Handout 2

What is web scraping

Web scraping is used to extract or “scrape” data from any web page on the Internet.Copying a list of contacts from a web directory is an example of “web scraping”. But copying and pasting details from a web page into an Excel spreadsheet works for only a small amount of data and it requires a significant amount of time. To gather larger amounts of data, automation is necessary and web scrapers perform exactly that function.

Web scraping is performed using a “**web scraper”** or a “bot” or a “web spider” or “web crawler” (words used interchangeably). A web-scraper is a program that goes to web pages, downloads the contents, extracts data out of the contents and then saves the data to a file or a database.

Origins of the word Web Scraping

The origins of the word are most likely from the term “screen scraping” which was widely used prior to the wide use of the web to integrate non-web based applications such as mainframe “green screens” (terminal applications) or native windows applications. These “screen scrapers” would “scrape” data from one application to be used to insert them into other applications – quite a bit from Mainframe to PC applications.

With the advent of the Web or Internet, the reliance on web scraping has continued and by some accounts a huge portion (52%) of the Internet traffic to websites (excluding streaming) is comprised of bots.

The Internet at its start was just a few web pages. As the pages grew in number, it was hard to “find” these pages and some of the big names of the .com era such as Yahoo etc got their start by creating a “directory” named“Jerry and David’s Guide to the World Wide Web”. The Guide was a directory of other websites, organized in a hierarchy, as opposed to a searchable index of pages.

As you can see this approach could not scale well as the web grew. Jerry and David couldn’t possibly hand curate and add links to their “guide” fast enough. The logical next step was to automate this task – and “Web Scraping” was born !

How vital is Web Scraping

The Internet would be far less useful and terribly small without Web Scraping.

The lack of availability of “real integration” through APIs has turned Web Scraping into a massive industry with trillions of dollars in impact on the Internet economy. The amount Google alone contributes to this number – not just Google’s revenues but all companies that rely on this “search engine” – the amount if mind-boggling. McKinsey put a number of 8 trillion dollars on it in 2011 and it has only increased exponentially since.

> There is an enormous amount of data “available” on the Internet but it is hardly “accessible”.

Web scraping makes this data accessible to all kinds of applications and uses.

Web Crawling vs. Web Scraping

People often use Web Scraping and Web Crawling interchangeably. Although the underlying concept is to extract data from the web, they are different.

Web Crawling mostly refers to downloading and storing the contents of a large number of websites, by following links in web pages.

Search Engines depend heavily on web crawlers. Googlebot is an example of a web crawler. The Googlebot crawls the Internet following links from one page to another. Google then uses this information to extract all kinds of data to make its search engine useful to us all. All other search engines use their own bots in a similar manner. e.g. Bing, Yahoo, Duck Duck Go, Yandex, Baidu etc.

A Web scraper is built specifically to handle the structure of a particular website. The scraper then uses this site-specific structure to extract individual data elements from the website. The data elements could be names, addresses, prices, images etc.

For example, SERP monitoring services scrape search engine results periodically to show you how your search rankings have changed over time. They use a separate scraper for each search engine.

Uses of Web Scraping

People use web scrapers to automate all sorts of scenarios. Web scrapers along with other programs can do almost anything that a human does in a browser and more. They can order your favorite food automatically when you hit a button, buy the tickets for a concert automatically the moment they become available, scan an e-commerce website periodically and text you when the price drops for an item, etc. The uses are as infinite as the uses of the Internet.

Web scrapers have a variety of uses in the enterprise. We have listed a few below:

•**Search Engines –** One of the largest companies whose whole business is based on Web Scraping. It is hard to imagine going by one day without using Google.

•**Price Monitoring –** Scrapers can gather data about specific products from E-commerce websites such as Amazon.com, Walmart, eBay, etc. There are many price comparison and competitor monitoring services built on top of web scraping.

•**Sales and Marketing –** Scrapers can be built for business directory websites to extract contact details. A combination of web scrapers can enrich the data with emails, phone numbers and social media profiles for sales or marketing campaigns.

•**Content Aggregators –** almost all the content aggregators use web scraping. News Aggregators scrape news websites frequently to provide updated news data available to its users. Job Aggregators scrape job boards and company websites and grab latest job openings. Services such as Pocket, Instapaper, Flipboard, etc. extract articles from pages using scraping techniques and augment the data with Machine Learning.

•**Sales intelligence –** Tools such as Full Contact and ClearBit provide details about a lead based on just an email address. They also depend on multiple types scrapers that scour the web to provide you with more information.

•**SEO Monitoring –** SEO Tools such as Moz, Majestic, SEMRush, a-hrefs, etc. scrape Google and other search engines daily to tell business how they rank for the search keywords that matter to them. They also extract backlinks, do SEO audits, etc. using web scraping. These are scraper built upon the data initially scraped by the Search engine scrapers.

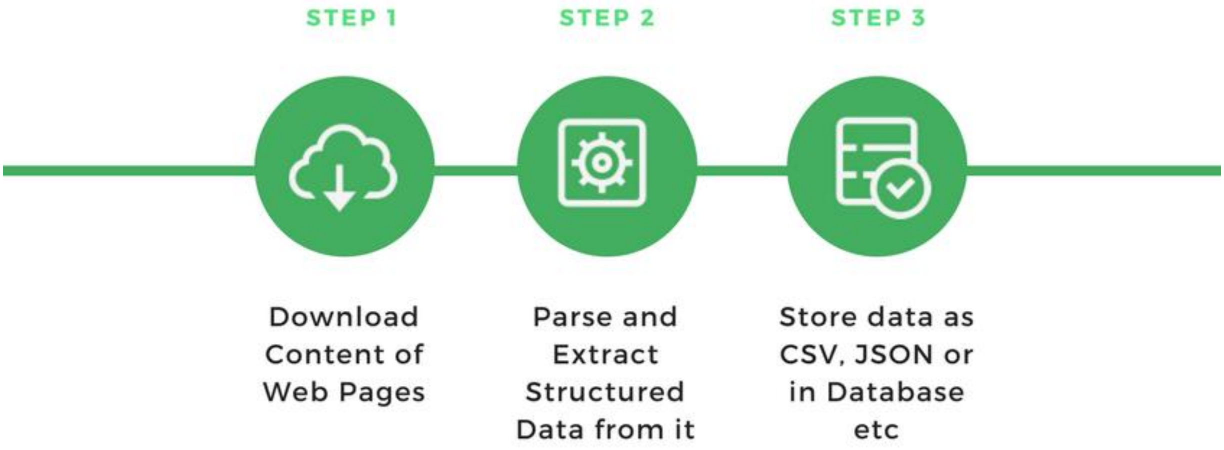
•**Training datasets for Machine Learning –** Not all data on the web is readily available as a structured dataset, nor do all websites have an API. Many data scientists depend on data collected through web scrapers, for publishing reports and training their machine learning models.

•**Data for Research** – Researchers and Journalists spend a lot of time manually collecting and cleaning data from websites. These days many of them use web scrapers to automate most of this manual labor.

We have barely touched the tip of an iceberg when it comes to applications of web scrapers.

How does a web scraper work?

A web scraper is a software program or script that is used to download the contents (usually text based and formatted as HTML) of multiple web pages and then extract data from it.


STEP 1
STEP 2
STEP 3
Download Content of Web Pages
Parse and Extract Structured Data from it
Store data as CSV, JSON or in Database etc

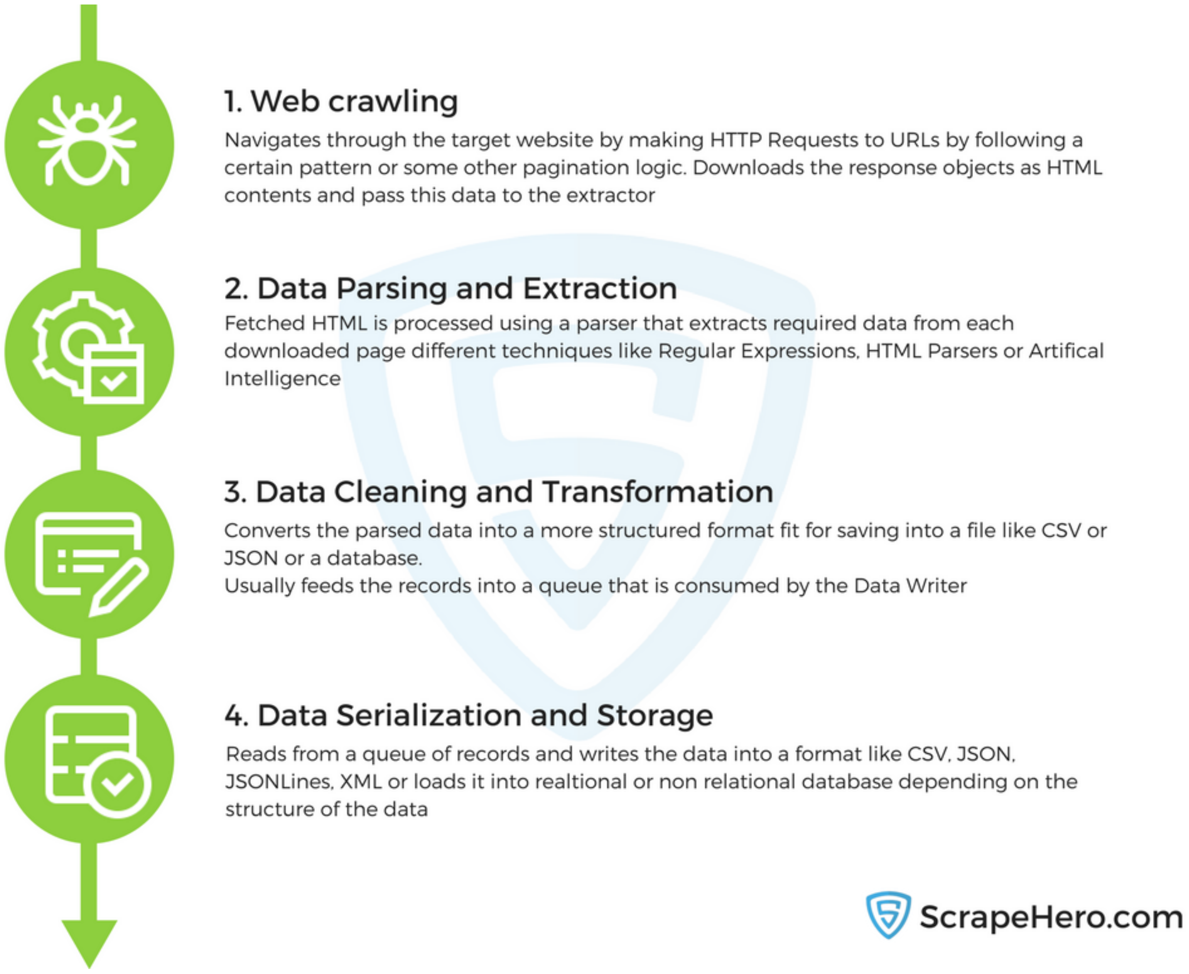
We will show you how to build a very simple web scraper in the next post of this series.

Web scrapers are more complicated than this simplistic representation. They have multiple modules that perform different functions.

What are the components of a web scraper

Web scraping is like any other Extract-Transform-Load (ETL) Process. Web Scrapers crawl websites, extracts data from it, transforms to a usable structured format and load it to a file or database for subsequent use.

A typical web scraper has the following components.



1. Web crawling

Navigates through the target website by making HTTP Requests to URLs by following a certain pattern or some other pagination logic. Downloads the response objects as HTML contents and pass this data to the extractor

2. Data Parsing and Extraction

Fetched HTML is processed using a parser that extracts required data from each downloaded page different techniques like Regular Expressions, HTML Parsers or Artifical Intelligence

3. Data Cleaning and Transformation

Converts the parsed data into a more structured format fit for saving into a file like CSV or JSON or a database.
Usually feeds the records into a queue that is consumed by the Data Writer

4. Data Serialization and Storage

Reads from a queue of records and writes the data into a format like CSV, JSON, JSONLines, XML or loads it into realtional or non relational database depending on the structure of the data

ScrapeHero.com

1. A “focused” web crawler module

A web crawler module navigates the target website by making HTTP or HTTPS Requests to URLs following a specific pattern or some pagination logic. The crawler downloads the response objects as HTML contents and passes this data to the extractor. *e.g the crawler will start at https://scrapehero.com and crawl the site by following links on the home page.*

2. An extractor or a parser module

The fetched HTML is processed using a parser that extracts the required data from the HTML into semi-structured form. There are different kinds of parsing techniques:

1.Regular Expressions – a set of Regular Expressions (RegExes) can be used to perform pattern matching and text processing tasks on the HTML data. This method is useful for simple data extraction tasks such as getting a list of all emails from a web page. But it is not suitable for more complicated extraction jobs – such as getting different fields from a product description page on an

E-commerce website. However, regular expressions are incredibly useful later in the process of data transformation and cleansing.

2.HTML Parsing – is the most commonly used method of parsing data from a web page. Most websites have an underlying database from which it reads content and creates different pages with similar templates. For example – this page you are reading comes from a MySQL Table with fields such as Title, Content, Date, Author, URL, etc. If you visit any other blog post of ours, you can see that it has the same template but different content. HTML Parsers convert HTML into a Tree Like structure that can be navigated programmatically using semi-structured Query Languages such as XPaths or CSS Selectors.

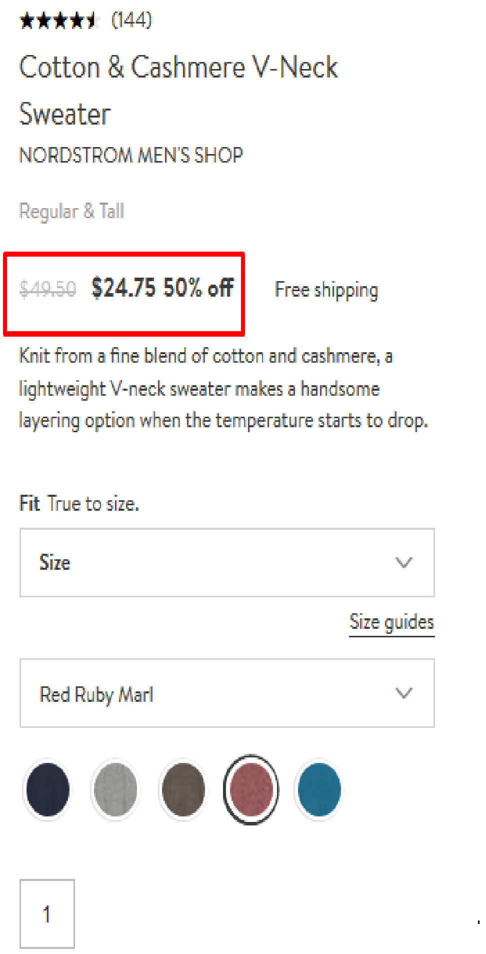
3.DOM Parsing using real or headless Browsers – As the web evolved into complex web applications that depend heavily on javascript, just downloading the web page and HTML wasn't enough. Such pages update data dynamically inside the browser without sending you to another page (using AJAX requests). When you download the HTML of such web pages, you will only get an outer HTML shell of the web app. It will only contain relative links and not much relevant content or data. For such websites, it's easier just to use a full fledged web browser such as Firefox or Chrome. These browsers can be controlled by a browser automation tool such as Selenium or Puppeteer. The data accessed by these browsers can then be queried using Document Object Map (DOM) Selectors such as CSS Selector or Xpaths.

4.Automatic Extraction using Artificial Intelligence – This advanced technique is more complicated and mostly employed when you are scraping multiple websites that fall under a specific vertical. You can train web scrapers using machine learning models to extract data from web pages. You can use Named Entity Recognition models to retrieve data such as contact details from crawled web pages.

3. A data transformation and cleaning module

The data extracted using a parser won't always be in the format that is suitable for immediate use. Most of the extracted datasets need some form of "cleaning" or "transformation." Regular expressions, string manipulation and search methods are used to perform this cleaning and transformation.

Here is an example to illustrate this



★★★★★ (144)

Cotton & Cashmere V-Neck Sweater

NORDSTROM MEN'S SHOP

Regular & Tall

~~$49.50~~ **$24.75 50% off**

Free shipping

Knit from a fine blend of cotton and cashmere, a lightweight V-neck sweater makes a handsome layering option when the temperature starts to drop.

Fit True to size.

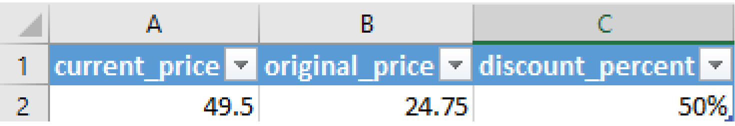
Size

Size guides

Red Ruby Marl

1

When you extract the area marked in red from a page like the one above, it could look like this – "$49.50 **$24.75 50% off**". If you need this data in three separate fields such as current_price, original_price, and discount_percent, you will need to split and clean it using regular expressions or string manipulation before you can get it into a format such as this.



	A	B	C
1	current_price	original_price	discount_percent
2	49.5	24.75	50%

Extraction and transformation are usually performed together in a single module if the scraper isn’t extracting data from a large number of pages.

4. Data Serialization and Storage Module

Once you get the cleaned data, it needs to be serialized the according to the data models that you require. This is the final module that will output data in a standard format that can be stored in Databases (Oracle, SQL Server, MongoDB etc), JSON/CSV files or passed to Data Warehouses for storage.

How to build a web scraper

See the reference link

https://www.scrapehero.com/web-scraping-tutorial-for-beginnerss