CONNL-U PROJET IDL

CAHIER DES CHARGES

Pour une application gérant les fichiers CoNNL-U.

AVRIL 2023

Préparé par

OVSEV BELIZ OZKAN
BEINING YANG

Approuvé par

OLIVIER KRAIF

Table des matières

Contexte et objectifs 3
Fonctionnalités principales 3
Exigences techniques 3
Livrables 4
Planning prévisionnel 4
Conclusion 5

Contexte et objectifs

Le but de ce projet est de développer une série de fonctions Python pour fusionner les résultats de différents analyseurs, tels que Spacy, UDPipe et Stanza, dans un même fichier CoNLL-U. Les fonctions développées seront capables d'intégrer des colonnes telles que LEMMA, POS, FEA et DEPREL dans le fichier CoNLL-U.

Ce cahier des charges a pour but de définir les besoins, le design et les fonctionnalités de l'application, ainsi que les exigences techniques et les contraintes de développement.

Fonctionnalités principales

- Développer une fonction de fusion pour intégrer les sorties des différents outils d'analyse.
- Développer des fonctions wrapper pour chacun des outils d'analyse (Spacy, UDPipe, Stanza).
- Créer une fonction de tokenisation pour les fichiers XML.
- La fonction de tokenisation devra fournir des informations de position et de présence d'espaces pour chaque token.
- Définir des "processors" à la carte pour l'analyse des fichiers CoNLL-U.

Exigences techniques

- Les fonctions développées doivent être écrites en Python.
- Les fonctions wrapper pour les outils d'analyse doivent être capables de travailler avec des entrées et des sorties compatibles avec le format CoNLL-U.
- La fonction de tokenisation pour les fichiers XML doit être capable d'identifier le texte à traiter à l'aide d'une formule XPATH.
- La fonction de tokenisation doit fournir des informations de position et de présence d'espaces pour chaque token dans le fichier CoNLL-U.
- Les "processors" doivent pouvoir être définis de manière personnalisée pour l'analyse des fichiers CoNLL-U.

Livrables

- **Mini-cahier des charges**

  Il s'agit d'un document décrivant les exigences fonctionnelles de l'application, les exigences techniques, l'interface, les livrables ainsi que le planning prévisionnel.

- **Scripts**

  Un ensemble de fonctions Python pour fusionner les résultats des différents outils d'analyse dans un fichier CoNLL-U, des fonctions wrapper pour chacun des outils d'analyse (Spacy, UDPipe, Stanza), un main qui permet d'exécuter le code et une fonction de tokenisation pour les fichiers XML qui fournit une tokenisation CoNLL-U avec des informations de position et de présence d'espaces pour chaque token.

- **GitLab**

  Tous les éléments du projet, y compris le mini-cahier des charge et les scripts doivent être gérés et déposés sur GitLab. Il doit y avoir un répertoire dédié pour le projet qui comprend un fichier README.md décrivant le contenu de chaque fichier déposé.

Les livrables doivent être bien documentés et facilement compréhensibles pour que toute personne puisse comprendre et utiliser l'application.

Planning prévisionnel

- **Phase d'imprégnation (2 semaines)**

  Étude des différentes bibliothèques d'analyse, définition des fonctions nécessaires pour chaque bibliothèque.

- **Rédaction du mini-cahier des charges (2 semaines)**

  Définir et documenter les exigences fonctionnelles de l'application, les exigences techniques, l'interface, les livrables ainsi que le planning prévisionnel.

- **Développement et tests (x semaines)**

  - Semaine 1-2 : Développement de la fonction de tokenisation pour les fichiers XML.
  - Semaine 3-5 : Développement des fonctions wrapper pour chaque bibliothèque.

- Semaine 6-8 : Développement de la fonction de fusion pour intégrer les sorties des différentes bibliothèques dans un fichier CoNLL-U.

- Semaine 9-10 : Tests des différentes fonctions et intégration des "processors" personnalisés.

- Semaine 11-12 : Finalisation des livrables et rédaction des instructions.

- **Gestion et suivi du projet sous GitLab (tout au long du projet)**

  Gérer les versions, les branches, les merges, les commits, les tests, et les commentaires.

- **Démonstration à l'oral du travail réalisé à l'équipe enseignante (x semaines) :**

  Préparer une présentation du projet, démontrer les différentes fonctionnalités, et répondre aux questions.

Ce planning prévisionnel est donné à titre indicatif et peut être ajusté en fonction de la complexité des fonctionnalités, des imprévus, des contraintes techniques, et des disponibilités des membres de l'équipe.

Conclusion

Ce projet consiste à développer des fonctions Python pour intégrer les sorties de différents outils d'analyse dans un fichier CoNLL-U. Les fonctions développées incluent des wrappers pour les outils d'analyse, une fonction de tokenisation pour les fichiers XML et une fonction de fusion pour intégrer les sorties. Les "processors" personnalisés peuvent être définis pour l'analyse des fichiers CoNLL-U.