Web Scraping Approaches and Performance

Abstract:

Web scraping is nothing but extracting data available on websites. In modern world there are number of businesses which rely on data, in such cases web scraping is the tool one needs to be aware of. Web scraping is computerized method to gather data from website and store it in the form of unstructured HTML, which later on can be transformed into a digital structured document. Using that document and analyzing it, can help user to make certain decisions required for his/her business or any work. But along with the benefits, there are also something called web scraping bots, which are set loose to make thousands of requests to the website, causing it to be working inefficiently. So, we are going to talk about how the web scraping is useful and also what improvements are needed to protect website from web scraping.

Introduction:

As internet being the gold mine for data, no matter whatever the topic is, one can easily get information they are looking for. This information can also be collected from the websites and store them for analysis, the process is known as Web Scraping. As AI has taken over in most of the field, to extract data from website bots are used. Which in turns causes high internet traffic which sometimes results into damage to organization. Also, according to the Bot Activity Report, 2/3rd of the bot traffic is malevolent. Also, there's one more type of scraping called database scraping which is similar to web scraping and also the end goal is same (i.e.; to collect data). Database scraping is considered to be the most advanced version of scraping. Data scraping means to mine the data and try to find the pattern/trend hidden under the data. In [5] research paper the authors went through different methods of scraping and tried to figure out the performance of scrapers. Eight categories of websites were chosen; they were E-commerce, Educational, real-estate, stock-data, movie, sports, news and travel.

Approach and Tools:

Here the authors go through various approaches for the web scraping which includes the ones from the very basic and the one which is currently in a great shape in scraping the data from websites without identifying as a bot. These are some of the approaches which are talked about in the paper:

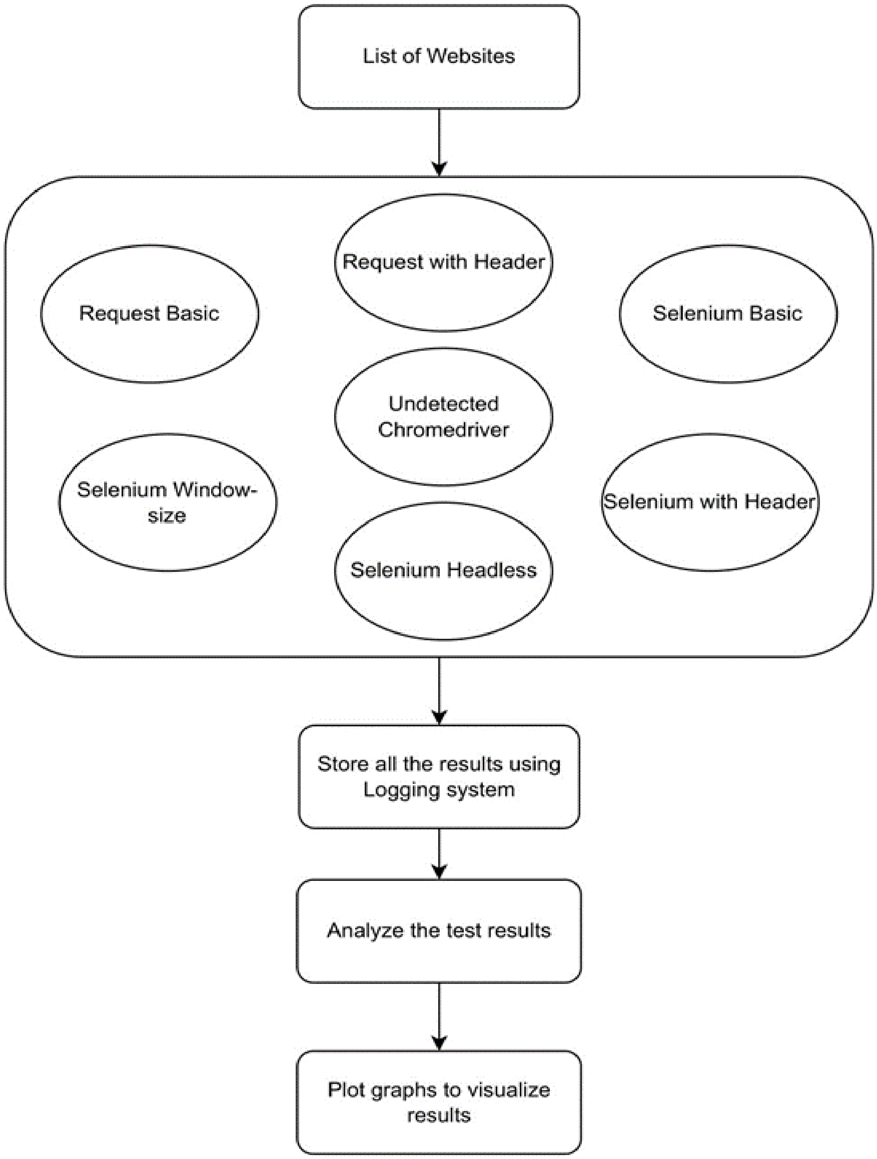
- Basic requests library tests: This being the basic way of web scraping, in these the http request is made to the website's server for retrieving the data on its page and store it somewhere.
- Requests library with header: A request with header means a HTTP header is used in an HTTP request to provide information about the request context, so that the server can tailor the response. If header doesn't look familiar or valid to the website's server then it directly blocks the request made.
- Basic Selenium test: Its an Automation tool for web scraping. It uses the web driver protocol to automate processes on various popular browsers. There are different drivers for different browsers. In this paper the author had used chromedriver across all the selenium tests.

- Selenium with header: Its similar to previous one but with an extension of header in request. The role of header is to pass information like application type, operating system, etc. to the website’s server to get the results in proper format the user has requested.
- Headless Selenium test: It runs the test without any header which in turn helps the user to have greater testing reach, Improved speed and performance and Multitasking. This was included in the test by the author because many modern-day websites are now able to identify the headless browsers as a bot.
- Headless Selenium test with window-size argument: This test was included because the modern-day websites have now started declaring the headless browsers as bots. So, the window size can help them to interact with the web elements.
- External library test: This test was included by the authors because it changes the keywords to alternative which in turns helps to protect it from anti-bot services.
- Timed test using Selenium: This was included in the test because in real world the scripts are executed for hours and days, whereas the website allows the scripts to run only limited number of times; to keep record of it, this test was included by the authors.

The tools used by the authors are Python programming language and some of the libraries like Requests library, Selenium library and Undetected Chromedriver library which helps in web scraping.

Implementation:

Top 15 websites were selected from each of the eight categories which were selected for the implementation for this paper making it total of 120. The flow diagram of the implementation done by the authors is shown figure 1.1. So, here’s how the flow of implementation goes, at first all the websites are provided and all the eight approaches mentioned above are applied to each one of them and the results obtained from that are stored using the logging system (it sores basically the record of activities related to a specific server mentioned). The results are analyzed on the basis of some parameters and finding out which parameters have high impact on the results obtained. Then it is displayed in the graph format to visualize it and draw conclusion.



[5]

Figure 1.1

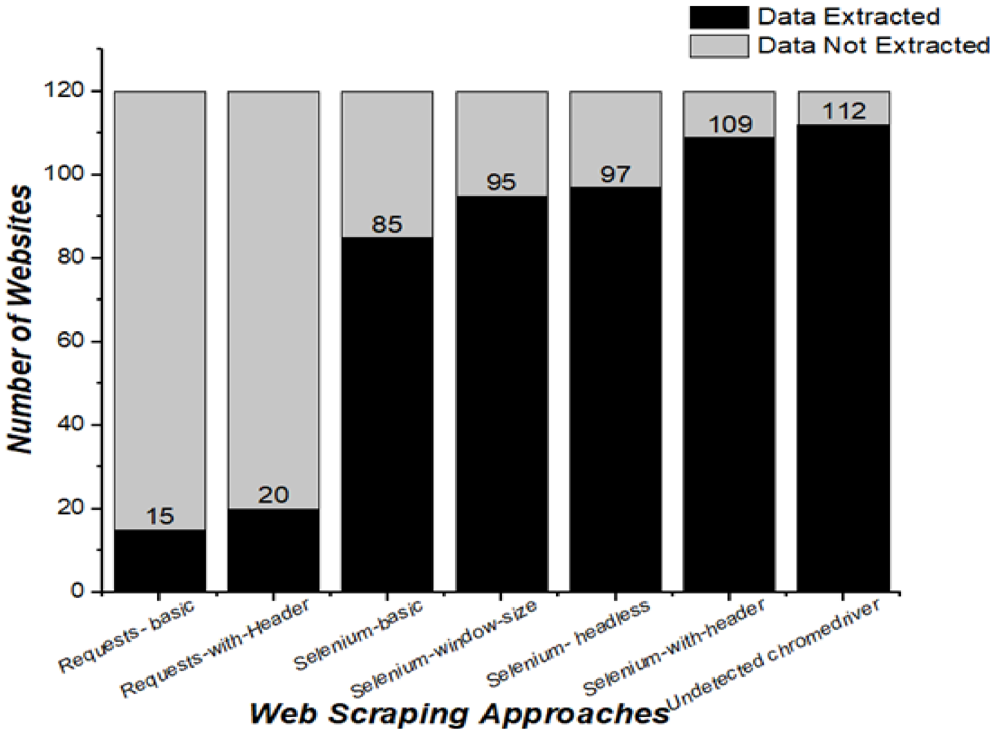
Requests Library tests: In these tests the request was sent to the server and the response code which was sent by the server was checked. This test was performed in two parts, one where the request was sent to the server along with the header and another being headless request to the server. The execution time was around 5 minutes for 120 websites which is quite impressive.

Selenium and External library test: This test was performed in four parts, the basic test without any header, using headless mode, using user-agent and the last one being along with window sized argument. This test was computationally expensive and also time consuming.

Timed test: This test had the similar approach as the above ones, but the time was the constraint in this as it only stores the records which don’t goes beyond the time limit. This test was also very expensive in terms of computation and time.

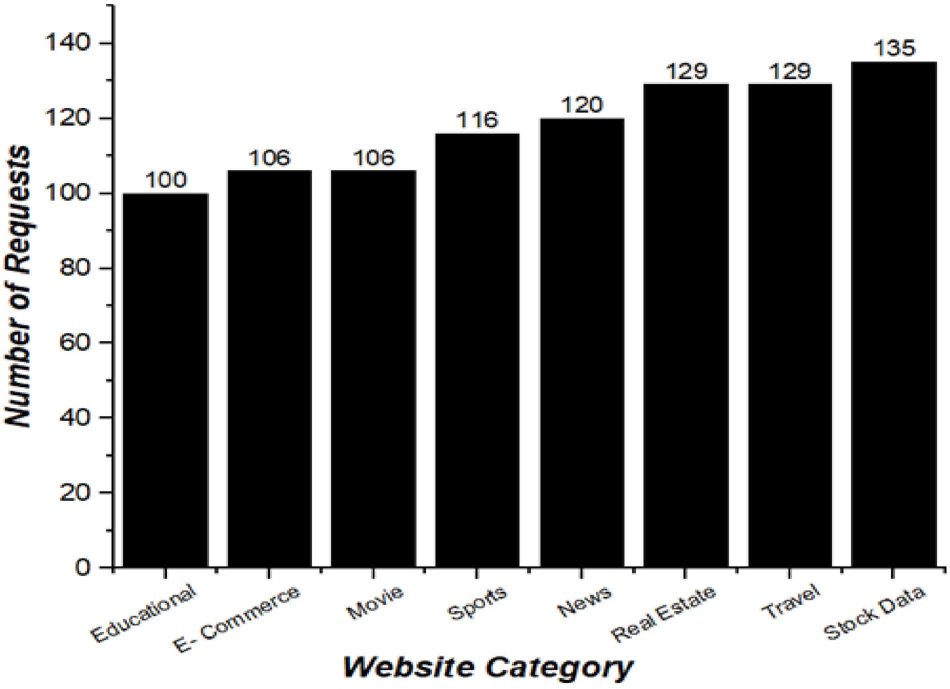
Evaluation of Results:

The results obtained are as follows:



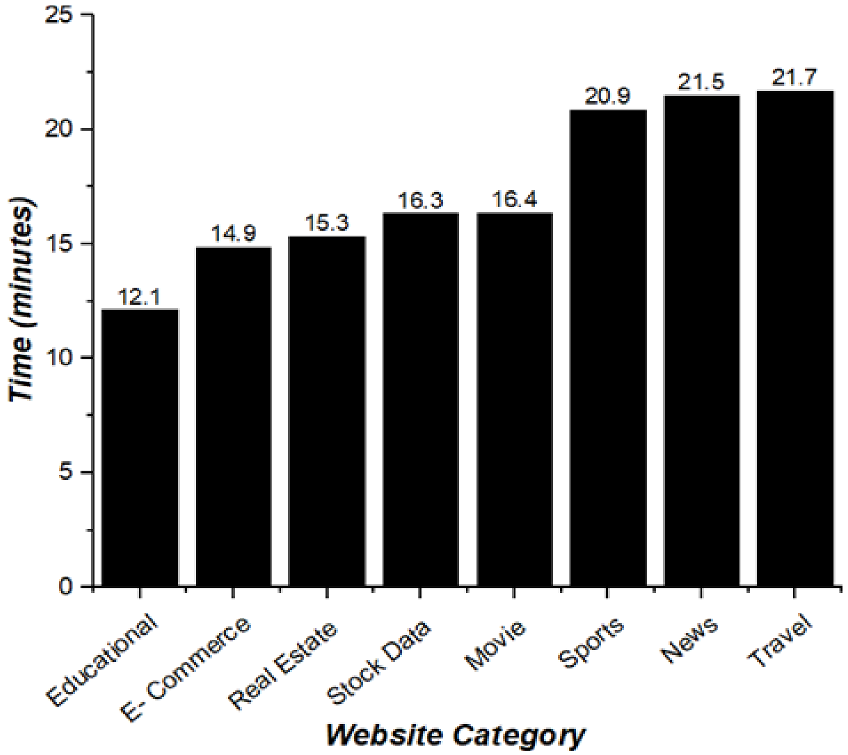
[5]

From the above figure, we can say that as the approaches went from basic to modern, the chances of getting the data extracted from the website got increased. We can evaluate that the modern approach of using undetected chromedriver provides the highest efficiency from all the approaches mentioned.



[5]

The above figure shows how many numbers of requests are made before the website recognizes it as a bot. As we know that nowadays, E-commerce websites are targeted more than any other because everyone wants to know the best deal of any product before buying it. Also, many people who wants to start their business and want to know about the trend, web scraping can be helpful.



[5]

Here in the above figure, it shows that after how much time of scraping, the server detects the bot and block them. The results obtained are similar to that of the number of requests which were made on website's server. The educational and E-commerce websites find out the bot within a less amount of time.

Conclusions:

From the above evaluation of effects of web scraping methods and techniques on a modern-day website we can conclude many results like, nowadays many websites are tended to get attacked by such bot scrapers, also we get to know why the tools like selenium is used more widely in such field today. One can easily depict that which websites are very well protected from such bot attacks (i.e., Educational and E-commerce). For future work [2] we can think about adding more libraries for testing; also, we can add more parameters for the tests to figure out which one works better or which combination of the parameters works better. Also, to increase the number of requests made to the web server can be increased by using the interactivity and navigation capability.

References:

[1] Erdinc¸ Uzun,” A Novel Web Scraping Approach Using the Additional Information Obtained From Web Pages,” IEEE Access, vol. 20,April 2020, 10.1109/ACCESS.2020.2984503.

[2] Eric C. Dallmeier,” Computer Vision-based Web Scraping for Internet Forums,” 2021 7th International Conference on Optimization and Ap- plications (ICOA) ,May. 2021, 10.1109/ICOA51614.2021.9442634.

[3] https://365datascience.com/tutorials/python-tutorials/request-headers-web-scraping/

[4] https://www.analyticssteps.com/blogs/what-web-scraping-top-5-tools-web-scraping

[5] Ajay, Naveen, Rohith S, Rohith R, Rohan, Kamalesh, “ Web Scraping Approaches and their Performance on Modern Websites ”, IEEE Xplore, 20 September 2022, 978-1-6654-7971-4/22

[6] https://towardsdatascience.com/how-to-use-selenium-to-web-scrape-with-example-80f9b23a843a

[7] Bhavya Bhardwaj,Syed Ishtiyaq Ahmed,J Jaiharie,R Sorabh Dadhich, M Ganesan,” Web Scraping Using Summarization and Named Entity Recognition (NER),” 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), March 2021,10.1109/ICACCS51430.2021.9441888.