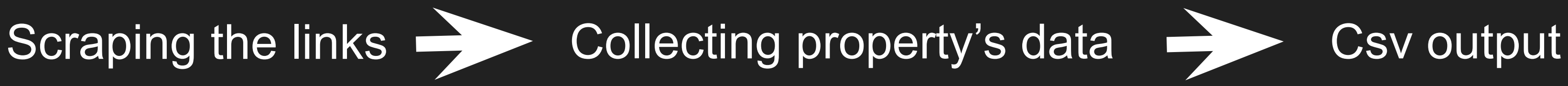
Challenge Collecting Data
-
Web Scraping

Minh Hien Vo, Michel Ombessa, Jose Roldan and Logan Vendrix



Scraping the links → Collecting property’s data → Csv output

How we did it

1. Inspection
2. Using Selenium, BeautifulSoup and threading to get the links of properties
3. Using requests and PyQuery to collect the properties’ data
4. Creating a csv file with the results
5. Issues

Inspection

- Captcha on Zimmo ⇒ too hard to scrape
- Cookie on **Immoweb** ⇒ manageable
- Inspection of the HTML to understand the structure of the site

⇒ Let’s go!

Getting the links

- Selenium to bypass the cookie click button (interaction)
- BeautifulSoup to find the ‘a’ and ‘href’ of all the properties’ links
- “For loop” to go through all the pages
- Thread class to speed up the process of data collection
- Write results to a .txt file (+/- 33.000 links)

Getting the properties’ data

- PyQuery to get the desired features from the HTML
- Check the differences between different types of properties
- Formatting the data (‘Yes’ = 1 , ‘No’ = 2, ...)
- Cleaning the data

Result csv

- “For loop” to go through all the links
- Dict.writer to create a row for each property
- Sample of 5.001 results (not enough time)

Issues

- Who does what?
- Problem with requests so Selenium/BeautifulSoup instead
- Problem with timeout ⇒ time.sleep()
- Inconsistency in HTML structures: table (th, td, th, …)
- Hard to locate all desired features
- Multi-threading not applicable ⇒ time lost
- “None” values ⇒ not showing in csv
- Multi-contribution in git repo