2023/2024

eFrei
PARIS PANTHÉON-ASSAS UNIVERSITÉ

MY FIRST CHATBOT

Rapport du projet semestriel de programmation python de L1 à L'EFREI

Groupe de TD B

Noah Guichard et Gabriel Geniez

TABLE DES MATIÈRES

INTRODUCTION

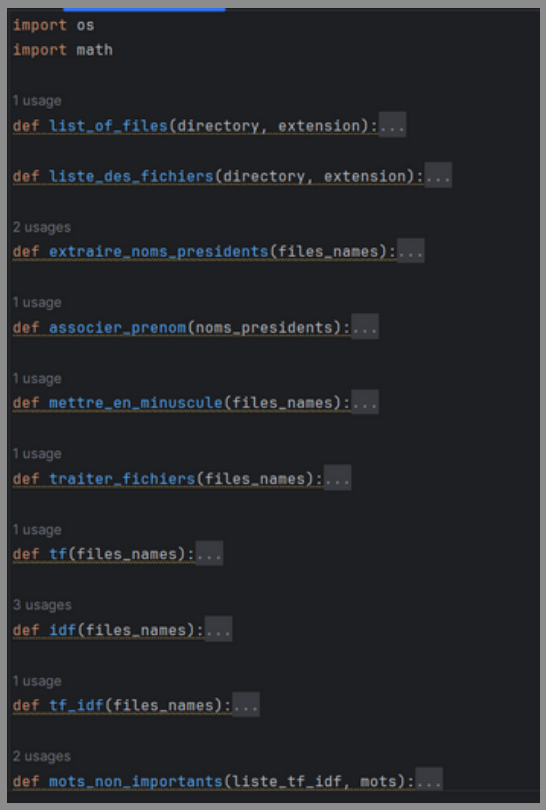
L'analyse de texte est un domaine qui vise à extraire des informations pertinentes à partir de documents écrits en langage naturel. Elle peut avoir de nombreuses applications, telles que la recherche d'information, la classification de documents, la génération de résumés, ou encore la création de chatbots. Un chatbot est un système capable de dialoguer avec un utilisateur humain en lui fournissant des réponses adaptées à ses questions. Pour cela, il doit être capable de comprendre le sens des questions et de trouver les informations pertinentes dans un corpus de textes.

Dans ce projet, nous allons nous intéresser à une méthode simple mais efficace pour développer un chatbot basé sur la fréquence des mots dans le corpus. Cette méthode repose sur le calcul d'un score appelé TF-IDF (term frequency-inverse document frequency) qui mesure l'importance d'un mot dans un document par rapport à l'ensemble du corpus. En utilisant ce score, nous allons pouvoir représenter les questions et les documents sous forme de vecteurs, et calculer leur similarité à l'aide du cosinus. Ainsi, nous pourrons identifier les documents les plus proches de la question et sélectionner la meilleure réponse possible.
L'objectif de ce projet est de mettre en pratique les concepts de base de la programmation Python, tels que les structures de données, les boucles, les fonctions, les modules, et les fichiers. A la fin de ce projet, nous aurons réalisé un chatbot fonctionnel capable de répondre à des questions sur un corpus de textes donné.

LES FONCTIONS DU PROJET

Dans ce projet nous avons réalisé toutes les fonctions de la partie 1, ainsi que les fonctions de la partie 2 jusqu'à la question 5. Voici la liste de nos fonctions que nous avons pu coder.

Ces différentes fonctions avait pour rôle de d'abord nettoyer le texte de sa ponctuation pour rendre le texte utilisable pour répondre aux questions



Ensuite on a construit une matrice TF-IDF pour la suite du projet.
Et enfin on a crée de petites fonctions pour répondre aux différentes question.

PRÉSENTATION TECHNIQUE

Le principal algorithme de notre projet notre matrice TF-IDF (Term Frequence-Inverse Document Frequency). Sans trop rentrer dans les détails, grâce à la fréquence des termes dans chaque texte et grâce à l'inverse de la fréquence du document qui mesure l'importance d'un terme dans l'ensemble du corpus, on calcule le poids TF-IDF.

```
def afficher_menu():
    print("1. Afficher la liste des mots les moins importants dans le corpus de documents.")
    print("2. Afficher le(s) mot(s) ayant le score TD-IDF le plus élevé.")
    print("3. Indiquer le(s) mot(s) le(s) plus répété(s) par le président Chirac.")
    print("4. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé de la  Nation et celui qui l'a répété le plus de fois")
    print("5. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé du climat et/ou de l'écologie")



1 usage
def principal():
    directory = "./speeches"
    nom_fichiers = list_of_files(directory, extension: "txt")
    mettre_en_minuscule(nom_fichiers)
    traiter_fichiers(nom_fichiers)
    noms_presidents = extraire_noms_presidents(nom_fichiers)
    mots = tf(nom_fichiers)
    idf(nom_fichiers)
    liste_tf_idf = tf_idf(nom_fichiers)
    mots_non_importants(liste_tf_idf, mots)
    mots_importants(liste_tf_idf, mots)
    noms_presidents = extraire_noms_presidents(nom_fichiers)
    prenoms_presidents = associer_prenom(noms_presidents)
    chirac_index = prenoms_presidents.index("Jacques Chirac")
    chirac_tfidf_scores = liste_tf_idf[chirac_index]

    # Récupérer les mots les plus fréquents pour Chirac (en excluant les mots non importants)
    mots_importants_chirac = mots_importants( liste_tf_idf: [chirac_tfidf_scores], mots)
```

la matrice TF-IDF assigne des poids à chaque terme dans chaque document en tenant compte à la fois de la fréquence du terme dans le document (TF) et de son importance dans le corpus (IDF).

```
continuer = True
while continuer:
    print()
    print()
    afficher_menu()
    choix = input("Entrez le numéro de l'action que vous souhaitez effectuer (0 pour quitter) : ")
    if choix == "1":
        print(mots_non_importants(liste_tf_idf, mots))
    elif choix == "2":
        print(mots_importants(liste_tf_idf, mots))
    elif choix == "3":
        print("Les mots les plus fréquents pour Jacques Chirac (hors mots non importants) :", mots_importants_chirac)
    elif choix == "4":
        resultat = nation(nom_fichiers)
        print(f"les fichier qui ont le termes 'nation' sont :  {resultat[0]}")
        print(f"Le fichier avec le plus grand nombre d'occurrences du mot 'nation' est : {resultat[1]}")
        print(f"Nombre total d'occurrences du mot 'nation' dans ce fichier : {resultat[2]}")
    elif choix == "5":
        print(climat(nom_fichiers))
    elif choix == "6":
        documents = str(input("Saisir une question :"))
        questions = nettoyer_et_tokeniser(documents)
        tf_dict = calculer_tf_question(questions)
        print(calculer_tfidf_question(nom_fichiers, documents, tf_dict))
    elif choix == "0":
        print("fin")
        continuer = False
    else:
        print("Choix invalide. Veuillez entrer un numéro valide.")
```

PRÉSENTATION DES RÉSULTATS

A l'issue du projet nous avons obtenus divers résultats et nous avons crée une interface simple pour les utilisateurs

INTERFACE UTILISATEUR

Nous avons opté pour une interface simple et inclusive que voici pour répondre aux questions:

```
1. Afficher la liste des mots les moins importants dans le corpus de documents.
2. Afficher le(s) mot(s) ayant le score TD-IDF le plus élevé.
3. Indiquer le(s) mot(s) le(s) plus répété(s) par le président Chirac.
4. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé de la  Nation et celui qui l'a répété le plus de fois
5. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé du climat et/ou de l'écologie
Entrez le numéro de l'action que vous souhaitez effectuer (0 pour quitter) :
```

EXEMPLE D'APPLICATION

Pour illustrer l'utilisation du chatbot voici un exemple d'une réponse qu'il fait quand on lui pose la question 4:

```
1. Afficher la liste des mots les moins importants dans le corpus de documents.
2. Afficher le(s) mot(s) ayant le score TD-IDF le plus élevé.
3. Indiquer le(s) mot(s) le(s) plus répété(s) par le président Chirac.
4. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé de la  Nation et celui qui l'a répété le plus de fois
5. Indiquer le(s) nom(s) du (des) président(s) qui a (ont) parlé du climat et/ou de l'écologie
Entrez le numéro de l'action que vous souhaitez effectuer (0 pour quitter) : 4
les fichier qui ont le termes 'nation' sont :  ['Nomination_Chirac1.txt', 'Nomination_Chirac2.txt', 'Nomination_Hollande.txt', 'Nomination_Macron.txt', 'Nomination_Mitterrand2.
Le fichier avec le plus grand nombre d'occurrences du mot 'nation' est : Nomination_Chirac2.txt
Nombre total d'occurrences du mot 'nation' dans ce fichier : 4
```

CONCLUSION

Ce projet nous a permis de réaliser un chatbot basé sur la fréquence des mots dans un corpus de textes. Nous avons appris à utiliser la méthode TF-IDF pour représenter les questions et les documents sous forme de vecteurs, et à calculer leur similarité à l'aide du cosinus.
Ce projet nous a également permis de développer nos compétences sur le plan technique, mais aussi sur le plan organisationnel et communicationnel. Nous avons travaillé en équipe, en répartissant les tâches et en respectant les délais. Nous avons communiqué régulièrement, en utilisant GitHub. Nous avons également documenté notre code et rédigé un rapport détaillé sur notre démarche et nos résultats.
Ce projet a été une expérience enrichissante et formatrice, qui nous a fait découvrir le domaine de l'analyse de texte et ses applications. Nous avons pu mettre en pratique les concepts de base de la programmation Python, tout en explorant une méthode simple mais efficace pour créer un chatbot. Nous sommes satisfaits du résultat obtenu, qui répond aux objectifs fixés. Nous sommes également conscients des limites de notre méthode, qui pourrait être améliorée en utilisant des techniques plus avancées, telles que les réseaux de neurones.