Title: Movie Recommender System
DSC680-Applied-DataScience

Author: Sheetal Munjewar
Institution: Bellevue University
https://github.com/smunjewar/SheetalM.github.io

Abstract

The increasing volume of content on platforms like Netflix, YouTube, and Hulu makes it challenging for users to find movies or shows they like. Recommender systems help solve this problem by filtering large catalogs and suggesting content based on user preferences. This paper discusses the development of a movie recommender system, exploring various recommendation approaches, including content-based, collaborative filtering, and hybrid models. It also discusses data sources, methodologies, results, and potential future enhancements.

Business Problem

As online media platforms grow, the challenge of helping users find content suited to their preferences becomes more significant. Without a recommendation system, users may spend too much time searching for content, which could lead to frustration and platform abandonment. This paper explores how a well-implemented recommender system can improve user engagement, satisfaction, and retention on platforms like Netflix and YouTube.

Background/History

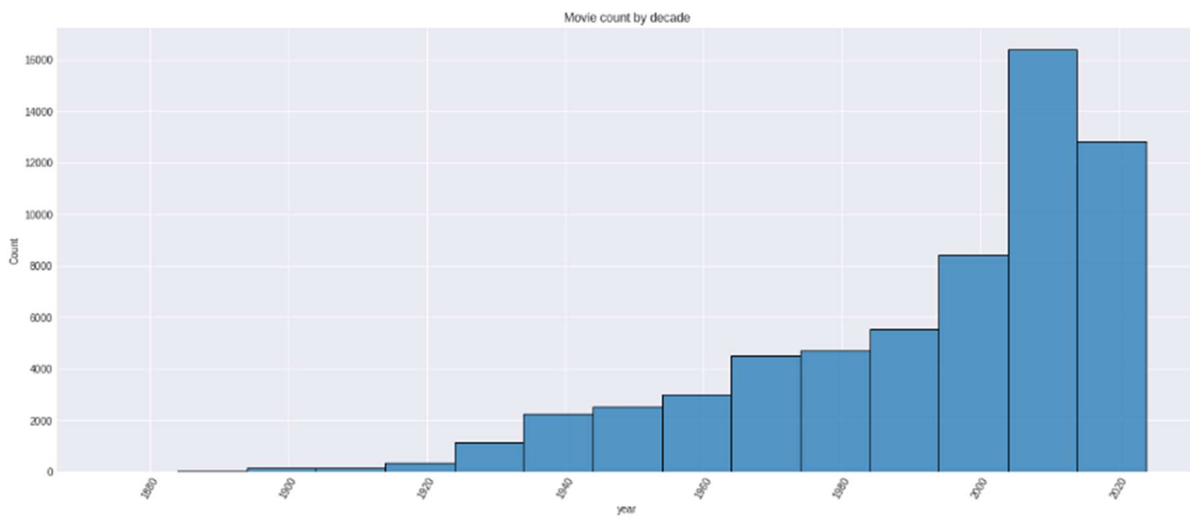
Recommender systems have evolved as a key technology in personalizing user experiences across various industries, especially in streaming and e-commerce. With the advent of big data and machine learning, different types of recommender systems, such as content-based and collaborative filtering, have been developed to offer more accurate and efficient suggestions. This paper explores these systems' historical development and how they are applied to the entertainment industry.

Data Explanation

The data used for this project comes from two main sources:

Rating data: Obtained from GroupLens, containing ratings for 62,000 movies by 162,000 users (25 million ratings in total).

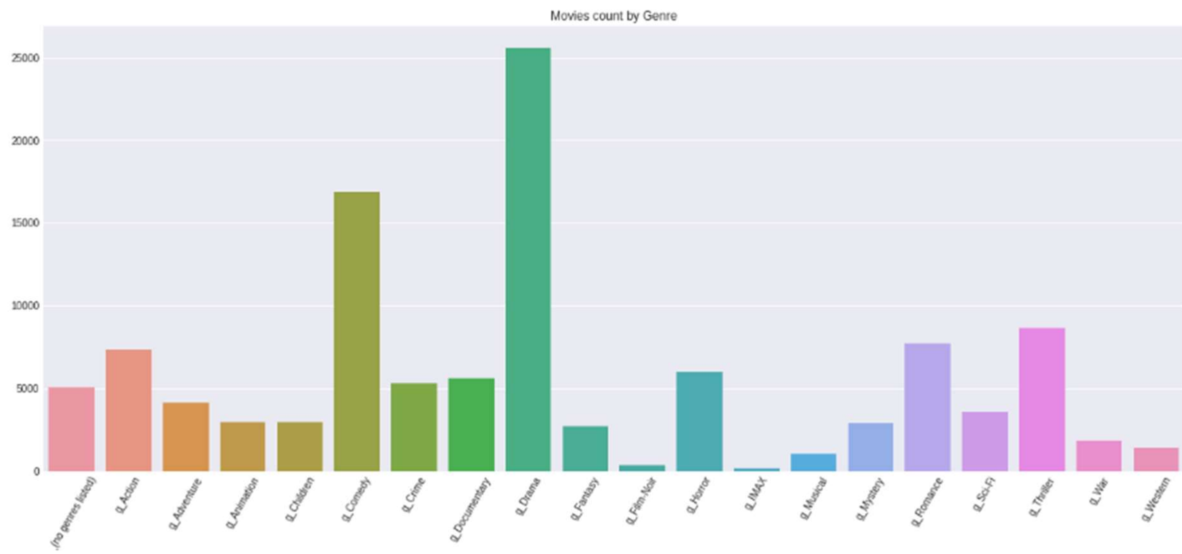
Metadata: Sourced from Kaggle, which includes information about 45,000 movies (movieId, title, genres, etc.).



Data preprocessing steps include:

Conversion of genres from a pipe-separated string to dummy columns.

Extraction and cleaning of movie descriptions (overview, tagline) for content-based filtering.



Methods

Several recommendation models are explored:

Weighted Rating: This model incorporates the number of users who rated a movie alongside the average rating to improve recommendations.

Content-Based: Using cosine similarity, this method compares movie descriptions (overview and tagline) to recommend similar movies.

Collaborative Filtering: A user-based model that predicts ratings based on similar user behaviors, achieving 93% accuracy on test data.

Hybrid Model: Combines content-based and collaborative filtering to further enhance recommendation accuracy.

Analysis

The analysis focuses on evaluating the performance of different recommendation models. Key metrics include:

Accuracy of the collaborative filtering model, which predicts movie ratings with a 93% accuracy.

Similarity Scores derived from content-based filtering, showing improved recommendation quality.

Hybrid Model Performance, which combines the strengths of both content-based and collaborative models.

Conclusion

The results demonstrate that hybrid recommender systems significantly outperform individual models. By combining content-based recommendations with collaborative filtering, the system can provide more relevant and personalized movie suggestions. This method not only improves user satisfaction but also enhances engagement, making it a vital tool for platforms dealing with large content libraries.

Assumptions

The dataset contains sufficient diversity in user ratings and movie genres to generate meaningful recommendations.

The recommendation system assumes that user behavior (ratings) is a good indicator of preferences.

Limitations

Data Quality: Missing or sparse data in the movie ratings can affect recommendation quality.

Scalability: As the number of movies or users increases, the performance of the system might degrade unless optimized for large-scale computations.

Challenges

Cold Start Problem: New users or movies without prior ratings make it difficult to generate accurate recommendations.

Bias: Certain movies may be recommended more frequently due to their higher ratings, potentially ignoring niche content.

Future Uses/Additional Applications

The recommender system can be extended to other media platforms (e.g., music, e-commerce).

Integration with real-time data can allow the system to adapt dynamically to changing user preferences.

Recommendations

Implementing a **Hybrid Recommender System** to combine both content-based and collaborative filtering models can provide better recommendations.

Real-Time Adaptation: Incorporating user behavior in real-time to refine recommendations further.

Implementation Plan

Model Development: Build the recommender system using the specified models (weighted rating, content-based, collaborative filtering, hybrid).

Data Processing: Clean and preprocess the data as described, ensuring it is ready for model input.

Testing and Deployment: Evaluate model performance on test data, implement optimizations, and deploy the system on a platform.

Ethical Assessment

The system must respect user privacy and confidentiality, ensuring that all data used for recommendations is anonymized and used only for the intended purpose. Additionally, the system should avoid reinforcing biases or recommending content that could be considered harmful or offensive.

Acknowledgments

The author acknowledges the contributions of GroupLens and Kaggle for providing the datasets used in this project. Thanks to various online tutorials, including DataCamp and Kaggle notebooks, for valuable insights.

References

Sharma, A. (2020). *Beginner Tutorial: Recommender Systems in Python*. DataCamp. *https://www.datacamp.com/tutorial/recommender-systems-python*

Yurtekin, Ş. (2020). *Movies Recommendation*. Kaggle. *https://www.kaggle.com/sevvalyurtekin/movies-recommendation*

Rocca, B. (2019). *Introduction to Recommender Systems*. Towards Data Science. *https://en.wikipedia.org/wiki/Recommender_system#:~:text=A%20recommender%20system%2C%20or%20a,would%20give%20to%20an%20item*

Isinkaye, F.O., Folajimi, Y.O., & Ojokoh, B.A. (2015). *Recommendation systems: Principles, methods, and evaluation*. ScienceDirect.*https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada*