DEAL Python course III: web scraping

March 28, 2023

Types of websites

- Web 1.0
  - Static (mostly html)
  - Public
  - Mostly no interactive elements other than links
  - E.g., Wikipedia
- Web 2.0 (most of the modern internet)
  - Dynamic (mix of html and javascript, appearance and elements change based on your interaction with the page)
  - Possibly behind a login or paywall
  - Interactive elements (e.g., search bars, forms)
  - E.g., social media sites

Python web scraping tools

- Requests + BeautifulSoup
  - Use these to get and parse the html source code of any webpage
  - Not interactable
  - Suitable for fetching data from Web 1.0 style pages
- Selenium webdriver
  - Use this to remotely control a browser instance
  - You can interact with webpages as you would when browsing manually
  - Suitable for fetching data from Web 2.0 style pages
  - May be combined with Requests and BeautifulSoup (e.g., use Selenium to navigate and BeautifulSoup to collect page text)

Web scraping caveats

- Many sites don’t want you scraping their data, and have safegaurds in place
    - E.g., famously, LinkedIn (https://www.linkedin.com/robots.txt)
    - check the ”/robots.txt” extension of a page to see the site’s policy for bots
- Even sites that are neutral towards bots may block you if you spam their servers with too many requests
    - Important to build in wait time between requests
- Some basic knowledge of html is helpful, but not critical