MOVIE MANIA

Anushka Singhal (23B0928)

April 2024

Contents

1 Introduction **2**

2 Project Setup **2**
2.1 Submission files 2
2.2 Working of website 2

3 Basic Tasks **3**
3.1 Curation of movie database 3
3.1.1 Scrapping 3
3.1.2 Converting into JSON 4
3.2 Data Structuring and Conversion to JSON 4
3.3 Website Interface 5
3.3.1 Search Bar 5
3.4 Recommender System 5
3.4.1 Algorithm and its demonstration 6
3.4.2 Genre Preferences 6
3.4.3 Release Year Influence 6
3.4.4 Time Duration Consideration 7

4 Customisations **7**
4.1 Enhanced UI 7
4.2 Suggestion box 8
4.3 User Registration 8
4.4 Filter 9
4.5 Sort 10

1 Introduction

The Movie Repository Website is a one-stop platform for movie enthusiasts, providing a curated database sourced exclusively from IMDB. Users can easily search for movies and access detailed information. The website features a sophisticated movie recommendation system, user registration, rating meter, filtering options, and sorting mechanisms for an enhanced user experience.

2 Project Setup

2.1 Submission files

The following files are part of the project setup:

- `scraping.py` - Python script used for web scraping.
- `data.html` - HTML file containing curated movie data.
- `movie_details.html` - HTML file with additional modifications to curated movie details.
- `site.html` - Main HTML file for the website interface.
- `login.html` - HTML file for the login page.
- `register.html` - HTML file for user registration.
- `recommended.html` - HTML file for displaying recommended movies.
- `server.js` - JavaScript file for implementing the Node.js server.
- `script.js` - JavaScript file for client-side scripting.
- `styles.css` - CSS file for styling the website.
- `final_movie_details.json` - JSON file containing curated movie details in JSON format.
- `users.json` - JSON file for storing user information.

2.2 Working of website

Before using the website, follow these steps to set up the environment:

1. Extract all files from the provided zip folder. Ensure that all files are in the same directory.
2. For web scraping (if required), ensure the following modules are installed on your system:
    - `requests` - for making HTTP requests

- `json` - for handling JSON data
- `bs4` (Beautiful Soup) - for parsing HTML content
- `fake_userAgent` - for generating fake user agents
- `html5lib` - for parsing HTML documents

3. For starting the server, ensure the following Node.js modules are installed:
   - `http` - for creating HTTP servers
   - `path` - for working with file and directory paths
   - `fs` (File System) - for file operations

Once the setup is complete, follow these steps to start the server and use the website:

1. Open a terminal or command prompt.
2. Navigate to the directory where the files are located.
3. Start the server by running the command:

```
node server.js
```

4. Once the server is running, you can access the website by opening a web browser and entering the URL provided by the server (e.g., `http://localhost:3000`).

3 Basic Tasks

3.1 Curation of movie database

3.1.1 Scrapping

The process of curating data involved several steps and challenges, which are outlined below:

1. **Initial Web Scraping Issues:** I initially attempted web scraping using Beautiful Soup without employing fake user agents. Unfortunately, this approach faced obstacles as IMDB blocked web scrapers, leading to either erroneous or empty data.
2. **Implementing Fake User Agents:** To tackle this challenge, I delved into implementing fake user agents. I referred to resources like reference 1 to gain insights and successfully integrate fake user agents into the scraping process.
3. **Encoding Errors Resolution:** Despite incorporating fake user agents, encoding errors persisted during scraping. To resolve this issue, I explicitly set the encoding to utf-8, ensuring proper handling of character encodings and eliminating encoding-related issues.

4. **Successful Completion of Scraping:** With these adjustments in place, I successfully completed the scraping process, resulting in curated data that was now ready for utilization in the website.

5. In this I extracted first from the website "https://www.imdb.com/chart/top/" then from the website "https://www.imdb.com/list/ls094833375/?sort=user_rating,desc&st_dt=&mode= because the first one not containing the detailed information.

3.1.2 Converting into JSON

3.2 Data Structuring and Conversion to JSON

After gathering data through web scraping, my objective was to organize it into a format suitable for storage and manipulation. JSON (JavaScript Object Notation) was chosen due to its simplicity and compatibility with various programming languages.

The initial step involved parsing the scraped HTML files to extract relevant information about each movie, such as title, year, rating, age rating, duration, genre, summary, director, stars, and gross earnings. However, I encountered two significant challenges during this process:

1. **Missing Detailed Information:** The HTML structure of IMDb's pages did not directly provide detailed information like director, cast, and other crucial details. This was attributed to the information being dynamically loaded or accessible only through interactive elements like the "info" button on the IMDb website.

   To address this challenge, I adopted an alternative approach by exploring different versions of IMDb movie pages where such detailed data was readily available without requiring additional interactions.

2. **Age Rating Discrepancies:** Some movies in the scraped data had age ratings listed as "G" or marked as "not rated," leading to inconsistencies in the dataset. The actual age rating span (¡span¿) was missing in these cases.

   To ensure consistency and accuracy, I manually added the missing age rating spans for the affected movies, correcting the data for further processing.

   To integrate the refined data into the dataset, I created a new HTML file named `movie_details.html`. This file contained the corrected and complete information, including the added age rating spans and any other missing details.

With the comprehensive and refined dataset now available, the final step was to convert it into JSON format. This conversion process transformed the structured data into a JSON file, making it ready for storage and utilization within the website's backend systems.

3.3 Website Interface

The website interface provides a platform for interacting with the movie repository. It displays a list of all available movies from the repository. It has a search bar on the top right for searching a particular movie. On the top left, it has many features including filter, sort, register and login, and personalised movie recommendations on the basis of ratings given by the user. These features will be explained in detail later in the report.

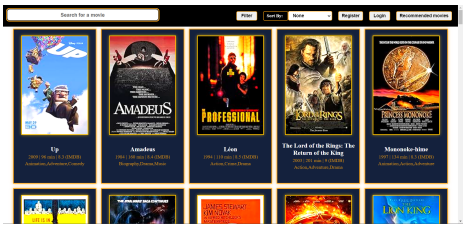


Figure 1: Website interface

3.3.1 Search Bar

Users can easily search for movies by typing in the search box located prominently on the interface. The search function is dynamic, providing suggestions as the user types and showing the full details of the movie searched.

3.4 Recommender System

The website incorporates a personalized recommender system designed to enhance the user experience by providing tailored movie recommendations based on user ratings. The recommender system operates as follows:

1. **User Ratings:** Users have the option to rate movies based on their preferences and experiences. These ratings serve as valuable feedback that helps the recommender system understand each user's movie preferences and tastes.

2. **Recommender Button:** After users have rated movies according to their liking, they can click on the "Recommender" button to receive personalized movie recommendations. This button triggers the recommender system to analyze the user's ratings and generate a list of recommended movies tailored to their preferences.

3. **Algorithmic Analysis:** The recommender system employs algorithms to analyze user ratings and identify patterns in movie preferences. By considering factors such as genre preferences, ratings given by the user, and possibly other metadata, the system generates intelligent recommendations that align closely with the user's movie-watching preferences.

4. **Personalized Recommendations:** The recommendations provided by the system are personalized and curated specifically for each user. Instead of generic suggestions, users receive movie recommendations that reflect their unique tastes and interests, enhancing their overall movie discovery and enjoyment.
5. **Enhanced User Engagement:** The recommender system enhances user engagement by offering relevant and tailored recommendations. This personalized approach encourages users to explore new movies, discover hidden gems, and enjoy a curated movie-watching experience that aligns with their individual preferences.

3.4.1 Algorithm and its demonstration

he recommendation algorithm takes into account various factors to personalize movie suggestions based on user preferences.

3.4.2 Genre Preferences

- Each genre's importance is determined by calculating a weighted average using IMDb ratings of movies within that genre.
- The weighted average reflects the user's interest in different genres, with higher-rated genres receiving more weight in recommendations.
- The algorithm adjusts genre weights by subtracting a factor proportional to the inverse of the number of movies in that particular genre. This adjustment aims to refine genre importance based on user preferences.

  For example, suppose a user rates five movies, with one movie being a romance genre rated highly at 10. However, this movie also includes drama as a genre. The remaining movies, all having atleast one genre but not the only genre as drama, are rated around 6 or 7.

  Without adjustment, the algorithm might give equal weight to drama and romance due to the increased number of drama genre movies, affecting the recommendations' accuracy. By subtracting a factor inversely proportional to the number of movies in each genre, the algorithm ensures that genres with fewer movies but higher user ratings receive appropriate weightage.

  This adjustment optimizes genre relevance in recommendations, enhancing the algorithm's ability to suggest movies aligned with user preferences.

3.4.3 Release Year Influence

- The average release year of user-rated movies is considered to capture temporal preferences.
- This factor is weighted based on IMDb ratings to ensure its significant contribution to the recommendation process.

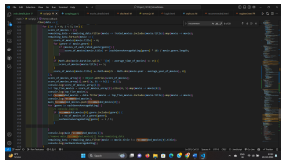
3.4.4 Time Duration Consideration

- The average time duration of user-rated movies is factored in, albeit with less weight compared to genres and release years.
- This factor acknowledges the impact of time duration on viewing experiences but emphasizes genres and release years in shaping recommendations.

The algorithm prioritizes genre preferences as the primary driver of recommendations, supplemented by considerations for release years and time durations, resulting in personalized and comprehensive movie suggestions. This algorithm might have some drawbacks or some limitations in some cases, but it will work fine for most of the cases.

(a) Part 1

(b) Part 2



(c) Part 3

Figure 2: Code for algorithm

4 Customisations

4.1 Enhanced UI

The website interface is designed to enhance user interaction and facilitate easy access to movie information. It includes the following user-friendly features:

- **Search Bar:** Users can quickly find movies by typing in the search box. The search function provides dynamic suggestions as the user types, making it convenient to locate specific movies.
- **Filter and Sort Options:** Users can filter movies based on criteria such as genre, year of release, or rating. They can also sort movies according to various parameters like title, rating, or year of release. These options streamline the movie selection process.
- **Hover for Scaling Out:** When users hover their cursor over a movie thumbnail or title, the interface automatically scales out the basic information for that movie. This feature enhances user experience by providing a clearer view of the movie's details without requiring additional clicks or actions. By incorporating this feature, the website becomes more interactive and user-friendly, enabling users to explore movie details effortlessly while browsing through the movie repository.

- **Navigation Menu:** The interface also includes a navigation menu with options such as register, login, and recommended movies. This menu enhances user experience by providing additional functionalities and personalized recommendations.

  Some elements are also made responsive.

4.2 Suggestion box

The website incorporates a dynamic suggestion box that enhances the user's search experience. When a user begins typing in the search bar to look for a movie, the suggestion box instantly appears, providing a list of movie titles or keywords that match the user's input. This feature offers several advantages:

1. **Fast Access to Movie Details:** The suggestion box provides users with quick and easy access to movie information. By simply clicking on a suggestion from the box, users can view the full details of the selected movie without the need for manual searching or navigating through multiple pages.

2. **Efficient Selection Process:** If a user presses the "Enter" key anytime while typing in the search bar, the suggestion box automatically selects and displays the movie shown at the top of the list. This streamlined selection process saves users time and effort by eliminating extra clicks or navigation steps.

3. **User-Friendly Interface:** The suggestion box is designed to be user-friendly and intuitive, making it easier for users to find and explore movies of interest. Its responsive nature ensures that users can interact with it seamlessly across different devices and screen sizes.

By integrating the suggestion box enhancement, the website provides a more efficient and engaging search experience, allowing users to discover and access movie details with speed and convenience. Adjustments can be made to the suggestion box's appearance, behavior, and content based on user feedback and evolving usability requirements.

4.3 User Registration

The website offers a user-friendly registration process for new users to create accounts and access personalized features. Here's how the registration and login process works:

1. **User Registration:** New users can navigate to the registration page from the registration button. On the registration page, users are prompted to enter their desired username, name, email address, and password in a fixed regex. The system validates the entered data to ensure that all required fields are filled and that the chosen username is unique. Upon successful

registration, users receive an alert message indicating that their account has been created.

2. **Login Authentication:** Registered users can log into their accounts using their username and password on the login page. The system authenticates the entered credentials against the database to grant access to authenticated users. If the login credentials are correct, users are redirected to the home page. In case of incorrect credentials, the system displays error messages.

4.4 Filter

The website incorporates a comprehensive genre-based filtering system to enhance user experience and facilitate precise movie searches based on genre preferences. This feature provides users with the flexibility to filter movies based on single or multiple genres, as follows:

1. **Single Genre Selection:** Users can navigate through the available genres and select a single genre from the provided options. Upon selecting a genre, the system dynamically filters the movie database to display all movies that belong to the chosen genre. This functionality enables users to explore a curated list of movies within their preferred genre, streamlining the movie discovery process and catering to specific genre preferences.

2. **Multiple Genre Selection:** For users with diverse genre preferences or specific genre combinations in mind, the website offers the capability to select multiple genres simultaneously. Users can check boxes corresponding to their desired genres, and the system intelligently filters the movie database to present movies that encompass all of the selected genres. This advanced filtering mechanism ensures that only movies containing all of the selected genres are displayed, providing users with tailored recommendations and refining the search results to align with their nuanced genre preferences.

3. **Intersection of Genres:** When multiple genres are selected, the filter operates by finding the intersection of the selected genres. This means that only movies that fall into all of the selected genres are included in the filtered results. By leveraging this intersection approach, the filter delivers highly relevant movie suggestions that align closely with the user's specific genre criteria, enhancing the overall browsing experience and facilitating efficient movie discovery.

4. **User-Friendly Interface:** The genre-based filtering feature is integrated into the website's intuitive user interface, ensuring ease of use and seamless navigation for users. Clear categorization of genres and interactive checkboxes make it straightforward for users to select their preferred genres and access tailored movie recommendations effortlessly.

4.5 Sort

The sorting feature on the website is designed to provide users with comprehensive options for organizing and exploring movies based on different criteria. Let's delve into each sorting criterion in detail:

1. **Alphabetical Order:** The alphabetical sorting option allows users to arrange movies in ascending or descending order based on their titles. This sorting criterion simplifies navigation by presenting movies in a structured and easy-to-follow manner, enabling users to quickly locate specific titles or genres of interest. Whether users are looking for a particular movie or prefer browsing alphabetically, this sorting functionality enhances the accessibility and organization of the movie database.

2. **IMDb Rating:** Sorting movies by IMDb rating enables users to discover top-rated movies and prioritize their movie selections based on quality and popularity. This sorting criterion arranges movies according to their IMDb ratings, with highly rated movies appearing at the top of the list. By sorting movies based on IMDb ratings, users can explore critically acclaimed titles, discover hidden gems, and make informed decisions about which movies to watch, enhancing their overall movie-watching experience.

3. **Year:** Sorting movies by year offers users a chronological perspective on movie releases, allowing them to explore movies based on their release dates, the recently released will be showed on top. This sorting option is particularly beneficial for users interested in specific time periods, genres, or historical contexts, providing a curated list of movies that align with their preferences and interests.

The sorting functionality adds depth and customization to the user experience by offering diverse sorting criteria tailored to users' browsing preferences. Whether users prioritize alphabetical organization, seek top-rated movies, or wish to explore movies by release years, the sorting feature enhances navigation, discovery, and engagement with the vast movie collection available on the website.

References

[1] GeeksforGeeks. Extract json from html using beautiful soup in python. 2022.

[2] GeeksforGeeks. Implementing web scraping in python using beautiful soup. 2022.

[3] W3Schools. Javascript tutorial - w3schools, 2022.