Trending Now: Leveraging Social Media to Identify and Analyze Movie and Genre Trends

Dimple Singh
Computer Science
Binghamton University
Binghamton, NY, USA
dsingh27@binghamton.edu

Jay Balaram Sankhe
Computer Science
Binghamton University
Binghamton, NY, USA
jsankhe1@binghamton.edu

Debangana Ghosh
Computer Science
Binghamton University
Binghamton, NY, USA
dghosh2@binghamton.edu

Jeremy Anton
Computer Science
Binghamton University
Binghamton, NY, USA
ajeremy1@binghamton.edu

Ritika Kale
Computer Science
Binghamton University
Binghamton, NY, USA
rkale2@binghamton.edu

Abstract

The core objective of our study is to gain insights into the ever-changing landscape of movie trends.We want to identify and understand what makes a movie popular at any given time. To achieve this, we focus on several key aspects such as what topics are currently trending, how audiences rate these movies, and the discussions and interactions happening on platforms like TMDb and Reddit. Our ultimate goal is to uncover the key factors that contribute to a movie's popularity. Our research findings are expected to reveal emerging patterns within the movie industry. They will also provide valuable insights into what audiences like and dislike, and how social media plays a crucial role in determining a movie's success in today's world. This exploration offers valuable insights for professionals in the film industry, filmmakers, and movie enthusiasts who want to better understand and navigate the ever-changing landscape of movie popularity trends. The Reddit and TMDB Data Crawler is a Python script developed for the purpose of collecting data from two prominent online platforms: Reddit and The Movie Database (TMDB). This script serves as a tool to extract vital information related to movies and television shows, including their levels of popularity, and relevant posts and discussions on Reddit. All the collected data is then stored neatly in a PostgreSQL database for later analysis.

Keywords: Reddit API, TMDb API, PostgreSQL, Movies, Data Collection, Analysis, Plotting, Visualization

ACM Reference Format:
Dimple Singh, Jay Balaram Sankhe, Debangana Ghosh, Jeremy Anton, and Ritika Kale. 2023. Trending Now: Leveraging Social Media to Identify and Analyze Movie and Genre Trends. In *Proceedings of ACM Binghamton conference (Binghamton'23)*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145.1234567890

1 Introduction

This report provides an overview of the implementation of our data collection system, focusing on the extraction of data from Reddit and The Movie Database (TMDB). We present the progress made since the project proposal, highlight any challenges encountered, and provide insights into the collected data. Additionally, we include a plot illustrating the data collection process over time and updated projections for the amount of data to be collected. In today's ever-changing digital landscape, the world of movies exerts a profound influence. The film industry is undergoing remarkable transformations, spanning various platforms, genres, and international audiences. Our research centers on the intersection of movies and social media. By exploring platforms such as TMDb and Reddit, we aim to unveil how the movie community molds trends and dialogues. By examining TMDb ratings, user-generated content, and community discussions on Reddit, we aspire to explore the intricate relationship between movie popularity, viewership trends, and audience sentiment.

2 Data Source

Here, in this project, we will collect information about top-rated movies. We will be using TMDb rating data that we obtain from the TMDb API and Reddit data from r/movies and r/tv through Reddit Stream API. For implementation, we will use Python, and some of its libraries to fetch and handle the Requests and store the data collected in PostgreSQL.

Binghamton'23, Binghamton, NY, USA,

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/10.1145.1234567890

2.1 TMDb API

TMDb API offers several endpoints that provide access to movie-related data, facilitating real-time and historical queries of TMDb ratings, audience scores, and reviews. Information about movie and TV titles, including their popularity. The TMDb API provides various access points to access data related to movies. This enables us to make both real-time and historical queries for information such as TMDb ratings, audience scores, and reviews, making it a valuable resource for movie-related data.

2.2 Reddit API

User-generated content and discussions related to movie and TV titles. The Reddit REST API allows us to access user-submitted and rated content. With the use of this API, we have the capability to gather valuable audience reviews from Reddit for the latest trending movies. This allows us to gain insights into the level of interest and engagement that these films generate among Reddit users. Moreover, this API offers advanced features, such as the ability to access user account information and monitor the number of upvotes on comments. These additional functionalities enhance our ability to thoroughly analyze the sentiments and preferences expressed by the Reddit community when it comes to these movies. In summary, this API is a powerful tool that enables us to tap into the collective opinions and reactions of Reddit users, providing a deeper understanding of how these movies are received and appreciated within the online community.

3 API Methods

We will utilize standard HTTPS requests to interact with an API endpoint. This Web API employs common HTTP methods, including GET, POST, and PUT to access and manipulate data resources. The API returns all response data in the form of a JSON object. Both the TMDb and Reddit APIs can search, fetch, engage with, or create various resources. To retrieve data from the TMDb and Reddit APIs, we primarily employ the GET method.

- Reddit API Authentication: The get_reddit_token method handles authentication, obtaining a token for our application to access the Reddit API securely.
- Data Collection from Reddit: get_reddit_posts _paginated_per_media method is responsible for retrieving Reddit posts based on media titles, handling pagination and search criteria.
- Data Filtering and Cleaning: The filter_posts function selects relevant posts, while cleanup_posts organizes post data for storage.
- Data Storage: insert_data_into_postgresql connects to a PostgreSQL database, creates tables, and inserts various data types, including titles, scores, and links.

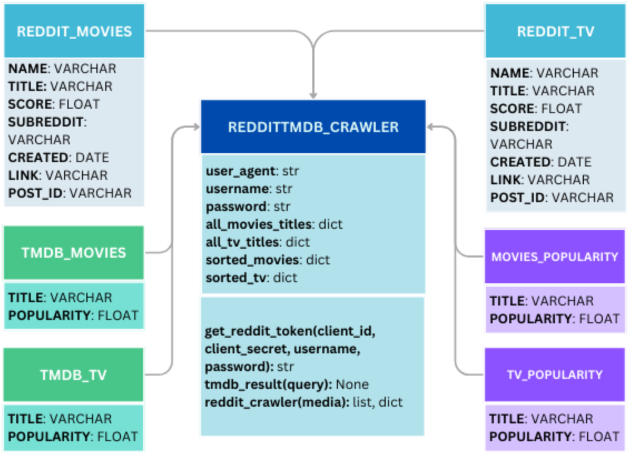


Figure 1. UML Diagram

- TMDB API Data Retrieval: tmdb_result sends requests to the TMDb API to retrieve movie and TV titles, along with their popularity, storing the data in text files.
- Overall Project Flow: The project follows a structured flow, encompassing data collection, filtering, cleaning, and storage, utilizing sorted data for analysis.
- Rate Limiting and Progress Reporting: The project manages rate limits to ensure uninterrupted data collection and employs progress reporting techniques.
- Exception Handling: Exception handling mechanisms are in place for addressing errors during API requests and database operations.

4 Measurements and Analysis

Our research aims to conduct a robust analysis of movie data, focusing on the popularity of movies and the prevailing genre trends. This analysis will primarily source data from the TMDb and Reddit platforms.

- Data Acquisition: Utilizing the dedicated APIs provided by TMDb and Reddit, we will systematically extract pertinent movie-related data. As part of our real-time data integration approach, every new piece of data will be promptly updated in our database.
- Data Visualization: Periodic retrieval of data from our database will be executed, enabling us to synthesize and visualize the information. Employing plotting and visualization libraries in Python, we will generate graphical representations, such as bar charts or heatmaps, to depict movie trends and genre preferences.
- Sentiment Analysis: An integral component of our research will involve sentiment analysis. By evaluating the positive or negative reactions of users to trending movies, we aim to gain a holistic understanding of audience reception and sentiment towards specific movies or genres.

5 Challenges

During the implementation, we encountered several challenges:

- Reddit API Authentication: Ensuring that Reddit API authentication was correctly set up and obtaining the necessary access token posed initial challenges. The script needed adjustments to handle authentication issues effectively.
- Data Volume: The volume of data collected was larger than anticipated. This required fine-tuning the data storage and processing procedures to prevent exceeding storage limits.
- Data Cleansing: Cleaning Reddit posts to extract relevant information while handling variations in data structure posed challenges. Customized data processing was required for different types of posts.

6 Data Collection Over Time

The following plot indicates the progress of data collection over time. The x-axis represents the time in days, and the y-axis shows the cumulative data collected in terms of the number of titles.

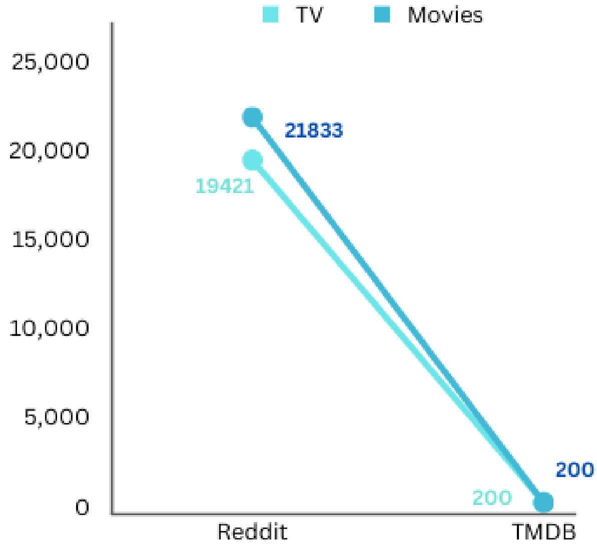


Figure 2. Plot of data collected from Reddit and TMDB

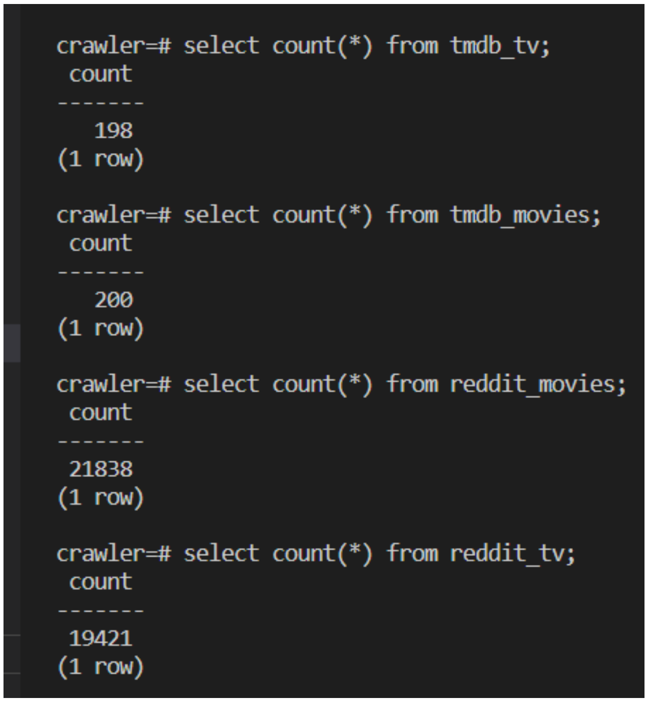
As the plot demonstrates, data collection increased steadily over time, showing that the system efficiently collected data from both Reddit and TMDB.

7 Updated Projections

Based on the current data collection rate and available storage, we project the following: By the end of the project timeline, with the assumption of collecting 20 titles per page and using 3 pages for TMDb, we can expect to have collected approximately 3 * 20 = 60 movie and 60 TV titles from TMDb. For Reddit, with the assumption of 20 titles per page and 5 pages, we would collect 20 * 5 = 100 Reddit posts every hour. Given that we run this every hour and extrapolated from the code to estimate that you collect 193.65 for 20 shows and 20 movies, we can calculate the number of posts collected in a day as follows: 193.65 * 120 = 23238 i.e approximately 25000 These projections provide an estimate of the data volume and will guide our data management strategy to prevent storage limitations.

8 Results

Following figure represent output for the data collected till now.



```
crawler=# select count(*) from tmdb_tv;
 count
-------
   198
(1 row)

crawler=# select count(*) from tmdb_movies;
 count
-------
   200
(1 row)

crawler=# select count(*) from reddit_movies;
 count
-------
 21838
(1 row)

crawler=# select count(*) from reddit_tv;
 count
-------
 19421
(1 row)
```

Figure 3. Ouptut

The database query results indicate the following counts: 198 records in the tmdb_tv table, an error in querying the tmdb_reddit table (possibly due to a nonexistent table), 200 records in the tmdb_movies table, 19,421 records in the reddit_tv table, and 21,833 records in the reddit_movies table. These figures illustrate the progress of data collection over time, revealing substantial information from both Reddit and TMDB sources.

9 Conclusion

The Reddit and TMDB Data Collection system has made substantial progress since the project proposal. We have successfully retrieved data from both sources, overcome authentication challenges, and handled a significant amount of data. The updated projections help us manage data collection efficiently, ensuring we stay within storage limits.

10 Acknowledgements

We would like to acknowledge the Reddit and TMDB APIs for providing access to valuable data. We also appreciate the guidance and support received during the project's development.

References

[1] Meet Singh, Dhrubasish Sarkar. *Content-Based Movie Recommendation System with Sentiment Evaluation of Viewer's Reviews.* https://link.springer.com/chapter/10.1007/978-981-16-6893-7_15

[2] Sandipan Sahu, Raghvendra Kumar, Mohd Shafi Pathan, Jana Shafi, Yogesh Kumar, Muhammad Fazal Ijaz. *Movie Popularity and Target Audience Prediction Using the Content-Based Recommender System.* https://ieeexplore.ieee.org/abstract/document/9758691