Advances in Computational Design, *Vol. 10, No. 1 (2025) 91-109*
https://doi.org/10.12989/acd.2025.10.1.091

An intelligent hybrid recommendation system for enhancing viewer experience

Kulvinder Singh*, Sanjeev Dhawan[a] and Manoj Yadav[b]

Department of Computer Science & Engineering, University Institute of Engineering & Technology (U.I.E.T), Kurukshetra University, Kurukshetra, Haryana, India

(Received March 8, 2024, Revised April 14, 2024, Accepted February 4, 2025)

Abstract. We introduce HybridRecSys, a hybrid recommendation system that integrates collaborative filtering (CF) using enhanced Singular Value Decomposition (SVD) and advanced content-based (CB) filtering techniques enriched by Natural Language Processing (NLP). The proposed system addresses critical challenges such as sparsity and cold start by leveraging a dual approach: explicit ratings for strong user profiling and implicit preferences derived from content and genre analysis. Novel contributions include the application of weighted cosine similarity alongside RBF and cosine similarity, significantly improving similarity metrics. Experimental validation on IMDb and Netflix datasets demonstrates superior performance, with HybridRecSys achieving RMSE and MAE scores of 0.6991 and 0.6987 on IMDb, and 0.2364 and 0.2357 on Netflix, respectively. The system outperforms existing methods by efficiently addressing sparsity and cold start challenges, ensuring highly personalized and accurate recommendations.

Keywords: collaborative filtering; content-based filtering; enhanced SVD; HybridRecSys; temporal dynamics

1. Introduction

The exponential growth of online platforms has necessitated the development of sophisticated recommendation systems (RSs) to enhance user experience and engagement. While traditional RSs based on collaborative filtering (CF) and content-based (CB) filtering (Li *et al*. 2024) have shown effectiveness, they often struggle with challenges like sparsity, cold start, and dynamic user preferences. To address these issues, hybrid systems combining CF and CB techniques have gained prominence, leveraging the strengths of both approaches.

Recommendation systems have become indispensable tools across various industries, including social media, online shopping, and entertainment. Their primary objective is to provide consumers with relevant and personalized recommendations, considering their preferences, behavior, and item attributes (Felfernig *et al*. 2024). These systems are pivotal in enhancing user experiences and increasing user engagement.

*Corresponding author, Ph.D., E-mail: ksingh2r015@kuk.ac.in
[a] Ph.D., E-mail: sdhawan2015@kuk.ac.in
[b] Ph.D. Student, E-mail: manoj200.yadav@gmail.com

http://www.techno-press.org/?journal=acd&subpage=7 ISSN: 2383-8477 (Print), 2466-0523 (Online)

By offering tailored suggestions, they can significantly boost user satisfaction and loyalty. Users presenting relevant and captivating information are more likely to interact with a platform for extended periods, fostering increased user loyalty and platform success. RSs can facilitate unexpected and delightful discoveries, presenting users with novel and captivating content or knowledge they may not have encountered otherwise. This enhances user satisfaction and evokes a sense of astonishment and exploration, motivating users to delve deeper into the platform or application (Rajput *et al*. 2023).

Given the vast data and the growing prevalence of online platforms, recommendation systems have become indispensable for consumers, content producers, and companies. They help consumers navigate the vast information landscape and enhance company success and profitability by increasing user engagement and driving sales. With the progress of technology and the increasing availability of data, the demand for efficient RSs will continue to grow. RSs are now an essential part of current information retrieval and personalization systems. Traditional recommendation systems can be classified into two main categories: content-based and collaborative filtering methods (Gao *et al*. 2024). Collaborative filtering utilizes user-item interactions to identify patterns and offer suggestions based on similar users' preferences. Content-based filtering analyzes the characteristics and attributes of products to recommend preferences based on user preferences.

Hybrid recommendation systems have garnered significant interest due to their potential to overcome traditional systems' limitations effectively (Bahrani *et al*. 2024). By combining multiple recommendation strategies, these hybrid systems can harness the strengths of each approach and mitigate their weaknesses, offering a promising solution for the future of recommendation systems (Kilanioti *et al*. 2019). The inability of current recommendation systems to handle sparsity and cold start issues prevents them from offering precise and pertinent recommendations. More data is needed for traditional collaborative filtering approaches to forecast preferences for new users or products. Similarly, while helpful, content-based filtering techniques may only partially utilize user interaction data. These single-approach systems should notice the dynamic nature of user preferences and implicit evaluations, which might offer deeper insights into user behavior. A more capable and flexible recommendation system that can easily combine different data sources and analytical methods is required.

By fusing enhanced SVD and content-based filtering with sophisticated collaborative filtering, HybridRecSys fills these gaps and improves its capacity to handle sparsity and cold start issues. Integrating user profiling and similarity metrics produces effective and impactful predictions, even for newly added individuals and things. The hybrid technique employs natural language processing to analyze implicit data from movie themes and genres and temporal dynamics to capture changing user preferences. Additionally, the integration of the Netflix popular movie datas*et al*ongside the IMDB datas*et al*lows for a more comprehensive analysis, utilizing these multiple data sources. This integration greatly raises the system's accuracy by enabling the extrapolation of unknown ratings from known ones. Moreover, by incorporating weighted cosine similarity in addition to Cosine and RBF, the system's precision and robustness in similarity measurements are significantly enhanced. HybridRecSys stands as a novel solution to the shortcomings of current recommendation systems due to these advances, which guarantee more relevant and exact recommendations.

This paper presents a comprehensive review of deep learning-based RSs and highlights the originality of HybridRecSys in addressing sparsity and cold start challenges. By integrating datasets from IMDb and Netflix, the system's efficacy is validated through rigorous experimental

evaluation, demonstrating its superiority over existing methods. The objectives of this research are aligned as follows:

• Development of HybridRecSys, combining advanced CF and CB techniques for improved recommendation accuracy and robustness.

• Integration of temporal dynamics and NLP for detailed user profiling.

• Comparative analysis with existing systems using the IMDb and Netflix datasets.

• To develop an approach to overcome issues faced by a single approach recommender system by solving the sparsity problem.

• To design and implement a collaborative filtering algorithm using enhanced SVD for improved user-item interaction predictions.

• To develop advanced content-based filtering methods that effectively utilize item attributes and user profiles.

• To integrate weighted cosine similarity, along with cosine and RBF, to improve the accuracy and robustness of similarity measurements within the hybrid approach.

• To integrate user profiling with a hybrid approach to derive more accurate ratings for the system and extrapolate unknown ratings from the known ones and information from surroundings by using implicit rating of data.

• To provide a comparative analysis amongst HybridRecSys with different similarity metrics and existing frameworks, establishing the novelty and significance of our proposed system.

2. Background work

This section explores previous research in recommendation systems, their contributions, and limitations.

2.1 Deep learning methods used in recommendation systems

Over time, deep learning approaches have significantly improved recommendation systems by providing cutting-edge ways for identifying intricate patterns and producing precise recommendations. Neural network and matrix factorization deep learning techniques are often used in recommendation systems (Gündoğan *et al*. 2022). Moreover, Da'u *et al*. (2019) explores various kinds of neural networks used in recommendation systems and their limitations.

Neural networks have successfully modeled intricate user-item interactions and non-linear correlations. Convolutional neural networks (CNNs) and multi-layer perceptron (MLP) techniques have been effectively used in recommendation systems. Antony Rosewelt *et al*. (2020) explores a take on the performance of CNNs in context-based recommendation systems. MLPs' ability to learn high-level representations of user preferences and item properties makes accurate predictions and tailored recommendations possible. CNNs, on the other hand, are particularly advantageous for recommendation tasks requiring sequences, such as clickstream or session-based recommendations, since they are well-suited for collecting spatial and temporal patterns in sequential data. Jannach *et al*. (2020) explains why deep learning models must consistently provide fruitful results in recommendation systems.

Another popular deep learning technique in recommendation systems, matrix factorization, seeks to break down the user-item interaction matrix into low-dimensional latent components, as

deciphered in Koren *et al*. (2009) and Yu *et al*. (2014). Meaningful representations of users and objects can be analyzed using methods like Singular Value Decomposition (SVD) and autoencoders. The use of multiple datasets, including the IMDB and Netflix popular movie datasets, has been explored to enhance the diversity and robustness of these representations. In particular, autoencoders have demonstrated promise in reconstructing the input data to develop rich representations (Zhang *et al*. 2020), which may be used to provide customized recommendations based on learned user and object embeddings (Xue *et al*. 2017).

Deep learning techniques utilize neural networks and matrix factorization to identify patterns, model user preferences, and provide tailored recommendations for consumers.

2.2 Machine learning models used in recommendation systems

2.2.1 Content-based filtering RSS (CBF):

By comparing the user profile and item description, CBRS uses CBF to propose products (Son *et al*. 2017). The recommendation system thus suggests products comparable to previous records of user preferences. The similarity of items is determined based on their attributes. Hannech (Hannech *et al*. 2017) concludes that if a users' primary contextual similarity does not match well then system provide recommendations based on the second best choices available.

2.2.2 Content-based filtering RSS (CBF):

Several research works have explored collaborative filtering in RSs, such as (Bobadilla *et al*. 2011, Ekstrand 2011, Schafer *et al*. 2007). They critically surveyed the methodology, its limitations, and future research directions. Performance of collaborative techniques suffers from sparsity problem. Zhao *et al*. (2022) proposes a novel item-based method of collaborative filtering for enhancing prediction accuracy while working with a sparsity of data. Based on the experimental outcomes, better performance than the current approaches and the successful handling of the data sparsity issue of the applicated research is determined at the proposed strategy.

2.2.3 Demographic RSS:

Based on user demographics like age, education, employment, location, etc., demographic operations are based. Clustering algorithms group target customers into specific categories based on demographic data. However, the same selection of things will be recommended if the user's demographic characteristics do not change over time. They might thereby overlook a fresh and valuable recommendation. The accuracy of RS can be increased by knowing a user's demographics (Wang *et al*. 2012).

2.2.4 Knowledge-based RSS:

Knowledge-based RSs provide recommendations based on connections and similar preference probability between the user and goods. Case-based reasoning in knowledge-based RSs divides the user's requirements into multiple instances based on various criteria and provides recommendations that closely match the user's expected selection (Tarus *et al*. 2018). Constraint-based RS is a different kind of KBRS that adapts to the user's preferences and only suggests products that fit those preferences (Martínez *et al*. 2008). During the nonavailability of an easily accessible user preference item, a set of close equivalent items is suggested. Using semantic web technologies, a user's preference knowledge base with a range of viewpoints is built using ontologies.

2.2.5 Hybrid RSS:

Hybrid Recommendation Systems combine several models' characteristics to achieve an admirable result. Burke (2002), Ghazanfar *et al*. (2010) and Çano *et al*. (2017) analyze different hybrid approaches used in RSs over time. Li and Kim (Qing Li *et al*. 2003) stresses on a clustering-based hybrid system on the MovieLens dataset, while Zhang *et al*. (2015) has combined random forest and k nearest neighbors in its hybrid model. Additionally, the integration of weighted cosine similarity along with cosine and RBF within these hybrid models has shown to improve the accuracy and robustness of recommendations, particularly when dealing with diverse datasets like IMDB and Netflix popular movie datasets.

2.3 limitations

The availability of high-quality and diversified data is a significant restriction. Large datasets with enough user-item interactions, ratings, and contextual data are needed to build accurate and reliable recommendation systems (Cacheda *et al*. 2011). Such datasets can be difficult because of user feedback issues, data access, and privacy. A lack of data might make it difficult to identify user preferences and produce trustworthy recommendations effectively. The creation of methods for data collecting, data augmentation, or utilizing auxiliary data sources to improve the quality and variety of the dataset is required to address data restrictions. The cold-start problem arises when there is insufficient data to generate recommendations for new users or items. Recommendation systems usually use user input.

Creating data collection and augmentation methods, using contextual data, and experimenting with alternate recommendation systems to overcome these constraints is a fruit-providing stratagem (da Silva *et al*. 2023). By overcoming these restrictions, recommendation systems can develop more reliable, accurate, and flexible to accommodate a variety of user demands.

3. Proposed system – HybridRecSys

The architecture of HybridRecSys, as depicted in Fig. 1, is explained as follows:

3.1.1 Dataset collection

We collect datasets from IMDb and Netflix, with diverse user-item interaction data.

3.1.2 Preprocessing

Data cleaning involves handling missing values, normalization and one hot encoding of categorical features. NLP techniques are used to process movie overviews such as tokenization, stop word removal and TF-IDF vectorization.

3.1.3 User profiling

Explicit ratings and implicit preferences are combined to create comprehensive profiles. Explicit feedback is average user ratings, implicit feedback is genres and movie content by NLP.

3.1.4 Hybrid recommendation:

3.1.4.1 Collaborative filtering with enhanced SVD

User preferences are integrated to capture temporal dynamics. To predict missing ratings and generate recommendations, matrix factorization is used.

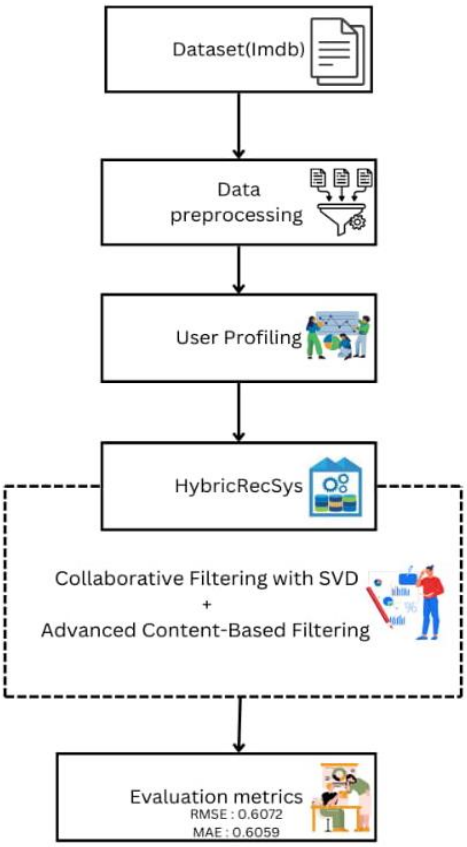

Fig. 1 Architecture of HybridRecSys

3.1.4.2 Advanced content-based filtering

TF-IDF and weighted cosine similarity are used to evaluate content similarities, which capture thematic and stylistic elements.

3.1.5 Evaluation

RMSE, MAE and similarity scores are used to measure performance, and robust validation across datasets is ensured.

4. Methodology

4.1 Dataset collection

The experimentation utilizes two primary datasets: the IMDB dataset (Johari *et al*. 2021) and the Netflix Popular Movie Dataset. The IMDB dataset, containing 50,000 highly polarized movie reviews, is split equally for training and testing purposes. This dataset is chosen for its significant size and valuable information, making it ideal for binary sentiment classification tasks. It provides a mixture of unprocessed text and pre-processed bag-of-words formats, which adds versatility to

the preprocessing and feature extraction processes, accommodating different experimental requirements. Additionally, the inclusion of extra unmarked data offers opportunities for semi-supervised learning techniques, further enhancing model performance.

In addition to the IMDB dataset, the Netflix Popular Movie Dataset is incorporated to diversify the data sources and test the system's robustness across different platforms. This dataset contains information on popular movies, including user ratings, genres, and summaries, which is valuable for training and validating the hybrid recommendation system. The combination of these datasets ensures that HybridRecSys is tested on a comprehensive and widely recognized benchmark, boosting the credibility and applicability of the findings.

4.2 Dataset preprocessing

The data preprocessing steps for our research encompass various crucial tasks to guarantee the dataset's cleanliness, coherence, and readiness for model training. The dataset is imported into a pandas DataFrame from a CSV file. To ensure the uniqueness of each record, duplicate rows are eliminated. Ratings that include the letters 'K' (for thousand) or 'M' (for million) are converted into numerical values by using the following:

$$Rating = \begin{cases} Rating\ X\ 1000, & if\ 'K' \\ Rating\ X\ 1000000, & if\ 'M' \\ Rating\ X\ 1, & otherwise \end{cases}$$

The duration is converted to minutes, and the year is converted into a numerical representation. NaN values in numerical columns are replaced with the average value using:

$$Missing\ Value\ = \frac{\sum_{i=1}^{N} x_i}{N} \tag{1}$$

The genres and directors are one-hot encoded. The overview is vectorized using the TF-IDF vectorizer.

TF – IDF(t, d, D) = TF(t, d) X IDF(t, D)

Here, t is the number of terms, d is the document, and D is the total number of records.

4.3 User profiling

User profiling in HybridRecSys entails developing comprehensive user profiles by considering explicit and implicit feedback. Every user is given a distinct identifier created by combining their director and genre preferences using label encoding. The user's explicit feedback is obtained by calculating the average rating of the movies they have watched, which is determined by taking the mean of their ratings. Implicit feedback refers to combining genre and content preferences through TF-IDF vectorization. Genre preferences are determined by averaging the TF-IDF scores of all movies the user watches. Similarly, the content preferences are determined based on the TF-IDF scores of the movie summaries.

$$\begin{aligned} Avg\ Rating_k &= \frac{1}{N_k}\sum_{i=1}^{N_k} Rating_{ki} \\ Genre\ Preference &= \frac{1}{N_k}\sum_{i=1}^{N_k} TF - IDF_{genre,ki} \end{aligned} \tag{2}$$

$$Avg\ Rating_k = \frac{1}{N_k}\sum_{i=1}^{N_k} TF - IDF_{overview,ki}$$

4.4 HybridRecSys

This subsection delves into the two individual recommendation systems trained on the dataset integrated into HybridRecSys.

4.4.1 Collaborative filtering with enhanced SVD

Collaborative filtering approaches are applied to identify the underlying patterns in the user-item interaction matrix. SVD is deterministically used to break down the matrix into latent components. SVD reduces the data's dimensionality, revealing hidden relationships between users and things. This allows the system to record how user preferences change over time accurately. The system can make more accurate recommendations that represent the user's interests at that moment by considering how the user's preferences vary. This temporal feature helps to preserve the relevancy of recommendations, especially when predicting the tastes of long-term users who may change. The missing ratings are forecasted, and recommendations are made to the users using low-rank approximations of the original user-item matrix derived using SVD. The SVD-based forecasts are one of the sources of recommendations in the collaborative filtering part of the hybrid RSs.

$$\mathrm{R} = \mathrm{U}.\mathrm{S}.\mathrm{V}^{\mathrm{t}}$$

Here,

R = User-item interaction matrix U = User Matrix

S = Diagonal singular value matrix V = Item matrix

Using matrix factorization, CF partitions the user-item interaction matrix R into three separate matrices: U (representing user preferences), S (a diagonal matrix holding singular values), and V^t (representing item attributes). With the help of this decomposition, recommendation algorithms can identify hidden patterns in user-item interactions, resulting in precise and individualized recommendations. Its capacity to forecast missing values in the interaction matrix presents an important strategy for creating effective and efficient recommendation systems.

The enhanced SVD approach decomposes the user-item interaction matrix into latent factors, incorporating temporal dynamics to account for changing user preferences. This approach significantly improves the accuracy of predictions, as shown in Algorithm 1.

Algorithm 1 depicts the procedure followed for collaborative filtering with enhanced SVD.

Input: Vector Matrix

- Initialize the user (P_u) and item (Q_i) latent factor matrices with small random values.
- Initialize biases for users (b_u) and items (b_i) to zero.
- Calculate the global mean (μ) of all ratings.

$$\mu = \frac{1}{N}\sum_{(u,i)} \mathrm{r_{ui}}$$

- Initialize temporal biases for items (γ_i) to zero.
- Begin for
- End for
- Compute predicted rating (r_{ui}).

$$r_{ui} = \mu + b_u + b_i$$

• Compute the error.
• Update biases.
• Update latent factors.
End

Algorithm 1: Collaborative filtering with enhanced SVD

As for the problem of cold start, HybridRecSys equipped the content-based filtering component to solve it. Using information like availability of genre standards such as action, comedy, drama, biography and content attributes such as martial, decades, Johnny, new users or items are suggested even when interaction data is limited. There are other cases that we find out when a new user joined the platform and has no ratings yet, but the user matches by content preferences, for example, genres and subjects, which we identified in user ids 5953 and 1404. HybridRecSys makes it possible to ensure that even if collaborative filtering cannot work because of the cold start, content-based filtering has to offer relevant recommendations since these will depend on item characteristics rather than prior interactions. They also make it possible that new users and items can still get recommended to other users.

HybridRecSys minimizes the extent of sparsity by adding content based filters to the collaborative filters. For IMDB, the individual use of collaborative filtering based on SVD is problematic due to the sparse matrix since it has a high RMSE and MAE (RMSE = 1.2559 and MAE = 0.9717). CB filtering uses the on-domain metadata of items to compute similarity between items even in case of limited user interaction. With the help of such techniques as the advanced similarity measures, content-based filtering offers recommendations based not only on interaction data. This fill in the blank for the zero value in the user-item matrix but also guess on the similarity of two items from the content feature. The system can stand in for areas where the collaborative filtering might fail due to its sparse characteristics.

4.4.2 Advanced content-based filtering

To enhance the quality of content-based recommendations, we preprocess the dataset by eliminating duplicates and standardizing the text in movie overviews. More precisely, the Overview column is divided into individual tokens, and common words are eliminated using the Punkt and stopwords modules from NLTK. The processed text is subsequently transformed into numerical features using TF-IDF (Term Frequency-Inverse Document Frequency) to represent the textual data. Each document is limited to 50 features to ensure computational efficiency. Evaluating the similarity between movie overviews involves using three similarity measures: cosine similarity, Radial Basis Function (RBF) kernel and Weighted Cosine Similarity.

$$\begin{aligned} RBF(x,y) &= exp(-\frac{||x-y||^2}{2\sigma^2}) \\ cosine\ sim(A,B) &= \frac{A.B}{||A||||B||} \\ Wt\ cosine\ sim(A,B) &= \frac{\sum_{i=1}^{n} \omega_i A_i B_i}{\sqrt{\sum_{i=1}^{n} \omega_i A_i^2}\sqrt{\sum_{i=1}^{n} \omega_i B_i^2}} \end{aligned} \tag{3}$$

The function evaluates the similarity scores of all other movies about a target movie. It then selects the top five scores and calculates their average. This average measures how closely related

the recommended movies are to the target movie. This integrated approach capitalizes on the advantages of both similarity measures, providing a more nuanced comprehension of content similarities. The system can comprehend the thematic and stylistic features of films and produce recommendations tailored to the individual users' subtle preferences for content. This thorough content analysis aids in creating a more precise profile of user preferences, which is particularly helpful for items with limited explicit rating data or new users.

The HybridRecSys framework is derived from these two approaches and therefore reduces the sparsity problem. A reduced RMSE and MAE in HybridRecSys indicates that the sparsity impact has been significantly minimized in the proposed model. From 1.2559 of collaborative filtering alone, to 0.6991 of HybridRecSys, integrating content-based filtering reduces the RMSE, showing that the inclusion of the sparse interaction matrix is compensated for HybridRecSys also addresses the sparsity issue through content similar it completes the user-item interaction matrix by recommending items even when the interaction information is limited.

4.4.3 Evaluation metrics

The evaluation metrics used for testing the performance of HybridRecSys are:

4.4.3.1 RMSE

Root Mean Square Error measures the average of the squared differences between predicted and actual ratings.

$$RSME = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{4}$$

where, y_i is the actual or observed value.

$\hat{y}_i$ is the predicted value.

n are the total number of data points or observations

4.4.3.2 MAE

Mean Absolute Error calculates the mean absolute difference between the predicted and actual.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} j - y^i \tag{5}$$

n is the number of data points.

i represents the index of each data point.

j is the predicted value.

y^i is the true value.

5. Experimental results

This section displays the results obtained after experimentation on the dataset, as per the proposed model in Section 3. The distribution of movie ratings is graphically illustrated in Figs. 2 and 3, where a unimodal distribution is obtained on the Count vs Rating plot.

Figs. 4 and 5 presents the number of movies by genre. The highest number of ratings have been allotted to movies of the drama genre. Figs. 6 and 7 presents the data for the yearly movie release

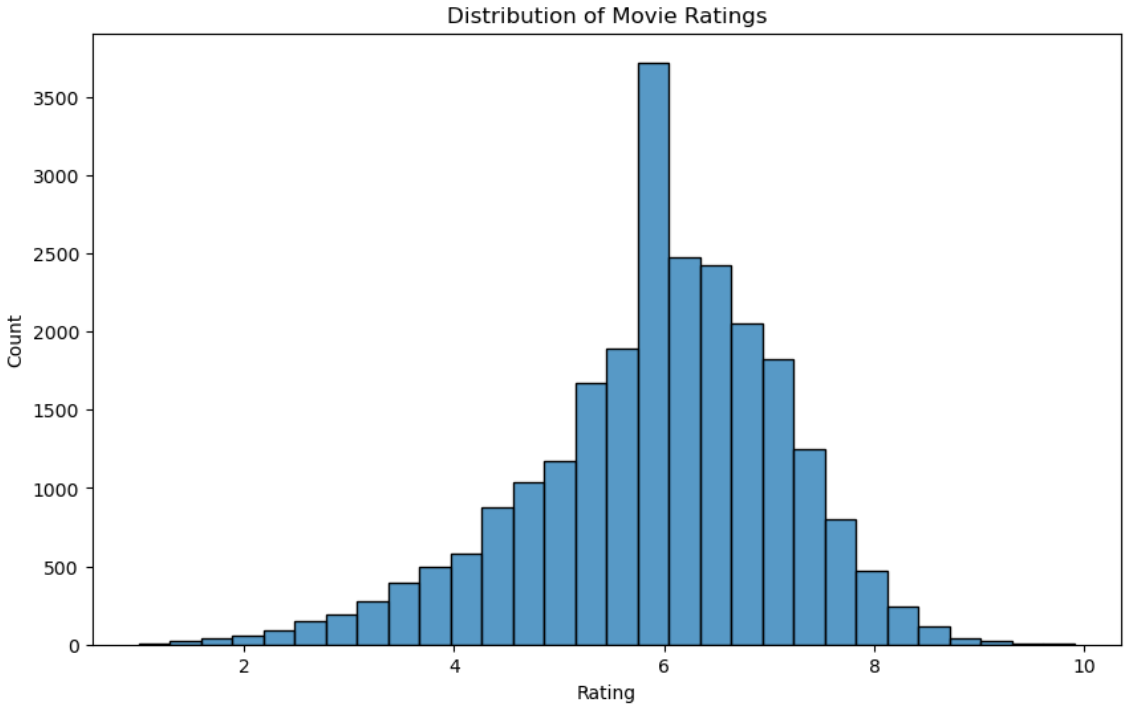

Fig.2 Distribution of Movie Ratings (IMDB Dataset)

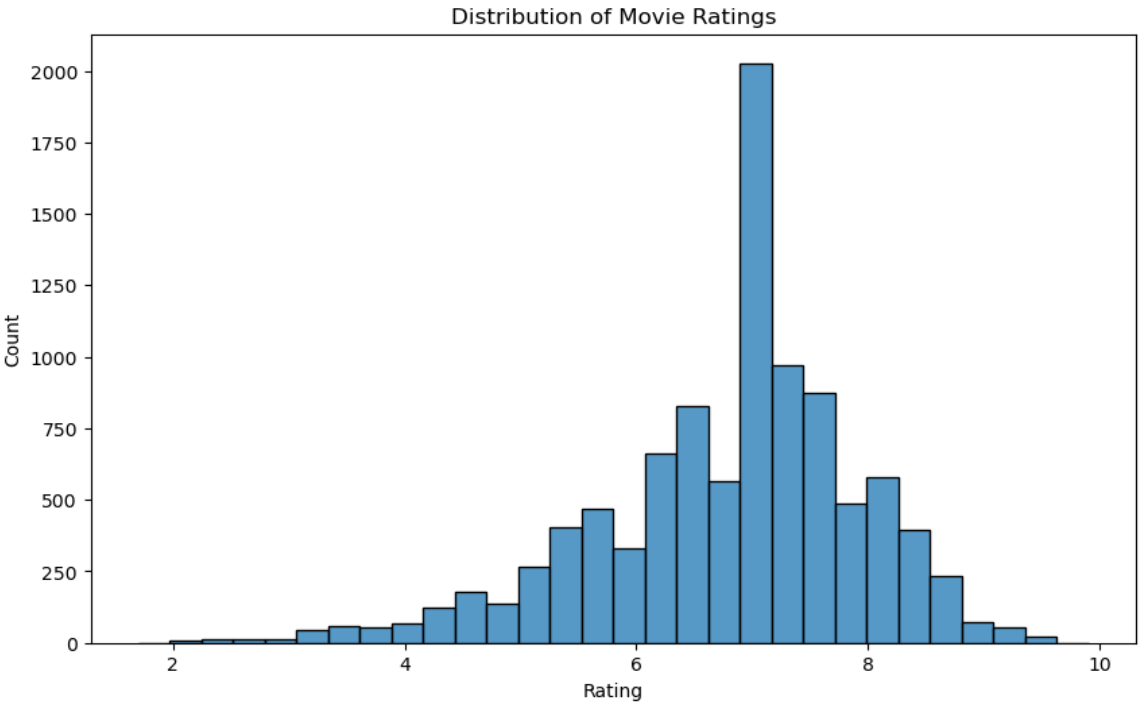

Fig.3 Distribution of Movie Ratings (Netflix popular movie Dataset)

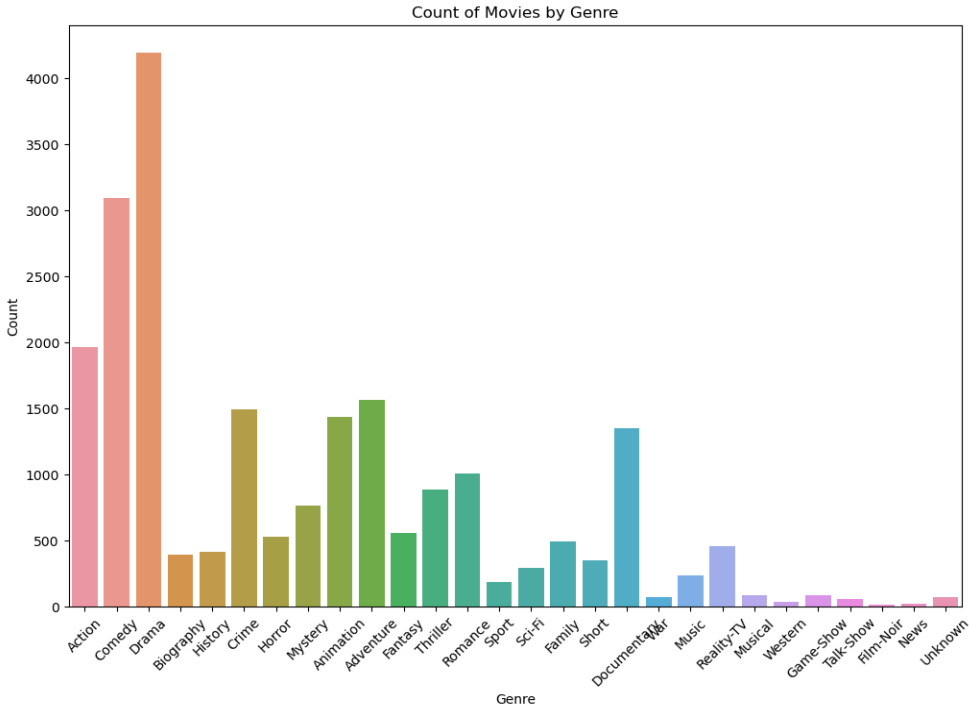

Fig. 4 Distribution of Movies by Genre (IMDB Dataset)

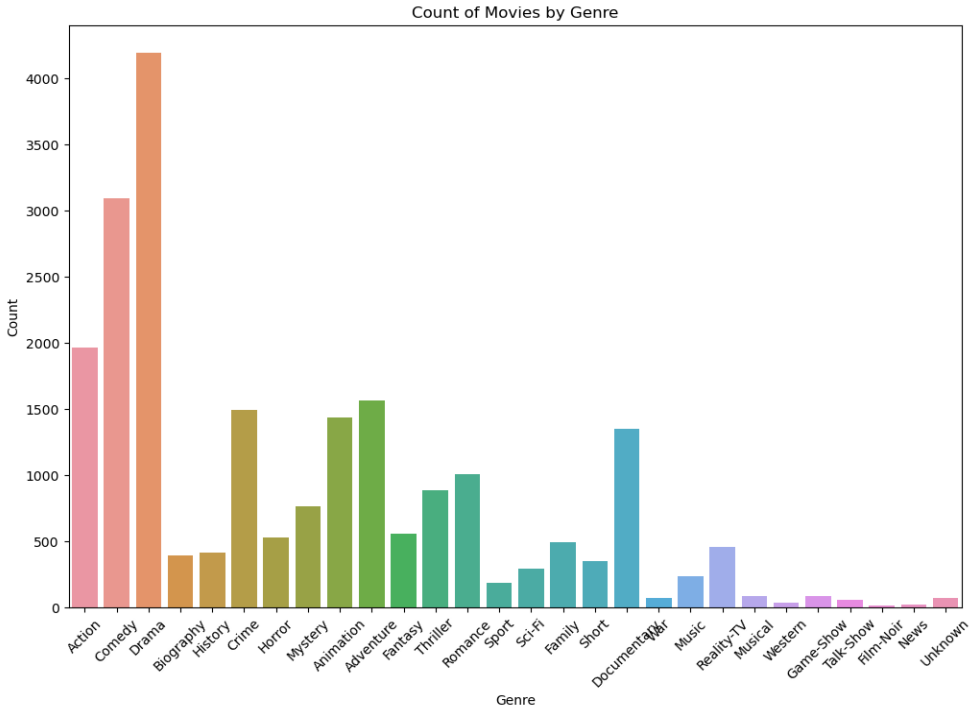

Fig.5 Distribution of Movies by Genre (Netflix popular movie Dataset)

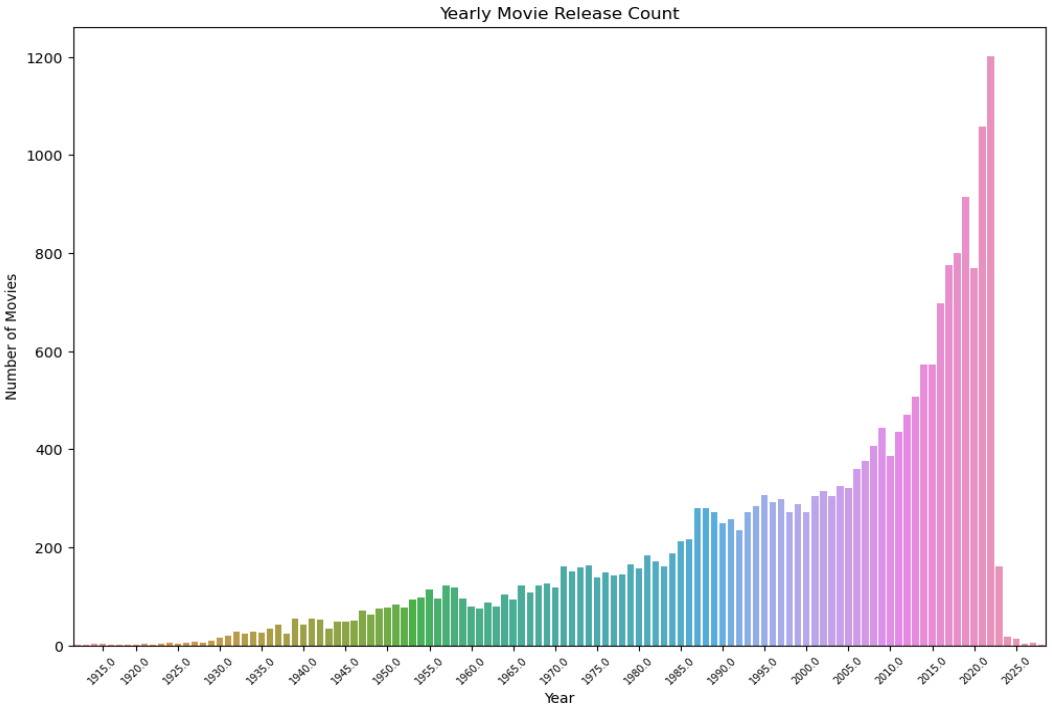

Fig.6 Number of movies released over the years (IMDB Dataset)

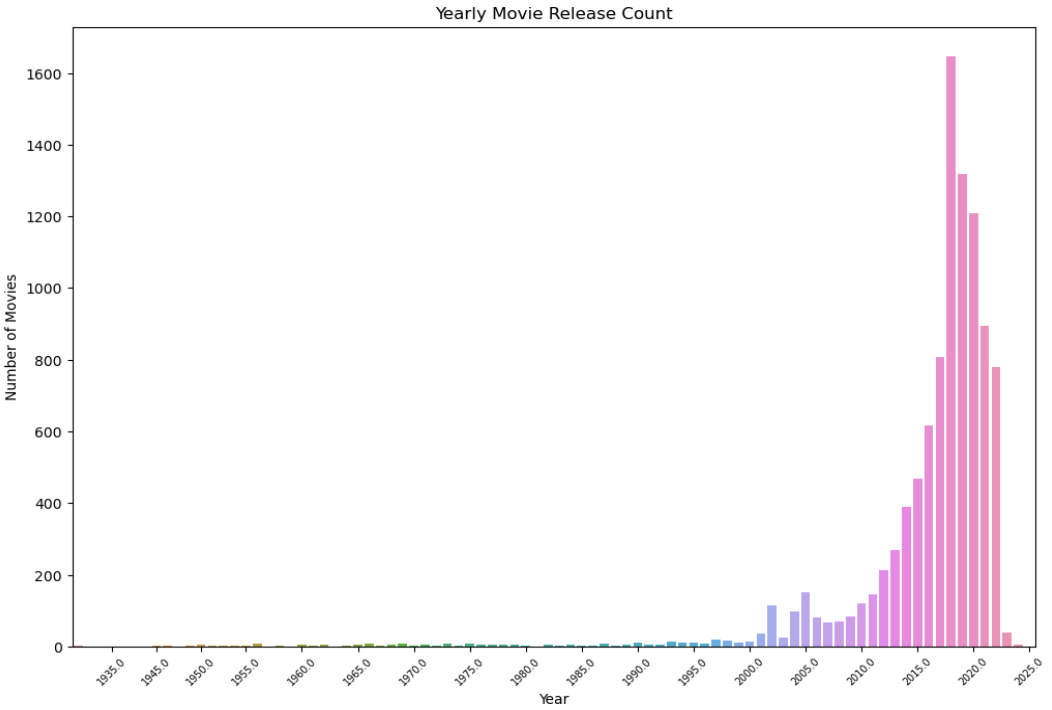

Fig. 7 Number of movies released over the years (Netflix popular movie Dataset)

Table 1 Genre and Content Preferences for Two Selected Users via User Profiling for IMDB Dataset

User ID	Average Rating	Top Genre Preferences		Top Content Preferences	
		Term	Score	Term	Score
10691	8.60	action	0.787422	navy	0.497776
		drama	0.616414	test	0.474553
		game	0.00000	service	0.466209
		adult	0.00000	pilot	0.457547
3852	6.45	Sci-fi	0.578224	reality	0.373922
		adventure	0.451587	future	0.323111
		action	0.356906	hunt	0.194652

Table 2 Genre and Content Preferences for Two Selected Users via User Profiling for Netflix Popular movie Dataset

User ID	Average Rating	Top Genre Preferences		Top Content Preferences	
		Term	Score	Term	Score
5953	8.50	action	0.675677	martial	0.469500
		comedy	0.558855	decades	0.456239
		drama	0.480772	johnny	0.435593
		adult	0.00000	middle	0.411978
1404	8.70	biography	0.679696	half	0.357272
		history	0.669755	second	0.348137
		drama	0.299069	century	0.341211
				queen	0.340288

count graphically. The results for randomly selected two users via user profiling are demonstrated in Table 1. User ID 10691 has an average rating of 8.60, suggesting a significant level of satisfaction with the rated movies. The user's main genre preferences are predominantly action and drama, as indicated by the highest TF-IDF scores for these terms. The user is inclined to movies that revolve around subjects such as the Navy, testing, service, and piloting, as indicated by their prominent content preference terms. This comprehensive profiling assists in customizing recommendations that closely correspond to the user's preferences regarding genre and specific thematic elements. The TF-IDF scores of all the movies a user has seen are averaged to identify their genre preferences, and the TF-IDF scores of movie descriptions are analyzed to discover their content preferences. The hybrid technique and thorough user profiling aid the algorithm in forecasting ratings and preferences more precisely. The collaborative filtering component uses these profiles to infer ratings that are unknown from ratings that are known, and the content-based filtering component uses the implicit data to improve these predictions further. By using explicit and implicit user input, HybridRecSys's dual methodology enables it to extrapolate detailed and accurate recommendations, successfully overcoming sparsity and cold start issues.

The results for randomly selected two users by using user profiling are shown in the following Table 2. It has been found that, the average rate for the user under study, User ID 5953 is 8. 50, thus showing high level of satisfaction with the given movies that he or she has rated. Action and

Table 3 Evaluation metrics for IMDB dataset

Model	Evaluation Metrics	
Collaborative filtering with Enhanced SVD	RMSE Score 1.2559	MAE Score 0.9717
Advanced Content-Based Filtering	Average Similarity Score 0.8497	
HybridRecSys (Collaborative filtering with SVD+Advanced Content-Based Filtering)	RMSE Score 0.6991	MAE Score 0.6987

Table 4 Evaluation metrics for netflix popular movie dataset

Model	Evaluation Metrics	
Collaborative filtering with Enhanced SVD	RMSE Score 1.0738	MAE Score 0.7543
Advanced Content-Based Filtering	Average Similarity Score 0.8163	
HybridRecSys (Collaborative filtering with SVD+Advanced Content-Based Filtering)	RMSE Score 0.2364	MAE Score 0.2357

comedy are the most preferred genres by the user due the high frequency scores that have been assigned to these genres. Also, the user has a great passion for movies particularly those that relate to martial arts, movies that span several decades and characters like 'Johnny' based on the user's top content preference terms. The extensive user characterization proves useful in prescribing suggestions that are in harmony with the particular genre and themes preferred by the user.

Likewise, User ID 1404 has an average rating of 8, which is even higher than the one mentioned above. 70, which can be considered as a very high level of satisfaction. If we look at the genre the user is most interested in, biography has got the highest TF-IDF value, followed by history and drama. Their content wants show an affinity towards topics that concern historical eras as well as historical personalities which are "half," "second," "century," and "queen."

The hybrid model with both collaborative and content-based filter applies this comprehensive characterization to make more precise estimations of rating and desire. Therefore, by incorporating both the explicit and implicit feedback from the users, HybridRecSys can produce more accurate recommendation solution hence mitigating the problems of sparsity and cold start.

Table 3 presents the evaluation metric scores obtained by the two individual models and HybridRecSys. It also provides a detailed comparison with related systems, demonstrating HybridRecSys's superiority in recommendation accuracy. The collaborative filtering method using Singular Value Decomposition (SVD) achieved accuracy scores of 1.2559 and 0.9717, respectively, indicating a reasonable level of accuracy. The advanced content-based filtering method achieves an average similarity score of 0.8497, indicating its efficacy in capturing content similarities. These metrics indicate the model's capacity to deliver precise and pertinent recommendations. HybridRecSys performs better than the individual models, obtaining an RMSE Score of 0.6072 and an MAE Score of 0.6060.

Table 4 presents the evaluation metric scores obtained by the two individual models and HybridRecSys for the Netflix Popular Movie Dataset. The collaborative filtering method using

Table 5 Comparative Analysis

Paper	Model	Dataset	Evaluation Metric	Score
(Johari *et al*. 2021)	Demographic+Content Based Filtering	IMDb	RMSE	0.7812
(Sanwal *et al*. 2021)	DT+SVR+RF	IMDb	RMSE	0.7999
Our Paper	HybridRecSys	IMDb	RMSE	0.6072
		Netflix Popular Movie Dataset	RMSE	0.2364

Enhanced Singular Value Decomposition (SVD) achieved RMSE and MAE scores of 1.0738 and 0.7543, respectively, indicating a reasonable level of accuracy in the recommendations provided.

7. Future research directions

Subsequent studies could investigate the incorporation of supplementary contextual data, such as user demographics, social media interactions, and real-time behavioral data, to further improve the precision and customization of HybridRecSys. By integrating advanced deep learning techniques, such as neural collaborative filtering and attention mechanisms, the system's capacity to capture intricate user-item interactions could be enhanced. Explosion can be done on the model's scalability for larger datasets and its applicability to diverse domains like music, books, and e-commerce. Ultimately, incorporating adaptive learning algorithms that continuously update user profiles in real-time as new data is gathered can enhance the system's responsiveness and efficacy in dynamic settings.

8. Discussion

The development of HybridRecSys showcases the effectiveness of combining collaborative and content-based filtering to leverage their complementary strengths. Incorporating temporal dynamics in the collaborative filtering approach helps address the issue of evolving user preferences, providing more accurate predictions over time. Using natural language processing in content-based filtering allows for a deeper understanding of movie themes, enhancing the relevance of recommendations.

8.1 The cold start issue

HybridRecSys solves the cold start problem using sophisticated content-based filtering and extensive user profiling. The system can provide initial recommendations for new users by analyzing movie attributes to infer their genre and content preferences. User profiling and HybridRecSys together address the cold start problem, offering solutions by leveraging user preferences even when no interaction data exists. This enables the system to offer pertinent recommendations even when comprehensive interaction data is needed. Content-based filtering recommends new items to users whose profiles align with the item's attributes, thereby addressing the cold start problem.

8.2 The sparsity problem

HybridRecSys utilizes a hybrid methodology that integrates collaborative filtering and content-based filtering. The collaborative filtering component utilizes an improved SVD technique to deduce user preferences from existing rating data, even in cases where the data is limited. Collaborative filtering suffers from the sparsity problem; however, SVD and HybridRecSys provide effective solutions to overcome this issue. By incorporating user profiles that include explicit and implicit feedback, the system can predict unknown ratings based on known ones, thus filling in gaps where direct interaction data may be missing. Content-based filtering enhances this process by employing NLP to extract user preferences from the characteristics of the items. This ensures that recommendations remain precise and pertinent despite scarce interaction data.

9. Conclusions

HybridRecSys enhances movie recommendations by combining collaborative filtering with temporal dynamics in SVD and advanced content-based filtering using natural language processing. The system, tested on both IMDb and Netflix Popular Movie Datasets, shows strong performance across different datasets. The addition of weighted cosine similarity further improves recommendation accuracy. For the IMDb dataset, the system achieved an RMSE of 0.6991 and an MAE of 0.6987. For the Netflix dataset, the RMSE was 0.2364 and the MAE was 0.2357. This approach effectively addresses cold start and sparsity, ensuring personalized recommendations for all users.

References

Bahrani, P., Minaei-Bidgoli, B., Parvin, H., Mirzarezaee, M. and Keshavarz, A. (2024), "A hybrid semantic recommender system enriched with an imputation method", *Multimed. Tools Appl.*, **83**(6), 15985-16018. https://doi.org/10.1007/s11042-023-15258-4

Bobadilla, J., Hernando, A., Ortega, F. and Bernal, J. (2011), "A framework for collaborative filtering recommender systems", *Expert Syst. Appl.*, **38**(12), 14609-14623. https://doi.org/10.1016/j.eswa.2011.05.021

Burke, R. (2002), "Hybrid recommender systems: Survey and experiments", *User Modell. User Adapt. Interact.*, **12**(4), 331-370. https://doi.org/10.1023/A:1021240730564

Cacheda, F., Carneiro, V., Fernández, D. and Formoso, V. (2011), "Comparison of collaborative filtering algorithms", *ACM T Web*, **5**(1), 1-33. https://doi.org/10.1145/1921591.1921593

Çano, E. and Morisio, M. (2017), "Hybrid recommender systems: A systematic literature review", *Intell. Data Anal.*, **21**(6), 1487-1524. https://doi.org/10.3233/IDA-163209

da Silva, F.L., Slodkowski, B.K., da Silva, K.K.A. and Cazella, S.C. (2023), "A systematic literature review on educational recommender systems for teaching and learning: Research trends, limitations and opportunities", *Educ. Inform. Technol.*, **28**(3), 3289-3328. https://doi.org/10.1007/s10639-022-11341-9

Da'u, A. and Salim, N. (2019), "Recommendation system based on deep learning methods: A systematic review and new directions", *Artif. Intell. Rev.*, **53**(4), 2709-2748. https://doi.org/10.1007/S10462-019-09744-1

Ekstrand, M.D. (2011), "Collaborative filtering recommender systems", *Found. Trends Human Comput. Interact.*, **4**(2), 81-173. https://doi.org/10.1561/1100000009

Felfernig, A., Tintarev, N., Trang Tran, T.N. and Stettinger, M. (2024), "Explanations for groups", *Signal*

Commun. Technol., **Part F1811**, 109-131. https://doi.org/10.1007/978-3-031-44943-7_6

Gao, C., Zheng, Y., Wang, W., Feng, F., He, X. and Li, Y. (2024), "Causal inference in recommender systems: A survey and future directions", *ACM T. Inform. Syst.*, **42**(4), 1-32. https://doi.org/10.1145/3639048

Ghazanfar, M.A. and Prugel-Bennett, A. (2010), "A scalable, accurate hybrid recommender system", *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, *WKDD 2010*, **94-98**. https://doi.org/10.1109/WKDD.2010.117

Gündoğan, E. and Kaya, M. (2022), "A novel hybrid paper recommendation system using deep learning", *Scientometrics*, **127(7)**, 3837-3855. https://doi.org/10.1007/s11192-022-04420-8

Hannech, A., Adda, M. and McHeick, H. (2017), "Recommendation model based on a contextual similarity measure", *Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications*, *ICMLA 2016*, **394-401**. https://doi.org/10.1109/ICMLA.2016.9

Jannach, D., de Souza P. Moreira, G. and Oldridge, E. (2020), "Why are deep learning models not consistently winning recommender systems competitions yet?", *Proceedings of the Recommender Systems Challenge 2020*, **44-49**. https://doi.org/10.1145/3415959.3416001

Johari, M. and Laksito, A. (2021), "The hybrid recommender system of the Indonesian online market products using IMDb weight rating and TF-IDF", *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, **5**(5), 977-983. https://doi.org/10.29207/resti.v5i5.3486

Koren, Y., Bell, R. and Volinsky, C. (2009), "Matrix factorization techniques for recommender systems", *Computer*, **42**(8), 30-37. https://doi.org/10.1109/MC.2009.263

Li, Q. and Kim, B.M. (2003), "Clustering approach for hybrid recommender system", *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, **33-38**. https://doi.org/10.1109/WI.2003.1241167

Li, Y., Liu, K., Satapathy, R., Wang, S. and Cambria, E. (2024), "Recent developments in recommender systems: A survey", *IEEE Comput. Intell. Mag.*, **19**(2), 78-95. https://doi.org/10.1109/MCI.2024.3363984

Martínez, L., Barranco, M.J., Pérez, L.G. and Espinilla, M. (2008), "A knowledge-based recommender system with multigranular linguistic information", *Int. J. Comput. Intell. Syst.*, **1**(3), 225-236. https://doi.org/10.1080/18756891.2008.9727620

Rosewelt, L.A. and Renjit, J.A. (2020), "A content recommendation system for effective e-learning using embedded feature selection and fuzzy DT-based CNN", *Journal of Intelligent & Fuzzy Systems*, **39**(1), 795-808. https://doi.org/10.3233/JIFS-191721

Sanwal, M. and Çalışkan, C. (2021), "A hybrid movie recommender system and rating prediction model", *International Journal of Information Technology and Applied Sciences (IJITAS)*, **3**(3), 161-168. https://doi.org/10.52502/ijitas.v3i3.128

Schafer, J.B., Frankowski, D., Herlocker, J. and Sen, S. (2007), "Collaborative filtering recommender systems", *The Adaptive Web*, **291-324**. https://doi.org/10.1007/978-3-540

Son, J. and Kim, S.B. (2017), "Content-based filtering for recommendation systems using multiattribute networks", *Expert Syst. Appl.*, **89**, 404-412. https://doi.org/10.1016/j.eswa.2017.08.008

Tarus, J.K., Niu, Z. and Mustafa, G. (2018), "Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning", *Artif. Intell. Rev.*, **50**(1), 21-48. https://doi.org/10.1007/s10462-017-9539-5

Wang, Y., Chan, S. C. F. and Ngai, G. (2012), "Applicability of demographic recommender system to tourist attractions: A case study on TripAdvisor", *Proceedings of the 2012 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops*, *WI-IAT 2012*, **97-101**. https://doi.org/10.1109/WI-IAT.2012.133

Xue, H.J., Dai, X.Y., Zhang, J., Huang, S. and Chen, J. (2017), "Deep matrix factorization models for recommender systems", *Proceedings of the IJCAI International Joint Conference on Artificial Intelligence*, 3203-3209. https://doi.org/10.24963/ijcai.2017/447

Yu, H.F., Hsieh, C.J., Si, S. and Dhillon, I.S. (2014), "Parallel matrix factorization for recommender systems", *Knowledge and Information Systems*, **41**(3), 793-819. https://doi.org/10.1007/s10115-013-0682-2

Zhang, G., Liu, Y. and Jin, X. (2020), "A survey of autoencoder-based recommender systems", *Front*. *Comput*. *Sci*., **14**(2), 430-450. https://doi.org/10.1007/s11704-018-8052-6

Zhang, H.R., Min, F., He, X. and Xu, Y.Y. (2015), "A hybrid recommender system based on user-recommender interaction", *Math*. *Probl*. *Eng*., 1-11. https://doi.org/10.1155/2015/145636

Zhao, W., Tian, H., Wu, Y., Cui, Z. and Feng, T. (2022), "A new item-based collaborative filtering algorithm to improve the accuracy of prediction in sparse data", *Int*. *J*. *Comput*. *Intell*. *Syst*., **15**(1), 1-15. https://doi.org/10.1007/S44196-022-00068-7

CC