🎥 Hook - Vintage versus Modern: Movie Manner 🎬

Case Study by Catherine Young

Prompt

An inspired UVA student in the School of Data Science wants to collect an organized film archive, Movie Manner. They have taken on the project to classify films as either vintage or modern based on their posters and titles alone. With hundreds of thousands of posters spanning from the 1900s to 2024, Movie Manner wishes to develop an automated classifier that helps the student quickly sort newly added posters into two broad categories (vintage and modern). They hope that training a machine learning model on features like design style, color palette, and title fonts will help distinguish whether a poster has the visual cues typical of older films or contemporary ones. This model would be significant in categorizing movie materials and preserving the film heritage for future generations.

Deliverable

To gather training data, you will need to web scrape IMDB for movie posters and titles from both classic and contemporary film collections. This dataset will allow you to train the model to distinguish visually between old and new, making use of visual trends that have evolved over the years. Trends to analyze include the bright, hand-drawn posters of the 20th century and the digitally polished ones of today. You will be able to edit the provided Jupyter Notebook to help classify the model. Additionally, include a section that explains the edits you made to your neural network model that will classify these posters. What are you justifying as a successful model in this process? Lastly, submit the final dataset from web scraping and the edited Jupyter Notebook.

Case Study Rubric

General Description:

In this assignment you will be able to practice web scraping and neural networks by following the steps provided.

Why am I doing this?

This assignment will allow you to gain practice in areas that are not as common in courseloads. With this, more practice and exposure will allow you to grow.

What am I going to do?

Read the hook outlined above to better grasp the scenario at hand. Beforehand, download the Web Scraper Chrome extension. Next, open two different IMDB websites to scrape the data (one of older, vintage movies and the other of newer, modern movies). Open the web scraping tutorial and follow the instructions to gather all your data. Once you finish gathering the data, use the sample code to build your neural network. Test the model accordingly. Lastly, create a GitHub repository with your dataframe from web scraping, code, and any other materials you used.

Tips for success:

Use your creativity! I wanted to keep the IMDB links broad, so you can pick a specific vintage versus modern era of your choosing. Be comfortable with failing, this is meant to be a learning environment!

How will I know that I have succeeded?

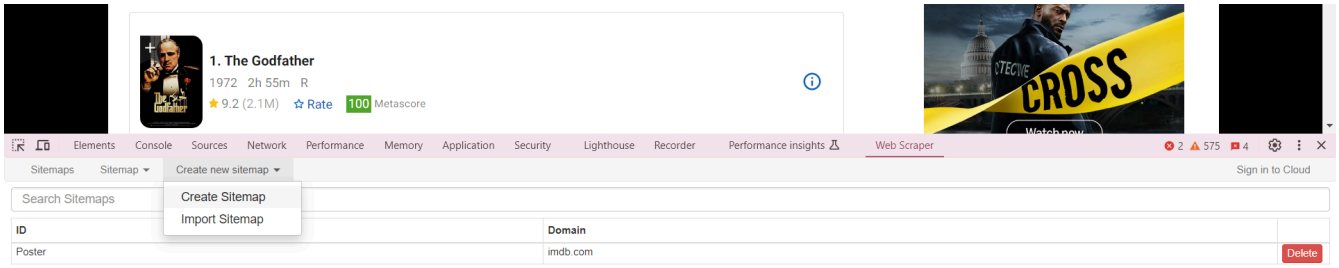
Follow the rubric below:

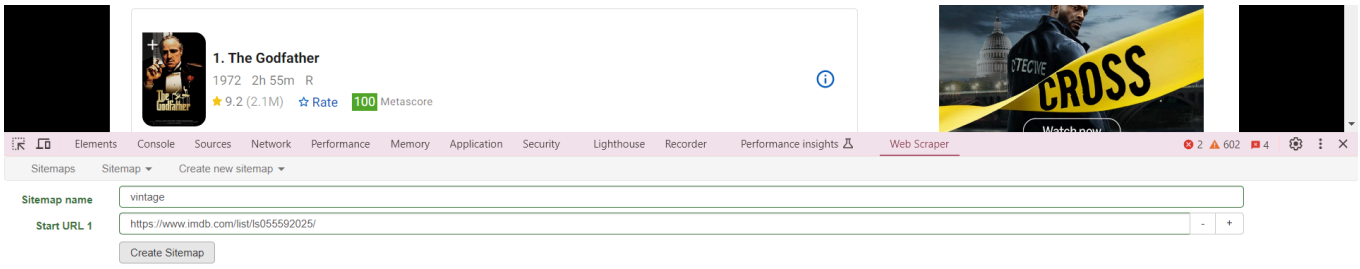
Spec Category	Spec Details
Formatting	A new GitHub repository, including: - README.md - Code file - Images folder - LICENSE
README.md	Goal: this is what will be assessed for this case study - This should be easily interpreted
Source Code	Goal: edit the sample code - Code should have comments to help reproducibility
Images Folder	Goal: include useful images - Any plots made from research
LICENSE	Goal: explain to the users the terms under which they may use/share work - The MIT license is recommended

How to Scrape a Website

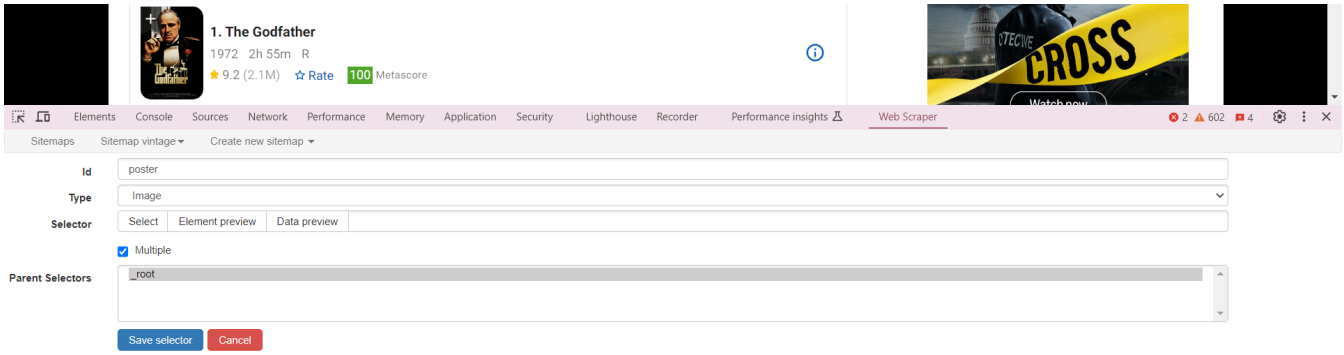
Reproduction Steps

1. Install WebScraper Extension: Add the WebScraper extension to your Chrome browser for easier data extraction. *Note: must use personal email, not edu
2. Choose IMDB Sources: Suggestion: go to IMDB's collections of "Top 100 Classic Movies" and "Top 100 Recent Movies" for diverse visual styles.
3. Inspect Page Elements: Right-click the IMDB page and select "Inspect" to open the developer tools panel and locate the poster and title elements.
4. Create a New Sitemap: Open WebScraper and click "Create Sitemap." Name your sitemap and input the IMDB page URL in the "Start URL" field for each category (classic and recent).

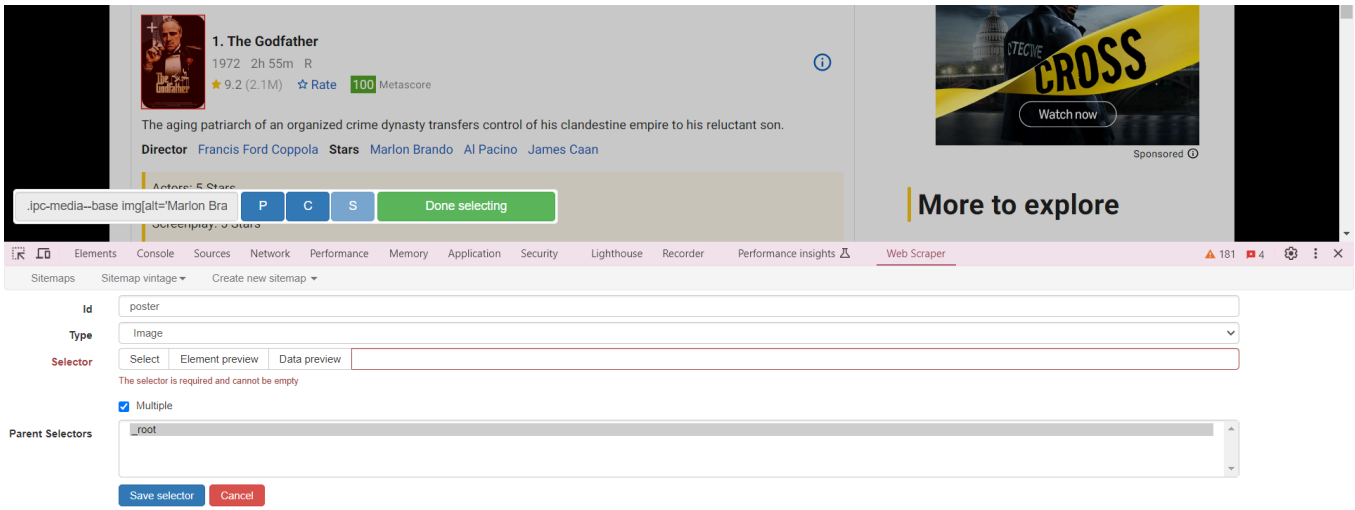




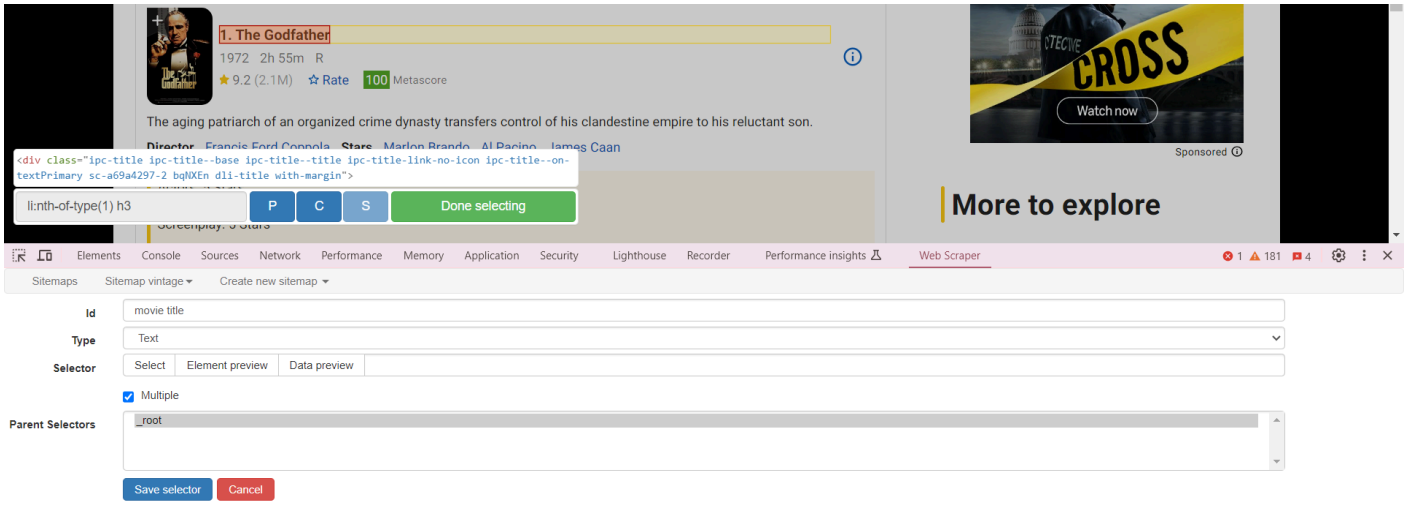
5. Set Up Image Collection: In WebScraper, click "add selector" and name the image field (e.g., "Poster") and choose "Image" as the type. Enable "multiple" selection, then click "Select" to hover over each poster image and save the selector.



6. Set Up Text Collection: Similarly, name the text field (e.g., "Title") and select "Text" as the type. Enable "multiple" and use "Select" to capture each movie title, following the same steps to save the selector.
7. Select Movie Poster Image: Use for images: name the piece of information you are selecting in the ‘id’ spot, and select ‘Image’ where you choose the Type. Then, click the check next to multiple because we are scraping every image. Now you are able to start selecting by clicking the ‘Select’ button that is above the image check. Toggle your mouse over the movie poster image, highlighting it in red. Hold the shift key and click to select the poster and then repeat for the next image to select all the images on that page. Click the green ‘Done Selecting’ button space and then the ‘Save selector’ to see the data you just created.



8. Select Movie Text (Title): Use for text: name the piece of information you are selecting in the ‘id’ spot and select ‘Text’ where you choose the Type. Then, click the check next to multiple since we need every title from the page. Next, click the ‘Select’ button above the image check. Again, toggle your mouse over where the movie title appears, highlighting it in red. Repeat the same steps highlighted in step 7 from holding the shift key to clicking ‘Save selector’.



9. Preview and Export Data: After previewing the data by clicking "Data preview," copy it into an Excel file. Add a column labeled "Type" and assign either “Vintage” or “Modern” to each movie based on its category, then export as a .csv for future model training.

You are finished! :)