A Review on Web Scrapping and its Applications

Conference Paper · January 2019

DOI: 10.1109/ICCCI.2019.8821809

3 authors:

Vidhi Singrodia
Amity University Kolkata
1 PUBLICATION 49 CITATIONS
SEE PROFILE

Anirban Mitra
Amity University Kolkata
95 PUBLICATIONS 553 CITATIONS
SEE PROFILE

Subrata Paul
Facts Aout Color
21 PUBLICATIONS 95 CITATIONS
SEE PROFILE

2019 International Conference on Computer Communication and Informatics (ICCCI -2019), Jan. 23 – 25, 2019, Coimbatore, INDIA

A Review on Web Scrapping and its Applications

Vidhi Singrodia, Anirban Mitra
Amity University
Newtown, Kolkata
vidhisingrodia1@gmail.com, mitra.anirban@gmail.com

Subrata Paul
Research Scholar, MAKAUT
Kalyani, NADIA
subratapaulcse@gmail.com

Abstract- Internet grants a wide scope of facts and data source established by humans. Though, it shall consist of an enormous assortment of dissimilar and ailing organized data, challenging in collection in a physical means and problematical for its usage in mechanical processes. Since the recent past, procedures along-with various outfits have been developed to permit data gathering and alteration into organized information to be accomplished by B2C and B2B systems. This paper will focus on various aspects of web scraping, beginning with the basic introduction and a brief discussion on various software's and tools for web scrapping. We had also explained the process of web scraping with an elaboration on the various types of web scraping techniques and finally concluded with the pros and cons of web scraping and an in detail description on the various fields where it can be applied. The opportunities taking an advantage of these data are numerous which shall include expanses concerning Open Government Data, Big Data, Business Intelligence, aggregators and comparators, development of new applications and mashups amongst formers.

Keywords- Web Scrapping, Internet, Big Data, Business Intelligence.

I. Introduction

Presently the internet world is enormously enormous considering the web pages with huge quantity of explanatory substances obtainable with dissimilar designs such as text, graphical, audio-video, etc. which will focus on the contradiction in repossession of facts owing to the insignificance regarding the fact user is seeing. The data that is displayed by the websites can only be viewed using a web browser. They do not offer the functionality to save a copy of this data for personal use. The only option then is to manually copy and paste the data exhibited in the website at browser into the hard drive of our computer which is quite tiresome job. This is where web scraping comes into play.

Web scraping (also known as Screen Scraping, Web Data Extraction, and Web Harvesting etc.) is a procedure of automatic web data extraction instead of manually copying it. It is a technique in which meaningful data from the HTML of websites are extracted and stored into a central local database or spreadsheet. It uses the URL of the website for this purpose. It is performed by web scrapers with the help of specially coded programs. It can either be traditionally assembled for some precise website or can be one that is organized easily for working with any website. The goal of a Web scraper is concentrated on conversion of unstructured data while preserving in organized databases. Few Web scraping procedures are HTTP programming, DOM parsing, and HTML parsers. The data generated is later used for retrieval or analysis. It is a huge advantage as it provides us with error-free data, saves our time to give lightening quick results and stores all the data in one place. We can also choose the format in which it should be available to us. This allows an ease of access and makes life easier in analyzing the data.

Web scraping is presently cast-off on various aspects including online price comparison, weather data monitoring, website change detection, Web mashup, Web research and Web data integration. Further, it may be noted that Web scraping might be alongside the tenures of usage of few websites.

II. The necessity to scrape web sites and PDF documents

Roughly 70% of the facts stimulated in Internet is obtained from PDF documents which is unstructured and difficult in handling. Moreover, a web page is an organized format (consisting of HTML code), though in a non-reusable approach. This vast quantity of information distributed but confined of this type of design is typically termed as “the tyranny of PDF”.

Ensuring the key possibility of this document (HTML documents), its organized nature reproduces the potentials exposed with scraping procedures. Web scraping procedures and tools trust on organization as well as assets of the HTML language. This will enable the job of scraping for tools and robots thereby concluding humans apart from tedious monotonous and faulty duties of physical data retrieval. To conclude, these tools compromise the data in approachable formats for advanced dispensation and integration: JSON, XML, CSV, XLS o RSS [1].

A. The API mana

Data collection a handcrafted technique is accurately incompetent in searching, copying and pasting data in spreadsheet for processing. This is a monotonous, annoying and frustrating technique.

Consequently there is a need for automation of the procedure. Scraping documents of this type will automate as a popular tool for providing an API which will enable entrance to the content produced. An API (Application Programming

978-1-5386-8260-9/19/$31.00 ©2019 IEEE

Interface) is a procedure for the association of two applications, allowing in sharing of data. Scraping tools streamlines an URL, an Internet address which may be noticed as an address bar of a web browser, permitting an access to the scrapped data [1].

III. Software for Web data scraping

The prevalent methods in the implementation of Web data scraper can be grouped in three main categories: (i) libraries for general-purpose programming languages, (ii) frameworks and (iii) desktop-based environments.

A. Libraries

The widely available method commonly worn by bioinformaticians comprises of construction of individual Web data scrapers with the usage of a known programming language. Generally, third party libraries grant access to the site with the implementation of client side of the HTTP protocol, while the parsing of recovered substances are done using built-in string functions like comparison of regular expression, tokenization and trimming. Complicated form of parsing like HTML tree building and XPath matching can be provided by Third-party packages.

Amongst the widely accepted site access libraries is libcurl (http://curl.haxx.se/) which sustains the main characteristics of the HTTP protocol, together with SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form-based upload, proxies, cookies and HTTP authentication. Furthermore, it has beneficial attachments with numerous programming languages.

Perl, commonly handled by bioinformatics, combines the WWW::Mechanize Web automation module. Additionally, it permits usage of XPath around supplementary modules.

In Java, the Apache HttpClient package simulates HTTP key characteristics including entire request routines, cookies, SSL and HTTP authentication which is merged with HTML parsing libraries like jsoup (http://jsoup.org/). Java even sustains XPath and serve numerous HTML cleaning libraries, like htmlcleaner.

Correspondingly, the BeautifulSoup, Python HTML parsing library can be merged alongside language native support for HTTP connections. Even in Unix-like environments using the piping operating system command-line programs within shell scripts, programmers will facilitate in the creation of Web data scrapers with meagre lines of code. Programs resembling curl (libcurl) and wget apply HTTP client layer, though services such as grep, awk, sed and cut and paste may effectively be used in parsing and transforming matters appropriately.

Considering server side robots, distinctively operating within Web applications, a 100% adaptable with the programming language (mainly PHP, Perl or Java) is suggested [2].

B. Frameworks

Several limitations have been observed while the usage of general-purpose language in creation of robots. At times, numerous libraries requires integrating Web scraping technologies to the API world for accessing the Web, parsing and deducing useful information from HTML documents. Moreover, robots are considered to be feeble software, immensely influenced with the alteration of HTML of extracted resources thereby requiring consistent sustenance. Considering compiled languages, like Java, an alteration in implementation of robot facilitate re-recompilation with redeployment of the total application.

Scraping agendas shall demonstrate an added integrated explanation. For instance, Scrapy, an influential Web scraping agenda in Python outlines robots as classes sourced from BaseSpider class, describing a collection of ‘starting urls’ and a ‘parse’ function entitled for every Web iteration.

XPath expressions are inevitably used in parsing Web pages and extraction of Web contents. Alternative agendas shall demonstrate domain-specific languages (DSL), projected for specific domains and consequently robots are preserved as self-governing and peripheral objects. An instance is Web-Harvest, a Web data scraping agenda for Java describing the Web extraction procedures with XML (by means of visual environment) and comprised of numerous ‘pipelines’, including procedural commands, like variable definitions and loops, along with numerous primitives, for example ‘http’ (for retrieval of Web contents), ‘htmlto-xml’ (for cleaning HTML) and ‘xpath’ for extraction of content.

Additional instance of Java Web data scraping agenda is jARVEST, which besides defining DSL, uses JRuby for an added application of dense robots.

C. Desktop-based environments

Desktop applications appear the necessities of layman programmers. This type of tools is authorized with graphical design environments enabling the conception alongside conservation of robots. Generally, the software comprises of combined browser allowing user to steer towards the target Web with collaborative selection of elements from extracted page, thus evading descriptions of regular expressions, XPath queries or supplementary mechanisms. The main problems of desktop solutions are its commercial circulation with restricted API access making it problematic for embedding scrapers within additional programs (which it is often an obligation) [3].

IV. The web scraping system

Web scraping is the method of spontaneous collection of information from the World Wide Web. It is an arena with vigorous advancement which shares an objective with semantic web vision grounded on a device that traverses with abstraction of website constituents and preserving them in a local data base. It is witnessed that, commencing a legal perspective, web scraping is contrary to the terms of usage in few websites: courts are organized for the preservation of registered contents of commercial sites from objectionable usages although the amount of defense for these contents is not evidently established. Subsequently two dissimilar explanations for web scraping are designated: preliminary one is previously accessible and castoff precisely for this experiment, whereas the succeeding one is still in the expansion phase.

A. The web scraping application grounded on JSOUP and ADaMSoft

A primary choice was developing a scraping application with creation of an Open Source library called JSOUP[4] with its integration to the ADaMSoft system[5], for dealing with entire navigation and data management complications. In reality, a numerous complications have come across in the course of this phase owing to websites which are not entirely accessible thereby making usage of machineries not completely grounded on standard html text, which makes it not simply available. Consequent to the accomplishment of this phase, two group of data have been attained comprising of text (observable by a universal user) for each website while an alternative information described for every tag in relation with the objects of html pages (for illustration, “type”, “name”, “value”). We practiced that this additional group of data significantly surges the quantity of valuable information.

B. The web scraping application grounded on the Nutch/Solr/Lucene Apache suite

The Apache suite castoff for crawling, content extraction, indexing and searching consequences is forwarded by Nutch and Solr. Nutch[6] is an extremely extensive and accessible open source web crawler which will facilitate in parsing, indexing, generating a search engine, modifying search according to requirements, scalability, robustness, and counting filter for conventional applications. Grounded on Apache Lucene and Apache Hadoop, Nutch can be arranged on a single machine in addition to a cluster, if huge scale web crawling is necessary. Apache Solr[7] is an open source enterprise search platform that is based on Apache Lucene castoff aimed in examining any kind of data; nevertheless, it is precisely utilized in searching web pages. Its main characteristics comprises of full-text search, hit emphasizing, faceted search, dynamic clustering, database integration, and rich document handling [8, 9].

There are fundamentally six stages in extraction of text-based data from a website:

1. Identification of information on the internet which is wanted to be used.
2. If this information is preserved in multiple web pages, justification of the procedure of navigation of web pages. Considering the best case situation, we will have a directory page or the URL will have a dependable design that can be recreated — e.g., www.somewebsite.com/year/month/day.html.
3. Discovering the structures on the website which ensign’s information which is to be extracted. This funds observing the fundamental HTML for finding the elements we wanted and/or recognizing certain patterns within website’s text that can be exploited.
4. Writing a procedure for extraction, formating, and saving the information which we wanted with the usage of recognized flags.
5. Circling around every websites obtained from step 2, with an application of script for every one of them.
6. Do certain overwhelming scrutiny on recently gaping data [9].

V. Tools for Webscraping

A. rvest

The rvest platform is the workhorse toolkit. The workflow characteristically is as follows:

1. Reading a webpage with the usage of function read_html() which downloads the HTML and stores so that rvest can traverse it.
2. Selection of essentials we require with usage of function html_nodes(). This function yields an HTML object (from read_html) accompanied by CSS or Xpath selector (e.g., p or span) and preserve every components which matches the selector. SelectorGadget can be supportive in this aspect.
3. Extraction of constituents of nodes being selected with usage of functions like html_tag() (the name of the tag), html_text() (every text within the tag), html_attr() (substances of a solitary element) and html_attrs() (every elements).

The rvest package comprises of certain additional characteristics like its ability in filling forms on websites and navigating websites like using a browser [10].

B. Regular Expressions

Frequently we will view a pattern in text which is needed to be exploited. For illustration, a novel variable might continually monitor a colon which comes after a single word in a new line. Regular expressions (or regex) specifically describe these patterns. They’re very fundamental for web scraping and text analysis. In R, few regex commands can be used are:

- grep(pattern, string) which revenues a string vector and returns a vector of the indices of the string which matches the pattern

```
string = c("this is", "a string", "vector", "this")
grep("this", string)
## [1] 1 4
```

- grepl(pattern, string) which revenues a string vector with length n as an input returning a logical vector of length n which says whether the string resembles the pattern. Example:

```
grepl("this", string)
## [1] TRUE FALSE FALSE TRUE
```

• gsub(pattern, replacement, string) which bargains every occurrences of pattern in string and substitutes it with auxiliary [11]. Example:

```
gsub(pattern="is", replacement="WTF", string)
## [1] "thWTF WTF" "a string" "vector" "thWTF"
```

VI. Operating standard of Web Scrapper

For understanding the thought of web scraping, together with the visual lined web services, it is significant in understanding the technical working values of the technology.

Web scraping is prepared with the usage of definite techniques on the type of data to be gathered and combined. To facilitate its achievement, a sound perception of programming, web technologies like HTML, and the arrangement of web data is necessary. This requisite information and indulgence is condensed with a web scraping API.

Automated web scraping can be classified into 3 major methods which are extensively worn by web scraping software [12].

- Syntactic Web Scraping
- Semantic Web Scraping
- Computer vision web-page analyzing

A. Syntactic Web Scraping

Syntactic web scraping mines information from the arrangement of website by parsing HTML, CSS and further distinctive web languages. For this process, numerous methodologies are followed:

i. *Content Style Sheet selectors:* They define the illustrative properties of HTML essentials which are related to components in the course of CSS selectors, characterized throughout a specific language. Therefore, CSS is one technology that provides selection and extraction of data.

ii. *XPath selectors.* Likewise CSS selectors, the XML Path Language are dissimilar languages for the choice of HTML node.

iii. *URI patterns.* It permits selection of web resources in accordance with a regular expression which is functional on resource URI. While XPath or CSS selectors are capable in selection of an element at document stage, they permit selection of credentials, i.e. resources representations, according to the possessions URI.

iv. *Visual selectors.* They can be exploited with choice of nodes. HTML nodes are provided with a group of illustrative properties specified by used browser. It is a familiar fact that humans desire in consistent web designs. Web designers thus formulate essentials of similar type to be provided with comparable visual belongings for identifying assistance. It is thus a circumstance to facilitate combination of numerous visual belongings of ingredients for identification of elements class.

B. Semantic Web Scraping.

While the semantic web cultivates, traditions and structures for treatment of semantic data, were extended. Accordingly, the data mined by syntactic web scraping can be compared with semantic web resources, for enhanced demonstration and exploitation. For doing this, numerous structures can be worn, similar to Resource Description Framework (RDF) and the Web Ontology Language (OWL). The breakdown of semantic web, formulates this method as non-preferable by the majority of relevance's.

C. Computer Vision Web page Analysis.

In conclusion, machine learning and computer vision procedures can be used for identification and extraction of information from web pages by understanding pages illustration as human being may and then relate these with css selectors [13]. An instance is diffbot[14].

VII. Strengths and Weaknesses of Web Scrapping

Here we shall be centering supplementary on topic of mechanized APIs scraping through visual interface for web scraping. The two major web services to facilitate this technology are Kimono Labs and Import.io. They mutually necessitate user for creation of an account for usage of service.

A. Scraping by Coding

To start with, the web scraping offers a possibility in getting whichever data we desire in a structured method. This arrangement may be on the basis of syntactic, computer vision or semantic abstraction technologies. An immense power of web scraping is the reality that it facilitates user in structuring data in the means to which it outfits the finest for primary project. As a whole, we can fully govern the data which we do not govern. This data is the strong point of scraping which is founded on the data being figured for website audiences. This revenues the utmost precedence of web developer for keeping this data modern which delivers scraper through topmost eminence modern data. This opinion moreover enlightens the betterment of scraping data as an alternative of consuming a public API. The essential purpose of maximum web services lies in upholding the html front end for enabling their users to view. Connected to the point's overhead lies a detail which has no consistent rate restrictions for data queries. Roughly the flawless of scraping web data are the following. Foremost, an appraisal on the design of the website, or simply the retitling of definite essentials in the CSS could proceed to a failing scraper. Furthermore we shall require programming skills for writing a scraper, and also require a server for running the scraper. Finally there is certainly no documentation about the procedure of scrapping the website. Individually case is dissimilar, and necessitates a practice for manufacturing a scraper.

B. Scraping with Visual Interfaced Services

While comparing the visual strengths and weaknesses with mechanized APIs for instance Kimono or Import.io, it is informal for starting by means of the weaknesses and explaining the assets of automatic web scrapers. For using Kimono or Import.io the user may not require any expertise in programming. The graphic CSS element collection efforts fairly healthy in mutually services with a proposal to user for creation of whole working API lacking a little programming knowledge. It similarly empowers users in generating a web scraper shorn of without requiring any server as everything goes online. A huge benefit lies on the fact that the resultant API shall use typical API assemblies in several layouts, thus the data will be distributed smoothly through extra inventors. A supplementary service which can be obtained from Kimono and Import.io lies on the statement on disastrous apprises of the API signifying the needlessness in checking whether our application is functional. The detail which the user is delivered through these facilities implies the resources which can be relied on third party services for providing the fed data. It is very significant for keeping in mind about the fact that presently both of them are in beta rank. It also revenues of the fact that its usage and value can alter in the forthcoming days. Likewise there can be few rate restrictions, somewhat like a physical scraper which has not been come across. Finally it is significant in realizing that if we have the necessary skills we can build our individual scraper shall continuously be an additional customization than some of these services. If we require in scraping data which is not effortlessly recognized via CSS, such as script produced textboxes change on the basis of context of the position of our mouse location which has the potency to be healthier in writing our own scraper [15, 16].

VIII. Typical Applications of Web Scrapper

There are numerous features which can be reflected while the usage of a web scraper. This may be generalized on the solicitations of scraping. Likewise, the imprecise authority of construction of a project on the basis of scraped data creates it challenging to differentiate projects which are dependent on scraping machineries. We can, nevertheless provide an overview on the maximum characteristic arenas the technology is used in. Through some of these arenas will provide small instances on the means by which such a project could provide. The grounds on we deliberate for being distinctive for web scraping are:

- Data mining
- Research
- Marketing
- Company competition
- Personal tools
- Data combination

Foremost, data mining, is a ground which is wide. The basic segments where data mining is frequently castoff is in the Spam business. The use of web scrapers in aggregating email addresses is the distribution of unsolicited emails which is extensively used, but difficult in proving.

Secondly in case of research, this might be conducted by all parties. Let us consider a university which is conducting a research on the usage of Twitter. Here the eminence of the data should be excellent during its receipt, without the usage of API. Iinscription of a scraping bot for Twitter, or usage of a services revealed previously will supply research institute through identical data which are being used by researchers while investigation.

Considering the marketing applications which are intimately associated with the previous usage illustrations which were under consideration. For instance while investigating the expansion of brand awareness promotion on numerous social media and classifying all the activities intended for using in future.

In addition it might be utilized by companies for keeping path of their competitive activities. Such as the case that while governing a query might arise regarding the prices of the commodity which might be effortlessly anticipated on subsequent position within the list where an expectation can be made on the usage of scraping technologies for keeping path on the finest deals made online.

Finally scraping might also be worn in structuring analogous data commencing dissimilar foundations for their combination into a single source. Such as the amalgamation of every publication on crowdfunding websites could be coupled collectively for keeping a path of the advancements in the specifed arena.

Numerous probable applications are merely inadequate with thoughts of the user/developer [16].

IX. Conclusion

Web scraping is a recognizable phrase which has expanded significance owing to the requirement of “free” data accumulated in PDF documents or web pages. Numerous professionals and researchers require the data for processing, analysis and extraction of significant consequences. Alternatively, people dealing with B2B use cases require the admittance of data from several sources for its integration into innovative applications which will offer supplementary values and novelty. Throughout this paper we have reviewed the various aspects of Web Scrapper. Starting with the tools and software for web scrapping, we have seen the operating principle, strength and drawbacks and finally viewed the applications of web scrapping system.

REFERENCES

[1]. Osmar Castrillo-Fernández, “Web Scraping: Applications and Tools”, European Public Sector Information Platform Topic Report No. 2015 / 10, December 2015.

[2]. Kanehisa M, Goto S, Sato Y, et al. KEGG for integration and interpretation of large-scale molecular data sets. Nucleic Acids Res 2012;40:D109–14.

[3]. Glez-Pen‹a et al. , “Web scraping technologies in an API world”, Briefings in Bioinformatics Advance Access, doi:10.1093/bib/bbt026, published April 30, 2013

[4]. http://jsoup.org/

[5]. http://adamsoft.sourceforge.net/

[6]. https://nutch.apache.org/

[7]. https://lucene.apache.org/solr/

[8]. James, G., Witten, D., Hastie, T., Tibshirani R. (2013), An Introduction to Statistical Learning with Applications in R, Springer Texts in Statistics

[9]. Giulio Barcaroli et Al, “Use of web scraping and text mining techniques in the Istat survey on “Information and Communication Technology in enterprises””, European Conference on Quality in Official Statistics (Wien 2014), June 2014

[10]. Grimmer, Justin. 2013. Representational Style in Congress: What Legislators Say and Why It Matters. Cambridge University Press.

[11]. William Marble, “Web Scraping With R”, stanford.edu, August 11, 2016

[12]. Carlos A. Iglesias Mercedes Garijo Jose Ignacio Fernandez-Villamor, Jacobo Blasco-Garcia. A Semantic Scraping Model for Web Resources, Applying Linked Data to Web Page Screen Scraping.

[13]. Muntasir Mashuq MichelZiyan Zhou. Web Content Extraction Through Machine Learning.

[14]. Diffbot: Extract content from standard page types: articles/blog posts, front pages, image and product pages. http://www.diffbot.com/.

[15]. Alex Gimson. This Just In: A Data Journalism Webinar with BeaSchofield. http://blog.import.io/post/this-just-in-a-data-journalism-webinar-with-bea-schofield.

[16]. Daan Krijnen, “Automated Web Scraping APIs”, mediatechnology.leiden.e