

Final Project Write-up

Part 1: Website Description

The **purpose** of this website is to provide a data visualization that compares movie characteristics between four streaming services: Netflix, Amazon Prime, Hulu, and Disney+. The **information conveyed** is obtained from a dataset found on Kaggle.com. My **target audience** is showrunners/directors and executives who are interested in the catalog details of streaming services. My website presents a scatterplot visualization on the IMDb scores, Rotten Tomatoes scores, target age groups, and runtimes of each platform's movie catalog. Before starting the website development, I first needed to perform preprocessing on the dataset. I conducted these operations using Python and the Pandas library. Once the dataset was cleaned, I was able to utilize the revised CSV on my website. It is important to note that this dataset is heavily skewed towards Netflix.

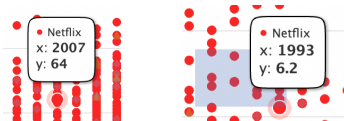
Interesting elements of this website include the interactions users can perform with the data visualization. Utilizing the High chart API, I was able to construct an interactive visualization that allows for an engaging presentation of the dataset. Notable features include:

- Animation of data points on scatterplot visualization
- Accessibility features that support screen readers (ChromeVox)
- Sonification of data points when clicked
- Export file options for the visualization

Part 2: User Interactions

The core experience and interactions of the website revolve around the scatterplot visualization. There are many ways the user can get both a high-level and detailed view of this streaming service dataset.

- (1) Using the **navigation bar**, the user can switch between different feature visualizations. For example, if they click "IMDb", they will be able to view the visualization for the IMDb score of streaming service movies over time.
- (2) When the user hovers over a point on the scatterplot, the user gets more details on the highlighted movie.
- (3) By dragging diagonally on the visualization, the user can zoom in to get a closer look at the points in that area.
- (4) At the bottom of the visualization, the legend also serves as a toggle. By clicking on any of the categories the user can select or deselect the categories which would affect its appearance on the visualization. Also, when the user hovers over a category, the associated points will be highlighted in the visualization.



● Netflix ◆ Amazon Prime ■ Disney+ ▲ Hulu

Part 3: External Tools

(1) Highchart API

- I chose this API because it contained built-in accessibility features and allows for quick data visualization through various interactive graphs and charts.
- The API provided documentation on specific ways to format JavaScript code. I primarily refer to the API documentation for formatting or syntax requirements.
- This API added advanced interactivity to the data visualization on my website.

(2) Bootstrap

- I chose to use the JavaScript library because it allowed for the easy construction of a responsive website.
- I primarily used this library to quickly layout my website.
- This library added responsiveness and a concise layout. This allows both the data visualization and the HTML elements to be formatted well in both the desktop and mobile view.

(3) Highchart API Accessibility:

<https://www.highcharts.com/docs/accessibility/accessibility-module-feature-overview>

(4) Streaming Movies Dataset: <https://www.kaggle.com/ruchi798/movies-on-netflix-prime-video-hulu-and-disney>

Part 4: Mockup Iterations

In my initial prototype, I considered using a synchronized line graph from the High chart API. At the time, I believed it would be a good way to simultaneously show all the information from the dataset in one place. However, once I analyzed the data further, I found that it was not well suited for this type of visualization. This caused me to iterate on my prototype and take a different design approach. The main difference was the switch from a synchronized line graph to an interactive scatterplot.

Part 5: Implementation Challenges

I did have some implementation challenges in mapping other elements of the dataset to the data visualization. This is my first time using the High chart API, and I was not able to find any documentation on applying multiple values to one data point. I also struggled with properly adding the sonification functionality.

Github Link:

https://github.com/suber-mhci/pui_final

Website Link:

https://suber-mhci.github.io/pui_final/