Data Analytics Report

Objective

The main goal of this task was to clean, explore, and add new features from the MovieLens dataset to uncover meaningful insights about user behaviour and movie performance. The idea was not just to analyse data for patterns, but to develop an understanding of how these features can help build an effective recommendation system. By the end of the project, I aimed to transform raw, scattered data into a structured dataset rich enough to power future machine learning or recommendation models. The dataset consists of multiple CSV files, including movies, ratings, tags, and links. I merged movies and ratings for my analysis using the movieId field to create a unified data frame that could be analysed comprehensively.

Data Preparation

The first step was to clean and merge the datasets properly. I loaded the four files, movies.csv, ratings.csv, tags.csv, and links.csv, and merged movies and ratings on the shared column movieId. This ensured that every movie rating was connected to its title, genre, and related data. Then, I checked for missing values and duplicates to make sure the dataset was consistent. The timestamps in the ratings dataset were converted from Unix time into human-readable datetime format so for easy understanding.
Some titles contained the release year within parentheses (for example, Toy Story (1995)). I used a regular expression to extract this year and stored it in a new column named release_year. After extracting the release years, I found a few missing entries (18), which were dropped to keep the dataset clean and consistent.

The resulting dataset was large but structured, containing 100,818 ratings from 610 unique users across 9,711 movies.

Feature Engineering

Feature engineering was another important part of this task because it adds depth and analytical power to the dataset. The goal was to create new variables that better describe

user behaviour and movie performance, which in turn could improve recommendation accuracy in the future.

The first feature I created was release_year, extracted from each movie's title. This allowed me to track temporal trends and study how preferences evolved over time. Using this, I computed movie_age, which measures how old a movie is compared to the current year (2025). This feature helps distinguish between classic and modern films and can be used to identify user preferences based on recency.

I also created num_genres, which counts the number of genres each movie falls under. Some movies belong to a single genre, like "Drama", while others belong to multiple, such as "Action|Adventure|Sci-Fi". This count turned out to be useful in identifying whether multi-genre films tend to receive higher ratings.

From the genre list, I extracted the main_genre, which refers to the first genre listed for a movie. This helps simplify analysis by grouping movies under their primary category (for example, "Action" in "Action|Comedy").

The most important engineered features were avg_movie_rating and num_ratings. These were computed by grouping the dataset by movie and calculating the mean and count of ratings, respectively. The average rating indicates how well a movie is perceived, while the count of ratings measures its popularity or level of engagement. Both metrics are fundamental in building a recommendation system, since they help determine which movies are both popular and well-liked.

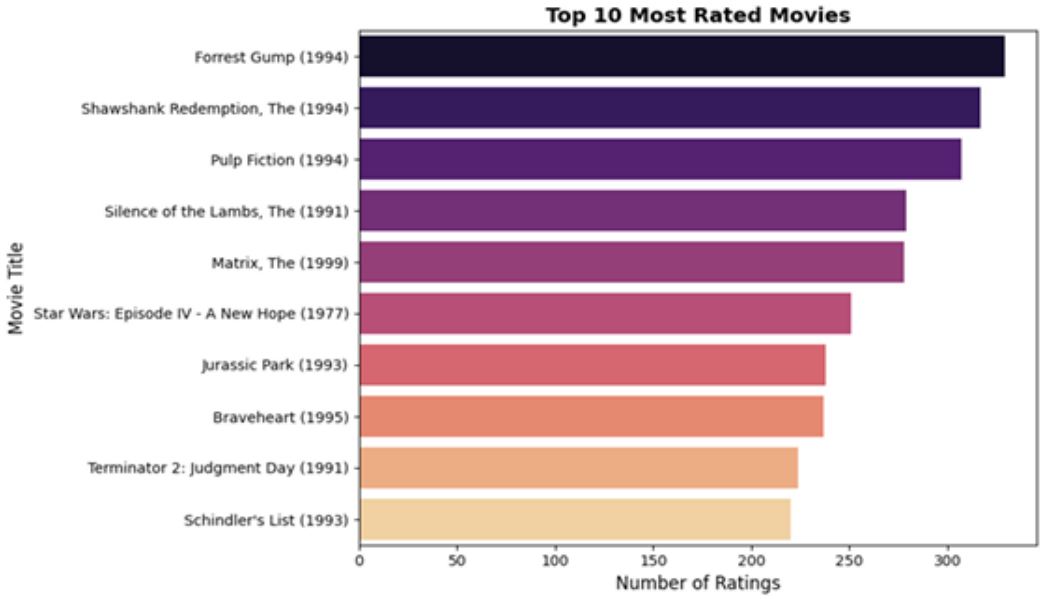
Finally, I created user_avg_rating, representing each user's personal average rating. This helps identify whether a user tends to rate movies generously or strictly. In future recommendation systems, this can help normalize ratings to account for individual biases.

By the end of this stage, the dataset contained meaningful new columns that revealed both movie-level and user-level behaviour.

Exploratory Data Analysis

With the dataset prepared, I performed exploratory data analysis (EDA) to identify patterns and generate insights. The distribution of ratings showed that most users tend to rate movies positively, with most ratings falling between 3.0 and 4.5. The average rating across

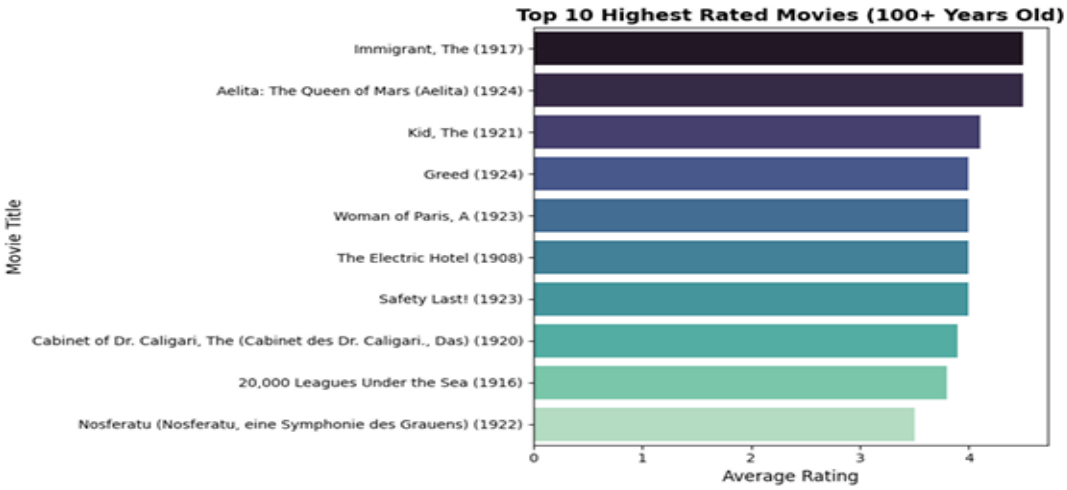
the dataset was 3.50, indicating that users generally enjoy the movies they watch. Extreme low ratings were rare, suggesting that users tend to rate movies they already expect to like.



When examining the most rated movies, a few titles stood out as audience favourites. Films such as Forrest Gump (329), The Shawshank Redemption (317), and Pulp Fiction (307) appeared at the top, each with hundreds of ratings. These popular titles are not only widely viewed but also form a stable foundation for recommendation systems since they have enough user feedback to be statistically reliable.

Next, I explored genres. The most common genres in the dataset were Action, Comedy, and Drama. These categories dominated the dataset, showing that users tend to watch and rate movies that fall under these familiar and mainstream genres. The bar chart representing genre counts reinforced how a few key genres drive the bulk of user engagement.

Then, I analysed the most recent movies, those released within the last 10 years. Among these, films that spanned more than three genres, such as The Girl with All the Gifts (2016) and Ice Age: The Great Egg-Scapade (2016), consistently had higher average ratings. This implies that modern audiences appreciate complex, genre-blending stories, and such insights could guide recommendation algorithms toward suggesting multi-genre titles for users with diverse preferences.



I also examined older movies by filtering those that were over 100 years old. Surprisingly, several of these classic films maintained strong average ratings above 4.0. Titles like The Immigrant (1917) and Aelita: Queen of Mars (1924) were among the top-rated classics. This suggests that even very old films can retain their cultural value and appeal to modern audiences. It also opens the possibility of recommending timeless classics to users with a taste for vintage cinema.

Finally, I focused on the latest release year in the dataset and calculated the average user ratings per genre for that year (2018). The top-performing genres were Documentary, Adventure, and Action, indicating that recent audiences are drawn toward storytelling that balances realism and excitement.

Key Insights

From these analyses, several strong insights emerged. First, movie popularity does not always align with high ratings. Some less popular movies had outstanding average ratings, while blockbuster hits often had more moderate scores. This distinction between “popular” and “highly rated” movies is important when designing recommendation systems to ensure a balance between mainstream and niche suggestions.

Second, classic movies continue to perform well even after a century, which indicates that older films should not be excluded from recommendation engines. Their inclusion could appeal to viewers who appreciate film history or classic storytelling styles.

Third, multi-genre movies tend to perform better, especially in recent years. This trend reflects changing audience preferences toward more dynamic narratives that combine elements from multiple categories.

Lastly, certain genres such as Action, Drama, and Comedy dominate user engagement, but emerging genres like Documentary and Sci-Fi are gaining traction. A recommendation engine that understands these evolving preferences can adapt more effectively over time.

Building a Recommendation System

The additional features and insights discovered in this project lay a solid foundation for building a movie recommendation system. The average movie rating and number of ratings provide a measure of both movie quality and reliability, which can be used as baseline scores in popularity-based recommendations.
The user average rating allows the model to account for individual rating tendencies, ensuring that strict or lenient users are normalized appropriately. The main_genre and number of genres support content-based filtering by allowing recommendations of similar films based on shared genre profiles. The movie age helps the system balance classic recommendations with more recent releases, making suggestions that align with user preferences for either timeless or trending content.

Overall, these features enable both collaborative filtering and content-based methods to work together, improving personalization, diversity, and relevance of recommendations.

Conclusion

This project provided hands-on experience in cleaning real-world data, designing new features, and performing exploratory analysis to uncover trends in user ratings and movie characteristics. The process transformed a raw dataset into a refined one rich with analytical value. Through careful cleaning, thoughtful feature engineering, and detailed visualization, it became clear that a well-prepared dataset can reveal deep insights about audience behaviour. The insights gathered here form the backbone for a future recommendation

engine, one that can not only suggest movies users are likely to enjoy but also introduce them to new genres, classics, and hidden gems. In summary, this project was an important step toward understanding the connection between data engineering and intelligent system design.