Lexicologie: mots clés et bigrams

Kata Gábor

30 Janvier 2024

Exercices.

1. Ouvrez et complétez le fichier exo1.1.py selon les instructions en commentaire pour préparer une liste de fréquence de mots et une liste de bigrams à partir d'un corpus.

2. Complétez exo1.1.py pour écrire les deux listes (sortie standarde) un mot par ligne, en ordre décroissant selon la fréquence. Exécutez-le avec le fichier acl.txt en entrée. Sauvegardez les résultats.

3. Parcourez les 100 mots unigram les plus fréquents. Ne gardez que les mots grammaticaux, supprimez les mots lexicaux et les informations de fréquence. Nous cherchons à obtenir une liste des stopwords fréquents, un mot par ligne.

4. Modifiez exo1.py de façon à ce qu'il lise le fichier de stopwords, et lors du calcul il ignore les mots stopwords et les bigrams qui contiennent un stopword. (Attention : les variables N et B devront changer aussi par rapport à 1) !). Re-exécutez exo1.py avec acl.txt et comparez les bigrams les plus fréquents.

5. Complétez exo1.py avec une fonction de normalisation "normalize(word)" qui prend un mot en entrée et — transforme les majuscules en minuscules — coupe les ponctuations attachées Vous pouvez utiliser le module re pour la substitution avec une expression régulière. Appelez la fonction à l'endroit approprié dans exo1.py (attention : le nombre des types de mots devra changer) et ré-exécutez. Faites en sorte qu'il ne reste pas de mots qui correspondent à des strings vides après la suppression des ponctuation, ni parmi les mots unigram, ni parmi les composants de bigrams.

6. Complétez exo1.py de façon à ignorer les mots et les bigrams avec une fréquence inférieure à 5 (attention : N et B changent encore).