International Journal of Computer Sciences and Engineering
Vol.**11**, Issue.**7**, pp.**15-22**, July **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

Research Paper

An Improved Hybrid Recommender System Using Machine Learning Techniques

Mukul Kumar[1]

[1]UIET, M.D. University, Rohtak, India

Author's Mail Id: mukulpanchal97@gmail.com

Received: 28/May/2023; **Accepted**: 30/Jun/2023; **Published**: 31/Jul/2023. **DOI**: https://doi.org/10.26438/ijcse/v11i7.1522

Abstract: Recommender system is an AI-based tool which suggests items to users based on their preferences, helping overcome information overload and improving user experience. This paper provides an introduction to recommender systems and their applications in variety of fields such as music streaming, networking sites, internet shopping, and digital media platforms. It highlights the benefits of personalized recommendations and identifies common challenges faced by recommender systems, including privacy concerns, cold start, data sparsity, and scalability issues. The paper proposes a hybrid model that combines content-based filtering (CBF) and collaborative filtering (CF) techniques for movie recommendations. The Movielens 1M dataset is used for evaluation, and the performance of the model is measured using the root mean squared error (RMSE). The results show that the hybrid recommender system outperforms both CBF and CF systems in terms of RMSE and accuracy, providing more accurate and personalized movie recommendations.

Keywords: Recommender System; Hybrid Recommender System; Root Mean Square Error; Machine Learning.

1. Introduction

To address the problem of information overload, recommender systems are required to provide personalized suggestions, assist users in discovering relevant content, and improve their user experience. Recommender system is an AI-based tool which gives personalized suggestions based on our preferences and interests [1]. It helps us to discover new things we might like, such as movies, music, products, and more. By analyzing our data and behavior, recommender systems save our time and effort by offering options that align with our tastes. RS promotes consumer happiness, engagement, and decision-making processes, resulting in higher revenue and user retention for enterprises. Here are some examples of where recommender systems are commonly used [4,5]:

- Spotify, a digital music, podcast, and video service, uses a recommender system as well to give its users suggestions for songs, podcasts, and videos.
- LinkedIn is a social media site that people use to build and maintain professional relationships and find employment or internships. In order to offer the user, the best job, it also makes use of a RS.
- Flipkart (online shopping website) uses recommender system that helps users discovering products similar to the ones they have browsed earlier.
- Amazon kindle (e-reader) also uses recommender system for recommending books, newspapers, magazines and other digital media.

Recommender systems work by analyzing user data and item characteristics to generate personalized recommendations. They employ algorithms and techniques such as machine learning and data mining to identify patterns and relationships. Content-based filtering examines the attributes of items and recommends similar ones based on user preferences. Collaborative filtering considers user behavior and recommends items liked by similar users. Hybrid approaches combine both methods for enhanced accuracy. By reducing information overload and providing relevant suggestions, recommender systems improve user experience and assist in decision-making. While recommender systems offer numerous benefits, they also face several challenges and problems. Here are some common problems associated with recommender systems [4,13]:

- **Cold Start:** When we lack any prior or historical information about the user or item, a cold start problem occurs. As his/her taste is unknown to the system, it is difficult for the system to recommend in such a situation.
- **Privacy:** Nowadays, privacy is a major concern while developing a recommender system. As cybercrime becomes more prevalent, people are unwilling to share personal information such as their name, age, sex, email address, phone number, or location. These demographic

data are required to provide a relevant suggestion system, but they may violate the user's privacy.
- **Sparsity:** Sparsity refers to the phenomenon where the user-item interaction data is extremely sparse.
- **Scalability:** As the recommender system's dataset and the number of users grow over time, users expect the recommender system to respond within a section so that they remain engaged. This is one of the problems that occurred with CF approach [2].

This paper gives an introduction to RS, which are designed to offer personalized suggestions and mentions some common applications of recommender systems. In this paper, a hybrid model is proposed using both CBF and CF techniques and after experimenting on MovieLens dataset, the proposed model is proven to be best in terms of recommendations accuracy rate. This section also highlights the overview and the challenges faced by recommender systems. Section II focuses on the related work in recent years by different researchers. Section III focuses on the proposed model, which combines CBF and CF techniques. Fig. 1. depicts the suggested model's architecture. Movielens 1M dataset is used for evaluation. The implementation of the model involves preprocessing the dataset, dividing it into tést and training séts. Section IV assesses the model's performance using RMSE measure. Hybrid recommender system, which combines both approaches, achieves the lowest RMSE value, indicating superior performance. The section also presents the accuracy analysis of the recommender systems, showing that the hybrid system achieves the highest accuracy, followed by the CF system and the CBF. In section V**,** conclusion of the work is provided.

2. Related Work

In this section, some of the RSs looked into by the authors of previously published works are discussed.Walek, et al. [31] used monolithic hybrid RS in this paper. It is a combination of content-based system, collaborative filtering system (using the SVD algorithm) and fuzzy expert system. The system verification results based on standard metrics (precision, recall, and F1-measure) are greater than 80%. the proposed model outperforms when compared to others in terms of relevant ratio. There are several difficulties in doing RS. One of them is scalability and cold start. Devi, et al. [32] used an upgraded hybrid approach to solve the Scalability and Cold start problem. A naïve Bayesian classifier with Gaussian correction and feature engineering is used in this approach. The author also calculated prediction performance using mean absolute error. When utilising the MovieLens 100k dataset, experimental findings outperform previous approaches. Agrawal, et al. [40] proposed a hybrid approach that combined both CBF and CF techniques. The methodology uses a genetic algorithm and support vector machine as classifiers. The MovieLens 1M dataset, MovieLens Latest Small dataset, and MovieLens 10M dataset were the three different datasets used by the authors. The proposed movie recommendation system's quality is evaluated using common quality metrics like Mean Absolute Error, precision, recall, f-measure, and coverage. Although the proposed approach requires more memory than the existing approaches, it requires less computing time. To represent people's online shopping behaviours, the author created a database of sequences and extract the common behavioural patterns from the sequences database. Fang, et al. [41] integrate traditional model with behavior pattern extraction method and proposed an approach. Proposed approach is combination of CF and Behaviour Prediction Modél. To create a hybrid dynamic recommender system, T-mall and Alibaba's desensitised mobile transaction record is used. The algorithm is found to be more personalised and to provide high-quality recommendations after testing it on the dataset. Implementing collaborative filtering has run into numerous issues. One of them is Cold Start. Additionally, it is unable to offer the user-specific recommendations. A hybrid recommendation system was proposed by Duzen, et al. [42] to address these problems. By combining collaborative filtering and Case-Based Reasoning, they put into practise a hybrid recommendation system. Input data for the proposed approach is taken from LastFm, which contains user-listened music tracks over a specified time period. The outcomes show that our suggested approach is effective. Pal, et al. [43] present a fresh alternative. They showed how to use set intersection to determine the connection between two characteristics in content-based filtering, as well as how to use Collaborative Filtering to determine the similarity between two items and predict them for recommendations. The dataset used is from MovieLens. For Evaluating the technique MAE is employed. For college students, Khoja, et al. [44] created a course recommendation system. Given the shortcomings and advantages of both collaborative and content-based filtering, the proposed system combines them. Students can use it to identify their interests and guide better course selection when registering for college courses. Websites offering online courses, such as edX and udemy, can also use it.

3. Proposed Model

A hybrid technique with weights allocated is presented in this study, and a broad flow chart of the model is provided. Initially, all datasets are read, and then two distinct datasets are generated from the original data set. Following the grouping of users and movies, a ML algorithm is applied to the dataset to provide predictions. The suggested method combines CBF and CF methods [39].

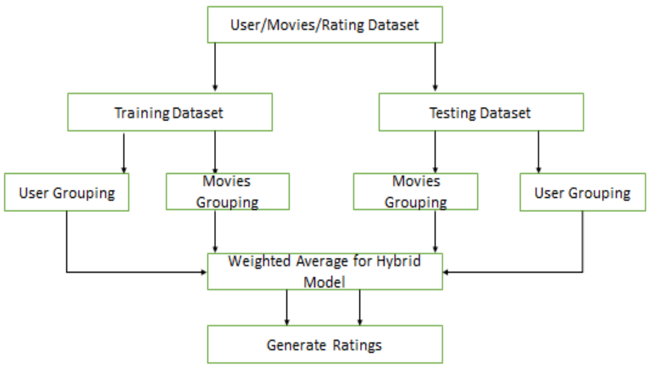

Figure 1 Model's Architecture

The model is evaluated using the Movielens 1M dataset [34]. The dataset is separated into users, movies, and ratings of movies. All of the MovieLens dataset's tables are in.csv format.

Users: The "users.csv" file in the MovieLens 1M dataset contains user information. Each entry contains the following information: UserID, Gender, Age, Job, and Postal Code. Users provide this information voluntarily, but it is not verified [34]. The dataset only includes users who share their demographic information. Gender is represented by the capital letters "M" for male and "F" for female. Age is divided into categories such as "Under 16," "16-22," "23-30," "31-45," "46-54," "55-64," and above 64. Among the occupation options are "student," "doctor," "programmer," and others. When analysing movie ratings and making recommendations, researchers can use this data to better understand the characteristics and preferences of users.

Movies: The "movies.csv" file in the MovieLens 1M dataset contains movie information. Each entry contains the following information: MovieID, Title, and Genres. The movie titles are identical to those provided by IMDB. The "movies.csv" file contains a large amount of movie data which can be used for a variety of research purposes. It allows researchers to study movie genres, analyze user preferences within specific genres, and develop recommendation algorithms based on genre similarities or user preferences.

Ratings: In the MovieLens 1M dataset, the "ratings.csv" file contains user ratings for movies. The UserID, MovieID, Rating, and Timestamp are all included in each rating entry. The timestamp indicates when the rating was recorded. This dataset is useful for researchers to analyse patterns and preferences and for understanding how users rate movies. Each user provided minimum of 20 ratings, resulting in a large amount of data for studying movie preferences and developing recommendation systems.

The implementation of the model begins by reading three CSV files containing user data, movie ratings, movie information. It preprocesses the user data by grouping users into male and female age groups such as "Under 16," "16-22," "23-30," and so on.

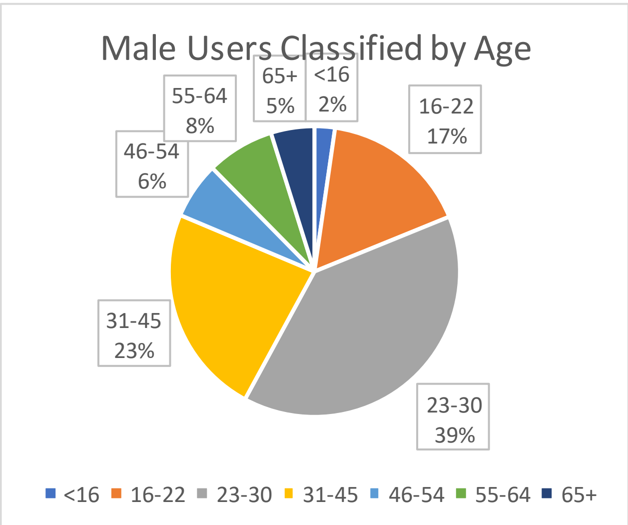

Figure 2 Male Users Distribution

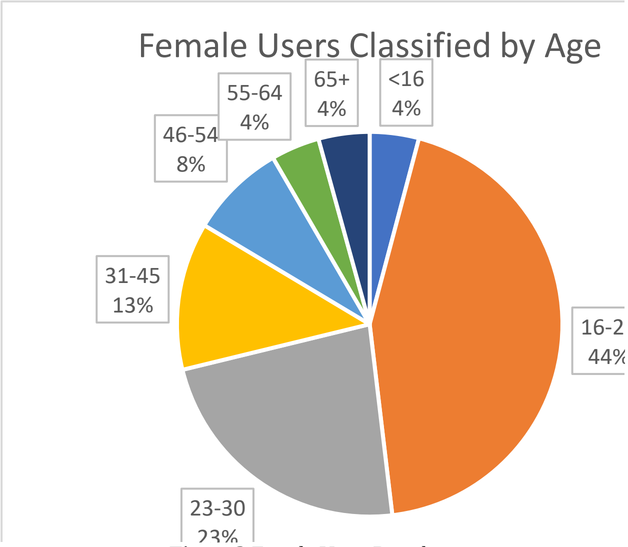

Figure 3 Female Users Distribution

As shown in Figures 2 and 3, the distribution of users is based on gender as well as age. Figure 2 shows that male users aged 23 to 30 make up the majority of users, while Figure 3 female users aged 16 to 22 make up the majority. Grouping of user data into male and female age group is done to analyse and understand the distribution of users based on their age and gender. By grouping the data, we can drive insight and statistics about different age segments within each gender category. Further, it also calculates the count of users in different age ranges for both genders.

To analyse the trend of ratings or to identify the genre with the highest rating, we must divide the rating by genre. The code allows for a thorough examination of rating distribution by extracting the count of specific ratings as well as the total number of ratings for each genre, providing insights into the popularity and user preferences for various genres.

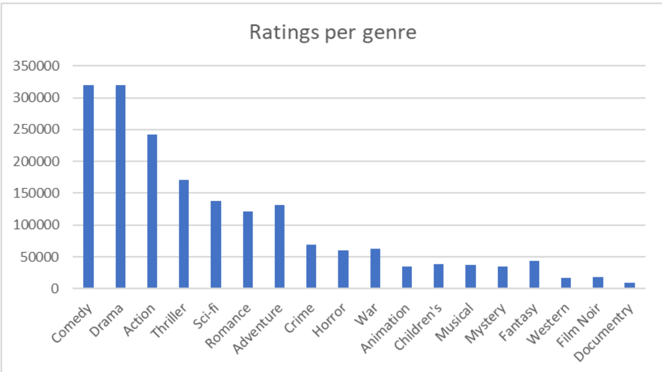

Figure 4 Ratings per genre

In Figure 4, we can see that there are many more action, comedy, and drama movies ratings in the dataset compared to other genres. This means that these genres have a larger number of movies ratings available for analysis. Because of this, I expect the average rating for these popular genres to be similar to the overall average rating of all movies. On the other hand, genres like Western, Documentary, or Film-Noir have fewer movies ratings in the dataset.

The employed dataset is then separated into two separate datasets i.e., first one is training dataset and second is test datasets in the next phase. When examining the performance and generalisation capabilities of a machine learning model, it is necessary to partition the dataset into sets for training and for testing. The training set is used to train the model so that it recognises patterns and correlations in the data. In contrast, the testing set is used to evaluate the model's performance on previously unknown data. We can estimate how well the model would perform on fresh, unknown data and examine its ability to generalise beyond the training data by evaluating its predictions on the testing set. to divide the set of data into training sets and testing sets, we used the "train_test_split" function from the "sklearn.model_selection" package. The train-test split function is a method used in machine learning and data analysis to divide a dataset. The train-test split function randomly partitions the dataset into two subsets based on a specified ratio or size. The training set receives 70% of the data, whereas the test set receives 30%. After that the training dataset is merged with movie information. To prepare the dataset which is in suitable format for the ML algorithms to work effectively and ensuring compatibility with the chosen machine learning algorithms, we needed preprocessing of the dataset. Preprocessing also includes handling missing values, removing irrelevant or redundant features, and transforming the data into a consistent and standardized format. It also enabling accurate training and evaluation of the models. It helps to enhance the quality of the input data, enhance model performance, and ensure reliable and meaningful results.

In the next phase, we train and evaluate a content-based filtering model that predicts movie ratings based on movie features such as genres. In CBF model, the information about a user's movie preferences to predict recommendations is used. Specifically, the CBF model consider what movies the user has rated, how they rated them, and the genres of those movies. Based on this information, the model creates a personalized model for each user. Using this personalized model, the CBF model can predict ratings for other movies in the dataset that the user hasn't seen yet. The models analyze the genres of these unseen movies and compare them to the user's preferences and rated movies. By doing this, the models make predictions about how much the user would like or dislike these unseen movies. For classification SVR is used. SVR (Support Vector Regression) is an approach used to predict user preferences for the items based on their attributes or features. Its primary function is classification. It is employed as a regression model to forecast the numerical ratings or preferences that a user may assign to various items. A regularisation parameter value is used to initialize the SVR algorithm. This value indicates a moderate regularisation strength, with the goal of striking a balance between model complexity and training data fit [35]. The RMSE is used to determine the model's effectiveness [36].

Additionally, the model also performs CF by calculating the Pearson Distance between users in the training data. It creates a distance matrix and trains k-nearest neighbors (kNN) models using the Pearson Distance as a similarity metric [37]. In this case, user-based collaborative filtering is utilised. due to the fact that suggestions are exclusively based on user' shared interests and prior interactions. Age, gender, and location are not taken into account while making suggestions. The algorithms forecast movie ratings by analysing the ratings of comparable users. The performance of CF models is assessed using the RMSE. Both CBF and CF have their strengths and weaknesses, but when we merge them, we hope to overcome the limitations of each model. Finally, the code integrates the methodologies of CBF and CF in a hybrid model. Weights are used to combine CF and CBF models in a hybrid RS. These weights determine the relative importance or contribution of each model to the final recommendation. The purpose of assigning weights is to balance the influence of CF and CBF in the hybrid model. By adjusting the weights, we can control the degree to which each model's recommendations impact the final results. This allows us to emphasize the strengths of each approach and compensate for their weaknesses. After that, RMSE is used to evaluate performance.

Overall, proposed system for recommending movies leverages both CBF and CF techniques to provide personalized movie suggestions based on user interests and shared characteristics with other users is implemented.

4. Results

Various analytical metrics are employed in recommender systems to examine and assess the performance and efficacy of the recommendation algorithms. Some common analysis measures in RS include [38]: -

- Precision: Precision calculates the proportion of relevant items recommended out of the total recommended items. It focuses on suggestion accuracy, revealing how many of the recommended things are genuinely relevant to the user.
- Recall: Recall measures the proportion of relevant items recommended out of the total relevant items available. It emphasizes the coverage of the recommendations, indicating how well the system can retrieve relevant items.
- RMSE: The RMSE is a statistic that evaluates the average difference between projected and actual values, with lower values indicating higher predicting accuracy.

In this paper, RMSE and accuracy is used as an analysis measure. They are frequently employed in RS to evaluate the prediction quality of the system's recommendations. RMSE, which stands for Root Mean Squared Error, is a measure used to assess how accurately a predictive model's predictions match the actual values. In simpler terms, it quantifies the average difference between the predicted values and the observed values. Due to the fact that it uses real measurements at each projected data point and requires them, RMSE is frequently utilised in supervised learning applications. The formula for RMSE is as follows [36]:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(a_i - p_i)^2}{n}}$$

Where:

n = the total number of observations.

a_i = represents the actual or observed value.

p_i = represents the predicted value.

RMSE is useful because it provides a single value that represents the typical or average error of the model's predictions. Lower RMSE values suggest a better match between the anticipated and observed values, indicating more accuracy.

Accuracy is an also another common evaluation metric used to measure the performance and correctness of a classification or prediction model. It represents the proportion of correct predictions made by the model out of the total number of predictions.

The formula for accuracy is:
Accuracy = (Number of Correct Predictions) / (Total Number of Predictions)

In this formula, the "Number of Correct Predictions" refers to the count of instances where the model's prediction matches the ground truth or the true labels. The "Total Number of Predictions" represents the overall number of instances or observations that the model has made predictions on. The accuracy value is typically expressed as a percentage, ranging from 0% to 100%. A higher accuracy indicates that the model has made more correct predictions, while a lower accuracy suggests that the model's predictions are less accurate.

1. Content -based RS

The purpose of content-based RS is to suggest goods on the basis of their content or qualities [1]. In the case of movie recommendations, the system or model makes use of information about the users' movie preferences, such as the ratings they have given to specific movies and the genres of those movies. The system creates personalised models for each user based on user movie ratings and movie genres. These models are then used to predict ratings for movies that the user has not yet seen, allowing personalised movie recommendations to be generated [2,4,17].

Table 1. Models' Performance

Model	RMSE
SVR	1.054499

The RMSE is a metric used to evaluate regression tasks (i.e. SVR). The performance of SVR model is represented by the value "1.054499" as shown in Table 1. This value represents the SVR algorithm's level of accuracy or error in predicting the target variable. A lower value indicates better predictive performance, implying that the SVR model has achieved a relatively small error in its predictions [36].

2. Collaborative Filtering RS

We created a CF-based RS using a user-based collaborative method. This technique offers suggestions based on comparable users' conduct and preferences [18]. The KNN algorithm, with a specific value of 'k' set to 25, is used. The KNN algorithm works by identifying the 'k' most similar users to a target user based on their item preferences. In this case, we consider the 25 users who are the most similar to our target user. These similar users have previously demonstrated similar tastes and preferences. Once we've identified these closest neighbours, we can use their ratings and opinions to make recommendations for the target user [37].

Table 2. Models' Performance.

Model	RMSE
KNN with k =25	0.984499

The RMSE measure is utilized in our experiment to assess the performance of the KNN algorithm. The average difference between the expected and actual ratings for the items is shown by the RMSE of 0.984499 as shown in Table 2. A lower RMSE number implies higher performance since it means the algorithm's predictions are more accurate. Comparing the outcomes of both models, as given in tables 1 and 2, it is evident that the CF model has a lower value, indicating that it performs better than the CBF model.

3. Hybrid Recommender System

Both content-based RS and collaborative filtering RS have advantages and disadvantages, but when combined, they complement each other and outperform when compared to pure versions of themselves in terms of performance. So, by combining both of the above approaches, we can create a superior hybrid recommender system. The model's performance is then calculated using RMSE [32]. Table 3. Shows the RMSE value of the proposed model.

Table 3. Models' Performance.

Model	RMSE
Hybrid RS	0.925439

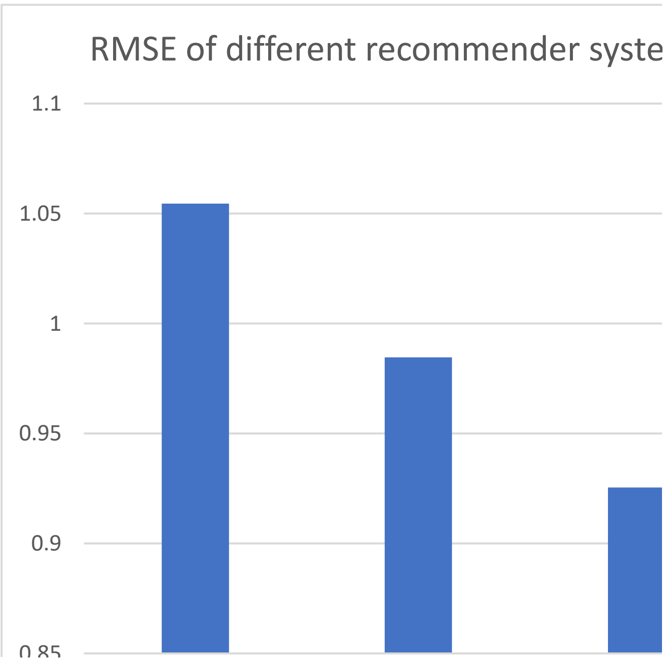

Fig. 5. Architecture Performance analysis of RS

A lower RMSE number implies higher performance since it means the algorithm's predictions are more accurate. The

Collaborative RS outperforms the Content RS, while the Hybrid RS beats both the Collaborative RS and the Content RS in terms of RMSE [36]. Fig. 5. Shows the compared results of the different model. We can clearly conclude that Hybrid RS has lowest RMSE value and it performs better. The higher performance of the Hybrid RS implies that combining collaborative and content-based filtering strategies improves the accuracy and quality of the suggestions.

After evaluating the performance of the various recommender system approaches using RMSE, an accuracy-based analysis is also carried out. The percentage of items that are accurately recommended out of all the recommendations made is defined as accuracy. A higher accuracy shows better alignment between suggested and relevant items, which reflects how well the system works to deliver individualised suggestions.

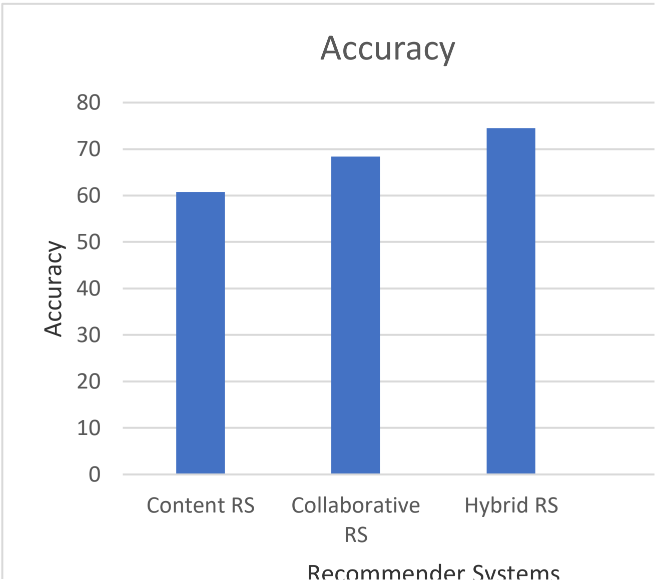

Fig. 6. Accuracy of Different RS

Fig. 6. shows that, among the others, the hybrid recommender system has the highest accuracy of 74.48%. The bar graph shows that the collaborative recommender system has accuracy of 68.39% while the content-based recommender system has the lowest accuracy of 60.78%.

5. Conclusion and Future Scope

This paper discusses the findings of a research that used a mix of content-based filtering (CBF) and collaborative filtering (CF) approaches to create a movie recommendation system. The Movielens 1M dataset was used to test the suggested model. The hybrid recommender system, which integrates both CBF and CF techniques, outperformed the separate content-based and collaborative filtering systems, according to the results. The RMSE measure was used to assess the performance of the models. The hybrid recommender system achieved the best results with an RMSE of 0. 925439.In addition to RMSE, the accuracy of the recommendation systems was also evaluated. The hybrid recommender system achieved an accuracy of 74.48%, outperforming both the collaborative and content-based systems. The collaborative filtering system achieved an accuracy of 68.39%, while the content-based system had the lowest accuracy at 60.78%. One of the model's potential future additions is the incorporation of textual and contextual data for enhanced recommendation accuracy. Use user feedback and incremental learning to modify recommendation models over time. Modifying code to provide real-time suggestions, rapid processing of large datasets, and timely recommendations. We can also design a user-friendly interface.

Conflict of interest
None

Funding Sources
None

Author's contribution
As the lone author, I developed and designed the study, collected and preprocessed data, and contributed to the hybrid recommender system's implementation.

Acknowledgment
Dr. Dhiraj Khurana, University Institute of Engineering & Technology (UIET MDU) M. D. University, Rohtak, Haryana, has provided constant support, advice, and essential insights throughout the study process. His guidance has been crucial in improving the quality and precision of this work. I am also grateful to my colleagues and friends for their encouragement, support, and insightful talks, which have greatly aided this study.

References

[1] Wang, D., Liang, Y., Xu, D., Feng, X., & Guan, R. A content-based recommender system for computer science publications. *Knowledge-Based Systems*, *157*, **1-9, 2018.**

[2] Ferdousi, Z. V., Colazzo, D., & Negre, E. (March). Correlation-based pre-filtering for context-aware recommendation. In 2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops) pp.**89-94, 2018.** IEEE.

[3] Aghdam, M. H. Context-aware recommender systems using hierarchical hidden Markov model. Physica A: Statistical Mechanics and Its Applications, 518, pp.**89-98, 2019.**

[4] Abbas, A., Zhang, L., & Khan, S. U. A survey on context-aware recommender systems based on computational intelligence techniques. Computing, 97, pp.**667-690, 2015.**

[5] Cami, B. R., Hassanpour, H., & Mashayekhi, H. (December). A content-based movie recommender system based on temporal user preferences. In 2017 3rd Iranian conference on intelligent systems and signal processing (ICSPIS) (pp.**121-125, 2017.** IEEE.

[6] Ding, W., Xu, R., Ding, Y., Zhang, Y., Luo, C., & Yu, Z. (December). Context aware recommender system for large scaled flash sale sites. In 2018 IEEE International Conference on Big Data (Big Data) pp.**993-1000, 2018.** IEEE.

[7] Nadschläger, S., Kosorus, H., Boegl, A., & Kueng, J. (September). Content-based recommendations within a QA system using the hierarchical structure of a domain-specific taxonomy. In 2012 23rd International Workshop on Database and Expert Systems Applications, pp.**88-92, 2012.** IEEE.

[8] Mandave, D., & Pole, G. (August). A Syntactic Content-Based Recommender Based on combination of ACO and GA in large scholarly data. In 2017 International Conference on Computing,

Communication, Control and Automation (ICCUBEA), pp.**1-6, 2017.** IEEE.
[9] Dooms, S., Audenaert, P., Fostier, J., De Pessemier, T., & Martens, L. In-memory, distributed content-based recommender system. Journal of Intelligent Information Systems, 42, pp.**645-669, 2014.**
[10] Sundermann, C. V., Domingues, M. A., Marcacini, R. M., & Rezende, S. O. (October). Using topic hierarchies with privileged information to improve context-aware recommender systems. In 2014 Brazilian Conference on Intelligent Systems, pp.**61-66, 2014.** IEEE.
[11] Son, J., & Kim, S. B. Content-based filtering for recommendation systems using multiattribute networks. Expert Systems with Applications, 89, pp.**404-412, 2017.**
[12] Sun, F., Shi, Y., & Wang, W. (August). Content-based recommendation system based on vague sets. In 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics, Vol.**2**, pp.**294-297, 2013.** IEEE.
[13] Lops, P., De Gemmis, M., & Semeraro, G. Content-based recommender systems: State of the art and trends. Recommender systems handbook, pp.**73-105, 2011.**
[14] Zarzour, Hafed, et al. "A new collaborative filtering recommendation algorithm based on dimensionality reduction and clustering techniques." *2018 9th international conference on information and communication systems (ICICS)*. IEEE, **2018.**
[15] Sneha, V., Shrinidhi, K. R., Sunitha, R. S., & Nair, M. K. (July). Collaborative filtering-based recommender system using regression and grey wolf optimization algorithm for sparse data. In 2019 International conference on communication and electronics systems (ICCES), pp.**436-441, 2019.** IEEE.
[16] Pujahari, A., & Sisodia, D. S. Pair-wise preference relation based probabilistic matrix factorization for collaborative filtering in recommender system. Knowledge-Based Systems, 196, 105798, **2020.**
[17] Anwar, T., & Uma, V. (March). Mrec-crm: Movie recommendation based on collaborative filtering and rule mining approach. In 2019 international conference on Smart Structures and Systems (ICSSS) pp.**1-5, 2019**. IEEE
[18] Gazdar, A., & Hidri, L. A new similarity measure for collaborative filtering-based recommender systems. *Knowledge-Based Systems*, *188*, 105058, **2020.**
[19] Mustaqeem, A., Anwar, S. M., & Majid, M., A modular cluster based collaborative recommender system for cardiac patients. Artificial intelligence in medicine, 102, 101761, **2020.**
[20] Fazziki, A. E., El Aissaoui, O., El Alami, Y. E. M., Allioui, Y. E., & Benbrahim, M. (October). A new collaborative approach to solve the gray-sheep users problem in recommender systems. In 2019 Third International Conference on Intelligent Computing in Data Sciences (ICDS), pp.**1-4, 2019.** IEEE.
[21] Ghaleb, H., & Abdullah-Al-Wadud, M. (December). An Enhanced Similarity Measure for Collaborative Filtering-based Recommender Systems. In 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), pp.**1-4, 2019.** IEEE.
[22] Véras, D., Prudêncio, R., & Ferraz, C. (2019). CD-CARS: Cross-domain context-aware recommender systems. Expert Systems with Applications, 135, pp.**388-409, 2019.**
[23] Martín-Vicente, M. I., Gil-Solla, A., Ramos-Cabrer, M., Pazos-Arias, J. J., Blanco-Fernández, Y., & López-Nores, M. (2014). A semantic approach to improve neighborhood formation in collaborative recommender systems. Expert Systems with Applications, Vol.**41**, Issue.**17**, pp.**7776-7788, 2014.**
[24] Al-Shamri, M. Y. H. (2014). Power coefficient as a similarity measure for memory-based collaborative recommender systems. Expert Systems with Applications, Vol.**41**, Issue.**13**, pp.**5680-5688, 2014.**
[25] Ayyaz, S., & Qamar, U. (March). Improving collaborative filtering by selecting an effective user neighborhood for recommender systems. In *2017 IEEE International Conference on Industrial Technology (ICIT)*, pp.**1244-1249, 2017.** IEEE.
[26] Chen, M. H., Teng, C. H., & Chang, P. C. (2015). Applying artificial immune systems to collaborative filtering for movie recommendation. Advanced Engineering Informatics, Vol.**29**, Issue.**4**, pp.**830-839, 2015.**
[27] Guo, Y., & Deng, G. (2006, October). An improved personalized collaborative filterinng algolrithm in E-commerce recommender system. In 2006 International Conference on Service Systems and Service Management, Vol.**2**, pp.**1582-1586.** IEEE.
[28] Al Hassanieh, L., Abou Jaoudeh, C., Abdo, J. B., & Demerjian, J. (April). Similarity measures for collaborative filtering recommender systems. In 2018 IEEE Middle East and North Africa Communications Conference (MENACOMM), pp.**1-5, 2018.** IEEE.
[29] Murali, M. V., Vishnu, T. G., & Victor, N. (March). A collaborative filtering-based recommender system for suggesting new trends in any domain of research. In 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), pp.**550-553, 2019**. IEEE.
[30] Zhang, Yihao, et al. "Hybrid recommender system using semi-supervised clustering based on Gaussian mixture model." *2016 international conference on cyberworlds (CW)*. IEEE, **2016.**
[31] Walek, B., & Fojtik, V., A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, *158*, 113452, **2020.**
[32] Devi, S. S., & Parthasarathy, G. (April). A hybrid approach for movie recommendation system using feature engineering. In 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), pp.**378-382, 2018.** IEEE.
[33] Paradarami, T. K., Bastian, N. D., & Wightman, J. L., A hybrid recommender system using artificial neural networks. Expert Systems with Applications, 83, pp.**300-313, 2017.**
[34] Zhuhadar, Leyla, et al. "Multi-model ontology-based hybrid recommender system in e-learning domain." *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*. Vol. 3. IEEE, **2009.**
[35] Rho, Seungmin, Byeong-jun Han, and Eenjun Hwang. "Svr-based music mood classification and context-based music recommendation." *Proceedings of the 17th ACM international conference on Multimedia*. **2009.**
[36] Amatriain, Xavier. "Building industrial-scale real-world recommender systems." *Proceedings of the sixth ACM conference on Recommender systems*. **2012.**
[37] Grčar, Miha, et al. "kNN versus SVM in the collaborative filtering framework." *Data Science and Classification*. Springer Berlin Heidelberg, **2006.**
[38] Schwarz, Mykhaylo, Mykhaylo Lobur, and Yuriy Stekh. "Analysis of the effectiveness of similarity measures for recommender systems." *2017 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. IEEE, **2017.**
[39] Pathak, Dharmendra, Sandeep Matharia, and C. N. S. Murthy. "ORBIT: Hybrid movie recommendation engine." *2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN)*. IEEE, **2013.**
[40] Agrawal, S., & Jain, P. (February). An improved approach for movie recommendation system. In 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), pp.**336-342, 2017.** IEEE.
[41] Fang, Z., Zhang, L., & Chen, K. (March). A behavior mining based hybrid recommender system. In 2016 IEEE International Conference on Big Data Analysis (ICBDA), pp.**1-5, 2016.** IEEE
[42] Duzen, Z., & Aktas, M. S. (August). An approach to hybrid personalized recommender systems. In 2016 International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), pp.**1-8, 2016.** IEEE.
[43] Pal, A., Parhi, P., & Aggarwal, M. (August). An improved content based collaborative filtering algorithm for movie recommendations. In 2017 tenth international conference on contemporary computing (IC3), pp.**1-3, 2017.** IEEE.
[44] Khoja, Z., & Shetty, S. (December). Hybrid recommender system for college courses. In 2017 International Conference on Computational Science and Computational Intelligence (CSCI), pp.**1167-1171, 2017**. IEEE.

AUTHORS PROFILE

Mukul Kumar earned his B. Tech. in computer science from Vaish College of Engineering, Rohtak, Haryana, India. I am currently pursuing in M. Tech in computer science from University Institute of Engineering & Technology (UIET MDU) M.D. University, Rohtak, Haryana, India. His areas of interest include Recommender System and Machine Learning.

