

ENTERTAINMENT CONTENT RECOMMENDATION SYSTEM USING MACHINE LEARNING

Assistant Professor: Ms. Karuna Middha¹, Student: Munzir², Sarthak Goja³, Sourabh Choudhary⁴

^{2,3,4}UG Student, Dept. of CSE Engineering, Maharaja Agrasen Institute of Technology, Delhi, India.

Abstract – Entertainment Content recommendation systems are important for several reasons.

First, they can help users discover new movies that they may not have otherwise found. With so many movies available to watch, it can be overwhelming for users to sift through all of the options and find something that they will enjoy. A recommendation system can narrow down the choices and present the user with a curated selection of movies that are tailored to their personal preferences.

Finally, recommendation systems can improve the efficiency of finding movies to watch. Instead of spending a lot of time searching through various movie platforms or scrolling through long lists of movies, a recommendation system can present the user with a short list of options that are likely to be relevant and interesting to them.

Keywords: content-based approach; sentimental analysis; recommendation system; movie ratings; inductive learning

1. INTRODUCTION

In recent years, there has been a proliferation of movie streaming platforms and other online sources of video content. As a result, consumers have access to a vast and ever-growing selection of movies and TV shows to watch. However, with so much choice, it can be challenging for users to find movies that they will enjoy and that fit their personal preferences.

There has been significant research on the development and evaluation of movie recommendation systems in recent years. Researchers have explored various methods for collecting and analyzing user data, as well as different approaches for making recommendations. In this paper, we will review the state of the art in movie recommendation systems and discuss the challenges and opportunities that exist in this field. Despite the progress that has been made in this field, there are still many challenges and opportunities for future research. For example, one key challenge is to effectively handle the "cold start" problem, where the system must make recommendations for a new user with little or no

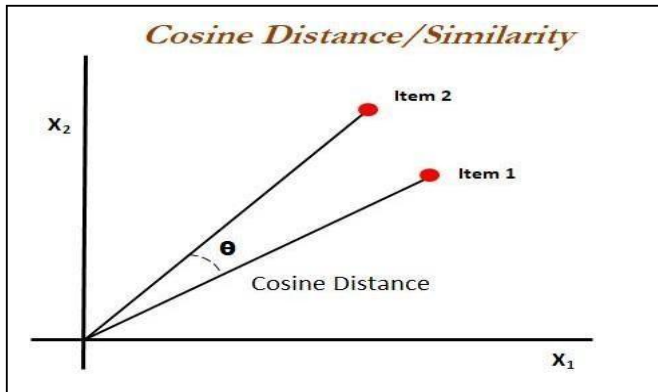
previous data. Another challenge is to improve the diversity and novelty of recommendations, while still ensuring that they are relevant and personalized. Finally, researchers are also exploring ways to incorporate additional sources of data, such as social media, to enhance the performance of movie recommendation systems.

2. RELATED WORK

There are two main types of recommender systems: collaborative filtering and content-based filtering. Collaborative filtering relies on user-related information, preferences, and interactions to identify similarities between users and recommend movies that similar users have enjoyed. There are two subtypes of collaborative filtering: model-based and memory-based algorithms. Memory-based methods do not have a training phase and use measures like Pearson correlation coefficient and Cosine similarity to identify similar users. Model-based methods, on the other hand, try to predict user ratings of a movie using estimated models. Collaborative filtering methods can be computationally intensive and may not perform well with sparse data. They also assume that users with similar tastes will rate movies similarly, which may not always be the case. Content-based methods, on the other hand, use information about the content of movies, such as audio and visual features or textual metadata, to find similarities among movies and recommend those that are similar to ones that the user has accessed or searched for. These methods do not incorporate user behavior in their recommendations. In this paper, we will be exploring the latter approach.

3. LITERATURE REVIEW

Nessel stated in the movie oracle that working with examples is an essential part of human interaction and tried to provide a movie recommendation engine based on this behavior. Which of course requires considerably more computing power, as the compared bodies of text are much larger, but the algorithms are essentially the same [3]. In a content-based movie recommendation system, the proposed algorithm uses textual metadata of the movies like plot, cast, genre, release year and other production information to analyze them and recommend



$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n A_i^2} \times \sqrt{\sum_{i=1}^n B_i^2}}$$

D. Sentimental analysis

Sentiment analysis is a field within natural language processing that involves the evaluation of subjective opinions, views, or feelings about a particular subject that have been collected from various sources. In more practical business terms, sentiment analysis can be described as a set of tools used to identify and extract opinions and utilize them for the benefit of business operations. Such algorithms delve into the text to identify the underlying attitude towards a subject or its specific elements. An example of a commonly used algorithm in sentiment analysis is the multinomial naive Bayes classifier. This algorithm assumes that the features being analyzed are produced from a simple multinomial distribution. The multinomial distribution defines the probability of observing counts within a number of categories, making it well-suited for features that represent counts or count rates. The basic idea is the same as before, except that instead of modeling the data distribution with a best-fit Gaussian curve, we model it with a best-fit multinomial distribution.

P (positive | overall liked the movie) = P (overall liked the movie | positive) * P (positive) / P (overall liked the movie)

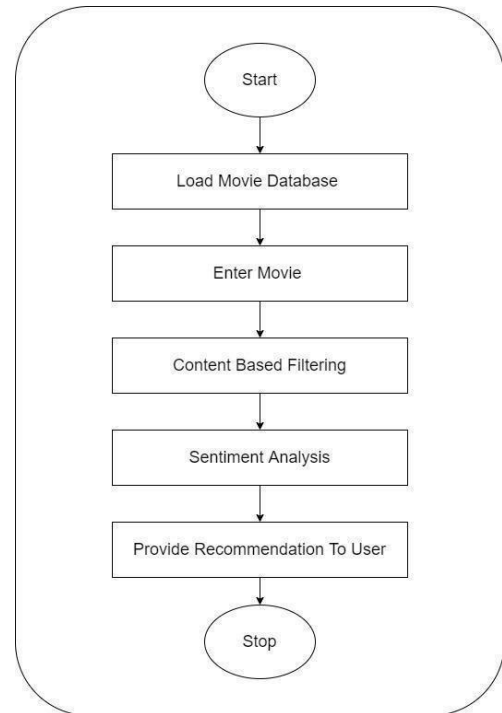


Fig -2: Flowchart of the Movie recommendation system

5. PROPOSED SYSTEM

A. Dataset

For our research, we utilized three different datasets from MovieLens, a collection of datasets generated by the GroupLens Research team for the purpose of evaluating recommender systems. These datasets are commonly used by developers to test their recommendation systems. These are:

1. IMDB 5000 Movie Dataset
2. The Movies Dataset
3. List of movies in 2018
4. List of movies in 2019
5. List of movies in 2020

The Movies dataset includes metadata for all 45,000 films listed in the Full MovieLens dataset, which includes movies released up until July 2017. The data includes information about the cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, TMDb vote counts and vote averages. This dataset also includes files with 26 million ratings from 270,000 users for 45,000 movies, with ratings ranging from 1 to 5, obtained from the official GroupLens website. The datasets for movies released from 2018 to 2020 were obtained by web scraping their respective Wikipedia pages.



To determine the weighted rating of each film, we will select the 25 most similar movies according to their similarity scores. We will then calculate the vote of the movie that falls at the 60th percentile of this group. Finally, we will apply IMDB's formula to compute the weighted rating of each movie using this value.

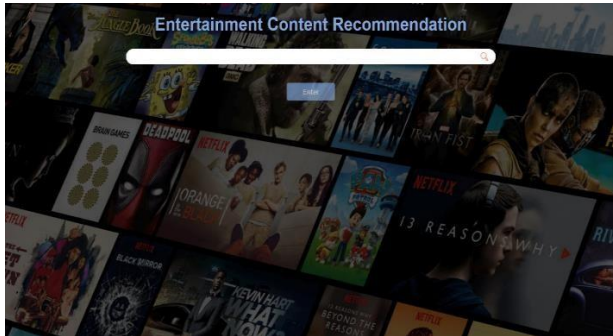


Fig -6: Home page of the Entertainment recommendation system

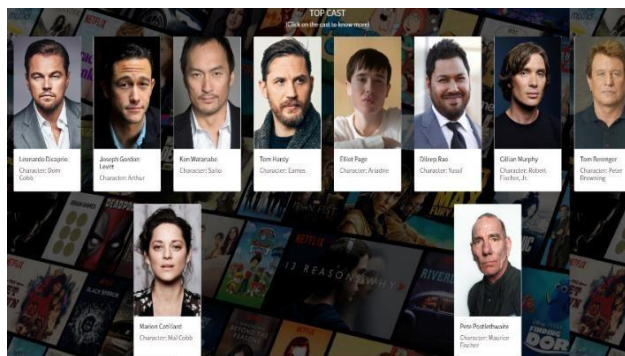


Fig -7: Cast of the Movie

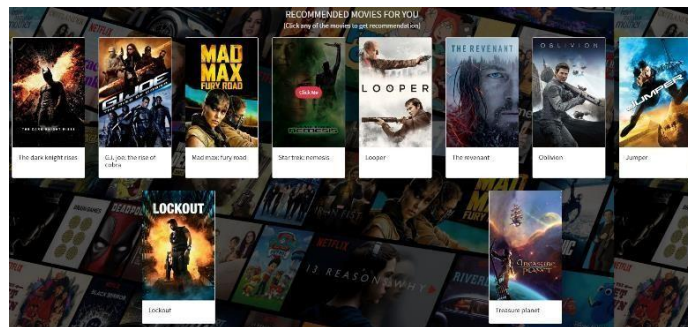


Fig.8 : Recommendation Entertainment Content

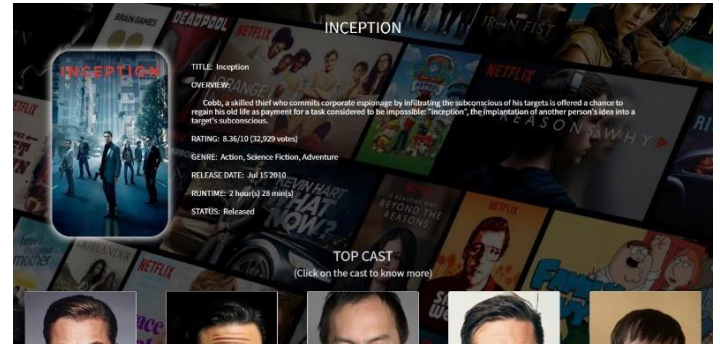


Fig -9: Poster and info page of the Movie



Fig.10: Sentimental analysis of the Entertainment Content recommendation

7. CONCLUSION

Our proposed algorithm utilizes textual metadata, such as the plot, cast, genre, release year, and other production details of movies, to analyze and recommend the most similar films. The user only needs to input a movie of interest, and the system will generate appropriate recommendations. We tested our algorithm on a subset of the movies available on IMDb and found that the cosine similarity measure was effective for forecasting recommendations in recommendation systems. Additionally, we implemented a feature that allows for retraining of the system by rating results as "good" or "bad," resulting in more accurate predictions than simply selecting one movie or providing a single piece of text.

In the future, we plan to track movies searched by users in nearby locations to recommend popular films. We can also consider incorporating the watch history of geographically contextual users (those living in close proximity) with the watch history of the user to provide more "location-relevant" recommendations. Additionally, by using user ratings of movies from websites like Rotten Tomatoes, Metacritic, and IMDb, we can explore the possibility of combining collaborative filtering techniques with our method to create a hybrid model that combines the advantages of both approaches.

REFERENCES

- [1] Mykhaylo Schwarz, Mykhaylo Lobur, Yuriy Stekh, Analysis of the Effectiveness of Similarity Measures for Recommender Systems, 978-1-5090-5045-1/17/\$31.00 ©2017 IEEE M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
- [2] Rujhan Singla, Saamarth Gupta, Anirudh Gupta, Dinesh Kumar Vishwakarma, FLEX: A Content Based Movie Recommender, 978-1-7281-6221-8/20/\$31.00 ©2020 IEEE.
- [3] Jochen Nessel, Barbara Cimpa, The MovieOracle - Content Based Movie Recommendations, 978-0-76954513-4/11 \$26.00 © 2011 IEEE.
- [4] Shreya Agrawal, Pooja Jain, An Improved Approach for Movie Recommendation System, 978-1-5090-32433/17/\$31.00 ©2017 IEEE
- [5] Yibo Wang, Mingming Wang, and Wei Xu, A SentimentEnhanced Hybrid Recommender System for Movie Recommendation: A Big Data Analytics Framework, Hindawi Wireless Communications and Mobile Computing Volume 2018, Article ID 8263704
- [6] F. Furtado, A, Singh, Movie Recommendation System Using Machine Learning, Int. J. Res. Ind. Eng. Vol. 9, No. 1 (2020) 84-98
- [7] Robin Burke. Hybrid recommender systems: Survey and experiments. User modeling and user- adapted interaction, 12(4):331-370, 2002.
- [8] Erik Cambria. Affective computing and sentiment analysis. IEEE Intelligent Systems, 31(2):102-107, 2016.
- [9] Ivan Cantador, Alejandro Bellog ´ın, and David Vallet. Content-based recommendation in social tagging systems. In Proceedings of the Fourth Conference on Recommender systems, pages 237- 240. ACM, 2010.
- [10] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In Proceedings of the Fourth Conference on Recommender Systems, pages 39-46. ACM, 2010. ISBN 978-1-60558-906-0