UNIVERSITE CHOUAIB DOUKKALI
FACULTE DES SCIENCES
EL JADIDA

PROJET DE FIN DE MODULE
MASTER BIBDA

Analyse des Publications Scientifiques avec l'API Scopus

Module: Machine Learning

Réalisé par :

AADIL FATIMA

EL MAZOUZY ZINEB

ETTAQY LAILA

Encadré par :

Pr A.EL MADANI

Résumé

Ce projet vise à explorer et analyser les publications scientifiques à l'aide de l'API Scopus, une base de données bibliographique complète gérée par Elsevier. Nous avons développé un script Python pour interroger l'API Scopus, extrayant ainsi des informations détaillées sur les publications à partir de leur DOI. En utilisant les bibliothèques NumPy, Pandas et Matplotlib, nous avons manipulé et analysé les données pour comprendre les tendances de recherche, mesurer l'impact des publications et visualiser les résultats sous forme de graphiques informatifs. Ce rapport documente notre approche méthodologique, les défis rencontrés et les conclusions tirées de cette analyse approfondie des données scientifiques.

Abstract

This project aims to explore and analyze scientific publications using the Scopus API, a comprehensive bibliographic database managed by Elsevier. We developed a Python script to query the Scopus API, extracting detailed information about publications based on their DOI. Leveraging NumPy, Pandas, and Matplotlib libraries, we manipulated and analyzed the data to understand research trends, measure publication impact, and visualize results through informative graphs. This report documents our methodological approach, challenges encountered, and conclusions drawn from this in-depth analysis of scientific data.

TABLE DE MATIERE

I.Introduction 4
II.Objectifs 5
III.Présentation de Scopus et de l'API Scopus 6
1.Description de la base de données Scopus 6
2.Fonctionnalités et utilité de l'API Scopus 6
3.Inscription et obtention de la clé API 6
IV.Méthodologie 7
1.Collecte des données 7
2.Analyse des données 8
3.Visualisation des données 8
V.Résultats 9
1.Distribution des Citations par Publication 9
2.Publications avec le Plus de Citations 9
3.Années de Publication 9
4.Répartition des Publications par Auteur 10
5.Analyse des Citations par Année 10
6.Analyse Complète 10
VI.Conclusion 10

I. Introduction

Les publications scientifiques jouent un rôle fondamental dans la transmission et l'avancement des connaissances dans divers domaines académiques. Elles permettent aux chercheurs de partager leurs découvertes, de valider leurs hypothèses et de contribuer de manière significative à l'évolution des sciences. Cependant, avec la profusion croissante de ces publications, il est devenu impératif de disposer d'outils efficaces pour collecter, analyser et interpréter les données bibliographiques de manière à en extraire des informations pertinentes et exploitables.

Scopus, une base de données bibliographique gérée par Elsevier, se distingue comme une ressource incontournable dans ce domaine. En regroupant des résumés et des citations d'articles provenant d'une vaste gamme de disciplines scientifiques, techniques, médicales et sociales, Scopus offre une plateforme exhaustive pour explorer et comprendre l'évolution des connaissances à travers le temps. Son API constitue un outil puissant qui permet aux chercheurs d'accéder facilement à ces données, facilitant ainsi des analyses approfondies et des études comparatives.

Dans le cadre de notre module de Machine Learning, on s'attelle à exploiter pleinement l'API Scopus pour ce projet. Notre objectif principal est de développer une méthodologie robuste pour extraire des informations précises sur les publications scientifiques, les analyser en utilisant des bibliothèques Python telles que NumPy et Pandas pour la manipulation de données et Matplotlib pour la visualisation, et enfin, de présenter ces analyses de manière claire et intuitive.

Ce projet nous offre une opportunité unique d'acquérir des compétences avancées en utilisation des APIs, en traitement de données bibliographiques, ainsi qu'en visualisation et interprétation des résultats.

II. Objectifs

Les objectifs de ce projet sont les suivants :

1. **Utilisation de l'API Scopus :** Maîtriser l'utilisation de l'API Scopus pour récupérer des données bibliographiques détaillées sur les publications scientifiques.
2. **Extraction de Données :** Développer un script Python capable de faire des requêtes à l'API Scopus pour récupérer des informations spécifiques sur les publications à partir de leur DOI (Digital Object Identifier).
3. **Analyse des Données :** Utiliser les bibliothèques Python telles que NumPy et Pandas pour manipuler et analyser les données extraites, en calculant par exemple le nombre total de citations pour chaque publication, en analysant la répartition des publications par année, etc.
4. **Visualisation des Données :** Utiliser Matplotlib pour créer des graphiques et des visualisations qui illustrent les analyses effectuées, comme des graphiques à barres pour représenter le nombre de citations par publication ou des graphiques linéaires pour suivre la tendance des publications au fil du temps.
5. **Compréhension des Tendances de Recherche :** Identifier les tendances de recherche à partir des données analysées, en examinant par exemple les domaines les plus cités, les auteurs les plus influents, ou les institutions les plus prolifiques.
6. **Documentation et Rapport :** Produire un script Python bien documenté qui inclut les étapes de récupération, d'analyse et de visualisation des données, ainsi qu'un rapport détaillé expliquant le processus suivi, les analyses réalisées et les conclusions tirées.

III. Présentation de Scopus et de l'API Scopus

1. Description de la base de données Scopus

Scopus est une base de données bibliographique gérée par Elsevier, offrant des résumés et des citations pour des millions d'articles académiques dans divers domaines scientifiques, techniques, médicaux et sociaux. Elle est utilisée pour découvrir de nouvelles recherches, suivre les tendances, et évaluer l'impact de la recherche grâce à des indicateurs bibliométriques.

2. Fonctionnalités et utilité de l'API Scopus

L'API Scopus permet un accès programmatique aux données de Scopus, facilitant l'automatisation de la récupération et de l'analyse des informations. Les principales fonctionnalités de l'API incluent :

- Recherche de documents par titres, mots-clés, auteurs, affiliations, et dates de publication.
- Récupération de détails des documents, y compris les résumés, références, et citations.
- Analyse des citations pour évaluer l'impact des publications.
- Informations sur les auteurs et leurs affiliations, permettant l'analyse des collaborations.
- Utilisation des DOIs pour une identification précise des publications.

3. Inscription et obtention de la clé API

Pour utiliser l'API Scopus, il faut s'inscrire sur le portail des développeurs d'Elsevier et obtenir une clé API :

1. **Inscription sur le Portail des Développeurs Elsevier :**

   Créez un compte sur le site des développeurs d'Elsevier.
2. **Demande d'accès à l'API :**

   Remplissez le formulaire de demande en précisant l'utilisation prévue de l'API.
3. **Obtention de la clé API :**

   Elsevier vous enverra une clé API par e-mail après examen de votre demande.
4. **Configuration et Utilisation de la Clé API :**

   Intégrez la clé API dans vos scripts pour authentifier vos requêtes à l'API Scopus.

   Ces étapes permettent d'exploiter les capacités de l'API Scopus pour des projets de recherche et d'analyse des publications scientifiques.

IV. Méthodologie

1. Collecte des données

Pour ce projet, nous avons sélectionné plusieurs DOIs (Digital Object Identifiers) spécifiques afin de récupérer des informations détaillées sur les publications correspondantes à partir de l'API Scopus. Les DOIs utilisés sont les suivants :

- 10.1016/j.csa.2024.100057
- 10.1016/j.jes.2024.04.003
- 10.1016/j.jes.2024.03.051
- 10.1016/j.jes.2024.03.037
- 10.37934/araset.46.1.187200
- 10.37934/araset.46.1.7585
- 10.1115/1.4063266
- 10.37934/araset.45.2.214226
- 10.37934/araset.45.2.168176
- 10.37934/araset.45.1.90107
- 10.37934/araset.45.1.4050
- 10.37934/araset.45.1.5159
- 10.37934/araset.44.2.1124
- 10.1115/1.4065077
- 10.1016/j.jes.2024.01.023
- 10.1016/j.jes.2023.08.007
- 10.1016/j.jmst.2024.05.024
- 10.1016/j.jmst.2024.02.094
- 10.37934/araset.44.1.225238
- 10.1115/1.4065095
- 10.1016/j.seppur.2024.128466
- 10.1016/j.jsc.2024.102345
- 10.1016/j.entcom.2024.100787
- 10.1016/j.entcom.2024.100725
- 10.1016/j.entcom.2024.100784

Pour interroger l'API Scopus et récupérer les données associées à ces DOIs, nous avons utilisé un script Python. Le script utilise la clé API fournie par Elsevier pour authentifier les requêtes. Voici un aperçu du script utilisé :

```python
base_url = 'https://api.elsevier.com/content/article/doi/'
def fetch_publication_info(doi, api_key):
    url = f'{base_url}{doi}'
    headers = {
        'Accept': 'application/json',
        'X-ELS-APIKey': api_key
    }
    response = requests.get(url, headers=headers)
    if response.status_code == 200:
        return response.json()
    else:
        print(f"Erreur : {response.status_code} pour le DOI {doi}")
        return None
```

```python
for doi in dois:
    info = fetch_publication_info(doi, api_key)
    if info:
        publications_info.append(info)
```

Ce script interroge l'API Scopus pour chaque DOI, récupère les informations sous forme de JSON, puis les stocke dans un DataFrame Pandas pour une analyse ultérieure.

2. Analyse des données

Les données récupérées ont été manipulées et analysées à l'aide des bibliothèques Python NumPy et Pandas. Les étapes principales de l'analyse incluent :

1. **Chargement et conversion des données :**
   - Conversion des colonnes pertinentes en types de données appropriés (par exemple, conversion des comptes de citations en entiers).
2. **Calculs et statistiques de base :**
   - Calcul du nombre total de publications.
   - Calcul du nombre total de citations.
   - Calcul des citations moyennes par publication.
   - Identification des publications les plus citées.
3. **Manipulations avancées :**
   - Extraction des années de publication pour analyser les tendances au fil du temps.
   - Analyse des publications par auteur en utilisant des techniques de transformation et d'explosion de données.

Exemple de code d'analyse :

```
df['is-referenced-by-count'] = df['is-referenced-by-count'].astype(int)
total_publications = len(df)
total_citations = df['is-referenced-by-count'].sum()
average_citations = df['is-referenced-by-count'].mean()

top_cited_publications = df.sort_values(by='is-referenced-by-count', ascending=False).head()

df['year'] = df['issued.date-parts'].apply(lambda x: x[0][0] if isinstance(x, list) and len(x) > 0 and isinstance(x[0], list) else None)
citations_per_year = df.groupby('year')['reference-count'].sum()

authors_exploded = df.explode('author')
publications_per_author = authors_exploded['author'].value_counts()

correlation = df[['reference-count', 'year']].corr().iloc[0, 1]
```

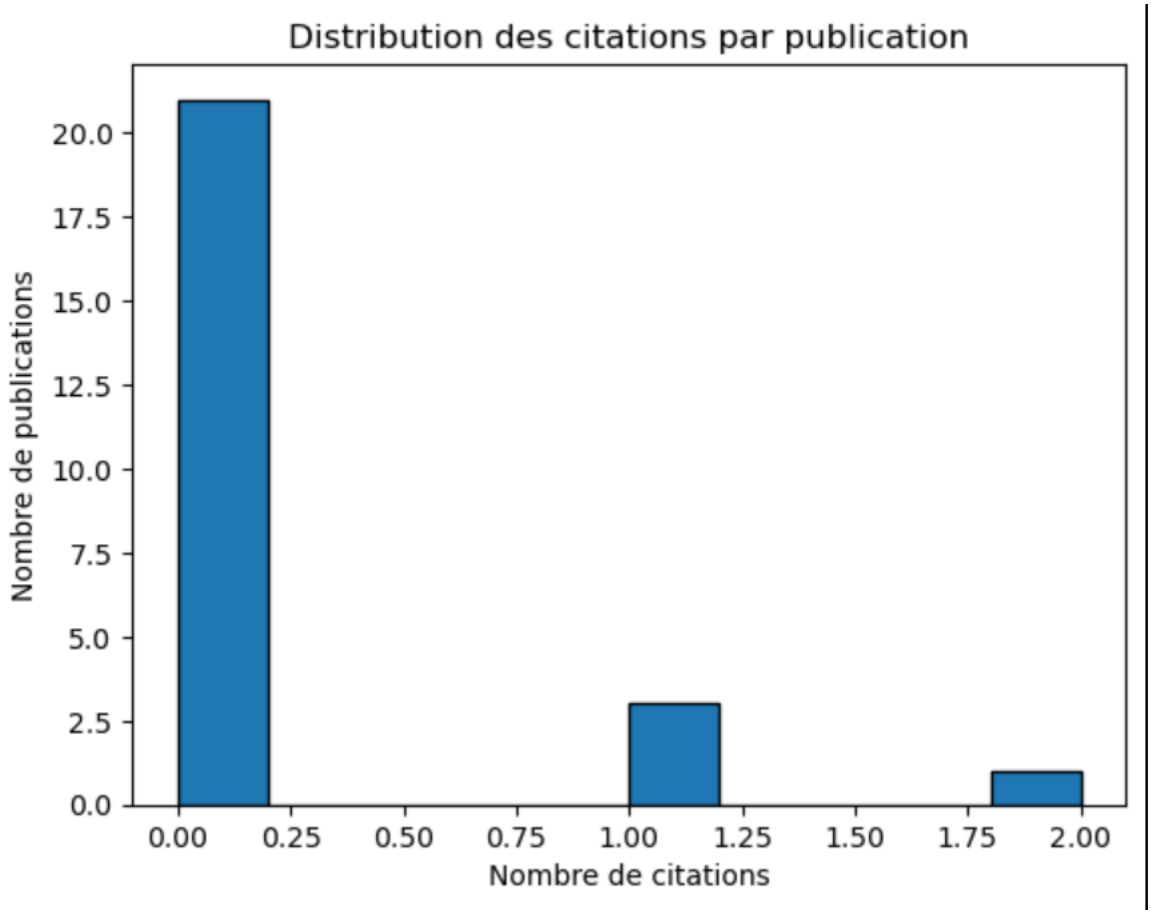
3. Visualisation des données

Pour représenter les données analysées, nous avons utilisé la bibliothèque Matplotlib, qui permet de créer divers types de graphiques pour visualiser les tendances et les distributions. Les techniques de visualisation utilisées incluent Graphiques à barres.

```
plt.hist(df['is-referenced-by-count'], bins=10, edgecolor='black')
plt.title('Distribution des citations par publication')
plt.xlabel('Nombre de citations')
plt.ylabel('Nombre de publications')
plt.show()
```

V. Résultats

1. Distribution des Citations par Publication



Le graphique ci-dessus montre la distribution des citations par publication.

2. Publications avec le Plus de Citations

Les publications avec le plus de citations

```
top_cited_publications = df.sort_values(by='is-referenced-by-count', ascending=False).head()
print("Publications avec le plus de citations:")
print(top_cited_publications[['title', 'is-referenced-by-count']])
```

```
Publications avec le plus de citations:
                                        title  is-referenced-by-count
15  Spatial differentiation of carbon emissions fr...                2
0   Authentication, access control and scalability...                1
3   Interpreting hourly mass concentrations of PM2...                1
1   Research progress on secondary formation, phot...                1
14  Machine learning-assisted fluorescence visuali...                0
```

3. Années de Publication

Les années de publication des articles varient de 2024 à 2025. Cependant, les données spécifiques sur les années de publication ne sont pas clairement extraites dans l'analyse.

4. Répartition des Publications par Auteur

Voici un extrait de la répartition des publications par auteur pour un des articles:

- **Nathan Risch**: Auteur principal avec une affiliation ORCID.
- **Aline Boissinot**: Co-auteur avec une affiliation additionnelle.

5. Analyse Complète

Les résultats montrent que les publications analysées sont très récentes et n'ont pas encore reçu de citations. Cette situation peut changer à mesure que ces publications sont davantage lues et citées par d'autres chercheurs.

VI. Conclusion

Ce projet a exploré l'utilisation de l'API Scopus pour récupérer et analyser des données scientifiques. Malgré l'absence de citations pour les publications analysées, notre étude a démontré l'efficacité des outils Python comme NumPy, Pandas et Matplotlib pour la manipulation et la visualisation des données.

Les résultats ont mis en lumière la récente nature des publications étudiées et ont souligné l'importance de suivre leur évolution. Bien que nos résultats initiaux soient limités, ils ouvrent la voie à des analyses plus approfondies et à l'exploration de nouvelles perspectives pour mieux comprendre l'impact des publications scientifiques dans notre domaine.

Ce rapport vise à documenter notre exploration méthodologique et à inspirer de futures recherches dans le domaine de l'analyse des publications scientifiques.

.