ETL語意相似度分析

郭益華

GitHub

目錄

1. 簡介說明

專案說明

ETL實作

- 使用社群平台資料集作為MetaData建置於PostgreSQL
- 利用Sentence-BERT語言模型將資料集進行語意相似度分析
- 將分析結果匯入至MongoDB
- 將流程整合為一自動化API

資料集:

某社群平台貼文資料集

資料筆數: 32266004

使用套件

- psycopg2==2.9.3
- sentence-transformers==2.2.2
- Flask==2.1.2
- pandas==1.4.3
- numpy==1.23.0
- pymongo==4.1.1

前置準備

- 自行準備任意有文本的資料集
- 建置PostgreSQL資料庫
- 建立資料表(table) & 索引(index)
- 匯入資料集
- 建置MongoDB Atlas資料庫
- 建立MongoDB Atlas資料庫中之Collection

實務知識

- ETL流程
- 大數據處理
- PostgreSQL搜尋效能提升
- Python Flask後端開發
- API開發
- PostgreSQL
- MongoDB

ETL Flow

Extract(E)

PostgreSQL

Social Media Metadata

Transform(T)

Data Processing & Semantic Similarity Analysis

Load(L)

mongoDB

Load in MongoDB Atlas

專案架構

專案架構		
main.py		
package 資料夾		
Extract(E)	Transform(T)	Load(L)
ExtractPostgreSQL.py	sentenceBERT.py TransformData.py	LoadMongoDB.py
test 資料夾		
integration_test.py		
Extract(E)	Transform(T)	Load(L)
test_ExtractPostgreSQL.py	test_sentenceBERT.py	test_LoadMongoDB.py

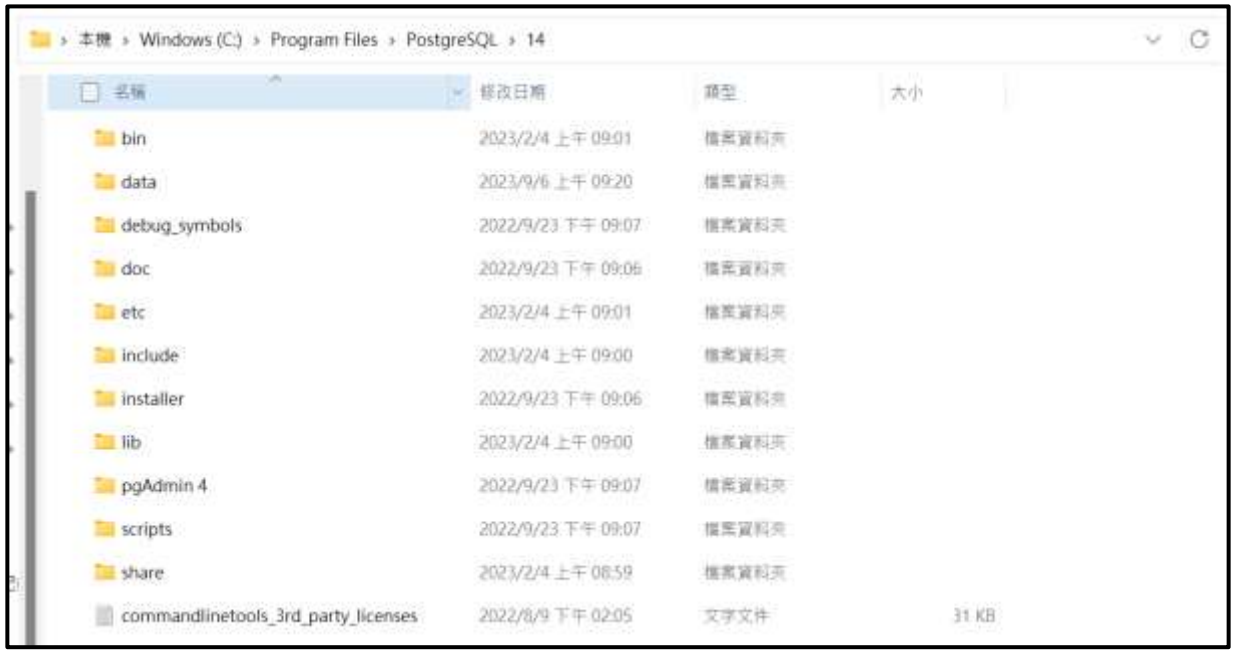
專案目錄

```
C:.
│   main.py
│   requirements.txt
│
├─package
│   │   ExtractPostgreSQL.py
│   │   LoadMongoDB.py
│   │   sentenceBERT.py
│   │   TransformData.py
│   │   __init__.py
│   │
│   └─__pycache__
│           ExtractPostgreSQL.cpython-310.pyc
│           LoadMongoDB.cpython-310.pyc
│           sentenceBERT.cpython-310.pyc
│           TransformData.cpython-310.pyc
│           __init__.cpython-310.pyc
│
└─test
        integration_test.py
        test_ExtractPostgreSQL.py
        test_LoadMongoDB.py
        test_sentenceBERT.py
```

2. PostgreSQL搜尋效能提升

pgroonga下載

- pgroonga 為 PostgreSQL 的文本搜尋外掛套件，支援全語系
- 可至官方網站下載: https://pgroonga.github.io/
- 載完將相關檔案放置與本機PostgreSQL相對應之資料夾



搜尋效能測試

連續跌代搜尋286個關鍵字		
Number of data	32266004	
	Iteration keyword	Time
Pgroonga Text Search	286	3 m 41 s
LIKE Search	286	1 h up
可發現 pgroonga 明顯快上許多		

3. 程式碼撰寫

程式碼 & API說明

main.py		整合package的主程式碼及API
	package	
	ExtractPostgreSQL.py	根據keyword至PostgreSQL搜尋並獲取相關資料
	TransformData.py	將資料進行處理
	sentenceBERT.py	將處理後的資料進行語意相似度分析
	LoadMongoDB.py	將分析後的資料結果匯入至MongoDB

API	
http://127.0.0.1:<port>/api/socialnetwork/v1/similarity? start_date=2021-12-15&end_date=2021-12-20&search萊豬	
參數	**說明**
start_date	起始日期 ex: 2021-12-01
end_date	結束日期 ex: 2021-12-20
search	關鍵字(語言不限) ex: 萊豬

4. 測試碼撰寫

測試碼說明

Test	
integration_test.py	程式碼整合測試(包含TransformData)
test_ExtractPostgreSQL.py	測試PostgreSQL連線及搜尋
test_sentenceBERT.py	測試sentence-BERT運作語意相似度分析
test_LoadMongoDB.py	測試MongoDB連線及CRUD

5. 測試

integration_test.py 畫面

```
PS C:\Users\jerry\Desktop\master course\dataEngineer\similarityETL\similarityETLbeauty\test> python .\integration_test.p
y
PostgreSQL connection successful!
first_clean successful!
Batches: 100%|                                                                                | 92/92 [01:44<00:00,  1.13s/it]
semanticSimilarity successful!
second_clean successful!
Pinged your deployment. You successfully connected to MongoDB!
```

ExtractPostgreSQL.py

TransformData.py

sentenceBERT.py

TransformData.py

LoadMongoDB.py

test_ExtractPostgreSQL.py 畫面

ExtractPostgreSQL.py

```
PS C:\Users\jerry\Desktop\master course\dataB
eSQL.py
PostgreSQL connection successful!
```

test_sentenceBERT.py 畫面

sentenceBERT.py

```
PS C:\Users\jerry\Desktop\master course\dat
py
Batches: 100%|
SemanticSimilarity successful!
```

test_LoadMongoDB.py 畫面

Test MongoDB CRUD

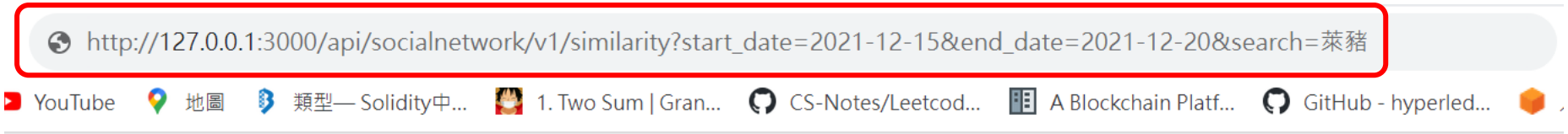
```
PS C:\Users\jerry\Desktop\master course\dataEngineer\similarityETL\similarityETLbe
y
Pinged your deployment. You successfully connected to MongoDB!
Add successful! Create
ObjectID: 64f87272c4de8596a996931f
Data: {'_id': ObjectId('64f87272c4de8596a996931f'), 'test': 'Hello World'}
Check successful! Read
Data: {'_id': ObjectId('64f87272c4de8596a996931f'), 'test': 'Hello World Update'}
Update successful!  Update
Data: <pymongo.results.DeleteResult object at 0x0000019E48C92980>
Delete successful!   Delete
```

6. 實際成果

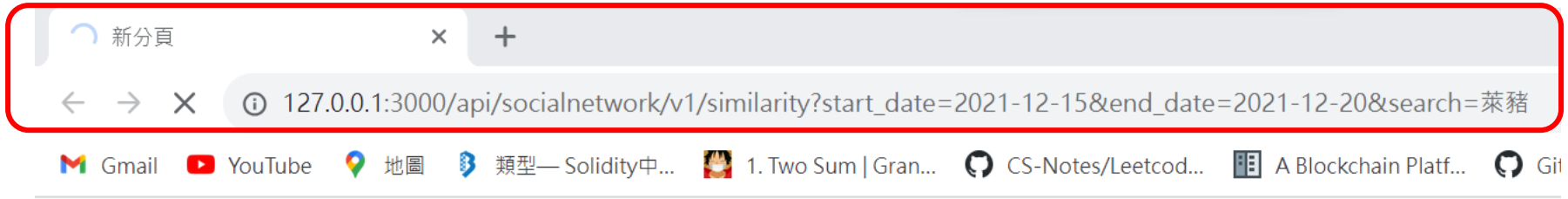
啟動API

```
PS C:\Users\jerry\Desktop\master course\dataEngineer\similarityETL\similarityETLbeauty> python main.py
 * Serving Flask app 'main' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:3000 (Press CTRL+C to quit)
```

輸入網址


http://127.0.0.1:3000/api/socialnetwork/v1/similarity?start_date=2021-12-15&end_date=2021-12-20&search=萊豬
YouTube
地圖
類型— Solidity中...
1. Two Sum | Gran...
CS-Notes/Leetcod...
A Blockchain Platf...
GitHub - hyperled...


Google


新分頁
127.0.0.1:3000/api/socialnetwork/v1/similarity?start_date=2021-12-15&end_date=2021-12-20&search=萊豬
Gmail
YouTube
地圖
類型— Solidity中...
1. Two Sum | Gran...
CS-Notes/Leetcod...
A Blockchain Platf...
Gi

ETL執行過程

```
PS C:\Users\jerry\Desktop\master course\dataEngineer\similarityETL\similarityETLbeauty> python main.py
 * Serving Flask app 'main' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:3000 (Press CTRL+C to quit)
ExtractPostgreSQL successful!
TransformData first_clean successful!
Batches: 100%|                                                                                | 92/92 [01:45<00:00,  1.15s/it]
semanticSimilarity successful!
TransformData second_clean successful!
LoadMongoDB successful!
127.0.0.1 - - [06/Sep/2023 21:06:41] "GET /api/socialnetwork/v1/similarity?start_date=2021-12-15&end_date=2021-12-20&sea
rch=萊豬 HTTP/1.1" 200 -
```

API啟動

ETL執行過程

資料成功匯入 MongoDB Atlas 畫面

End