DSC 680 -PROJECT MILESTONE 2

Avinash Alapati

TOPIC – MOVIE RECOMMENDATION

The goal of this project is to provide a recommendation system for video content providers to predict whether someone will enjoy a movie based on how much they liked or disliked other movies.

BUSINESS PROBLEM

Major companies like YouTube, Amazon, Netflix use recommendation systems in social and e-commerce sites use recommendation system for its users to suggest for an individual according to their requirement more precise and accurate.

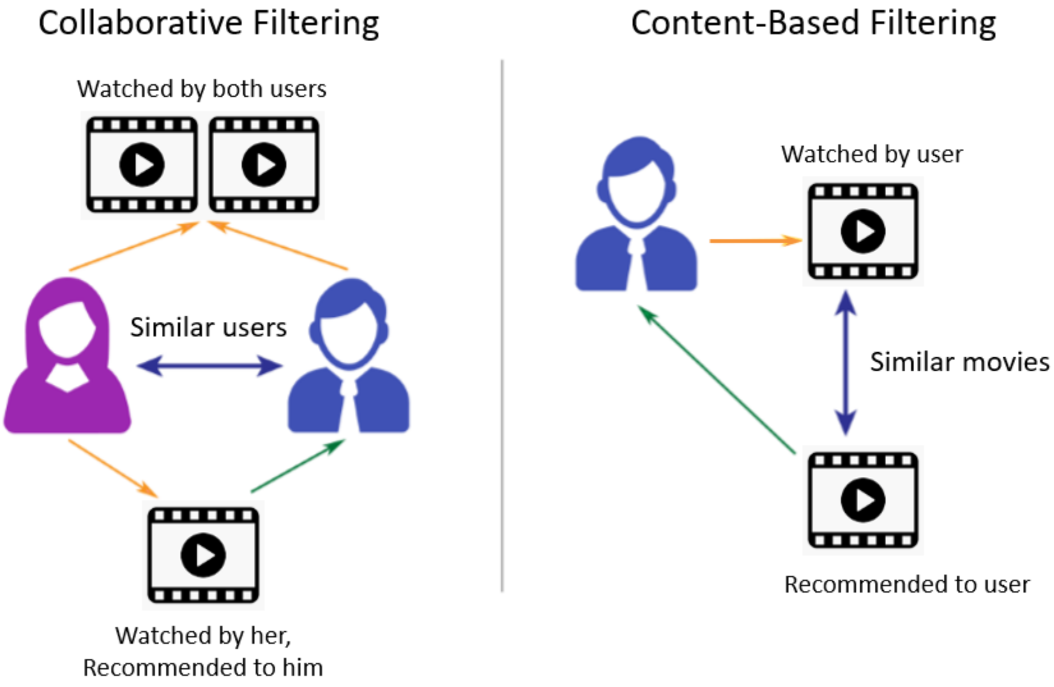
The project aims to address the challenge of personalized movie recommendations for users, enhancing user engagement on a streaming platform. By leveraging collaborative filtering and content-based recommendation methods, the goal is to improve user satisfaction and retention by suggesting movies tailored to individual preferences.

Recommendation Systems are classified in mainly three categories:

- Content-based systems
- Collaborative Filtering system
- Hybrid recommendation system.

Content based systems works based on the label or genre of an item. If a user watched a movie so it recommends similar movies based on director, a genre, and many more aspects.

The main theory behind the collaborative filtering is that if users ‘A’ and ‘B’ have rated correspondingly in the past, then there will be an assumption that they will rate correspondingly in the future



BACKGROUND/HISTORY:

The streaming industry has witnessed exponential growth, but the issue of keeping users engaged remains critical. Personalized recommendations contribute significantly to user satisfaction, prompting the need for an effective recommender system.

DATA EXPLANATION:

- **D**ata Source: MovieLens dataset.
- Data Prep: Cleaning, handling missing values, and feature engineering for user ratings, movie metadata, and additional information.
- Data Dictionary: Comprehensive documentation outlining the structure and meaning of each dataset field.

DATASETS

The primary dataset will be sourced from the MovieLens dataset, containing user ratings, movie metadata, and user information. Additionally, external data sources may be explored for enriching content-based recommendations, such as genre details, director information, and movie summaries.
from: https://grouplens.org/datasets/movielens/latest/

Data description

The dataset contains 100k+ ratings and 3k+ tag applications across 9k+ movies. The data was captured for 600+ users between 1995 and 2018. This dataset was generated on 2018.

METHODS

Recommendation system works on basically on two things - product details and user details. We must collect them from the system or from the database and make decisions on the basis of ratings if similar items were found then it will generate recommendation system otherwise no recommendation system will be generated.

We will apply item based collaborative filtering. The reason behind this is because user taste may change with respect to time, but item doesn't change it remains same. There are certain stages to make our recommendation system efficiently to respond.

• **Data Loading** – To load the data and display accordingly we have to perform some operation like merging the two files in the dataset.

• **Data Slicing** – we are removing unnecessary column and data.

• **Data Cleaning** – In the real-world data if we make a table of ratings in the recommendation system, we find that most of the user are not rating the movies and are mostly inactive. The same cases are with movies either users don't watch or it's gets too old. To make our computation more accurate we will remove such users and movies from our research.

Now to predict similarity we have two methods either we can use correlation method or cosine method.

Approach:

- The Simple Recommender offers generalized recommendations to every user based on movie popularity and genre.
- The basic idea behind this recommender is that movies that are more popular and more critically acclaimed will have a higher probability of being liked by the average audience.
- This model will not provide personalized recommendations based on the user.

Implementation:

- The implementation of this model is extremely trivial.
- All we have to do is sort our movies based on ratings and popularity and display the top movies of our list.
- As an added step, we can pass in a genre argument to get the top movies of a particular genre.

ANALYSIS:

Exploration of user preferences, performance evaluation of different recommendation methods, and the impact on user engagement metrics.

CONCLUSION:

Summarizing findings, evaluating the effectiveness of the recommender system, and addressing business problem resolution.

ASSUMPTIONS:

- Users provide genuine ratings.
- MovieLens dataset is representative of diverse user preferences.

LIMITATIONS:

- Cold start problem for new users or movies.
- Dependence on historical data.
- Privacy concerns regarding user data.

FUTURE USES/ADDITIONAL APPLICATIONS:

- Explore potential expansion of the recommender system for other media types (TV shows, music, etc.) or collaboration with external platforms.

RECOMMENDATIONS:

Continuously update and refine recommendation algorithms based on user feedback. Consider exploring advanced machine learning techniques and regularly update the dataset.

ETHICAL CONSIDERATIONS

- Privacy Concerns: Handling user data requires strict adherence to privacy regulations, ensuring anonymization and secure storage of user information.
- Bias in Recommendations: Careful consideration will be given to avoid reinforcing existing biases in the recommendation system, promoting diverse and inclusive suggestions.

CHALLENGES

- Cold Start Problem: New users or movies with limited data may pose challenges for accurate recommendations.

- Scalability: Ensuring the recommender system scales efficiently as the user and movie database grows.
- Data Quality: Cleaning and preprocessing the MovieLens dataset for accurate recommendations.

QUESTIONS:

1. New User: A newly released movie cannot be recommended to the user until it gets some ratings. A new user or item added based problem is difficult to handle as it is impossible to obtain a similar user without knowing previous interest or preferences. How to handle this scenario?
2. Synonymy arises when a single item is represented with two or more different names or listings of items having similar meanings, in such condition, the recommendation system can’t recognize whether the terms show various items or the same item. How can we address this issue?
3. Scalability of the model?
4. Drawbacks and limitations of Collaborative Filtering?
5. Drawbacks and limitations of Content Based Filtering?
6. Drawbacks and limitations of Demographic Filtering?
7. Which is the best Algorithm for Recommendation?
8. What are the factors that are taken into consideration while recommending a movie to the user, for eg. Age, demography, ethnicity, interests, language etc.?
9. How much data is enough to predict recommendations for a user or does these changes with the amount of data we process?
10. How soon this recommender system needs to be re-trained?

REFERENCES

[1]Herlocker, J, Konstan, J., Terveen, L., and Riedl, J. Evaluating Collaborative Filtering Recommender Systems. ACM Transactions on Information Systems 22 (2004), ACM Press, 5-53.

[2] Koren, Yehuda. "Factorization meets the neighborhood: a multifaceted collaborative filtering model." In Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 426–434. ACM, 2008.

[3] https://www.mygreatlearning.com/blog/masterclass-on-movie-recommendation-system/

[4] https://docs.microsoft.com/en-us/dotnet/machine-learning/tutorials/movie-recommendation

[5] MovieLens 2018 Introduction-to-Machine-Learning https://github.com/codeheroku/Introduction-toMachine Learning/tree/master/CollaborativeFiltering/dataset