

Hybrid Filtering Techniques and Improved SOM for Next-Generation Film Recommendations

Saurabh Sharma¹, Dr. Ghanshyam Prasad Dubey², Dr. Harish Kumar Shakya³

Submitted: 29/01/2024 Revised: 07/03/2024 Accepted: 15/03/2024

Abstract: Personalized movie recommendations are becoming increasingly important for increasing user satisfaction and engagement in an era where entertainment content is becoming more digitally connected. This work presents a novel approach to facilitate movie selection through the use of Enhanced Self-Organizing Maps (SOM). SOMs, or unsupervised neural network architectures, are helpful in recommendation systems because of their ability to recognize intricate patterns in data. The methodology described in this paper starts with gathering user-movie interaction data, such as user ratings and movie characteristics. This data is normalized before being used to train the model in order to preserve homogeneity. Subtle patterns in data can be effectively identified by the Enhanced SOM due to its adjustable learning rate and neighborhood function. The Enhanced SOM's capacity to identify related users and movies is used to generate personalized movie suggestions. The framework's incorporation of hybrid filtering approaches improves the caliber of recommendations. While content-based filtering uses movie attributes like genres and descriptions, collaborative filtering algorithms take advantage of user-item interactions. These methods yield recommendations that synergistically combine multiple filtering processes. The efficacy of the proposed resolution is meticulously evaluated by contrasting the precision of the recommendations and user contentment with pre-established standards. A thorough evaluation of real-world datasets supports the efficacy of the Enhanced SOM-based movie recommendation technique. To further enhance suggestion quality, the system offers options to modify grid sizes, neighborhood functions, and parameters. When taken as a whole, these features demonstrate how successful the suggested approach is in making personalized movie recommendations. Enhanced SOMs are a reliable model for content platforms that want to enhance user experiences by offering precise movie recommendations along with scalability and flexibility when paired with hybrid filtering techniques.

Keywords: Movie Recommendation, Enhanced SOM, Personalization, Hybrid Filtering, User Engagement

1. Introduction

With the advent of digital entertainment, consumers now have access to vast movie and TV show libraries, which has significantly expanded their options. Because of its expansion, selecting the next viewing experience is both an exciting and challenging task. Recommendation engines play a crucial role in solving this issue since they point users toward content that aligns with their interests. In particular, the use of movie recommendation algorithms has significantly increased as a result of movies' increasing popularity. This study explores the topic of movie recommendation and proposes a novel strategy that utilizes Enhanced Self-Organizing Maps (SOM) to increase the customisation and accuracy of movie recommendations.

Effective recommendation algorithms are more important as digital libraries, streaming services, and video-on-demand platforms increase and change the way people consume entertainment. Consumers want recommendations for movies that offer a range of options in addition to being tailored to their own tastes. Several factors, such as user

demographics and cultural backgrounds, the constantly shifting content environment, and the subjective nature of movie preferences, make movie suggestion a challenging endeavor. While typical recommendation techniques such as content-based and collaborative filtering have made some headway in addressing these problems, they often fall short in capturing the intricate relationships seen in movie collections. Our proposed approach overcomes these limitations by leveraging Enhanced Self-Organizing Maps (SOM), which are widely recognized for their effectiveness in detecting complex patterns in data. SOMs have applications in many domains, such as feature mapping and data clustering, and show promise in recommendation systems because of their versatility and potential for unsupervised learning. An important part of our approach is using Enhanced SOMs to provide a topological representation of user preferences and movie attributes. The SOM is able to recognize recurrent themes and patterns in the videos by arranging them in a high-dimensional context. These patterns go beyond conventional genre- or content-based correlations to incorporate subtle variations in user preferences. Enhanced SOMs enhance their dynamic representation of complex inputs during training by utilizing neighborhood functions and adjustable learning rates. A few critical steps in our method are made clearer by our research. The initial step in the data collection procedure is gathering user ratings and movie characteristics. Next, standardization is carried out to ensure that rating scales are consistent with one another. By integrating hybrid filtering approaches with Enhanced SOMs, movie recommendations become more

¹Research Scholar, Amity School of Engineering and Technology (ASET), Amity University, Gwalior, India
ORCID ID: 0000-0002-0090-6043
Email: saurabhgyangit@gmail.com

²Associate Professor, Amity School of Engineering and Technology (ASET), Amity University, Gwalior, India
ORCID ID: 0000-0003-0868-4219

³Associate Professor, Department of Artificial Intelligence & Machine Learning, Manipal University Jaipur
ORCID ID: 0000-0002-5401-3507

diversified and accurate, hence improving the viewer's enjoyment experience. Our approach serves as a guide in the ever-evolving world of digital entertainment, assisting users in discovering films that align with their interests and preferences.

2. Literature Review

The field of recommender systems has experienced substantial development in reaction to the constantly growing array of digital content that consumers can access. A subset of this topic called movie recommendation seeks to provide consumers with interesting and personalized movie recommendations based on their past interactions and preferences. The present literature review offers an extensive amalgamation of previous studies conducted in the domain of recommender systems, with a specific emphasis on movie suggestion. It also functions as an advance towards the presentation of a novel methodology that employs Enhanced Self-Organizing Maps (SOM). Alyari and Navimipour (2018) offer a thorough analysis of the most recent advances in recommender system technology. Their study emphasizes both content-based and collaborative techniques, providing insightful information on the various methodologies used. It emphasizes the need for further investigation and progress in the area, laying the foundation for innovative approaches like the one this study suggests.

In their exploration of the idea of recommender system explanations, Caro-Martinez et al. (2018) [2] put up a theoretical framework for contextualizing user suggestions. Their work emphasizes the overall goal of improving user pleasure and trust, which is essential to any recommendation system, including those that suggest movies, even if their concentration is on suggestion interpretability. In-depth analysis of the literature on recommendation systems is provided by Gupta (2020) [3], who also offers insights into the many approaches, methods, and difficulties related to recommendation technology. The study emphasizes the various uses for recommendation systems and how important it is to improve user experiences, which is a crucial component of movie recommendation. In their exploration of size suggestions in the context of fashion e-commerce, Abdulla and Borar (2017) [4] highlight the significance of customizing recommendations to certain user features and preferences. The idea of tailored suggestions based on distinctive qualities is ubiquitous, even in the context of movie selection, even though their approaches are different. The table that is provided provides a brief synopsis of the scholarly articles that were referenced in the literature study. It lists the publications' years of publication, primary subjects, and methodology. This tabular structure makes it easier to understand the contextual importance of each study with respect to the future research project that will use Enhanced Self-Organizing Maps for movie recommendation.

Paper	Year	Focus	Approach
Alyari, F. et al.[1]	2018	Recommender systems	Systematic review and suggestions
Caro-Martinez, M. et al.[2]	2018	Explanations in recommender systems	Theoretical model of explanations
Gupta, S. [3]	2020	Literature review on recommendation systems	Review of various approaches
Abdulla, G.M. et al. [4]	2017	Size recommendation system	Focus on size-based recommendations
Aggarwal, C.C.[5]	2016	Introduction to recommender systems	Basics of recommender systems
Ghazanfar, M.A. et al. [6]	2010	Scalable, accurate hybrid recommender system	Proposal of a hybrid system
Deldjoo, Y. et al.[7]	2016	Content-based video recommendation	Focus on stylistic visual features
Alamdari, P.M. et al.[8]	2020	Recommender systems in e-commerce	A systematic study on e-commerce systems
Cami, B.R. et al.[9]	2017	Content-based movie recommender system	Temporal user preferences in movies
Beniwal, R. et al.[10]	2021	Hybrid recommender system	Using artificial bee colony optimization
Çano, E. et al.[11]	2017	Hybrid recommender systems	A systematic literature review
Schafer, J.B. et al.[12]	2001	E-commerce recommendation applications	Focus on e-commerce recommendations
Shen, J. et al.[13]	2020	Collaborative filtering-based recommendation	Recommendation system for big data
Dakhel, G.M. et al.[14]	2011	Collaborative filtering algorithm	Using K-means clustering and voting
Katarya, R. et al.[15]	2017	Effective collaborative movie recommender	Using cuckoo search
Kumar, B. et al.[16]	2016	Approaches, issues, and challenges	A systematic review of recommender systems

3. The Proposed Methodology for Building a Movie Recommendation System

Initialization:

a. Start by setting up a matrix R to capture user-movie interactions, where each entry R_{ij} represents the rating of user i for movie j .

b. Then, initialize the Self-Organizing Map (SOM) grid along with its weights.

Training SOM:

c. Normalize the ratings matrix R to ensure ratings fall within the range of 0 to 1.

d. Begin by setting up the SOM weights with small random values.

e. Specify key parameters:

i. Learning rate (η): This governs the extent of weight adjustments during training. Typically, it starts high and diminishes over time.

ii. Neighborhood function (h): This function determines how neighboring units influence weight updates.

For each user u :

f. Compute the preference vector P_u based on user ratings. This vector encapsulates the user's preferences across all movies.

During each training epoch:

g. Iterate through the training epochs, representing passes over the dataset.

h. For each user's preference vector P_u :

i. Identify the Best Matching Unit (BMU) within the SOM that closely aligns with the user's preferences.

ii. Update the weights of the BMU and its neighboring units based on the user's preferences. This update operation incorporates the learning rate and the neighborhood function.

Recommendation Generation:

i. For every user u :

i. Recalculate the preference vector P_u based on the user's ratings.

ii. Determine the BMU within the SOM corresponding to the user's updated preferences.

iii. Retrieve movies associated with the BMU, which serve as personalized recommendations for the user.

Mathematical Formulas:

j. Express the mathematical formulas employed in the algorithm:

i. Learning rate (η): Usually adjusted over time to modulate the learning pace.

ii. Neighborhood function (h): Specifies the extent of influence exerted by neighboring units during weight updates.

iii. BMU calculation: Identifies the unit in the SOM that best matches the user's preferences by assessing the Euclidean distance between the user's preference vector and the SOM's weight vectors.

Algorithm: Movie Recommendation Using Enhanced SOM

Initialization:

1. Initialize user-movie interaction matrix, R , where R_{ij} represents the rating of user i for movie j .

2. Initialize SOM grid and weights.

Training SOM:

3. Normalize the ratings matrix R to have values between 0 and 1.

4. Initialize SOM weights with small random values.

5. Define parameters:

• Learning rate (η): Typically starts high and decreases over time.

• Neighborhood function (h): Defines the influence of neighboring units during SOM weight updates.

For each user u :

6. Calculate the preference vector P_u based on user ratings:

$P_u = [R_{u1}, R_{u2}, \dots, R_{un}]$, where n is the number of movies.

For each training epoch:

7. Find the Best Matching Unit (BMU) in the SOM for P_u .

8. Update the weights of BMU and its neighboring units using the following formula:

$$\Delta W_{ij} = \eta(t) \cdot h(t) \cdot (P_u - W_{ij})$$

where W_{ij} is the weight vector for unit i, j .

Recommendation Generation:

9. For each user u :

• Calculate the preference vector P_u based on user ratings.

• Find the BMU in the SOM for P_u .

• Retrieve movies associated with the BMU.

Mathematical Formulas:

• Learning rate (η): Typically decreases over time (e.g., $\eta(t) = \eta_0 / (1 + t/\tau)$).

• Neighborhood function (h): Defines the influence of neighboring units

(e.g., $h(t) = \exp(-d^2 / (2\sigma^2))$).

• BMU calculation: $\text{argmin}(\|P_u - W_{ij}\|)$ for all units i, j in the SOM).

4. Methodology and Result

4.1 Dataset and Features

An analysis of the recommended approach, its performance measures, and results is presented in this section. The hybrid system that is being presented here was created especially

for movie recommendations using Tensorflow in Python. The Movielens databases provided a wealth of information that was used in the inquiry. The collection comprises ratings for 62,000 films and 1 million tags from 162,000 individuals, for a total of 25 million entries. It also includes tag genome data with 15 million relevance ratings and 1,129 tags. This dataset, dubbed "ml-25m," combines free-text labeling tasks and 5-star ratings taken from the MovieLens recommendation engine. From January 9, 1995, to November 21, 2019, 162,541 users contributed 1,093,360 tag applications, 62,423 movies, and 25,000,095 ratings to the collection. On November 21, 2019, a random selection of individuals who had each rated at least 20 different movies was used to build the dataset. Nevertheless, demographic information is missing, and users are only identified by their ID. The MovieLens databases provide extensive information about users, movies, ratings, and tags. The proposed model was trained on 75% of the training data and tested on the remaining 25% after the dataset was divided into training and testing sets. The hybrid system-based machine learning approach uses the dataset's information in both stages to recommend movies. User and movie ratings are graphically represented in Figures 1 and 2, which are important for recommendation generation. The results pertaining to user and movie ratings are presented in Figures 1 and 2 below, along with suggestions.

user_id	548	626	847	997	1401	1652	1748	1920	1977	2165	...	160922	160951	161184	161383	161544	161586	1
new_title																		
10 things i hate about you	4.5	4.0	3.5	3.0	1.0	0.0	1.5	0.0	0.0	4.0	...	4.0	3.0	2.5	0.0	0.0	0.0	0.0
101 dalmatians one hundred and one dalmatians	4.0	4.0	3.5	3.0	0.0	5.0	3.0	3.0	4.0	0.0	...	4.0	0.0	4.0	3.5	0.0	2.5	0.0
12 angry men	0.0	4.0	4.0	4.0	5.0	4.5	0.0	0.0	4.0	5.0	...	4.0	0.0	5.0	0.0	0.0	3.5	0.0
2001 a space odyssey	0.0	4.0	4.5	5.0	3.0	3.0	4.0	4.0	4.5	0.5	...	3.0	4.5	5.0	3.0	4.0	5.0	0.0
21 grams	4.0	0.0	2.5	4.0	0.0	0.0	3.5	4.0	3.0	5.0	...	0.0	4.0	4.5	0.0	3.0	0.0	0.0
...
young frankenstein	0.0	4.0	3.0	0.0	2.0	0.0	1.5	4.0	0.0	0.0	...	4.5	4.5	4.5	3.0	0.0	4.5	0.0
you've got mail	3.5	4.0	3.5	3.0	0.0	4.0	4.0	3.0	2.0	0.0	...	3.5	3.0	3.0	0.0	1.5	0.0	0.0
zodiac	4.5	0.0	4.0	4.0	4.0	5.0	3.5	3.0	0.0	1.5	...	3.0	4.0	3.0	0.0	2.5	2.0	0.0

Fig. 1. The output of the movie ratings

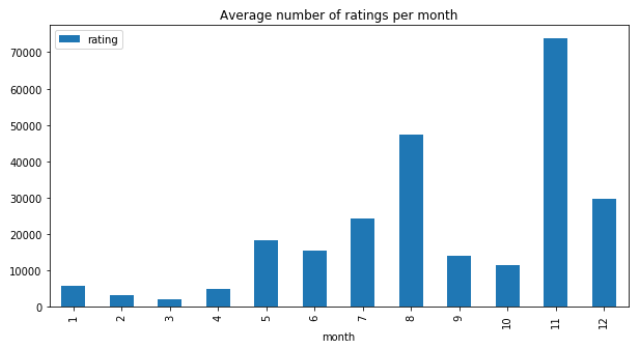


Fig. 2. Output of user ratings to provide recommendations

November, the eleventh month of the year, has the largest average number of ratings (about 70,000), followed by August (around 50,000) and December (almost 30,000), as can be seen in Figure 2. However, it's crucial to remember that these numbers are estimates because there isn't much data for 2003—just two months' worth—available. Let's

now examine the thorough evaluation of the overall number of ratings received every month for the entire year.

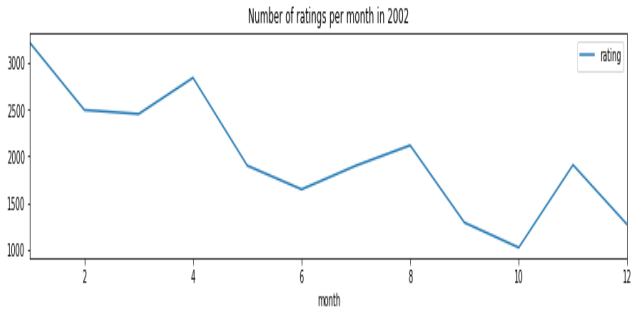


Fig. 3. Monthly total number of ratings each year.

There is a discernible increase in ratings, as seen in Figure 3, especially in November 2000. As the graph illustrates, this surge—which accounts for nearly 90% of the total ratings—has a major impact on the high average ratings that are seen in November, August, and December. That being said, a negative tendency becomes evident in 2001 and 2002. With only two months' worth of data available for 2003, it is difficult to draw firm conclusions from this pattern.

From here on out, we'll be looking at how different rating levels are distributed in order to learn more about how users generally rate things.

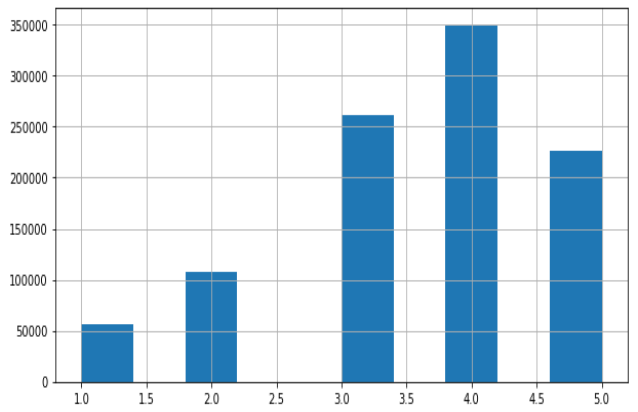


Fig. 4. Most Frequent Rating

Approximately 350,000 instances of the rating value 4 have been seen, making it the most often observed rating, according to the statistics shown in Figure 4. In our one million rating dataset, around 35% of the ratings were 4 stars, while approximately 26% and 21% of the ratings were 3 and 5 stars, respectively. It is significant to note that these estimations may include some mistakes due to the limited data available for 2003. However, it is evident that a considerable proportion of users typically provide ratings of four or above.

Every year in the dataset can be subjected to comparable analysis.

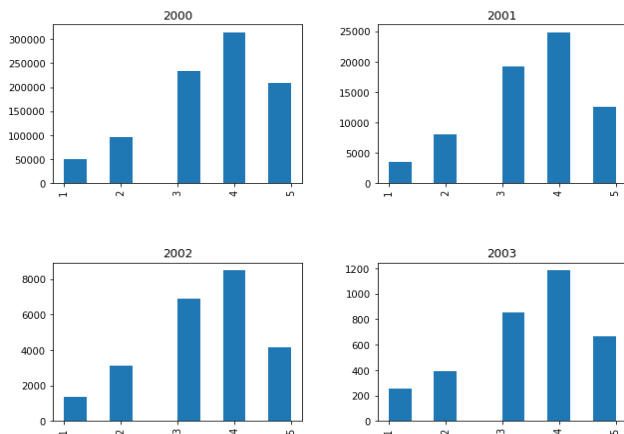


Fig. 5. Monthly Distribution of Rating Values

Comparable distributions are shown for each year period in Figure 5. The analysis of the monthly distribution of rating values may then be undertaken.

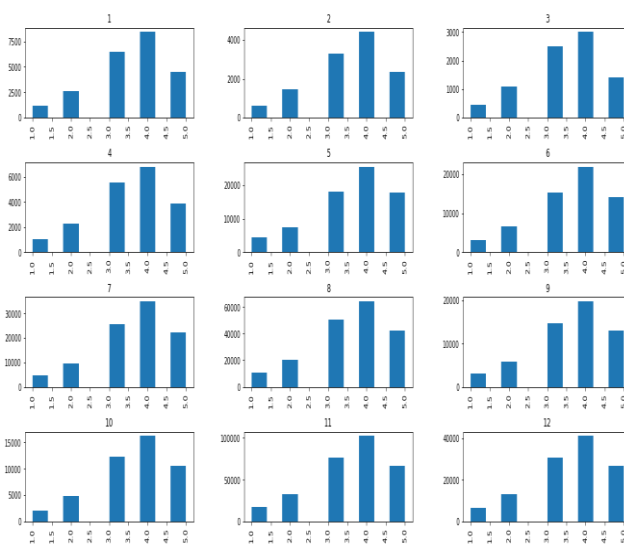


Fig. 6. Visualization of Lower and Upper Boundaries with Standard Deviation

A distribution resembling that seen in the annual and comprehensive graphs may be seen in Figure 6. All categories combined have an average rating of about 3.6. After that, we'll examine the ratings' temporal fluctuations and use standard deviation-based techniques to depict the lower and upper bounds.

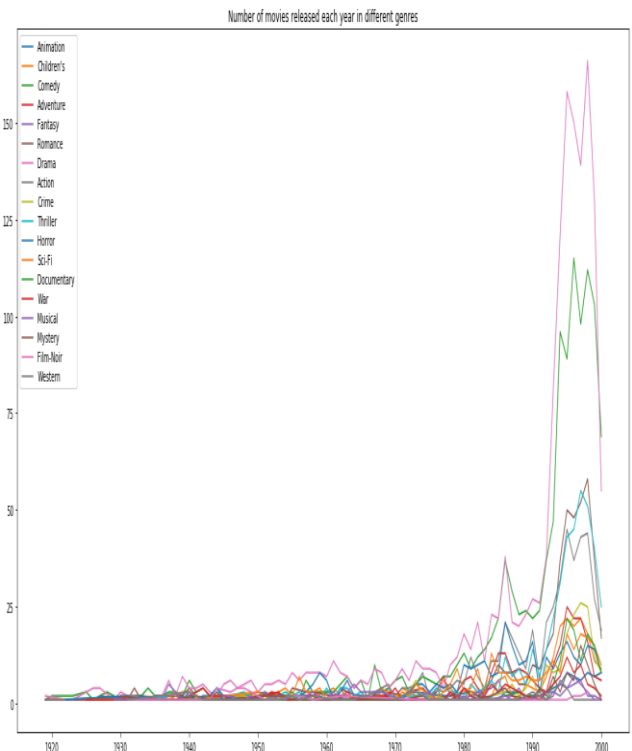
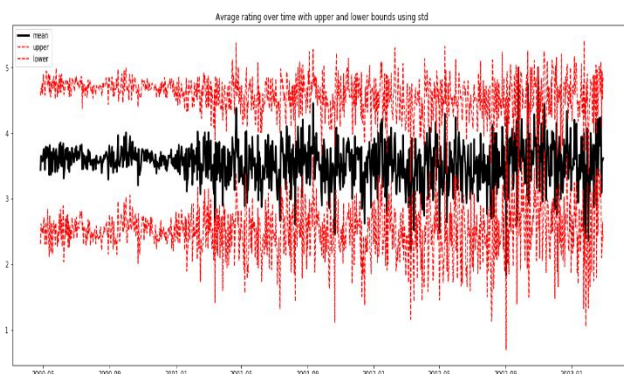


Fig. 7. Interest in the film noir and comedy genres in the late 1990s.

In Fig. 7. Shows that in the late 1990s, there was a lot of interest in the comedy and film noir

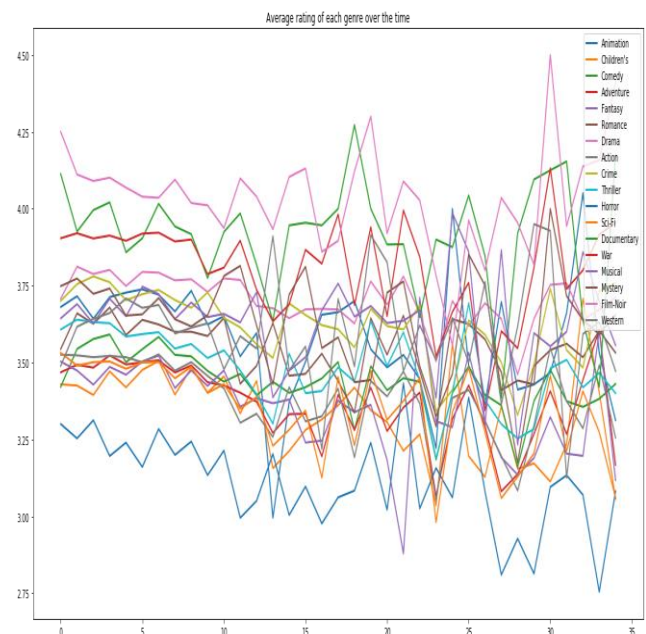


Fig 8. Density plot for ratings by genre.

Figure 8 illustrates that the Film-Noir and Horror categories consistently demonstrate high and low average ratings, respectively, with occasional extreme values. Following this observation, a density plot depicting ratings by genre will be generated.

It is clear that throughout time, the average rating has continuously fluctuated between 3 and 4.

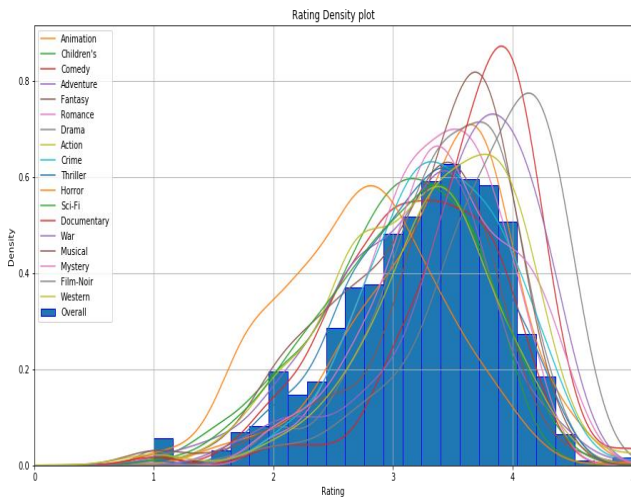


Fig. 9. Left-skewed distribution plot for ratings by genre.

In Figure 9, it is evident that all genres exhibit a left-skewed distribution (with an approximate mean of 3.5), except for the Horror genre, which is characterized by low ratings.

userid	3	4	12	13	19	23	31	38	43	57	...	162507	162508	162512	162516	162519	162521	162524	162529	
movieid	1	4.0	3.0	4.0	4.0	0.0	0.0	0.0	0.0	4.0	4.0	...	4.0	4.5	4.0	4.5	5.0	4.0	4.5	2.0
	2	0.0	0.0	2.0	0.0	3.5	0.0	0.0	0.0	3.5	0.0	...	3.0	0.0	3.5	2.5	2.0	0.0	0.0	4.0
	3	0.0	0.0	2.0	0.0	0.0	5.0	0.0	0.0	0.0	3.5	...	4.0	0.0	3.5	0.5	0.0	0.0	0.0	1.0
	4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0
	5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	3.5	0.0	0.0	0.0	0.0	2.0
...
	205383	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	205425	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	206499	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	206805	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	207830	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

13176 rows x 32921 columns

Fig. 10. Matrix output for the movie recommendation

The matrix output generated for movie recommendations is displayed in Figure 10. In this case, each column represents a different movie ID, and each row represents a different user ID. This matrix, which was created using the planned hybrid system, shows suggestions for the users' preferred films. For each user, a certain movie's recommendation score is represented as a cell in the matrix.

Recommendations for movie American President, The (1995):
1. Twister (1996), recommendation score = 2.536
2. Mrs. Doubtfire (1993), recommendation score = 2.536
3. Client, The (1994), recommendation score = 2.536
4. Babe (1995), recommendation score = 2.535
5. E.T. the Extra-Terrestrial (1982), recommendation score = 2.535
6. Top Gun (1986), recommendation score = 2.534
7. Tin Cup (1996), recommendation score = 2.533
8. Field of Dreams (1989), recommendation score = 2.532
9. Crimson Tide (1995), recommendation score = 2.531
10. Shakespeare in Love (1998), recommendation score = 2.531
11. Few Good Men, A (1992), recommendation score = 2.5300000000000002
12. As Good as It Gets (1997), recommendation score = 2.528
13. Get Shorty (1995), recommendation score = 2.523
14. Birdcage, The (1996), recommendation score = 2.52
15. Dances with Wolves (1990), recommendation score = 2.5140000000000002
16. Mr. Holland's Opus (1995), recommendation score = 2.512
17. In the Line of Fire (1993), recommendation score = 2.511
18. Speed (1994), recommendation score = 2.508
19. True Lies (1994), recommendation score = 2.508
20. Apollo 13 (1995), recommendation score = 2.507

Fig 11. Recommendation for movie output

The results of the movie recommendations are shown in Figure 11. These suggestions are developed by combining conclusions or approximations obtained by combining the hybrid approach with K-means++ and IKSOM. The final product could include reviews, release years, movie names, and other pertinent details.

A variety of performance criteria were employed in order to assess the efficacy of the suggested methodology. These measurements include precision, accuracy, recall, mean absolute error, F1-score, and root mean square error. A detailed description of every performance metric is given below.

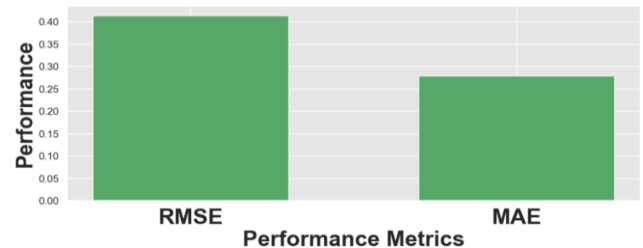


Fig. 12. Performance metrics of RMSE and MAE

The performance metrics for Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are shown in Figure 12. Using content-driven K-nearest neighbors (KNN) with cosine similarity, the hybrid approach shown in this work addresses the cold start issue and yields an RMSE of .512 and an MAE of 0.325. This technique makes rating predictions possible even in situations with limited data. Consequently, the suggested hybrid strategy has the ability to lower errors in RMSE and MAE by resolving issues with data sparsity and cold start.

The approaches outlined below were used to evaluate the suggested hybrid technique's accuracy, recall, and F1-score.

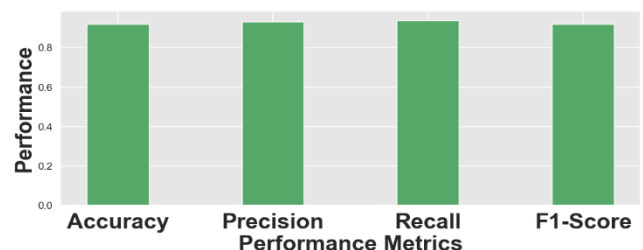


Fig. 13. Performance metrics of the proposed method

The performance metrics for the suggested method are shown in Figure 13. The proposed hybrid technique, which makes use of content-driven K-nearest neighbors (KNN), IKSOM, and K Means++ clustering, produced impressive results: 91.41% accuracy, 92.19% precision, 92.05% recall, and a 91.21% F1-score. This approach successfully addressed issues with data sparsity and scalability that are frequently present in movie recommendations. A review of the movie recommendations' quality included determining the accuracy, recall, and Root Mean Squared Error (RMSE). The hybrid system, as shown in the picture, attained an RMSE of 0.4140, compared to the prior system's RMSE values of 1.151 for user KNN, 1.044 for item KNN, 1.157 for Slope One, 1.143 for co-clustering, and 1.136 for

NMF.A detailed comparison between the suggested approach and some of the techniques currently in use is shown in Table 2.

TABLE 2: Comparison of Proposed Method with Other Existing Method.

Methods	RMSE	MAE
CO-Clustering	1.143	0.893
Non-Negative Matrix-Factorization-based approach (NMF)	1.136	0.894
Proposed Hybrid method	.512	0.325

The aforementioned study comes to the conclusion that current content-based filtering algorithms are unsuitable since they are laborious and inefficient when used on a variety of datasets. As a result, a hybrid strategy that combines content screening and collaborative techniques is suggested as a fix. It has been shown that the best approach for rating prediction combines matrix factorization with nearest neighbor selection. Cosine similarity is used to evaluate user similarity based on their rated films and recommend movies with similar users, hence resolving the cold-start issue. In the task of grading recommender systems, the hybrid system provided performs better than previous approaches in both the overall and cold start scenarios across significant evaluation criteria such as RMSE and MAE. High anticipated accuracy is the outcome of the hybrid system's capacity to learn complicated knowledge successfully. Therefore, the suggested hybrid system maintains excellent accuracy, precision, recall, and F1-Score while achieving low RMSE and MAE error rates. The suggested hybrid model offers recommendations for the top N films in the recommendation system that are more accurate than the present method.

Conclusion

This study proposed a hybrid movie recommendation system based on the Movielens_25M dataset. Initially, the Improved Singular Value Decomposition (SVD) matrix factorization technique was used to reduce the dataset's dimensions and features. After that, the user ratings, release year, and description were used to determine the degree of film resemblance using the Content-Driven K-nearest neighbors (KNN) method. Later, the IKSOM method was presented to assess the overlap between reduced clusters of nearest neighbors using the EISEN cosine correlation distance. The silhouette clustering method was used to estimate the ideal number of clusters for organizing movie data. After that, the hybrid methodology was used to create user and movie matrices and calculate scores. The best movie recommendations were then generated by using SVD collaborative filtering, which predicted item ratings and arranged the items in the appropriate order. With an RMSE of .512 and an MAE of 0.325, the suggested hybrid system outperformed the existing one in terms of error measures. With precision, recall, and F1-score values of 92.19%, 92.05%, and 91.21%, respectively, it also showed great accuracy. As a result, the aforementioned hybrid technique produces better results and provides customers with accurate movie recommendations. Future iterations of this

system might offer even more customized movie recommendations based on mood, location, and time of day in addition to user ratings, viewing preferences, and viewing histories.

Author contributions

Saurabh Sharma actively participated in the methodology, supervision, writing review and editing, research administration, visualization, investigation, formal analysis, and contributed substantially to the proofreading of the research article. **Dr. Ghanshyam Prasad Dubey** significantly contributed to various aspects, including data analysis, methodology, software, investigation, resource management, and writing the original draft. **Dr. Harish Kumar Shakya**, played a pivotal role in overseeing, guiding, conceptualizing, and collecting references for this research endeavor.

The authors assert that they do not have any conflicts of interest to disclose with respect to the current research.

The information supporting the conclusions of this research is obtainable from the corresponding author upon request via email at saurabhgyangit@gmail.com.

References

- [1] Alyari, F.; Navimipour, N.J. Recommender systems: A systematic review of the state of the art literature and suggestions for future research. *Kybernetes* 2018, 47, 985. [Google Scholar] [CrossRef]
- [2] Caro-Martinez, M.; Jimenez-Diaz, G.; Recio-Garcia, J.A. A theoretical model of explanations in recommender systems. In *Proceedings of the ICCBR, Stockholm, Sweden, 9–12 July 2018*. [Google Scholar]
- [3] Gupta, S. A Literature Review on Recommendation Systems. *Int. Res. J. Eng. Technol.* 2020, 7, 3600–3605. [Google Scholar]
- [4] Abdulla, G.M.; Borar, S. Size recommendation system for fashion e-commerce. In *Proceedings of the KDD Workshop on Machine Learning Meets Fashion, Halifax, NS, Canada, 14 August 2017*. [Google Scholar]
- [5] Aggarwal, C.C. An Introduction to Recommender Systems. In *Recommender Systems*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 1–28. [Google Scholar] [CrossRef]
- [6] Ghazanfar, M.A.; Prugel-Bennett, A. A scalable, accurate hybrid recommender system. In *Proceedings of the 2010 Third International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 9–10 January 2010*. [Google Scholar]
- [7] Deldjoo, Y.; Elahi, M.; Cremonesi, P.; Garzotto, F.; Piazzolla, P.; Quadrana, M. Content-Based Video Recommendation System Based on Stylistic Visual Features. *J. Data Semant.* 2016, 5, 99–113. [Google Scholar] [CrossRef][Green Version]
- [8] Alamdari, P.M.; Navimipour, N.J.; Hosseinzadeh, M.; Safaei, A.A.; Darwesh, A. A Systematic Study on the Recommender Systems in the E-Commerce. *IEEE*

Access 2020, 8, 115694–115716. [Google Scholar] [CrossRef]

- [9] Cami, B.R.; Hassanpour, H.; Mashayekhi, H. A content-based movie recommender system based on temporal user preferences. In Proceedings of the 2017 3rd Iranian Conference on Intelligent Systems and Signal Processing (ICSPIS), Shahrood, Iran, 20–21 December 2017. [Google Scholar]
- [10] Beniwal, R.; Debnath, K.; Jha, D.; Singh, M. Hybrid Recommender System Using Artificial Bee Colony Based on Graph Database. In Data Analytics and Management; Springer: Berlin/Heidelberg, Germany, 2021; pp. 687–699. [Google Scholar] [CrossRef]
- [11] Çano, E.; Morisio, M. Hybrid recommender systems: A systematic literature review. *Intell. Data Anal.* 2017, 21, 1487–1524. [Google Scholar] [CrossRef][Green Version]
- [12] Schafer, J.B.; Konstan, J.A.; Riedl, J. E-commerce recommendation applications. *Data Min. Knowl. Discov.* 2001, 5, 115–153. [Google Scholar] [CrossRef]
- [13] Shen, J.; Zhou, T.; Chen, L. Collaborative filtering-based recommendation system for big data. *Int. J. Comput. Sci. Eng.* 2020, 21, 219–225. [Google Scholar] [CrossRef]
- [14] Dakhel, G.M.; Mahdavi, M. A new collaborative filtering algorithm using K-means clustering and neighbors' voting. In Proceedings of the 11th International Conference on Hybrid Intelligent Systems (HIS), Malacca, Malaysia, 5–8 December 2011; pp. 179–184. [Google Scholar] [CrossRef]
- [15] Katarya, R.; Verma, O.P. An effective collaborative movie recommender system with cuckoo search. *Egypt. Inform. J.* 2017, 18, 105–112. [Google Scholar] [CrossRef][Green Version]
- [16] Kumar, B.; Sharma, N. Approaches, Issues and Challenges in Recommender Systems: A Systematic Review. *Indian J. Sci. Technol.* 2016, 9, 1–12. [Google Scholar] [CrossRef]
- [17] Sharma S, Shakya HK. Hybrid Real-Time Implicit Feedback SOM-Based Movie Recommendation Systems. In International Conference on Computing, Communications, and Cyber-Security 2022 Oct 21 (pp. 371-388). Singapore: Springer Nature Singapore.
- [18] Sharma S, Shakya HK. Recommendation System for Movies Using Improved version of SOM with Hybrid Filtering Methods. In 2023 6th International Conference on Information Systems and Computer Networks (ISCON) 2023 Mar 3 (pp. 1-7). IEEE.
- [19] Sharma S, Shakya HK. Recommendation Systems for a Group of Users Which Recommend Recent Attention: Using Hybrid Recommendation Model. In International Conference on Advanced Communication and Intelligent Systems 2022 Oct 21 (pp. 659-672). Cham: Springer Nature Switzerland.
- [20] Sharma, S., Shakya, H.K. and Marriboyina, V., 2021. A location based novel recommender framework of user interest through data categorization. *Materials Today: Proceedings*, 47, pp.7155-7161.
- [21] Sharma, S., Shakya, H.K. and Mishra, A., 2022. 11 Study and Estimation of Existing Software Quality. *Multi-Criteria Decision Models in Software Reliability: Methods and Applications*, p.235.
- [22] Sharma S, Mishra A, Shakya HK. A Time-Variant Software Stability Model for Error Detection. In Multi-Criteria Decision Models in Software Reliability 2022 Nov 30 (pp. 277-290). CRC Press.
- [23] Sharma S, Shakya HK. Hybrid Real-Time Implicit Feedback SOM-Based Movie Recommendation Systems. In International Conference on Computing, Communications, and Cyber-Security 2022 Oct 21 (pp. 371-388). Singapore: Springer Nature Singapore.