CS 410 Text Transformation Systems – Fall 2021 Final Project

Goodreads Web Scraper on New Releases of the Year

Team – 1 member: Yasaman Sabersheikh

yasaman2@illinois.edu

TABLE OF CONTENTS

Goodreads Scraper

SUMMARY

Goodreads is a free social website on books that allows individuals to freely search for books and reviews. Users can sign up and generate libraries, create reading lists and challenges, and interact with other readers.
The website URL: https://www.goodreads.com/

The amount of information on the website is massive and in result it can be difficult to find information or search easily.
Originally the goal of the project was to find books that are award winners. Goodreads has a link for Reader's Choice awards but other than that it is hard to find other literary awards such as The Man Booker Prize or Giller Prize. I could not find a way to scrape the literature award winners for the time limit of the project.
The project completed by scraping all the monthly new released books for 2 years and provide the 5 highest rating books for that year in each genre category.
A user interface was developed to make the search easy. The information can be used to provide book recommendation or blog posts.

PROJECT GOALS

1. Create a scraper to scrape the data in Goodreads website
2. Gather all the monthly new releases for 2 years
3. Provide the 5 highest rating books for each genre category in each year
4. Develop a user interface to make the information search easier for user

SPECIFICATIONS

- Python 3.8 programming language for scripting in Jupyter Notebook
- Selenium to retrieve the Html source code along with WebDriver-Manager which helps with version numbers

- BeautifulSoup to extract book-data from the Html source code
- HTML and D3.js for creating the user interface. D3.js is a JavaScript library that is widely used for creating and accessing databases on webpages. It treats both data and the webpage as two separate database for data visualization.
- Chrome Browser and Chrome Web Driver Manger
- Github link: https://github.com/yasiss/CourseProject_GoodreadsScraper
- Python file: GoodreadsScraper_NReleases.ipynb
- User Interface page: https://yasiss.github.io/CourseProject_GoodreadsScraper/
- Presentation Link UIUC Box: https://uofi.box.com/s/bs4jceo5wpu01qdb3ssoh3zxfdmia4zd

Implementation Details

The project is a web scraper on Goodreads website. The coding language is python 3.8, and it was implemented using Jupyter Notebook.
Selenium and Beautifulsoup was used to retrieve HTML source and extract the data.
The code gathers all the books for 2020 and 2021 which they are by year, month and genre on website. All the books puts together from all the months for each year and provides the 5 highest rating books for that year in each genre. The books are over 1200.

login() Function

- A login function was created to pass the username (in this website is an email address) and password, automatically when the code starts running. The code navigates to the website and login to it automatically.

Scraping

process_books() Function

- The data for books is gathered in two steps
    1. The code navigates to the New Releases page which it goes to current month new releases by Genre.
    2. The data that we can gather from the first URL is
        - Book URL, genre, title, and book cover image, the year, and the month

3. In the next step code will navigate to each URL and gather additional information such as

   - Author, Rating, Description, Number of pages and the date the book was published

- process_books() function will gather information as mentioned in the first step above, creates a data dictionary, upload the information to the dictionary and creates empty columns for the information that will be gathered in 2nd step of navigation to books URLs.

List, Data Dictionary and Data Frame

- master_list_books is a list created to hold all the book URLs
- Using Selenium and Beautifulsoup we retrieve the data from each URL and populate the empty columns in our Data Dictionary with the information.
- Code will grab all the “new releases” for last 24 months and categorized their information in Dictionary and a panda DataFrame.
- As they are over 1200 books in 2 years, scraping of the data, takes over 1.5h
- DataFrame makes the troubleshooting and visualization of the steps easier as we can see the result in table format

CSV Files

- 2 CSV files were created
  1. NewReleases.csv
  2. TopFivePerGenre.csv
- NewReleases.csv

  The file has all the books information for 1200 books in it. It can be easily used for additional projects on the data in Goodreads website
- TopFivePerGenre.csv

  This file is a subcategory of Meta file and has the information for highest 5 rating books in 12 different genre and in 2 different years: 2020 and 2021. Both CSV were created through python

User Interface with HTML and D3.js

Although we can look at the data in CSV files, it is more user friendly and easier to use, if we could access the data through an interface. On that basis, a user interface was created with HTML and D3.js. D3.js is a JavaScript library that is widely used for creating and accessing databases on webpages. It treats both data and the webpage as two separate database for data visualizations.

- User interface can be accessed through web
- UI has Two dropdown lists one for Year and one for Genre
- Base on the user interest and selection, the dropdown lists update the information on the page in a table, showing five highest rating books in the year and genre that were chosen.

CHALLENGES

- The time was limited and so much can be done, especially as a book lover and a one-person team, it was hard to define and limit myself to the realistic deliverables.
- I had to change the project definition slightly as the data I originally intended to gather, books with literature awards, was not available easily or on all books.
- When I was trying to scrape the html pages, because of the structure of the pages and their html, I could not get the data I needed very straightforward. It took me much longer than what I anticipated.
- I decided to get 2 years instead of 1, and as number of books increased, the scraping time was long, made the troubleshooting time consuming.

CONCLUSION

There are so much can be done by web scraping. Access to data can be much easier and optimized to the need of the user. On top of that if we add Natural Processing Language models, we can create powerful tools to create the future.

The plan is to use this project as a base and add more search and gather useful information on different subjects and create larger packages for future use.