ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijareeie.com

Vol. 8, Issue 6, June 2019

A Study on Web Scraping

Rishabh Singh Tomar[1]

Delhi International School, Indore, Madhya Pradesh, India[1]

ABSTRACT: Web Scraping is the technique which allows user to fetch data from the World Wide Web. This paper gives a brief introduction to Web Scraping covering different technologies available and methods to prevent a website from getting scraped. Currently available software tools available for web scraping are also listed with a brief description. Web Scraping is explained with a practical example.

KEYWORDS: Web Scraping, Web Mining, Web Crawling, Data Fetching, Data Analysis.

I.INTRODUCTION

Web Scraping (also known as Web Data Extraction, Web Harvesting, etc.) is a method used for extracting a large amount of data from websites, the data extracted is then saved in the local repository, which can later be used or analysed accordingly.

Most of the websites' data can be viewed on a web browser, they do not provide any explicit method to save this data locally. Manually copying and pasting the data from a website to a local device becomes a tedious job. However, web scraping makes things a lot easier. It is the automated version of the manual process; the computer takes care of copying and storing the data for further use in the desired format.

This paper will describe the legal aspects of Web Scraping along with the process of Web Scraping, the techniques available for Web Scraping, the current software tools with their functionalities and possible applications of it. We will also be writing our own Web Scraper in Python for providing a real-world example for all the discussed topics

II. TECHNIQUES OF WEB SCRAPING

This section provides insight on the different techniques used for scraping the data from a website.

- **Manual Copy and Paste**, is the best option in some situations, which are:
  1. When the data is minimal.
  2. When setting up automated scraping would take longer than scraping the data itself.
  3. When the data being scraped does not require a repetitive task.
  4. When security measures prevent automatic data fetching from a website.
- **HTML Parsing**, Many websites doesn't provide their data in .csv or in .json formats for easy access, instead the server displays the information as a HTML page. A simple HTML page structure is provided in Fig. 1 above. The analysis of HTML will provide repetitive elements, with another tool/script we can Search for such elements in each page as a source for data.

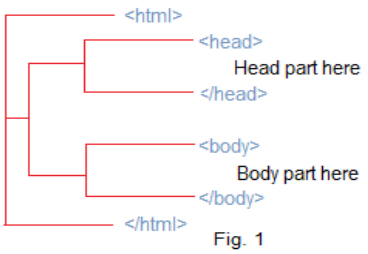


Fig. 1

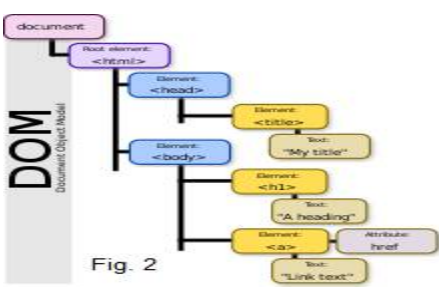


Fig. 2

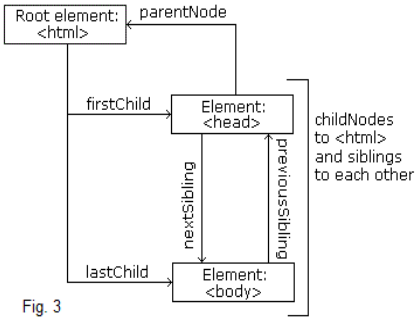


Fig. 3

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijareeie.com

Vol. 8, Issue 6, June 2019

- **DOM (Document Object Model) Parsing**, is an evolution to HTML Parsing based on developments of languages and browsers. DOM is heavily used in CSS (Cascading Style Sheet) and JavaScript, integration of DOM provided new possibilities for addressing some specific parts of the webpage. Web Scraping uses DOM for easier navigation through the webpage content. (Figure 2, shows a DOM structure)

- **XPath**, XPath is similar to DOM. XML Path Language, as the name suggests usage for XML documents is also applicable to HTML format. It provides more structured webpage than DOM and has possibilities to address segments of webpage.(Figure 3, shows a document structure in XPath)

- **APIs**, Application Program Interface, provides an application as a communicating partner. A standard HTTP Request sent to an API Endpoint returns an answer from server. Each API has its own specification and options. The format of the answer can be set as option in the request. For API communication JSON is most widely used

III. SOFTWARE TOOLS FOR WEB SCRAPING

There are plenty of software tools and frameworks available for Web Scraping. These software attempts to recognize the data structure of a web page or provides an interface which removes the necessity of writing a web scraping code manually. Some soft wares can also extract data directly from an API.

- **Desktop Software**
  In these type of soft wares, website's data is downloaded, parsed and saved locally. These require an active broadband connection. For web scraping in a desktop software, workstation of RAM (Random Access Memory) size 8GB is considered minimum.
  1. **ParseHub**is a desktop web scraping software that supports complex data extraction from websites which use AJAX, JavaScript, redirects and cookies. The data is analyzed by a machine learning program, which returns relevant output.
  2. **FMiner** provides visual configuration with scripting features. Whether you want to scrape data from simple web pages or carry out complex data fetching projects, both scenarios are possible with FMiner.
- **Cloud Software**
  The major advantage of a cloud software than a desktop software is that the process of Web Scraping is done in backend on a cloud server. There is no minimum RAM size required. In some situations, the location of server is very important. For instance, some webpages are not fully accessible from Asia in their full range. Whereas, a cloud server based in Europe ISP (Internet Service Provider) may be able to access other sections of such a website. Cloud Solutions are generally financed by subscription fees. The price depends on allocated resources.
  1. **Dexi.io** has a very sleek user interface (UI). Web Scraping starts with creation of a robot. The user needs to select requested areas and define the robot tasks from a drop down menu
  2. **Octoparse** has a desktop client. The client communicates with the cloud and submits the task. The tasks are then processed in the cloud and then sent back to the client. The UI of Octoparse is complex, the user has to create a task and define the steps. The task then executes on the cloud and returned to the client for output.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijareeie.com

Vol. 8, Issue 6, June 2019

- **Programming Libraries**

There are a lot of programming libraries available for building your own web scraper. These frameworks allows the developer to use generic modules and focus on more specific areas of the scraper. Frameworks for 4 different

	Dynamic Content	DOM navigation	DB integration	Programing language
Scrapy	YES	YES	YES	Python
Goutte	NO	YES	YES	PHP
Capybara	YES	YES	YES	Ruby
Jaunt	NO	YES	YES	Java

Figure 4.Features of Programming Libraries (Adamuz, 2015)

programming languages are shown in figure 4.

IV. APPLICATIONS OF WEB SCRAPING

In today's world Internet, data plays a major role in every field. Companies can make fortunes, if they utilize this data properly.

Candidates can use data to analyze what requirements are companies demanding and build their profiles accordingly. Data is very important everywhere and a Scraper can give you access to the data. Below are some applications of how people use scraped data.

- **Real Estate Listings**: Businesses in this area are constantly using scraped data to gather already listed properties. All MLS (Multiple Listing Services) companies are using it. Many real estate agents are using web scraping, but for specific websites.
- **E-Commerce Companies**: E-Commerce companies are in a perpetual war of monitoring prices of products of their rivals. Any business owner running a B2C (Business-to-Consumer) or even a B2B (Business-to-Business) platform, needs to get into the game and participate in these wars. This is only be possible by real-time tracking of prices of products on their own website and their competitors' website as well. To facilitate this, a system based on web scraping is essential.
- **Brand Monitoring**: In today's world, a company needs to know what public thinks of them more than marketing. One wrong tweet, one bad review by any famous person, and the company's done. For these cases, web scraping can help a lot. A company can get data from different review websites and analyze them for finding their weak spots and work on them. They can scrape social websites to see, if they got featured in any stories, positive or negative. A good public image can be maintained with web scraping at your disposal.
- **Recruitment**: The HR teams needs to know two things, the job position to be filled and the candidates to fill them. With web scraping, they can scrape websites like LinkedIn to find candidates for their vacancies. This is better than manual searching and can be used by recruiters, looking for suitable candidates to fill vacancies.
- **Search Engine Optimization**: A SEO can scrape Google, Bing, and DuckDuckGo, for their website's ranking for a given keyword. He/she can scrape competitors' blog for keyword extraction. However, they are many paid tools for these purpose but, Web Scraping gets the job done.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijareeie.com

Vol. 8, Issue 6, June 2019

- **Academic Research**: Some websites are filled with important data for researches, a student can scrape websites for research purposes.

V. METHODS TO PREVENT WEBSITE GETTING SCRAPED

No website owner wants their data to be scraped.But, automated data extraction is inevitable, if a website presents data in such a way that browser can extract and render data for the visitor. While, it is impossible to completely prevent your data from getting scraped, there are many ways by which a site owner can make it difficult for a scraper to get the data.

1. **Monitor Logs and Traffic Patterns,** if you are receiving thousands of requests from a single IP, then it might be an automated action. Blocking requests from computers that are making them too fast is usually a measure one can take.
2. **Use CAPTCHAs,** these are set of problems which a human generally finds easy, but a bot has a hard time with. This technique separates humans and computers, you can deploy a CAPTCHA if a particular client has made dozens of requests in a very short time.
3. **Regularly Change Your HTML,** scrapers rely on HTML for data extraction. If the HTML is changed frequently, or is inconsistent then a scraper might give up on your website.
4. **Create "Honey Pot" Pages,** honey pots are pages that a human visitor would never visit, but an automated bot clicking every link on the page will visit. These links can be disguised with the page background, and the user who have visited this link is no one other than a bot preying your data.
5. **Embed the Information in Media Objects,** most of the scrapers just pull out strings from your HTML. If the data is in the form of media objects such as, image, pdf, video, or non-other text formats, then scrapers have to find a way to pull out information from media object.However, this step is not suggested as it not only make your website slower but difficult for disabled users, and updating content becomes very difficult.

VI. PRACTICAL EXAMPLE OF WEB SCRAPING

For the purpose of making the process more clear, we will see practical presentation of web scraping.

1. **Task**
   For the purpose of demonstration, we will scrap the data from Instagram. The goal is to download all of the posts of the target automatically and save it locally.
2. **Our Scraper**
   For this demonstration, I have written a scraper in Python using Selenium module to scrap the required data. The algorithm of the scraper is provided in figure 4.
3. **Results**
   On running scraper.exe, a command prompt window opens which asks for the target's username. After, getting the input the scraper opens chrome, logs into Instagram and completes its job. The posts are then stored in **ScrapedImages** folder in **C** drive.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

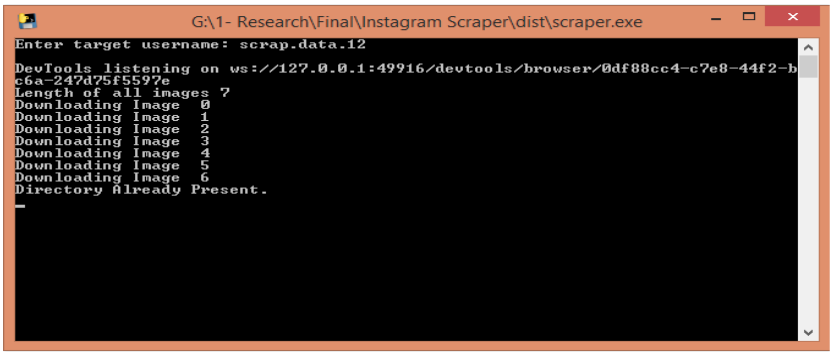
International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijareeie.com

Vol. 8, Issue 6, June 2019

The code can be found here. The scraper can be downloaded from here.



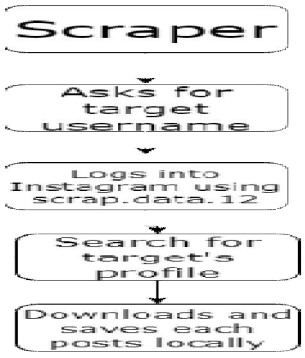


Fig. 4

VII. CONCLUSION

Depending on primary purpose, different websites can be scraped. Different techniques can be used to extract data, taken amount of data, periodicity, and required outcomes into consideration. With the amount of tools and techniques available for a scraper to fetch data from a website, a site owner can use some techniques to prevent the website getting scraped.

REFERENCES

[1] "Web Scraping" Available: https://en.wikipedia.org/wiki/Web_scraping
[2] P. Kolari, A. Joshi, "Web mining: research and practice Computing in Science & Engineering", *IEEE Transactions on Knowledge and Data Engineering*, vol. 6, no. 2, 2004
[3] VojtaDraxl, FH Technikum Wien, "Web Scraping Data Extraction from Websites", Bachelor Thesis Paper.
[4] Jakob G. Thomsen, E. E. B. a. M. S., 2015. *WebSelF: A Web Scraping Framework*, s.l.: IT University of Copenhagen.
[5] Emilio Ferraraa, P. D. M. G. F. R. B., 2014. Web data extraction, applications and techniques: A survey. *Knowledge-Based Systems*, Band 70, pp. 301-323.
[6] B.C., B., 2016. Scraping Data. In: Data Wrangling with R. Use R!.. Cham: Springer.
[7] Berlind, D., 2015. *APIs Are Like User Interfaces--Just With Different Users in Mind.* [Online]
[8] Ryan Mitchell, "Web Scraping with Python: Collecting Data from the Modern Web", 2015
[9] Hartley Brody, "Preventing Web Scraping: Best Practices for Keeping Your Content Safe", 2014
https://blog.hartleybrody.com/prevent-scrapers/
[10] Selenium Documentation
https://www.seleniumhq.org/docs/
[11] Beautiful Soup Documentation
https://www.crummy.com/software/BeautifulSoup/bs4/doc/
[12] SiceloMasango, "Web Scraping using Python", 2018
https://www.datacamp.com/community/tutorials/web-scraping-using-python
[13] APIs Are Like User Interfaces--Just With Different Users in Mind.
https://www.programmableweb.com/news/api-economy-delivers-limitless-possibilities/analysis/2015/12/03