MovieLens Dataset Recommendation System Report

1.Executive Summary

This report outlines the development of a recommendation system using the MovieLens dataset. The project's primary goal is to provide personalized movie recommendations to users based on their ratings of other movies, leveraging collaborative filtering techniques. Key findings and methodologies are detailed, along with evaluation metrics and challenges encountered. The report concludes with recommendations for enhancing user experience and engagement through targeted suggestions.

2. Introduction

Project Overview

The MovieLens Recommendation System project aims to build a model capable of providing top 5 movie recommendations to users based on their historical ratings. This project uses the MovieLens dataset—a widely-used benchmark in recommendation system research—to demonstrate the effectiveness of collaborative filtering and hybrid approaches.

Background

The MovieLens dataset contains explicit user ratings and metadata, making it ideal for building and evaluating recommendation systems. By addressing the cold start problem and improving prediction accuracy, this project aims to enhance user satisfaction and retention on platforms utilizing such systems.

3.Business Understanding

Objectives

1. Conduct thorough exploratory data analysis (EDA) to understand user preferences and rating trends.
2. Develop a recommendation system that provides personalized movie recommendations based on user ratings.
3. Implement collaborative filtering and content-based approaches to improve recommendation accuracy.
4. Evaluate the recommendation system using appropriate performance metrics to ensure relevance and accuracy
5. Provide actionable insights to stakeholders to enhance user satisfaction and engagement strategies.

Business overview

Recommendation systems are critical for platforms like MovieLens, which rely on user engagement to thrive. A robust recommendation engine can drive user satisfaction, increase platform activity, and improve retention rates.

Problem statement

Users often face difficulties in discovering movies that match their preferences due to the overwhelming number of choices available. This lack of personalized recommendations leads to lower user engagement and potential dissatisfaction with the platform.So, how can we improve user engagement and retention by providing personalized movie recommendations based on their past ratings?

This project aims to develop a collaborative filtering-based recommendation system that provides personalized movie suggestions by analyzing a user's past ratings, thereby improving their overall experience and engagement.

4. The metrics of success

To evaluate the effectiveness of the recommendation system, the following metrics will be used:

- Precision: Measures the proportion of recommended movies in the top 5 results that are relevant to the user.
- Mean Absolute Error (MAE): Quantifies the average magnitude of errors between predicted and actual ratings.
- Root Mean Square Error (RMSE): Provides a measure of prediction accuracy by penalizing larger errors more heavily.

These metrics will ensure the recommendation system is both accurate and effective in providing meaningful movie suggestions to users.

5.Data understanding

The dataset used in this project is sourced from the GroupLens Research team, titled "MovieLens Latest Small Dataset." It is a CSV-based dataset containing user ratings, movie information, and additional features relevant for building a recommendation system. You can access it here: [MovieLens Dataset](https://grouplens.org/datasets/movielens/latest/).

Summary of features in the dataset:

Only movies with at least one rating or tag are included in the dataset. Movie ids are consistent between `ratings.csv`, `tags.csv`, `movies.csv`, and `links.csv` (i.e., the same id refers to the same movie across these four data files).

All ratings are contained in the file `ratings.csv`. Ratings are made on a 5-star scale, with half-star increments (0.5 stars - 5.0 stars).

Timestamps represent seconds since midnight Coordinated Universal Time (UTC) of January 1, 1970.

Movie information is contained in the file `movies.csv`. Movie titles include the year of release in parentheses. Errors and inconsistencies may exist in these titles.

The data are contained in the files `links.csv`, `movies.csv`, `ratings.csv` and `tags.csv`.

The links and movies datasets each consist of 9,742 rows and 3 columns.

- The ratings dataset contains 100,836 rows and 4 columns.
- The tags dataset includes 3,683 rows and 4 columns.

Column Breakdown:

- **Ratings Dataset:**
    - **userId**: Unique identifier for each user.
    - **movieId**: Unique identifier for each movie.
    - **rating**: User rating for the movie (1.0 to 5.0).
    - **timestamp**: Timestamp of the rating.

- **Movies Dataset:**
    - **movieId**: Unique identifier for each movie.
    - **title**: Title of the movie.
    - **genres**: Genres of the movie (e.g., "Drama|Comedy").
- **Tags Dataset:**
    - **userId**: Unique identifier for each user.
    - **movieId**: Unique identifier for each movie.
    - **tag**: User-provided tag for the movie.

    - **timestamp**: Timestamp of the tag.
- **Links Dataset:**
    - **movieId**: Unique identifier for each movie.
    - **imdbId**: IMDb identifier for the movie.
    - **tmdbId**: TMDb identifier for the movie.

6.Methodology

Data Collection

The MovieLens dataset includes:

- User ratings of movies (on a scale of 0.5 – 5.0).
- Movie metadata, such as genres and release years.
- User demographic information (limited in the small dataset).

Data Preparation

- Import all the necessary libraries
- Load the dataset using pandas library
- Mereged the datasets(ratings dataset as the primary dataset merged with the movies dataset)
- Cleaned missing values, duplicates and outliers.

Data Analysis:

Conducted analysis to understand the distribution of ratings, frequency of movies rated, and user engagement levels.

Merging Datasets

The primary dataset, ratings_df, will be enriched by merging it with the supplementary datasets:

1. **Merging Movies Dataset**: A left join was performed between movies_df and ratings_df to retain all rows from ratings_df while incorporating metadata from movies_df, such as titles and genres. Both datasets share a common key, movieId.
2. **Incorporating Tags Dataset**: Further enhancement of the primary dataset will be achieved by merging with tags_df. However, due to the sparsity of the tags data, this merge is likely to introduce many missing values into the primary dataframe.
3. **Excluding Links Dataset**: The links_df was not merged into the primary dataset because it primarily contains identifiers for linking to external sources (like IMDb and TMDb), which are not immediately useful for building recommendations. If external data becomes necessary in the future (e.g., IMDb ratings), a merge can be conducted at that time.

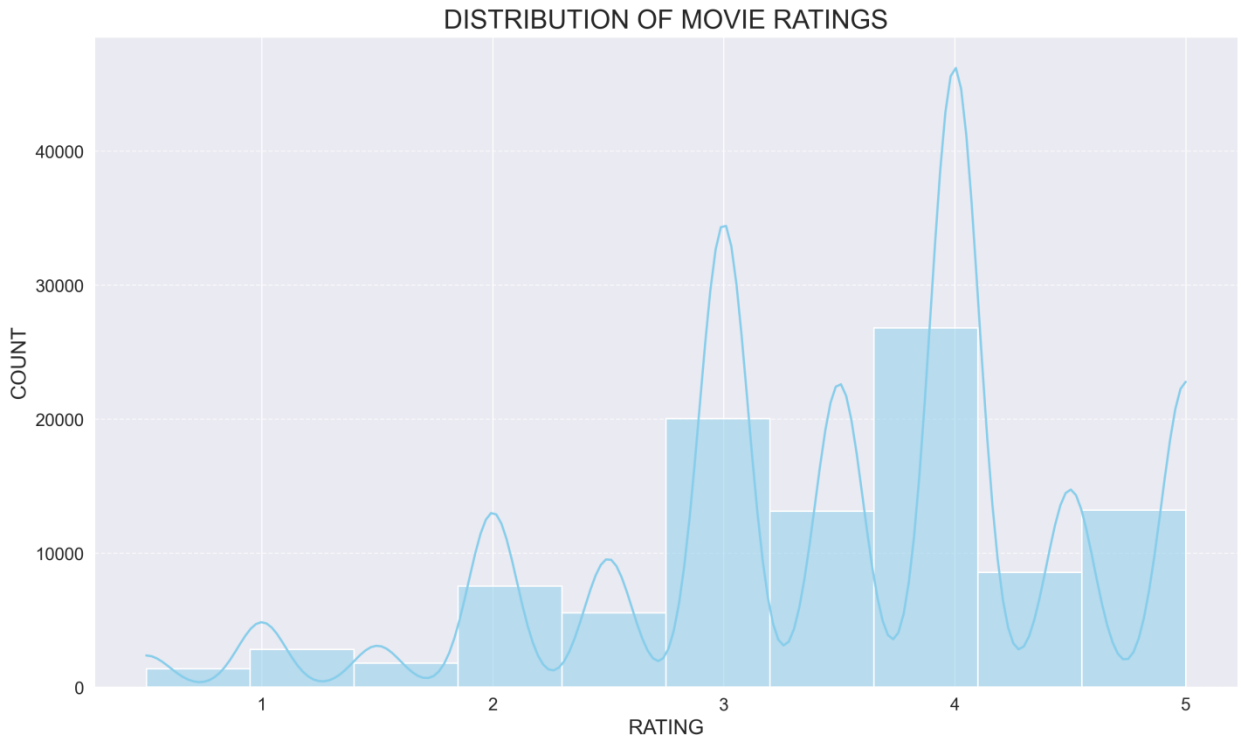
EDA & VISUALIZATION

This section presents the Exploratory Data Analysis (EDA) performed on the dataset to evaluate its quality and extract valuable insights. The analysis involves a detailed review of key columns to uncover trends and relationships within the data.

A variety of visualizations are employed to summarize the findings, detect patterns, and enhance the understanding of the dataset, which will support the development of an effective movie recommendation system.

1. Distribution of ratings

Started by exploring the distribution of ratings within the dataset to identify trends, such as whether users generally provide higher or lower ratings.

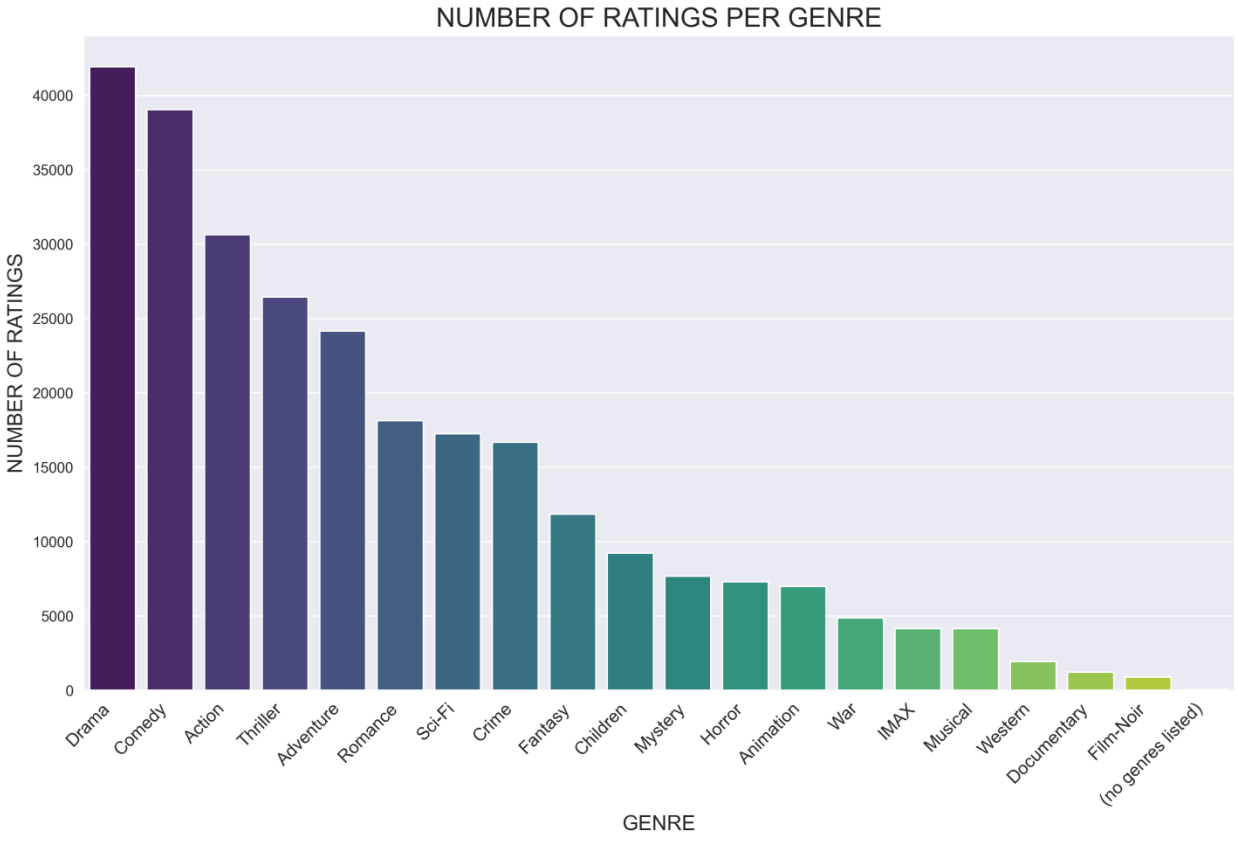


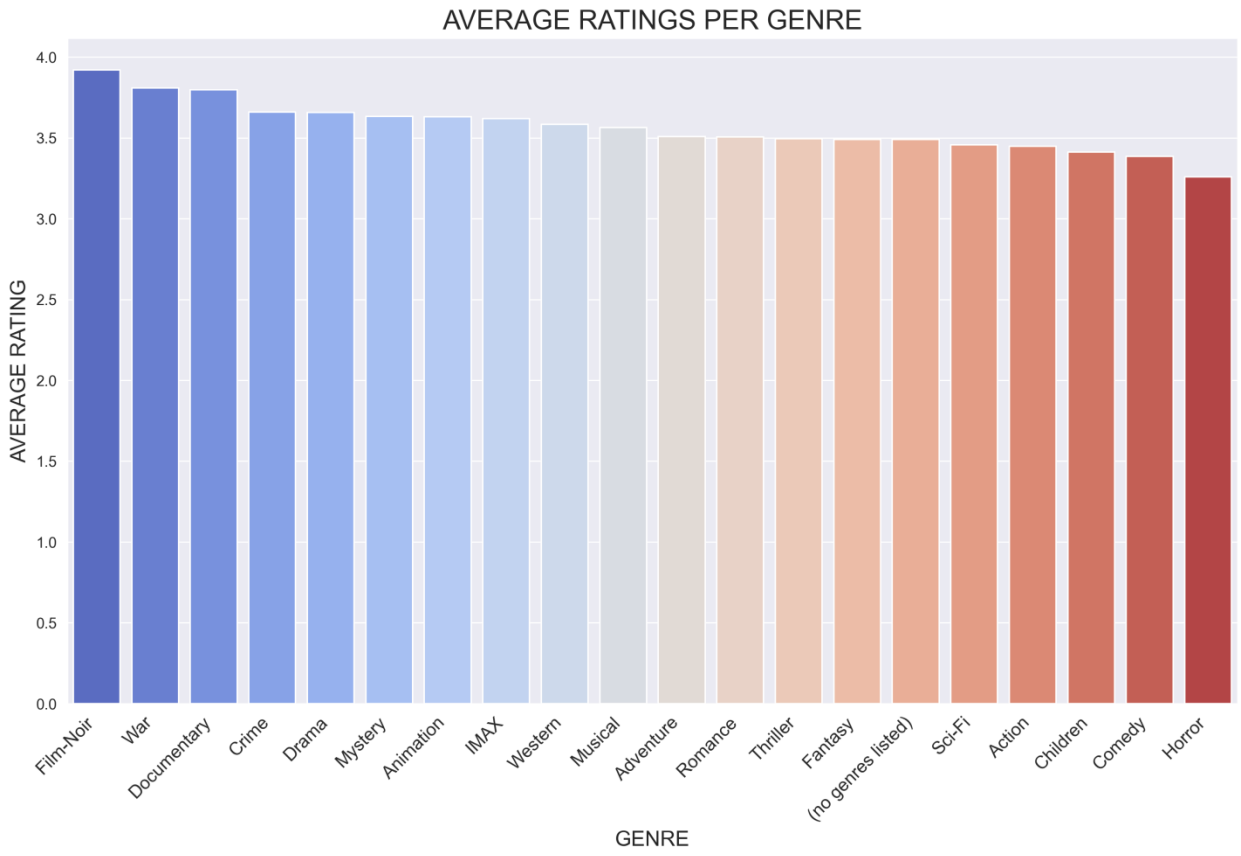
Observation

The plot indicates that the highest peaks occur at ratings 3 and 4, while ratings 1 and 2 are notably low. The distribution of movie ratings suggests that users tend to give moderate to positive ratings, reflecting a preference for average or slightly favorable reviews rather than extreme ratings of 1 or 5.

2. Genre popularity analysis

Analyzed which genres receive the most ratings and the highest average ratings to better understand audience preferences.


NUMBER OF RATINGS PER GENRE
NUMBER OF RATINGS
0
5000
10000
15000
20000
25000
30000
35000
40000
Drama
Comedy
Action
Thriller
Adventure
Romance
Sci-Fi
Crime
Fantasy
Children
Mystery
Horror
Animation
War
IMAX
Musical
Western
Documentary
Film-Noir
(no genres listed)
GENRE



Observation

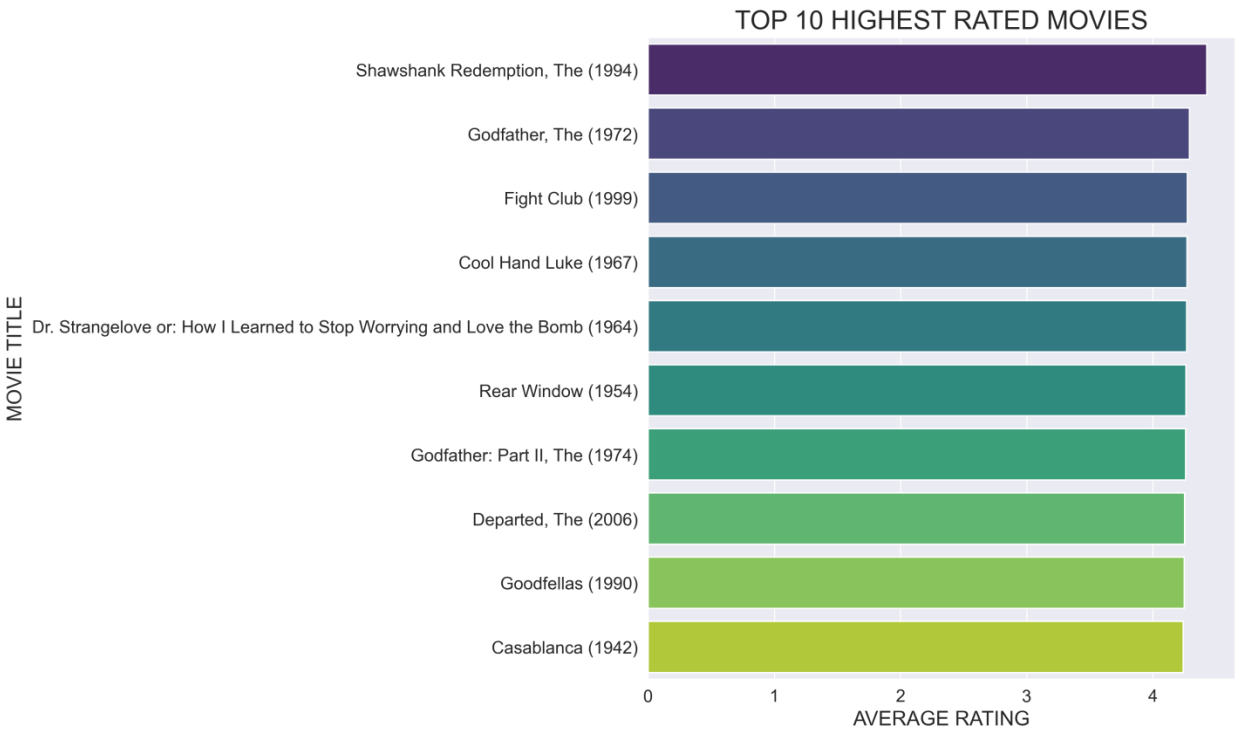
The **first plot** illustrates the number of ratings across various movie genres, revealing that some genres attract significantly more engagement than others. Notably, genres such as drama, comedy, action, thriller, and adventure are among the most-rated, while film-noir, musicals, documentaries, and westerns receive considerably fewer ratings.

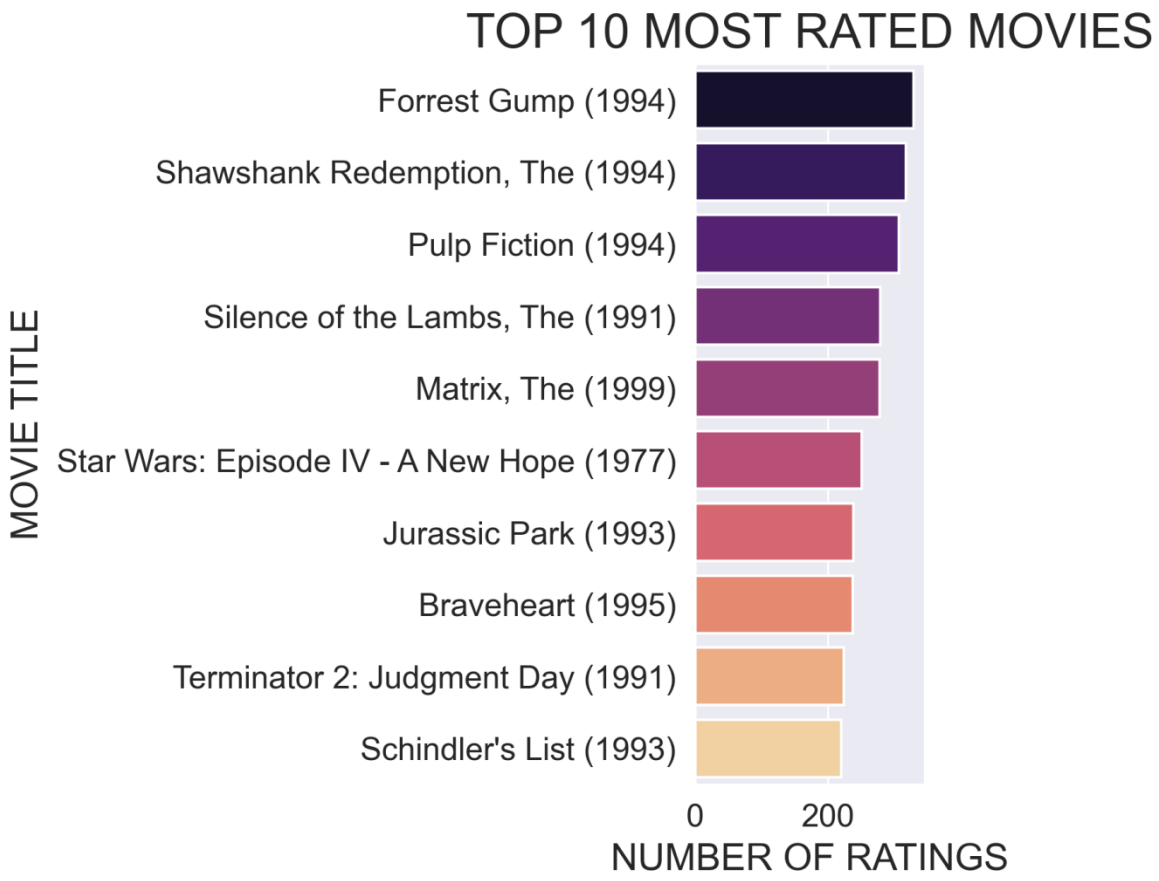
The **second plot** presents the average rating for each genre, highlighting differences in user satisfaction. Interestingly, despite being among the least-rated genres, film-noir achieves the highest average rating, closely followed by war and documentary genres.

Genres like Drama and Comedy garner the highest number of ratings, indicating their broad popularity. However, genres with fewer ratings may still boast higher average ratings, suggesting a niche but content audience.

3. Most-rated and top-rated movies

In this section, we will identify movies with the highest average ratings and those with the most ratings to determine popular films. We will also assess whether these results align with the genre popularity analysis.





Observation

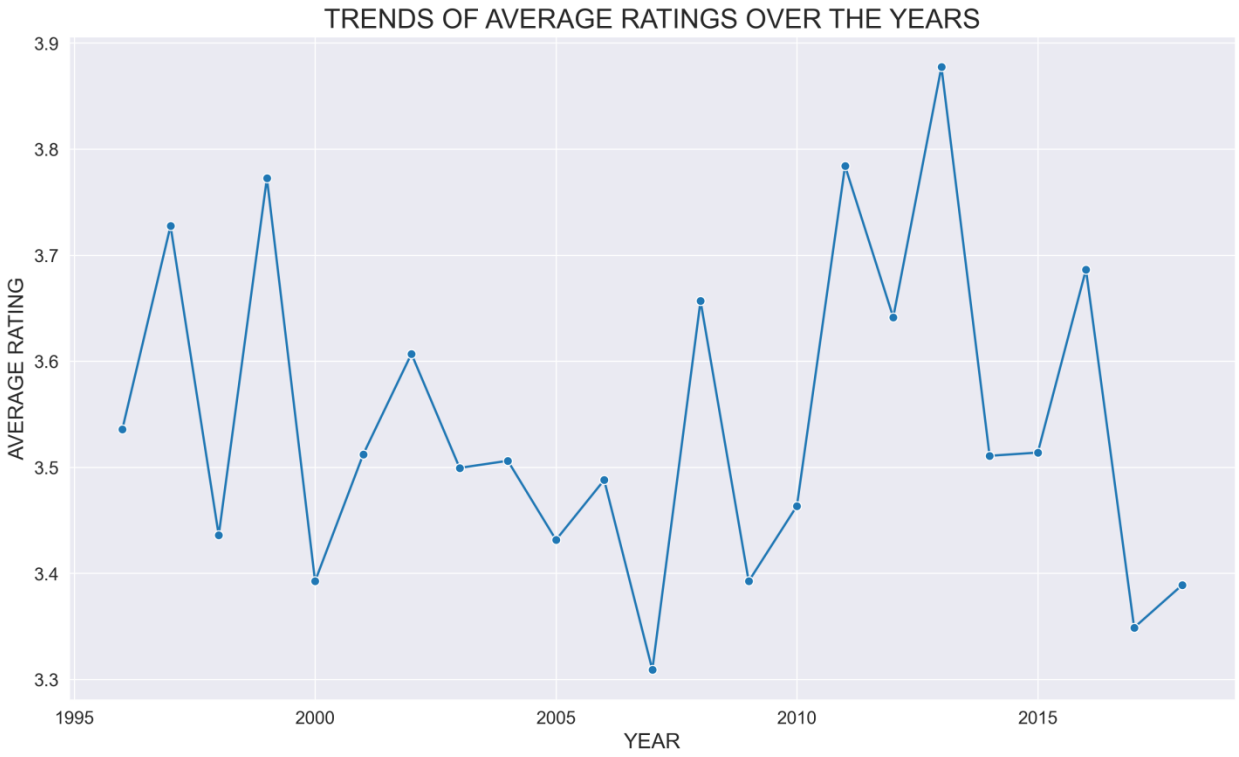
The visualization indicates that certain movies have received significantly higher average ratings, with the top-rated films averaging above 4.5. However, as noted in previous plots, these films may have a lower overall number of ratings, suggesting a niche but satisfied audience.

Conversely, the most-rated movies indicate broader audience engagement, reflecting their widespread popularity, albeit with slightly lower average ratings.

Movies like ***The Shawshank Redemption* (1994)** appear on both the top-rated and most-rated lists, indicating it is a highly popular film that performed well.

4. Rating trends overtime

In this section, we will visualize the trends in ratings over time to examine how ratings have evolved throughout the years.

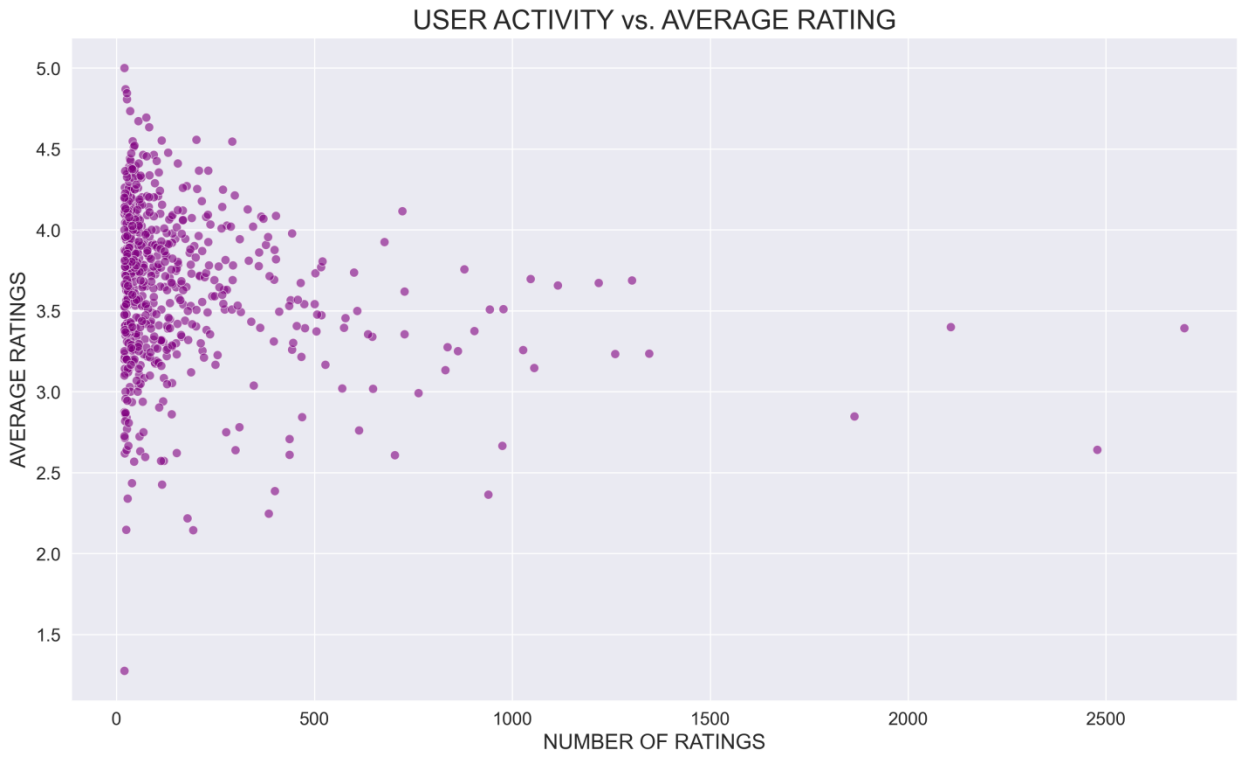


Observation

The visualization reveals that the average rating of movies has varied over the years, with a noticeable decline around **2006-2007**. These fluctuations may be influenced by factors such as shifts in user preferences, changes in movie quality, or the impact of social media.

5. Correlation analysis between user activity and ratings

In this section, we will analyze whether highly active users rate movies differently compared to less active users.

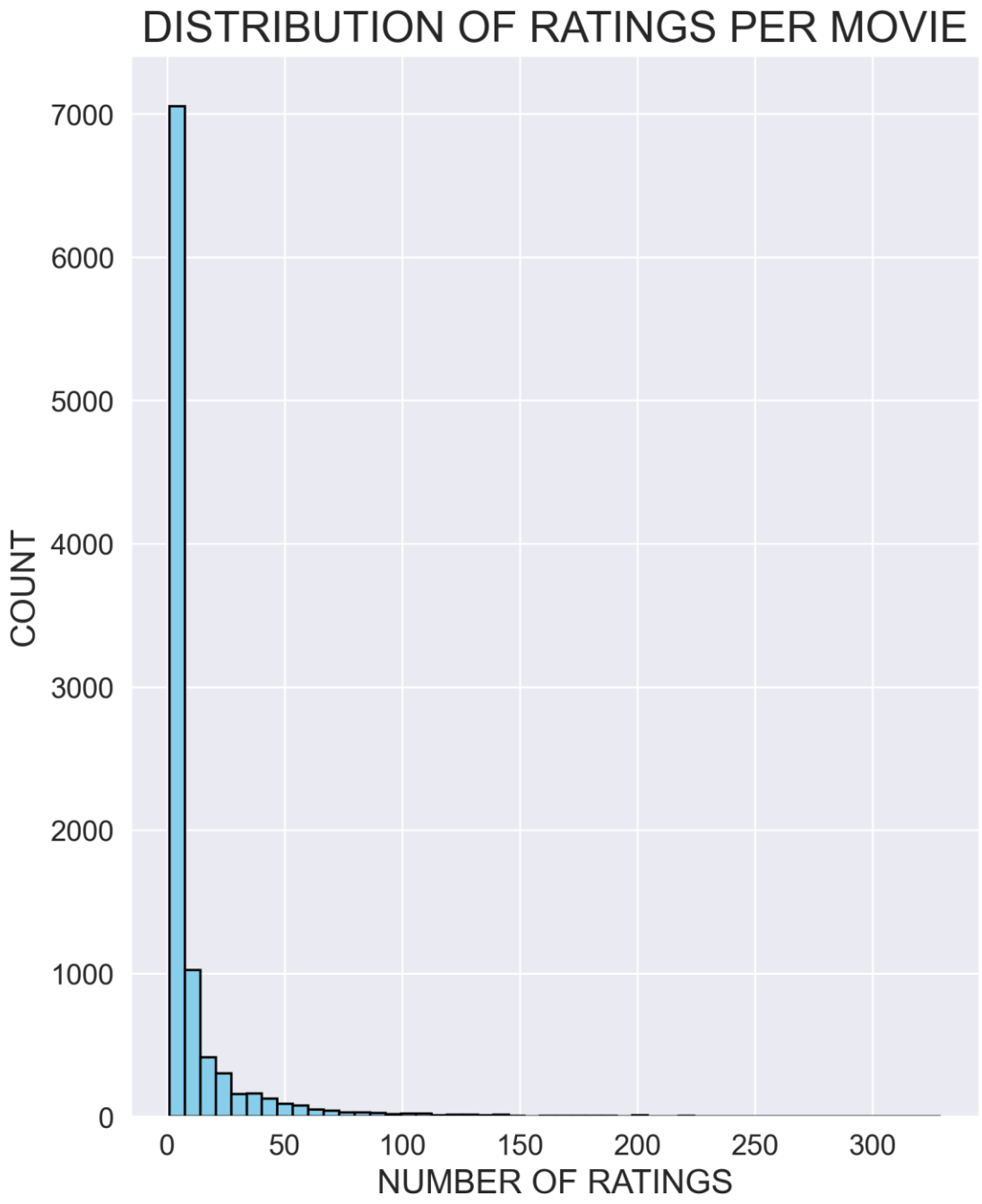


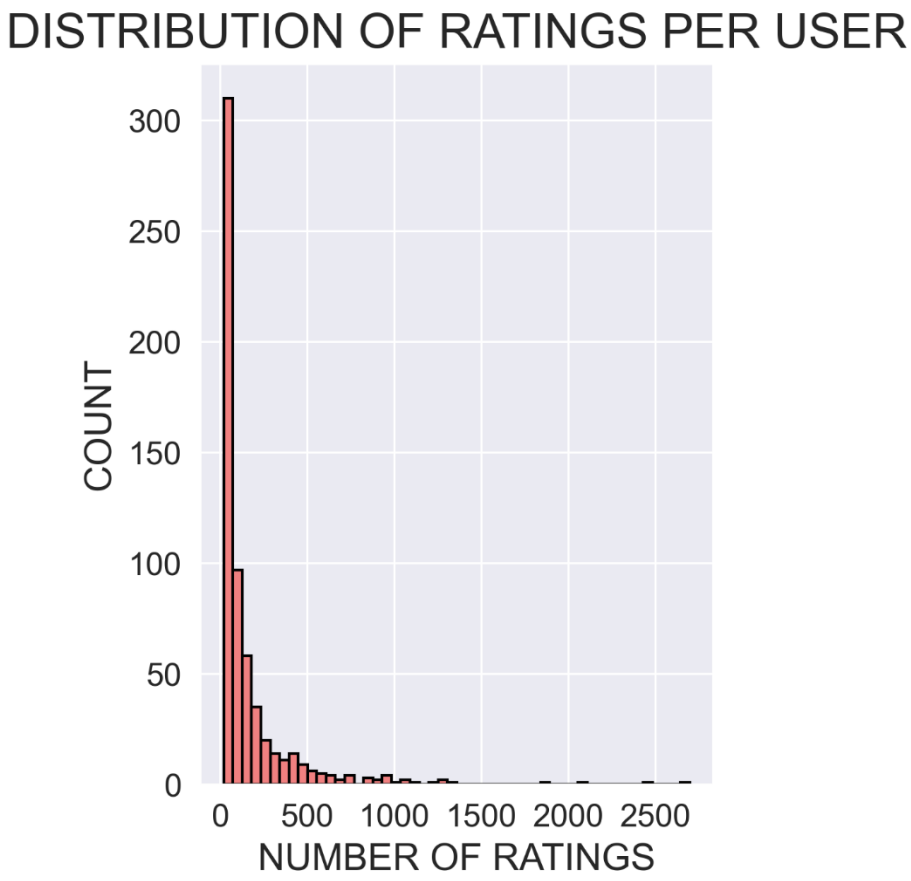
Observation

The scatter plot suggests that users who provide more ratings (more active users) tend to have average ratings within a specific range, while less active users exhibit greater variability in their ratings. Highly active users may have more consistent preferences compared to those who rate sporadically.

6. Cold-start problem investigation

Here, we investigate new users or movies with very few ratings, which could affect the effectiveness of recommendations.

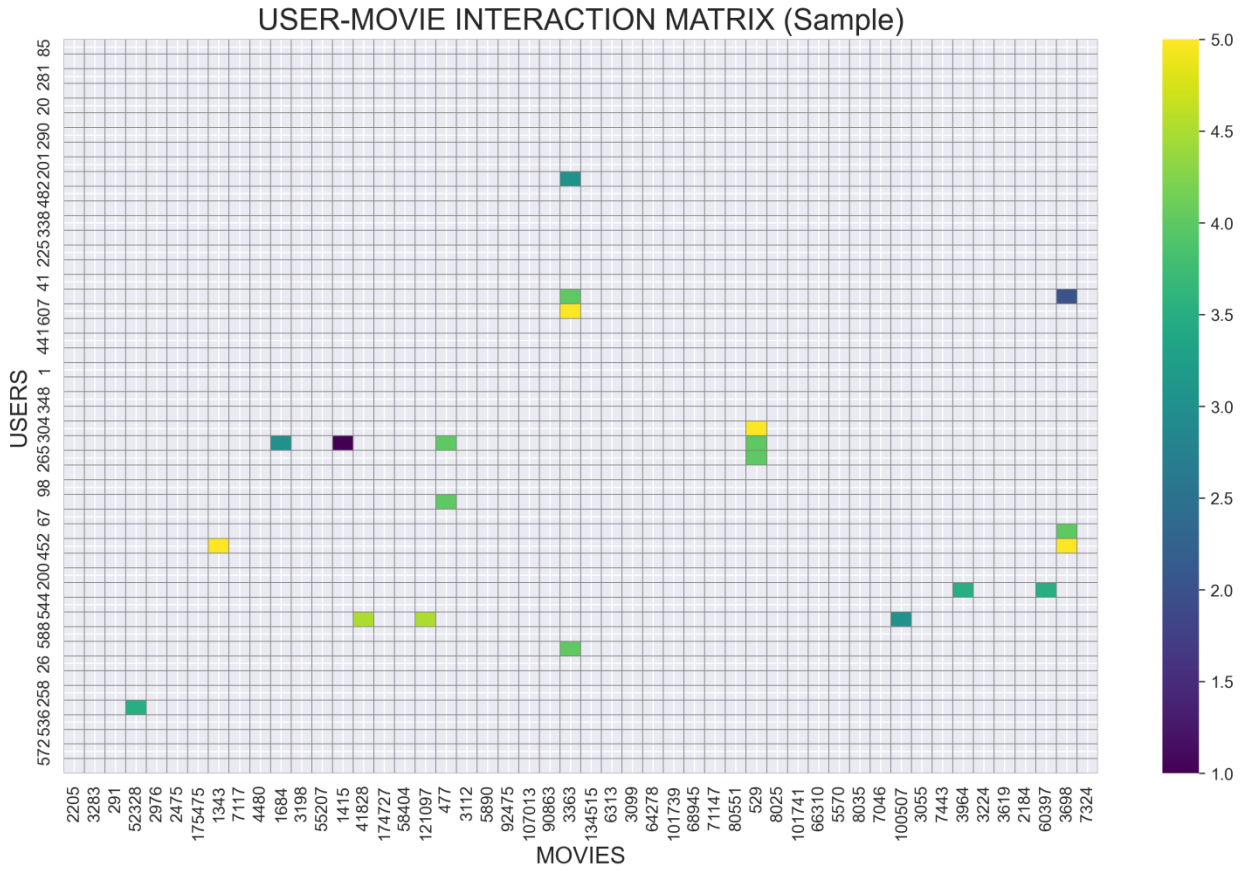

DISTRIBUTION OF RATINGS PER MOVIE
COUNT
7000
6000
5000
4000
3000
2000
1000
0
0
50
100
150
200
250
300
NUMBER OF RATINGS



Observation

The distributions indicate that a significant number of movies and users have very few ratings, highlighting a potential cold start problem. A few popular movies receive the majority of ratings, while many have very few, which could hinder recommendation accuracy for new items or users.

7. User-movie interaction matrix analysis



Observation

The visualization above demonstrates that the user-movie interaction matrix is **highly sparse**, with most cells having missing values (no rating). This sparsity indicates that users rate only a small fraction of available movies, which is typical in such a dataset.

SUMMARY FINDINGS IN EDA

- **Majority of the artings cluster around 3 and 4**: The distribution of ratings shows that most users tend to rate movies with a score of 3 or 4. With this insight, stakeholders can focus on movies that consistently receive ratings above 4 for promotional efforts and personalized recommendations.

- **Certain genres dominate user preferences**: Popular genres such as Drama, Comedy, and Action receive the highest number of ratings, showing they align with audience interests. Therefore, these genres can be prioritized in marketing campaigns, and content acquisition can focus on similar genres to boost engagement.
- **Highly active users provide more stable ratings**: Analysis showed that users with higher activity levels tend to rate movies more consistently, whereas less active users show greater variability. With this insight, stakeholders can consider loyal and active users segmentation for premium recommendation services or loyalty rewards.
- **Fluctuating trends in ratings over the years**: The number of ratings fluctuates over time, with notable increases during specific years, possibly due to platform growth or popular movie releases. Understanding peak activity periods can help the stakeholders plan marketing campaigns or feature releases during high-engagement times.
- **Most-rated movies vs. highest-rated movies differ**: Some of the most frequently rated movies don't necessarily have the highest average ratings, indicating that **popularity does not always align with quality**. Stakeholders should differentiate between "popular" and "high-quality" movies when curating recommendations for users.
- **Variation in ratings across different genres**: Some genres, such as Documentary and Film-Noir, tend to receive higher average ratings but have fewer overall ratings compared to mainstream genres. Promoting niche genres with high satisfaction rates can attract dedicated audiences and differentiate content offerings.

MODELING

PREPROCESSING DATA FOR MODELING

Summary of the Preprocessing Steps;

- Split data for model training and evaluation.
- Check for missing values and duplicates.
- Drop unnecessary columns (title, genres, timestamp, year).
- Encode categorical columns (userId, movieId).
- Create a user-item interaction matrix to prepare for collaborative filtering.
- Converted data into matrix format for model input

BASELINE MODEL

RMSE: 0.8855

MAE: 0.6768

Baseline RMSE: 0.8854999286817727

Baseline MAE: 0.6768416189880477

A lower RMSE value means better performance because it indicates smaller differences between predicted and actual ratings

In this case, an RMSE of 0.8855 means that the predicted movie ratings deviate from the actual ratings by around 0.88 rating points, considering squared errors.

Since the movie ratings range from 0 to 5, an RMSE of 0.88 means the model is doing a reasonable job but still has some room for improvement. If the RMSE were closer to 0, it would mean near-perfect predictions. Values above 1 would indicate higher deviations, meaning less accuracy.

An MAE of 0.6768 means that, on average, the model's predictions are off by around 0.67 rating points from the actual ratings. Lower values indicate better model accuracy. In this case, an MAE of 0.67 suggests the model is fairly accurate but still can be improved to minimize prediction errors further.

Overall the model is performing fairly well, with prediction errors under 1 and while the initial results are decent, there's room for improvement through hyperparameter tuning and optimization.

OPTIMIZATION I

In this setion, we'll optimize the SVD model's performance by tuning its hyperparameters using GridSearchCV

Best RMSE Parameters: {'n_factors': 5, 'n_epochs': 10, 'lr_all': 0.01, 'reg_all': 0.02}

Best RMSE Score: 0.8795363842060014

Interpretating our first optimization step

n_factors: 5 → The best model uses 5 latent features to represent user and movie relationships.

n_epochs: 10** → The model performs best when trained for 10 iterations.

lr_all: 0.01** → The best learning rate for all parameters is 0.01.

reg_all: 0.02** → The best regularization term to prevent overfitting is 0.02.

The best RMSE score obtained during cross-validation is 0.8795, meaning that the model's average prediction error is 0.87 rating points away from the actual ratings in the training data.

RMSE: 0.8911

MAE: 0.6796

Optimization I RMSE: 0.8910951022741277

Optimization I MAE: 0.6796376338220568

The RMSE of the tuned model is slightly higher (0.8911) than the baseline RMSE (0.8855), meaning the tuned model's predictions have slightly larger errors compared to the baseline. Ideally, RMSE should be lower after tuning, indicating better performance, but in this case, the increase suggests that tuning didn't improve the model as expected and may have led to overfitting.

Additionally, the tuned model's MAE (0.6796) is slightly higher than the baseline (0.6768), indicating that the absolute prediction error has also increased. With this MAE being close to the baseline value means the model isn't significantly worse, but it also doesn't show improvement.

Therefore, in this case, tuning with GridSearchCV didn't improve our model since the tuned model performed slightly worse than the baseline model, which suggests that the original hyperparameters might have been closer to optimal, or the tuning process overfit the training data.

OPTIMIZATION II

Best RMSE Params: {'n_factors': 10, 'n_epochs': 20, 'lr_all': 0.005, 'reg_all': 0.02}

Best RMSE Score: 0.8761547524492181

RMSE: 0.8838

MAE: 0.6741

Optimization II RMSE: 0.8837715035039327

Optimization II MAE: 0.6741281347847476

The RMSE from RandomizedSearchCV is 0.8821, which is lower than the RMSE from GridSearchCV, which was 0.8911.

A lower RMSE indicates that the predictions from the model are closer to the actual ratings on average.

The MAE from RandomizedSearchCV is 0.6742, slightly better than the MAE from GridSearchCV, which was 0.6795. This indicates a smaller average error in absolute terms when using RandomizedSearchCV.

So, there has been a slight improvement with the parameters found using RandomizedSearchCV compared to those from GridSearchCV. Both RMSE and MAE are slightly lower, which indicates that the model's predictions are more accurate this time. The small but noticeable improvement suggests that the optimized hyperparameters found by RandomizedSearchCV are better suited for the dataset and task at hand than those found by GridSearchCV.

GENERATING TOP 5 RECOMMENDED MOVIES

A sample user_id is specified and the top-5 recommendations for that specific user are generated using the latest trained optimized model. The recommendations are in terms of movie IDs.With this fuction, we're able to replace the user_id with any user id inorder to generate personalized recommendations for that user.

Best RMSE Params: {'n_factors': 10, 'n_epochs': 20, 'lr_all': 0.005, 'reg_all': 0.02}

Best RMSE Score: 0.8761547524492181

Top 5 Recommended Movies for User 1: [871, 660, 586, 896, 4673]

From the output above, based on user_id 1's past ratings, we would recommend the listed movie ids and since we have access to the original dataset, we can still be able to find out the movies' title and genre.

Top 5 Genres:

- **Comedy**
- **Children**
- **Drama**
- **Action**
- **Crime**

8. Conclusion

Based on the analysis and modeling conducted on the MovieLens dataset, we've gained valuable insights into user preferences, rating behaviors, and the effectiveness of recommendation models. Our exploratory data analysis highlighted key patterns in user engagement, genre preferences, and rating distributions. The recommendation system, optimized using collaborative filtering techniques, achieved a promising RMSE score indicating a reasonable level of accuracy. These findings provide actionable strategies to enhance movie recommendations, user experience, and business growth. Future improvements can further optimize the system to address challenges such as cold start problems and evolving user preferences.

9. Recommendations

1**. Personalized promotion strategies:**

Focus marketing efforts on movies consistently rated above 4 and promote highly rated niche genres to attract dedicated audiences.

2. Genre-based content expansion:

Increase content acquisition and production in high-demand genres like drama, comedy, and action to align with user preferences.

3. User segmentation for targeted engagement:

Develop loyalty programs and premium recommendation services for highly active users who provide more stable and valuable feedback.

4. Peak engagement planning:

Leverage historical rating trends to schedule content releases and promotions during peak engagement periods to maximize impact.

5. Differentiated recommendation strategies:

Offer separate recommendation lists for "popular" and "high-quality" movies to cater to diverse user preferences and expectations.

6. Cold-start mitigation:

Introduce content-based filtering or hybrid models to enhance recommendations for new users with limited rating history.

10.Next steps

- Build a hybrid recommendation model: Combine collaborative and content-based filtering to improve accuracy and address limitations of individual approaches.

- Expand feature engineering: Incorporate additional user and movie features, such as demographics, review text, and browsing history, to enrich the recommendation model.

- Deploy and monitor the system: Implement the model in a real-world environment, track its performance over time, and continuously refine it based on user feedback and new data.

- Collect more data that is also diverse: Collect additional data to improve the model robustness.

REFERENCES

- **Dataset Source:** Provided by GroupLens research lab, University of Minnesota.
- **Tools:** Python (pandas, scikit-learn, surprise, matplotlib, seaborn), Jupyter Notebook.
- **Documentation:** Surprise Library Documentation, Scikit-learn User Guide, GroupLens Dataset Guide.

This report serves as a comprehensive overview of the MovieLens recommendation system project, providing insights and methodologies to enhance personalized movie recommendations.