

CS-429

INFORMATION RETRIEVAL PROJECT REPORT

Submitted by

Sudireddy Raghavender Reddy

A20554654

Abstract

The goal of this project is to use Python 3.10+ and the associated libraries to create a web-based document retrieval system. The system includes query processing using Flask, indexing with Scikit-Learn, and web crawling with Scrapy. Giving users accurate and pertinent search results based on free-text searches is the goal.

Overview

In this project, we aim to design a search engine capable of retrieving and filtering web content based on TF-IDF scores. The development process involves several key steps to ensure effective and accurate search functionality. A web crawler, an indexing engine, and a query processor make up the three primary parts of the suggested system and solution outline. To guarantee reliable and accurate search capabilities, the development process consists of multiple crucial phases. Top search results are generated using two different APIs, one for regular indexing and the other for advanced indexing. The Scikit-Learn documentation, current semantic search research are the sources of information for this project.

Design

Web Scrapy

The initial part emphasizes developing a thorough web scraper using Scrapy. This crawler is designed to explore websites and gather relevant web pages for further analysis.

TF-IDF Cosine Similarity with Scikit-Learn

The first method involves constructing an index using TF-IDF values. We use the Scikit-Learn library to compute TF-IDF vectors and cosine similarity measures between the indexed documents and user queries. We've developed a specific API to return the top 5 search results based on this approach, which are then presented in JSON format.

Query Processing:

The Flask processor evaluates and verifies user queries using cosine similarity and TF-IDF scores before returning the best results. Spell check and query expansion are also available.

Architecture

Software Components:

Web crawler, Indexing Engine, Query Processor, APIs

Interfaces: Restful APIs

Operation

Install Python and Install Linux in windows

wsl –install

Install required libraries

Pip install scrapy

Pip install scikit-learn

pip install beautifulsoup4

pip install flask

pip install requests

Instructions to run the project

Step 1: Navigate to spiders folders in the terminal – cd myproject - cd spiders
and enter Scrapy crawl <file name>

The index.pkl file is created and the TF-IDF scores, cosine similarities for the
html documents are computed and stored in the index.pkl file.

Step 2: The content of index.pkl file will be displayed in the terminal when you
access it through pickle folder

Step 3: Start the Flask server by going to the flask folder in the terminal and run
the Python file

Step 4: After the flask server is started, Open new terminal and make a request
as below:

```
curl -X POST http://localhost:5000/query -H "Content-Type:  
application/json" -d '{"query": "DSA"}'
```

After this, the json format response with cosine similarity and document name
from the server with top k results are generated.

Conclusion:

The project demonstrates how both APIs return relevant documents as output. However, rating the documents involves effort. The Scikit-learn API also delivers documents that are relevant to all searches. However, the precision is satisfactory. Certain extra inquiries get consistent results across all three scenarios.

Output:

Scraping:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

raghu@DESKTOP-NOEDAT7:/mnt/c/Users/DELL/Desktop/myproject/myproject/spiders$ scrapy crawl my_crawler
2024-04-22 21:55:06 [scrapy.utils.log] INFO: Scrapy 2.5.1 started (bot: myproject)
2024-04-22 21:55:06 [scrapy.utils.log] INFO: Versions: lxml 4.8.0.0, libxml2 2.9.13, cssselect 1.1.0, parsel 1.6.0, w3lib 1.22.0, Twisted 22.1.0, Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0], pyOpenSSL 21.0.0 (OpenSSL 3.0.2 15 Mar 2022), cryptography 3.4.8, Platform Linux-5.15.146.1-microsoft-standard-WSL2-x86_64-with-glibc2.35
2024-04-22 21:55:06 [scrapy.utils.log] DEBUG: Using reactor: twisted.internet.asyncioreactor.AsyncioSelectorReactor
2024-04-22 21:55:06 [scrapy.utils.log] DEBUG: Using asyncio event loop: asyncio.unix_events.UnixSelectorEventLoop
2024-04-22 21:55:06 [scrapy.crawler] INFO: Overridden settings:
{'BOT_NAME': 'myproject',
 'CLOSESPIDER_PAGECOUNT': 20,
 'DEPTH_LIMIT': 3,
 'FEED_EXPORT_ENCODING': 'utf-8',
 'NEWSPIDER_MODULE': 'myproject.spiders',
 'ROBOTSTXT_OBEY': True,
 'SPIDER_MODULES': ['myproject.spiders'],
 'TWISTED_REACTOR': 'twisted.internet.asyncioreactor.AsyncioSelectorReactor'}
2024-04-22 21:55:06 [scrapy.extensions.telnet] INFO: Telnet Password: 628df6de62dcaa40
2024-04-22 21:55:06 [scrapy.middleware] INFO: Enabled extensions:
['scrapy.extensions.corestats.CoreStats',
 'scrapy.extensions.telnet.TelnetConsole',
 'scrapy.extensions.memusage.MemoryUsage',
 'scrapy.extensions.closespider.CloseSpider',
 'scrapy.extensions.logstats.LogStats']
```

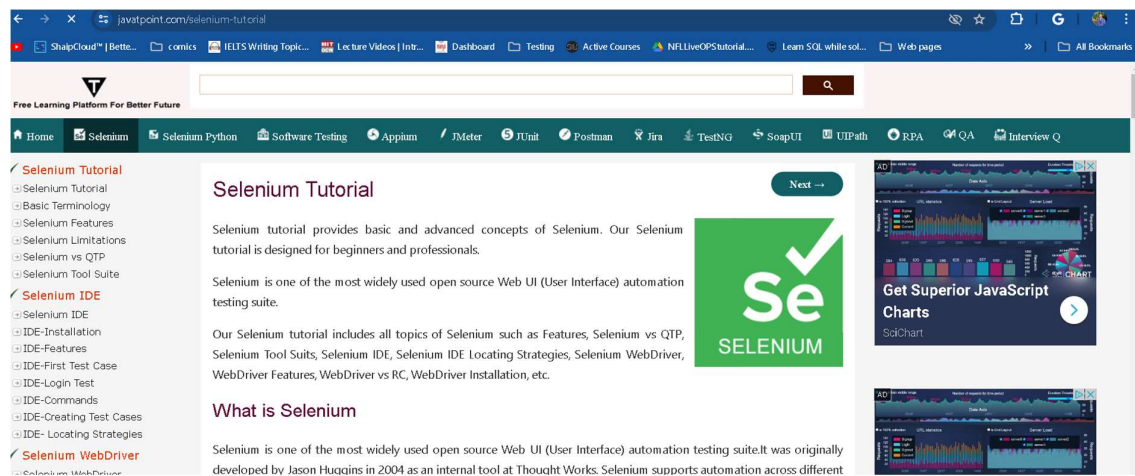
Server started

```
DEBUG CONSOLE TERMINAL PORTS COMMENTS

raghu@DESKTOP-NOEDAT7:/mnt/c/Users/DELL/Desktop/myproject/myproject/flask$ python3 query_processor.py
* Serving Flask app "query_processor"
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 104-742-208
```


Test Cases -

```
raghu@DESKTOP-MOEDAT7: /mnt/c/Users/DELL/Desktop/myproject$ curl -X POST http://localhost:5000/query -H "Content-Type: application/json" -d '{"query": "selenium"}'
[
  {
    "cosine_similarity": 0.41304793867290474,
    "document_name": "selenium-tutorial.html",
    "snippet": " selenium tutorial javatpoint home selenium selenium python software testing appium jmeter j
unit postman jira testing soapui uipath rpa qa interview q "
  },
  {
    "cosine_similarity": 0.02108368035209246,
    "document_name": "javascript-tutorial.html",
    "snippet": " mobile application testing tutorial javatpoint home mobile testing software testing compile
r java coa computer network html css selenium servlet jsp "
  },
  {
    "cosine_similarity": 0.011603297415025905,
    "document_name": "interview-questions-and-answers.html",
    "snippet": " interview questions and answers 2023 javatpoint scroll to top home interview questions java
sql python javascript angular selenium spring boot hr c d"
  },
  {
    "cosine_similarity": 0.00802512405291943,
    "document_name": "splunk.html",
    "snippet": " splunk tutorial javatpoint home java splunk python c c javascript bootstrap html xhtml css
jquery xml json comment forum splunk tutorial splunk tutor"
  },
  {
    "cosine_similarity": 0.007356153519572587,
    "document_name": "cpp-tutorial.html",
    "snippet": " learn c tutorial javatpoint home net c adonet aspnet sql server f angularjs nodejs expressj
s sql html javascript ajax interview q net framework net f"
  }
]
```



Free Learning Platform For Better Future

Home Selenium Selenium Python Software Testing Appium JMeter JUnit Postman Jira TestNG SoapUI UIPath RPA QA Interview Q

Selenium Tutorial

Selenium tutorial provides basic and advanced concepts of Selenium. Our Selenium tutorial is designed for beginners and professionals.

Selenium is one of the most widely used open source Web UI (User Interface) automation testing suite.

Our Selenium tutorial includes all topics of Selenium such as Features, Selenium vs QTP, Selenium Tool Suite, Selenium IDE, Selenium IDE Locating Strategies, Selenium WebDriver, WebDriver Features, WebDriver vs RC, WebDriver Installation, etc.

What is Selenium

Selenium is one of the most widely used open source Web UI (User Interface) automation testing suite. It was originally developed by Jason Huggins in 2004 as an internal tool at Thought Works. Selenium supports automation across different

Next →

Get Superior JavaScript Charts
SolChart

```

raghu@DESKTOP-NOEDAT7:/mnt/c/Users/DELL/Desktop/myproject$ curl -X POST http://localhost:5000/query -H "Content-Type: application/json" -d '{"query": "javascript"}'
[
  {
    "cosine_similarity": 0.5914165419872691,
    "document_name": "html-tutorial.html",
    "snippet": " learn javascript tutorial javatpoint home javascript html css bootstrap jquery nodejs php python c c java c sql android interview q javascript tutori"
  },
  {
    "cosine_similarity": 0.03162552052813869,
    "document_name": "javascript-tutorial.html",
    "snippet": " mobile application testing tutorial javatpoint home mobile testing software testing compile r java coa computer network html css selenium servlet jsp "
  },
  {
    "cosine_similarity": 0.02599530108273735,
    "document_name": "c-sharp-tutorial.html",
    "snippet": " learn php tutorial javatpoint home php mysql laravel wordpress magento 2 codeigniter yii ht ml css javascript jquery python java sql interview q php t"
  },
  {
    "cosine_similarity": 0.02407537215875829,
    "document_name": "splunk.html",
    "snippet": " splunk tutorial javatpoint home java splunk python c c javascript bootstrap html xhtml css jquery xml json comment forum splunk tutorial splunk tutor"
  },
  {
    "cosine_similarity": 0.023994032967314487,
    "document_name": "sql-tutorial.html",
    "snippet": " learn html tutorial javatpoint home html xhtml css javascript bootstrap jquery php xml json python c java c sql interview q html tutorial introductio"
  }
]

```



Source Code

Stackoverflow- <https://stackoverflow.com/>

Scrapy documentation site - <https://docs.scrapy.org/en/latest/>

Flask Documentation site - <https://flask.palletsprojects.com/en/3.0.x/tutorial/>

Data Sources

Scrapy – Version 2.8.0

Beautiful Soup – Version 4.12.2

Scikit-learn – version 1.4.2

Flask – version 2.2.5

Bibiliography

<https://flask.palletsprojects.com/en/3.0.x/>

<https://scikit-learn.org/stable/>

<https://requests.readthedocs.io/en/latest/>

<https://scrapy.org/>

Resources:

Used Microsoft Bing to some extent and Stack Overflow for error correction.