**Movie Recommendation and Rating Prediction System**

**Group-3 Members**

Anusha Chilakamarri-11664602

Ram Srinivas Katragadda-11664589

Poojitha Pothini-11664530

Bhuvaneswar Reddy Sriyyapu Reddy-11659017

**Goals and Objectives**

**Motivation**

The main goal of this project is to improve user experience by developing a recommendation system which suggests the movies.

For developing this system, we are using different recommendation methods.

Such as like demographic, content-based, and collaborative filtering.

Ans also to display the importance of recommendation systems for giving a customized idea.

**Significance**

While developing a movie recommendation system we are using a large dataset of movie features including revenue, cast, crew, and budget is crucial. With this dataset diversity, the recommendation system can be fully evaluated and trained to understand different characteristics of each movie.

Firstly, it efficiently arranges categorical data for movie attribute identification and analysis. It uses JSON data processing and feature engineering to extract significant information from cast and crew columns, giving deeper understanding and better suggestions. The dataset also consist of information like economic factors which are budget and income. This addition gives us a lot more information about what people want and how the movie business works. By learning about these financial factors, the computer can better understand movie tastes and market trends, which leads to more accurate and targeted movie choices.

A complete system for suggesting movies needs a lot of data and improved ways to organize, process, and connect the data economically. A deeper understanding of how the movie industry works and a better understanding of how people like movies leads to more accurate and personalized movie ideas for customers.

**Objectives**

Data Processing and Loading: Firstly, we are loading the required data set by adding user ratings, movie credits, and metadata for developing the system.

Exploratory Data Analysis (EDA): Patterns in the information can be found using tools like mean vote averages and 90th percentile vote counts. Then we are using

pie charts and bar graphs to show data trends which helps us to understand them better.

Demographic Filtering: By using demographic filtering, we sort popular movies by vote count and weighted rating methods. Then shows most famous movies based on how popular they are.

Content-Based Filtering: It is used for putting text vectorization methods (like TF-IDF) to use for movie recaps. Also, to make personalized suggestions, similarity scores are calculated using cosine similarity. Then cast, crew, genres and keywords are processed for improved analysis.

Collaborative Filtering: Singular Value Decomposition (SVD) and the Surprise library are being used for joint filtering. Making predictions about how users will rate movies.

**Features**

The dataset includes movie’s title, cast, vote average, vote count, popularity, overview, genres, budget, revenue, keywords, and director.

Personalized movie suggestions are generated by considering user ratings, demographic data, movie content, and business factors.

**Background/Related Work**

The Project is about a movie recommendation system using several techniques, such as demographic, content-based, and collaborative filtering. Every technique utilizes different aspects of the movie data to offer personalized recommendations to customers. The motivation behind this project comes from the subject of recommendation systems, which provides personalized suggestions and is crucial to enhancing the user experience.

A basic technique that recommends movies based on broad popularity metrics like average ratings and vote totals is called demographic filtering, and it is implemented in the code. Films with higher average ratings and more votes are considered more appealing to a wider spectrum of audiences, according to this criterion. This is a traditional and simple method.

Sent Content Filtering is another part of the program that focuses on looking at movie specifics like characters, genres, and synopses to find trends and provide relevant suggestions. This strategy is based on the idea that people who enjoy a certain movie would most likely enjoy other movies with similar content. The program uses cosine similarity and text vectorization to determine the degree of similarity between videos.

Furthermore, the project integrates information from movie credits, genres, and keywords using a hybrid technique. By merging directors, actors, and genres, it creates a movie vector that allows for more complex recommendations.

The third technique, known as collaborative filtering, forecasts consumer preferences by using similar user behavior. The Surprise library is used to accomplish matrix factorization by Singular Value Decomposition (SVD) in the code. Recommendation systems frequently employ this technique, which extracts implicit features that impact a user's preferences.

**Dataset**

* The dataset comprises 4,800 movies, all of which came from The Movie Database, a reputable source for movie-related information. huge datasets like this one ensure a huge selection of videos for analysis and training systems.
* Diversity of features: With 20 attributes for each movie, the dataset provides a large number of variables, including revenue, cast, crew, and budget. The diversity allows for a multitude of approaches to be taken to a full exploration of each movie, which improves overall performance of the recommendation system.
* JSON Data Processing: In order to make the most use of the information included in JSON columns, the dataset underwent data pre-processing and data cleaning. To identify relevant aspects for further research, such as cast and crew information, JSON data columns were evaluated.
* Cast, Director, and Genre Variables: Binary variables were made for the following: actors, directors, and genres. This improvement increases the granularity and accuracy of the system, which facilitates the integration of category data into recommendation algorithms.
* Part of the dataset preparation process included organizing categorical data efficiently and formatting it so that it could be analyzed. A crucial step in the process is ensuring that cast and genres have a significant influence on the recommendation system.
* Revenue and Budget: The inclusion of financial metrics, in particular revenue and budget, gives the dataset a more economic perspective. These elements might influence consumer preferences and a more nuanced view of the movie business, which could have an effect on the recommendation system's relevancy and accuracy.

**Details of the Features**

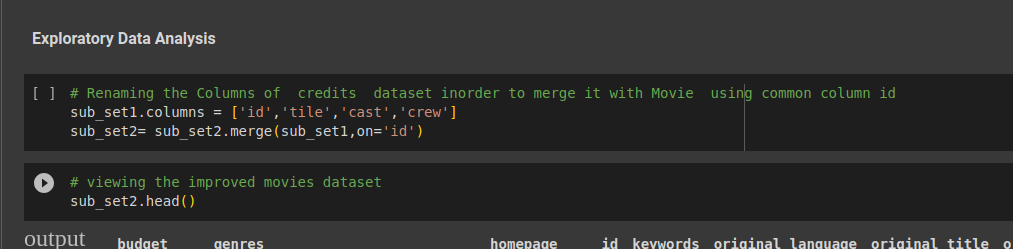
The study's dataset contains a wide range of characteristics that are vital to the recommendation system. These features offer a range of informative aspects about each movie, enabling a thorough analysis. The primary characteristics include:

* Title: The title of the movie acts as a fundamental method of recognition for each submission.
* Cast & Staff: These characteristics, which are derived from the "credits" dataset, include details on the actors and crew members that worked on each movie. They provide demographic screening in addition to content-based filtering.
* Vote Average: The mean score that viewers have given the movie, indicating how excellent they believe it to be overall.
* Vote Count: The total number of votes cast for a film, indicating how well-liked and involved viewers are with it.
* Popularity: A metric that expresses a film's level of popularity, possibly impacted by outside forces like public interest or marketing campaigns.
* Overview: An e-text summary of the film's plot that is essential for content-based filtering that makes use of natural language processing methods.
* Genres: Binary variables that show whether a given movie belongs in a particular genre or not. The hybrid and content-based filtering techniques are improved by this category feature.
* Budget and Revenue: Economic indicators that show the financial aspects of each film, impacting user choices and adding to the financial viewpoint of the recommendation system.
* Keywords: These are taken from the "credits" dataset and offer more contextual details about the movie's plot, which improves content-based suggestions.
* The film's director is a valuable source of information about the creative leadership of each picture and helps with content-based filtering.

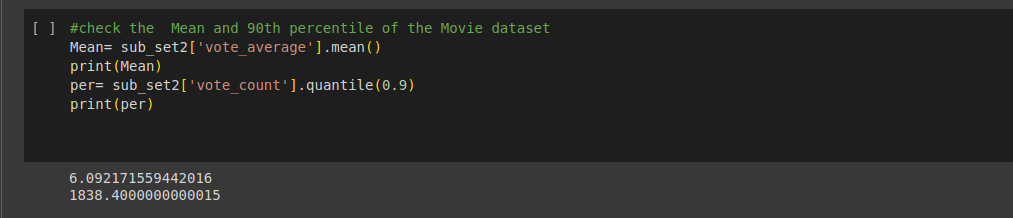
By combining user ratings, demographic data, movie content, and economic considerations, these elements enable the recommendation engine to provide a wide range of tailored and unique movie recommendations.

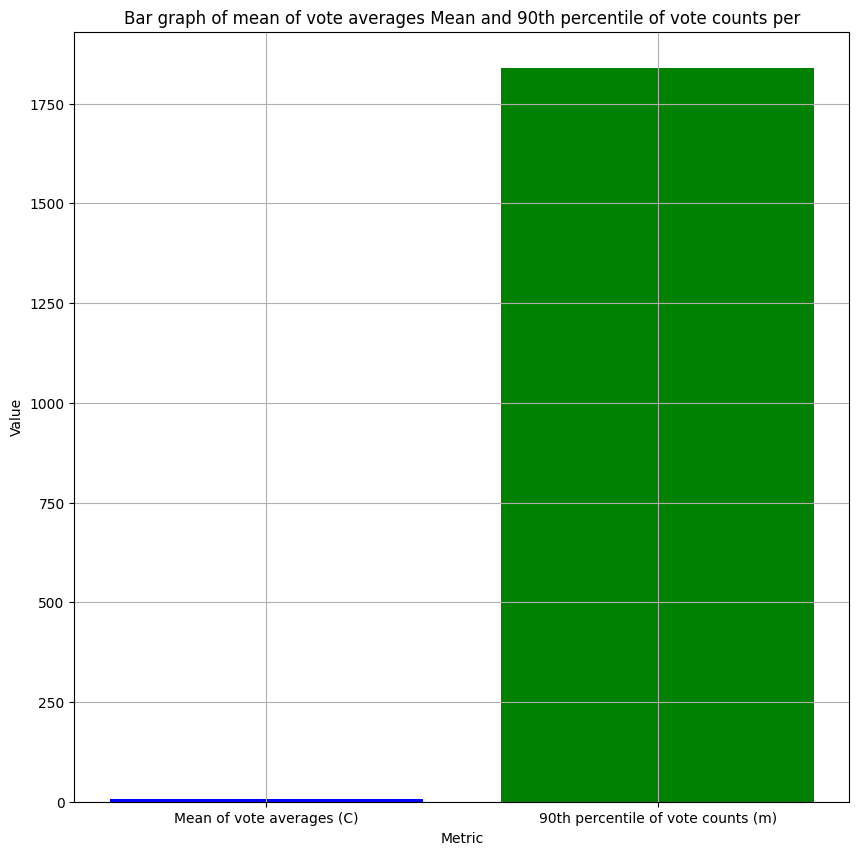
**Analysis**

This exploratory data analysis's (EDA) primary objective is to comprehend and present significant components of the movie dataset. First, the common identifier "id" is used to merge the "credits" and "movie" datasets. A consolidated dataset that you may evaluate is produced by renaming columns to ensure that the merging process proceeds smoothly.

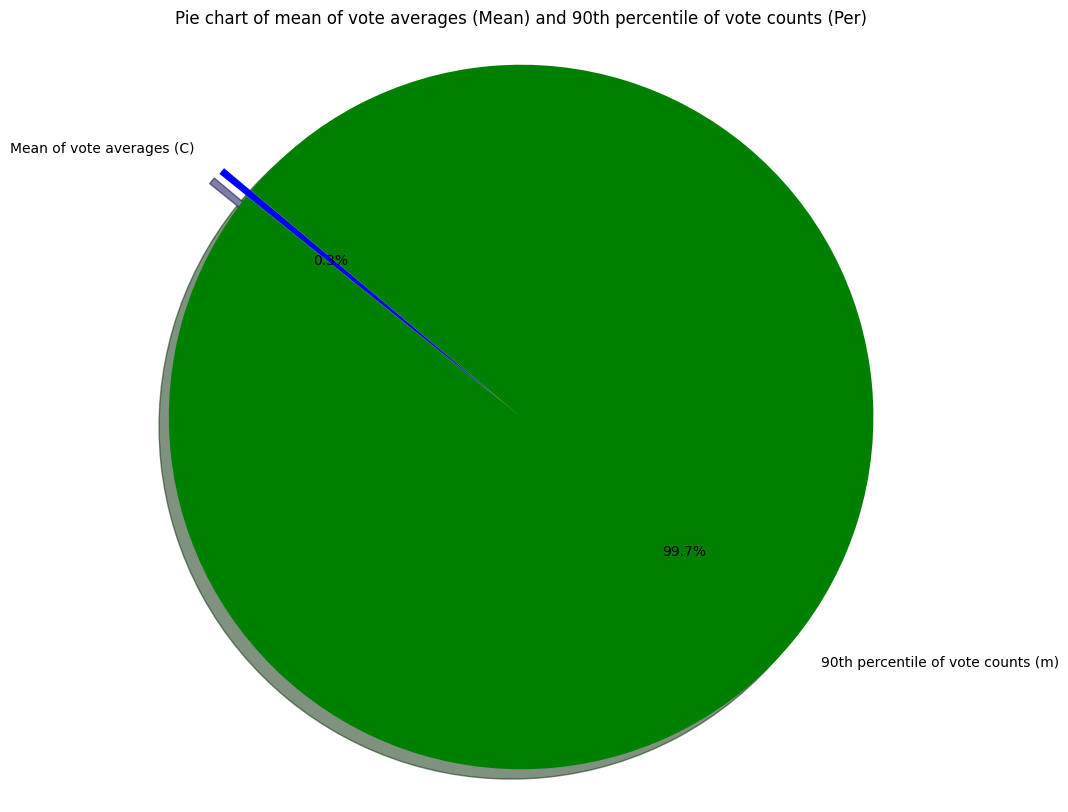
Figure 1a. Renaming and merging columns

Subsequently, the EDA examines the principal trends in the dataset. Calculating the mean of vote averages Mean and the 90th percentile of vote counts per yields estimates for the distribution and popularity measures. For better comprehension, visual aids are employed.

figure2a. Mean and 90th percentile

****figure 3a. Bar graph of averages mean and 90th percentile.

The mean of vote averages and the 90th percentile of vote counts are effectively contrasted in a bar graph to give a clear visual comparison. Pie charts provide yet another method of displaying this data in percentage terms. These graphics highlight two subjects that assist give a comprehensive overview of the dataset's main patterns: the average user ratings and the threshold for a film to be considered highly voted.

Figure 4a. Pie chart of averages and 90th percentile.

A graph showing a number of orange dots

Description automatically generated

Figure 5a. Scatter plot Budget vs Revenue

A graph of a person with a blue line

Description automatically generated with medium confidence

Figure 6a. Release date vs Movie release

**Implementations**

This project includes a movie recommendation system with many implementations, such as content-based filtering, collaborative filtering, and demographic filtering. The key steps are as follows, broken down:

**1.Data Processing and Loading**

Three datasets are loaded at the beginning of the project: user ratings, movie credits, and movie metadata.

The libraries required for modeling and data manipulation are imported.

The columns in the credits dataset are renamed before they are combined with the movie information dataset.

**2.Data analysis**

To provide an idea of the structure of the data, the first rows of each dataset are shown.

To obtain information about the popularity distribution of the dataset, EDA computes key trends such as the mean of vote averages and the 90th percentile of vote counts.

The core patterns may be easily seen using visualizations such as pie charts and bar graphs.

**3.Screening by Demographic**

The popular films are identified by filtering out the films with vote counts more than or equal to the mean.

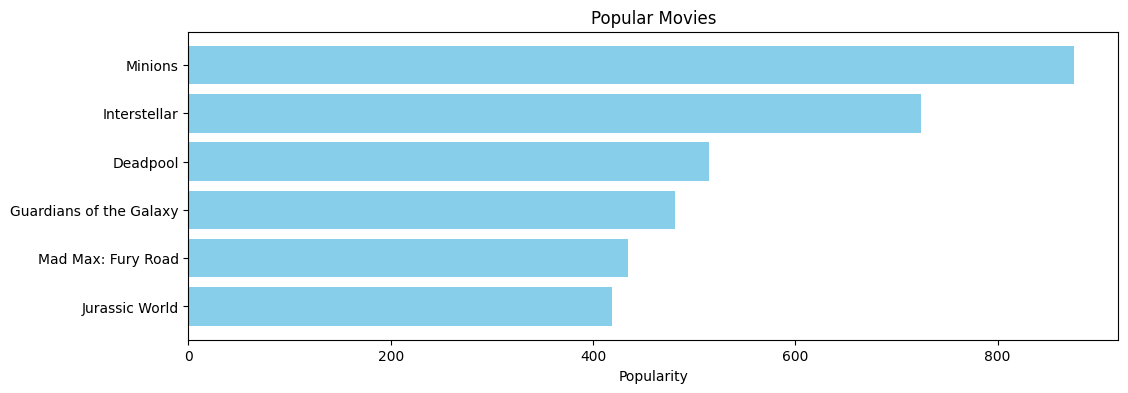
A screen shot of a computer

Description automatically generated

Figure 7a. Sorting score for q\_movies data.

Every movie is assigned a score using a weighted rating system that is defined using the IMDB methodology.

Based on this rating, movies are arranged, with the top 10 shown.

 Figure 8a. Horizontal plot of popular movies vs popularity

Using popularity measurements, a horizontal bar plot illustrates the most popular films.

**4. Through Content Retention**

Text vectorization techniques such as TF-IDF are applied to the movie summary.

The linear kernel and cosine similarity are used to calculate the similarity scores between movies.

The recommendation system provides tailored options based on the degree of similarity between movies.

A screen shot of a computer program

Description automatically generated

Figure 9a. Tdidf Matrix shape

**5.Based on Credits, Genres, and Keywords Prediction**

Cast, crew, genres, keywords, and other details are processed into Python objects.

Directories and name lists are extracted from features using function definitions.

The introduction of new features and data cleansing lead to improved analysis.

Relevant data is compiled to produce a "soup" feature, depending on content.

A screenshot of a computer program

Description automatically generated

Figure 10a. Parsing and extracting features.

**6.Collaborative Filtering**

The Surprise library is used to create collaborative filtering with singular value decomposition (SVD).

The SVD model is trained and evaluated using a surprise dataset that contains user ratings data.

Predictions are generated for user-specific movie ratings using collaborative filtering.

A screenshot of a computer

Description automatically generated

Figure 11a. SVD results prediction vs actual rating

**SVD vs KNN:**

Here we performed SVD Vs KNNBasic on our dataset to determine which has more accuracy.

**A screenshot of a computer program

Description automatically generated**

Figure 12a. Results of KNNBasic Vs SVD

**Preliminary Results**

**Use and Interaction**

Users can input the title of a movie to receive content-based suggestions.

The entire implementation provides a comprehensive movie recommendation system that provides a well-rounded user experience through the combination of demographic, collaborative, and content-based filtering algorithms.

A screenshot of a computer

Description automatically generated

Figure 13a. Demo of Output

**Project Management**

**Implémentation/ Status Report**

We have completed 80% of the project which is as follows:

Worked on implementing data processing, data analysis and demographic filtering and contributed to Documentation by Anusha Chilakamarri. (20%)

Implemented content-based filtering, including text vectorization and cosine similarity which helps in movie recommendation. Contributed to Documentation by Bhuvaneswar Reddy Sriyyapureddy(20%).

Have worked on collaborative filtering using Singular Value Decomposition and the Surprise library. Training and evaluating the SVD with user ratings. Predicting ratings for movies vs actual ratings and contributed to Documentation by Poojitha Pothini (20%).

Worked on implementing the recommendation system and on comparing different recommendation algorithms using A/B testing which includes evaluation metrics such as RMSE, MAE. Also contributed to Documentation by Ram Srinivas Katragadda (20%).

**Work to be completed. (20%)**

Team to work on implementing the recommendation using either python libraries or web interface to present the output in the webpage or dialog or popup box which helps in searching the movies and works as an interface to surf regarding the movies which is like Netflix interface. This is the last part of the project which needs to be completed.

Team is together working on this part.

**References**

Tahmasebi, H., Ravanmehr, R., & Mohamadrezaei, R. (2021). Social movie recommender system based on deep autoencoder network using Twitter data. *Neural Computing and Applications*, *33*, 1607-1623. [Link](https://link.springer.com/article/10.1007/s00521-020-05085-1)

Wang, S., Hu, L., Wang, Y., Cao, L., Sheng, Q. Z., & Orgun, M. (2019). Sequential recommender systems: challenges, progress and prospects. *arXiv preprint arXiv:2001.04830*. [Link](https://researchers.mq.edu.au/en/publications/sequential-recommender-systems-challenges-progress-and-prospects)

Wang, A., GHADI, A., & FENNAN, A. (2021). ExMrec2vec: explainable movie recommender system based on Word2vec. *International Journal of Advanced Computer Science and Applications*, *12*(8). [Link](https://www.researchgate.net/profile/Abdelhadi-Fennan/publication/354332958_ExMrec2vec_Explainable_Movie_Recommender_System_based_on_Word2vec/links/61708f0c766c4a211c030541/ExMrec2vec-Explainable-Movie-Recommender-System-based-on-Word2vec.pdf)

nal, S. S., PrasKhaad, P. W. C., Alsadoon, A., & Maag, A. (2020). A systematic review: machine learning based recommendation systems for e-learning. *Education and Information Technologies*, *25*, 2635-2664. [Link](https://link.springer.com/article/10.1007/s10639-019-10063-9)

Cui, Z., Xu, X., Fei, X. U. E., Cai, X., Cao, Y., Zhang, W., & Chen, J. (2020). Personalized recommendation system based on collaborative filtering for IoT scenarios. *IEEE Transactions on Services Computing*, *13*(4), 685-695. [Link](https://ieeexplore.ieee.org/abstract/document/8951278)

Wang, S., Cao, L., Wang, Y., Sheng, Q. Z., Orgun, M. A., & Lian, D. (2021). A survey on session-based recommender systems. *ACM Computing Surveys (CSUR)*, *54*(7), 1-38. [Link](https://dl.acm.org/doi/abs/10.1145/3465401?casa_token=Y177GO87bYUAAAAA:AUP_UZVoWaYQSS6Kfnf_IagaIoP7FdvQkySEid7fMBxBTuU5cJhr6cmUeQIQaNaE0fpqJAC8vCvT8w)

Choudhury, Sasmita Subhadarsinee, Sachi Nandan Mohanty, and Alok Kumar Jagadev. "Multimodal trust based recommender system with machine learning approaches for movie recommendation." *International Journal of Information Technology* 13 (2021): 475-482. [Link](https://link.springer.com/article/10.1007/s41870-020-00553-2)

Mitra, A. (2020). Sentiment analysis using machine learning approaches (Lexicon based on movie review dataset). *Journal of Ubiquitous Computing and Communication Technologies (UCCT)*, *2*(03), 145-152. [Link](https://irojournals.com/jucct/article/view/2/3/4)