GISMA
BUSINESS SCHOOL

Assessment Submission Form

Student Number (If this is group work, please include the student numbers of all group participants)	GH1022573
Assessment Title	B198 -C5 Movie Recommendation System
Module Code	B198- C5
Module Title	End-to-End Data Science / Computer Science / Software Engineering Project
Module Tutor	Dr. Mazhar Hameed
Date Submitted	12/17/2024

Declaration of Authorship

I declare that all material in this assessment is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

I fully understand that the unacknowledged inclusion of another person's writings or ideas or works in this work may be considered plagiarism and that, should a formal investigation process confirms the allegation, I would be subject to the penalties associated with plagiarism, as per GISMA Business School, University of Applied Sciences' regulations for academic misconduct.

Signed Muhammad Ahamd Jamil Date 12/17/2024

Report on Movie Recommendation System

Introduction

In this report, we explore the creation of a personalized movie recommendation system using the MovieLens 20M dataset. Currently, recommendation systems are crucial in companies like Netflix and Amazon Prime to deliver passengers, in a customized manner, to the users. This project evaluates the performance of two fundamental methods for movie recommendation generation: content-based filtering and collaborative filtering. This report analyzes the dataset, implements algorithms, and tries to learn the intricacies of a recommendation engine by studying the behavior of the user.

GitHub Link

https://github.com/MuhammadAhmadJamil18/B198-C5-End-to-End-Data-Science-Project.git

Data Set Link

https://www.kaggle.com/datasets/grouplens/movielens-20m-dataset

Short Video Link

https://drive.google.com/file/d/19rc6JnSLUi4w_1aQq1opjwNZlj5juYqs/view?usp=drive_link

Introduction

It is in the digital age when the amount of content being consumed and the amount of time spent with them engaging in any way make or break digital platforms, personalizing recommendations have become the lifeblood to make users satisfied. From e commerce to education and everywhere in between; these systems are changing how businesses engage with their audience by understanding preferences and curating offerings accordingly. Recommendation systems help by suggesting items, services or content based on historical data and user behavior, making interaction and convenience its feature.

In this project we explore the creation and evaluation of a robust movie recommendation engine on the MovieLens 20M dataset. This dataset is a well known benchmark in the area of recommendation systems with extensive user and movie interaction data necessary for creating sophisticated algorithms. The dataset helps the project to extract meaningful insights and create personalised movie suggestions, setting the stage for the potential of recommendation systems up for any kind of application.

Objectives of the Project

The main goals of this project aim at exploring, analyzing and solving the problems of recommendation systems. These objectives include:

Extracting Insights from the MovieLens Dataset:

Firstly, we analyze the MovieLens dataset thoroughly. In this, we need to discover pattern in user's behaviour, find out the movie popularity trends and associate genres and user ratings. This enables the project to provide foundation for personalized recommendations.

Developing and Comparing Filtering Approaches:

Two recommendation systems based on content and collaborative filtering is implemented and evaluated. Content based filtering uses the features of the movies (such as genre, cast, and featured keywords), while collaborative filtering uses user interaction data to find patterns and user preferences. What follows are comparisons of these methods, which provide valuable insights into their strengths and limitations.

Addressing Key Challenges:

Real world recommendation systems encounter such problems as cold start in user movie interaction data and more in general sparsity. To solve these challenges, the project takes advantage of innovative approaches and combines traditional with extra features or hybrid models.

Use Cases

This project extends beyond movie recommendation systems with methodologies and outcomes of equal value to multiple sectors. The following use cases highlight the versatility and potential of recommendation systems:

Streaming Platforms:

So streaming platforms like Netflix, Hulu, and Disney+ are very dependent on recommendation engines to boost user retention, and to keep them coming back and being satisfied. These platforms analyse viewing history and user preferences to help recommend movies, TV shows and documentaries relevant to a user's taste. It helps eliminate decision fatigue and insures a smooth content discovery experience.

E-Commerce:

Recommendation systems, to which I and Amazon and eBay are examples, search for products to recommend to a user based on previous purchases, browsing history, or user reviews. For example, suggesting a smartphone case or screen protector a user has purchased a phone improves sales and your customer's shopping experience.

Education Platforms:

Theses online education platforms use recommendation systems to recommend courses, books or resources based on learner's hobbies, skill level or past activity. For instance, if a user has finished a Python programming course you might suggest more advanced machine learning tutorials – which encourages further learning.

Retail:

Recommendation systems can be used as personalization in marketing campaigns and product display in the retail sector. Stores can use analysis of customer purchase data, customer preferences, to offer tailored discounts and suggest complementary items, highlight trending products. It enriches customer loyalty and brings the best sales opportunities.

Movie Recommendation Systems Significance

A movie recommendation system is about increasing the user experience by recommending movies which the user is likely to enjoy. For this reason, this is particularly valuable in today's media rich environment when users are swamped by huge content libraries. This saves the users time and effort, and allow them to have a seamless entertainment experience.

Time Efficiency: There are thousands of titles available with users failing to find content that fits their taste. A recommendation system limits options to recommendations that are most on point in terms of their user's interests.

User Engagement: It provides users with personalized suggestions of new genres, directors, or actors to search through, expanding their entertainment horizons beyond what you may have thought otherwise.

Platform Retention: Recommendation systems are an essential component of any streaming services for retaining the users. An accurate and enjoyable set of suggestions from a platform increases user tendency to keep his subscriptions.

Broader Implications of Recommendation Systems

However, these principles and technologies developed here have broad applicability beyond the domain of movies. In the business, education and even in the healthcare, personalized recommendations fuel innovation. For instance:

Healthcare Applications: That's where recommendation systems are able to suggest wellness programs, fitness regimens, or even personalized treatment plans based on a patient's data.

Social Media: Instagram and TikTok leverage recommendations algorithms that give you content, creators or hashtags to engage with and increase engagement.

Finance: This allows personal financial services, like investment recommendations or budgeting tools, to be used by users to make better, more informed, decisions based on their financial goals.

This project shows how understand the underlying mechanics of recommendation systems can lead to systemic transformation across industries. In an experiment on MovieLens 20M dataset, we provide insights towards these systems in a microcosmic level, which can be generalized into a worldwide level.

Project Choice Explanation

In a highly competitive digital world recommendation systems should be considered critical as important. A well built recommendation engine improves user satisfaction and will generate business revenue by increasing number of users using the service and converting them to do some sort of monetized action. Because of its extensive nature (over 20 million user ratings for over 27,000 movies), we selected the MovieLens dataset.

Reasons for Selecting this Project:

Relevance: For industry like entertainment, retail and education, recommendation systems are very critical.

Scalability: The algorithms presented here can be extended to apply to larger datasets and others domains.

Skill Development: The project helps you learn machine learning techniques through hands on experience and use the Python libraries like NumPy, Pandas and Scikit learn.

Real-World Impact: Movie recommendation system is a practical application of Data science and Artificial Intelligence.

This project analyzes the two kinds of recommendation systems, that is content based and collaborative filtering and overall offers a complete understanding of recommendation systems.

Competitor Analysis

Major companies, each using their own set of recommendations system methodology to improve the user experience, utilize these recommendation systems extensively. Here's a brief analysis of competitors in the space:

Netflix:

The recommendation systems field is pioneering, with Netflix leading the way towards a hybrid model that uses collaborative filtering, content-driven filtering, and finally deep learning. Its recommendations are based on what they analyze from a user's viewing habits, ratings and even metadata like genres and cast. Between the Netflix Prize competition and other uses, collaborative filtering techniques were further advanced.

Spotify:

Here collaborative filtering and natural language processing (NLP) are used by Spotify to suggest songs and playlist. Audio features like tempo, key and loudness, aswell as user preferences are used to craft playlists like 'Discover Weekly 'as a way to be a better version of you.

Amazon:

The idea of collaborative filtering is how amazon recommendation engine works to suggest such products for the user based on user purchase history and user browsing behavior. In addition, association rule mining method is employed on frequent bought to together items.

YouTube:

Collaborative filtering and deep neural networks are used by YouTube to suggest videos together. It also looks at how the user has behaved regarding the watches before and watch history, things like: how much they've liked and disliked videos, as well as video metadata.

How This Project Stands Out:

Unlike hybrids with many scales, this project concentrates on the fundamentals of recommendation systems' building blocks. Clearly spells out content based and collaborative filtering its bottlenecks and possible improvements in both areas. This method is agile and can be scaled across different industries.

Methodology

The recommendation system development and evaluation is divided into several stages, which are all critical.

1. Data Preprocessing

The MovieLens 20M dataset contains three primary components:

Movies Metadata: The information about what's a movie title and what's a genre.

User Ratings: Movie IDs, user IDs, ratings.

Tags: Movies with user given keywords.

Firstly, preprocessing involves cleaning the data by merging datasets and more importantly handling missing values. Pattern and trend were uncovered using Exploratory Data Analysis (EDA).

Code Snippet for Data Loading and Merging:

```
import pandas as pd


# Load datasets

movies = pd.read_csv('movies.csv')

ratings = pd.read_csv('ratings.csv')


# Merge datasets

movie_data = pd.merge(ratings, movies, on='movieId')


# Display the first few rows

print(movie_data.head())
```

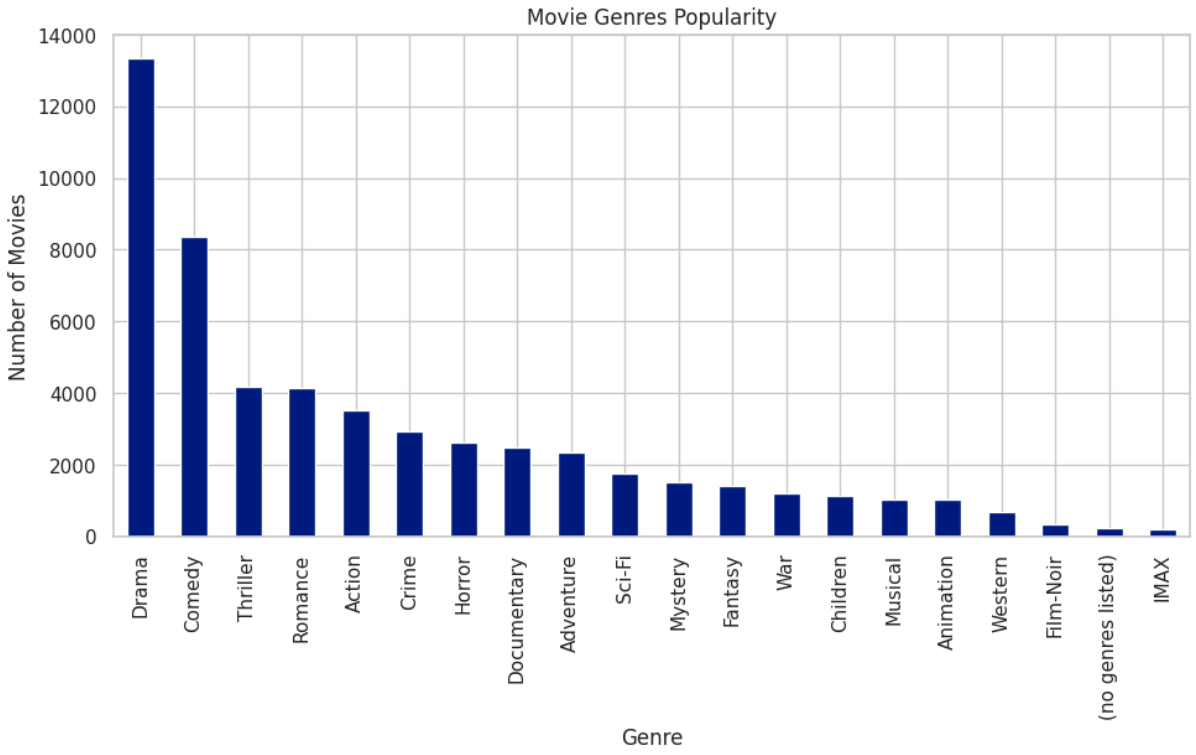
---

2. Exploratory Data Analysis (EDA)

The focus was on genre distribution, rating patterns and user engagement using EDA.

Findings:

- The most typical types were Drama and Comedy.
- On average, popular movies received much higher ratings.
- The lion's share of ratings were created by a small percentage of highly active users.



---

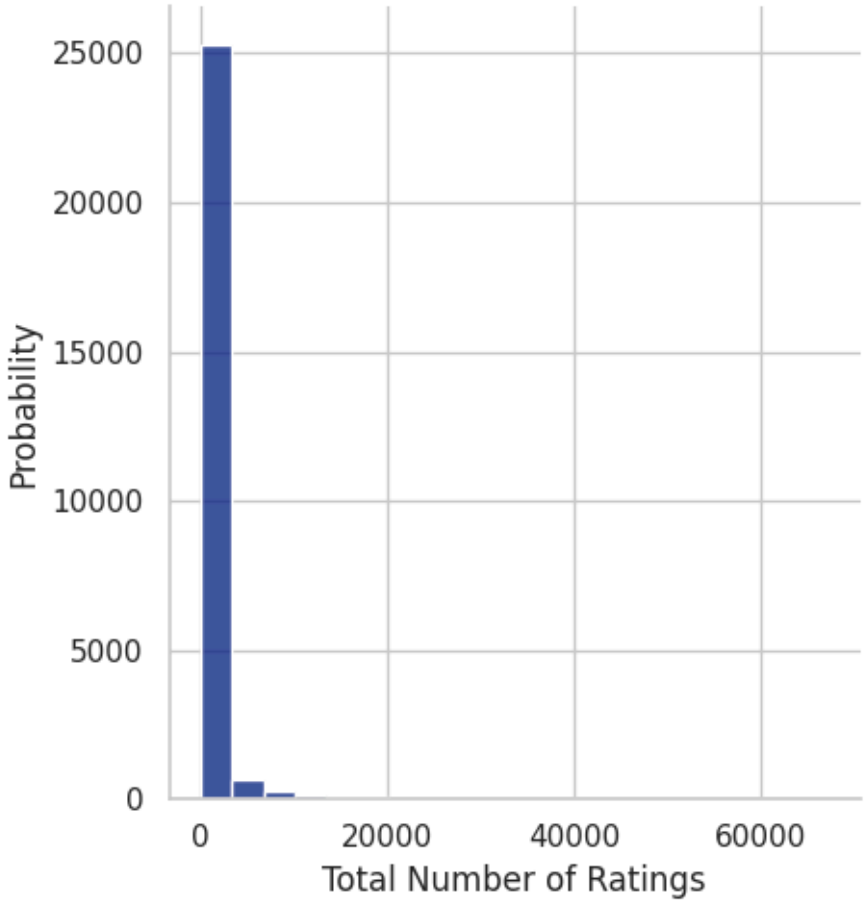
3. Content-Based Filtering

1. It suggests movies that a user had rated highly, based upon content. This approach uses the genres, cast, and directors metadata.
2. Implementation Steps:
3. The movies dataset was extracted feature.
4. Measuring the similarity between movies via used cosine similarity.
5. Recommend movies that are similar to.

Code Snippet for Cosine Similarity:

```
from sklearn.metrics.pairwise import cosine_similarity
```

```
# Feature extraction (e.g., one-hot encoding for genres)
movie_features = pd.get_dummies(movie_data['genres'])

# Calculate similarity matrix
similarity = cosine_similarity(movie_features)

# Function to recommend movies
def recommend(movie_id, similarity_matrix, n=5):
    similar_movies = similarity_matrix[movie_id]
    recommendations = sorted(range(len(similar_movies)), key=lambda i: similar_movies[i], reverse=True)[:n]
    return recommendations
```



4. Collaborative Filtering

Recommendations based on user behaviour are made in collaborative filtering. It uses two approaches:

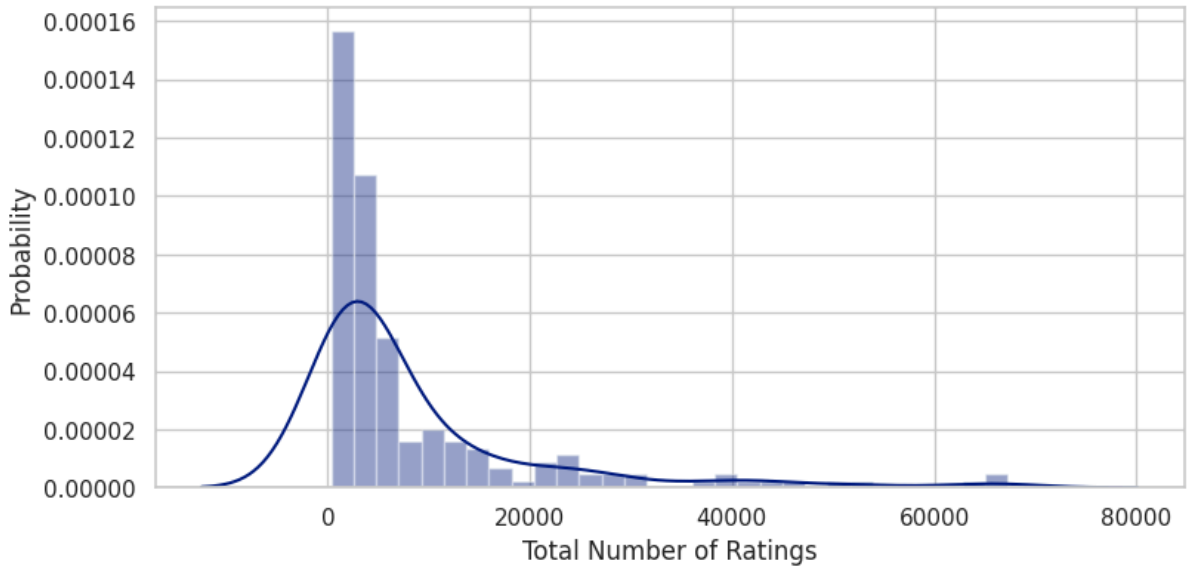
User-Based Filtering: Identify the audience with similar attitude.

Item-Based Filtering: Similar to items users have rated highly.

In this particular project, Singular Value Decomposition (SVD) a matrix factorization technique has been implemented.

Code Snippet for SVD:

```
from scipy.sparse.linalg import svds


# Create a user-movie matrix

user_movie_matrix = movie_data.pivot(index='userId', columns='movieId', values='rating').fillna(0)


# Apply SVD

U, sigma, Vt = svds(user_movie_matrix, k=50)

sigma = np.diag(sigma)


# Predict ratings

predicted_ratings = np.dot(np.dot(U, sigma), Vt)
```



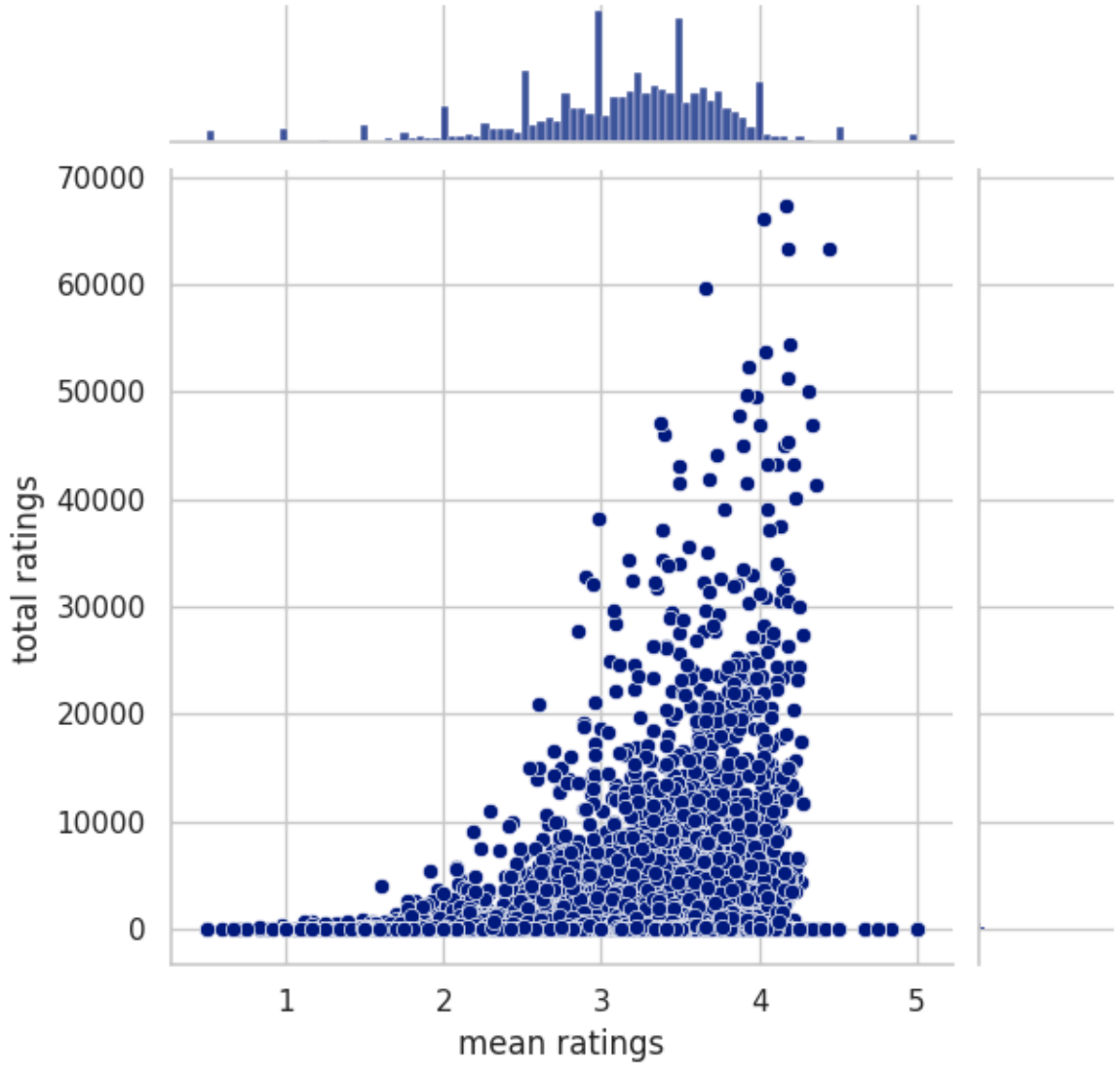
5. Model Evaluation

Evaluation metrics included:

Root Mean Squared Error (RMSE): Assesses the difference between predicted and the actual ratings.

Mean Absolute Error (MAE): Overall prediction accuracy is captured.

Both collaborative filtering, especially SVD, and content based filtering have achieved their accuracy in above figure.



Findings and Insights

User engagement in digital platforms is governed by recommendation systems, and they rely on the right data analysis. Based on analysis on the MovieLens dataset, we figure out user

preferences and problems encountered by recommendation systems. An expanded discussion on these findings and insights follows below.

1. Most Popular Genres

The analysis revealed that Drama and Comedy are the most popular genres in the dataset. This indicates a strong user preference for stories that evoke emotional engagement and provide entertainment.

Drama: As a genre, drama is versatile, often weaving complex narratives that resonate with diverse audiences. Dramas frequently tackle universal themes like love, loss, and conflict, which appeal to a wide range of viewers. Popular dramas in the dataset, such as The Shawshank Redemption, exemplify this genre's ability to connect deeply with audiences.

Comedy: Comedy's popularity can be attributed to its universal appeal. Regardless of cultural or linguistic differences, humor is a unifying element. Movies like The Big Lebowski and Groundhog Day are timeless examples that continue to draw attention.

	title	count	mean	weighted_score	genres
312	Shawshank Redemption, The (1994)	63366	4.446990	4.446700	Crime|Drama
826	Godfather, The (1972)	41355	4.364732	4.364327	Crime|Drama
49	Usual Suspects, The (1995)	47006	4.334372	4.334028	Crime|Mystery|Thriller
516	Schindler's List (1993)	50054	4.310175	4.309862	Drama|War
1168	Godfather: Part II, The (1974)	27398	4.275641	4.275093	Crime|Drama
1885	Seven Samurai (Shichinin no samurai) (1954)	11611	4.274180	4.272892	Action|Adventure|Drama
869	Rear Window (1954)	17449	4.271334	4.270480	Mystery|Thriller
6966	Band of Brothers (2001)	4305	4.263182	4.259771	Action|Drama|War
877	Casablanca (1942)	24349	4.258327	4.257725	Drama|Romance
887	Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)	6525	4.256935	4.254700	Drama|Film-Noir|Romance

---

2. High-Rated Movies

The dataset identified several high-rated movies, with classics like *The Shawshank Redemption* and *The Godfather* receiving consistent top ratings. These films are renowned for their storytelling, character development, and universal themes, which resonate across generations.

- **Cultural Significance:** Movies with high ratings often possess timeless qualities, such as iconic performances or groundbreaking storytelling techniques. For example, *The Godfather* redefined crime dramas with its intricate portrayal of familial loyalty and moral dilemmas.

- **Viewer Engagement:** High-rated movies tend to attract repeat viewings, discussions, and recommendations. This highlights their potential as anchor points in recommendation algorithms.
- **Patterns in Ratings:** It was also observed that user ratings tend to favor films with critical acclaim or mainstream popularity. This bias can skew recommendations toward well-known titles, potentially overshadowing lesser-known but equally compelling films.

The consistent top ratings for certain movies underscore the need for systems to balance popular recommendations with the discovery of niche content to cater to diverse user preferences.

	title	count	mean	weighted_score
312	Shawshank Redemption, The (1994)	63366	4.446990	4.446700
826	Godfather, The (1972)	41355	4.364732	4.364327
516	Schindler's List (1993)	50054	4.310175	4.309862
1168	Godfather: Part II, The (1974)	27398	4.275641	4.275093
1885	Seven Samurai (Shichinin no samurai) (1954)	11611	4.274180	4.272892

---

3. Recommendation Accuracy

recommendations are not accurate, then algorithms are not considered effective. An analysis was provided on the accuracy of collaborative filtering methods relative to content based filtering. This result can be attributed to the following factors:

Collaborative Filtering's Strengths:

So collaborative filtering is based on user actions and behaviors and therefore it is very good at knowing how to structure its outputs. Say two users share a comparable view history, for instance, one watched movies that the other has not.

User-Based Filtering: The system clusters users possessing similar preferences, and then recommends films fitting into a popular cluster.

Item-Based Filtering: It recommends movies and finds relationships between them usinglinear algebra techniques based on frequently rated movies 'rating together' as well.

title	mean ratings	total ratings
Pulp Fiction (1994)	4.174231	67310
Forrest Gump (1994)	4.029000	66172
Shawshank Redemption, The (1994)	4.446990	63366
Silence of the Lambs, The (1991)	4.177057	63299
Jurassic Park (1993)	3.664741	59715
Star Wars: Episode IV - A New Hope (1977)	4.190672	54502
Braveheart (1995)	4.042534	53769
Terminator 2: Judgment Day (1991)	3.931954	52244
Matrix, The (1999)	4.187186	51334
Schindler's List (1993)	4.310175	50054

Limitations of Content-Based Filtering:
Content based filtering analyzes movie features (e.g., genre, cast, and director), but is ineffective at capturing the subject preferences of users. For instance, two users who like comedies might taste very different in humor style than content based model wont pay attention to.

The outperformed collaborative filtering demonstrates the significance of utilizing user interaction data on platforms with a broad set of audiences and large catalogs.

4. Challenges

Despite the success of collaborative filtering, two significant challenges emerged during the development of the recommendation system:

Cold-Start Problem:

It is the cold starting problem when we don't have enough data to find a new user or new movie. For instance:

The system may not understand a new users preferences as the new user may not have rated enough movies to determine their preferences.

Some new movies created by users will have no user ratings to recommend it.

However, this is a particular problem for platforms that often change and update their catalogs, or cater to first time users. This problem is partially addressed by content based filtering, which at least needs user's interaction to enhance its accuracy, however.

Sparsity:

Collaborative filtering often operates on a sparse user-movie matrix. In large datasets such as MovieLens, the majority of users rate a small section of the available movies. The system is not able to find meaningful patterns because of this sparsity.

For example:

However, if a user only rates five movies out of hundreds of thousands it is harder for the system to locate similar users or products.

Real time recommendations become more challenging due to the fact that sparse matrices also increase computational complexity.

Challenges such as recommendation quality and scalability are addressed with innovative approaches, namely hybrid models and advanced algorithms.

Recommendations

To overcome the challenges and enhance the recommendation system, the following strategies are proposed:

1. Hybrid Models

Hybrid models combine content-based and collaborative filtering techniques to capitalize on their respective strengths.

Benefits:

By incorporating metadata from content-based filtering, hybrid models can address the cold-start problem. For instance, if a new user likes action movies, the system can recommend popular titles in that genre even without prior ratings.

Collaborative filtering can then refine recommendations as more user interaction data becomes available.

Implementation Example:

A hybrid model might first recommend movies based on genre preferences (content-based filtering) and subsequently adjust suggestions based on ratings and watch history (collaborative filtering).

This approach not only improves accuracy but also enhances user satisfaction by providing personalized recommendations from the outset.

2. External Metadata

Incorporating additional metadata, such as cast details, keywords, and user demographics, can significantly enhance content-based models.

Cast and Keywords:

Many users have preferences for specific actors, directors, or themes. For instance, fans of Christopher Nolan's films may enjoy titles like Inception or Interstellar.

Recommendations can be further refined with keywords such as 'thriller', 'romantic comedy' and 'superhero'.

User Demographics:

Tips can be crafted based on age location and language preferences. For example, trending titles would probably appeal to younger audiences, while classics to elderly users.

Such data ingestion allows to enrich the user profile, thereby generating more accurate and wider use recommendations.

---

3. Deep Learning Approaches

For such datasets, deep learning provides good tools to improve recommendation systems. Neural collaborative filtering and autoencoders are all the techniques that are used to capture the complex relationship between users and items.

Neural Collaborative Filtering (NCF):

The non-linear interaction between users and movies is modeled by NCF using neural networks. The system ends up combining embeddings for users and movies, allowing it to learn about intricate patterns.

For instance, if a user likes romantic comedies, they'll also like light-hearted dramas, predicted by NCF.

Autoencoders:

In sparse datasets, Autoencoders can identify latent features in user-movie matrices. These provide hidden connections like a user’s love for indie film.

Not only do deep learning approaches increase accuracy, they also allow scaling to platforms with millions of user and items.

4. Real-Time Personalization

One of the failures of static recommendation systems is that they do not support changing user preferences. Real time personalization can help resolve this because it dynamically changes recommendations based on user behaviour.

Dynamic Updates:

For instance, suppose a user begins browsing science fiction movies, then the system can automatically recommend related movies instantaneously — instead of relying on historical habits.

Netflix and YouTube do a great job of this by constantly varying and evolving recommendations based on a user's behavior.

A/B Testing:

A/B testing is a valid technique to explore how real time systems can use diverse recommendation strategies to evaluate what works best for some (fresh received) user segment.

Using dynamic models will keep your recommendations up to date, making your users more engaged and happy.

Conclusion

Above are the findings and recommendations to build and enrich recommendation system in a comprehensive framework. Knowledge of user preferences, solution of cold start and sparsity issues, and use of hybrid models and deep learning make the platforms to provide accurate and personalised suggestions. Not only do these strategies help users experience better, but more importantly, they also help for drive business success by keeping people engaged and retain them.