DÉVELOPPEUR DATA

SCRAPING GITHUB

Collecte des données via web Scraping

Réalisé par : OMAR DBAA

16 /06/2023

Introduction

Dans le cadre de ce projet, nous nous intéressons à l'obtention d'informations précieuses à partir des dépôts de code hébergés sur GitHub. Le contexte de notre projet repose sur la nécessité de disposer d'une connaissance approfondie des dépôts de code hébergés sur GitHub. Il s'agit d'une plateforme renommée qui regorge d'informations précieuses pour notre entreprise. En comprenant les tendances du développement, les langages les plus utilisés et les projets intéressants présents sur GitHub, nous pourrons prendre des décisions stratégiques éclairées et d'explorer de potentielles opportunités de collaboration.

Objectifs

Les objectifs de ce projet sont les suivants :

- Comprendre les tendances de développement : Nous souhaitons analyser les dépôts de code pour identifier les tendances actuelles du développement logiciel. Cela nous permettra de rester informés des évolutions technologiques et des bonnes pratiques.
- Identifier les langages de programmation les plus utilisés : Il est essentiel de connaître les langages de programmation les plus populaires sur GitHub. Cela nous permettra de mieux cibler nos ressources et d'adapter nos stratégies de développement.
- Trouver des projets intéressants : Nous voulons repérer des projets novateurs et prometteurs sur GitHub. Cela peut nous ouvrir de nouvelles opportunités de collaboration.

- Suivre les évolutions technologiques : GitHub est un reflet des avancées technologiques. En surveillant les nouveaux projets et les mises à jour, nous pourrons anticiper les changements et rester à la pointe de la technologie.

Les Étapes

1. Compréhension du Contexte : La première étape consiste à comprendre pleinement le contexte et les objectifs du projet. Cela implique une analyse approfondie des besoins de l'entreprise.
2. Recherche et Documentation : Une recherche approfondie est effectuée pour trouver les meilleures méthodes et bibliothèques permettant de collecter les données depuis GitHub. La documentation pertinente est rassemblée et analysée pour assurer une compréhension approfondie des mécanismes de collecte de données sur GitHub.
3. Définition de Méthodologie de Collecte de Données : Une méthodologie de collecte de données est mise en place en utilisant les outils et les approches identifiés lors de la phase de recherche. Des scripts Python peuvent être développés pour effectuer le scraping des données depuis GitHub de manière automatisée et efficace.
4. Surmonter les Défis : Tout au long du projet, des défis peuvent être rencontrés, tels que la gestion des taux limites imposés par l'API de GitHub. Des solutions sont adoptées pour surmonter ces défis, comme l'utilisation de stratégies de limitation de la fréquence des requêtes ou la mise en place de mécanismes de pagination pour récupérer toutes les données nécessaires.

Défis Rencontrés

Plusieurs défis ont été relevés lors de la réalisation de ce projet, notamment :

1. Gestion des taux limites de l'API de GitHub : L'API de GitHub impose des limites sur le nombre de requêtes pouvant être effectuées dans une certaine période. Il a donc été nécessaire de mettre en place des stratégies pour gérer ces taux limites afin d'éviter les restrictions d'accès et de garantir une collecte de données continue et efficace.
2. Traitement de volumes importants de données : GitHub héberge une quantité considérable de données, et il était essentiel de mettre en place des mécanismes pour gérer ces volumes importants de manière efficiente. Cela inclut l'utilisation de techniques de pagination, de parallélisation et d'optimisation des requêtes pour garantir une collecte rapide et efficace des données.
3. Sélection et extraction des informations pertinentes : Les dépôts de code sur GitHub contiennent de nombreuses informations, et il était important de sélectionner et extraire les données pertinentes pour répondre aux objectifs du projet. Cela a nécessité une analyse minutieuse des structures de données disponibles et le développement de techniques d'extraction spécifiques pour obtenir les informations souhaitées.
4. Garantie de la qualité et de la précision des données collectées : La collecte de données à partir de sources externes peut présenter des défis en termes de qualité et d'exactitude. Il a été nécessaire de mettre en place des processus de validation et de vérification des données pour s'assurer de leur fiabilité. Cela comprend la comparaison avec des sources de données connues et l'utilisation de méthodes de nettoyage et de normalisation des données.

5. Collecte de plus de données dans un délai limité : Le projet visait à collecter un large éventail de données pertinentes sur une période de temps définie. Cependant, le temps imparti était limité, ce qui a nécessité une optimisation du processus de collecte pour obtenir le maximum de données dans les délais impartis.
6. Un autre défi important rencontré lors de la réalisation de ce projet était le manque de familiarité avec les bibliothèques Python appropriées pour la collecte de données depuis GitHub.

Résultats

	id	full_name	url	language	license	topics	owner_type	description	stars_count	forks	issues_count	year	created_at	updated_at
0	631962458	bigcode-project/starcoder	https://github.com/bigcode-project/starcoder	Python	Apache License 2.0	[]	Organization	Home of StarCoder: fine-tuning & inference!	5427	344	39	2023	2023-04-24T12:32:21Z	2023-06-23T06:40:59Z
1	631993417	novicezk/midjourney-proxy	https://github.com/novicezk/midjourney-proxy	Java	Apache License 2.0	['midjourney', 'midjourney-api']	User	代理MidJourney的discord频道，实现api形式调用AI绘图	1601	666	79	2023	2023-04-24T13:43:45Z	2023-06-23T07:46:54Z
2	631957497	GFW-knocker/gfw_resist_tls_proxy	https://github.com/GFW-knocker/gfw_resist_tls_proxy	Python	GNU General Public License v3.0	[]	User	knock up GFW sni detection in tls client hello	1453	243	56	2023	2023-04-24T12:20:11Z	2023-06-22T21:00:54Z
3	631998225	erictik/midjourney-client	https://github.com/erictik/midjourney-client	TypeScript	Apache License	['midjourney', 'midjourney-	Organization	MidJourney client.	716	124	44	2023	2023-04-	2023-06-

```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30096 entries, 0 to 30095
Data columns (total 14 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   id            30096 non-null  int64
 1   full_name     30096 non-null  object
 2   url           30096 non-null  object
 3   language      23790 non-null  object
 4   license       13030 non-null  object
 5   topics        30096 non-null  object
 6   owner_type    30096 non-null  object
 7   description   20249 non-null  object
 8   stars_count   30096 non-null  int64
 9   forks         30096 non-null  int64
 10  issues_count  30096 non-null  int64
 11  year          30096 non-null  int64
 12  created_at    30096 non-null  object
 13  updated_at    30096 non-null  object
dtypes: int64(5), object(9)
memory usage: 3.2+ MB
None
```

```
print(data.shape[0])
```

```
30096
```

Nettoyage des données issues du scraping sur Github

Introduction

Le processus de nettoyage des données est une étape essentielle dans l'analyse de données. Il permet de préparer les données brutes en les rendant cohérentes, complètes et prêtes à être utilisées pour des analyses ultérieures. Dans cet ensemble de données spécifique, nous allons effectuer plusieurs étapes de nettoyage des données afin d'améliorer la qualité et la fiabilité de notre jeu de données.

Chargement des données et identification des problèmes

Il s'agit de comprendre la structure des données et d'identifier les problèmes potentiels tels que les erreurs de format, les valeurs manquantes, les doublons et les valeurs aberrantes.

Gestion des valeurs manquantes :

Lorsque certaines données sont manquantes, il est important de décider comment les traiter. Cela peut inclure le remplissage des valeurs manquantes avec des estimations appropriées ou la suppression des lignes ou des colonnes contenant des valeurs manquantes.

Dans notre jeu de données, nous avons identifié les colonnes suivantes contenant des valeurs manquantes :

- language : 6306 valeurs manquantes
- license : 17066 valeurs manquantes
- description : 9847 valeurs manquantes

```
valeurs_manquantes= data.isnull().sum()
display(valeurs_manquantes)
```

```
id                  0
full_name           0
url                 0
language         6306
license         17066
topics              0
owner_type          0
description      9847
stars_count         0
forks               0
issues_count        0
year                0
created_at          0
updated_at          0
dtype: int64
```

Pour la colonne "language" : Étant donné qu'il y a 6306 valeurs manquantes, nous avons décidé de supprimer les lignes correspondantes. Cela signifie que les enregistrements qui ne spécifient pas la langue ont été exclus de notre jeu de

données final. Cette approche a été choisie car la langue est une caractéristique importante et la présence de cette information est cruciale pour notre analyse ultérieure.

Pour les autres colonnes telles que "license" et "description", nous avons utilisé l'étape de standardisation des données pour traiter les valeurs manquantes.

Correction des erreurs de format :

Les erreurs de format dans les données peuvent entraîner des incohérences et des problèmes lors de l'analyse. Dans notre jeu de données, nous avons identifié des problèmes de format dans les colonnes "created_at" et "updated_at". Ces colonnes contiennent des valeurs de date qui sont mal codées ou dans un format incorrect.

Afin de résoudre ce problème, nous utilisons la fonction pd.to_datetime() pour convertir ces colonnes en format de date approprié.

```
data['created_at'] = pd.to_datetime(data['created_at'])
data['updated_at'] = pd.to_datetime(data['updated_at'])
print(data.dtypes)
```

```
id                                int64
full_name                        object
url                              object
language                         object
license                          object
topics                           object
owner_type                       object
description                      object
stars_count                       int64
forks                             int64
issues_count                      int64
year                              int64
created_at      datetime64[ns, UTC]
updated_at      datetime64[ns, UTC]
dtype: object
```

La gestion des valeurs redondantes :

Les doublons dans les données peuvent causer des problèmes dans l'analyse. Il est donc important de détecter et de supprimer les lignes en double pour avoir des données précises.

Dans notre jeu de données, nous avons trouvé 60 lignes en double. Pour résoudre ce problème, nous avons supprimé ces lignes en utilisant une méthode appelée "drop_duplicates()".

En supprimant les doublons, nous nous assurons que chaque enregistrement dans notre jeu de données est unique. Cela garantit que nos données sont fiables et prêtes pour l'analyse.

```
valeurs_redondants = data.duplicated().sum()
display(valeurs_redondants)
```

```
60
```

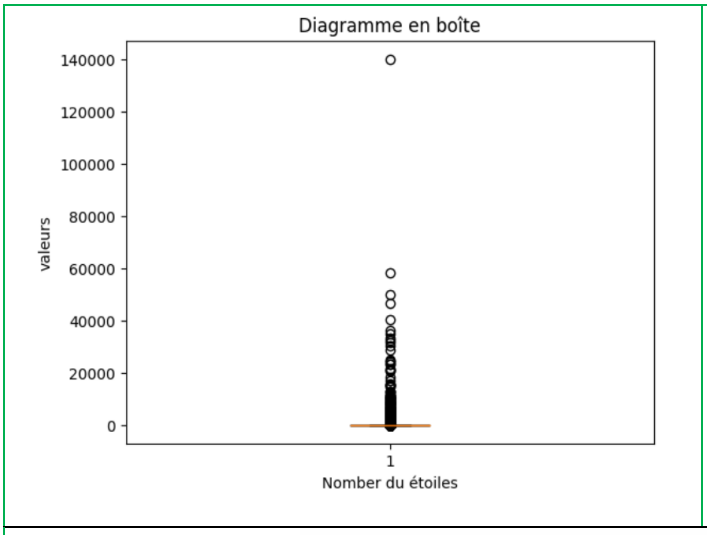
La gestion des valeurs aberrantes :

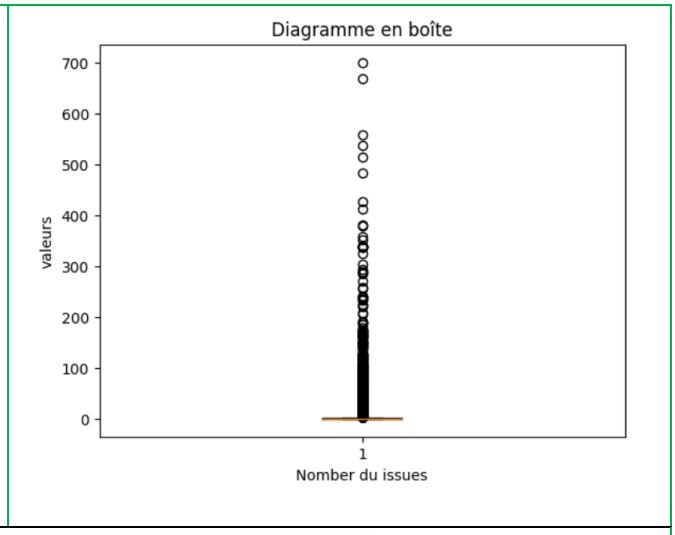
Nous calculons les statistiques descriptives (moyenne, écart-type, médiane, quartiles, etc.) des colonnes 'stars_count', 'forks' et 'issues_count' avec data[['stars_count','forks','issues_count']].describe().

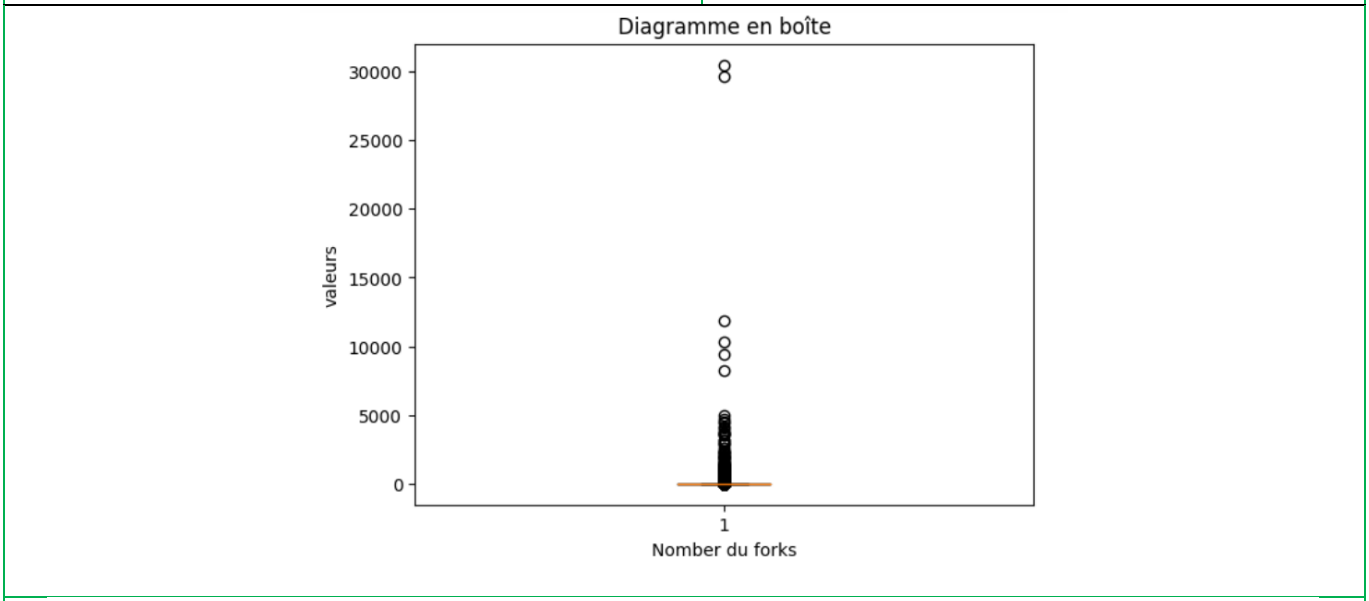
Ensuite, nous identifions les valeurs aberrantes dans la colonne 'stars_count' supérieures à 10000 et les affichons avec display(data.loc[data['stars_count'] > 10000]).

Nous utilisons la méthode des intervalles interquartiles (IQR) pour détecter les valeurs aberrantes dans les colonnes 'stars_count', 'forks' et 'issues_count'. Nous

affichons les intervalles aberrants et le nombre de valeurs aberrantes pour chaque colonne.



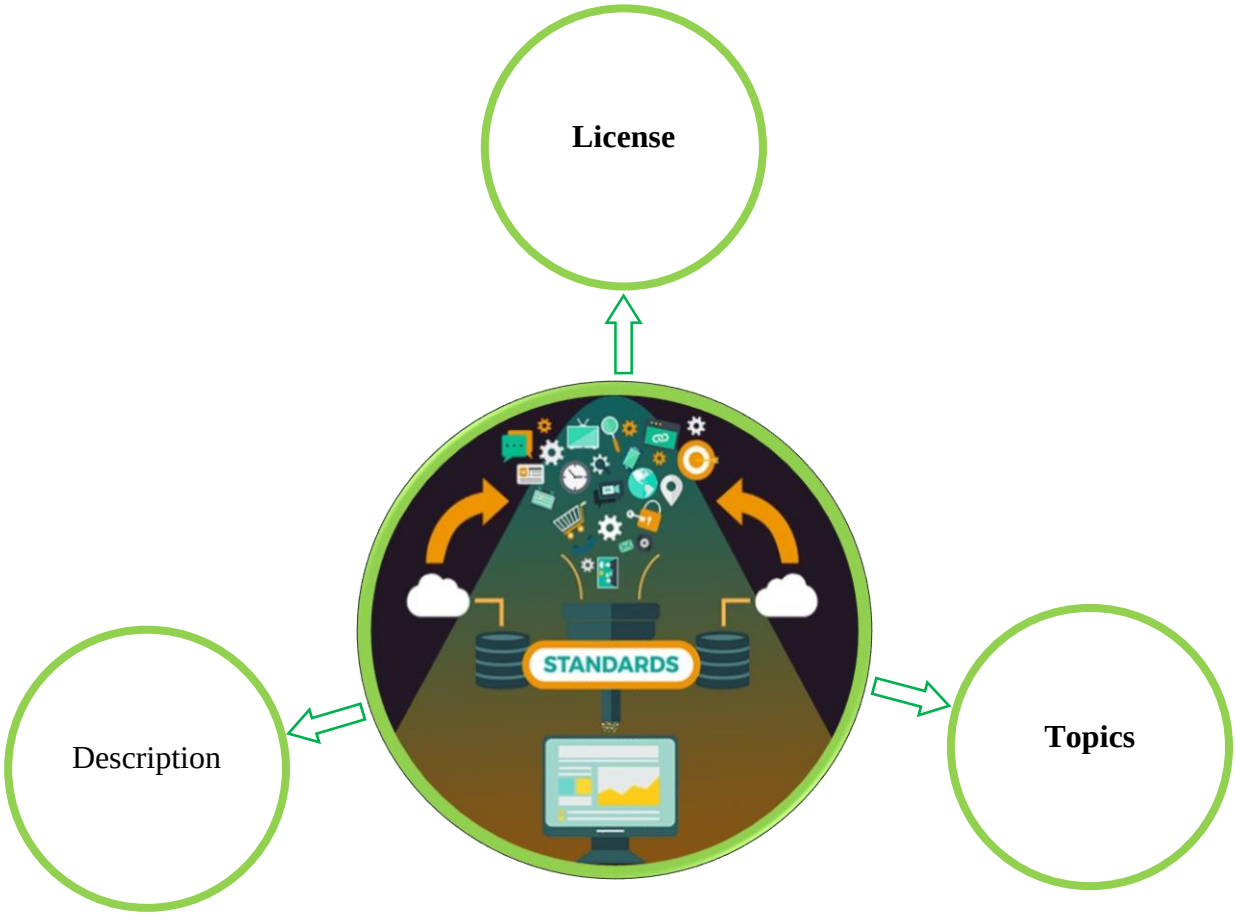




url	language	license	topics	owner_type	description	stars_count	forks	issues_count	year	created_at	updated_at
https://github.com/AntonOsika/gpt-engineer	Python	MIT License	[]	User	Specify what you want it to build, the AI asks for clarification, and then builds it.	30342	4823	62	2023	2023-04-29 12:52:15+00:00	2023-06-23 08:16:26+00:00
https://github.com/mlc-ai/mlc-llm	Python	Apache License 2.0	['language-model', 'llm', 'machine-learning-compilation', 'tvm']	Organization	Enable everyone to develop, optimize and deploy AI models natively on everyone's devices.	10070	669	62	2023	2023-04-29 01:59:25+00:00	2023-06-23 05:25:52+00:00
ps://github.com/imartinez/privateGPT	Python	Apache License 2.0	[]	User	Interact privately with your documents using the power of GPT, 100% privately, no data leaks	31236	3706	361	2023	2023-05-02 09:15:31+00:00	2023-06-23 08:14:23+00:00
https://github.com/s0md3v/roop	Python	GNU Affero General Public License v3.0	['ai', 'deepfake', 'deepfakes', 'face-swap']	User	one-click deepfake (face swap)	13197	2327	11	2023	2023-05-28 14:37:54+00:00	2023-06-23 08:11:40+00:00

Il est important de noter que dans ce cas, nous ne supprimons pas les valeurs aberrantes, car elles peuvent être pertinentes pour notre analyse ou refléter des caractéristiques intéressantes des données.

Standardisation des données



❖ Description :

- **Détection des langues :**

Dans le but de comprendre les langues présentes dans les descriptions, nous avons utilisé la bibliothèque langdetect. Nous avons utilisé la méthode "detect" de langdetect pour détecter la langue d'une description donnée. Cette fonction renvoie le code de langue correspondant à la langue détectée.

- **Traduction des descriptions :**

Nous avons utilisé la bibliothèque googletrans pour traduire les descriptions en anglais. Nous avons créé une fonction appelée "translate_to_english" qui vérifie si la description est déjà en anglais ou si elle est identifiée comme "NO DESCRIPTION". Si c'est le cas, la fonction renvoie la description d'origine. Sinon, la fonction utilise la classe Translator de googletrans pour traduire la description en anglais à l'aide de la méthode "translate". En cas d'erreur de traduction, la description d'origine est renvoyée.

- **La suppression de les emojis et les caractères spéciaux :**

La suppression des emojis et des caractères spéciaux est une étape supplémentaire dans le processus de nettoyage des données. Nous avons créé une fonction appelée "remove_emojis" qui permet de supprimer les emojis et les caractères spéciaux d'une chaîne de texte. Elle utilise des expressions régulières pour détecter et supprimer ces éléments. Si la chaîne de texte passée en paramètre est une chaîne de caractères, la fonction applique le modèle "pattern" pour supprimer les emojis et les caractères spéciaux. Sinon, elle renvoie une chaîne vide.

- **Gestion des valeurs manquantes dans la colonne "description" :**

  La méthode "fillna" de Pandas est utilisée pour remplacer les valeurs manquantes par la valeur spécifiée. Dans le cas de description, toutes les valeurs manquantes sont remplacées par "NO DESCRIPTION ".

❖ License :

Dans le cadre de la standardisation des données, la méthode "fillna" est utilisée pour remplacer les valeurs manquantes dans la colonne "license" du jeu de

données. Lorsqu'il y a des valeurs manquantes dans cette colonne, elles sont remplacées par la chaîne de caractères "NOT INCLUDED".

Cette étape est importante pour garantir que toutes les entrées de la colonne "license" ont une valeur attribuée, même si les données d'origine étaient incomplètes. En remplissant les valeurs manquantes avec une valeur standard telle que "NOT INCLUDED", on assure la cohérence des données et on évite les problèmes lors des analyses ultérieures.

En standardisant les données de cette manière, on améliore la qualité du jeu de données en rendant toutes les entrées dans la colonne "license" consistantes et prêtes à être utilisées dans des analyses ou des traitements ultérieurs.

❖ Topics :

Dans cette étape, nous effectuons une standardisation des données en remplaçant les valeurs vides, représentées par des crochets vides [], dans la colonne 'topics' du jeu de données. Nous utilisons la méthode "apply" de Pandas pour appliquer une fonction lambda à chaque valeur de la colonne.

La fonction lambda vérifie si la valeur de la colonne 'topics' est égale à '[]'. Si c'est le cas, elle remplace cette valeur par la chaîne de caractères "NO TOPIC". Sinon, elle conserve la valeur d'origine.

En remplaçant les valeurs vides par "NO TOPIC", nous rendons les données plus compréhensibles et prêtes à être utilisées dans des analyses ou des modèles qui nécessitent des valeurs cohérentes et non manquantes.