé sur le système actuel de l'entreprise BiblioMondo dans un premier temps. Les tests d'acceptations des clients finaux ainsi que la mise en production ne seront pas inclus dans ce projet de recherche appliquée.

2.3 Méthodologie du projet

La méthodologie du projet (voir figure 2.1) a été conçue pour atteindre le but du projet et comprend, les étapes ci-dessous :

1. Élaboration de l'état d'art de l'art qui porte sur les différentes approches pour construire et gérer des thésaurus multilingues aussi l'utilisation de ces derniers pour avoir un moteur de recherche sémantique en plus des algorithmes de structuration des données par thèmes (chapitre 1);
2. Définition de la problématique du projet et de la méthodologie à suivre afin de réaliser le projet (détails dans le chapitre 2);
3. Développement des modules de la couche IA en choisissant les divers algorithmes utilisés. Cette étape est composée des étapes suivantes :
    a. Étude du système actuel, cette phase consiste à analyser les différents algorithmes et modules de recherche que l'entreprise utilise ainsi sa méthode de structuration de données;
    b. Modélisation de la couche Semantic IA en s'inspirant des limites projetées dans notre revue littéraire et les problématiques que l'entreprise veut résoudre;
    c. Développement des modules de la couche Semantic IA;
    d. Détection de la langue de la requête mise par l'utilisateur par le détecteur NPL;
    e. Décomposer la requête sous forme de triples (sujet -verbe -complément) par le décomposeur sémantique des requêtes;
    f. Traduction par machine, pour traduire la requête décomposée vers la langue anglaise;
    g. Modélisation thématique des données pour les structurer par thème et les hiérarchiser.

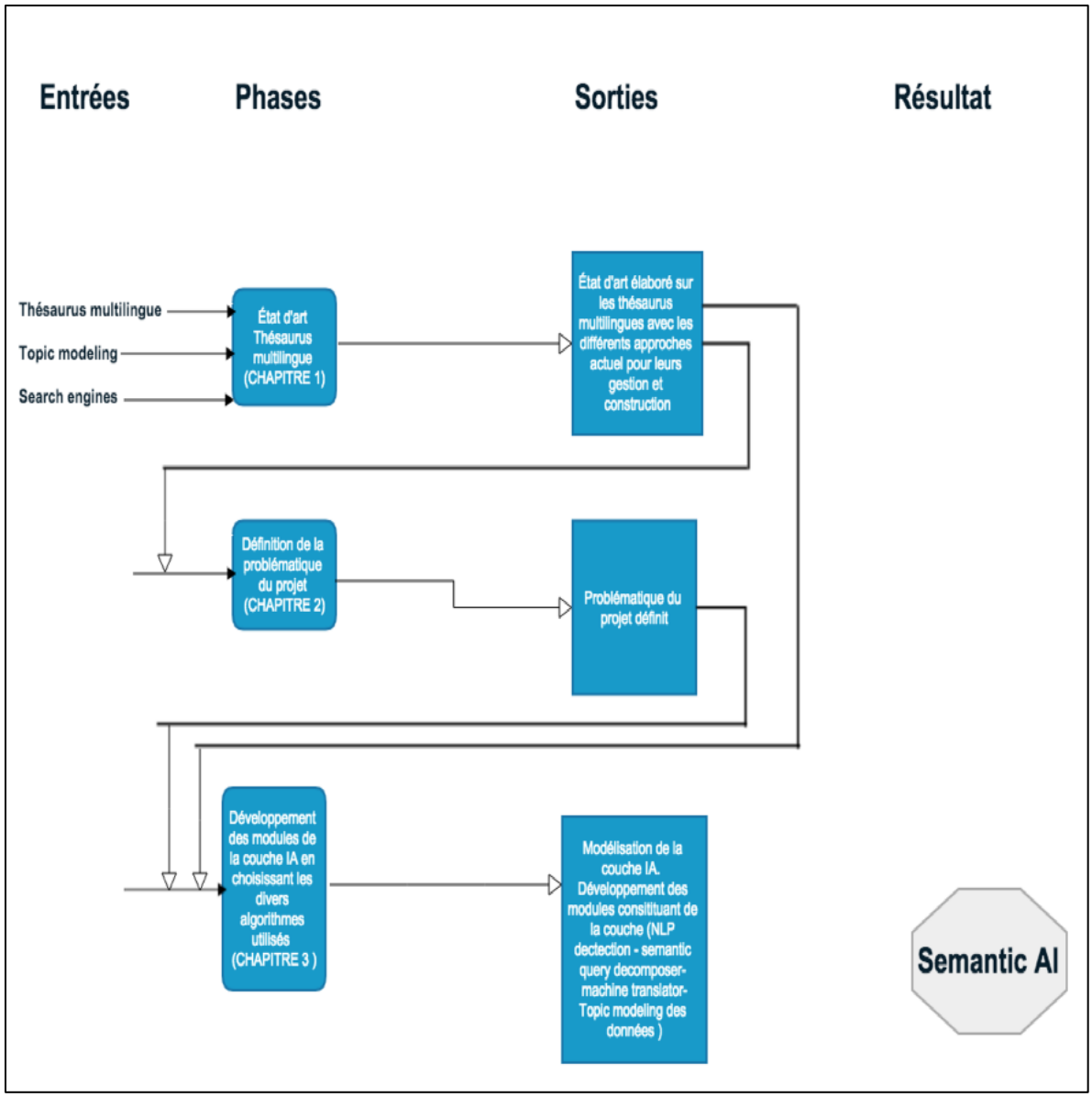

Figure 2.1 : Méthodologie du projet

2.4 Rétrospectif du chapitre 2

La couche Semantic AI sera découpée en :

1. NLP Detector : La détection de la langue naturelle d'une requête est très importante puisqu'elle autorise de faire des recherches directement dans des thésaurus qui supportent cette langue, sans parcourir tous les thesaurus;

Alors les résultats seront classés selon la langue utilisée par le client, par exemple :

Query1= Les livres de Jules verne

Résultat = liste des livres de Jules verne dans notre BD et qui sont en français.

Google Translate API traitera les langues et recevra les résultats sur le serveur Python.

2. Décomposeur sémantique des queries : après avoir détecté la langue de la requête, en utilisant l'algorithme de co-occurrence, la requête est décomposée en trois parties:
   - Query 1 = Lieu de naissance de Jules verne
   - Query1 = Sujet (Jules Verne) , Predicate (Lieu de de naissance)
   - Objet (Nantes=Resultat)

Cette décomposition permettra alors d'avoir un modèle unique pour toutes les requêtes et cela facilitera la tâche pour l'étape d'apprentissage machine. La décomposition suit la norme RDF-Triplet, qui est utilisée par plusieurs thésaurus. Cette approche optimise le temps de recherche au lieu d'essayer de comprendre le sens de la recherche pour chaque requête.

3. Traduction machine: La traduction automatique devient très importante à cause du thésaurus multilingue. Dans ce cas, il est nécessaire de gérer la dimension sémantique de la recherche effectuée.

La traduction vers Anglais, qui est une langue très répandue, permettra d'avoir plus de résultats, car tous les thésaurus multilingue supportent l'anglais. Ceci réduit la probabilité que l'utilisateur n'ait pas de résultats suite à sa requête. Cette traduction s'effectue en utilisant le Google Translate API à l'aide de la librairie Python libre : TextBlob.

4. Topic modeling : Afin d'optimiser le temps de recherche, au lieu de chercher dans tous les concepts, l'utilisation de la technique LDA permet de définir quel est le concept le plus proche de la requête effectuée. Pour se faire, une librairie libre nommée Mallet [xx] est utilisée sur un serveur web Java pour afficher les résultats.

Le chapitre 3 décrit les étapes du développement de la couche Semantic AI accompagnée d’une description des choix technologiques utilisés pour la preuve de concept.

CHAPITRE 3

Ce chapitre décrit toutes les phases de la réalisation du prototype expérimental. La première partie est consacrée à l'analyse du système existant. Puis, des choix technologiques et des descriptions détaillées de chaque module développé sont présentés. Ensuite, les algorithmes choisis sont traités. Enfin, limites du projet ainsi que les problèmes techniques rencontrés lors de la réalisation du prototype sont examinés.

3.1 Étude de l`existant:

3.1.1 Présentation du l'architecture :

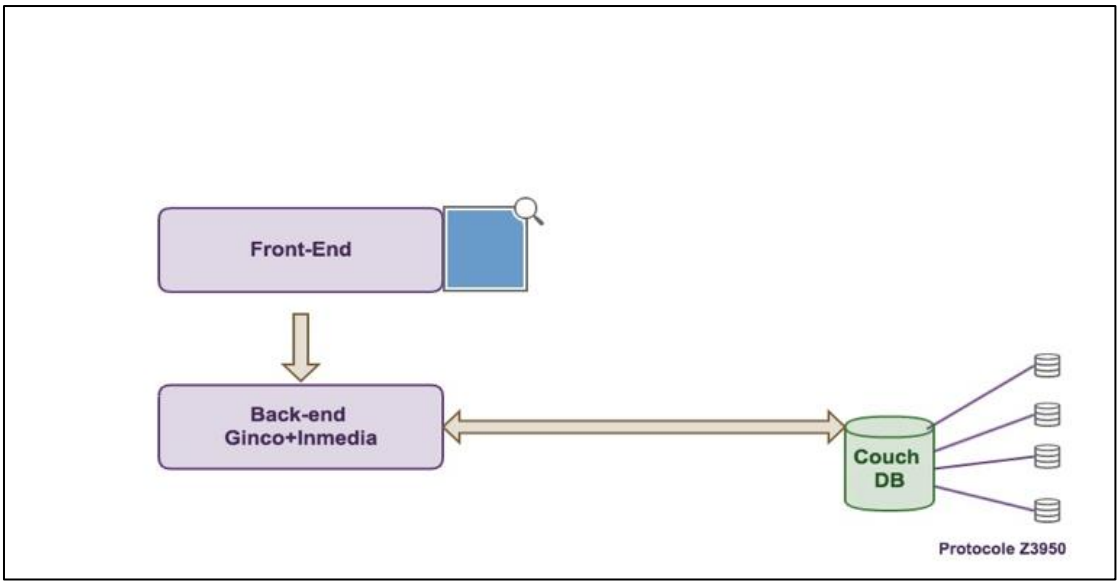

Figure 3.1 : Architecture actuelle du CMS de l'entreprise

La figure 3.1 décrit l'architecture ainsi que le système actuel qui utilise un patron de conception MVC (modèle-vue-contrôleur). Le prototype expérimental vise seulement le back-end qui est hébergé sur un nuage privé de l'entreprise. Le système contient deux modules interconnectés entre eux via des services web.

La technologie Ginco (voir la figure 3.1) est une application libre développée par le ministère de la Culture et la Communication françaises. Non seulement elle permet de gérer les thésaurus à l'aide des formats RDF et SKOS, mais aussi elle offre la possibilité de créer des thésaurus personnalisés. Bibliomondo envisage l'utilisation de Ginco afin de créer leur thésaurus multilingue à partir de leur base de connaissances et de vocabulaires. Sachant qu'un thésaurus est un regroupement de mots accompagnés de leurs synonymes avec des exemples d'utilisation, il fournit les relations sémantiques entre les mots et les groupes de synonymes.

D'ailleurs, les thésaurus sont utilisés dans le domaine de l'analyse automatique de texte (c.-à-d. text mining) et dans les applications nécessitant l'utilisation d'intelligence artificielle(AI).

De plus, les formats utilisés dans les thésaurus peuvent être soit SKOS, RDF avec ces différentes normes ou XML. Dans ce cas d'étude, l'intérêt se portera sur le RDF N-Triples qui est représenté ainsi :

Figure 3.2 : Composantes de la norme RDF N-Triples

Voici un exemple utile pour démontrer l'approche prise par cette recherche :

Query : Date de naissance d'Albert Einstein.

- Sujet : Jules Vernes
- Prédicat : Lieu de naissance
- Objet (Résultat recherché) : Nantes

InMedia (voir figure 3.1) est le CMS multilingue de BiblioMondo, conçu pour la gestion et la diffusion de ressources documentaires multiples (c.-à-d. physiques et numériques). Cependant, InMedia est un système personnalisable par les clients, mais il n'offre pas de

recherches sémantiques. Les bibliothèques contiennent souvent plus d'un million de titres, ce qui oblige à concevoir des moteurs de recherches plus intelligents ainsi flexibles à la langue d'utilisateur. InMedia est interconnecté avec Ginco via des services Web SOAP (Simple Object Access Protocol). Cela permet d'extraire des thésaurus ou d'obtenir des synonymes d'un mot recherché dans sa base de connaissances.

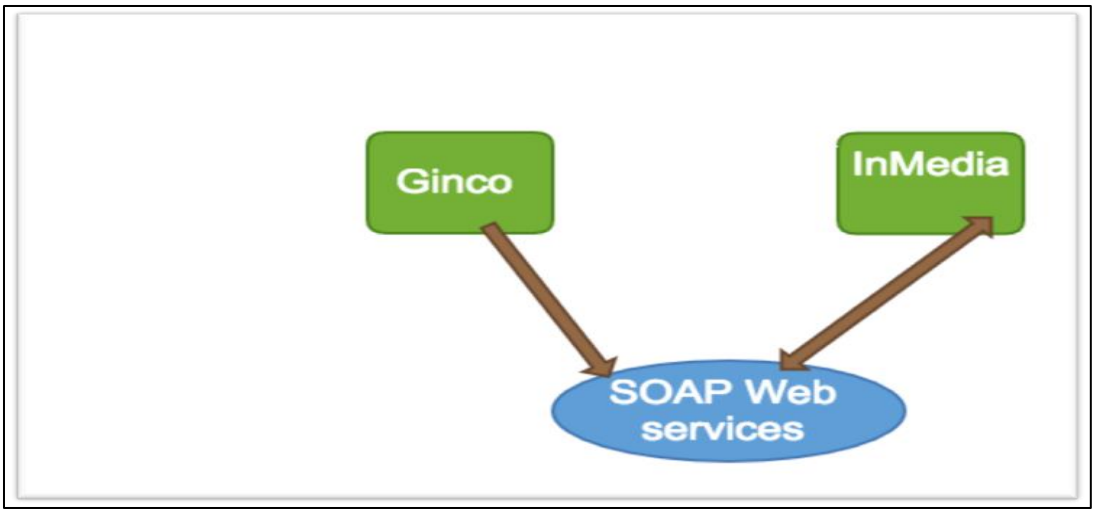

Figure 3.3 : Architecture détaillée de la communication entre Ginco et InMedia

La base de données utilise la technologie CouchDB [18]. Ce choix n'a pas été fait au hasard sachant que la bibliothèque électronique doit gérer un grand nombre de données. Cette technologie permet aussi de répartir le traitement sur plusieurs serveurs. En effet, elle actualise les données des bibliothèques via le protocole z39.50 [19]. Ce dernier, qui est un protocole de communication client-serveur, est utilisable pour récupérer simultanément les catalogues des bibliothèques et il permet de les stocker dans la base de données de l'entreprise de métadonnées.

3.1.2 Système proposé :

Le système proposé par cette recherche est le suivant :

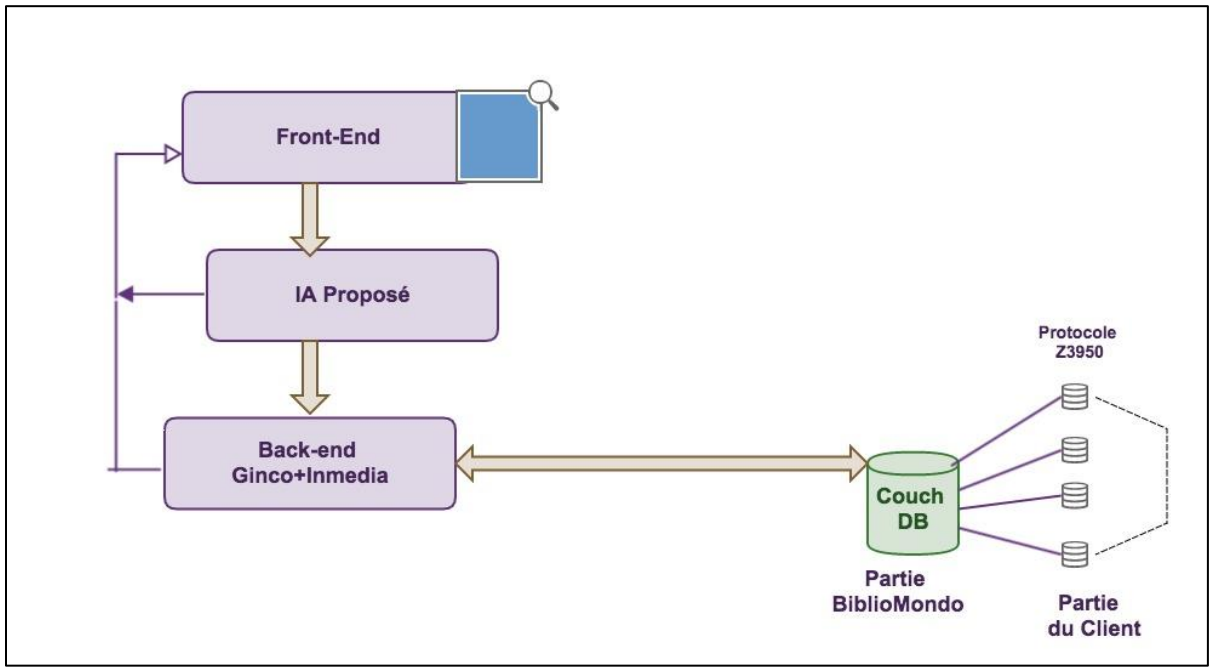

Figure 3.4 : Architecture du système proposé

La couche Semantic AI est un middleware entre le front-end et le back-end existant d'InMedia. Cette proposition a une grande importance pour BiblioMondo qui ne désirait pas intégrer des recherches sémantiques directement dans son système. Il faut faire le traitement des recherches d'une manière indépendante pour ne pas avoir un impact important sur le logiciel existant. Par conséquent, l'objectif voulu est de développer une couche qui peut être interconnecte avec les deux autres via des services web. Cette approche permet une indépendance ainsi qu'une répartition des tâches sur le système, ce qui veut dire que le projet se focalisera justement sur la recherche sémantique, ainsi que le problème du multilinguisme. Et Suite à l'analyse effectuée du système existant et la revue littéraire, trois limites ont été identifiées:

- Clusterization (Classification) : Quand il y a un thésaurus multilingue, il est difficile de classifier les données dans des thèmes distinguées, car le sens d'un mot peut

varier d'une langue à une autre. Dans ce cas, la société veut utiliser des thésaurus qui sont multilingues et dont l'évolution est exponentielle, tel que le thésaurus « VIAF (Virtual International Authority File) ». D'après la revue littéraire, plusieurs algorithmes permettent de faire la classification soit le TF-IDF, Clope, Cooccurrence et bien d'autres, avec une marge d'erreur que le projet cherche à diminuer.

- Semantic Mapping : Est la mise en place de relations sémantiques d'une manière automatique en utilisant les concepts et les bases de connaissances. À titre d'exemple, des publications où Wikipédia est utilisé comme base de connaissance afin d'extraire les relations sémantiques d'un document. Ces relations peuvent être soit synonymie soit patronymie ou homonymie, en revanche, cela est difficile lorsqu'il y a nécessité de représenter les relations entre des mots multilingues;
- Détection de la langue naturelle : La détection de la langue permettra d'orienter les résultats vers la langue utilisée dans la recherche. Ce concept est devenu répandu dans les moteurs de recherche avancés comme celui de Google. Alors que dans le système actuel InMedia le filtrage des résultats selon la langue de l'utilisateur n'est pas pris en considération.

3.1.3 Les solutions actuelles :

- L'algorithme TF-IDF: Cet algorithme est souvent utilisé pour le forage de textes. Il sert à définir le poids d'un terme dans un document (c.-à-d. un corpus). Ce poids augmente selon l'occurrence des mots dans le document. Voici comment effectuer ce calcul :

Pour un terme i dans un document j :

$$W(i,j) = TF(i,j) * \log(N/DF(i))$$

Où :

$TF(i,j)$: est le nombre d'occurrence du terme dans le document j

$DF(i)$: est le nombre de documents contenant le terme i (sachant qu'un corpus peut contenir plusieurs documents).

N : est le nombre total de documents.

- L’algorithme CLOPE : Cet algorithme est utilisé pour la classification automatique, il a été conçu afin de permettre la minimisation de similarité entre les classes extraites et maximiser la similarité entre les mots de chaque classes, alors pour un document **D** et un mot **W** et une classe **C**, le calcul arithmétique proposé par l’auteur permettra le calcul du Profit **P**. Ce dernier détermine la probabilité que le mot **W** du document **D** peut appartenir au cluster **C** , donc le maximum de profit de chaque mot pourra définir à quelle cluster le mot appartient

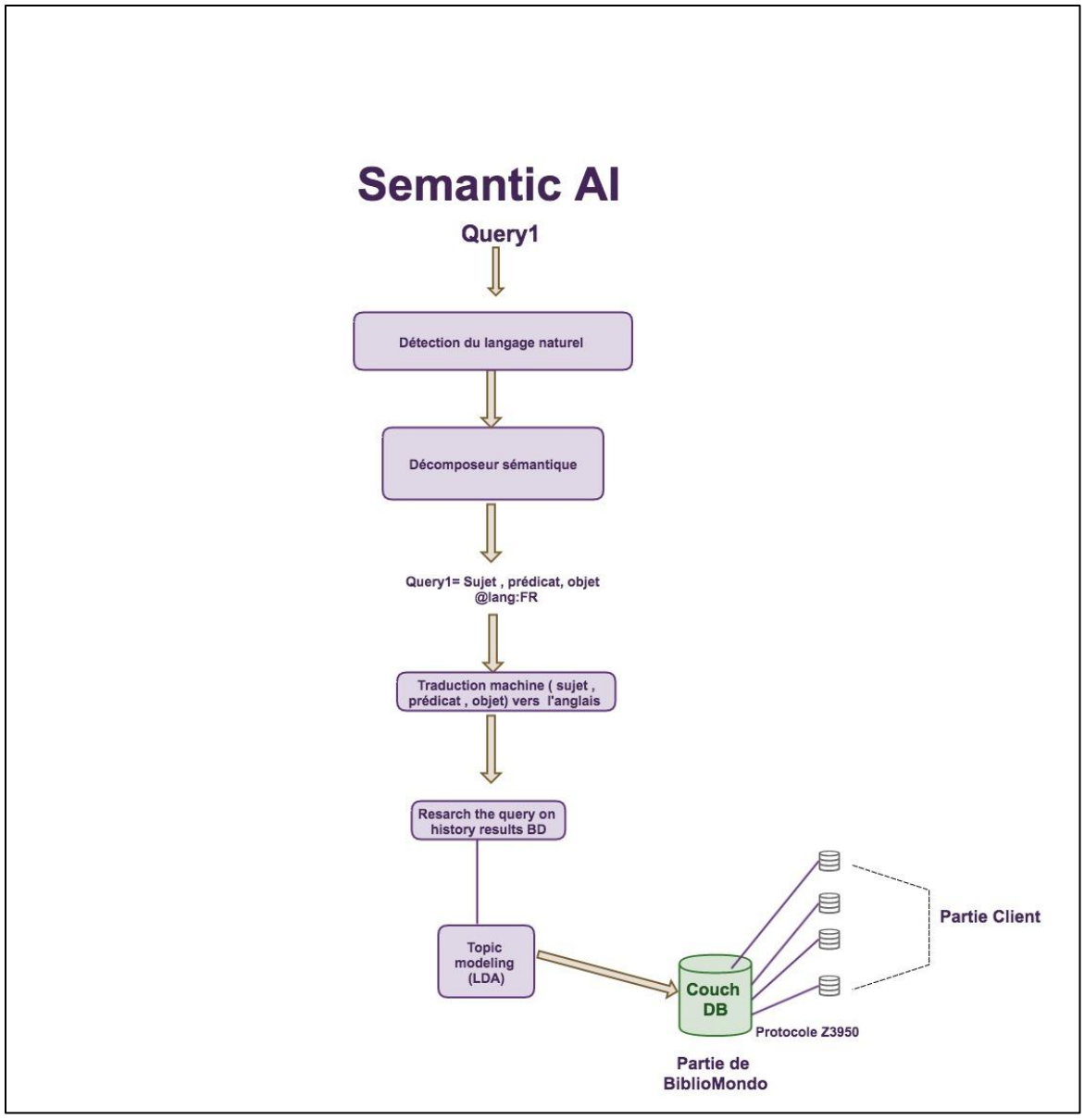

Figure 3.5 : Architecture du « Semantic AI » avec ses modules

3.1.4 Détection du langage naturel :

La détection de la langue est l'une des disciplines qui nécessite l'utilisation d'algorithmes dans le traitement automatique du langage naturel. Il est important de détecter la langue adoptée par l'utilisateur afin de filtrer les résultats.

Ce genre de traitement automatique utilise des algorithmes qui se basent sur les statistiques de prédiction, par exemple l'algorithme N-Gram utilisé par Google. Pour développer ce module, le langage Python a été choisi comme langage de programmation avec l'utilisation de l'API Google Translate. Pour le Back-end, Python se servira de la librairie TextBlob (un logiciel libre), qui s'assure la communication avec L'API de Google Translate. Ce module Python, qui met à profit TextBlob, permet d'effectuer des corrections automatiques du texte, au cas où il y avait des fautes de langue (c.-à-d. corrections orthographiques, correction de grammaire et biens d'autres)

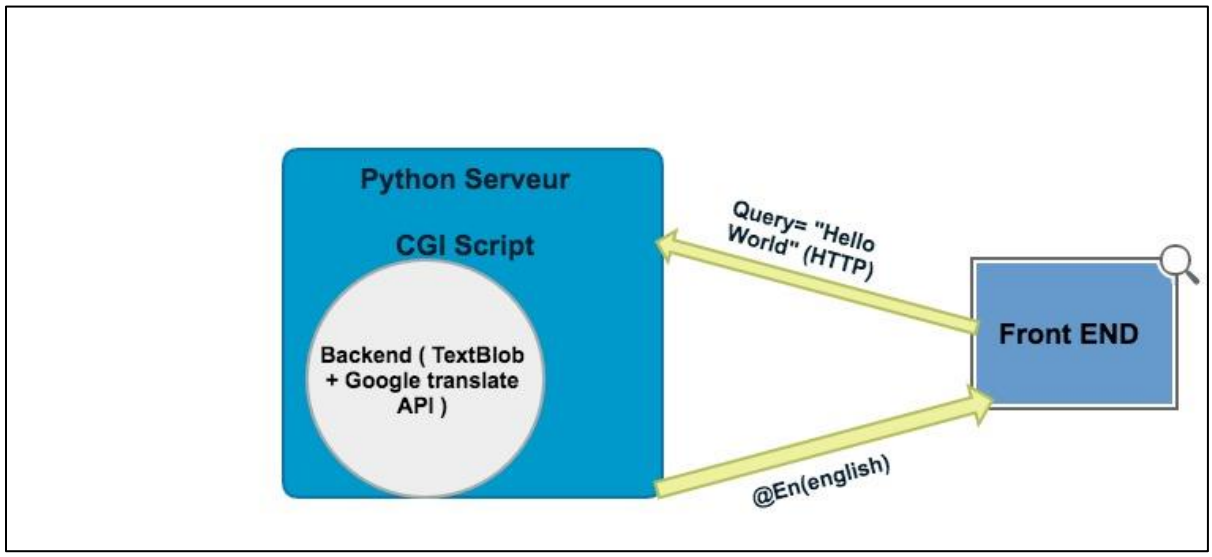

Figure 3.6 : Architecture du module Language detector

Le choix de cette architecture vise à éviter l'utilisation d'un cadriciel web pour Python du type FlasK ou Django. D'ailleurs, la complexité du problème ne nécessite pas un développement exigent le patron de conception MVC, donc CGI a été retenu pour ce prototype. Les saisies d'écran qui suivent décrivent la séquence du traitement de l'application :

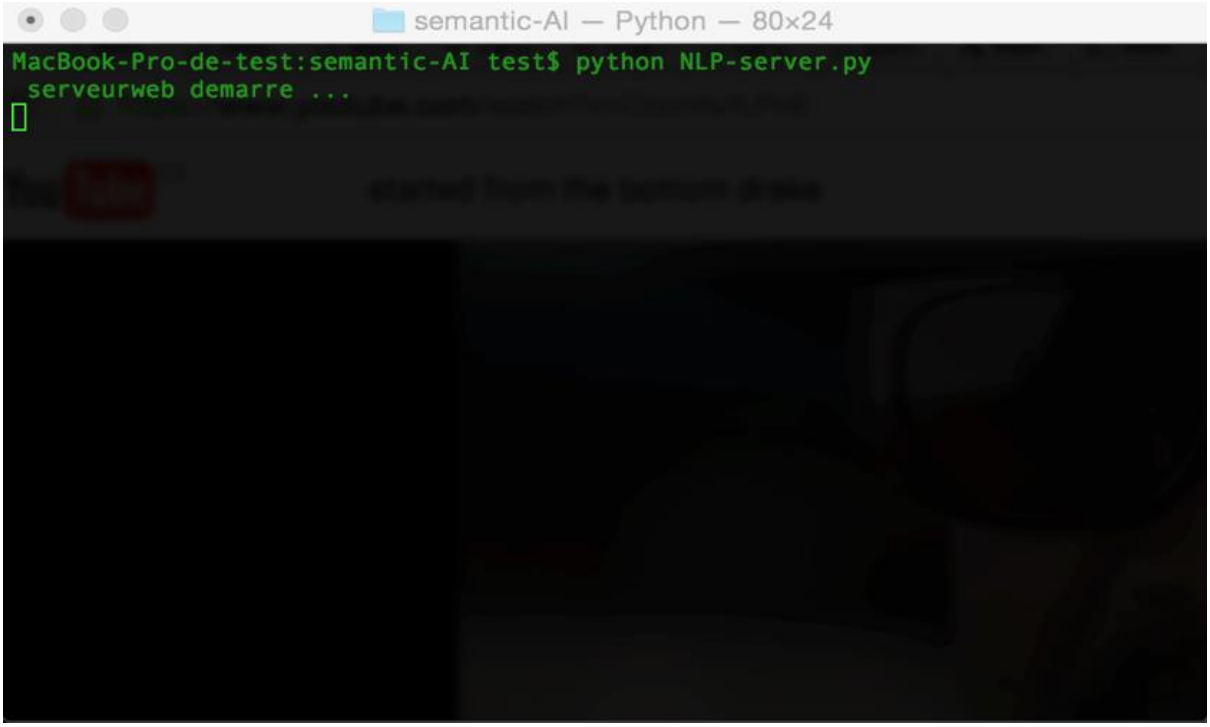

Figure 3.7 : Prise d'écran dans le lancement du serveur Python

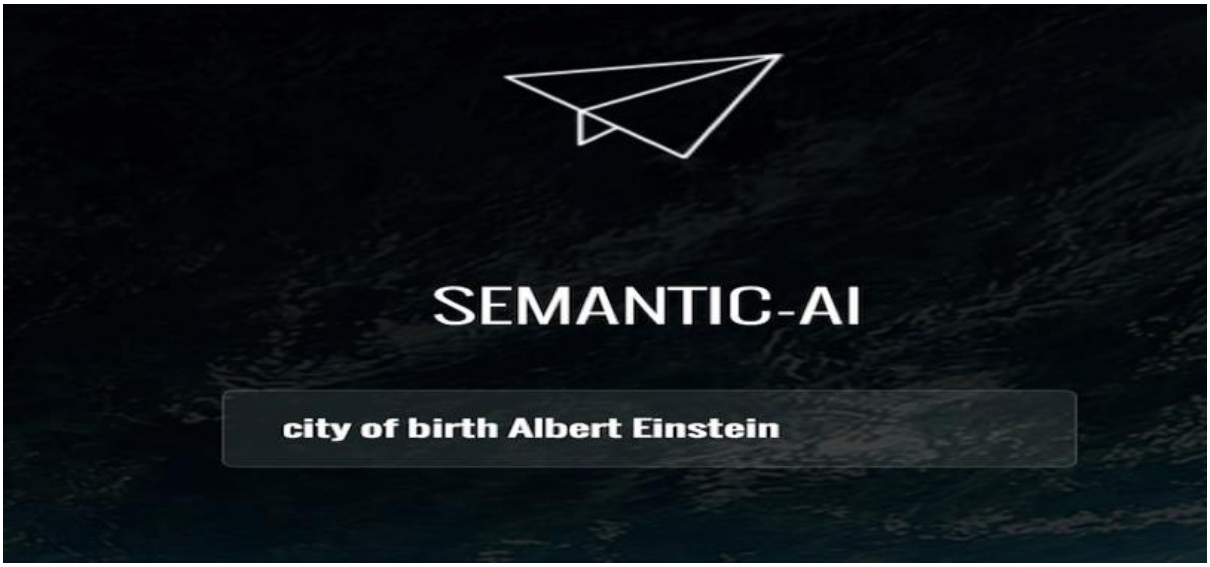

Figure 3.8 : Prise d'écran de l'interface web de la requête émis par un utilisateur

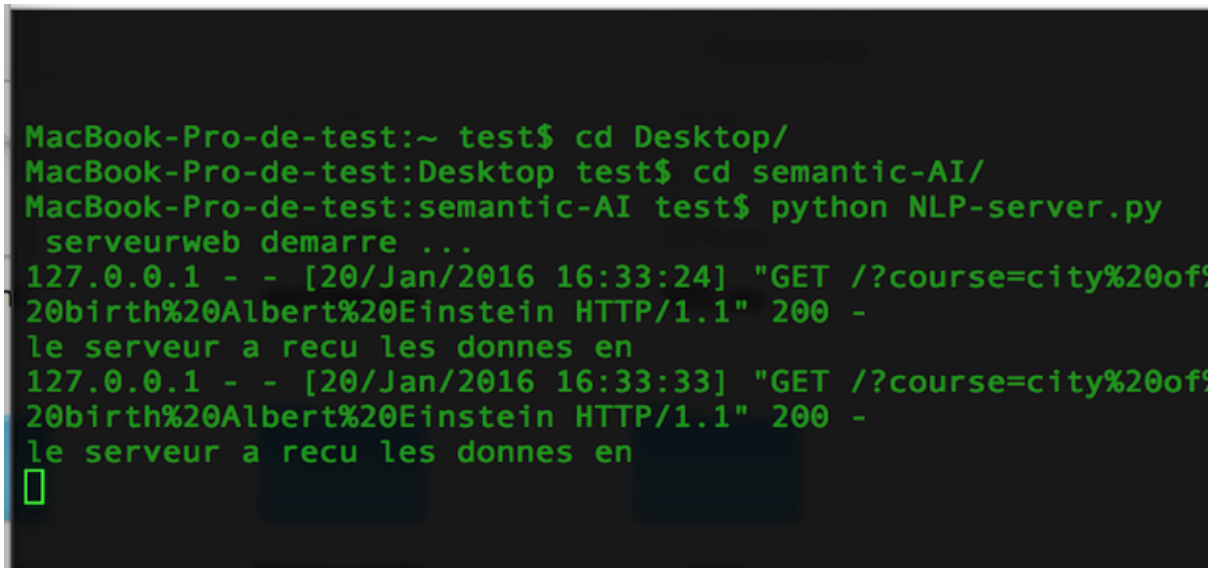

Figure 3.9 : Prise d'écran montrant la réception de la requête par le serveur et le renvoie de la langue détectée qui est EN= English

- **Décomposition sémantique des requêtes :**

La décomposition des requêtes est un module qui permet, après avoir détecté la langue, d'analyser les données et de les décomposer selon une norme. Ce traitement est important pour ce système. Puisque, le fait de simplifier la requête permettra d'effectuer un traitement sémantique plus facilement. Cependant, le défi était de découper la requête sous forme de triplets sans que le sens des données soit impacté. Pour ce faire, il fallait effectuer des recherches et des essais afin d'identifier un algorithme qui permet de faire ce type d'analyse et de traitement de texte. Dans ce cas, l'algorithme NLP Parser, qui utilise des règles de grammaire pour chaque langue, a été choisi.

Figure 3.10 : Composantes de la norme RDF N-Triple

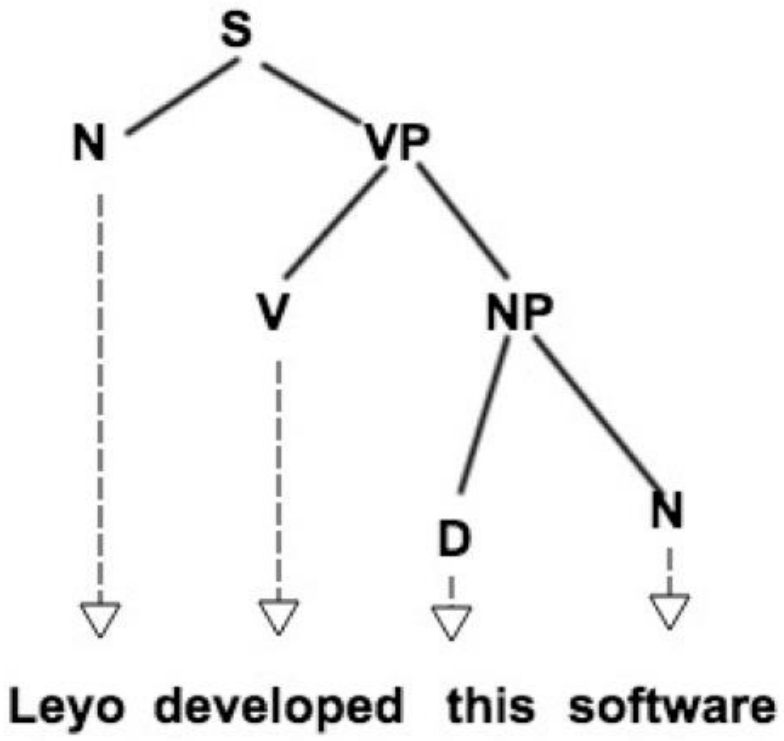

Figure 3.11: Figure imageant l'algorithme NLP Parser **[24]**

S : Sujet

N :Nom

VP : Phrase verbale

V : Verbe

NP : Phrase nominale

D : déterminant

N : Nom

L'analyse grammaticale est indispensable pour découper la phrase en triplets (sujet, verbe, complément). Elle permet de former un arbre grammatical et de faire la décomposition. Cette solution nécessite l'utilisation d'un dictionnaire, afin d'analyser le texte, et appliquer la transformation. Cependant, WordNet est utilisé dans ce cas à titre de corpus afin de détecter la classe grammaticale du mot, parce que c'est un thésaurus populaire du domaine du logiciel libre, et il est multilingue.

Le multilinguisme est essentiel pour structurer l'arbre grammatical, car les règles diffèrent d'une langue à une autre. Dans ce projet, WordNet sera le corpus utilisé pour effectuer cette analyse.

La détection de la langue permet à WordNet de définir les règles grammaticales de la langue à utiliser. Au lieu qu'il parcoure toutes les langues afin de détecter quel est le type du mot, il filtrera la recherche juste dans une seule langue.

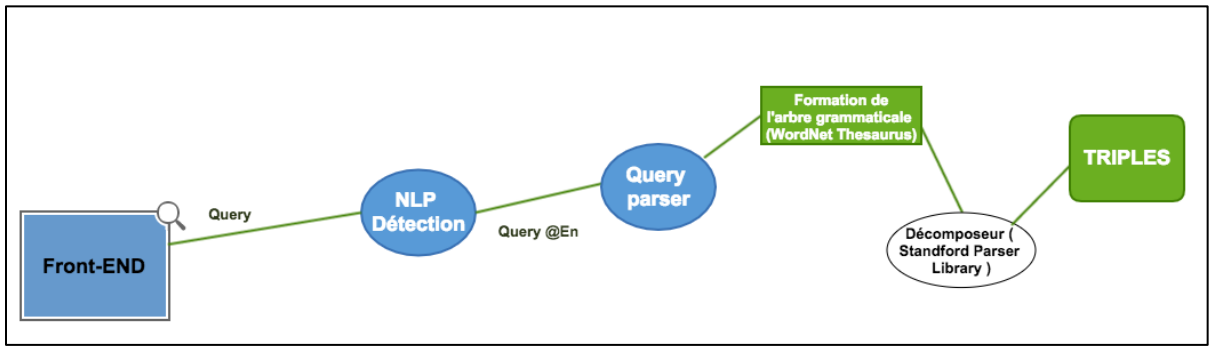

Figure 3.12 : Architecture du module NLP-Parser

Techniquement, une combinaison de deux librairies du domaine du logiciel libre a été essentielle pour développer ce module. Premièrement, pour la formation de l'arbre grammatical, NLTK [11], une librairie intégrée dans le langage python, va extraire toutes les informations du thesaurus WordNet afin de former l'arbre. Deuxièment, la librairie Standford Parser [12] permet de décomposer l'arbre grammatical formé au précèdent en triplets.

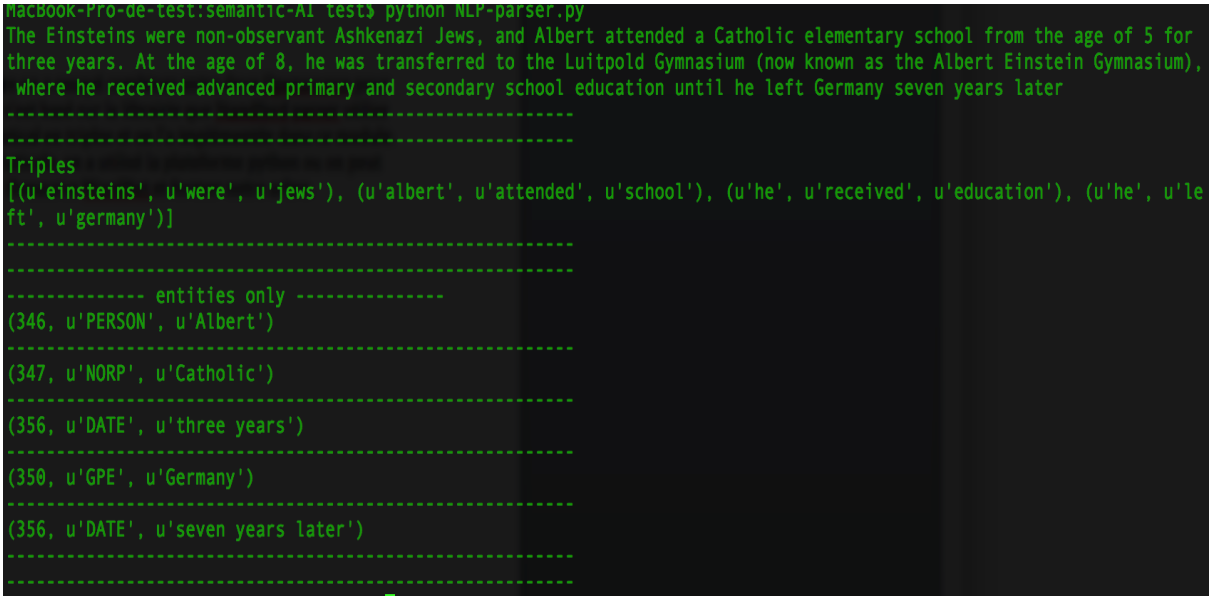

Figure 3.14 : Prise d'écran du traitement de la requête par le module NLP-Parser

Person : personne

NORP : Religion

GPE : Entité géopolitique.

DATE : date

D'après la figure 3.14 , les résultats obtenus sont satisfaisants. La simplification des requêtes est bien illustrée après le découpage sous forme de triplets. La nécessité d'avoir cette forme provient de la norme RDF N Triplets que les thesaurus adoptent dans la structuration de leurs contenus. Ce module est le coeur du programme, car il est responsable de faciliter la compréhension des thèmes sémantiques de la requête. Le Standford Parser permet l'extraction des entités importantes de la requête selon le poids dans la phrase. Ce dernier est calculée à l'aide de l'algorithme TF-IDF

- **Traduction Machine** :

La formation des triplets qui a été faite s'avère utile afin de simplifier le langage naturel ainsi que de faciliter la traduction machine. Le fait de traduire une phrase contenant un sujet, un verbe et un complément en est preuve. Par conséquent, la simplification de la requête et

l'extraction des entités nécessaires réduisent le taux d'erreur de traduction et elles offrent une conservation du sens (c.-à-d. de la sémantique). L'adoption d'une stratégie de traduction vise toujours à utiliser la langue anglaise au court de toutes les requêtes. Cependant les utilisateurs obtiendront les résultats selon la langue qui a été détectée lors de leur requête. Afin de gérer des thésaurus multilingues, il est incontournable d'avoir une langue commune partagée entre tous les thésaurus, la langue anglaise est celle qui est la plus utilisée dans les grands thésaurus que la société envisage d'utiliser. Ces exigences ont été prises en compte afin de développer ce module et concevoir la couche Semantic AI.

D'un point de vue technique, le choix portait toujours sur l'utilisation de Google Translate API, et il repose sur le fait que c'est une solution en logiciel libre, qui fonctionne bien et qui offre une grande variété de langues supportées. La libraire Python « Text Blob » a été adoptée afin de communiquer avec L'API de Google. Les requêtes effectuées en anglais n'auront pas la possibilité de passer par ce traducteur automatique.

- **Topic Modeling (LDA)** :

La technique du Topic Modeling permet de déterminer les sujets et les thèmes abstraits d'un corpus. Cette approche permet de définir les grands titres de la base de connaissance ainsi que de structurer les données par thèmes. Lors d'une recherche, le programme ciblera directement les sujets qui auront une relation avec les mots de la recherche. Le Topic Modeling peut être réalisé à l'aide de plusieurs algorithmes dont le TF-IDF, le LDA, le LSA, la cooccurrence et bien d'autres. Pour ce projet, l'algorithme LDA est un choix intéressant puisqu'il est plus précis selon plusieurs auteurs (Cheng, C., Lau, G., Pan, J., Law, K., & Jones, A. (2008)). Plusieurs de ces publications stipulent que la combinaison de TF-IDF et de LDA donne de très bons résultats. Cette approche sera utilisée dans le projet, sachant que le TF-IDF a été déjà été utilisé (pour structurer le corpus) avant d'appliquer le LDA qui va extraire les thèmes [25].

Figure 3.14 : Processus effectué par l'algorithme LDA.[15]

Le LDA permet de concaténer une matrice de distribution qui relie les thèmes et les mots ainsi que la matrice de mots et des documents. Cependant, lors de la démonstration de ce module, un changement de technologie a été exigé par les architectes systèmes de la société (pour qu'ils puissent intégrer ce module à des logiciels déjà planifiés). Étant donné que Java est le langage le plus maîtrisé par les développeurs de la société, il était indispensable de redévelopper cette partie du prototype en ce langage. Avec une adoption du traitement LDA Mallet [10], qui est une solution développée en Java et offerte en logiciel libre, propose un calcul de LDA d'un corpus avec l'aide de services web.

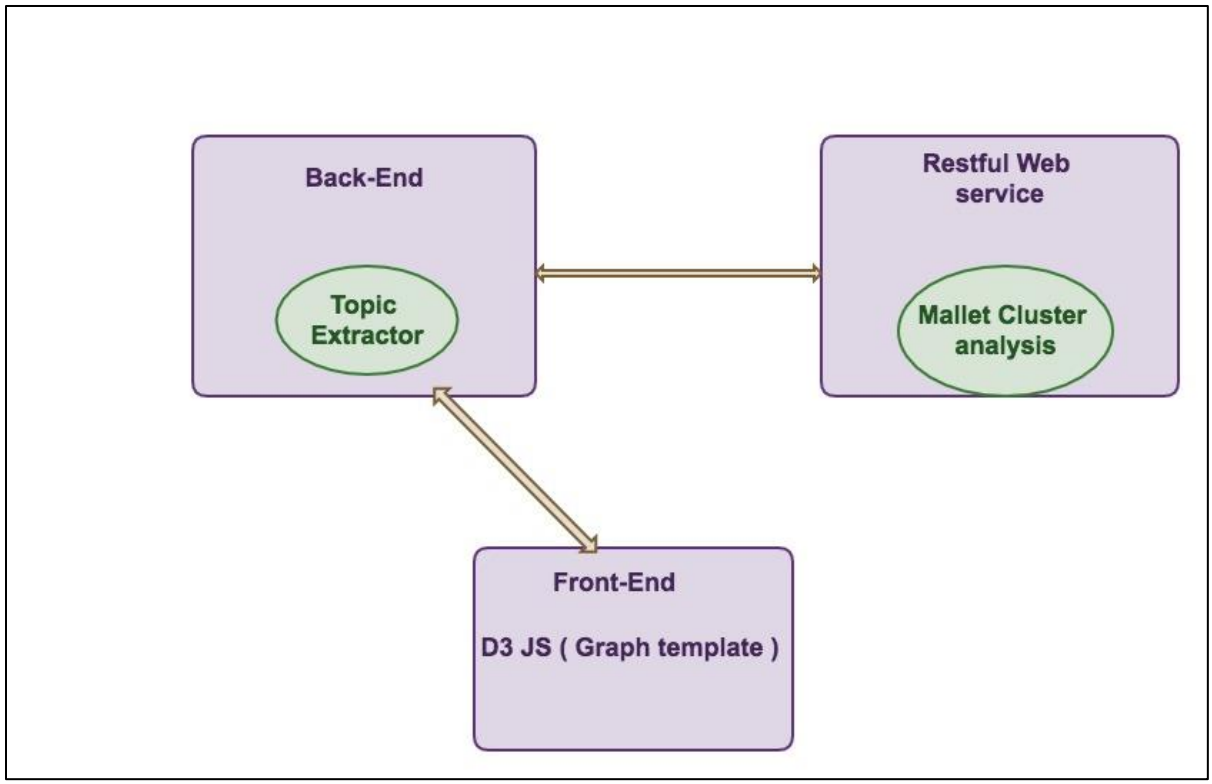

Figure 3 .15 : Architecture du module Topic Modeling avec l'algorithme LDA

Topic Extractor : Le « topic extractor » permet d'extraire les thèmes les plus influents après avoir fait le calcul de l'occurrence, à l'aide de Mallet, qui implante l'algorithme LDA.
Pour ce faire la communication passe par des services web de type Rest. La récupération des résultats dans un fichier au format Json, puis l'utilisation d'une bibliothèque JavaScript [26] est nécessaire afin de raffiner la présentation des résultats sous forme de graphes dans le but de représenter les thèmes qui ont été extraits.

Avant de passer à l'extraction, le thésaurus DBpedia[29] a été exploité à titre de base de connaissance de Wikipédia, autrement dit est un corpus contenant tous les bases de données d'une manière structurée de Wikipédia. Ainsi pour améliorer les résultats, il fallait éliminer plusieurs mots, de ce corpus, qui ne servent à rien (ces mots sont connus sous le nom de « stopwords»). Par exemple, les mots suivants de la langue anglaise (a, able, about, above, according, accordingly, across, actually, …) ont été éliminés. Pour chaque cluster de mots résultants, le cluster est indexé en utilisant le nom de l'article.

Ensuite, pour effectuer cette analyse, l'algorithme LDA, qui a été originalement développé par David Blei, Andrew Ng et Michael Jordan[26], a été utilisé pour la classification des documents, l'analyse des sentiments ainsi que dans le domaine de la bio-informatique.

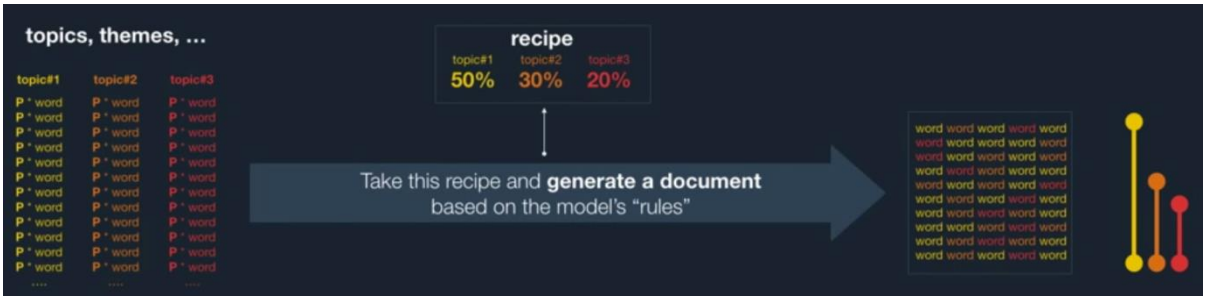

Figure 3.16: Processus de l'algorithme LDA [15]

La figure 3.16 décrit un texte qui contient plusieurs mots. Ces derniers appartiennent à certains thèmes bien définis. L'algorithme essaie de définir le taux des thèmes suite à la présence des mots appartenant à ces thèmes. L'analyse extrait les thèmes les plus influents dans le texte. Le taux de présence est calculé par la valeur de la pertinence qui va être développée dans le modèle de données.

Le Topic modeling peut ainsi se faire d'une manière périodique à chaque fois que la base donnée reçoit de nouveaux documents des bibliothèques. L'analyse de ces nouveaux documents peut être faite pour en extraire leurs thèmes.

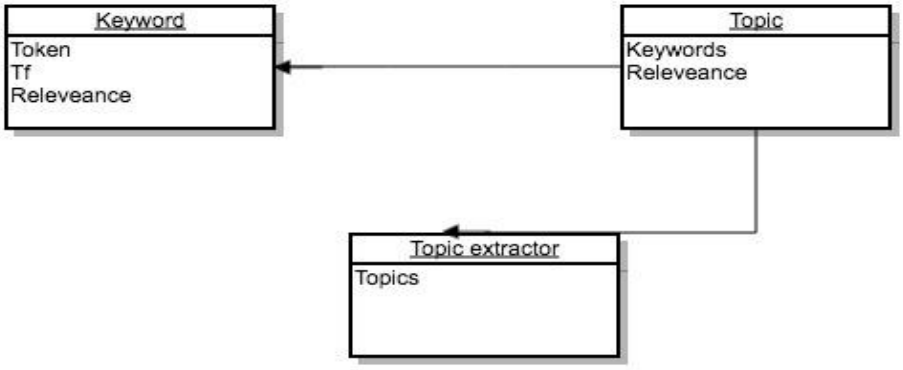

Figure 3.17 : Modèle de données du topic extractor

```
pipeList.add( new TokenSequenceRemoveStopwords(new File("/Users/test/Desktop1/semantic-AI/mallet/stoplists/en.txt"), "UTF-
pipeList.add( new TokenSequence2FeatureSequence() );

InstanceList instances = new InstanceList (new SerialPipes(pipeList));

Reader fileReader = new InputStreamReader(new FileInputStream(new File(args[0])), "UTF-8");
instances.addThruPipe(new CsvIterator (fileReader, Pattern.compile("^(\\S*)[\\s,]*(\\S*)[\\s,]*(.*)$"),
                                       3, 2, 1)); // data, label, name fields
```

Problems Javadoc Declaration Search Console Metrics - tp2 Dependency Graph View Metrics

```
<terminated> Topics [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/Contents/Home/bin/java (2016-02-10 11:34:10)
Total time: 2 seconds
gutenberg-79 ebook-97 falling-97 love-97 grant-97 allen-97
0       0.001   banana (17) coco-nut (17) tropical (12) make (11) plant (9)
1       0.001   sea (24) feet (22) length (13) ancient (13) hundred (12)
2       0.001   animals (17) make (16) single (13) australian (11) broad (10)
3       0.001   kind (22) ant (13) plants (12) feet (10) special (10)
4       0.001   fish (30) flying (13) existing (11) form (10) frogs (9)
5       0.001   large (16) nature (16) due (12) popular (11) wholly (9)
6       0.001   period (20) men (15) long (13) qualities (10) darwin (10)
7       0.001   close (17) left (17) hand (13) made (9) acquired (7)
8       0.001   century (15) england (12) true (12) children (10) existence (8)
9       0.001   things (16) family (13) impossible (11) long (10) specially (9)
10      0.001   origin (18) fact (17) matter (17) thousand (14) mouth (14)
11      0.001   seeds (22) master (10) show (9) produce (9) great (9)
12      0.001   point (14) genius (14) find (13) life (13) learn (9)
13      0.001   long (39) ogbury (12) mud (12) times (11) cold (10)
14      0.001   sense (14) means (14) world (13) made (12) cunning (10)
15      0.001   public (21) begin (10) domain (9) insects (9) fish (9)
16      0.001   human (16) burnt (8) grew (7) heads (7) chief (6)
17      0.001   sir (19) creatures (10) george (10) type (8) system (8)
18      0.001   eye (15) fish (10) necessarily (10) coloured (9) country (8)
19      0.001   hard (14) great (14) vast (14) find (13) desert (10)
20      0.001   ants (44) aphides (16) primitive (15) hand (11) nest (10)
21      0.001   left (39) side (35) hand (21) doubt (15) reason (9)
22      0.001   species (23) ants (18) water (15) point (14) living (11)
23      0.001   buried (12) till (9) poet (9) cut (8) deserts (7)
24      0.001   part (26) earth (20) hand (17) comparatively (15) found (12)
25      0.001   project (49) gutenberg-tm (33) works (30) work (26) full (20)
26      0.001   present (47) day (36) society (12) size (8) equal (7)
27      0.001   man (18) date (10) thunderbolt (8) ground (8) shown (7)
28      0.001   time (18) head (12) geological (10) surface (10) considerable (8)
29      0.001   small (19) time (13) modern (13) single (10) back (10)
30      0.001   good (21) solution (9) result (9) toad-in-a-hole (8) effect (7)
31      0.001   essentially (12) high (10) axe (9) mountains (8) feeding (8)
```

Figure 3.18 : Prise d'écran qui représente les résultats après extraction des topics avec l'algorithme LDA d'un document sans graphes

D'après la figure 3.18, le module a pu extraire 32 de thèmes .Pour la pertinence du thème ils sont presque équitables (à 0.001). Suite à l'extraction des thèmes présents dans le document

(qui est un fichier .txt), il est possible de déduire d'autres documents qui ont relation avec le même document en calculant le cosinus de l'angle de similarité. Ce genre d'approche est utilisé lors de l'implantation d'un système automatique de recommandation. Les utilisateurs peuvent ainsi se voir proposer des documents ayant des liens avec leurs requêtes, en se basant sur des thèmes communs.

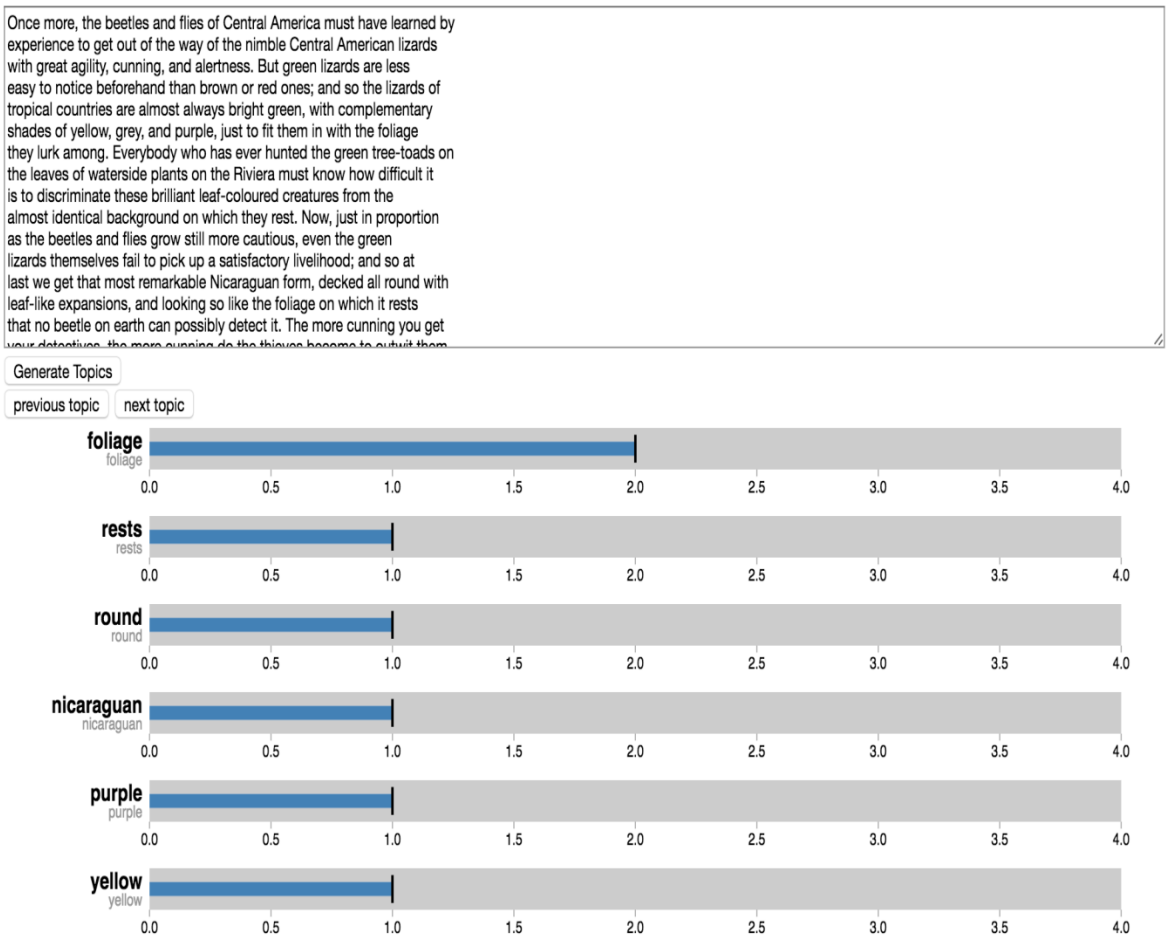

Figure 3.20 : Prise d'écran qui représente les résultats après extraction des topics avec l'algorithme LDA d'un document avec graphe (Topic 1 avec ses Keywords et leurs relevances)

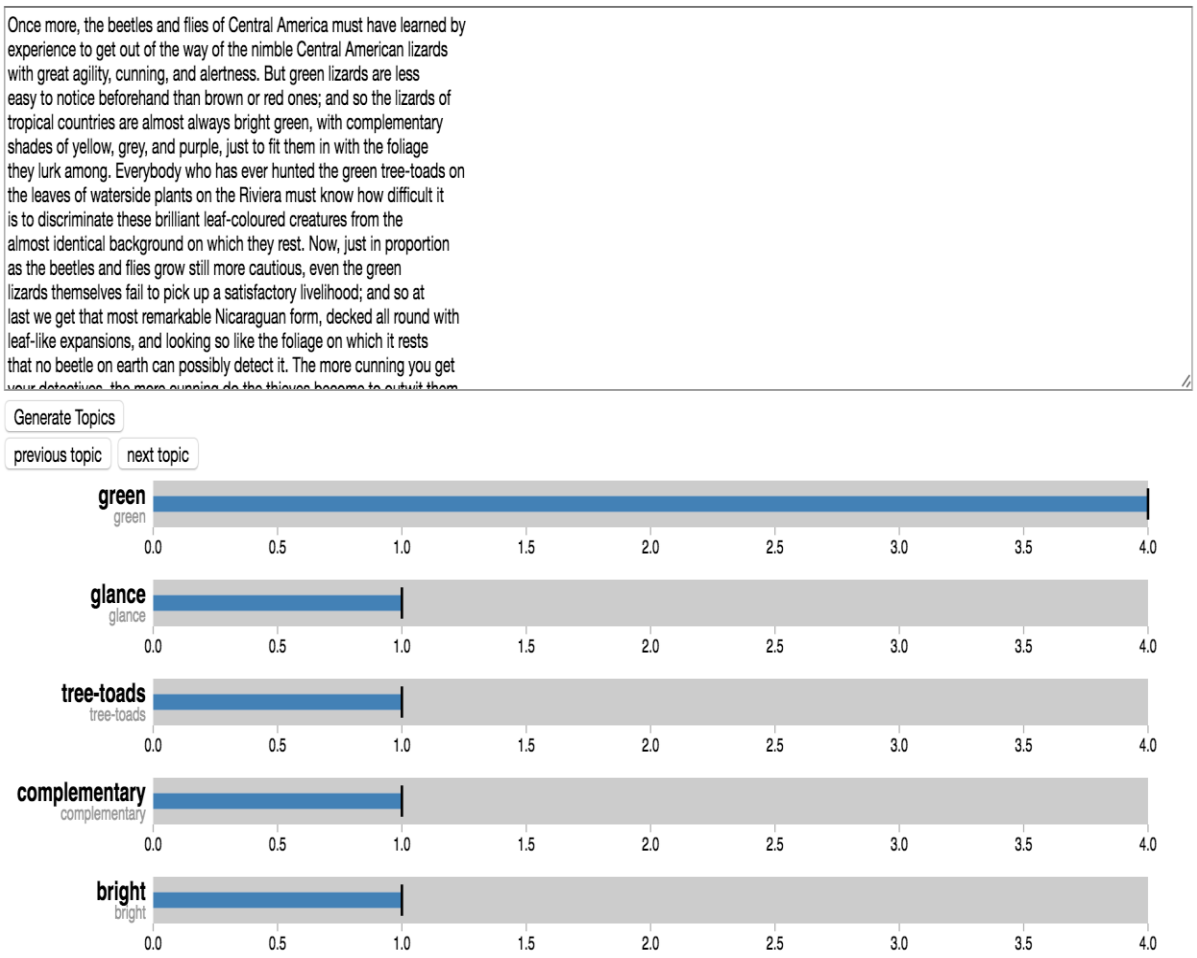

Figure 3.21 : Prise d'écran qui représente les résultats après extraction des topics avec l'algorithme LDA d'un document avec graphe (Topic 2 avec ses Keywords et leurs relevances)

Cette représentation sous forme de graphe a été mis en place à l'aide d'une bibliothèque Javacript (cadriciel D3JS logiciel libre), Pour ce faire, il utile de former les résultats de l'extraction sous forme d'un objet json (keyword,topic) et le cadriciel se charge de mettre les éléments extraites dans le graphe.

3.2 Forces et faiblesses de cette preuve de concept :

Semantic AI est composé en des modules reliés par un pipeline et qui sont parcourus par une requête composée de :

- Détecteur du langage;
- NLP parser;;
- Traduction machine;
- Topic Modeling avec l'algorithme LDA pour la base de données de l'entreprise.

Elle est inspirée des articles scientifiques qui expliquent pourquoi la recherche sémantique est plus pertinente que la recherche syntaxique. Il y est expliqué pourquoi il faut traduire toutes les requêtes en anglais de sorte à avoir une langue principale pour filtrer les résultats obtenus qui sont fournis à l'utilisateur.
Pour l'optimisation du temps de recherche et d'après le processus suivi lors des tests d'acceptation du système, cette approche est plus rapide que l'approche actuelle. Les algorithmes utilisés permettent d'obtenir des résultats pertinents d'une manière rapide.

La « clusterisation » de la base de données par thèmes (à l'aide du Topic Modeling) est fait d'une manière automatique. Ceci a été développé initialement sur des données arbitraires, afin de faire des tests.

En conclusion, Semantic AI répond exactement au besoin de la société. Maintenant elle peut développer une version robuste de son logiciel avec l'aide de thésaurus multilingues sans se soucier de la langue originale de requête. À vrai dire, les résultats des requêtes seront améliorés, mais ils restent quand même des améliorations possibles. La recherche dans le domaine sémantique développe actuellement plusieurs nouveaux algorithmes et architectures qui vont pouvoir améliorer ces résultats. Semantic AI est considéré comme une solution optimale pour faire le premier pas vers la recherche sémantique dans un moteur de recherche. Il est intéressant aussi de considérer l'ajout d'autres algorithmes pour améliorer la pertinence des résultats. Par exemple les mémoires caches qui permettent de traiter les recherches par type de profil d'utilisateur par exemple .

Donc, quels sont les points faibles de cette solution et oermettre l'améliorer dans les prochaines versions ? Tout d'abord, le Topic Modeling a été développé en Java alors que les autres modules ont été développés en Python. Cela va soulève un problème pour l'application par la suite, car il est préférable de ne pas développer des solutions tierces permettant de communiquer entre Python et Java. Il serais possible d'abandonner la librairie Java offerte par Mallet et utiliser Gensim, qui est une librairie Python offerte en open source, pour faire le traitement LDA. Il y a aussi d'autres algorithmes du domaine de l'apprentissage Machine qui pourraient être expérimentés.

En deuxième lieu, le Semantic AI se concentre juste sur la partie sémantique sans combiner la méthode syntaxique. Cela représente une grande faiblesse pour l'application. Si la prochaine version requiert un algorithme intelligent, l'approche de Google avec l'algorithme Hummingbird serait à considérer

Le multilinguisme reste toujours un défi. La méthode utilisée est optimale pour le moment et elle donne de bons résultats puisqu'elle permet de traduire une phrase composée d'un sujet verbe complément.

3.3 Décisions technologiques:

Voici les décisions technologiques prises lors de cette recherche:

- Python : Le choix de Python comme langage de programmation est dû à sa maîtrise par le développeur de cette application, ce langage est très utilisé pour ce genre d'application, il offre une multitude de solutions et d'exemples. D'ailleurs, les meilleures solutions existantes utilisent Python, à titre d'exemple, le module de Machine Learning proposé par le Cloud Azure est fait à 100% avec Python.

- Détecteur du langage : Pour développer ce genre de solution, il a été décidé d'utiliser Google Translate API, car grâce à sa vitesse de détection et également grâce à sa grande base de données. En outre, le choix était entre TextBlob et Langdetect. En essayant Langdetect, il ne fonctionnais pas, car ce dernier utilise sa propre base de données, et qui n'est pas vraiment riche. Mais après l'utilisation de TextBlob, comme il utilise le service de Google Translate API, les résultats étaient corrects. Il propose aussi d'autres services comme la traduction, qui est utilisée ensuite pour le module de traduction machine.

- Traduction machine : Pour ce module, il faut simplement choisir TextBlob, car il utilise Google Translate API.

- NLP Parser : La librairie open source Stanford Parser a été choisie, du fait qu'elle est multilingue, donc elle pourra déterminer par exemple que (Go : Verb, Partir : Verbe).

- Topic Modeling : Afin d'implémenter l'algorithme LDA dans le traitement des données, il faut utiliser la librairie Mallet, cette dernière était la seule solution qui avait de bonnes revues par les développeurs. Cependant, il y avait un deuxième choix de développer tout seul l'algorithme LDA, mais le délai de remise de l'application et le fait d'effectuer la démonstration étaient des vraies contraintes devant ce choix et par conséquent il fallait avoir recours à une solution open source.

Conclusion

Cette recherche appliquée a présenté l'étude et le développement d'une couche intelligente permettant l'utilisantion de la technique du Topic Modeling sur un engin de recherche pour les bibliothèques numériques. Il a permis d'améliorer les résultats des requêtes des utilisateurs en utilisant un thésaurus multilingue. Ce projet a été limité à l'étape de modélisation d'architecture de la solution et de du développement d'un prototype qui a été testé sur des données limitées.

Afin d'atteindre les objectifs du projet, d'abord une revue littéraire a été faite et une synthèse des techniques de création des thésaurus et des ontologies automatiques a été présentée. Des défis ont été surmontés afin d'aligner les relations sémantiques multilingues.

Un plan de mise en œuvre des modules constituants la couche intelligente a été effectué suivi d'une description détaillée des approches et algorithmes utilisés.

Le rapport se termine par une discussion des forces et faiblesses de l'approche ainsi que la justification des décisions techniques.

Bibliographies

- [1] Cássia Trojahn, et al. State of the art in multilingual and cross lingual ontology matching **(2014).** http://dx.doi.org/10.1007/978-3-662-43585-4_8

- [2] Nadezhda Lagutina ,et al. An Approach to Automated Thesaurus Construction Using Clusterization-Based Dictionary Analysis **(2015).** http://dx.doi.org/10.1109/FRUCT.2015.7117979

- [3] Jingzhi Guo, et al. Improving Multilingual Semantic Interoperation in Cross-Organizational Enterprise Systems Through Concept Disambiguation, IEEE Conference**(2012).** http://dx.doi.org/10.1109/TII.2012.2188899

- [4] Leyla Zhuhadar, et al. A synergistic strategy for combining thesaurus-based and corpus-based approaches in building ontology for multilingual search engines **(2015).** http://dx.doi.org/10.1016/j.chb.2015.03.021

- [5] Petraki, E, et al. Automated thesaurus population and management. **(2015).** http://www.qqml.net/papers/March_2015_Issue/4119QQML_Journal_2015_Petrakiet al_181-189.pdf

- [6] Zagorulko, Y, et al. Ontology-based program shell for building and editing multilingual thesauri of subject domains IEEE conference **(2013).** http://dx.doi.org/ 10.1109/SoMeT.2013.6645680

- [7] Salvatore Romeo, et al. Knowledge-Based Representation for Transductive Multilingual Document Classification (2015). http://dx.doi.org/ 10.1007/978-3-319-16354-3_11

- [8] Gavel, Ylva, et AL.Multilingual query expansion in the Svemed+ bibliographic database : a case study **(2014).** http://dx.doi.org/10.1177/0165551514524685

- [9] Liang, A. C., & Sini, M. (2006). Mapping agrovoc and the chinese agricultural thesaurus: Definitions, tools, procedures. The New Review of Hypermedia and Multimedia, 12(1), 51–62.

- [10] Libraire Java Mallet http://mallet.cs.umass.edu/

- [11] Librairie Python TextBlob https://textblob.readthedocs.org/en/dev/

- [12] Libraire Stanford NLP Parser http://nlp.stanford.edu/software/lex-parser.shtml#Download

- [13] Wikipédia 2016 Latent dirichlet allocation https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation, consulté le 18 janvier 2016

- [14] Wikipédia 2016 TF-IDF https://fr.wikipedia.org/wiki/TF-IDF consulté le 14 janvier 2016

- [15] Présentation de Andrius Knispelis sur le Topic Modeling https://issuu.com/andriusknispelis/docs/topic_models_-_video

- [16] Edward Gibson & Evelina Fedorenko (2013) The need for quantitative methods in syntax and semantics research, Language and Cognitive Processes, 28:1-2, 88-124, DOI: 10.1080/01690965.2010.515080

- [17] Ginco logiciel pour créer et gérer des thesaurus multilingue https://github.com/culturecommunication/ginco

- [18] Wikipedia 2016 CouchDB https://fr.wikipedia.org/wiki/CouchDB consulté le 14 janvier 2016

- [19] Wikipedia 2016 protocole Z3950 https://fr.wikipedia.org/wiki/Z39.50 consulté le 16 janvier 2016

- [20] Dernières technologies https://www.journaldunet.com/developpeur/expert/62991/les-techniques-et-technologies-semantiques-a-l-epreuve-du-big-data.shtml
- [21] Solr http://lucene.apache.org/solr/

- [22] Wikipedia 2016 https://fr.wikipedia.org/wiki/Ontologie_(informatique) consulté le 16 janvier 2016

- [23] Cheng, C., Lau, G., Pan, J., Law, K., & Jones, A. (2008). Domain-specific ontology mapping by corpus-based semantic similarity. In Proceedings of 2008 Engineering Research and Innovation Conference.

- [24] Exemple de NLPparser : http://ccl.pku.edu.cn/doubtfire/NLP/Parsing/Introduction/Grammars%20and%20Parsing.htm

- [25] Ramage, D., Dumais, S. T., and Liebling, D. Characterizing Microblogging Using Latent Topic Models. In Proc ICWSM'10: https://scholar.google.ca/scholar?q=Characterizing+Microblogs+with+Topic+Models+&btnG=&hl=fr&as_sdt=0%2C5

- [26]Charte graphique utilisé pour topic extractor :http://bl.ocks.org/mbostock/4061961

- [27] Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (January 2003). Lafferty, John, ed. "Latent Dirichlet Allocation". Journal of Machine Learning Research 3 (4–5): pp. 993–1022. doi:10.1162/jmlr.2003.3.4-5.993 .

- [28] PUBMED/MEDLINE http://www.ncbi.nlm.nih.gov/pubmed

- [29] BDpedia http://wiki.dbpedia.org/