Projet

TAL

Traitement automatique des langues

ESTEban neraudau – Maxime gourion

ET5 Info

Année

2023/2024

Table des matières

[1 Informations générales sur le projet 4](#_Toc127271329)

[1.1 Contexte du projet 4](#_Toc127271330)

[1.2 Description du projet 4](#_Toc127271331)

[1.3 Rappel des principaux objectifs 4](#_Toc127271332)

[1.3.1 Objectifs du projet 4](#_Toc127271333)

[1.3.2 Livrables du projet 4](#_Toc127271334)

[1.3.3 Limites du projet 4](#_Toc127271335)

[1.4 Moyens de réalisation du projet 4](#_Toc127271336)

[1.5 Objectif et Enjeux 5](#_Toc127271337)

[1.5.1 Facteurs de succès liés au projet 5](#_Toc127271338)

[1.5.2 Indicateurs 5](#_Toc127271339)

[2 Organisation du projet 6](#_Toc127271340)

[2.1 Liste des acteurs du projet 6](#_Toc127271341)

[2.1.1 Intervenants du projet 6](#_Toc127271342)

[2.2 RACI 6](#_Toc127271343)

[2.3 Communications au sein du projet 6](#_Toc127271344)

[2.3.1 Documentation du projet 6](#_Toc127271345)

[2.3.2 Signalement des anomalies 6](#_Toc127271346)

[2.3.3 Demandes de changement 7](#_Toc127271347)

[2.4 Procédures de validation des documents de projet 7](#_Toc127271348)

[2.5 Procédures de contrôle et d’actions correctives 8](#_Toc127271349)

[2.5.1 Réunions des structures de projet 8](#_Toc127271350)

[2.5.2 Procédures de contrôle et d’avancement 8](#_Toc127271351)

[2.5.3 Approbation des comptes-rendus 8](#_Toc127271352)

[2.5.4 Réception des résultats 8](#_Toc127271353)

[2.5.5 Droits de modification du Plan Qualité de Projet 9](#_Toc127271354)

[2.5.6 Responsabilités des membres participants au projet 9](#_Toc127271355)

[3 Plan de développement 9](#_Toc127271356)

[3.1 Cycle de vie du projet 9](#_Toc127271357)

[3.1.1 Articulations du plan de développement 9](#_Toc127271358)

[3.2 Matrice des risques 11](#_Toc127271359)

[4 Plan de gestion 13](#_Toc127271360)

[4.1 Méthodes et outils de la conduite de projet 13](#_Toc127271361)

[4.1.1 Choix de Gouti comme outil de gestion 13](#_Toc127271362)

[4.2 Planning général 13](#_Toc127271363)

[4.2.1 Diagramme de Gantt 13](#_Toc127271364)

[4.2.2 Calendrier des réunions des structures du projet 15](#_Toc127271365)

[5 Assurance qualité 16](#_Toc127271366)

[5.1 Méthodes et outils 16](#_Toc127271367)

[5.1.1 Choix de Figma 16](#_Toc127271368)

# Introduction

Présentation de la problématique : Traitement automatique de la langues, Comparaison des performances de plateformes d’analyse linguistique pour les tâches : désambiguïsation morpho-syntaxique (POS -Part Of Speech- tagging, reconnaissance d’entités nommées – Named Entity recognition-)

# Présentation des plateformes d’analyse linguistique

## Stanford

Paragraphe décrivant le fonctionnement de la plateforme Stanford Core NLP (à partir d’articles)

- Approches utilisées : à base de règles, statistique, neuronale, etc.

## NLTK

Paragraphe décrivant le fonctionnement de la plateforme NLTK (à partir d’articles : <http://aclweb.org/anthology-new/W/W08/W08-0208.pdf>, https://aclanthology.org/P06-4018.pdf)

- Approches utilisées : à base de règles, statistique, neuronale, etc.

.

# Evaluation de l’analyse morpho-syntaxique

Afin d’évaluer les deux plateformes sur le plan de l’analyse syntaxique, le corpus utilisé est composé de 481 phrases. Le fichier pos\_reference.txt montre que ce corpus est composé de 10554 mots ou chunk.

Afin d’évaluer les plateformes, le fichier pos\_reference.txt est utilisé comme référentiel. Ce fichier contient une liste des chunks et leur tag. Pour comparer les deux fichiers, les tags sont convertis vers leur équivalents Universel et stockés dans un fichier séparé. Les plateformes sont ensuite utilisées sur le texte pour obtenir l’analyse morpho-syntaxique, et le même processus est appliqué à ces listes de tags.

Une fois que la liste des tags Universel est disponible, on utilise le script evalute.py pour comparer les résultats et évaluer le pourcentage de ressemblance entre notre référentiel et les plateformes d’analyse

## Stanford

# Une image contenant texte, capture d’écran, Police, logiciel Description générée automatiquement

## NLTK

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

## Conclusion

Pour conclure cette analyse, la première chose à remarquer est que les pourcentages sont très faibles. Les pistes expliquant cet écart est tout d’abord le script evaluate.py qui est potentiellement non-fonctionnel. En deuxième lieu, il faut aussi remarquer la différence du nombre de chunk entre notre référence et la liste de tags obtenus après les analyses, ce qui fausse naturellement les calculs.

Cependant, malgré ces marges d’erreur, on remarqque que NLTK est 3 fois plus précis que Stanford. Mais ces mesures sont à prendre avec du recul, puisque avec son avance, NLTK ne ressemble qu’à 12% de notre fichier de référence.

# Evaluation de la reconnaissance d’entités nommées

## Stanford

Tableau des résultats

## NLTK

Tableau des résultats

# Points forts, limitations et difficultés rencontrées

Bla bla

# Organisation

Qui a fait quoi ?

* TPs
* Github du projet
* Code du projet
* Rédaction du rapport
* Etc.

# Annexes

**7. Annexes**

- Expérimentations en plus

- Corrections au niveau des corpus

- Captures d’écran des résultats

- Etc.