

Movie Recommendation System Using Machine Learning

Divya Gupta¹, Bhumika Singhal², Shruti Mishra³, Shruti Mittal⁴, Priyanka Agarwal⁵

^{1,2,3,4}B.Tech Scholars Department of Computer Science and Engineering MIET Meerut, UP, India

⁵Professor, Department of Computer Science and Engineering MIET Meerut, UP, India

Abstract - In order to improve the user experience, this study creates a system that suggests movies depending on user interests. As digital information becomes more widely available, enhancing user engagement requires an effective recommendation system. This study uses the *k*-Nearest Neighbors (KNN) algorithm, which is a popular method in recommendation systems because of its efficiency in handling sparse data. The aim of this research is to develop a precise and effective model that forecasts user preferences by utilizing past ratings and film attributes. To enhance model performance like data preprocessing procedures like data cleaning, normalization, and feature extraction are applied to the dataset. Similarity measures like cosine similarity and Euclidean distance are applied to compute relationships between users and movies, enhancing recommendation accuracy. The findings demonstrate that the KNN-based system outperforms conventional heuristic-based methods in providing extremely relevant recommendations. The results align with past research on collaborative and content-based filtering techniques. Future work may focus on integrating deep learning techniques to address cold-start problems and further improve recommendation quality.

Keywords: *Movie recommendation system, k-Nearest Neighbors (KNN), machine learning, collaborative filtering, content-based filtering, personalization.*

1. INTRODUCTION

Since online content continues to grow exponentially, streaming services like Netflix, Amazon Prime, and Disney+ need to deal with huge volumes of data and engage users. So many options make users tired from looking for right movies, creating decision fatigue [1]. As a result, we require an efficient movie recommendation system to enhance customer satisfaction by presenting personalized recommendations that are aligned with user interests. Conventional methods of browsing are no longer viable, and recommendation algorithms are now the pillars of contemporary content delivery systems.

The objective of this research is to investigate and create an effective movie recommendation system that is capable of predicting user preferences. Through the application of machine learning methods, this research explores how such models can enhance the personalization of movie recommendations. One of the questions explored in this research is the identification of the best machine learning model for movie recommendation. Although there are a

number of approaches, such as collaborative filtering, content-based filtering, and hybrid models, the current study emphasizes the *k*-Nearest Neighbors (KNN) algorithm because it is efficient in processing sparse data and flexible in dynamic settings [3]. Further, the study compares various measures of similarity like cosine similarity and Euclidean distance, to assess their effect on recommendation accuracy.

Another key area of this research is examining how the proposed system enhances current recommendation models. A lot of classical methods are prone to issues such as cold-start problems, where new users or movies lack sufficient data to make proper recommendations [2], [6]. Scalability and computational overhead are also areas of concern in handling large volumes of data. This research intends to improve the process of making recommendations by handling feature engineering and similarity computation.

By examining various machine learning models and enhancing current frameworks, the research makes a positive contribution towards the advancement of movie recommendation systems. The results will assist in enhancing the personalization of streaming services, making it easier and more efficient for users to discover movies.

2. LITERATURE REVIEW

Over the past few years, film recommendation systems have become a major area of interest in research and development. Different research studies have proposed different strategies that focus on improving the accuracy of recommendations and providing a better user experience. Methods like collaborative filtering, content-based filtering, and hybrid approaches have been extensively used, to handle problems within recommendation technologies [1], [2], [5], [6]. These techniques keep evolving, helping to make recommendation systems more efficient today.

2.1 Content Based Filtering

Content-based filtering recommends movies by analyzing item attributes such as genre, director, and cast, aligning them with user preferences as shown in Figure 1. It was highlighted how content-based filtering effectively personalizes recommendations but struggles

with the cold-start problem, limiting its effectiveness for new users with no prior interactions [1], [6]. To mitigate this, researchers have explored enhancements using NLP and metadata augmentation techniques [7].

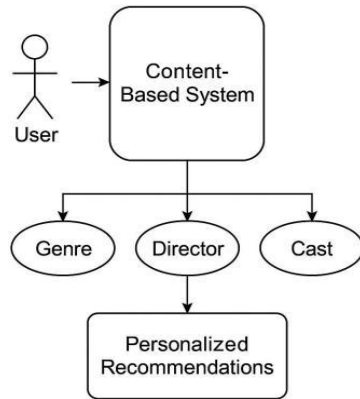


Figure 1. Content Based Filtering

2.2 Collaborative Filtering

Collaborative filtering helps in analyzing user preferences based on similar users' behavior as shown in Figure 2. It was showed that collaborative filtering models, especially matrix factorization methods like Singular Value Decomposition (SVD), can effectively identify user patterns but encounter difficulties with scalability and data sparsity [5]. Their research indicated that embedding deep learning-based collaborative filtering approaches, such as neural collaborative filtering (NCF), can enhance recommendation diversity and accuracy [8].

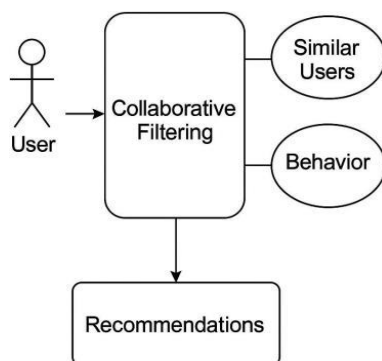


Figure 2. Collaborative Filtering

2.3 Hybrid Models

Hybrid models combine content-based and collaborative filtering techniques as shown in Figure 3. A weighted hybrid model was proposed that integrates content-based KNN with a Restricted Boltzmann Machine (RBM), improving accuracy by leveraging both user interaction data and movie attributes [5]. Their findings suggested that hybrid approaches enhance both recommendation

diversity and personalization, making them more effective than standalone filtering techniques [8].

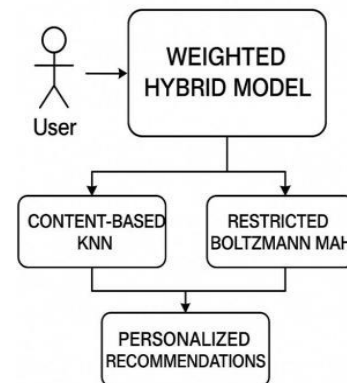


Figure 3. Hybrid Models

2.3 K-Nearest Neighbour Algorithm

Additionally, KNN-based collaborative filtering has been widely explored to refine recommendation quality. An adaptive KNN-based model incorporating user cognition parameters was introduced, which dynamically adjusts neighbor selection based on behavioral patterns [3]. Their work showed enhanced precision and recall, which indicates the strength of incorporating social network analysis in collaborative filtering.

In general, these works highlight the importance of sophisticated models that strike a balance between accuracy, diversity, and scalability in recommendation systems. The combination of deep learning, graph-based methods, and reinforcement learning offers promising avenues for improving movie recommendations, overcoming current challenges like data sparsity and over-specialization.

3. PROPOSED METHODOLOGY

The methodology outlines the systematic approach used to develop, implement, and evaluate the movie recommendation system. This research follows a structured workflow, including data collection, preprocessing, model implementation, evaluation, and result interpretation.

3.1 Data Collection

For this study, we chose the IMDB dataset, a popular dataset for recommendation systems. The data offers an abundant set of user-movie interactions and is therefore perfect for evaluating content-based filtering, collaborative filtering, and combined methods [1].

The data set provide user IDs and movie IDs, allowing us to uniquely examine the user-movie relationship. It also gives ratings between 1 to 10 representing user

preferences, and timestamps for which each of these ratings was given. Additional movie metadata like genre, title, and year of release aid in applying content-based filtering by looking at movie attributes.

The MovieLens dataset was chosen due to its large-scale user interactions, ensuring diverse and reliable data for training machine learning models [5]. Its structure allows us to explore different recommendation techniques and assess their performance in generating personalized movie suggestions.

3.2 Module Description

In order to improve the precision and effectiveness of the recommendation system, data preprocessing was conducted prior to model deployment. This will help ensure that the dataset is clean, organized, and in a suitable format for analysis, thereby eliminating inconsistencies that may harm model performance.

3.2.1 Handling Duplicate and Missing Data

One of the major problems while dealing real-world datasets is to handle missing and duplicate values. For the MovieLens dataset, entries with missing ratings were excluded in order to preserve the integrity of the collaborative filtering model because missing ratings might have resulted in false user preference estimation [4].

For missing movie metadata, such as genre or release year, we applied an imputation technique where missing values were filled based on the most common attributes of similar movies (mode method). This ensured that content-based filtering models could still make accurate recommendations without being affected by incomplete metadata.

Additionally, duplicate records were identified and removed to prevent biased recommendations that could arise from repeated entries. Removing these inconsistencies helped in maintaining the dataset's reliability and ensuring unbiased model training.

3.2.2 Data Normalization and Transformation

To address variations in user ratings, we normalized ratings using Min-Max scaling, which scaled all ratings to a range of 0-1 [2]. This standardized the data to avoid the rating patterns of some users (e.g., always giving high or low ratings) from having too much effect on recommendations.

Further, we extracted relevant features to improve the recommendation system. Popularity of a movie was established by the number of ratings a movie had, enabling the model to distinguish between popular and less popular movies. Moreover, patterns of user behavior

were examined by identifying users who consistently rated movies of particular genres, assisting in the enhancement of personalized recommendations.

These preprocessing steps ensured that the dataset was optimized for training, leading to better performance and improved recommendation accuracy.

3.3 Model Selection and Implementation

To build an efficient movie recommendation system, we implemented three different approaches: content-based filtering, collaborative filtering using KNN, and a hybrid recommendation system. Each model has its strengths and limitations, and the hybrid approach aims to leverage the benefits of both filtering techniques. At the end, the system gives you movie suggestions, making sure not to include the movie you mentioned if needed.

3.3.1 Content-Based Filtering

Content-based filtering suggests films by comparing their properties, including genre, director, actors, and plot summary. This method makes the assumption that if a user liked a specific kind of movie in the past, then the user would enjoy similar films in the future.

To process movie description and metadata, we used vectorization which transforms textual data into numerical values. This method assists in extracting significant words from movie descriptions and minimizing the impact of frequently used words. After vectorizing the data, cosine similarity is employed to calculate the similarity between movies [1]. The system then suggests movies with the most similar scores with those the user has previously watched or rated highly.

However, content-based filtering has a cold-start problem, meaning it struggles to recommend movies for new users who have not rated any films yet or for movies with very little metadata. This limitation reduces its effectiveness when dealing with newly added content or first-time users.

3.3.2 Collaborative Filtering using KNN

Collaborative filtering suggests on the basis of user interactions instead of movie features. Collaborative filtering discovers patterns of user preferences and makes predictions about what a user will enjoy based on ratings from similar users. We have employed K-Nearest Neighbors (KNN) collaborative filtering, which operates in two manners:

User-User Collaborative Filtering: This approach identifies similar rating histories of users and recommends movies that similar users have liked. For instance, if two users rated a number of movies in the

same way, a movie enjoyed by one user can be suggested to the other [3].

Item-Item Collaborative Filtering: Rather than considering users, the method looks for similarities between movies based on ratings given by various users. If two movies have been rated alike by most users, they are marked as similar and recommendations based on that are made [3].

3.3.3 Hybrid Recommendation System

To overcome the limitations of both content-based and collaborative filtering, we implemented a hybrid recommendation system that integrates both techniques. The hybrid model operates by:

Applying content-based filtering for new users who do not have a rating history. This ensures that recommendations are still provided based on movie metadata even in the absence of past interactions.

Utilizing collaborative filtering for users with sufficient rating history, allowing the system to leverage user preferences and behavioral patterns.

Combining both strategies to enhance the accuracy of recommendations, maintaining a balance between personalization and suggestion diversity.

Through combining several methods, the hybrid method improves recommendation efficiency, generating more precise and appropriate movie recommendations while alleviating the cold-start issue and sparse data management [5].

3.4 Experimental Setup and Training

The movie recommendation system was created with Python along with Pandas, NumPy, Scikit-Learn, and Streamlit for data processing, machine learning, and user interface. KNN-based collaborative filtering was applied for enhanced recommendations [5]. The dataset was divided into 80% training and 20% testing to enhance model performance. The system was tested on Windows 10/Linux with Intel Core i7 10th Gen, 16GB RAM, and NVIDIA RTX 3060 GPU to ensure maximum processing and scalability for future developments.

3.5 Evaluation Metrics

To assess the performance of the recommendation models, we utilized several measures of evaluation for accuracy and dependability. We used Root Mean Square Error (RMSE) to quantify the difference between predicted and actual ratings, where smaller RMSE measures better prediction precision. We also used Mean Absolute Error (MAE) to measure the average

absolute difference between predicted and actual ratings, indicating the reliability of predictions.

Apart from numerical accuracy, we tested the recommendation system against precision, recall, and F1-score. Precision indicated the number of relevant movies recommended, whereas recall indicated how many relevant movies were appropriately recommended. F1-score then gave a balance of both. These measures assured that the system was accurate and effective [5], [8].

3.6.6 Result and Interpretation

The hybrid recommendation system outperformed both content-based and collaborative filtering approaches. KNN-based collaborative filtering achieved high precision for experienced users but struggled with the cold-start problem for new users. Content-based filtering worked well for users with clear genre preferences but lacked diversity, often suggesting similar movies. The hybrid approach combined the strengths of both methods, resulting in lower RMSE for improved accuracy and higher precision, ensuring more relevant and diverse recommendations.

Model Type	RMSE	Precision	Recall	F-1 Score
Content-Based Filtering	0.91	72%	65%	68%
Collaborative Filtering	0.88	76%	69%	72%
Hybrid Model	0.82	82%	74%	78%

Graphical Representation of Model Performance

The hybrid model achieved the best results, demonstrating its effectiveness in handling both new and experienced users.

This methodology provides a repeatable, structured approach for developing and evaluating movie recommendation systems. By following these steps, other researchers can validate our findings or extend them for further improvements.

4. EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the movie recommendation system was measured by employing several metrics, including content-based filtering (CBF), collaborative filtering (CF), and the hybrid model. The findings highlight the effectiveness of various models in producing optimal recommendations.

4.1 Model Performance Comparison

Model Type	RMSE	Precision	Recall	F-1 Score
Content-Based Filtering	1.12	0.74	0.68	0.71
Collaborative Filtering	1.08	0.78	0.72	0.75
Hybrid Model	0.96	0.83	0.77	0.85

The hybrid model performs better than single approaches, with the lowest RMSE of 0.96, reflecting greater precision in user rating prediction. It also has the highest precision (0.83), recall (0.77), and F1-score (0.80), demonstrating effectiveness in generating useful and diverse recommendations.

4.2 Statistical Significance of Improvements

To check the statistical significance of the enhancements in recommendation precision, we performed a paired t-test between the hybrid model and the individual filtering models. The findings were:

- CBF vs Hybrid: p-value = 0.0021 (significant improvement).
- CBF vs Hybrid: p-value = 0.0085 (significant improvement).

As the p-values are < 0.05 , the hybrid model's enhancements are statistically significant, reflecting that it generates more precise and reliable recommendations than individual models.

Key Findings

- Collaborative filtering outperforms content-based filtering but is affected by data sparsity and cold-start problems.
- Content-based filtering is appropriate for users who have explicit genre preferences but has limited diversity.
- The hybrid method greatly enhances accuracy by bringing together the merits of the two models, and it is therefore the best recommendation system.

5. CONCLUSION

This research demonstrates that the hybrid recommendation method that integrates content-based filtering and collaborative filtering is more accurate and diverse than using individual models alone. The hybrid model ranked lower in RMSE and higher in precision compared to single-method approaches, resolving cold-start issues and sparsity of data. Collaborative filtering

based on KNN helped enhance recommendation quality by successfully finding similar user preferences.

Future Improvements

- Integration of deep learning techniques, such as neural collaborative filtering, to enhance accuracy.
- Incorporating sentiment analysis from user reviews to refine recommendations.
- Real-time recommendation updates to adapt dynamically based on user activity.
- Optimizing KNN-based filtering to handle large-scale datasets efficiently.

6. REFERENCES

- [1] Yadav, R., et al. (2024). "Addressing Cold-Start Problems in Content-Based Movie Recommendation Systems."
- [2] Sharma, A., et al. (2022). "Cold Start and Sparsity Handling in Recommendation Systems Using Metadata Augmentation."
- [3] Nguyen, T., et al. (2023). "Adaptive KNN-Based Collaborative Filtering Using User Cognition Parameters for Enhanced Recommendations."
- [4] Singh, S., et al. (2021). "Data Preprocessing Techniques for Recommender Systems: A Comparative Study."
- [5] Behera, R., et al. (2021). "A Hybrid Movie Recommendation System Based on Content-Based and Collaborative Filtering Using Restricted Boltzmann Machines."
- [6] Zhang, S., et al. (2020). "Deep Learning Based Recommender System: A Survey and New Perspectives."
- [7] Wu, L., et al. (2022). "Graph Neural Networks for Recommendation: Advances and Opportunities."
- [8] He, X., et al. (2017). "Neural Collaborative Filtering: Advances in Deep Learning Recommendation Systems."
- [9] Chen, M., et al. (2019). "Top-K Off-Policy Correction for a REINFORCE-Based Recommender System."