TP 3
Expansion de requête et ranking

Présentation du sujet

Vous partirez de l'index créé au TP précédent ou de ceux fournis au format json.

Il faudra créer un système qui :
- lit une requête de l'utilisateur
- La tokenise et la transforme pour filtrer les documents qui ont tous les tokens de la requête
- Créer une fonction de ranking linéaire pour ordonner les documents qui auront survécu au filtre. Cette fonction linéaire peut prendre en compte plusieurs features comme par exemple une importance sur le compte des tokens dans les documents ou leurs positions.
- Renvoyer une liste de documents au format json (**results.json**), chaque document étant présenté comme suit:
  - Titre
  - Url

  En plus des documents, les résultats devront présenter le nombre de documents dans l'index et le nombre de documents qui ont survécu au filtre.

Attention, pour rappel, tous prétraitement fait sur la requête doit être le meme que celui fait sur les documents.
Si vous n'avez pas fini le TP2 et que vous utilisez les index fournis, il a été appliqué un prétraitement minimal sur les documents:
- une tokenization avec un split sur les espaces
- Tous les tokens ont été lowerisés

En bonus

- Modifier la fonction qui transforme la requête pour qu'elle prenne en compte non pas les documents qui ont tous les tokens de la requête, mais ceux qui en ont au moins un. Faites en sorte que ce soit un paramètre et que la personne qui lance le programme puisse choisir entre un filtre de type ET ou un filtre de type OU.
- Appliquer un poids plus important aux tokens qui ont du sens par rapport aux stop words au moment du ranking
- Calculer le score de bm25 dans votre fonction de ranking

Ce qui est demandé

Un dossier avec votre index écrit en **python**.
Le code devra s'executer dans un fichier **main.py** à la racine du projet.
Vos fonctions publiques devront être testées dans un fichier **tests.py**.

L'explication du code et de son execution devra être décrite dans un fichier **README.md** à la racine du projet. Ce fichier devra aussi comprendre le nom des contributeurs du projet.

Si vous êtes familier avec GitHub, vous pouvez aussi m'envoyer le lien vers votre repository.

Ce qui est fourni

2 fichiers au format json:

- **documents.json**, chaque dictionnaire contient 3 informations:
  - L'url du document,
  - Son id
  - Le titre extrait de l'HTML
- **index.json**, chaque dictionnaire est de forme:
  - Token: {docId: {position: [position indices], count: int}

Les librairies dont vous aurez besoin

- Pour lire et écrire des fichier json: https://docs.python.org/fr/3/library/json.html
- Pour appliquer du prétraitement sur votre requête: https://www.nltk.org/