WEB SCRAPING

A Synopsis Submitted

Computer Science and Engineering

by

SWETANSHU PANDEY (20262)

(Department of Computer Science and Engineering)

KNIT SULTANPUR

UTTAR PRADESH, INDIA

December, 2021

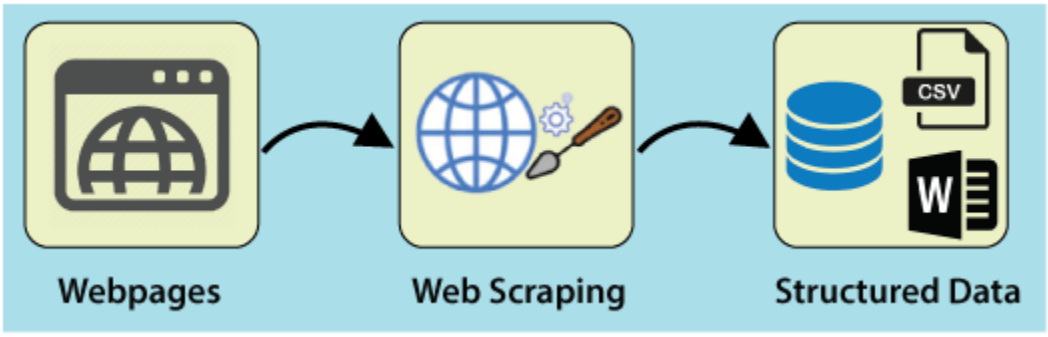
INDEX

ABSTRACT OF THE PROJECT

Web Scraping is a technique to extract a large amount of data from several websites. The term "scraping" refers to obtaining the information from another source (webpages) and saving it into a local file.

For example: Suppose you are working on a project called "Phone comparing website" where you require the price of mobile phones, ratings, and model names to make comparisons between the different mobile phones.

If you collect these details by checking various sites, it will take much time. In that case, web scrapping plays an important role where by writing a few lines of code you can get the desired results.



Web Scrapping extracts the data from websites in the unstructured format.

It helps to collect these unstructured data and convert it in a structured form.

Use of Web Scraping

- Web Scrapping is perfectly appropriate for market trend analysis. It is gaining insights into a particular market.
  The large organization requires a great deal of data, and web scrapping provides the data with a guaranteed level of reliability and accuracy.

- Many companies use personals e-mail data for email marketing.
  They can target the specific audience for their marketing.

- It is widely used to collect data from several online shopping sites and compare the prices of products and make profitable pricing decisions.
  Price monitoring using web scrapped data gives the ability to the companies to know the market condition and facilitate dynamic pricing.
  It ensures the companies they always outrank others.
- Web Scrapping plays an essential role in extracting data from social media websites such as **Twitter, Facebook,** and **Instagram,** to find the trending topics.

UNDERSTANDING WEB SCRAPING

The web scrapping consists of two parts: a web crawler and a web scraper.

The crawler leads the scrapper and extracts the requested data.

- **The crawler**

  A web crawler is generally called a "spider."

  It is an artificial intelligence technology that browses the internet to index and searches for the content by given links.

  It searches for the relevant information asked by the programmer.

- **The scrapper**

  A web scraper is a dedicated tool that is designed to extract the data from several websites quickly and effectively.

  Web scrappers vary widely in design and complexity, depending on the projects.

LIBRARIES FOR WEB SCRAPING

1. Scrapy

Scrapy was initially designed to build web spiders that can crawl the web on their own.

It can be used in monitoring and mining data, as well as automated and systematic testing.

Scrapy fetch the HTML content of a website using the fetch function.

Code

```
fetch("https://www.xyz.com")
```

Then use the `view` function to open up this HTML file in your default

browser.

```
view(response)
print(response.text)
```

2. Requests

Requests allow the user to sent requests to the HTTP server and GET response back in the form of HTML or JSON response.

It also allows the user to send POST requests to the server to modify or add some content.

Code

```
pip install requests
```

3. Urllib

Urllib is a Python library that allows the developer to open and parse information from HTTP or FTP protocols.

Functionality of Urllibs-

urllib.request **:** opens and reads URLs.

urllib.error: catches the exceptions raised by urllib.request.

urllib.parse: parses URLs.

urllib.robotparser: parses robots.txt files.

Code

pip install urllib

Urllib is a little more complicated than Requests*;* however, if you want to have better control over your requests, then Urllib is the way to go.

4. Beautiful Soup

Beautiful Soup is a Python library that is used to extract information from XML and HTML files.

Beautiful Soup is considered a parser library.

Parsers help the programmer obtain data from an HTML file.

If parsers didn't exist, we would probably use Regex to match and get patterns from the text, which is not an efficient or maintainable approach.

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.

Code

pip install beautifulsoup4

5. Selenium

We can use it to open a webpage, click on a button, and get results.

SeleniumLibrary uses the Selenium WebDriver modules internally to control a web browser.

Code

pip install selenium

TESTING URL LINKS

Testing the project on various url links like

- https://en.wikipedia.org/wiki/Python_(programming_language)

- https://en.wikipedia.org/wiki/Web_scraping

- https://en.wikipedia.org/wiki/Albert_Einstein

- https://news.google.com/topstories?hl=en-IN&gl=IN&ceid=IN:en

- https://www.flipkart.com/samsung-galaxy-a03-core-black-32-gb/p/itm7f39ac244845c?pid=MOBG9BZ3BU9BDUBB&lid=LSTMOBG9BZ3BU9BDUBBWFYGAG&marketplace=FLIPKART&fm=neo%2Fmerchandising&iid=M_59f3b2e2-b299-411e-9002-2e5bc25f9e2a_1_1BUWY8OBA8L9_MC.MOBG9BZ3BU9BDUBB&ppt=None&ppn=None&ssid=symbkwk2zk0000001641048344166&otracker=clp_pmu_v2_Latest%2BSamsung%2Bmobiles%2B_5_1.productCard.PMU_V2_SAMSUNG%2BGalaxy%2BA03%2BCore%2B%2528Black%252C%2B32%2BGB%2529_samsung-mobile-store_MOBG9BZ3BU9BDUBB_neo%2Fmerchandising_4&otracker1=clp_pmu_v2_PINNED_neo%2Fmerchandising_Latest%2BSamsung%2Bmobiles%2B_LIST_productCard_cc_5_NA_view-all&cid=MOBG9BZ3BU9BDUBB

REFERENCES

- https://www.javatpoint.com/web-scraping-using-python

- https://likegeeks.com/python-web-scraping/

- https://www.w3resource.com/python-exercises/web-scraping/index.php

- https://www.youtube.com/watch?v=uufDGjTuq34

- https://realpython.com/beautiful-soup-web-scraper-python/

- https://www.tutorialspoint.com/requests/requests_web_scraping_using_requests.htm

- https://docs.scrapy.org/en/latest/intro/tutorial.html

- https://selenium-python.readthedocs.io/