Scripts PALS pour la textométrie

Mot d'introduction

Le but de cette feuille est de créer **un nouveau script** afin de formater votre corpus afin de le rendre lisible par PALS [1]. Il s'agit d'un ensemble de scripts pour effectuer de la textométrie qu'on utilisera comme base pour l'analyse des données.

Il faudra créer divers fichiers au cours de ce TD, à ranger selon l'architecture de dossiers suivante, qui enrichit celle déjà existante évoquée dans les feuilles précédentes. En **bleu** les dossiers censés exister au début de cette feuille, en *rouge* les dossiers à créer pour la séance. Les noms de type `lang1-1.txt` reprennent le nom du fichier d'URL correspondant :

- URLs
  - lang1.txt
  - lang2.txt
- ...
- contextes
  - lang1-1.txt
  - lang1-2.txt
  - ...
  - lang2-1.txt
  - lang2-2.txt
  - ...
- dumps-text
  - lang1-1.txt
  - lang1-2.txt
  - ...
  - lang2-1.txt
  - lang2-2.txt
  - ...
- *pals*
  - contextes-lang1.txt
  - contextes-lang2.txt
  - dumps-text-lang1.txt
  - dumps-text-lang2.txt

Note 1 Nous travaillerons ici sur les fichiers de dump et de contextes. Il ne sera pas demandé d'avoir une connexion internet.

Note 2 Il est possible de vouloir comparer différents fichiers pour la même langue. On ne présente ici qu'une version minimale de l'attendu.

... PALS ?

PALS (*Python Autonomous Lafon Specificity Scripts*) sont des scripts python qui peuvent être utilisés avec uniquement un interpréteur python (aucune dépendance). Ces scripts permettent d'effectuer le calcul de spécificité de Lafon. [2] Cette spécificité mesure, comme son nom l'indique, la spécificité d'une forme dans une partie d'un corpus. Cette partie peut être issue d'un partitionnement quelconque (chapitres, livres, etc.) ou d'un découpage plus spécifique (les alentours d'un mot). À chacun de ces cas correspond un script : `partition.py` et `cooccurrents.py`. On illustrera principalement les cooccurrents ici.

Ces scripts permettent donc de lancer quelques analyses textométriques sur un corpus de manière très simple et configurable. Il est possible d'utiliser des expressions régulières afin de matcher différentes formes graphiques.

Les scripts peuvent se récupérer via GitHub à l'adresse suivante : `https://github.com/YoannDupont/PALS`

1. `https://github.com/YoannDupont/PALS`
2. Lafon, P. (1980). Sur la variabilité de la fréquence des formes dans un corpus. Mots. Les langages du politique, 1(1), 127-165.

Les scripts disposent d'une aide qui se veut assez complète. Vous pouvez les lancer dans un premier temps avec l'option `-h` ou `--help` pour afficher l'aide.

Exercice 1 Exemples de fichiers pour PALS

Des fichiers d'exemple pour PALS sont disponibles sur le git ainsi que sur icampus. Vous pourrez vous en servir comme base pour créer la structure de votre fichier. Le format de fichier attendu par PALS est le suivant : un mot par ligne, possibilité de séparer en phrases avec une ligne vide. Un exemple très court serait ce qui suit :

```
contenu
du
fichier
1

Et
du
fichier
2
!
```

L'idée est de créer un fichier par langue afin d'effectuer des analyses quantitatives dessus pour trouver des éléments notables pour ensuite effectuer une analyse qualitative. Vous pouvez éventuellement créer un fichier par URL pour faire potentiellement des analyses plus fines.

Exercice 2 Travail sur les dumps textuels

Créez avant de lancer les scripts suivants le dossier `pals`.

Créez un script `make_pals_corpus.sh`. Ce script prendra en argument :
- un dossier
- un nom de base qui correspond au nom du fichier URL sans son extension (`lang1` ou `lang2` dans l'exemple plus haut)

Le script devra créer un fichier au format attendu par PALS qui prendra la forme indiquée dans l'Exercice 1. Il faut itérer sur les différents fichiers dump (1 par URL), les prétraiter et les écrire sur la sortie standard (on pourra faire une redirection après). Pour le nom de base `lang1`, le fichier écrit aura le nom `dump-lang1.txt` et sera placé dans le dossier `pals`.

Important Pour lancer les scripts `PALS`, la tokenisation devra être faite en avance, par vos soins. La tokenisation a une influence sur le résultat final, mais on veut surtout pouvoir lancer les scripts sur vos données.

Exercice 3 Travail sur les contextes

Refaites les manipulations précédentes, mais en les appliquant aux fichiers de contexte. Les fichiers créés devront avoir des noms de type `contexte-lang1.txt` et le fichier final sera `contexte.txt` dans le dossier *pals*.