Shri Dharmasthala Manjunatheshwara College of Engineering and Technology

---

Department of Information Science and Engineering

Minor Project
V Semester 2021-22

Web Scraping: A Web Application

Guide • Prof. Leena Sakri

Aditya Bammanagoudar • 2SD19IS003
Guruprasad Bhagwat • 2SD19IS021
Kanishkvardhan A N • 2SD19IS023
Karthik Kavathekar • 2SD19IS024

Problem

1.State what problems you aim to solve?

When the user requests to a website the website throws huge amount of data and for user it is very hard to read to avoid that we can use the Web scraping tools to retrieve the data in structured manner.

What is Web Scraping?

- Web scraping is a process where a web page where fetching and extracting occurs. Fetching is the downloading of a page (which a browser does when a user views a page). The web scraping software may directly access the World Wide Web using the Hypertext Transfer Protocol or a web browser.
- Therefore, web crawling is a main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place. The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet or loaded into a database.
- Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else. An example would be to find and copy names and telephone numbers, or companies.

Solution

1.Solution benefitting the customer

By using various modules and programming them using Python, we plan to display the abridged version of the data extracted from the internet so that the user can digest them instantaneously. The summarized to-the-point information helps the user to comprehend on the spot without wasting any time.

Python packages used

1.Wikipedia

- Wikipedia is a Python library that makes it easy to access and parse data from Wikipedia.
- Goal of Wikipedia-API is to provide simple and easy to use API for retrieving information from Wikipedia.
- Wikipedia-API is easy to use Python wrapper for Wikipedia's API. It supports extracting texts, sections, links, categories, translations, etc from Wikipedia. Documentation provides code snippets for the most common use cases

2. BeautifulSoup4

- Beautiful Soup is a Python library that is used to extract information from XML and HTML files.
- One of Beautiful Soup's strengths is its ability to detect page encoding, and hence get more accurate information from the HTML text. Another advantage of Beautiful Soup is its simplicity and ease.
- If you're just starting with webs scraping or with Python, Beautiful Soup is the best choice to go. Moreover, if the documents you'll be scraping are not structured, Beautiful Soup will be the perfect choice to use.

3.Requests

- Requests library is one of the integral part of Python for making HTTP requests to a specified URL.
- Whether it be REST APIs or Web Scraping, requests is must to be learned for proceeding further with these technologies. When one makes a request to a URI, it returns a response.
- Python requests provides inbuilt functionalities for managing both the request and response.

4.Streamlit

- Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science.
- Streamlit allows you to write an app the same way you write a python code.
- Streamlit makes it seamless to work on the interactive loop of coding and viewing results in the web app.

Implementation

BeautifulSoup4 for source code and data extraction purposes

-

Streamlit for creating the Web-Application

Customer Segments

1. Students and Teachers - For faster understanding of concepts
2. Authors and Writers - For quick comprehension of their topics
3. Journalists - Attainable ideas for market demands/sentiments.
4. Stock Investors/Beginners - To get valuable ideas and information about the top 15 companies to understand the market

Goals for the project

1. We'll be using these packages to retrieve information on articles based on users' search URL.
2. These articles will be limited to number of search result in terms of data.
3. The extracted and formatted article (derived from the main URL) is given back to the user in the form of text output.
4. The working web application will be hosted on the internet, using the streamlit library.

Summary and Conclusion

1. A web app which displays summarized data to-the-point information helps the user to comprehend on the spot without wasting any time.
2. Programming Language - Python
3. Python Libraries - Beautiful Soup, Requests, yFinance, Streamlit
4. This web app is open source and free for all the users.

References

1. **A Review on Web Scraping and its Applications** - IEEE Xplore

2. **Web Scraping** - GeeksForGeeks
3. **Python Packages** - PyPi
4. **Comprehensive Tutorials** - YouTube

Thank You