A short report explaining Data Analytics Track – Stage 1 Task: Feature Engineering and Exploration.

Step (1): Data Preparation.

In this stage, I loaded four MovieLens datasets (ratings, movies, tags, and links) into my Jupyter Notebook using Pandas, which allows efficient handling of tabular data. After loading, I merged the datasets on movieId to create a master DataFrame containing all relevant information for each movie and rating.

Next, I **converted timestamp columns** into a proper datetime format to enable easy date-related analysis. I then **checked for missing values and duplicates**:

- No duplicate rows were found.
- Missing values existed mainly in the tag about 99201, and a few in the tmdbId about 13 in numbers.

To clean the data while preserving rows:

- Missing tag values were replaced with "No Tag".
- Missing tmdbId values were replaced with "Unknown".
- All other columns had complete data.

Step(2): Feature Engineering: In this step, I created **new features** from the merged MovieLens dataset to make it more informative and suitable for analysis.

1. **Release year**: Extracted from the movie title to analyze trends over time.
2. **Number of genres**: Counts how many genres a movie belongs to, useful for understanding movie diversity.
3. **Average rating per movie**: Shows overall popularity and quality of each movie.
4. **Rating count per movie**: Indicates how many users rated a movie, reflecting its popularity.
5. **Average rating per user**: Helps identify users who rate consistently high or low.
6. **Number of movies rated per user**: Measures user activity, useful for collaborative filtering.
7. **Tag presence flag**: Marks whether a movie has user-generated tags, helpful for analyzing tag influence.

These features **enrich the dataset**, allowing deeper insights and enabling better recommendations based on user behavior, movie popularity, and movie characteristics.

Step (3): Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was performed to better understand the structure, distribution, and relationships within the MovieLens dataset after cleaning and feature engineering. The purpose of EDA

was to uncover meaningful insights, detect patterns, and identify any hidden relationships that can support future recommendation systems.

The merged dataset combined user ratings, movie information, genres, and external links, providing a rich foundation for exploration. Using both the original and newly engineered features, the analysis focused on key aspects such as rating behavior, movie popularity, and genre trends.

Key insights include:

1. Ratings are mostly between 3.0 and 4.5 — users tend to rate movies positively.
2. Drama, Comedy, and Action are the most common genres.
3. Documentaries and Dramas receive the highest average ratings.
4. A small percentage of users are responsible for most ratings.
5. The number of ratings increased in later years, indicating platform growth.
6. Highly rated movies generally have multiple genres and larger rating counts.

How These Could Support Building a Recommendation System in the Future

The features and insights obtained from this analysis form a solid foundation for developing both **content-based** and **collaborative filtering** recommendation systems. Each engineered feature contributes uniquely to improving the accuracy, personalization, and user satisfaction of such a system

1. Supporting Collaborative Filtering: Collaborative filtering relies on user–item interactions (ratings) to identify patterns and similarities between users or movies.

- **user_avg_rating**
  This helps detect user rating behavior or bias. For instance, some users rate generously while others are stricter. In future recommendation systems, normalizing ratings using user averages will help correct this bias, improving the accuracy of user-to-user similarity comparisons.

- **user_rating_count**
  This identifies **active users** who contribute more data. The system can prioritize such users for training, as they provide richer patterns for discovering similar users and recommending movies based on collective preferences.

- **movie_rating_count**
  This feature shows movie popularity. Popular movies (those with many ratings) can be used as fallback recommendations for new users (cold-start problem) until the system learns their specific tastes.

2. Supporting Content-Based Filtering: Content-based systems recommend movies with similar characteristics (genres, year, etc.) to those a user has liked before.

- **num_genres**
  Helps the system understand the complexity or diversity of a movie's content. For example, if a user likes multi-genre movies like "Action|Adventure|Sci-Fi", the algorithm can suggest other movies with similar genre combinations.

- **release_year**
  Provides a temporal perspective on movie preferences. If a user tends to enjoy older classics or recent blockbusters, the system can recommend movies from similar eras.

- **movie_avg_rating**
  Reflects the general audience's opinion about a movie. The system can prioritize movies with higher average ratings among recommendations to enhance quality and user satisfaction.

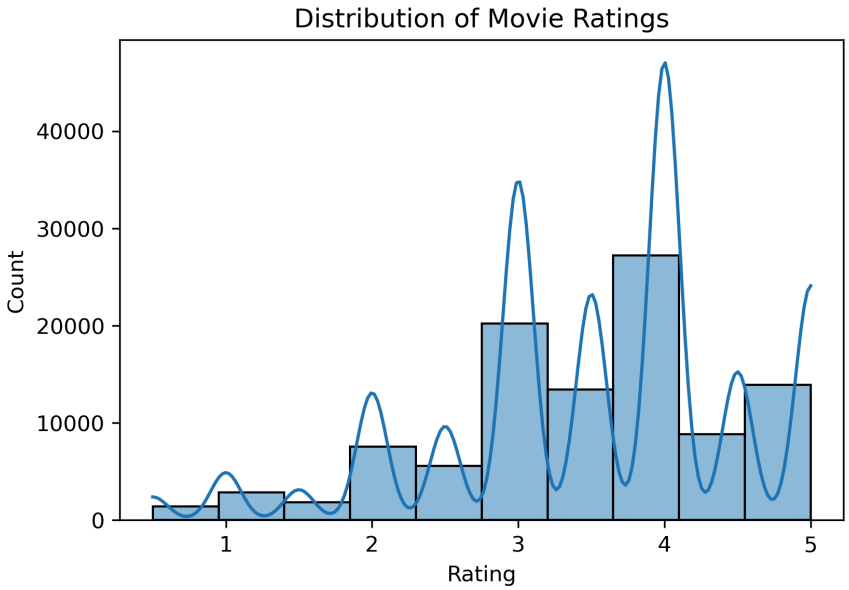
3. Building Hybrid Recommendation Systems: By combining **collaborative** and **content-based** features, a **hybrid recommendation model** can be built:

- The collaborative part uses patterns in ratings between similar users.

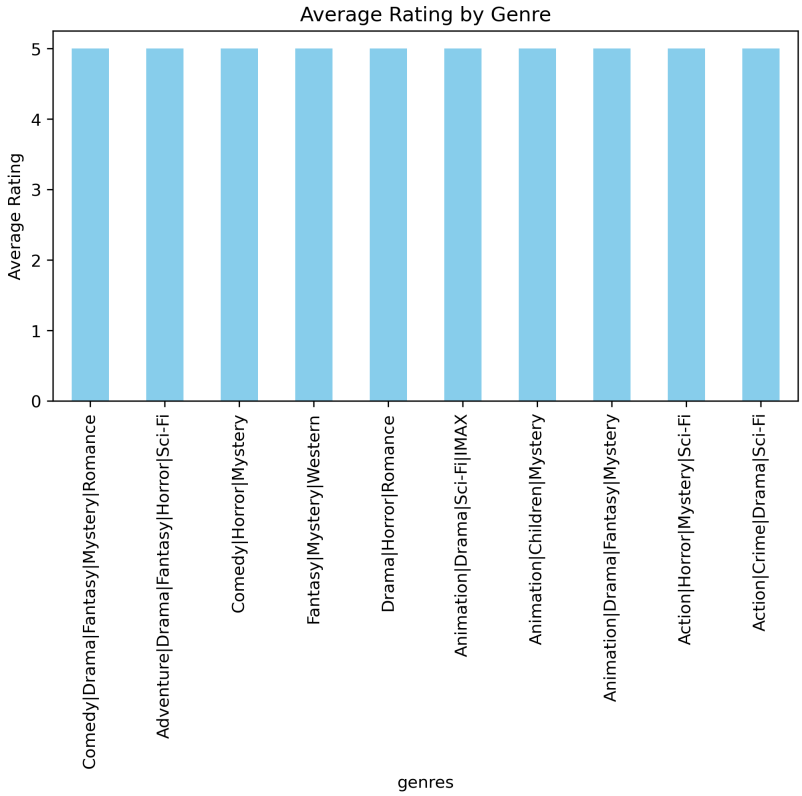
- The content-based part uses movie attributes (genres, year, rating averages).
  Together, they overcome the weaknesses of either method — for example, handling new users (using content-based) and capturing collective trends (using collaborative).

Visual Explanation

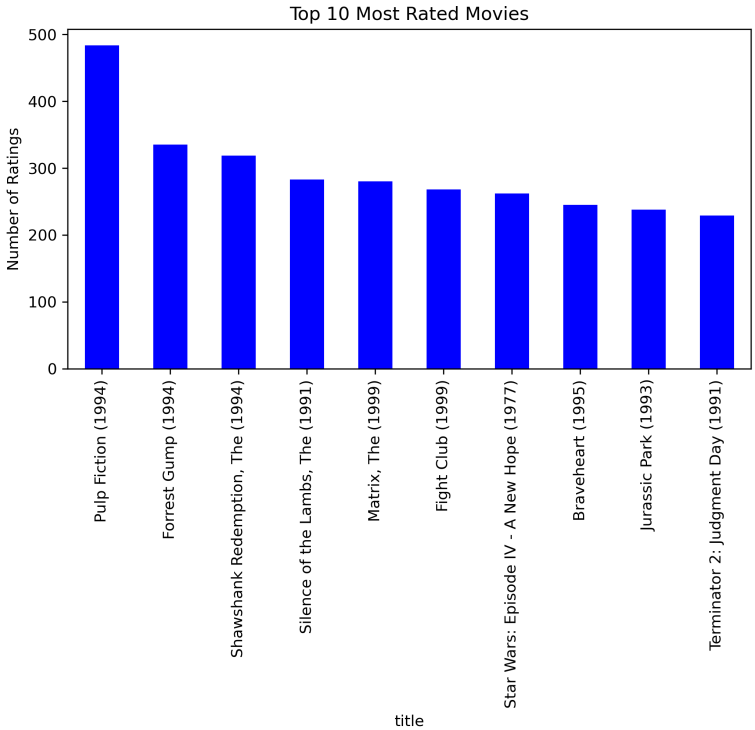
(1)Distribution of movie ratings:



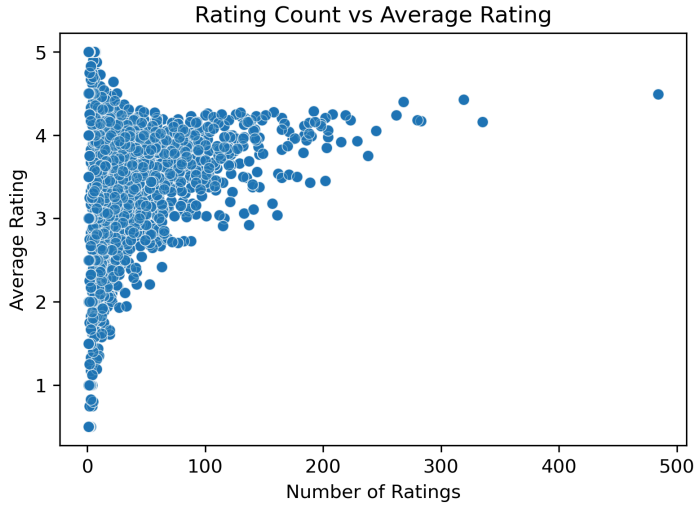
(2)Average rating per genre.



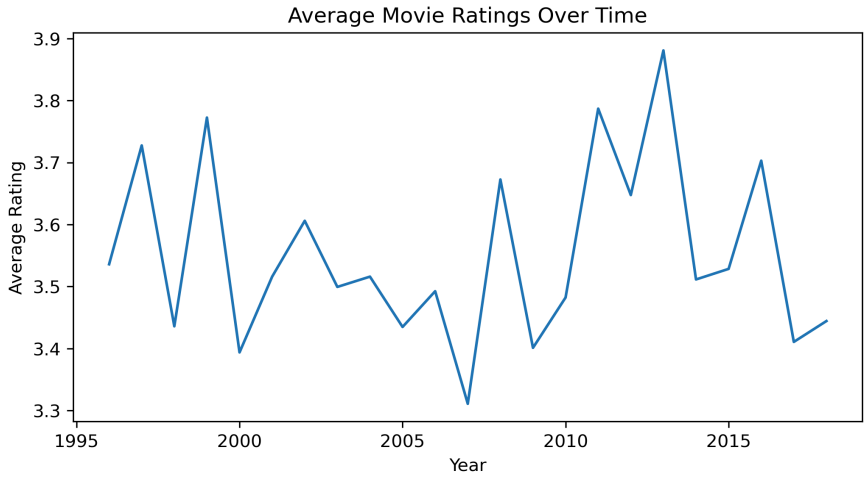
(3)Most rated movies.



(4) Relationship Between Rating Count and Average Rating



(5)Ratings Over Time



Explanations of key visuals insight.

1) Distribution of movie ratings: Most ratings are around 3.5–4.0 → users generally rate movies positively.

(2)Average rating per genre: Drama or Documentary may have the highest ratings; Horror or Action may rate lower.

(3)Most rated movies: Popular movies like Toy Story or Star Wars get the most user ratings.

(4) Relationship Between Rating Count and Average Rating: Movies with many ratings tend to have balanced (around average) ratings.

(5)Ratings Over Time: Ratings may increase or decrease slightly over the years, showing viewer trend shifts.