Next-Generation Movie Recommendation Systems: A Hybrid Collaborative Filtering Approach

Sireesha Moturi [1], Rongala Gangadhar [2], Nallamekala Gopikrishna [3], Kunchapu Akhil [4], Dodda Venkata Reddy [5]
[1, 5] Associate Professor, [2, 3, 4] Student
[1] sireeshamoturi@gmail.com, [2]gangadhar2820@gmail.com, [3]gopikrishna.nallamekala0405@gmail.com,
[4]Kumaradeiah9959436566@gmail.com, [5] doddavenkatareddy@gmail.com
Department of Computer Science and Engineering,
Narasaraopeta Engineering College, Narasaraopet, Andhra Pradesh, India

ABSTRACT- With the expansion of the internet, individuals are now able to get a lot of information on a lot of things. But if there are too many options, it is difficult to determine what is best for them. Recommendation systems are systems that solve this issue by providing alternatives that are suitable for users' interests. Typically, these systems belong to two broad categories: content-based, which recommend items similar to what a user is interested in, and collaborative filtering, which recommend items enjoyed by similar users. Researchers are investigating hybrid systems, which combine various approaches, to counter the limitations of conventional recommendation systems. Recommendations can further become more helpful by incorporating context. Methods such as direct context modeling, post-filtering, and pre-filtering can be employed to incorporate context information. For enhancing movie recommendations, this paper presents a new hybrid approach that employs pre- and post-filtering methods. This system is based on a huge database that contains ratings, context information, item descriptions, and user profiles. Pre-filtering is the initial step, where the database is simplified by being reduced based on a significant context detail that is applicable for the user. Then, a post-filtering process enhances the recommendations by taking into account other context features, so that the suggestions are as pertinent as possible to the user's context.

KEYWORDS: Movie Recommendation, Content-based, Collaborative-based, Hybrid, SVD

I. INTRODUCTION

Recommendation algorithms assist in making customers happier and more engaged with digital entertainment. Streaming media websites that display videos and films utilize these algorithms to generate personalized recommendations based on the users' interests, and therefore the service experience is improved. Two of the numerous methods utilized are Content-based Filtering (CBF) and Collaborative Filtering Collaborative filtering (CF) is extremely prevalent in this field [1,2,4]. This is a new technology that enhances product and service suggestions based on a consumer's previous actions. It has been extremely crucial to numerous companies, such as large internet motion picture streaming platforms like Netflix, Amazon Prime Video, and Hulu, e-commerce and music applications [5-8]. But such systems are suffering from various issues, particularly while recommending to new users having either no or very little interaction history. This is referred to as the "Cold-start" issue, and this poses critical issues for CF-based models [7,13]. Content-based and Collaborative Filtering methods both suffer from issues which have not been altered by the passage of time, even though they still apply in generating recommendations [9,10,12,14]. Systems that provide recommendations to users are essential for pointing them in the direction of goods or services, possibly influencing purchases that generate income for companies. The skillful application of these technologies has a substantial impact on a company's financial results in addition to enhancing user interaction by providing useful recommendations. Businesses can significantly improve the movie-watching experience by employing movie recommendation systems well. This will draw and keep customers, which will boost revenue growth and business expansion [11, 15, 16, 17].

II. LITERATURE SURVEY

In order to improve recommendation accuracy, S. Agrawal et al. [1] Proposed a hybrid strategy that combines content-based filtering and collaborative filtering techniques, employing Support Vector Machines (SVM) as a classifier. Their analysis, which used the Movie Lens dataset as a basis, showed a notable improvement above conventional technique. Similar comparisons were made between other recommender systems by N. Vaidya et al. [2], including keyword-based, hybrid, content-based, collaborative filtering, and demographic systems. They outlined the benefits and limitations of each strategy, stressing the value recommender systems in practical applications especially e-commerce platforms to increase revenue and enhance customer satisfaction

Singular Value Decomposition (SVD) algorithm was used collaboratively by Y. Xiong et al. [4] to suggest photographs with comparable styles. To increase accuracy, they reduced dependencies in training data, optimized the recommendation algorithm, and employed binary ratings (1 for used, 0 for not used). S. Bhat et al. [5] addressed the cold-start issue in their recommendation system with a feature-based solution by grouping similar users based on their preference. J. Chen et al. [6] enhanced the accuracy of their recommendations by proposing tourist attractions with reviews from actual travelers, taking into account the cost, accommodations, and travel. P. Darshna et al. [7] developed a collaborative filtering and content-based filtering experiment for music recommendation. They used k-means clustering to consider attributes such as loudness and quality of sound. S. Girase et al. [8] developed a hybrid recommender system, combining collaborative and content-based filtering for college recommendations. M.M. Reddy et al. [9] recommended movies based on above-average ratings by using collaborative filtering and the Pearson correlation coefficient. Last but not least, ChunYe Chien et al. [11] stressed efficient feature extraction and semantic understanding for better recommendations, especially in social media and movie-related situations, while Sushmita Roy et al. [10] addressed information overload using a variety of filtering techniques. Player statistics, and match conditions in accuratelypredicting the outcome of matches.

III. PROPOSED METHODOLOGY

The goal of this research work is to use a variety of machine learning techniques to create a better and optimized movie recommendation system. In this proposed methodology we are done Data preprocessing, collaborative filtering (CF), content-based filtering (CBF), and a hybrid recommendation strategy. Cleaning and converting ratings datasets and movie information are part of the data pre treatment step. Content-based filtering makes recommendations for related films based on user tastes by examining genres and movie descriptions. User-item interactions are used in collaborative filtering to forecast ratings and generate customized suggestions The hybrid recommendation system combines the best elements of collaborative filtering and content-based filtering to increase suggestion accuracy and address the cold start issue. Evaluation criteria including recall, accuracy, and RMSE will be employed to evaluate each approach's performance. By investigating the efficacy of hybrid models in offering customers more precise and customized movie recommendations, this research work advances the field of recommendation systems.

A. Datasets

The first step to comprehend the data is to select the appropriate participants for our research. Once we have selected them carefully, we collect the data and perform an exploratory analysis to comprehend good insights. We also check for any inconsistencies or errors in the data that we have collected. We select the MovieLens-100k dataset for movie recommendation to our systems. This dataset consists of five individual.csv files, which are: 1.Credits, 2.Keywords, 3.Links_small, 4.Ratings_small, 5.Movies_metadata

Credits dataframe

In [3]:
```
credits.head()
#credits.iloc[0:3]
#credits['cast'].iloc[0:3]
#credits.iloc[:,0:2]
```

Out[3]:

	cast	crew	id
0	[{'cast_id': 14, 'character': 'Woody (voice)',...	[{'credit_id': '52fe4284c3a36847f8024f49', 'de...	862
1	[{'cast_id': 1, 'character': 'Alan Parrish', '...	[{'credit_id': '52fe44bfc3a36847f80a7cd1', 'de...	8844
2	[{'cast_id': 2, 'character': 'Max Goldman', 'c...	[{'credit_id': '52fe466a9251416c75077a89', 'de...	15602
3	[{'cast_id': 1, 'character': "Savannah 'Vannah...	[{'credit_id': '52fe44779251416c91011acb', 'de...	31357
4	[{'cast_id': 1, 'character': 'George Banks', '...	[{'credit_id': '52fe44959251416c75039ed7', 'de...	11862

Fig 1: credits.csv dataset.

Keywords dataframe

In [63]:
```
keywords.head()
```

Out[63]:

	id	keywords
0	862	[{'id': 931, 'name': 'jealousy'}, {'id': 4290,...
1	8844	[{'id': 10090, 'name': 'board game'}, {'id': 1...
2	15602	[{'id': 1495, 'name': 'fishing'}, {'id': 12392...
3	31357	[{'id': 818, 'name': 'based on novel'}, {'id':...
4	11862	[{'id': 1009, 'name': 'baby'}, {'id': 1599, 'n...

Fig 2: keywords.csv dataset

credits.csv: This dataset typically includes information of the cast and crew of each film. It may include columns for the movie ID, the actors' names and their roles, and maybe even the crew people like directors, writers, and producers—off-stage. From this dataset, you can create features that show how actors and directors influence a film's box office draw or popularity, which can be very important to make recommendations. keywords.csv: This contains data for the keywords or tags of a movie. Such keywords can be utilized to capture the theme, characters, settings, and plot of the movie. You can search for the keywords to get movies with similar concepts or ideas, and subsequently, you can suggest those movies to the viewers who are looking for those similar things.

Link dataframe

In [11]:

```
links_small.head()
```

Out[11]:

	movieId	imdbId	tmdbId
0	1	114709	862.0
1	2	113497	8844.0
2	3	113228	15602.0
3	4	114885	31357.0
4	5	113041	11862.0

Fig 3: links _small.csv dataset

links_small.csv: This is probably a subset of some larger dataset of links. It could contain links to external databases such as IMDb and TMDb for all the movies in your dataset. This can be very helpful to include additional data from these external sources in your dataset or give people direct access to additional movie-related data. ratings_small.csv: This is a file containing user ratings of films. The ratings typically include the user ID, film ID, rating, and optionally a date that indicates when the rating was submitted. User ratings are critical in the creation of recommendation systems because they assist in collaborative filtering. This approach recommends films to users based on the way they utilize the ratings of other users with similar preferences. movies_metadata.csv: In most cases, the metadata file is the main dataset that has a tremendous amount of data about each film. It could have titles, release years, genres, budget, revenues, languages, countries of production, runtimes, etc. You can use this dataset to design numerous different features for your recommender system, such as by genre, popularity, and budget. It also offers in-depth movie analysis based on their metadata. See figure 4 below. This dataset contains a massive amount of movie data with numerous movies of different years and genres. Some of the primary information such as the title of the movie, original language, internal and external identifier numbers (such as IMDb IDs), and a full synopsis of the story are present in each component of the dataset.

```
md.iloc[0:3].transpose()
```

	0	1	2
adult	False	False	False
belongs_to_collection	{'id': 10194, 'name': 'Toy Story Collection', ...	NaN	{'id': 119050, 'name': 'Grumpy Old Men Collect...
budget	30000000	65000000	0
genres	[{'id': 16, 'name': 'Animation'}, {'id': 35, '...	[{'id': 12, 'name': 'Adventure'}, {'id': 14, '...	[{'id': 10749, 'name': 'Romance'}, {'id': 35, ...
homepage	http://toystory.disney.com/toy-story	NaN	NaN
id	862	8844	15602
imdb_id	tt0114709	tt0113497	tt0113228
original_language	en	en	en
original_title	Toy Story	Jumanji	Grumpier Old Men
overview	Led by Woody, Andy's toys live happily in his ...	When siblings Judy and Peter discover an encha...	A family wedding reignites the ancient feud be...
popularity	21.9469	17.0155	11.7129

Fig 4: Metadata dataset

The information also categorizes movies according to their genre, i.e., Drama, Action, Horror, Thriller, etc. This provides a clear pathway of realizing the themes that are dominant in a particular movie. The information marks a film as made for adult audiences, which acts to filter out material by what is suitable for audiences. This categorization is important in order to tailor recommendations to fit different audience tastes and sensitivities. The production companies and countries of the film listed also give information on where the film is from and the collaborations that helped produce the film. The dataset's careful focus on movie genres—which are represented by a structured list of names and identifiers—is one of its characteristic features. A more specific user experience is enabled by this categorization, which facilitates it to conduct in-depth analysis and offer recommendations based on theme interests. When films are tagged with genres such as "Family," "Animation," "Romance," and "Comedy," for instance, the recommendation system could identify and suggest movies suitable for the interests and viewing habits of the user. The library laboriously categorizes movies into numerous genres, serving as an approximation of the vast variety of movie experiences available to audiences. Some of the genres include Documentary, Science Fiction, Fantasy, Horror, Action, Drama, Family, Animation, Romance, Comedy, Thriller, and TV Movie. The recommendation engine can effectively traverse the vast pool of movie products with this categorization, ensuring that users will be shown with suggestions that are very close to their own likes and preferences. The data set presents a comprehensive view of the movie industry, from the relaxing stories found in Family and Animation movies to the heart-racing action and suspense of Thrillers and Horror movies. The genre classification of the dataset is a vital tool for producing a personalized viewing experience, independent of the user's preference for the real life stories narrated in Documentaries, the fanciful realms of Fantasy and Science Fiction, or the contemplative storylines of Drama.

B. CONTENT-BASED FILTERING

This recommendation system that is based on content makes recommendations to users based on both the item's description and the user preference profile. Taking into account the user's past behavior or direct feedback, such a system employs item attributes in this instance, film descriptions and taglines to recommend more items that are similar to what they like. Merging Descriptions and Taglines: In generating a comprehensive description for every movie, the system first fills in gaps in the taglines. Second, it merges the movie descriptions with their respective taglines. The merged description facilitates the comprehension and comparison of the movie content. TF-IDF Vectorization: To transform the textual description into a numerical form (TF-IDF matrix), it uses a TF-IDF (Term Frequency-Inverse Document Frequency) Vectorizer. The text is converted into a matrix by this procedure, with each row denoting a movie and each column representing a distinct word or phrase (n-gram), weighted according to its importance throughout all movie descriptions. Word analysis, taking into account unigrams and bigrams, excluding frequently occurring stop words in English, and including terms that occur in at least one document are all configured features of the TF-IDF Vectorizer. Cosine Similarity for Recommendations: The system computes the cosine similarity between movies after the TF-IDF matrix is available. Cosine similarity is a measure of the cosine of the angle between two non-zero vectors in a multi-dimensional space, in this instance the TF-IDF vectors that represent movie descriptions. When two movies have a cosine similarity score of 1, it indicates that their descriptions are precisely the same; when it is zero, it indicates that they are entirely different. Since the cosine similarity scores are obtained by taking the dot product of the TF-IDF vectors, the system employs the linear_kernel function from scikit-learn, which is more effective than computing cosine similarities directly Making Recommendations: To make movie recommendations, the system uses the input title to determine the movie index. It then compares each movie to all others to determine its cosine similarity scores. Next, it sorts the movies in descending order to identify which ones are the most similar, and finally, it returns the titles of the top 30 most similar films.

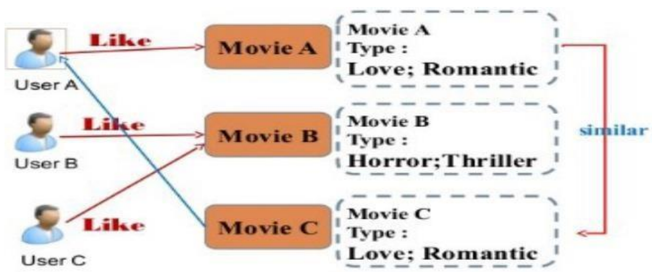

Fig 5: Content Based Recommendation

C. COLLABORATIVE-BASED FILTERING

Without needing to know anything about the item itself, Collaborative-Based Filtering (CBF) is a way to propose products based on previous interactions between users and the item. The main tenet of this strategy is that consumers who have previously expressed agreement will continue to do so in the future regarding their preferences. Recommendation systems employ a technique known as Collaborative Filtering (CF) to forecast that a user will enjoy by gathering preference or taste data from numerous users. The basis of the CF technique is that if two users have the same opinion on a particular matter, there is a greater chance that user A will concur with user B on another matter than with a randomly chosen user. Our Collaborative Filtering method does not require starting from scratch to build the algorithm. Rather, we leverage the Surprise library, a domain-specific toolset that is particularly focused on the building and analysis of recommender systems[4]. The Surprise library is renowned for having sophisticated algorithms like Singular Value Decomposition (SVD) in it. These algorithms are good at reducing prediction errors, which can be measured by metrics like Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). This technique is based on the manner in which what the community enjoys can drive individual users to content that they will enjoy. It takes into account the common tastes and interests of the users. **The Singular Value Decomposition (SVD) algorithm:** An algorithm called the Singular Value Decompositionn (SVD) is employed in recommendation systems, such as those using the MovieLens-100k dataset [movies.csv, ratings.csv]. The initial user- item rating matrix is broken down into three smaller matrices using SVD [18, 19, 20, 21, 22, 23, 24]. These three matrices are employed to predict missing ratings in the matrix by capturing the latent attributes of users and items [4] (e.g., tastes or qualities not directly quantified). SVD is used to identify underlying patterns in user ratings information. This allows the system to forecast how a user would rate products they have not used before. This technique is highly efficient in Collaborative-Based Filtering. In this method, item recommendations are acquired through similarities and user behavior.

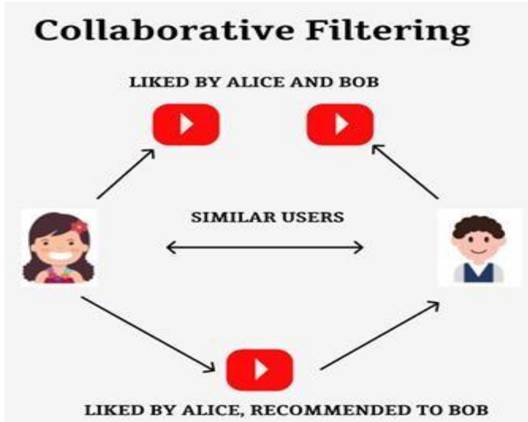

Fig 6: Collaborative Based Recommendation

D. HYBRID RECOMMENDATION SYSTEM

The drawbacks of using Collaborative-Based Filtering (CF) and Content-Based Filtering (CBF) separately are addressed by a Hybrid Recommendation System, which combines their respective advantages. In order to deliver more precise and tailored recommendations, our integrated approach makes advantage of both item properties and user-item interactions. To improve its recommendations, a Hybrid Recommendation System would use information from the movies.csv and ratings.csv files in the framework of our research work, which uses the MovieLens-100k dataset. Data Preparation and Integration: For content-based filtering, the system prepares and integrates data from several sources, such as user-movie interaction data and movie metadata (for collaborative filtering). In order to maintain consistency and make content and collaborative data merging easier, movie IDs are mapped across datasets. Content-Based Filtering Component: This component uses taglines and movie descriptions to determine how similar two movies are to one other based on their content. This is accomplished by calculating the cosine similarity scores between movies and creating a TF-IDF matrix. Collaborative-Based Filtering Component: Singular Value Decomposition (SVD) from the Surprise library is used in the Collaborative Filtering Component to forecast user ratings for movies based on similar users' rating patterns. To do this, user-rated movie data is used to train the SVD model, and measures such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were used to assess the model's performance. Hybrid Recommendation Generation: A movie title and a user ID are entered into the system. It initially searches for similar movies on the basis of content scores. It then applies a collaborative filtering model to estimate the rating the user would give to each similar movie. The system is able to make personalized recommendations using this two-step approach by ranking similar movies on the basis of how much similarity the content of these movies shares with the entered movie and the probability of the viewer to enjoy it.

A Workflow for a Hybrid Movie Recommendation System: The process of a hybrid movie recommendation system show in below figure 7. That deftly blends user input, content-based filtering, collaborative filtering, and generates personalized movie choices is illustrated by the flow chart in the image below. The user initiates the procedure by entering their unique ID and a desired movie title. This input is essential since it establishes the parameters for the process of recommendations that follows. In finding similarities, the algorithm compares the input movie's description and tagline to other movies in the database with content-based filtering. At the same time, the collaborative filtering module considers the possible ratings the user would give to movies he has not watched, using prior rating history and methods like Singular Value Decomposition (SVD) to make educated guesses about user preferences from usage patterns.

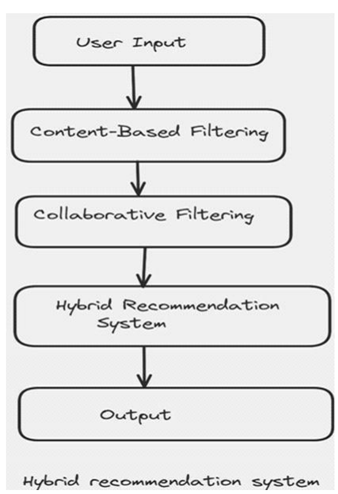

Fig 7 : A Workflow for a Hybrid Movie Recommendation System

E. Weighted rating

A sophisticated technique called weighted rating was developed to yield a more accurate determination of the rating of an item by taking into account both the absolute count of ratings that each item has received and the average of individual ratings. Weighted rating becomes essential in optimizing the recommendation system in our research work based on the MovieLens-100k dataset, especially based on the ratings.csv data. Weighted rating assists in giving movies that are more highly rated by taking into account that the movies with a higher count of ratings get their rightful place using a process that compares each film's average rating with the total count of ratings that are provided to all movies. Utilizing this method gives the guarantee that the recommendations actually represent both the quality of movies and the reliability of ratings while eliminating any skew that may arise based on the scarcity of films with high ratings. The final result is an enhanced and a more stable list of recommended films.

Weighted Rating (WR)= ((v/(v + m)). R) +((m/v+m) .C)

Where :

v is the movie's total number of votes.

The minimum number of votes (m) needed to appear in the chart.

R is the film's average rating.

The average vote for the whole report is C.

IV. Result Analysis

The integration of collaborative and content-based filtering in the hybrid recommendation system probably led to a notable increase in recommendation accuracy.

Fig 8: Hybrid Function

ratings and movies.csv are both used. CSV files would have made it possible for the system to produce individualized movie recommendations by efficiently utilizing user ratings and movie data. To ensure that customers receive recommendations that closely match their interests, cosine similarity, which compares things (movies) and individuals based on their features or ratings, should have been integrated into the recommendation process. Integration of collaborative and content-based filtering in the hybrid recommendation system probably led to a notable increase in recommendation accuracy. ratings and movies.csv are both used. CSV files would have made it possible for the system to produce individualized movie recommendations by efficiently utilizing user ratings and movie data. To ensure that customers receive recommendations that closely match their interests, cosine similarity, which compares things (movies) and individuals based on their features or ratings, should have been integrated into the recommendation process. The user interaction page of our hybrid movie recommendation application is seen in the interface image below. The suggestion process is started by users typing the title of a movie into the designated field and selecting the "Recommend" option. By combining content-based and collaborative filtering techniques, the underlying hybrid recommendation system is triggered by this action, which results in the generation and presentation of a customized list of movie recommendations based on the user's preferences as well as the thematic and content features of the input movie. The following screenshot represents the results page of our app (figure 9), which displays a list of suggested films based on users' search keyword. The strength of the hybrid recommendation system in synthesizing and analyzing the past preferences of the user and the content features of the chosen movie is reflected through this dynamically composed list. The details of every suggested film that follows each suggestion, such as the title, year of release, and description, thus offers the consumers a direct notion of the attractiveness of each film. The effectiveness of the system in personalizing the recommendations to allow the user's film discovery and watching experience comes through in the ranks of the recommendations, as calculated by the level of their relevance and expected degree of user satisfaction.

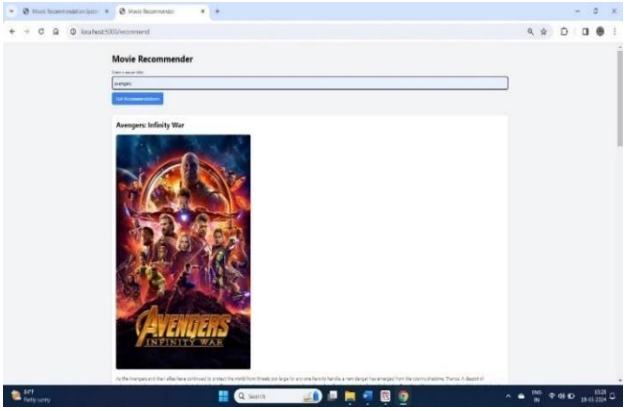

Fig 9: Input movie name taken from the user.

This recommendation process starts with the user searching for a particular movie by exploring the characteristics of the movie, such as its cast, genre, director, and user ratings. The system then applies algorithms to find and examine other movies with similar characteristics or the favorites of users with similar preferences. These algorithms can be collaborative filtering, content-based filtering, or a combination of both. To make the recommendations not only relevant but also reflect the quality and popularity of the user base in a proper manner, this process makes use of a dataset which can have parameters like number of votes and average score of votes. Thus, the user's search and interests are satisfied in a personalized list of recommended movies, thus improving their interaction and discovery process on the site.

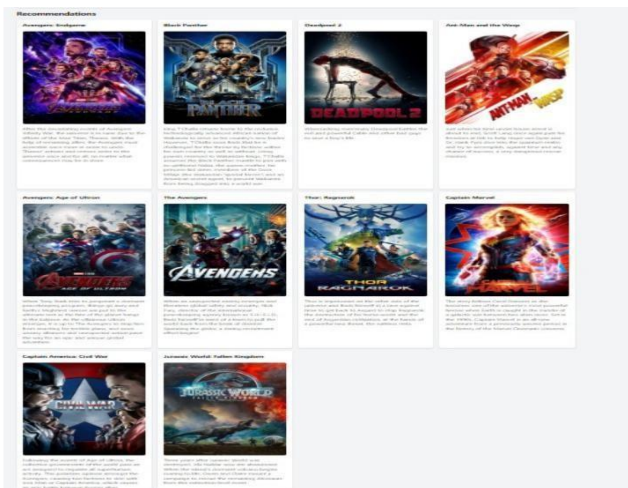

Fig 10: Screenshots for Movie Recommendation System

This recommendation process starts with the user searching for a particular movie by exploring the characteristics of themovie, such as its cast, genre, director, and user ratings. The system then applies algorithms to find and examine other movies with similar characteristics or the favorites of users with similar preferences. These algorithms can be collaborative filtering, content-based filtering, or a combination of both. To make the recommendations not only relevant but also reflect the quality and popularity of the user base in a proper manner, this process makes use of a dataset which can have parameters like number of votes and average score of votes. Thus, the user's search and interests are satisfied in a personalized list of recommended movies, thus improving their interaction and discovery process on the site.

Evaluating RMSE, MAE of algorithm SVD on 5 split(s).

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean	Std
RMSE (testset)	0.8907	0.8962	0.9023	0.8989	0.8969	0.8970	0.0038
MAE (testset)	0.6863	0.6886	0.6927	0.6926	0.6907	0.6902	0.0025
Fit time	1.24	0.90	1.05	1.09	0.95	1.05	0.12
Test time	0.14	0.10	0.40	0.10	0.17	0.18	0.11

Fig 11: RMSE scores of SVD model

We calculate the SVD method's root mean square error (RMSE) and mean absolute error (MAE) and obtain RMSE values of 0.89 and 0.68, respectively. We carefully calculate two significant prediction accuracy measures, the Root Mean Square Error (RMSE) and the Mean Absolute Error (MAE), in order to determine the success of our Singular Value Decomposition (SVD) method in the collaborative filtering component of our hybrid movie recommendation system. The RMSE score of 0.89 and MAE score of 0.68 acquired are critical points of reference for measuring the accuracy with which the system is able to predict user movie ratings. The RMSE metric gives feedback on the accuracy of the model by penalizing larger errors more severely. It accomplishes this by considering the square root of the average squared differences between estimated and actual ratings. With an RMSE of 0.89, it can be concluded that the average deviation between the true user ratings and the predictions generated by the SVD model is less than one rating point. This level of accuracy illustrates how well the model can detect the subtleties of user tastes and the underlying patterns in movie Ratings Likewise, without squaring the ratings, the MAE score that determines the average absolute difference between expected and observed ratings provides a straightforward means of measuring prediction errors. The model's capacity to give predictions that best approximate users' actual judgments is reflected in the resulting score of 0.68, which highlights the model's reliability in a more intuitive manner than RMSE. When taken cumulatively, these results exhibit the SVD algorithm's anticipated accuracy as well as its utility in actual world recommendation systems. The ability of SVD to minimize rating prediction errors and enhance user experience with personalized recommendations is illustrated by the relatively low values of RMSE and MAE. Through latent factor analysis, the SVD-based collaborative filtering approach effectively unravels the complex web of user-item interactions to guarantee that recommendations are both relevant and customized to each user's profile, resulting in a more enjoyable and fulfilling movie discovery process.

Accuracy table

Method	Existing Work	Proposed Work
Content-based	0.9325	0.9023
Collaborative	0.8888	0.8907
SVD	0.8735	0.6863
Hybrid Model	0.76	0.6902

When comparing our work to the findings of the base publication being referred, some noticeable disparities and correspondences show themselves. In the first instance, if our study on content-based as well as collaborative filtering mechanisms was compared with Singular Value decomposition (SVD) based method of existing reference that they obtained RSME of hybrid model 0.76, the RMSE values were marginally superior. Alternatively, our SVD approach reflected a slightly smaller RMSE value 0.68, which implies potentially higher prediction accuracy. Interestingly, despite having an extensively lower RMSE value, our resultant hybrid model was better than our individual content-based and collaborative filtering techniques as well as the SVD approach documented in our research. This would imply that our hybrid model may possess enhanced predictive ability, reflecting the viability of our collaborative strategy in mitigating the inherent limitations of individual approaches.

V. Conclusion

The combination of CBF and CF works well because it capitalizes on the specific properties of films and user behavior, respectively. Content-Based Filtering searches for similarities among films based on their unique characteristics, such as genres, synopses, and directorial styles. In contrast, Collaborative Filtering examines previous user selections to discover shared patterns and preferences among different groups of users. With the integration of different methodologies, hybrid systems can serve a broader range of user preferences, resulting in suggestions that are accurate and closely aligned with personal preferences. In addition, the use of cosine similarity within these hybrid systems is necessary in order to gauge the level of similarity between user profiles or between products (in this case, movies). The measure enhances the system's ability to match matches the user with films which suit their individual tastes by calculating the cosine of two vectors' angle in a higher-dimensional space. This enables more precise evaluation of movie attributes or user tastes. The findings of the study affirm the benefits of hybrid recommendation systems for movie recommendations. Hybrid solutions support a more advance and user-oriented recommendation experience by leveraging the strengths of both CBF and CF with the reduction of each technology's weaknesses individually. The value of sophisticated analytical methods and the utilization of large datasets in forming extremely personalized and precise suggestions are also emphasized by this work. The knowledge gained from this study is extremely significant to the area and provides a strong basis for the continuous improvement of recommendation systems. The study's ideas will play a crucial role in steering the creation of future recommendation systems that aim to fulfill the growing need for customized content consumption experiences as the digital content market develops. This innovation-driven approach will inevitably help platforms by increasing user happiness and engagement as well as users by improving their journeys of discovery and consumption of content.

REFERENCES

[1] S. Agrawal and P. Jain, “An improved approach for movie recommendation system,” 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017, pp. 336-342, doi: 10.1109/ISMAC.2017.8058367.

[2] N. Vaidya and A. R. Khachane, “Recommender systems-the need of the ecommerce ERA,” 2017 International Conference on Computing Methodologies and Communication (ICCMC), 2017, pp. 100-104, doi: 10.1109/ICCMC.2017.8282616.

[3] M. Talukder, M. M. Rahman, S. Halder and M. J.Uddin, “Novel recommendation systems in social networks,” 2017 IEEE International Conference on Imaging, Vision and Pattern Recognition (icIVPR), 2017, pp. 1-6, doi: 10.1109/ICIVPR.2017.7890864.

[4] Y. Xiong and H. Li, “Collaborative Filtering Algorithm in Pictures Recommendation Based on SVD,” 2018 International Conference on Robots and Intelligent System (ICRIS), 2018, pp. 262-265, doi: 10.1109/ICRIS.2018.00074.

[5] S. Bhat and K. Aishwarya, “Item-based Hybrid Recommender System for newly marketed pharmaceutical drugs,” 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2013, pp. 2107-2111, doi: 10.1109/ICACCI.2013.6637506.

[6] J. Chen, K. Chao and N. Shah, “Hybrid Recommendation System for Tourism,” 2013 IEEE 10th International Conference on e-Business Engineering, 2013, pp. 156161, doi: 10.1109/ICEBE.2013.24.

[7] P. Darshna, “Music recommendation based on content and collaborative approach and reducing cold start problem,” 2018 2nd International Conference on Inventive Systems and Control (ICISC), 2018, pp. 1033-1037, doi: 10.1109/ICISC.2018.8398959

[8] V. Powar, S. Girase, D. Mukhopadhyay, A. Jadhav, S. Khude and S. Mandlik, “Analysing recommendation of colleges for students using data mining techniques,” 2017 International Conference on Advances in Computing, Communication and Control (ICAC3), 2017, pp. 1-5, doi: 10.1109/ICAC3.2017.8318781.

[9] M. M. Reddy, R. S. Kanmani and B. Surendiran, “Analysis of Movie Recommendation Systems; with and without considering the low rated movies,” 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020, pp. 1-4, doi: 10.1109/icETITE47903.2020.453.

[10] S. Roy, M. Sharma and S. K. Singh, “Movie Recommendation System Using Semi-Supervised Learning,” 2019 Global Conference for Advancement in Technology (GCAT), 2019, pp. 1-5, doi: 10.1109/GCAT47503.2019.8978353.

[11] C. Chien, G. Qiu and W. Lu, “A Movie Trailer Recommendation System Based on Pre-trained Vector of Relationship and Scenario Content Discovered from Plot Summaries and Social Media,” 2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI), 2019, pp. 1-6, doi: 10.1109/TAAI48200.2019.8959918.

[12] J. Zhang, Y. Wang, Z. Yuan and Q. Jin, “Personalized real time movie recommendation system: Practical prototype and evaluation,” in Tsinghua Science and Technology, vol. 25, no. 2, pp. 180-191, April 2020, doi: 10.26599/TST.2018.9010118.

[13] J. E. Purnomo and S. N. Endah, “Rating Prediction on Movie Recommendation System: Collaborative Filtering Algorithm (CFA) vs. Dissymetrical Percentage Collaborative Filtering Algorithm (DSPCFA),” 2019 3rd International Conference on Informatics and Computational Sciences (ICICoS), 2019, pp. 1-6, doi:10.1109/ICICoS48119.2019.8982385.

[14] N. Ifada, T. F. Rahman and M. K. Sophan, “Comparing Collaborative Filtering and Hybrid based Approaches for Movie Recommendation,” 2020 6th Information Technology International Seminar (ITIS), 2020, pp. 219-223, doi: 10.1109/ITIS50118.2020.9321014.

[15] M. Kim, S. Jeon, H. Shin, W. Choi, H. Chung and Y. Nah, “Movie Recommendation based on User Similarity of Consumption Pattern Change,” 2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), 2019, pp. 317-319, doi: 10.1109/AIKE.2019.00064.

[16] H. Wang, N. Lou and Z. Chao, “A Personalized Movie Recommendation System based on LSTM-CNN,” 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2020, pp. 485-490, doi:10.1109/MLBDBI51377.2020.00102.

[17] S. Salmani and Sarvesh Kulkarni, “Hybrid Movie Recommendation System Using Machine Learning” 2021 International Conference on Communication information and Computing Technology (ICCICT),2021, doi: 10.1109/ICCICT50803.2021.9510058

[18] M.Sireesha, S. N. TirumalaRao, Srikanth Vemuru, Optimized Feature Extraction and Hybrid Classification Model for Heart Disease and Breast Cancer Prediction International Journal of Recent Technology and Engineering Vol - 7, No 6, Mar - 2019 ISSN - 2277-3878, Pages – 1754 – 1772

[19] M.Sireesha, S. N. TirumalaRao, Srikanth Vemuru, Frequent Itemset Mining Algorithms: A Survey Journal of Theoretical and Applied Information Technology Vol - 96, No .3, Feb - 2018 ISSN - 1992-8645, Pages – 744 – 755

[20] M. Sireesha, Srikanth Vemuru and S. N. TirumalaRao, ”Coalesce based binary table: an enhanced algorithm for mining frequent patterns", International Journal of Engineering and Technology, vol. 7, no. 1.5, pp. 51-55, 2018.

[21] Sireesha Moturi, S.N.Tirumala Rao, Srikanth Vemuru,Grey wolf assisted dragonfly-based weighted rule generation for predicting heart disease and breast cancer, Computerized Medical Imaging and Graphics, Volume 91, 2021, 101936, ISSN 0895-6111, https://doi.org/10.1016/j.compmedimag.2021.101936.

[22] Sireesha Moturi , Srikanth Vemuru, S. N. Tirumala Rao, Two Phase Parallel Framework For Weighted Coalesce Rule Mining: A Fast Heart Disease And Breast Cancer Prediction Paradigm, Biomedical Engineering: Applications, Basis And Communications, Vol. 34, No. 03 (2022), https://doi.org/10.4015/S1016237222500107

[23] Moturi, S., Vemuru, S., Tirumala Rao, S.N., Mallipeddi, S.A. (2023). Hybrid Binary Dragonfly Algorithm with Grey Wolf Optimization for Feature Selection. In: Hassanien, A.E., Castillo, O., Anand, S., Jaiswal, A. (eds) International Conference on Innovative Computing and Communications. ICICC 2023. Lecture Notes in Networks and Systems, vol 703. Springer, Singapore. https://doi.org/10.1007/978-981-99-3315-0_47

[24] Sunayna, S.S., Rao, S.N.T., Sireesha, M. (2022). Performance Evaluation of Machine Learning Algorithms to Predict Breast Cancer. In: Nayak, J., Behera, H., Naik, B., Vimal, S., Pelusi, D. (eds) Computational Intelligence in Data Mining. Smart Innovation, Systems and Technologies, vol 281. Springer, Singapore. https://doi.org/10.1007/978-981-16-9447-9_25