MASKED WEB CRAWLER : THE SOLUTION

DIVY PATEL
VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE

NATURE OF PROBLEM

A traditional technique to extract unstructured data from websites is manually browsing websites and copying down the useful data to a local file. But, this could be hassling when a firm want to scrape hundreds or thousands of websites and at the same time problem of dynamic nature of data over websites. To automate the process of scraping data, smart bots are programmed which recursively extracts data from a website, attaining greater depth of browsing. But with bots, websites too became intelligent to stop bots adding more overhead to their servers. Websites incorporates Captcha, honeypots and many other technical lingos. But sooner, solution against this techniques were worked out to allow scraping of websites without causing any significant overheads on servers.

Now coming to problem which is faced after scraping the data, which is segregating it into meaningful classes from unstructured data. A general solution to this problem can be solved by designing classifiers and applying fundamentals of NLP (Natural Language Processing) on the corpus.

BRIEF SOLUTION

The solution can be made by reverse engineering the mechanisms, the websites administer to avoid web crawling. Generally, websites blacklist the IP address which makes greater number of requests per second. Websites also associate captcha with the website to detect a bot. Websites tend to detect patterns in requests from specific IPs. The actions of real users are generally very random in nature which is completely opposite in case of programmed bots or web crawlers. This segregates real users and the web crawlers. So generally a

random time interval is maintained between requests, so as to avoid javascript to detect illegal traffic. Sometimes complex download delay algorithms can be used. Highlights of a AutoThrottle Setting algorithm are:

- The download delay is calculated by getting latency time that the server required to respond.
    - Download_Delay = Latency Time / N
      where N is number of requests to be made in concurrency.
- At times when server responds 200 error codes, the responses are generally faster than successful responses this makes bot even more faster to request which can be fatal, as there is possibility of been blocked by websites.
- So in such case the latency value of previous successful request is taken as target latency time.

The IP of a system making requests are rotated by using pool of IP addresses which are usable. Some websites needs javascript code to be executed before getting response from the server. In this scenarios, we use headless browsers. Headless browsers are same as normal browsers in functionality but without any graphical user interface. Some headless browsers also offer some advanced tools for its super users for web scraping. A framework based on headless chrome "pupperteer" can be used. The framework is easy to incorporate with majorly used languages like Node Js and Python.

After successfully scraping data from websites, the data can be cleaned and made workable to go through token annotation analysis and classifiers. The preprocessing involves dealing with capitalization, stop words which are very frequently occuring words and which is of no use for further analysis. The data is lematized, tokenized and PoS (Part of Speech) tagged for

segregating significant meaningful words from a large corpus. Token annotation analysis is extracting useful named tokens from preprocessed data on the basis of large trained corpuses and feeding it to various classifiers to classify the offers on the basis of class of products.

TOOLS AND LANGUAGES

- NodeJs and Python for scripting.
- Srapy module for implementing AutoThrottling algorithm.
- Pupperteer framework for controlling headless chrome browser.
- SpaCy, a efficient Natural Language Processing Python library for token annotation analysis.
- NLTK library for preprocessing unstructured data to workable data.
- TensorFlow and Scikit Learn for modelling classifiers.