

Programmation de spécialité (python)

TD 7 : implémentation d'un moteur de recherche

Julien Velcin

2024-2025

L'objectif de ce TD est de vous apprendre à implémenter votre propre moteur de recherche. Ce moteur se basera uniquement sur la présence de mots-clefs correspondant à une requête entrée par l'utilisateur. Il sera basé sur la construction manuelle d'une matrice Documents x Termes. A nouveau, cette matrice ne doit pas être construite automatiquement par une librairie externe comme `scikit-learn`.

Partie 1 : matrice Documents x Mots

1.1 Tout d'abord, il convient de construire un dictionnaire qui comporte les mots servant à décrire le texte de vos documents. Pour cela, vous pouvez utiliser ce que vous avez fait dans le TD précédent à partir du découpage des chaînes en mots, sans oublier de retirer les doublons et tirer par ordre alphabétique. Vous appellerez ce dictionnaire **vocab**. Les clefs sont les mots eux-mêmes et la valeur est un nouveau dictionnaire qui contient plusieurs informations sur le mot (son identifiant unique, son nombre total d'occurrence...).

1.2 Il faut à présent parcourir votre collection de documents afin de construire une matrice de dimension Nombre de documents x Nombre de mots dans le vocabulaire. A l'intersection de la ligne i (document) et de la colonne j (mot), vous placerez le nombre d'occurrences du mot dans le texte (ie. le Term Frequency ou TF). La matrice étant très creuse, l'idéal serait d'utiliser une classe spécialement conçue pour ça, comme `sparse.csr_matrix` de la bibliothèque `scipy`. Vous appellerez cette matrice **mat_TF**.

1.3 A partir de cette matrice, pour chacun des mots, calculez le nombre total d'occurrences dans le corpus et le nombre total de documents contenant ce mot. Vous stockerez cette information dans **vocab**.

1.4 Une alternative intéressante au score TF est la mesure $TF \times IDF$ (cf. <https://fr.wikipedia.org/wiki/TF-IDF>). Calculez une deuxième matrice, appelée **mat_TFIDF**, qui implémente cette mesure.

Partie 2 : moteur de recherche

Afin de réaliser votre moteur de recherche, les principales étapes sont les suivantes :

- demander à l'utilisateur d'entrer quelques mots-clefs,
- transformer ces mots-clefs sous la forme d'un vecteur sur le vocabulaire précédemment construit,
- calculer une similarité entre votre vecteur requête et tous les documents,

- trier les scores résultats et afficher les meilleurs résultats.

La similarité peut être calculée à l'aide d'un simple produit scalaire entre le vecteur requête et le vecteur du texte visé. Une mesure qui est souvent plus appropriée est celle du cosinus (cf. https://fr.wikipedia.org/wiki/Similarite_cosinus).

Partie 3 : intégrer le tout dans une classe

Pour finir, vous devez intégrer tout le code produit dans ce TD dans une classe intitulée **SearchEngine**. Cette classe doit respecter certaines contraintes :

- On doit pouvoir donner un objet de type **Corpus** lorsqu'on instancie le moteur. La construction de la matrice Documents x Termes doit se faire dans la foulée.
- La classe doit proposer une fonction **search** avec deux arguments : les mots clefs de la requête et le nombre de documents à retourner à l'utilisateur.
- Le résultat de la recherche doit être retournée sous la format d'une table **DataFrame** de **pandas**.