Programmation de Modèles Linguistiques (II)

(L5SOPROG L3 Sciences du Langage)

2024-2025

Projet

Caroline Koudoro-Parfait, Sorbonne Université

Date et Rendu

Le projet se fait en binôme, néanmoins chaque étudiant doit faire le dépôt du travail sur Moodle.

La date de rendu est fixée au **30 mars à 23h59** sur MOODLE. Tout retard sera pris en compte dans la note. Pour tout souci sur ce dépôt envoyez un mail conjoint à : `caroline.parfait@sorbonne-universite.fr`

Le rendu doit être composé d'un rapport de 3-6 pages au format PDF et de votre code Python en `.ipynb` ET en `py`.

Vous devez commenter votre code, c'est à dire que l'on doit comprendre pour chaque fonction/chaque cellule quels sont les traitements que vous faites. Dans le rapport vous expliquerez pourquoi vous faites ces traitements.

Tout le rendu doit être archivé dans un seul fichier `.zip` et ensuite déposé sur le site Moodle (Rubrique: Projet).

Codes Python

Les codes Python attendus seront organisés dans un ou plusieurs fichiers. Chaque fichier devra être présent dans les deux formats `ipynb` ET `py` Factorisez vos codes Python: importer les bibliothèques et définir d'abord les fonctions, puis les appeler au fur et à mesure que vous avez besoin.

Rapport

Le rapport, composé de 3-6 pages, doit être organisé par les sections "Introduction", "Développement", "Conclusion et perspectives" et "Références bibliographiques".

N'oubliez pas de mettre sur la première page du rapport vos nom, prénom et numéro d'étudiant.

"Introduction" , expliquez le sujet et les problèmes traités (par ex. les avantages d'utiliser un moteur de recherche et ses limitations).

"Développement" , expliquez vos choix concernant les traitements effectués sur les données tels que : les choix de tokeniseur, le nettoyage ou non du vocabulaire, etc. Montrez les résultats intermédiaires et expliquez aussi les choix de traitements des résultats intermédiaires ou partiels s'il y a lieu.

"Conclusion et perspectives" , expliquez les difficultés que vous avez rencontrées et comment vous les avez résolues. Vos conclusions et les améliorations envisageables du programme que vous avez produit.

Présentation orale

Vous présenterez vos travaux en binôme pendant 10 min,la parole devra être équitablement répartie. Puis il y aura une phase de questions de 10 min. Pour cette présentation il est attendu 4-5 slides qui retracent le contexte du projet, les résultats et les conclusions et perspectives. Vous pouvez aussi aborder la gestion du projet.

- Ne mettez pas de codes Python dans le rapport sauf si vraiment vous devez illustrer quelque chose d'important.
- Le projet consiste essentiellement à améliorer ce que nous avons fait dans les derniers TDs.
- Les énoncés sont des pistes exploratoires, vous ne devez pas obligatoirement limiter votre travail de réflexion à y répondre.

1 Extraction d'Entités Nommées au format BIO avec spaCy dans différentes versions du texte + dépôt Github

Données :

1. ELTeC-fra sur Moodle :Corpus de textes comprenant une version de référence REF et deux versions OCR (Kraken, Tesseract)

Résultat attendu :
En vous appuyants sur les TDs précedents vous pratiquerez la REN avec spaCy pour obtenir en sortie :

1. dans la strucuture de données qui vous semble la plus pertinente : les entités nommées, leur label BIO et leur label de catégorie sémantique (PER, LOC, ORG, MISC). Chaque fichier d'entrée a un fichier de sortie.
2. Vous comparerez les sorties de la référence et des OCR afin de déterminer les VP, les FP, le FN et les VN.
3. vous calculerez la précision, le rappel et le f-score.
4. A l'aide de l'outil de morphologie syntaxique ou *Part-of-speech tagging* de spaCy[1] vous annoterez automatiquement les textes. Chaque fichier d'entrée a un fichier de sortie correspondant.
5. vous récupérerez les tokens dont le label est "PROPN" (*Proper Noun*).
6. Pour chaque fichier vous comparerez les sorties du *Part-of-speech tagging "PROPN"* de la référence avec les sorties pour les OCR, en utilisant par exemple les intersections, unions, différences.
7. vous comparerez selon une stratégie de votre choix, que vous expliciterez dans le rapport, les sorties obtenues avec le *Part-of-speech tagging "PROPN"* à celles obtenues avec l'outil de REN et vous observerez si les entités nommées sont bien annotées PROPN.

[1] `https://spacy.io/usage/linguistic-features#morphology`

8. Vous proposerez des graphiques permettant de représenter entre autres :

    - La proportion d'entités pour chaque label sémantique (PER, ORG, LOC, MISC) selon les différentes versions des textes.
    - Les proportions de VP, FP, VN, FN dans la REN sur des données OCR pour chaque textes selon les versions et globalement.
    - les intersections entre les sorties de REN et les sorties de Part-of-speech tagging "PROPN" avec des diagrammes de Venn.
    - La proportion de verbe, d'adjectif, de nom commun etc. qui ont été annotés comme des Entités nommées.
    - ...

9. Vous procéderez bien au dépôt final sur Moodle, mais tout au long des séances de travail vous ferez des dépôts réguliers sur Github que vous partagerez avec votre binôme et les enseignants.

Principaux outils nécessaires :

- Pandas
- Spacy
- Matplotlib/seaborn
- Github

2 Statistiques sur des textes dans différentes langues + Clustering

Données :

1. corpus-multi sur Moodle

Résultats attendus :
En vous appuyant sur les précédents TD vous créerez une représentation du vocabulaire du corpus français et du corpus dans deux langues de votre choix à partir de corpus-multi.

1. Vous procéderez à la tokenisation des textes selon la méthode qui vous semble la plus pertinente.
2. Vous procéderez à la lemmatisation des textes avec spaCy et les modèles de langue adaptés à chaque langue.
3. Vous procéderez à la REN avec spaCy et les modèles de langue adaptés à chaque langue.
4. Pour déterminer le vocabulaire vous exclurez les stop-words et les noms propres de chaque textes pour chacune des langues selon les méthodes qui vous semblent pertinentes et vous expliciterez pourquoi dans le rapport.
5. Vous représenterez graphiquement :

- le nombre de tokens par textes pour chaque langue ;
- le nombre de token type (vocabulaire) par textes pour chaque langue ;
- la proportion de lemmes par textes pour chaque langue
- la proportion de noms propres pour chaque langue

6. vous proposerez un partitionnement des données et sa représentation graphique qui permet de rassembler les mots les plus proches d'une même langue selon les bigrammes/trigrammes de caractères.

7. vous proposerez un partitionnement des données et sa représentation graphique qui permet de rassembler les mots les plus proches d'une même langue selon les 4grammes/5grammes de caractères.

Principaux outils nécessaires :

- numpy
- pandas
- scikit-learn
- seaborn