LETTRES SORBONNE UNIVERSITÉ

2024-2025

Programmation de Modèles Linguistiques (I)

(L5SOPROG L3 Sciences du Langage)

Examen Terminal 2024-2025

Caroline Koudoro-Parfait, Sorbonne Université.
Crédit : Gaël Lejeune, Sorbonne Université

- Durée 2h, tous documents autorisés
- Dépôt: 1 fichier NUMETU_X.ipynb pour chaque exercice (où NUMETU est votre numéro d'étudiant et X le numéro de l'exercice) + fichiers complémentaires pour l'exercice 3
- Afficher un résultat signifie utiliser un `print()`
- Sauvegarder un résultat signifie l'enregistrer dans un fichier

Vous démarrez votre activité de linguiste-informaticien chez Daniel, une entreprise un peu ancienne de la *French Tech*. c'est votre premier jour, sans vous mettre la pression il faut pas trop se louper. On n'a jamais deux fois l'occasion de faire une bonne première impression.

Votre première tâche consiste à reprendre un projet laissé par un ancien stagiaire de chez Daniel. Il y a trois parties indépendantes que vous pouvez faire dans l'ordre que vous souhaitez. Chaque exercice correspondra à un notebook Ipynb indépendant. Vous devrez donc avoir rendu 1 fichier python par exercice.

Exercice 1 : Commentaire de code (5 pts)

POur cet exercice vous auerz besoin des fichiers `Ex1.ipynb` et `FRA00701_Carraud.txt`. Commentez (en une phrase) le code à chaque ligne marquée par "###".

A déposer pour cet exercice :

- `NUMETU_1.ipynb` (où NUMETU est votre numéro d'étudiant)

Exercice 2 : Correction de code (7 pts)

Pour cet exercice vous aurez besoin des fichiers `Ex2.ipynb` et `DAUDET.txt`.

L'entreprise veut exploiter des représentations informatiques de textes. Ce travail a été demandé à un des anciens stagiaires mais malheureusement il n'a pas bien fini le travail. Vous devrez donc corriger le code pour qu'il respecte les instructions données :

Afin d'exploiter des représentations informatiques de textes nous avons besoin d'analyser rapidement un texte en langue française. Effectuez les opérations suivantes :

1. *Lisez le fichier* ***`DAUDET.txt`*** *et transformez tous les lettres en minuscule*
2. *Découpez le texte en mots*

3. *Affichez la taille du texte en mots et en caractères*
4. *Calculez la fréquence de chaque mot et stockez le tout dans un dictionnaire* PYTHON
5. *Affichez la taille du vocabulaire du texte*
6. *Extrayez les 15 mots les moins fréquents en donnant pour chacun le nombre d'occurrences*
7. *Extrayez les 20 mots les plus fréquents en donnant pour chacun le nombre d'occurrences*

Le code à corriger corresponds au notebook `Ex2.ipynb` disponible sur Moodle. En plus des corrections nécessaires, ajoutez en commentaire dans le code une ou deux propositions (en 2 phrases) visant à améliorer l'identification des mots décrivant vraiment le thème du texte (techniques vues en cours ou hypothèses personnelles)

A déposer pour cet exercice :

- `NUMETU_2.ipynb` (où NUMETU est votre numéro d'étudiant)

Exercice 3 : Json et Indexation (8 pts)

DANIEL vous demande d'exploiter maintenant le fichier `DAUDET.json`. NB : dans cet exercice vous n'avez pas de code de base.

1. Ouvrez ce fichier de manière appropriée et affichez son contenu
2. Dans une nouvelles cellule, expliquez en commentaires (en 2 phrases) la différence entre traiter un fichier `txt` et un fichier `json`
3. Indiquez pour chacun des mots suivants, la liste des phrases (par ex. Phrases_4, Phrases_7...) où il apparaît : "soleil", "maison", "révolution", "ruine"
4. Sauvegardez ce résultat sous forme dans un fichier `index.json` qui associe à chacun des mots précités la liste des phrases où ce mot apparaît
5. Ajoutez dans votre index le mot "ville" (qu'il soit au singulier ou au pluriel)
6. Sauvegardez ce résultat sous forme dans un fichier `index2.json`
7. Ajoutez dans votre index tous les mots contenant un "a"
8. Sauvegardez ce résultat sous forme dans un fichier `index3.json`
9. Ajoutez dans votre index tous les mots de plus de 4 lettres
10. Sauvegardez ce résultat sous forme dans un fichier `index4.json`

A déposer pour cet exercice :

- `NUMETU_3.ipynb` (où NUMETU est votre numéro d'étudiant)
- `index.json`
- `index2.json`, `index3.json`, `index4.json`