MOVIE RECOMMENDATION SYSTEM

Personalized Movie Recommendations Tailored to User's Viewing History

OCTOBER 2023

SAPTARSHI SYAM

Contents

1. Problem Statement

In the fiercely competitive landscape of online streaming services, including giants like Netflix, Amazon Prime, and Disney+, the battle to captivate and retain customers is intense. These platforms invest substantial efforts in luring users by curating movies and show recommendations that align precisely with individual viewing preferences. However, it's not just the streaming industry; other sectors like e-commerce and digital marketing also join the race, striving to deliver hyper-relevant content to their users. Across these diverse domains, the ultimate objective remains consistent: winning and keeping customers through finely tuned content recommendations tailored to their unique viewing or buying behaviors.

1.1 Benefits of a Movie Recommendation System:-

A movie recommendation system can benefit several industries, primarily those related to digital entertainment and content delivery. Here are some industries that can benefit:

- **Streaming Services**: Video streaming platforms like Netflix, Amazon Prime Video, and Disney+ can significantly benefit from movie recommendation systems to retain subscribers, increase user engagement, and boost content consumption.
- **TV Broadcasting**: Traditional television broadcasters and cable companies can use recommendation systems to enhance their on-demand and streaming services, improving viewer retention and advertising targeting.
- **E-commerce**: Online retailers can use similar recommendation algorithms to suggest movies and TV shows based on user interests, increasing customer engagement, and potentially leading to cross-selling opportunities.
- **Digital Marketing**: Marketers can employ recommendation systems to personalize content for online advertising, increasing the relevance of ads and improving click-through rates.
- **Content Production**: The entertainment industry, including film studios and production companies, can use recommendation systems to analyze user preferences and trends, aiding in content creation and distribution strategies.
- **Education**: Educational platforms can employ recommendation systems to suggest relevant educational videos and courses based on user preferences and learning history.
- **Research and Data Analysis**: Researchers and data analysts can utilize recommendation algorithms to discover patterns, trends, and insights in large datasets, beyond just movie recommendations.

In summary, the benefits of movie recommendation systems extend beyond the entertainment industry, influencing various sectors where personalization, user engagement, and content delivery play crucial roles in their success.

2. The Client

Our client is an online movie subscription service provider that caters a variety of movies across genres to its monthly or yearly subscribers. The subscribers seek a wide variety of content, user-friendly interfaces, and an enjoyable viewing experience. They want value for their subscription and expect the service to cater to their entertainment needs.

Our client is dedicated to enhancing its subscribers' experience by offering tailored content recommendations that align seamlessly with their viewing habits and preferences.

3. The Data

Our Data Source is "MovieLens" dataset which is a widely-used collection of data related to movie ratings and recommendations. The dataset contains 20 million ratings and 465,000 tag applications applied to 27,000 movies by approximately 138,000 users. It includes user ratings, movie metadata, and user-generated tags. We had 6 CSV Files as explained below:-

movies.csv:- This had the Movie id, Title and the Genre to which the movie belonged.

rating.csv:- This has the rating of all movies which the user has given along with their Timestamp.

genome_scores.csv:- This contains data related to the genome scores associated with movies. Specifically, it provides information about the relevance of various tags or attributes to different movies.

genome_tags.csv:- It contains information about the tags or attributes that are used to describe movies.

link.csv:- It provides links or references to external movie databases (IMDb and TMDb) for each movie in the dataset, allowing users to explore more details about the movies.

tag.csv:- This captures user-generated tags and the associated movies they were applied to.

In our project, we have leveraged the 'movies.csv' and 'ratings.csv' datasets as the cornerstone of our analysis, drawing upon these rich sources of information to derive profound insights and generate personalized user recommendations.

4. Data Wrangling

In this stage, we explored the Datasets which we would need for our Project and inspected it for finding the size of the Data, any missing values, outliers, type conversions to make it ready for further processing. Data Wrangling is an iterative process and the specific steps and techniques used may vary. The goal is to ensure that the data is accurate, complete, and properly structured for subsequent analysis.

i. **Data Inspection**: Explore the dataset's structure, inspect the first few rows, and check for the number of rows and columns to get a sense of the data's content and overall quality.

ii. **Handling Missing Values**: For our Project, the 'movie.csv' and 'rating.csv' are all we need to explore and build our recommendation system. We found no Null values in these two datasets.

iii. **Data Type Conversion**: Ensured that the data types of each column are appropriate for analysis. We extracted the Released Year from the Title of the movie dataset and used one hot encoding technique to extract the Genres of the movies.

iv. **Removing Redundant Columns**: We found dropped several columns as it had no relevance to our goal.

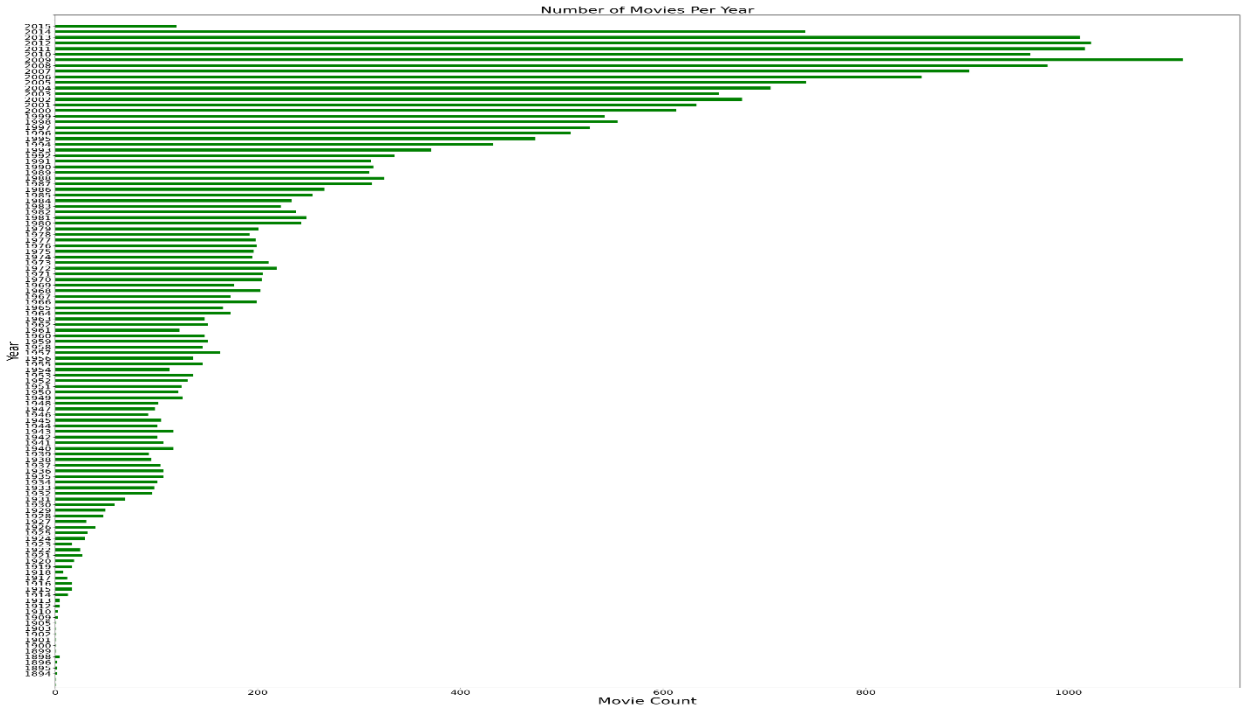
v. **Data Merge**: We merged the movie Dataset with rating dataset to come up with 1 Dataset which we will use for EDA and further stages.

4.1 Exploratory Data Analysis

We had a very rich source of Movie list which were released from 1894 till 2015. This gave us an opportunity to find various insights like the number of movie released per year, what was their genre and what are the most popular genre's of all the movie released. We explored few other as explained below:-

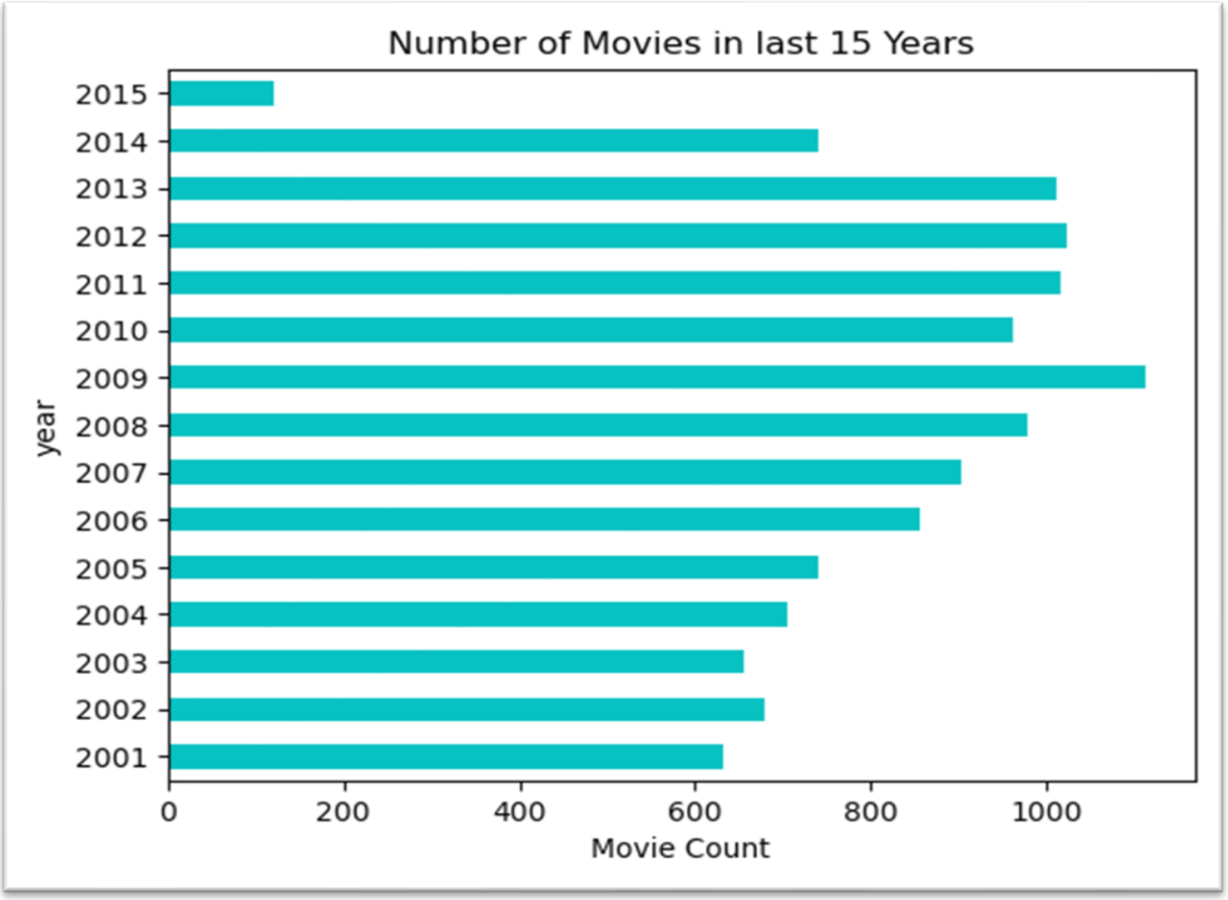
4.1.1 Finding the Number of Movies Released per Year

We found that the Number of Movie releases gradually increased until 2013.



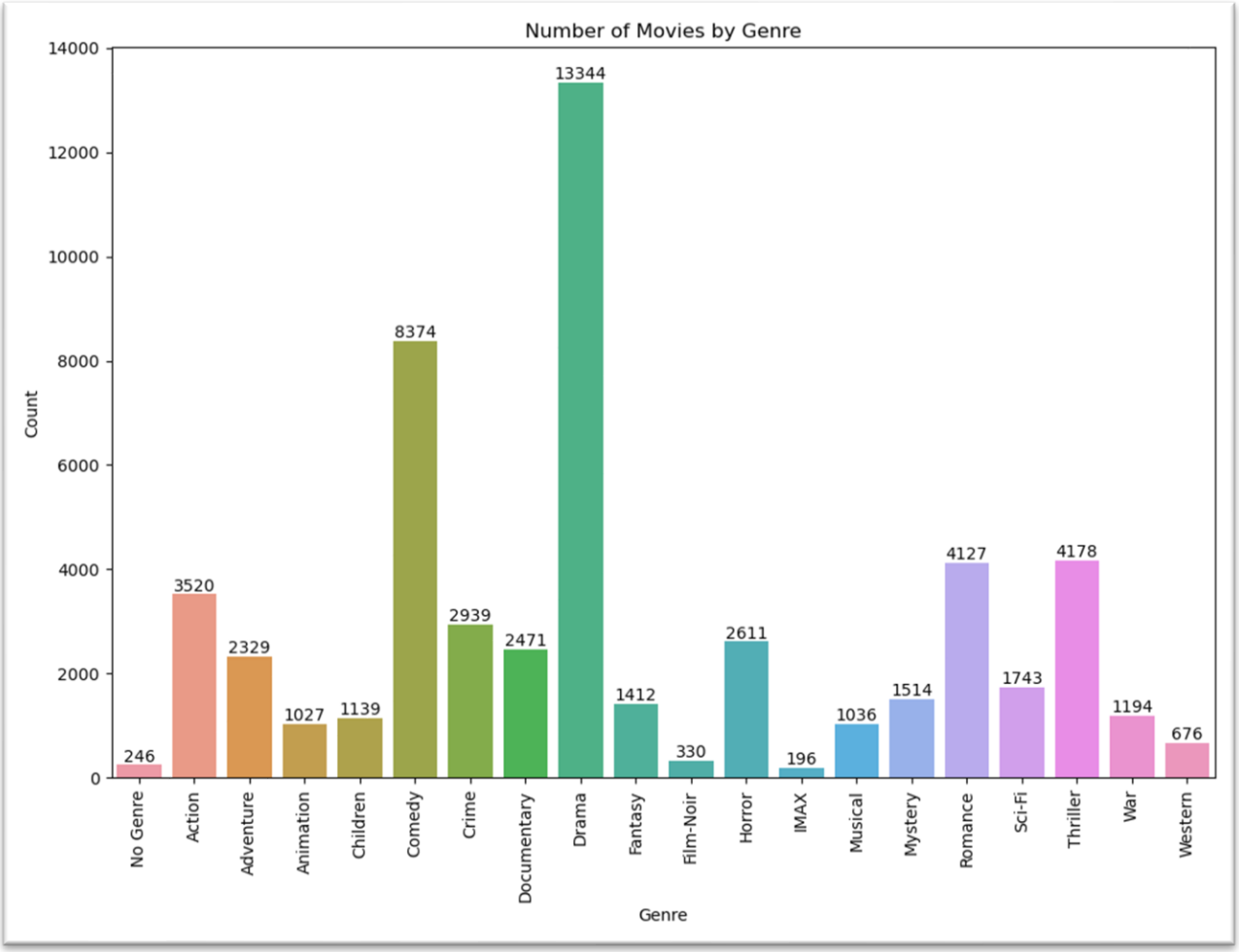
4.1.2 Number of Movies Releases in between 2000-2015

Our emphasis has been primarily on data from the past 15 years, as this timeframe boasts a substantially higher volume of user ratings in comparison to the preceding years.



4.1.3 Finding the Number of Genres of Movies released till date

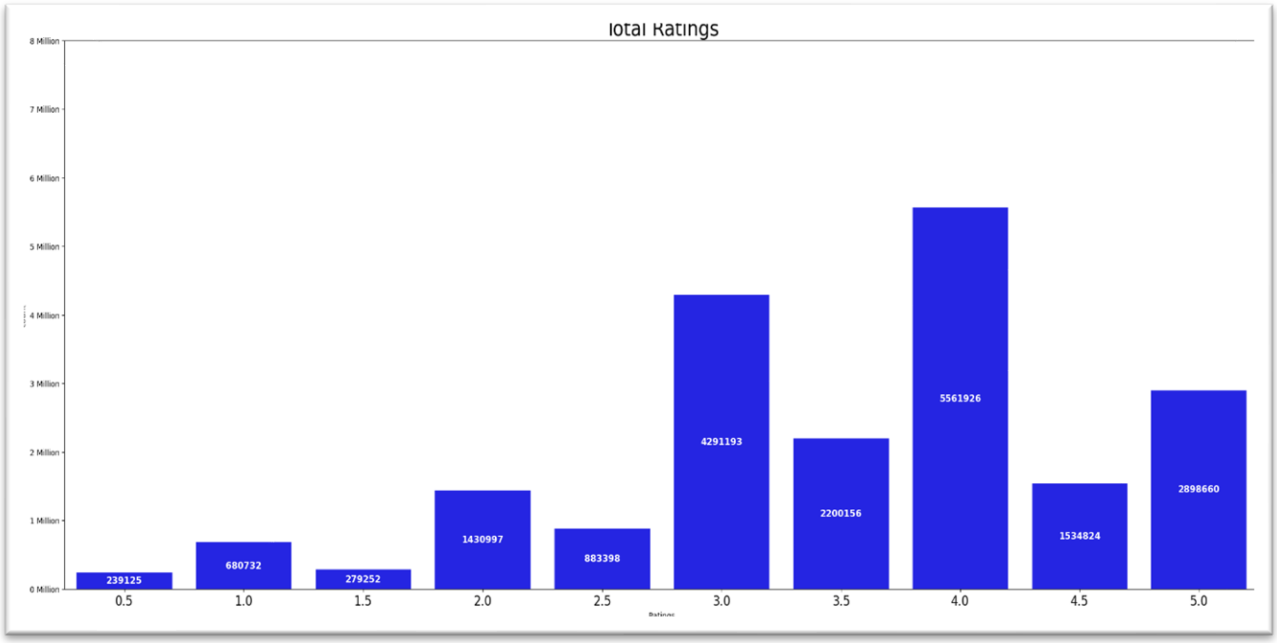
We explored the Total number of movie releases genre wise to see which Genre has most numbers of movies. This gives us an indication of movie choices by users.



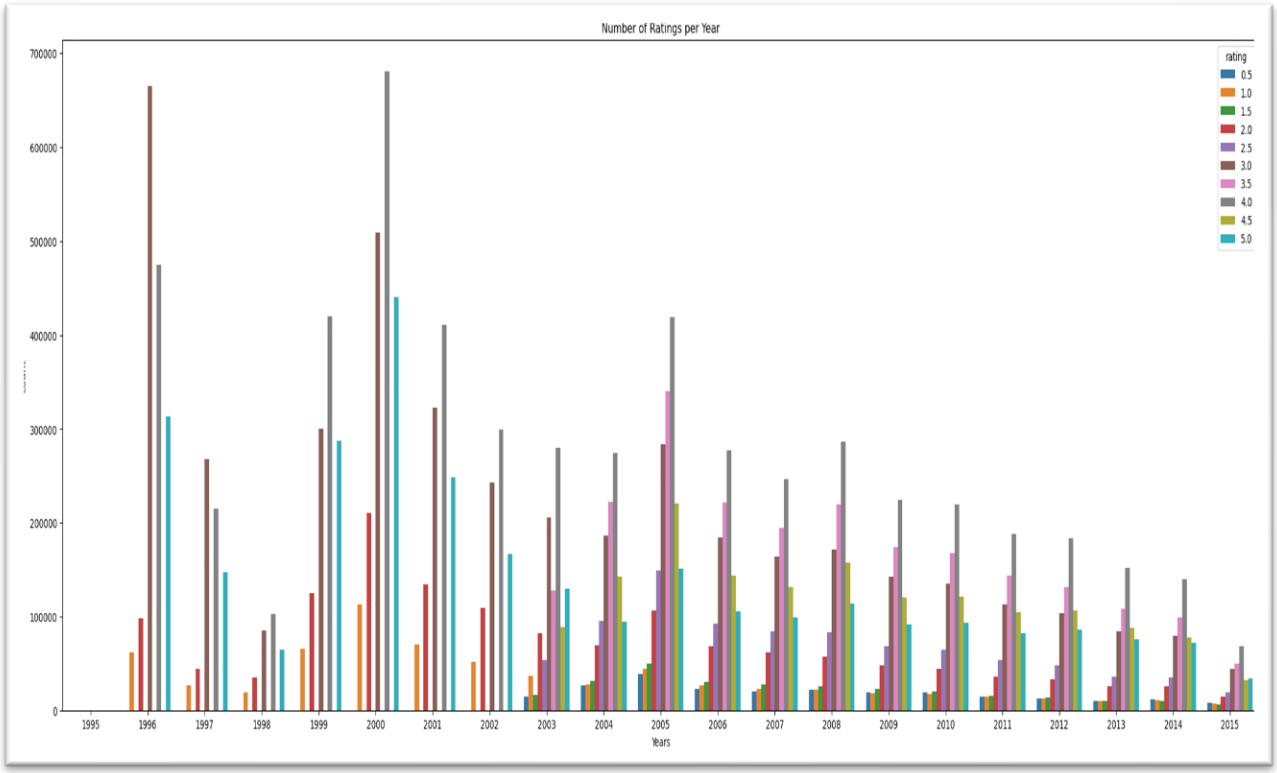
Drama emerges as the overwhelmingly popular genre, with a significant lead over Comedy. Following closely are Thrillers and Romantic Movies in the number of movies being made till 2015.

4.1.4 What is the Most Common Rating given by Users?

In our analysis of the Rating Dataset, we explored whether a dominant rating value emerged. Upon thorough examination of the data, it became apparent that most users leaned towards assigning a rating of 4, followed closely by 3 and 5. Additionally, we identified that the overall Average Rating across all movies stood at a very reasonable 3.53.

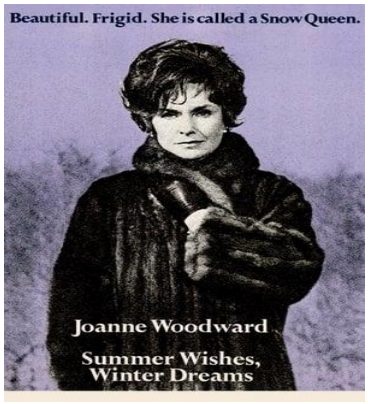


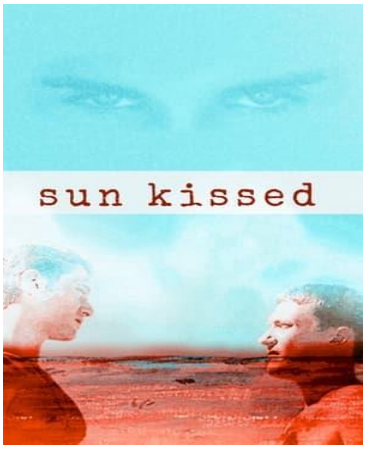
We conducted an in-depth examination of the distribution of ratings specifically within the timeframe spanning from 1995 to 2015. During this analysis, we scrutinized how user ratings were spread across movies released in this 20-year period. This exploration allowed us to gain insights into the patterns and trends in user preferences for movies released within that particular time frame.

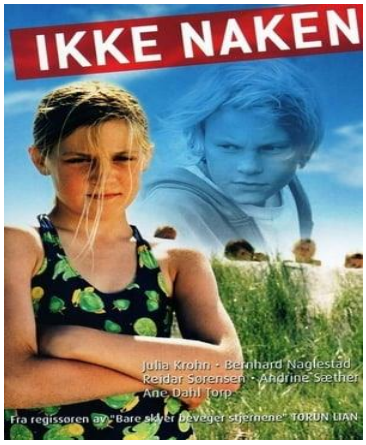


4.1.5 Uncovering Cinematic Trends

In the analysis of movie data, we unearthed some intriguing insights. Firstly, we identified the top 10 highest-rated movies, showcasing the films that garnered the most favorable reviews from viewers.

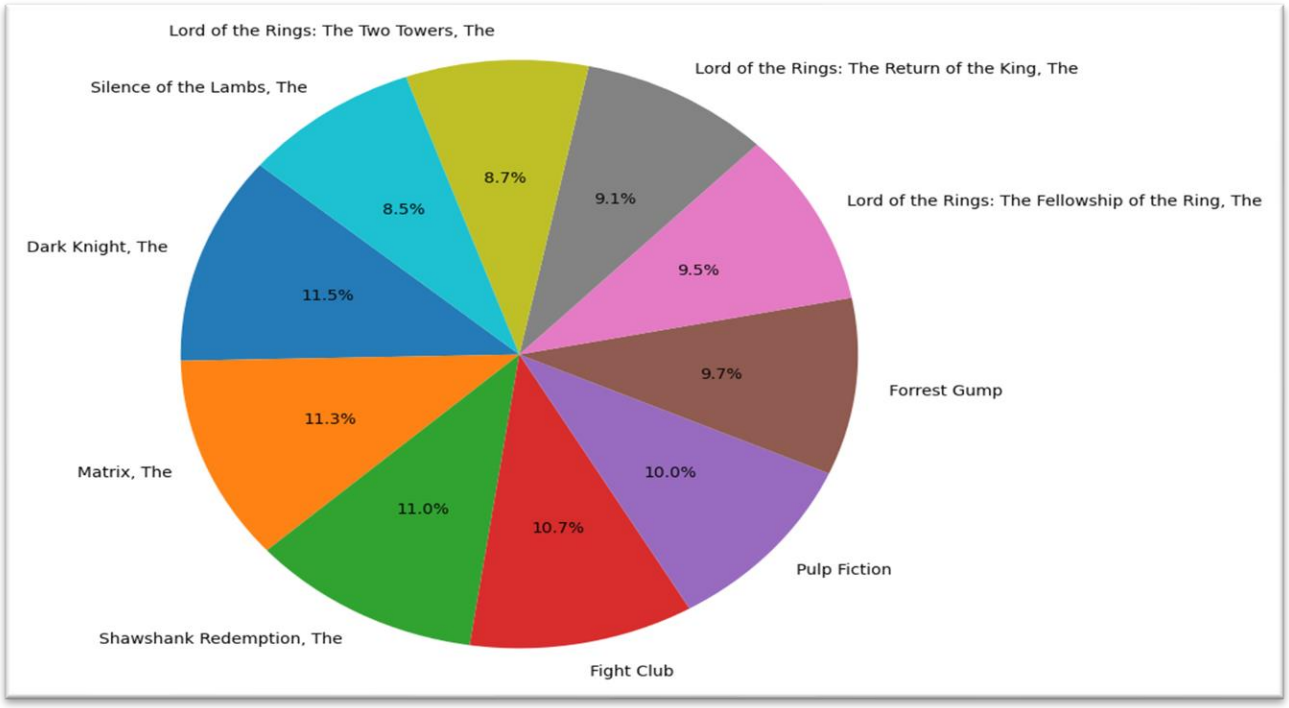




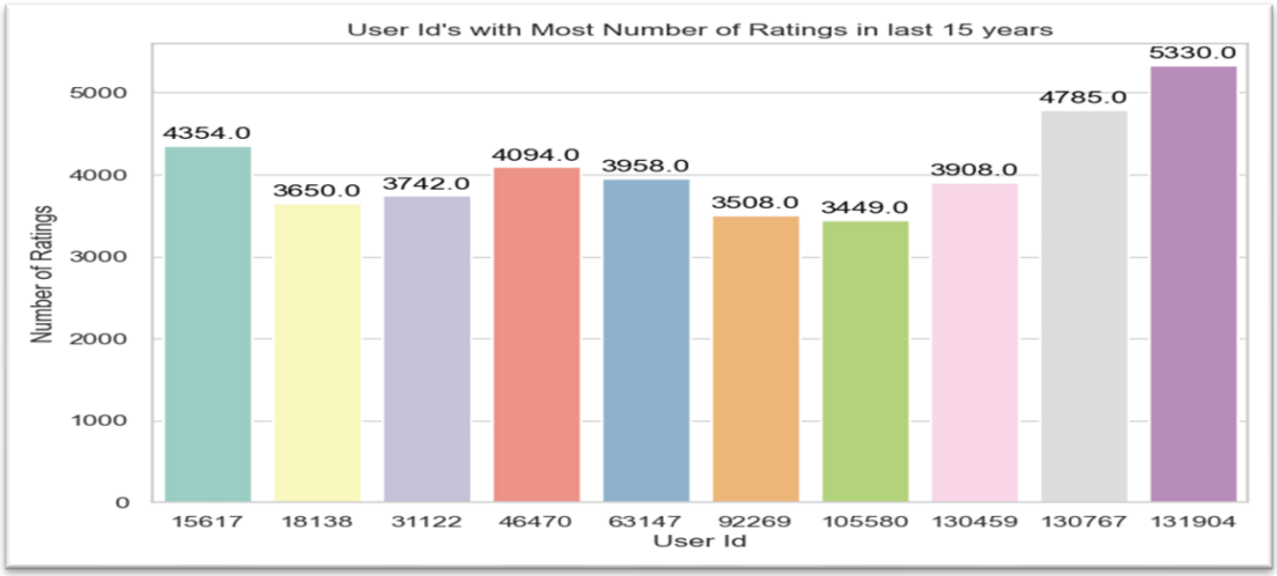


The Snow Queen (Lumikuningatar) took the spotlight as the movie with the highest overall rating across all films.

Our focus was on identifying movies from the last 15 years that achieved this top-rated status.



Finally looking at the User Data, we found which users gave the highest Number of Ratings across last 15 years.



5. Data Pre-Processing

The Goal of this Project is to define a Recommendation System which would suggest Movies based on User Interaction. It is worth noting that we have 2 Types of Feedback based on which we can build our Recommendation System. They are explained as below:-

5.1. Explicit Feedback

Under this type of Recommendation System, we build the Model based on Direct & Quantifiable Data collected from Users. For e.g. User Reviews or User Likes. However, this is not a great way of building the Recommendation System as there is a vast majority of users who do not provide User Reviews in streaming services like Netflix or Amazon Prime or Product websites like Amazon, Walmart.

5.2. Implicit Feedback

In this type of Recommendation System, we build our model based on the user interaction of a Product, Movie or Videos. For example, if a user watches a Movie or Video, we can suggest movies or videos based on the Genre of those. Similarly, if a user has browsed an item on Amazon or Walmart website, we can suggest similar items to the user.

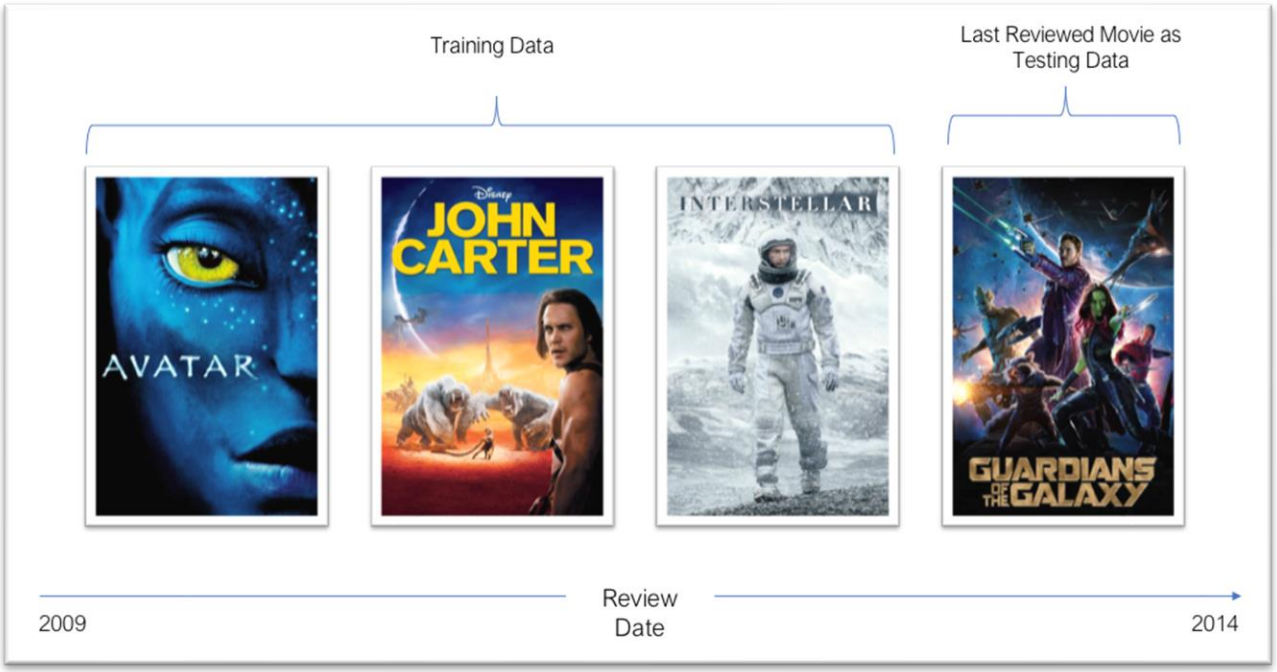
The advantage of Implicit Feedback is the abundance of data we have at our disposal. However, the one advantage is that we have no way to find the negative feedback of the user on a particular item or movie.

5.3. Data Selection

Since the Dataset has 20 Million Rows, it is beyond the scope of our Limited Hardware that we have to build a Deep Learning Model with this huge amount of Data. So we would build our model by selecting random 20% of the User data. We chose the Ratings given by 20% of the unique User Id's.

5.3.1 Train Test Split

We intend to apply the Leave-One-Out methodology, utilizing the Timestamp feature within the DataFrame. Our approach involves incorporating all the user's past ratings into the Training Set while reserving the most recent one for inclusion in the Testing Set.



5.3.2 From Explicit to Implicit Dataset

The primary objective of this project is to provide movie recommendations to users based on their historical viewing activity. While the Ratings Dataset can potentially be utilized for predicting user ratings for movies, our focus is not on this aspect. Instead, we aim to leverage the Ratings Dataset to infer user interactions with movies, enabling us to identify likely movie choices for users through our recommendation system (Implicit Feedback). To facilitate this, we plan to streamline our Ratings Dataset by introducing a new column labeled 'Viewed,' where a value of 1 signifies the user's interaction with that movie.

However, Upon binarizing our dataset, it becomes evident that all instances now fall into the positive class category. Nevertheless, it is essential to introduce negative samples for our model training, signifying movies that users have not engaged with. While this assumption implies a lack of user interest in these movies, it's worth noting that this is a generalized assumption that may not universally hold true, but it often proves effective in practical applications. To address this, we are introducing a framework where we have four negative class instances for every positive class data point, creating a balanced dataset for our recommendation system.

We have used a Py Torch Dataset to facilitate the Training.

6. Modelling

Neural Collaborative Filtering (NCF):-

Neural Collaborative Filtering (NCF) is an advanced recommendation system model that combines the strengths of neural networks and collaborative filtering techniques. It's a powerful choice for the MovieLens recommendation system for several reasons:

Flexibility: NCF can handle both collaborative filtering (user-item interactions) and content-based filtering (movie attributes) seamlessly. This flexibility allows it to capture complex patterns and relationships in movie data.

Personalization: NCF excels at providing personalized movie recommendations by learning user preferences from their historical interactions and considering item attributes like movie genres or directors.

Non-linearity: Neural networks in NCF can model non-linear relationships, allowing it to capture intricate user preferences and movie characteristics that traditional linear models may miss.

Scalability: With proper design and optimization, NCF can handle large-scale datasets like MovieLens, making it suitable for recommending movies to a vast user base.

State-of-the-art Performance: NCF has demonstrated strong performance in recommendation tasks, often outperforming traditional methods, due to its ability to capture intricate user-item interactions.

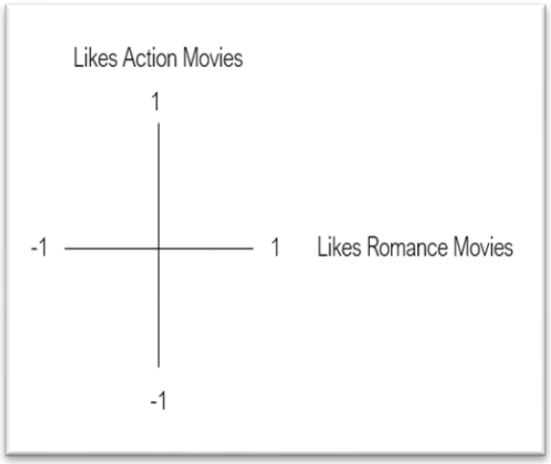
In summary, Neural Collaborative Filtering is a good model for the MovieLens recommendation system because it combines the strengths of neural networks and collaborative filtering, making it highly effective at providing personalized and accurate movie recommendations while accommodating the complexity of large-scale datasets.

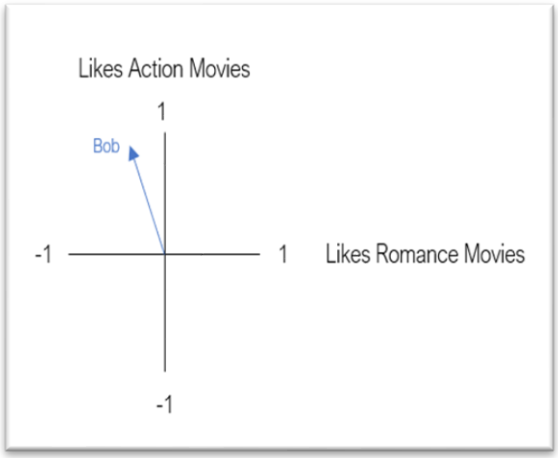
In relation to our Project, it is also worthwhile to know the concept of User Embeddings & Learned Embeddings.

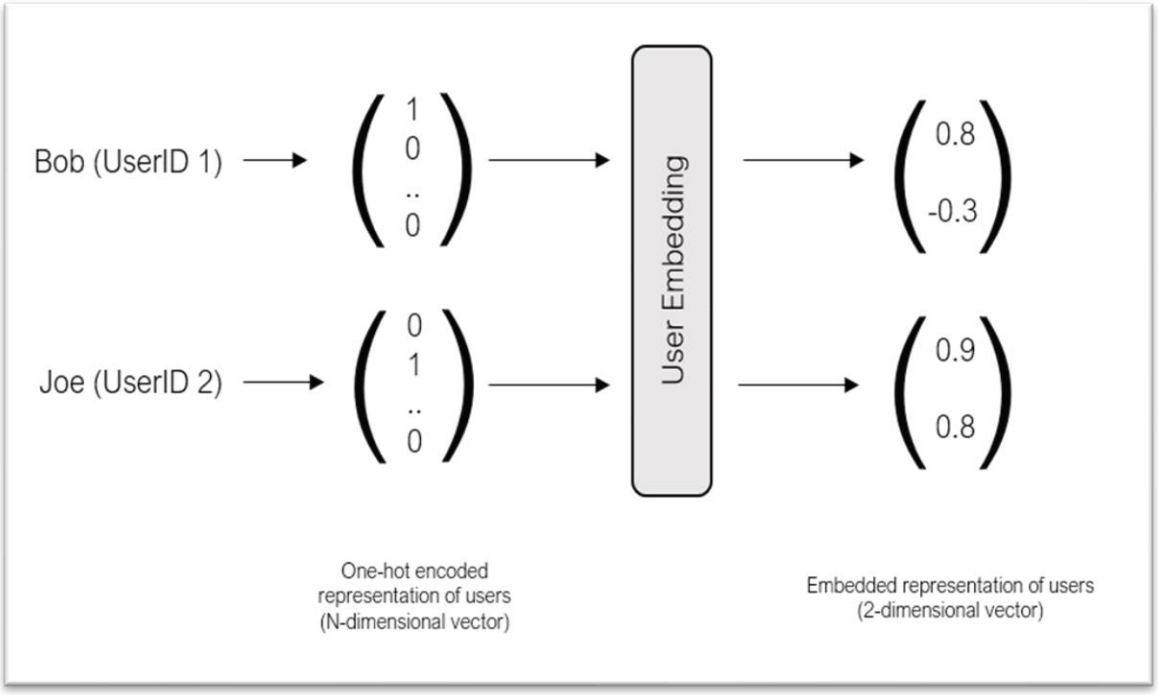
User Embeddings:-

User embeddings are a fundamental concept in recommendation systems and machine learning, particularly in collaborative filtering-based recommendation systems. User embeddings represent users in a lower-dimensional vector space, where each user is represented by a unique vector. These vectors capture latent features or characteristics of users based on their historical interactions with items (e.g., movies, products, articles) or other relevant data.

Each user is represented by a unique vector that captures their movie preferences and behavior based on their past ratings and interactions with movies. Users with similar preferences have similar embeddings.





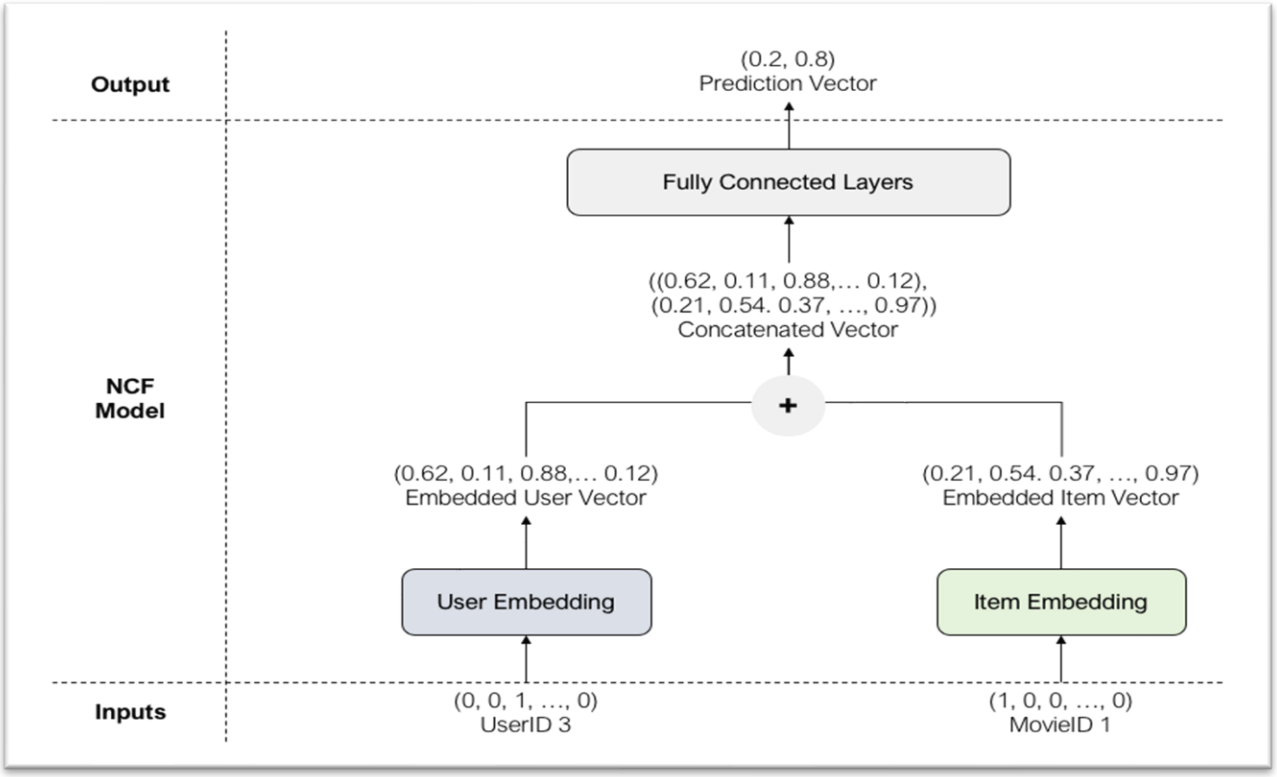


Learned Embeddings:-

Learned embeddings refer to low-dimensional, dense vectors that are automatically derived from the dataset's user-item interactions. These embeddings represent users and movies in a way that captures meaningful patterns and relationships based on historical movie ratings and preferences.

Learned embeddings are used to make movie recommendations. For example, to recommend movies to a specific user, the system can find movies with embeddings that are close to the user's embedding in the embedding space. This means recommending movies that users with similar preferences have liked.

Our model took user and item IDs as input, embedded them, passed them through fully connected layers, and predicted the user-item interactions using binary cross-entropy loss. It was configured for binary classification, aiming to predict whether a user would interact with a movie or not. The model used the Adam optimizer and DataLoader for efficient training on a MovieLens dataset. The architecture consisted of user and item embedding layers, followed by two fully connected layers, and a final sigmoid activation for binary prediction. The training step computes the loss, and the model was optimized using Adam.



We trained our NCF model for 5 epochs on the GPU. Importantly, we set **reload_dataloaders_every_epoch=True**, a practice that generated a fresh set of randomly selected negative samples for each epoch. This approach mitigated potential biases caused by the initial choice of negative samples, promoting model robustness and accuracy in recommendations.

7. Model Evaluation

To effectively assess the performance of recommender systems after training, it's crucial to recognize that traditional evaluation metrics like Accuracy (for classification) or RMSE (for regression) are often inadequate. Recommender systems have distinct characteristics and usage scenarios that require specialized evaluation metrics.

To develop robust evaluation metrics for recommender systems, it's essential to gain insights into how modern recommender systems are employed and the unique challenges they address.

We don't need an user to interact on every single item in the list of recommendations we suggest. Instead, we just need the user to interact with at least one item on the list - as long as the user does that, the recommendations have worked.

To simulate this, we did the following:-

Step 1: Randomly select 99 items for each user that the user has not previously interacted with.

Step 2: Combine the 99 selected items with the test item, which is the actual item that the user has interacted with. This results in a set of 100 items for each user.

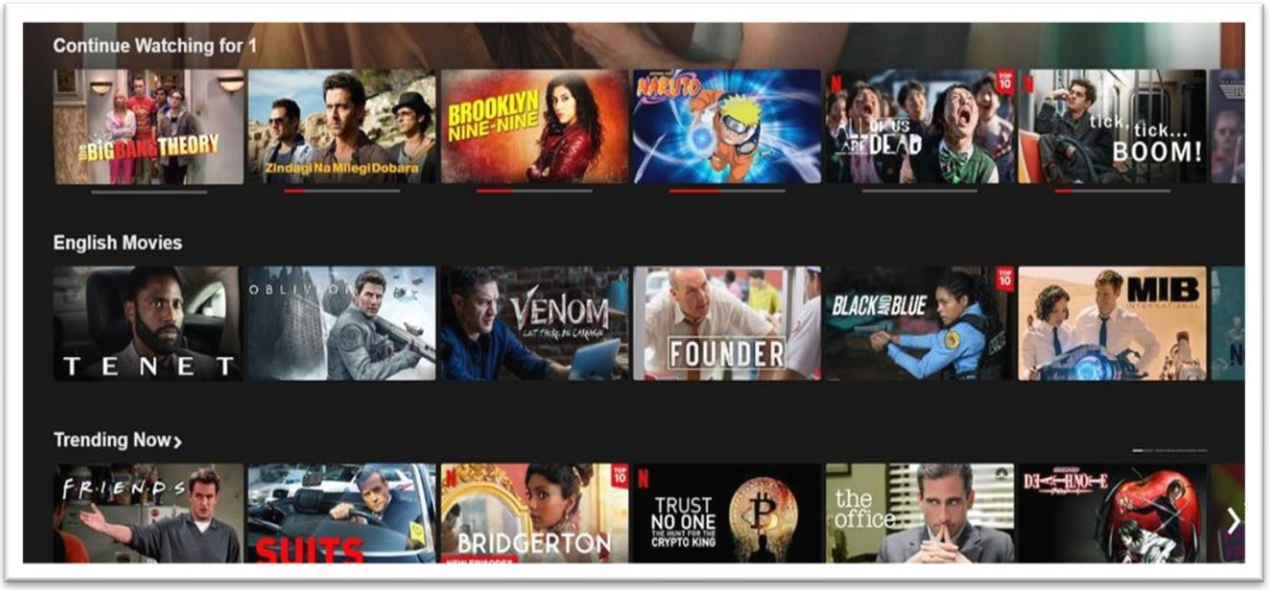
Step 3: Apply the trained recommendation model to predict the probabilities of interaction for these 100 items.

Step 4: Rank the 100 items based on their predicted interaction probabilities. Then, select the top 10 items from this ranked list.

Step 5: Check if the test item (the one the user actually interacted with) is present within the top 10 recommended items. If it is, count it as a "hit."

Final Evaluation: Repeat the entire process for all users in the dataset. Calculate the Hit Ratio by averaging the number of hits across all users.

The robustness of our model becomes apparent when we consider the Hit Ratio @ 10 score. It reveals that a remarkable ***80%*** of users were provided with a set of 10 recommended movies, and among those recommendations, the actual movies they engaged with was consistently included. This underscores the model's effectiveness in offering highly relevant suggestions, aligning closely with users' preferences, and enhancing their overall experience.



8. Conclusion

In this comprehensive exploration of the MovieLens dataset, we delved into various facets of movie data to extract meaningful insights and enhance our understanding of user preferences and film trends over the last 15 years. Through extensive analysis and the application of advanced data science techniques, we uncovered valuable information and built a robust recommendation system that achieved an impressive 80% Hit Ratio @ 10.

Our journey began by identifying the Top 10 Movies from the past 15 years, shedding light on the most popular and highly regarded films of recent times. This analysis allowed us to discern the evolving tastes of moviegoers.

Next, we delved into the realm of movie ratings, unveiling the Top 10 Highest Rated Movies from the last 15 years. This endeavor showcased cinematic gems that garnered both critical acclaim and audience appreciation.

For a deeper understanding of user engagement, we examined the Top 10 Movies with the highest number of User Ratings in the last 15 years. This provided valuable insights into the movies that captivated a broad and engaged audience.

Our exploration also included a study of the user base, where we determined the Count of Unique Users who actively rated movies in the last 15 years. This metric reflects the vibrant and dynamic community of movie enthusiasts.

Additionally, we identified the User ID that contributed the highest number of ratings, highlighting the significant impact of dedicated users on the dataset.

Finally, the culmination of our project involved the development of a sophisticated recommendation system using deep learning models. With an impressive 80% Hit Ratio @ 10, our system excelled in suggesting movies that closely matched users' preferences, enhancing their movie-watching experience and solidifying the effectiveness of our data-driven approach.

In conclusion, our exploration of the MovieLens dataset not only provided valuable insights into the world of cinema but also showcased the power of data science and deep learning in creating personalized and effective recommendation systems. This project serves as a testament to the importance of leveraging data to enhance user experiences and deliver tailored content recommendations in the ever-evolving landscape of entertainment.

9. Project's Versatility: Beyond Movie Recommendations

This project transcends movie recommendations, offering valuable applications in various sectors such as online retail (e.g., Amazon, Walmart, eBay) for product suggestions based on purchase history and in digital platforms (e.g., Facebook, YouTube) for personalized ad and video recommendations.

10. Reference

- The dataset was sourced from Kaggle.
- The code was developed using Python, PyTorch, scikit-learn, pandas, seaborn, matplotlib, and various other libraries.
- Movie posters were obtained from themoviedb.org, a freely accessible resource.
- Certain diagrams were gathered from the internet, with no copyright violations involved.