**Documentation du Script de Téléchargement d'Articles ArXiv**

Ce script Python a pour objectif de télécharger automatiquement les fichiers PDF d'articles académiques à partir du site ArXiv en se basant sur les catégories des articles. Il suit les étapes suivantes :

**1. Chargement des IDs des articles à partir du fichier JSON:**

- Le script commence par ouvrir un fichier JSON nommé "data.json" qui contient des données sur les articles académiques, chaque article étant représenté par un objet JSON sur une ligne.

- Il extrait l'ID de chaque article ainsi que ses catégories.

- Seuls les articles appartenant à la catégorie "cs.AI" (Intelligence Artificielle) sont retenus.

**2. Préparation des URLs des PDFs:**

- Le script génère les URLs complètes des fichiers PDF à télécharger en combinant l'ID de chaque article avec l'URL de base d'ArXiv.

**3. Définition du dossier de destination :**

- Le dossier de destination où les fichiers PDF seront enregistrés est défini comme "AI\_articles\_pdfs".

**4. Vérification de l'existence du dossier de destination :**

- Le script vérifie si le dossier de destination existe. S'il n'existe pas, il le crée automatiquement.

**5. Téléchargement des PDFs :**

- Le script parcourt la liste des URLs des PDFs à télécharger.

- Pour chaque URL, il extrait le nom du fichier à partir de l'URL et ajoute l'extension ".pdf".

- Ensuite, il utilise la bibliothèque urllib pour télécharger le PDF depuis l'URL et l'enregistre dans le dossier de destination.

Le script affiche des messages de succès ou d'échec pour chaque téléchargement.

Pour exécuter le script, assurez-vous d'avoir un fichier JSON nommé "data.json" contenant les données des articles et que vous disposez des modules Python nécessaires (json, urllib.request, os).

Pour personnaliser le script, vous pouvez modifier le nom du fichier JSON d'entrée, la catégorie cible, l'URL de base ou le dossier de destination.

**Documentation de la deuxième partie du Script : extraction du contenu des fichiers pdf**

Ce script Python a pour objectif de traiter des données d'articles ArXiv, notamment l'extraction de texte à partir de fichiers PDF, le nettoyage du texte, la fusion de données JSON et la création d'un dataset final.

**Étape 1 : Extraction de texte à partir de fichiers PDF**

- Le script commence par spécifier le chemin du dossier contenant les fichiers PDF à traiter (`pdf\_folder\_path`).

- Il itère à travers les fichiers PDF dans ce dossier, extrait le texte de chaque fichier, et le stocke dans la liste `extracted\_texts`.

- Les IDs correspondants sont extraits des noms de fichiers et stockés dans la liste `ids`.

**Étape 2 : Création d'un DataFrame**

- Les données extraites sont utilisées pour créer un DataFrame (`df`) avec les colonnes "ID" et "Texte Extrait".

- Le DataFrame est ensuite enregistré dans un fichier CSV appelé "contenu\_extracted\_with\_ids.csv".

**Étape 3 : Nettoyage du texte**

- Le script lit le DataFrame précédemment créé à partir du fichier CSV.

- Il applique une fonction de nettoyage au texte extrait, qui supprime les caractères non ASCII, les caractères indésirables, et convertit le texte en minuscules.

- Les données nettoyées sont enregistrées dans un nouveau fichier CSV appelé "contenu\_cleaned.csv".

**Étape 4 : Chargement de données JSON**

- Le script spécifie le chemin vers un fichier JSON contenant des données ArXiv (`json\_file\_path`).

- Il charge les données JSON depuis ce fichier et les stocke dans un DataFrame (`df`).

**Étape 5 : Fusion de données**

- Le script sélectionne les colonnes pertinentes du DataFrame JSON et les enregistre dans un fichier CSV appelé "selected\_information.csv".

- Il charge les deux datasets CSV créés précédemment ("contenu\_cleaned.csv" et "selected\_information.csv").

- Les deux datasets sont fusionnés en utilisant l'ID en commun pour créer un nouveau dataset final.

- Le nouveau dataset final est enregistré dans un fichier CSV appelé "final\_dataset.csv".

**Prérequis**

Assurez-vous d'avoir les bibliothèques Python nécessaires installées, telles que pandas et les bibliothèques pour l'extraction de texte depuis les fichiers PDF.

**Personnalisation**

- Vous pouvez personnaliser les chemins des fichiers d'entrée et de sortie en modifiant les variables appropriées (`pdf\_folder\_path`, `json\_file\_path`, noms de fichiers CSV de sortie, etc.).

- Vous pouvez également personnaliser la fonction de nettoyage du texte en fonction de vos besoins spécifiques.

**Auteur**

* Hajji chakir Zouhair
* Chebbab Aya
* Houssam Meryem
* Msalek Aicha
* Rakhiss Bouchra
* Elouafi Abderrahmane