

Effective Cross Synthesized Methodology for Movie Recommendation with Emotion Analysis through Ranking Score

R Lavanya¹, Dr. B Bharathi²

Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai, India^{1,2}
Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Chennai, India¹

Abstract—Providing accurate movie recommendations to a user with limited computing capability is a challenging task. A hybrid system offers a good trade-off between the accuracy and computations needed for such recommendations. Collaborative Filtering and Content-Based Filtering are two of the most widely employed methods of computing such recommendations. In this work, a high-efficient hybrid recommendation algorithm is proposed, which deeds users' contour attributes to screen them into various groups and recommends movie to a user based on rating given by other similar users. Compared to traditional clustering-based CF recommendation schemes, our technique can effectively decrease the time complexity, whereas attaining remarkable recommendation output. This approach mitigates the shortcomings of the individual methods, while maintaining the advantages. This allows the system to be highly reactive to new viewer inputs without sacrificing on the quality of the recommendations themselves. Building on other hybrids of a similar kind, our proposed system aims to reduce the complexity and features needed for calculation while maintaining good accuracy and further enhanced by utilizing Sentiment Analysis to rank the movies and take user reviews into consideration, which traditional hybrids do not take into account. Then analysis was performed on the data set and the results show that the proposed recommendation system outperforms other traditional approaches.

Keywords—Recommendation systems; collaborative filtering; styling; content based filtering; implicit feedback; hybrid recommendation; sentiment analysis; singular value decomposition

I. INTRODUCTION

Watching movies is one of the most popular forms of media entertainment. Viewers are incredibly engrossed and invested in the culture of motion films. Recent advancements in technology have enabled the widespread streaming of movies on demand [1]. This, in turn, has added to the popularity and ease of access to movies. There are thousands of movies for one to choose from. These movies are not only segregated by their genre, cast, production teams, direction, and numerous other factors. This makes it particularly difficult to pick a single movie to watch first. Everyone's preference of movies is also subjective and one may not enjoy a movie that another person loves. This creates an ambiguity as it is complex to determine what features the most impactful when are looking for a new movie to watch [2]. Movie recommender systems help combat this by providing recommendations based

on what the user may have already watched by suggesting similar movies. It attempts to figure out what movie a viewer is likely to rate among the highest.

Recommending a movie is not a straightforward task. This is particularly challenging also as the preferences of viewers may be very different. This leads to there being a distribution of niches that are not uniform to be immediately apparent [3]. Consequently, a movie that is not conventionally popular may be preferable to some viewer simply based on their subjective view towards movies in general. This can be tackled effectively by taking into account a large variety of movies and a large amount of them so as to encompass the likes and dislikes of users of all categories [4]. A robust recommender must be able to recommend movies that are more relevant to the user themselves as shown in Fig. 1.

Many techniques have been used to make recommenders effective in this regard and perform well with large data [5]. Recommender systems are a set of algorithms aimed at emulating information processing systems where the end goal is to suggest relevant items to users, items being movies that users watch. The various classification of recommender system is given in Fig. 2. Content Based methods also offer a way to deal with the issue of limited rating data [6]. Content Based methods work by taking into consideration the similarity between the movies themselves. This similarity could be between various aspects of the movies such as the genre, cast and other related data [7].

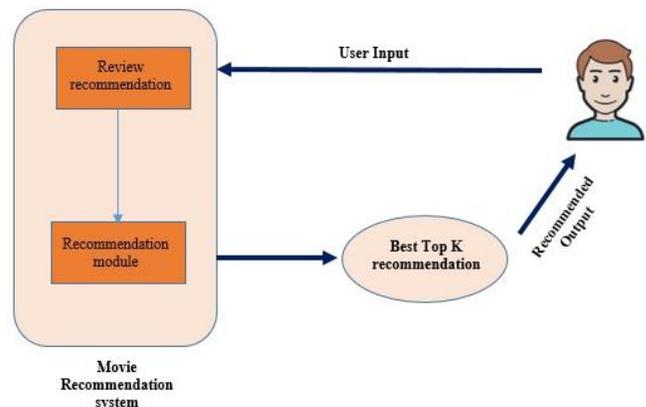


Fig. 1. Schematic of Recommendation System.

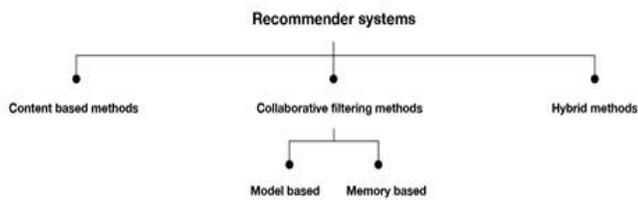


Fig. 2. General Classification of Recommendation System.

A problem that arises with the content-based system is an uncertainty if the system can take the user's behavior towards one genre or source and apply that preference shown to all the other content present in the system. If the engine simply keeps suggesting content similar to what the user is already watching, its value goes down quite a bit in some sectors of recommendation. The Collaborative Filtering approach recommends items by creating a profile for users [8]. The similarity between the profiles is based on the movies rated. If the user rates certain movies high then the algorithm would look for other users who have also rated the same movie highly or in the range of the rating by the user to whom it is to give recommendations. It then predicts what rating the user would give to a movie that they have not seen based on how they have rated the movies they have already seen. Therefore, Collaborative Filtering works well where there is history available for the user ratings on other movies [9]. This however is also a drawback as new users do not have many ratings and this poses a problem for Collaborative Filtering [10]. Based on the advantages, a hybrid system seems to be the most promising approach to mitigate the drawbacks of these common systems and to bring forth their advantages [11]. There are many ways in which Hybrid models can be used: by extrapolating separate content-based recommendation list and collaborative-based recommendation list of predictions and unifying them as one single list; using collaborative-based methods as a primary approach and enhancing them by adding content-based capabilities; using content-based methods as a primary approach and enhancing them by adding collaborative-based capabilities; combining the capabilities and features of both models and creating one single model. This paper is structured as follows: Section 1 emphasis the introduction. Section 2 emphasis the related study and the narrative of the objective. The procedures and resources are discoursed in Section 3 and their exploration and clarifications are exposed in Section 4. The conclusion was clarified in Section 5.

The rest of the paper flows as literature survey in Section II, Section III explain the methodology and implementation process of recommendation system, Section IV describes the comparison and analysis of various recommendation techniques, and Sections V and VI briefs the conclusion and future work.

II. RELATED WORK

Collaborative Filtering algorithm generates a user profile based on the ratings the viewer has given to other movies. If two users have rated similar kind of movies highly, this means that there is a good chance they may prefer similar movies. If the other viewer has already rated other movies, it allows the system to predict if the target viewer will enjoy it or not. This

is the basic premise of how a Collaborative Filtering system would recommend a movie [12]. Collaborative Filtering has been widely employed in many ways, and a lot of academic work has revolved around combining other techniques to boost the performance of such a method. One of the main issues when it comes to Collaborative Filtering is that the computations needed are heavy. Thus in most cases, it has trouble when trying to scale the data up [5]. The method uses Self-Organizing Map Neural Networks [13] to carry out Collaborative Filtering. This method offers a good alternative as the Self Organizing Map Neural Network is not very computationally heavy. While this method worked well for the data, the data itself only comprised of a few dozen movies and about a hundred and seventy users. This makes it hard to judge if the effectiveness of this technique will remain high when faced with a larger dataset, or one with more features.

This multilayer perceptron neural network system works by utilizing the reduction in error in prediction by subjecting the training data to go through multiple passes of a neural network [14]. This method is promising as it does not need to have a deep network for classification. This is particularly effective for polar sentiment data, which will be the focus of the proposed system. This is because for binary classification, a very deep neural network may in fact introduce overfitting to the data. Overfitting occurs when the data trains too well for the test set. This may result in the model performing really well for the trained data but not for the actual target or testing data itself. A shallow network also allows for faster inference time. A personalized Recommendation approach [15] grounded on Three Social Influences, Personal interest means user-item relationship and interpersonal influence and interpersonal interest similarity means user-user relationship of social networks. Probabilistic matrix Factorization makes experiments on the datasets, namely, MovieLens and yelp [16]. Tactically this removes the tricky cold start and data sparsity.

A recommendation system for real estate websites [17] is that it helps consumers in acquiring new properties or homes. Recommendation system is proven by merging case based reasoning (CBR) and Ontology. Former systems supports single characteristic exploration systems but this system support multivalued search system. Sentiment Analysis can also be used along with Collaborative Filtering for better and more inclusive results. The system [18] was trained on data where all the users had given a large number of ratings. This brings into question how well the system would perform where the ratings are limited. Also, the similarities between other features of the movies, such as genre, were not considered in the study.

The author in [19] used a Diverse Collaborative Prediction to combine Collaborative Filtering with Content-Based filtering. This system gave better results than just the individual techniques did, however this method does not consider reviews either. The author in [20] employed an Item-Based Collaborative Filtering model with a Content Based one. The predictions are reached by the TF-IDF method with the nearest neighbor predictions. The MovieLens and Film Trust datasets were used for training and testing this system. The author in [21] used a cosine similarity matrix that showed better

accuracy than that of the other systems, but it also did not take written reviews into consideration.

Another study [22] showed that the Singular Value Decomposition worked well for recommendations. These studies support that the TF-IDF and SVD methods take a lot of heavy computation, but they give some of the most accurate results for recommendations. The author in [23] focused on the hybrid recommendation model which encouraged the user's social data, reviews, and ratings available.

This model of recommendation consists of six processes, review transformation, the feature generation, community prediction, model training, feature blending, and prediction and the last one, evaluation criteria for ontology based recommendations. This [24,27] mainly focuses on a system which helps to provide details analysis about the items which are arranged by the wishes of the similar users. The recommendation system with the proper recommendation for this research will be used in suggesting the item selection system by making a recommendation system with the help of an item-based collaborative filtering methodology. Based on the literature, the associated research challenges are observed.

- Data sparsity may happen due to user/rating matrix is sparse and it is hard to find the users who have rated the same item.
- The existing recommendation technique requires enormous processing time and mostly user is prohibited in getting accurate recommendations that are similar to their profile.

Subsequently it is fortified about the essential for the proposed research to enhance the movie recommendation process competently. The associated objectives are proposed in this research work so as to address few issues in recommendation technique. The contribution of the research comprises.

- To provide enhanced movie recommendation system for the users through an improved hybrid recommendation algorithm combination of user-based CF (UBCF) and item-based CF(BCF) in the context of SVD dimension reduction to improve the speed and quality of recommendation.
- To providing content related to the collection of relevant and irrelevant items for users of online service providers and to recommend movies to users based on user / item base movie ratings.

To enhance the recommendation accuracy in hybrid recommendation system through optimized sentiment analysis for providing more diverse recommendations by satisfying the requirements recommendation features.

III. METHODOLOGY

A. User-Based Collaborative Filtering

User-Based Collaborative Filtering is a technique for predicting which products a user would enjoy based on the ratings provided to that item by other users who share the target user's tastes [1]. Collaborative filtering is used by many

websites to develop their recommendation systems. Steps for Collaborative Filtering with Users: Step 1: Identifying users who are similar to the target user U. The algorithm may be used to calculate similarity between any two users 'a' and 'b'. Step 2: Estimate of an item's missing rating done as follows: Now, the target user may be quite similar to certain people while being very different from others [13, 26]. The proposed system employs a combination of Collaborative Filtering and Content Based Recommendations, further enhanced using Sentiment Analysis to rank the movies.

The movielens dataset is hired in our research paper and collected from the GroupLens [25,28], which contains 20 million ratings for around 27000 different movie titles and has a user ID, movie ID, rating, and timestamp. The Characterization of the movie's content information includes over 54058 records and includes movie ID, title, genre, director, actor, and more. The graph in Fig. 3 represents the relation between categories and the movies rated accordingly.

The data contains a huge amount of reviews. This helps retain most of the movies while reducing the number of users by about a third and represented using a seaborn graph as given in Fig. 4. This can be important in the order that we are able to see the link between a movie's specific rating and therefore how much the movie got. Therefore, we must set a threshold for a minimum number of ratings while constructing a system that recommends. So, to create this new column we use the utility of pandas' groupby. We groupby the title columns, so use the calculation method to calculate the number of ratings each movie received as shown in Fig. 5. The tags for all the movies are combined with the genre to generate a larger metadata for the movies as shown in Fig. 6. This metadata can be used to perform a Content Based approach. The goal is to keep the number of features as low as possible without compromising on the accuracy of the results.

An added benefit of keeping the features lower is that it is less complex when it comes to calculation. A lighter model will help improve the inference time.

We create the value of movie data 'rating' using movie title and calculate rating count in 'title' by applying threshold and get the result.

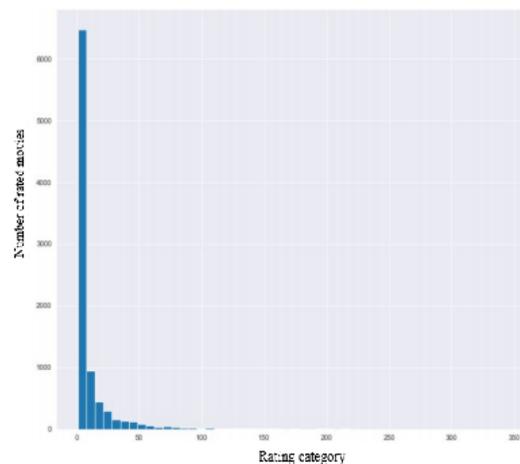


Fig. 3. Number of Rated Movies vs Number of Rated Category.

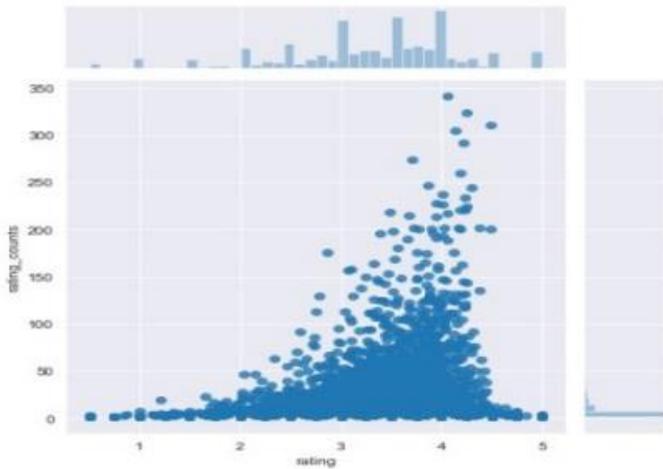


Fig. 4. Graph Representing Retained Movies vs Reduced users.

userid	movieid	rating	timestamp
0	1	1	4.0 964982703
1	1	3	4.0 964981247
2	1	6	4.0 964982224
3	1	47	5.0 964983815
4	1	50	5.0 964982931
...
100831	610	166534	4.0 1493848402
100832	610	168248	5.0 1493850091
100833	610	168250	5.0 1494273047
100834	610	168252	5.0 1493846352
100835	610	170875	3.0 1493846415
...
100836

100836 rows x 4 columns

Fig. 5. Movie Data with MovieId and Ratings.

Index	movieid	Title	Genres
0	1	Toy Story(1995)	Adventure Animation Children Comedy Fantasy
1	2	Jumanji (1995)	Adventure Children Fantasy
2	3	Grumpier Old Men (1995)	Comedy Romance
3	4	Waiting to Exhale (1995)	Comedy Drama Romance
4	5	Father of the Bride Part II (1995)	Comedy
...
9737	193581	Black Butler: Book of the Atlantic (2017)	Action Animation Comedy Fantasy
9738	193583	No Game No Life: Zero (2017)	Animation Comedy Fantasy
9739	193585	Flint (2017)	Drama
9740	193587	Bungo Stray Dogs: Dead Apple (2018)	Action Animation
9741	193609	Andrew Dice Clay: Dice Rules (1991)	Comedy

Fig. 6. Movies with Genres and Tags.

B. Filtering based on Content

The data is sampled to take a large chunk to make a training set on which the SVD loss will be trained. The movie genres are combined with tags to create the metadata of the movies. This metadata will be used to generate a Content Based Recommendation model. A segment of the data is segregated where it contains the user ID, the movie ID and the rating that the user gave to the movie as shown in Fig. 7.

Title	Rating	Rating count
Toy Story(1995)	1.750000	3
Jumanji (1995)	3.84333	2
Grumpier Old Men (1995)	2.00000	2
Waiting to Exhale (1995)	0.60000	1
Father of the Bride Part II (1995)	2.32000	16
Black Butler: Book of the Atlantic (2017)	3.50000	1
No Game No Life: Zero (2017)	5.00000	3
Flint (2017)	3.050632	4
Bungo Stray Dogs: Dead Apple (2018)	2.25000	24
Andrew Dice Clay: Dice Rules (1991)	3.75000	2

Fig. 7. Number of Rating of Movies after Threshold.

This data will be utilized to build the Collaborative model of the hybrid. Additionally, the movie ID and the genres as well as the tags related to the movie are segregated for building the content matrix for the hybrid system. A pass of Singular Value Decomposition is performed in order to flatten the matrix dimensions even further by introducing factorization. This also gives an idea of the variance, which indicates that the first 25 components in the ratings explain the majority of the variance.

This allows us to be even more selective for the data. SVD [22] was chosen as the preferred decomposition method as it gives reliable results and there is some flexibility on how many folds of data we can choose to train on. Furthermore, by our previous review it has been established that it is a good way to ensure high precision.

This adds up in the end when the actual recommendations are generated. Furthermore, TF-IDF is utilized to empower the hybrid recommendation module. This works well with the SVD used earlier.

IV. COMPARISON AND ANALYSIS OF VARIOUS RECOMMENDATION APPROACHES

The Hybrid Recommender System is built with two main components, the Collaborative matrix and the Content matrix. First, the matrix of movies and their ratings are transformed into a feature matrix as given in Fig. 8. This matrix contains the movies against the users and the data contained is the rating given by the user. This featurization is done by utilization of Term Frequency-Inverse Document Frequency. This creates a large number of features, but decomposition will allow these features to be lessened, ultimately bringing down the complexity of the calculations needed. When Singular Value

Decomposition (SVD) is used on this matrix it reduces the features hundredfold. Moreover, analysis after SVD revealed that most of the variance in the data comes from about the first 125 features, which limits the features even further.

userid	1	3	4	5	6	7	8	9	10	11	...	601
movieid												
1	4.0	0.0	0.0	4.0	0.0	4.5	0.0	0.0	0.0	0.0	...	4.0
2	0.0	0.0	0.0	0.0	4.0	0.0	4.0	0.0	0.0	0.0	...	0.0
3	4.0	0.0	0.0	0.0	5.0	0.0	0.0	0.0	0.0	0.0	...	0.0
4	0.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	0.0	...	0.0
5	0.0	0.0	0.0	0.0	5.0	0.0	0.0	0.0	0.0	0.0	...	0.0
6	4.0	0.0	0.0	0.0	4.0	0.0	0.0	0.0	0.0	5.0	...	0.0
7	0.0	0.0	0.0	0.0	4.0	0.0	0.0	0.0	0.0	0.0	...	0.0
8	0.0	0.0	0.0	0.0	3.0	0.0	0.0	0.0	0.0	0.0	...	0.0
9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
10	0.0	0.0	0.0	0.0	3.0	0.0	2.0	0.0	0.0	3.0	...	0.0

Fig. 8. Feature Matrix with Movies and Ratings.

The algorithm for Hybrid SVD is proposed and it is given with detailed procedure as shown in Fig. 9 for utilizing the concept of standard SVD and enhanced further to acquire the hybrid enhanced method for obtaining low computation time for recommendation procedure.

A. Comparison on Traditional Approach

Root-Mean-Square-Error (RMSE) between real ratings and predictions is a widely used measurement. The lower the RMSE, the more accurately the recommendation algorithm predicts user ratings. To get the initial phase of results on a smaller scale of dataset we decided to use Root Mean Square Error method to find out relevant recommendations as required by the user. Root Mean Square Error [26] method is frequently used method to calculate the difference between the observed measure and the predicted measure. This measurement is usually done using a mathematical formula which is as follows in (1):

where,

n = total number of values present.

p = predicted value.

o = observed value.

i = the value at given position.

So, using this formula and the data from the datasets, recommendations of movies were obtained at initial stage. Here, we used unsupervised learning to classify the data according to our needs from the dataset.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (P_i - O_i)^2}{n}} \quad (1)$$

We gave some input factors as to get the relevant recommendations.

Algorithm: The HybridSVD-based dimension reduction algorithm

Input: Matrix P

Output: Correlation matrix Q,P,T

1. P is factored into three matrices $Q_1 P_1 T_1$;
2. Reduce the $n \times n$ matrix P_1 to have only k largest diagonal values and obtain a matrix $P_i, i < n$.
3. The matrices Q_1 and T_1 are reduced accordingly, then the reconstructed matrix $S_k = QP_kT$ is the closest rank- n matrix to $S, S_k \approx S$;
4. Calculate the square root of P to be $P^{1/2}$ and calculate the two correlation matrices $QP^{1/2}, P^{1/2}T$.
5. Fill in the unrated data and smooth the matrix. Replace all missing values with the corresponding column average and subtract the corresponding row average from every matrix entry.
6. Calculate the SVD of S according to steps 3 and obtain the matrix Q,P,T.
7. Calculate the user correlation matrix $QP^{1/2}(m,k)$, denoted by U, and the item correlation matrix $P^{1/2}T(k,n)$, denoted by I.
8. For the matrix U(m,k), clustering method is used to obtain the similar rating user-modes and obtain the user-based commendation set P_u .
9. For the matrix I(k, m), a set of top-K similar items towards item i_j will be generated according clustering algorithm. Calculate the prediction for user on page u_a and obtain the item-based commendation set P_i .
10. Sort P_u and P_i by predicted rating and obtain the commendation set P.

Fig. 9. Hybrid SVD Algorithm.

In the experiment, comparing with the traditional UBCF, IBCF algorithm, we can learn that the HybridSVD algorithm can consistently get a lower RMSE and provide better quality of predictions as represented in Fig. 10. The density of a rating matrix can have a significant impact on the performance of collaborative filtering.

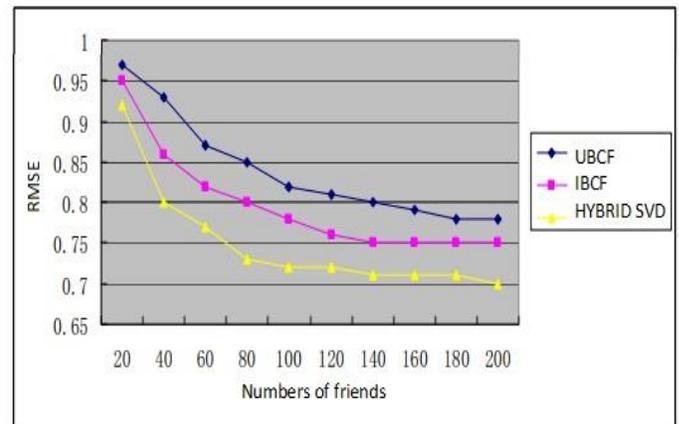


Fig. 10. Comparison of Customary and the HybridSVD Algorithm.

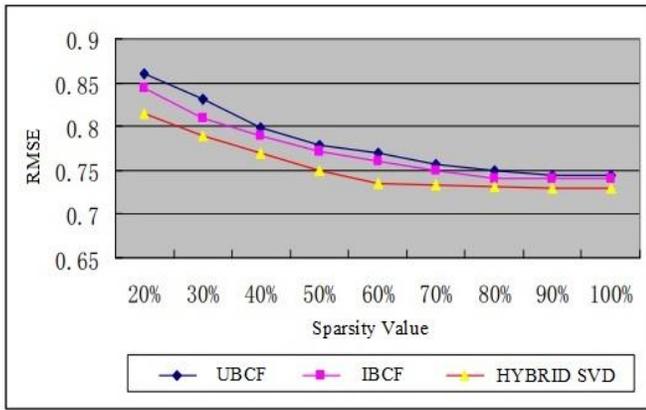


Fig. 11. Traditional vs Hybrid with Sparse Data.

In Fig. 11, we analyze how RMSE evolves with the density of rating matrix. The results indicate that the hybrid approaches consistently improves the recommendation performance regardless of sparsity of test users or items.

B. Sentiment Analysis for Ranking Calculation

The Sentiment Analysis [27] is done over the Large Movie Reviews Dataset. The reviews are categorized in a polar way, so 1 is for a positive review and 0 for a negative. The method used for prediction in the proposed system is a multilayer Perceptron model [14]. Such a model has shown to be effective in predicting sentiment. For each user, the interest movie ratings are used as dimensions to create a vector. The similarity between any two users is determined by the cosine of the angle between the vectors of those two users using the formula as given in (2).

$$\text{Similarity} = \cos \theta = \frac{A \cdot B}{||A|| \times ||B||} \quad (2)$$

For instance, interest movie ratings of two users are {3.5, 1.0, 4.0} and {2.0, 4.0, 0} respectively. The cosine of the angle between two vectors is calculated as 0.84624085163. This implies that the two users are approximately 84% similar to each other with respect to their interests.

Likewise, the calculation is performed for all users with respect to each other and a similarity matrix is generated. The comparison of different scores for movies based on different filtering approaches is given in Table I.

C. Rating Calculation

The rating calculation for the predicted system with sentiment score is calculated as given by (3). These values are linked with the movie titles and averaged according to the titles as shown in Fig. 12. This allows it to be merged with the data used for the hybrid recommendation. This however also reduces the number of movies drastically as the recommendations available are limited.

$$\text{Rating}_{\text{Sentiment}} = \frac{\sum_1^n s_{m,n}}{n} \quad (3)$$

The predictions generated are averaged to reach a general predicted number for the particular title. Finally, the recommendations from the hybrid system are used to predict the top k movies that would be the most relevant according to

the system. These recommendations also have the predicted ratings attached to the movie titles for each user. These ratings are generated by the similarity matrix between the hybrid recommender.

To reduce the time needed to calculate the final recommendations, the proposed system simply takes these movies and then calculates a final ranking for each movie. The rating from Sentiment is reached by averaging over the number of ratings (n) for the movie across the movie title (m) as given in (4). For instance, a user has selected his interest genre as humour. The similarity points of all opted movies in particular genre, say {10,9,9,8,9}, are listed. The mode is calculated as 9. So, the domain score is 9.

TABLE I. COMPARISON OF SIMILARITY SCORES

Title	CBF	CF	HYBRID SVD
Toy Story (1995)	1.000000	1.000000	1.000000
Jumanji (1995)	0.881076	0.541400	0.722641
Grumpier Old Men (1995)	0.874194	0.448877	0.713540
Waiting to Exhale (1995)	0.827519	0.253080	0.473511
Father of the Bride Part II (1995)	0.254270	0.038402	0.374106
Black Butler Book of the Atlantic (2017)	0.222232	0.437753	0.330236
No Game No Life: Zero (2017)	0.223195	0.431043	0.307581
Flint (2017)	0.224140	0.567751	0.249017
Bungo Stray Dogs: Dead Apple (2018)	0.216449	0.244214	0.378974
Andrew Dice Clay: Dice Rules (1991)	0.215749	0.326542	0.257101

movieId	Title	Sentiment
11	Toy Story(1995)	0.57344
62	Jumanji (1995)	0.376860
33	Grumpier Old Men (1995)	0.532132
44	Waiting to Exhale (1995)	0.985450
25	Father of the Bride Part II (1995)	0.890252
103	Black Butler: Book of the Atlantic (2017)	0.290543
105	No Game No Life: Zero (2017)	0.272899
53	Flint (2017)	0.108863
73	Bungo Stray Dogs: Dead Apple (2018)	0.744440
94	Andrew Dice Clay: Dice Rules (1991)	0.535082

Fig. 12. Predicted Sentiment Score based on user Tags.

The process is repeated for all the preferred interests. Using the scores of the interest domains, rather than the raw input of all users alone, can give us a better similarity and the overall precision shall be increased to a certain extent. This final ranking is reached by adding the averaged sentiment score with the predicted ranking. This allows taking into account the sentiment rating without having to compare it with a huge number of movies. In this way, the impact of the sentiment analysis is still relevant but keeps the sorting of the movies from the larger dataset largely dependent on the output from the hybrid recommendations module. For testing the accuracy of the system, the metrics of precision and recall have been used.

For testing the accuracy of the system, the metrics of precision and recall have been used. These have been used by many other works to indicate how accurate the system is. Precision is the ratio between the True Positives (TP) and the total positives predicted by the system. Recall is the ratio between the True Positives and the total TP with False Negatives (FN). So, Precision gives a measure of how accurate the actual predictions are, while recall gives an idea of how many of the predications are actually being considered. F-Measure gives a great idea of accuracy. For F-Measure to be high, both precision and recall have to be high. Precision, recall and F-Measure all have values between 0 and 1. Equation 1 gives the equation for precision, recall and F-Measure [28]. Fig. 13 shows the performance of the proposed system based on accuracy for validation sets containing 1 million review ratings. Both the Precision and Recall are above 0.7 and this causes the average F-Measure to be 0.93, which is highly competitive with other similar systems as can be seen by the study. Table II shows the performance of the proposed system on different number of ratings.

As the results show the best F-Measure comes from the lower amount of ratings. As the number of ratings increases, precision is seen to increase, while recall gets lower. This causes a lower average F-Measure.

However, the accuracy is still very high. As we can see from the measures, the hybrid system itself takes a lot of time to compute the recommendations. However, the addition of the Sentiment Analysis adds very little time to the overall merged system. So, it is still keeping the time relatively low than if the Sentiment Analysis was used with the total system instead.

We conclude from these experiments that the proposed hybrid algorithm is effective at improving the quality of recommendations and accuracy of the proposed technique improved with sentiment score added.

TABLE II. PERFORMANCE MEASURES FOR DIFFERENT RATING COUNTS

Ratings Count	F-Measure	Recall	Precision
20	0.9304	0.791	0.877
40	0.9334	0.728	0.861
60	0.9147	0.719	0.852
80	0.8012	0.7021	0.811

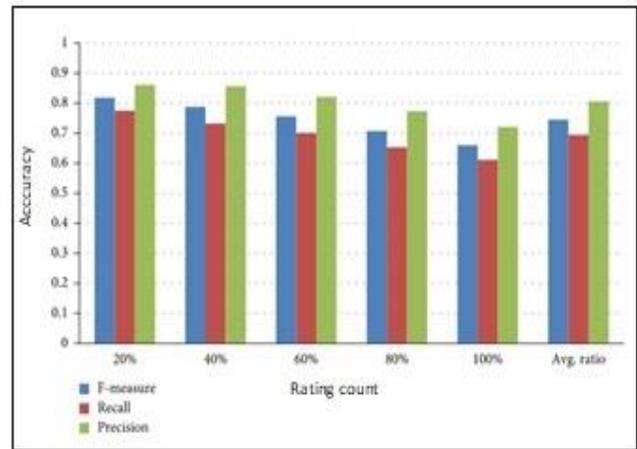


Fig. 13. Predicted Accuracy based on Sentiment Score.

V. CONCLUSION

In this paper a number of studies on recommendation were analyzed and a hybrid recommender system is proposed which works with a Sentiment Analysis model to filter the final results. This system focuses on keeping the computations lesser while still incorporating review data into the recommendations, which contains critical information about the opinions of the viewers. Hybrid SVD is used to generate effective movie recommendations, while a multilayer perceptron is used for Sentiment Analysis to optimize the accuracy level higher. The system performs competitively with other methods, while also incorporating written reviews. For future work, this proposed system might be tested further with a more comprehensive data, for generating recommendations.

VI. FUTURE WORK

The limitation of our work is, we did not have the sentiment score merged with movie dataset. In future research work, we are interested in analyzing the various techniques of sentiment analysis with the respect to the different types of recommendation techniques.

REFERENCES

- [1] Eyjolfsdottir E, Tilak G, Li N (2010) MovieGEN: A Movie Recommendation System. Comput Sci Dep.
- [2] Brusilovsky P, Kobsa A (2007) The Adaptive Web.
- [3] Zhang Z, Zeng DD, Abbasi A, et al (2013) A random walk model for item recommendation in social tagging systems. ACM Trans Manag Inf Syst 4. <https://doi.org/10.1145/2490860>.
- [4] Soni K, Goyal R, Vadera B, More S (2017) A Three Way Hybrid Movie Recommendation System. Int J Comput Appl 160:29–32. <https://doi.org/10.5120/ijca2017913026>.
- [5] B.Thorat P, M. Goudar R, Barve S (2015) Survey on Collaborative Filtering, Content-based Filtering and Hybrid Recommendation System. Int J Comput Appl 110:31–36. <https://doi.org/10.5120/19308-0760>.
- [6] Kirmemis O, Birturk A (2008) A content-based user model generation and optimization approach for movie recommendation. AAAI Work - Tech Rep WS-08-06:78–88.
- [7] Zenebe A, Norcio AF (2009) Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems. Fuzzy Sets Syst 160:76–94. <https://doi.org/10.1016/j.fss.2008.03.017>.
- [8] Sarwar B, Karypis G, Konstan J, Riedl J (2001) Item-based collaborative filtering recommendation algorithms. Proc 10th Int Conf

- World Wide Web, WWW 2001 285–295. <https://doi.org/10.1145/371920.372071>.
- [9] A. Patil S, Pagare R (2014) Enhanced Hybrid Recommender System using Social Friend Network. *Int J Manag Inf Technol* 10:2023–2031. <https://doi.org/10.24297/ijmit.v10i4.627>.
- [10] Wang Z, Yu X, Feng N, Wang Z (2014) An improved collaborative movie recommendation system using computational intelligence. *J Vis Lang Comput* 25:667–675. <https://doi.org/10.1016/j.jvlc.2014.09.011>.
- [11] Fang, Z., Zhang, L., & Chen, K. (2016). Hybrid Recommender System Based on Personal Behavior Mining. *ArXiv*, abs/1607.02754.
- [12] Raval T, Patel Y (2017) A Survey: Collaborative Filtering, Content-based Filtering, Hybrid Recommendation Approach. 193–197.
- [13] Ye H (2011) A personalized collaborative filtering recommendation using association rules mining and self-organizing map. *J Softw* 6:732–739. <https://doi.org/10.4304/jsw.6.4.732-739>.
- [14] Jotheeswaran J, Koteeswaran S (2015) Decision tree based feature selection and multilayer perceptron for sentiment analysis. *ARPN J Eng Appl Sci* 10:5883–5894.
- [15] Ashok M, Rajanna S, Joshi PV, Kamath SS (2016) A personalized recommender system using Machine Learning based Sentiment Analysis over social data. 2016 IEEE Students' Conf Electr Electron Comput Sci SCEECs 2016. <https://doi.org/10.1109/SCEECs.2016.7509354>.
- [16] Qian X, Feng H, Zhao G, Mei T (2014) Personalized recommendation combining user interest and social circle. *IEEE Trans Knowl Data Eng* 26:1763–1777. <https://doi.org/10.1109/TKDE.2013.168>.
- [17] Yuan X, Lee JH, Kim SJ, Kim YH (2013) Toward a user-oriented recommendation system for real estate websites. *Inf Syst* 38:231–243. <https://doi.org/10.1016/j.is.2012.08.004>.
- [18] Singh VK, Piryani R, Uddin A, Waila P (2013) Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. *Proc - 2013 IEEE Int Multi Conf Autom Comput Control Commun Compress Sensing, iMac4s 2013* 712–717. <https://doi.org/10.1109/iMac4s.2013.6526500>.
- [19] Nazim Uddin M, Shrestha J, Jo GS (2009) Enhanced content-based filtering using diverse collaborative prediction for movie recommendation. *Proc - 2009 1st Asian Conf Intell Inf Database Syst ACIIDS 2009* 132–137. <https://doi.org/10.1109/ACIIDS.2009.77>.
- [20] Bellogín A, Wang J, Castells P (2011) Text retrieval methods for item ranking in collaborative filtering. *Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics)* 6611 LNCS:301–306. https://doi.org/10.1007/978-3-642-20161-5_30.
- [21] Yang D, Zhou Z (2013) Personalized mining of preferred paths based on web log. *Proc 2013 IEEE 11th Int Conf Electron Meas Instruments, ICEMI 2013* 2:993–997. <https://doi.org/10.1109/ICEMI.2013.6743199>.
- [22] Paterek A (2007) Improving regularized singular value decomposition for collaborative filtering. *KDD Cup Work* 2–5. <https://doi.org/10.1145/1557019.1557072>.
- [23] Guia M, Silva RR, Bernardino J (2019) A hybrid ontology-based recommendation system in e-commerce. *Algorithms* 12:1–19. <https://doi.org/10.3390/a12110239>.
- [24] Chen J, Zhao C, Uliji, Chen L (2020) Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering. *Complex Intell Syst* 6:147–156. <https://doi.org/10.1007/s40747-019-00123-5>.
- [25] Harper FM, Konstan JA (2015) The movielens datasets: History and context. *ACM Trans Interact Intell Syst* 5:. <https://doi.org/10.1145/2827872>.
- [26] Silveira T, Zhang M, Lin X, et al (2019) How good your recommender system is? A survey on evaluations in recommendation. *Int J Mach Learn Cybern* 10:813–831. <https://doi.org/10.1007/s13042-017-0762-9>.
- [27] Soong HC, Jalil NBA, Kumar Ayyasamy R, Akbar R (2019) The essential of sentiment analysis and opinion mining in social media: Introduction and survey of the recent approaches and techniques. *ISCAIE 2019 - 2019 IEEE Symp Comput Appl Ind Electron* 272–277. <https://doi.org/10.1109/ISCAIE.2019.8743799>.
- [28] Martin Ward Powers D (2011) ABC Unconscious Computer Interface View project Autonomous Robotics View Project Evaluation: From Precision, Recall and F-Measure to Roc, Informedness, Markedness & Correlation. 2:37–63. <https://doi.org/10.9735/2229-3981>.