UNIVERSITÀ
DEGLI STUDI
DI PALERMO

UNIVERSITÀ
DEGLI STUDI
DI BRESCIA

SCRAPING DATA FROM WEB: EXAMPLES AND APPLICATIONS

Paola Zola

Palermo, May, 20-21, 2019

TODAY CONTENTS:

ADVANCED WEB SCRAPING WITH PYTHON APPLICATION.

- Real time data (Yahoo Finance).
- Text mining I (Wikipedia).
- Text mining II (Tripadvisor, Amazon).
- Text mining III (Facebook).
- Google Trends API.

REAL TIME FINANCIAL DATA

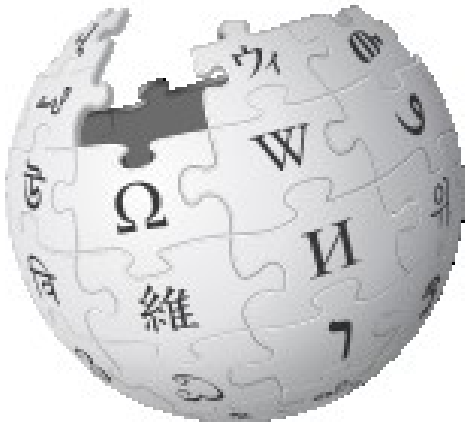
REQUEST

- It is an Apache2 Licensed HTTP library written in Python language and contains functions for requesting data across the web handing also some limitations as cookies or user agent.
- `requests` allow to create a specific `Session` with parameters (as username and password if required by the Web page).
- Another peculiarity of request module is its ability to automatically handle redirects:

```
request.get(url,
allow_redirects=True)
```

LXML

- The lxml XML toolkit is a Python module that extend the original C libraries libxml2 and libxslt.
- `html.fromstring()`: it parse the page requested using the html and save the results in a tree structure. Usually, the input required by the function has to be in bytes, thus the `page.content` is needed.
- `xpath()`: we can obtain the content of the HTML page following in the specified path.



WIKIPEDIA
L'enciclopedia libera

TEXT MINING I

URLLIB

- `Urllib` is a standard Python library and contains functions for requesting data across the web, handling cookies and even changing metadata as the user agent.
- `urrlib.urlopen()`: this function allows to open the connection with the remote object and read it.
- `Urllib` contais also a list of exceptions (as: `HTTPError`) that might be used in a try-except loop.

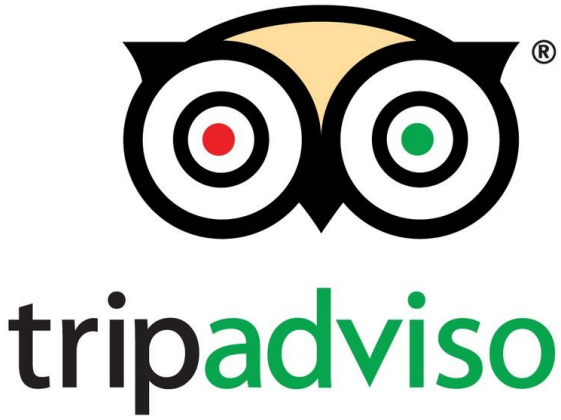
BEAUTIFULSOUP

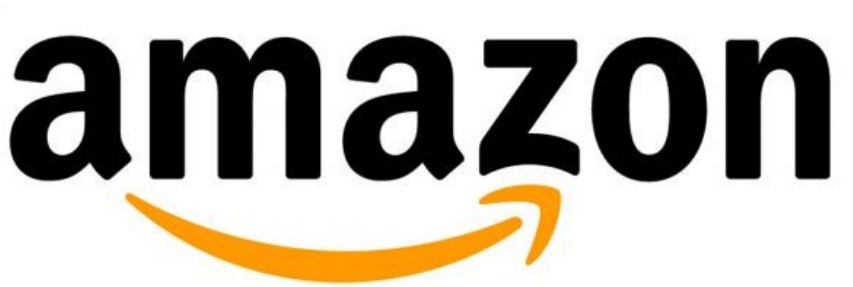
- It is a powerful module helping in organize the messy web pages' format. Similar to the requests module, it download the page content using either an HTML parser and a LXML one that has some likely property in dealing malformed HTML codes.
- `BeautifulSoup(htm, 'html.parser'):` this function allow to get the html source after having calling the `urlopen` function. The parser can be chosen among: `'html.parser'` (does not require extra installation), `'lxml'` (more powerful for messy html codes).

BEAUTIFULSOUP II

- `.find_all(tag,attribute, recursive, text, limit, keywords)`: finds all html tags following in the specified rules.

- `.find()`: this function is very similar to `.find_all()`, they differ just for the limit parameter that can be add into the `.find_all()`.





TEXT MINING II

USER-AGENT

- Agent is a tool that works on behalf of the user and tells the server about which web browser the user is using for visiting the website. Many websites do not let you view the content if the `user-agent` is not set.
- Amazon is one of this web sites, it required an `user-agent` specified.

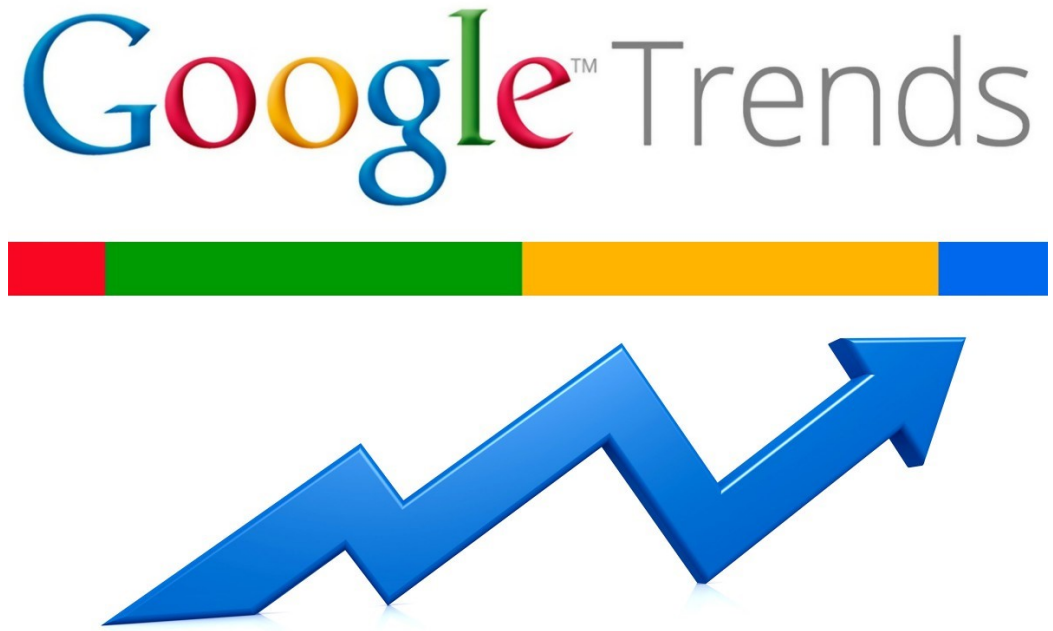
TEXT MINING III

SELENIUM

- Some Web pages, as Facebook are based on a scrolling system, allowing to upload content only scrolling the web page.
- For such kind of pages, we need to use a different approach based on `Selenium` module.
- `Selenium` is a powerful web scraping tool developed originally for website testing. Selenium does not contain its own web browser; it requires integration with thirdparty browsers (e.g., Firefox, Chrome) in order to run. Another powerfull web driver often associated to `Selenium` is PhantomJS.

SELENIUM II

- The `Selenium` library is an API called on the object `WebDriver`. The `WebDriver` is like a browser in that it can load websites, but it can also be to find page elements, interact with elements on the page (send text, click, etc.), and do other actions to drive the web scrapers.
- `Selenium` is based, similar to `BeautifulSoup`, on CSS selector to identify tags in HTML pages. The sintax is based on Xpath.
- The most important ability of `Selenium` that distinguesh it from other libraries as `BeautifulSoup` is its ability to locate buttons tags and select them, moving in different pages or loading hidden contents.



GOOGLE TRENDS API

PYTRENDS

- Documentation of the module at https://pypi.org/project/pytrends/
- The `pytrends` module allows to fetch data related to a specific query. The output includes the historical trend of the searched term, the location with different granularity levels, related queries and related topics.
- Limitation: Google limits the number of queries from a given IP adress. The solution involves the usage of http proxies.

REFERENCES

- Lawson, R. (2015). *Web scraping with Python*. Packt Publishing Ltd.
- Mitchell, R. (2018). *Web Scraping with Python: Collecting More Data from the Modern Web*. " O'Reilly Media, Inc.".
- Richardson, L. (2007). Beautiful soup documentation. *April*.