Movie Recommendation System
CSE 573 Project Proposal: Group 2

Anushka Atul Wakankar
awakanka@asu.edu
1228756154

Kuriakose Eldho
keldho@asu.edu
1229614869

Rohan Chhibba
rchhibba@asu.edu
1229817786

Yash Tomar
ytomar2@asu.edu
1229564078

Yashi Agnihotri
yagniho1@asu.edu
1229415085

***Abstract*—In the dynamic landscape of the entertainment industry, where streaming services offer an overwhelming array of content, the need for effective movie recommendation systems is paramount. This project aims to develop a robust movie recommendation system, drawing inspiration from datasets like Netflix Prize[1] and Movie Lens[2]. By combining collaborative filtering and content-based filtering models, we aspire to provide users with a refined selection tailored to their preferences. The challenge of content overload in the online domain will be addressed, enhancing the overall user experience. Key focuses include optimizing recommendation algorithms, mitigating the cold-start problem, and evaluating them using accuracy metrics such as Root-Mean-Square error (RMSE). This project endeavours to contribute to the evolving field of recommendation systems, navigating the complexities of the modern streaming era.**

***Keywords*—**Recommendation System, Collaborative Filtering, Content-Based Filtering, Personalization, Content Overload.

I. Problem Definition

In the realm of online movie streaming platforms, where user engagement is pivotal for success, the challenge lies in providing highly personalized movie recommendations. This project focuses on refining the movie recommendation system by addressing the specific hurdles posed by content-based filtering and collaborative filtering algorithms. Content-based systems rely on attributes like movie tags, actors, and genres, while collaborative filtering considers user preferences based on the behaviour of other users. We aim to create an efficient movie recommendation system inspired by datasets such as Netflix Prize and Movie Lens. By employing a hybrid approach that integrates collaborative filtering and content-based filtering models, the project seeks to offer users a tailored selection aligned with their preferences. Key challenges to be addressed include optimizing recommendation algorithms, mitigating the cold-start problem, and ensuring accuracy metrics like Root-Mean-Square error (RMSE). This project aims to contribute to the evolution of recommendation systems, specifically catering to the intricacies of the modern streaming era

II. Data Sets

In our exploration of datasets for the movie recommendation system, we delved into widely used movie datasets available in the open-source data community, particularly on Kaggle. One of the primary datasets we examined is the Netflix prize dataset[1], which comprises three distinct sets: training, qualifying, and probe datasets, encompassing over 100 million ratings. The training dataset, with ratings for 17,770 movies, will serve as the foundation for our training purposes. Concurrently, the probe dataset will be used in the validation process. The training set has the following attributes-

- MovieIDs range from 1 to 17770 sequentially.
- CustomerIDs range from 1 to 2649429, with gaps. There are 480189 users.
- Ratings are on a five-star (integral) scale from 1 to 5.
- Dates have the format YYYY-MM-DD.

Additionally, the ”movie title.txt” file contains pertinent movie information, presenting details such as movieId, year of release, and title.

The second dataset scrutinized for our movie recommendation system is the MovieLens dataset[2], specifically the MovieLens 25M dataset. This extensive dataset has been accumulating movie tags and ratings from users since 1995, amassing a total of over 25 million movie ratings spanning 62,000 movies by 162,000 users.

The MovieLens 25M dataset comprises three primary data files pivotal for our recommendation system:

- tag.csv: Contains information about tags applied by users.
- rating.csv: Encompasses data on ratings provided by users.
- movie.csv: Presents movie details such as movieid, title, and genre.

In addition to the aforementioned files, supplementary data is available in the form of genome scores, genome tags, and link files. These files provide further insights into movie-tag relevance data, tag descriptions, and external sources, including IMDB and TBMD Ids, associated with each movie.

III. State-of-Art Methods & Algorithms

A. Movie Recommendation System using collaborative filtering by implementing the K-Nearest Neighbors algorithm

Collaborative Filtering, a prevalent approach in recommendation systems, leverages the wisdom of the crowd by predicting user preferences based on the behaviors and preferences of like-minded individuals. K-Nearest Neighbors (KNN) enhances this collaborative paradigm, introducing a personalized touch to recommendations. In User-Based Collaborative Filtering, KNN identifies the most similar users to a target user, forming a cohort whose preferences guide recommendations. Alternatively, in Item-Based Collaborative Filtering, KNN identifies analogous items to the target item, suggesting items that align with users' past preferences. The choice of K, the number of nearest neighbors, profoundly influences the balance between accuracy and diversity in recommendations. By utilizing KNN, recommendation systems can intelligently tap into the collective preferences of users or items that share commonalities, providing users with tailored and relevant suggestions.

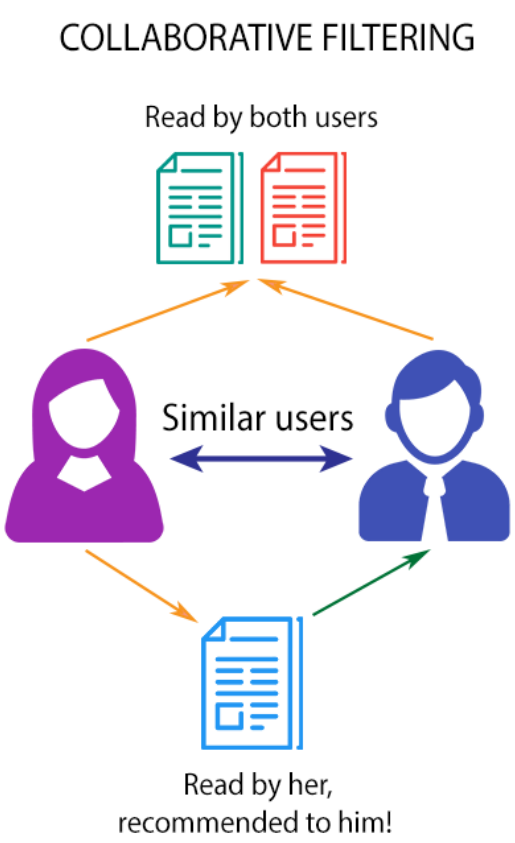

Fig. 1: Collaborative Filtering.

B. Collaborative Filtering based on Matrix Factorization

Matrix Factorization Based Collaborative Filtering is a technique commonly used in recommender systems to predict user preferences for items based on past interactions. Traditionally, Low Rank Matrix Approximation methods assume a fixed rank for the user and item feature matrices used in Matrix Factorization. However, recent research suggests that within the rating matrix, different rank sub-matrices may co-exist. This means that a fixed rank feature matrix may not accurately capture the complex patterns present in the data, leading to decreased accuracy in recommendation predictions. To address this limitation, a new approach called Mixture Rank Matrix Approximation (MRMA) has been proposed. MRMA characterizes user/item ratings as a combination of different fixed rank matrices. Specifically, it acknowledges that sub-matrices containing users and items with fewer ratings may have a lower rank, while sub-matrices with more ratings may have a higher rank. This approach is implemented using a probabilistic model with Gaussian noise to model the ratings. [5] The model parameters are learned using an iterated conditional modes (ICM) procedure, which iteratively updates the parameters to maximize the likelihood of the observed ratings. Matrix factorization serves as a collaborative filtering approach aimed at uncovering the associations between user and item entities. By identifying latent features within matrices representing users and items, this method seeks to assess similarity and provide predictions based on both user and item characteristics. The generation of user and item matrices through matrix factorization occurs by minimizing the mathematical cost function, Root Mean Square Error (RMSE). Gradient descent serves as a technique employed to minimize this cost function, facilitating the process of matrix factorization. The results of experiments conducted using the MRMA method demonstrate significant improvements in recommendation accuracy compared to traditional Low Rank Matrix Approximation methods. By allowing for the existence of multiple rank sub-matrices within the rating matrix and capturing the complex patterns present in the data, MRMA provides more accurate predictions, leading to enhanced performance in recommender systems.

C. Similarity-Based Collaborative filtering

Similarity measures play a crucial role in Collaborative Filtering-based Recommender Systems by quantifying the likeness or similarity between users or items based on their past interactions or attributes. These measures enable the system to identify users or items with similar preferences or characteristics, facilitating accurate recommendations. Common similarity measures include cosine similarity, Pearson correlation coefficient, and Jaccard similarity. Cosine similarity measures the cosine of the angle between two vectors, indicating their similarity in direction. [4] Pearson correlation coefficient assesses the linear correlation between two variables, reflecting their degree of association. Jaccard similarity calculates the intersection over the union of sets, determining the proportion of shared items between users. The performance of similarity measures relies primarily on the quality of the data being used. Consequently, the greater the richness and density of the data, the better the performance of the similarity measures. By employing appropriate similarity measures, Collaborative Filtering-based Recommender Systems can effectively identify relevant users or items, enhancing the accuracy of recommendations.

D. LlamaRec: Two-Stage Recommendation using Large Language Models for Ranking

LlamaRec [6], a novel framework designed to improve the performance and efficiency of recommendation systems by integrating large language models (LLMs) into a two-stage recommendation process. Initially, LlamaRec uses small-scale sequential recommenders to generate a candidate set of items based on user interaction history. This approach is innovative because it carefully constructs a prompt that includes both the user's interaction history and the candidate items, which is then fed into an LLM. The key innovation lies in the use of a verbalizer-based method to interpret the LLM's output logits, converting them into a probability distribution over the candidate items. This method deviates from traditional autoregressive text generation approaches, offering a more efficient and effective way to rank items without the need for extensive text generation.

What sets LlamaRec apart is its ability to leverage the generative capabilities of LLMs while significantly reducing the computational overhead typically associated with these models in recommendation systems. The framework demonstrates its novelty by showing that it can outperform existing methods on benchmark datasets, both in terms of recommendation accuracy and computational efficiency. The use of a verbalizer-based approach for converting logits into item probabilities is particularly novel, as it allows for the direct application of LLMs in ranking scenarios without the complexities and inefficiencies of generating and processing large amounts of text. This makes LlamaRec a promising solution for real-time recommendation scenarios where both accuracy and response time are critical.

IV. Research Plan

This research plan is designed to undertake a thorough investigation into the landscape of recommendation systems, focusing particularly on collaborative filtering, content-based filtering, and hybrid models. The literature review will meticulously categorize and examine a multitude of recommendation system studies sourced from reputable publications, scrutinizing their methodologies, algorithms, and performance metrics. The emphasis will be on identifying the strengths and weaknesses of each approach, considering factors such as accuracy, diversity, scalability, and computational efficiency. The methodology section will articulate the criteria for comparison,

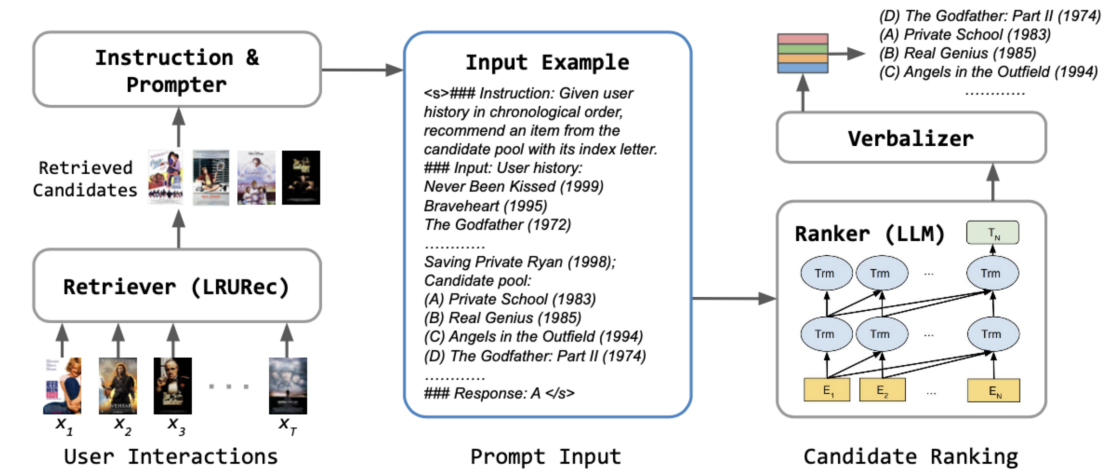

Fig. 2: The left subfigure illustrates the retrieval stage that generates candidate items with LRURec. Using an instruction template, we transform user history and candidates into text for ranking via Llama 2 (right subfigure).

choose a benchmark dataset representative of diverse user preferences, and outline the evaluation metrics. In the implementation phase, the chosen recommendation algorithms will be applied to the dataset, and the entire process will be meticulously documented. The results and analysis section will present detailed findings, drawing insightful comparisons with existing studies and highlighting any identified limitations. Importantly, this research plan envisions not only evaluating existing systems but also proposes the development of a novel recommendation approach. This innovative approach will be informed by the comparative evaluation insights, aiming to address potential gaps and enhance the current state of recommendation systems.

V. Evaluation Plan

A. Baseline Model:

Collaborative Filtering: This is one of the most commonly used techniques for recommendation systems. It works by making automatic predictions (filtering) about the interests of a user by collecting preferences from many users (collaborating).

B. Evaluation Metric:

The Root Mean Squared Error (RMSE) is a widely used metric for assessing a model's accuracy in predicting numerical data. It is calculated as the square root of the average of the squared differences between predicted and actual values. In the realm of recommendation systems, RMSE helps quantify the disparities between the ratings predicted by the model and the ratings provided by users.

$$RMSE = \sqrt{\frac{1}{N}\Sigma_{i=1}^{N}\left(y_i - \hat{y}_i\right)^2}$$

Where N is the number of samples, y_i is the predicted value and $\hat{y}_i$ is the actual value.

VI. Project Timeline: Tasks, Descriptions, Deadlines

TABLE I: Task List

Task	Description	Deadline
Literature review	Research related papers, explore possible algorithms, identify sub-problems and create specific tasks	Mar 12
Data Processing	Collection and preprocessing of the data	Mar 15
Algorithm Design	Finalize the algorithms to be used	Mar 16
Implementation	Implement the movie recommender system using the chosen algorithms	Mar 30
Project Presentation	Compose the project presentation including all important points	Apr 2
Final Code	Make final code changes and test the code. Start preparing for project demo	Apr 14
Project Report	Prepare the final project report	Apr 28

VII. Division of Work

- Literature Review - All group members
- Data Processing - Anushka and Yashi
- Algorithm Design - Yash and Kuriakose and Rohan
- Implementation - All group members
- Project Presentation - All group members
- Final code changes - All group members
- Project Report - All group members

References

[1] Netflix. Netflix prize data set. 2009.
[2] Sami Abu-El-Haija, Joonseok Lee, Max Harper, and Joseph Konstan. Movielens 20m youtube trailers dataset. In MovieLens, 2018.
[3] Nguyen, Luong Vuong, Quoc-Trinh Vo, and Tri-Hai Nguyen. "Adaptive KNN-Based Extended Collaborative Filtering Recommendation Services." Big Data and Cognitive Computing 7.2 (2023): 106.
[4] Fethi Fkih. Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison. Journal of King Saud University, Volume 34, Issue 9, October 2022.
[5] Christie Natashia, Recommender System: Collaborative Filtering with Matrix Factorization. 2023
[6] LlamaRec: Two-Stage Recommendation using Large Language Models for Ranking, Zhenrui Yue and Sara Rabhi and Gabriel de Souza Pereira Moreira and Dong Wang and Even Oldridge 2023