PROJET EIT

ETUDES EXPÉRIMENTATIVES DE DEUX PLATEFORMES D'ANALYSE LINGUISTIQUE

11 MARS 2019

Enseignant : Nasredine SEMMAR
Auteur : Djenna ZITOUNI

I. Introduction

Afin de concevoir des systèmes informatiques intelligents capables de reconnaître, de comprendre, d'interpréter et de reproduire ce langage courant sous ses différentes formes, il est important que le système "apprenne" ce langage courant. Un bon moyen d'apprendre est d'étudier l'information textuelle. Cette étude passe par l'analyse linguistique de corpus. L'analyse comprends plusieurs étapes :

- La Tokenisation
- L'analyse Morphologique :
- L'analyse Morpho-syntaxique
- L'analyse syntaxique
- Reconnaissance d'entités nommées

CEA List LIMA et **Stanford CoreNLP** sont deux plateformes d'analyse linguistiques différentes mais au fonctionnement semblable utilisées pour étudier des l'information textuelle. Grâce à leurs nombreux outils, il est par exemple possible d'apposer un tag à tous les mots d'un texte afin de savoir leur nature grammaticale ou encore déterminer la structure d'une phrase.

Le but de ce projet est d'étudier ces plateformes d'analyse linguistique par l'expérimentation. Afin de comparer les plateformes, nous devrons utiliser des scripts Python convertissant les étiquettes propres aux deux plateformes en étiquettes universelles. Pour cela, on dispose d'une table de correspondance entre les étiquettes Penn TreeBank et les étiquettes universelles (nous devons créer notre propre table de correspondance pour la plateforme de Stanford).

En lançant les analyseurs sur les mêmes étiquettes, nous pourrons comparer leur performance, leur précision et leur taux de réussite. A travers plusieurs analyses de résultats, nous évaluerons et comparerons leur approches, afin de mieux comprendre ce qu'ils sont capables de faire et leur performance respective.

Le code source du projet ainsi que certains résultats sont disponibles ici :
https://github.com/Djenna/EIT

État de l'art

Les différentes approches d'analyse linguistique

Il y a plusieurs approches d'analyse linguistique. Tout d'abord, l'approche basée sur les règles. Cette approche permet d'imiter la manière humaine de créer des structures grammaticales. Cela implique donc une intervention humaine dans le développement et l'amélioration d'un système progressif.

La seconde approche est basée sur le machine learning. Elle est basée sur des algorithmes qui "apprennent à comprendre" un langage sans que ce langage soit explicitement programmé.
Cela est possible par l'utilisation des statistiques (comme pour Stanford), où le système commence par analyser un corpus d'entraînement (corpus annoté) pour construire son propre "savoir" et produire ses propres règles et ses propres classifications.

Enfin, il existe une approche hybride qui combine les deux approches décrites précédemment. Toutefois, cette approche est peu présente donc il est encore assez difficile de l'évaluer précisément.

Pour réaliser ces analyses, plusieurs bibliothèques existent. Voici quelques exemples de bibliothèques :

CEA LIST LIMA

La pipeline de LIMA a quelques particularités :

- La **tokenization** de LIMA est "absolue", ce qui veut dire qu'elle ne traite pas l'ambiguïté des tokens générés. Ceux-ci peuvent en revanche être fusionnés ou éclatés lors des étapes suivantes.
- La **POS** est divisée en quatre étapes :
- Le **dictionary check** , qui cherche dans un dictionnaire des mots correspondant aux tokens traités pour déterminer leur lexeme.
- Le **hyphenated words**, qui va s'occuper des mots qui n'ont pas été trouvés tels quels dans le dictionnaire en y cherchant plutôt leurs sous-éléments (radicaux, mot composés...). Cela est particulièrement utile pour le traitement des langues comme l'allemand et le hongrois, qui comportent de nombreux mots composés à partir d'autres.
- La reconnaissance d'entités spécifiques : c'est ce qui correspond à la reconnaissance d'entités nommées décrite en introduction
- Le **POS-tagging** à proprement parler : il utilise l'algorithme "Viterbi"

STANFORD CORE NLP

Core NLP, présente les caractéristiques suivantes :

- L'étape **cleanxml** retire les tags d'un document, permettant de traiter efficacement des documents au format xml.
- **ssplit** sépare la suite de tokens en phrases.
- **truecase** rétablit la casse (estimée) correcte d'un mot.
- La **POS** utilise un algorithme d'entropie maximale.
- L'analyse morphologique est séparée en deux sous-étape : **lemma** et **gender** , qui reconnaissent, comme leurs noms l'indique, le lemme et le genre des tokens.
- La reconnaissance d'entités nommées se fait d'abord par ner , qui se base sur l'ensemble des entités issues des corpus d'apprentissage, et regexner , qui est basée sur des règles pour trouver les entités nommées restantes.
- **sentiment** peut analyser les sentiments présents dans une séquence.
- **dcoref** permet d'associer des mots, comme les pronoms, aux tokens auxquels ils réfèrent, générant ainsi un "graphe de coréférence".

NLTK

C'est une librairie en Python, très répandue dans le monde de la recherche et de l'éducation. Elle sert essentiellement au preprocessing d'un texte, à sa tokenisation et à son étiquetage morpho-syntaxique. Dû à sa nouveauté, elle présente des performances inférieures aux outils industriels.

SPACY

Un outil industriel compatible avec les librairies de machine learning comme Py- Torch ou scikit-learn. L'un de ses atouts principaux est la vitesse d'exécution. Il est basé sur une méthode de reconnaissance statistique.

Pourquoi LIMA et CORE NLP ?

Le choix de comparer ces deux plateformes a été fait pour différentes raisons.
Tout d'abord, car ces deux plateformes sont gratuites et disponibles en open source (licence publique) sur internet, contrairement à certaines plateformes de Microsoft, Google, Amazon ou IBM. La seconde motivation est le fait qu'il est intéressant de comparer ces deux plateformes, puisqu'elles suivent des méthodologies d'analyse linguistique différents (règles pour LIMA et apprentissage statistique pour Stanford). Les deux possèdent une documentation développée qui permettra de résoudre d'éventuels problèmes techniques. Ensuite, les deux plateformes possèdent des approches différentes : les outils de Stanford utilisent principalement de l'apprentissage automatique (statistique), alors que les outils du CEA List sont basés sur des règles (dictionnaire). Cette différence sera intéressante à comparer lorsque nous analyserons les résultats.

Résultats

CEA LIST LIMA

Afin de pouvoir analyser les résultats de LIMA en rapport avec l'extraction d'entités nommées et avec l'analyse morpho-syntaxique de manière pertinente, je dois tout d'abord implémenter quelques scripts.

Pour l'analyse morpho-syntaxique, il est nécessaire de modifier l'affichage des étiquettes du fichier de sortie donné par LIMA en un texte de la forme : "When_WRB it_PRP 's_VBZ ..." A la base, les étiquettes produites par LIMA respectent la syntaxe Penn TreeBank (PTB). On peut donc se lancer dans une première évaluation de l'outil en utilisant le fichier de référence correspondant aux étiquettes PTB. On obtient les **résultats** suivants :

Metrics	Number	Readable number

Word precision	0,8771929825	87,72%
Word recall	0,9090909091	90,91%
Tag precision	0,6403508772	64,04%
Tag recall	0,6636363636	66,36%
Word F-measure	0,8928571429	89,29%
Tag F-measure	0,6517857143	65,18%

Pour effectuer la même analyse sur les étiquettes universelles, il faut **convertir les étiquettes** produites directement par LIMA afin de réaliser l'évaluation. Afin de transformer les étiquettes PTB en étiquettes universelles, il faut dans un premier temps remplacer les étiquettes LIMA par leurs équivalents PTB suivants :

- SCONJ → CC
- SENT → .
- COMMA → ,
- COLON → :

LIMA considère les noms propres de plusieurs mots comme un seul et unique to- ken. Afin que la conversion se fasse bien et que le parsing du fichier à convertir soit cohérent, je peux avoir recours à l'ajout temporaire d'un mot-clé, "Espace", entre toutes les sous-parties de ce nom propre. Par exemple, "**Boca Raton**" deviendra "**BocaEspaceRaton**". Ensuite, je pourrai lancer le script *evaluate.py*. J'obtiens les résultats suivants :

Metrics	Number	Readable number
Word precision	0,8771929825	87,72%
Word recall	0,9090909091	90,91%
Tag precision	0,7105263158	71,05%
Tag recall	0,7363636364	73,64%
Word F-measure	0,8928571429	89,29%
Tag F-measure	0,7232142857	72,32%

On remarque dans un premier temps que les statistiques concernant Word (Word precision et Word recall) restent identiques dans les deux cas. Cela s'explique par le fait que la segmentation est réalisée de la même manière dans les deux cas. De plus, nous pouvons observer que les valeurs concernant le Tag sont plus hautes dans le cas des étiquettes universelles. En observant la table des correspondances, on peut observer que plusieurs étiquettes PTB sont réunies sous une seule étiquette universelle. Les tags universels étant donc moins complexes, il est donc logique d'observer plus de correspondances dans ce cas précis.

STANFORD CORE NLP

Cette partie va être réalisée en 3 étapes : tout d'abord l'évaluation de **l'outil de désambiguïsation morpho-syntaxique,** puis celle de **l'outil de reconnaissance des entités nommées**, et enfin celles des **outils de reconnaissances d'entités nommées des deux outils**, que sont Stanford et LIMA.

Évaluation de l'outil de désambiguïsation

Dans un premier temps, il faut lancer l'outil **POS tagger** sur le fichier sur lequel on veut effectuer les tests : wsj_0010_sample.txt. Le POS tagger produit alors un fichier contenant le texte originel taggé avec des étiquettes Penn Treebank (PTB). J'obtiens les résultats suivants pour ce fichier PTB lorsque comparé avec le fichier de référence :

Metrics	Number	Readable number
Word precision	0,746478873239	74,65%
Word recall	0,481818181818	48,18%
Tag precision	0,732394366197	73,24%
Tag recall	0,472727272727	47,27%
Word F-measure	0,585635359116	58,56%
Tag F-measure	0,574585635359	57,46%

Nous pouvons remarquer que le rappel (soit le pourcentage d'éléments pertinents sélectionnés) est considérablement inférieur à la précision (le nombre d'éléments pertinents sur le total des candidats sélectionné). Cela pourrait être expliqué par le fait que les étiquettes PTB sont plus précises et entraîneraient l'élimination de certains mots pertinents.
Nous pouvons ensuite faire passer les étiquettes PTB en étiquettes Universelles. En lançant l'évaluation, j'obtiens les résultats suivants :

Metrics	Number	Readable number
Word precision	0,990909090909	99,09%
Word recall	0,990909090909	99,09%
Tag precision	0,972727272727	97,27%
Tag recall	0,972727272727	97,27%
Word F-measure	0,990909090909	99,09%
Tag F-measure	0,972727272727	97,27%

Cette disparité peut s'expliquer par le fait que les étiquettes PTB sont plus complexes que les étiquettes universelles. Je peux donc dire que Stanford est plus performant en utilisant des étiquettes universelles qu'en utilisant des étiquettes PTB.

Évaluation de l'outil de reconnaissance d'entités nommées

Pour cette partie, un script permettant de représenter les données sous le format demandé (Entité nommée - Type - Nombre d'occurrences - Proportion dans le texte) a été créé. On peut noter que la reconnaissance d'entités nommées de Stanford est fonctionnelle et qu'elle est similaire à celle de LIMA, avec moins de complexité.

Évaluation de l'outil de reconnaissance d'entités nommées des deux plateformes

Dans cette partie, la capacité de reconnaissance d'entités nommées des deux plateformes que sont Stanford et LIMA sera évaluée.

Pré-traitement des données

Pour cela, il faut dans un premier temps réaliser un table de correspondance des étiquettes d'entités nommées entre les deux plateformes (cf tableau ci-dessous).

LIMA	STANFORD
Organization.ORGANIZATION	ORGANIZATION
Location.LOCATION	LOCATION
Person.PERSON	PERSON
_	/O
DateTime.TIME	/O
DateTime.DATE	/O
Numex.NUMEX	/O
Numex.NUMBER	/O

Grâce à cette table, je peux désormais transformer les outputs de LIMA pour qu'ils suivent la même syntaxe que ceux de Stanford.
Ensuite, il faut transformer les étiquettes des entités nommées de Stanford en syntaxe universelle (notamment transformer /O en _O). Dans ce cas, il n'y pas besoin de rajouter de mot-clé "Espace" puisque Stanford prend déjà compte de cas : par exemple, "Boca Raton" devient "Boca_LOCATION Raton_LOCATION".

Corpus de référence

Etant donné que notre script evaluate.py ne peut prendre en compte que des fichiers ayant des étiquettes universelles ou PTB, il faut transformer le fichier du corpus de référence, sous standard d'étiquetage ENAMEX.

Le script de conversion effectué ne permet d'obtenir que les informations contenues dans les balises. Le fichier en sortie est donc incomplet et ne me permet pas de faire la comparaison.

Difficultés rencontrées

Afin de comparer les deux outils, il a fallu passer par plusieurs étapes de transformations des étiquettes. Je n'ai pas eu le temps de compléter tous les scripts nécessaires à cette transformation : malheureusement, certaines lacunes subsistent. Ces scripts ne sont pas parfait et pourraient poser problème sur un corpus plus grand.

Cependant, une de mes plus grandes difficultés a été de réaliser ce projet seule.

Conclusion

Ce projet m'a permis d'en apprendre plus sur le fonctionnement de plateformes d'analyse linguistique, et plus particulièrement sur l'évaluation de la segmentation (les résultats liés à word) et la désambiguïsation morpho-syntaxique (les résultats liés à tag).
Malgré leur objectif commun, on peut remarquer que LIMA et Stanford présente des différences, essentiellement dans leur standards d'étiquetage. Lorsque je passe leurs étiquettes en étiquettes universelles, la précision et le rappel pour la segmentation ou la désambiguïsation morpho-syntaxique se rapproche. Stanford présente de faibles résultats lorsque les étiquettes universelles ne sont pas utilisées. LIMA possède donc un clair avantage.

Le point principal qui différencie les deux plateformes est la gestion des entités dans le cas des noms propres.. LIMA est donc objectivement un meilleur outil en ce qui concerne la tokenization. Cela pourrait faire la différence lors de la désambiguïsation morpho-syntaxique par rapport aux résultats de Stanford. La différence est sans doute causée par l'approche des deux plateformes : Stanford utilise en effet une approche statistique et LIMA une approche par règles. Avec une étude approfondie, il serait possible de comparer plus de points et révéler plus de différences.