Music Recommendation System

kishan Rai
Computer Science and Engineering
Rajkumar Goell Institute off Technology
Ghaziabad, India

Ritik Gangwar
Computer Science and Engineering
Rajkumar Goell Institute off Technology
Ghaziabad, India

Abstract

A Music Recommendation has its importance in our social life-style because it affords advanced entertainment as properly as custom designed man or woman person experience. Such recommendation systems can suggest a set of songs to users based on their interest or the popularity of the musics. Though we have hundreds of music recommendation systems, most of these either cannot recommend a songs to the existing users efficiently or to a new user by any means. In this paper, we have built a hybrid model by stacking content and collaborative models to increase the accuracy of movies recommended by the user. This recommendation system mines music databases to gather all the salient information, such as popularity and attractiveness, required for a recommendation. In many cases, various categories of items may show significant correlations, which can be leveraged to make more accurate recommendations. Experimental studies on the actual data reveal the efficiency and effectiveness of the proposed system. This Music Recommendation system will use a hybrid recommendation system which is the amalgamation of content and collaborative based filtering, to search the movies that are similar to the customer's taste or genre and would also provide a combination of the recommendations that are highly rated by different customers' experience so that the customer is not limited to a particular category or genre but is also able to explore new and different varieties as well. The system will get trained for the improvement of the recommendations through hybrid means.

Keywords— Recommendation System, Hybrid Filtering, Similarity Measures, Content, Collaborative, Similarity Measures.

I. INTRODUCTION

Recommendation System is an information tool that allows its users to discover the items that they want from a large number of items given [1], [2]. The main goal of the recommendation system is to forecast the rating that a specific user gives to an item. It allows the user to find the most appropriate solution from the given list of items. Many companies use recommendation systems so that they can serve their user and raise their profit like Netflix, YouTube, Amazon, and others. Even now it is one of the most important topics of research because discovering what the user wants from the given aid is a massive challenge, as the selection of the customers maintains on converting from time to time. Nowadays whatever we buy online is through recommendation. Taking example when we want to buy books, listen to music, watch movies, etc. there is one recommendation system that is working in the background which suggests the user based on his previous actions [3]. Many online platforms like Netflix which suggests movies, Amazon which suggests products, Spotify which suggests music, LinkedIn that is used for recommending jobs, or any social networking sites which suggest users, all are based on recommendation systems [4], [6]. By using these recommendation engines users can easily find out what the user wants according to their choice. So building an effective recommender system is also a challenge because users' preference keeps on changing with time. A advice gadget is an data filtering gadget that tries to are expectingThe opportunities of a purchaser and makes suggestions based mostly on the ones opportunities. There are tens of thousands and thousands of packages for advice systems. They have become demanding over the last few years and are now utilized in most online platforms that we use. The content of such platforms varies from films, songs, books,to friends and stories on social media platforms, To merchandise on e-trade websites, to human beings on expert and courting websites, to discover solutions lower back on Google. Often, those structures can acquire statistics approximately a user’s choices, and might use this statistics to enhance their tips withinside the future.For example, Facebook can display your interplay with numerous memories in your feed to examine what forms of memories enchantment to you. Sometimes, the recommender systems can make upgradation based on the performance of a huge number of folks. Due to the great progress in recommender systems, users continually look forward to good recommendations. They look down to services that are not able to make the most appropriate recommendations. If a music streaming app is not able to predict and play the music that the user likes, then the user will simply stop using it. This has caused major pressure on tech companies for improving their recommendation systems. However, the problem is greater complex than it appears. Every person has specific possibilities and likes. Moreover, even the aesthete of an character person can extrade relying on diverse factors, which includes mood, season, or form of paintings the person is doing. For example, the form of tune one would love to pay attention whilst exercise differs significantly from the form of tune one might pay attention to while cooking dinner. Another fundamental hardship of the advice structures is the exploration v/s exploitation problem. And it should try to solve it. They must try to research in more new areas to find about the user, while still making the most of what is already known about the user. The Movie Recommendation System recommends the alike movies to users with similar demographic attributes. Since each user is distinct in their ways, this approach is considered to be too plain. The main approach behind this system is that more celebrated and system is that more celebrated and lauded musics will have a higher chance of being liked by the average audience. Second is content-based filtering, where we try to profile the user interests using information collected, and recommend items based on that profile. The other is collaborative filtering, where we try to group similar users and use information about the group to make recommendations to the user.

II. LITERATURE REVIEW

S. J. Lee et al. [8] suggested a new method that implements a (self-constructing clustering) algorithm for reducing the dimensionality associated with the number of items. Related gadgets are grouped withinside the identical cluster, even as numerous gadgets are in one of a kind clusters[8].Then advice paintings is executed with the ensuing clusters. In the very last stage, retransformation is carried out and a (Top-N) of encouraged objects is obtainable to every user.In this approach, the time of processing is much reduced (the time has been reduced from 43.83s to 0.26s based on Movielens dataset) to provide recommendations. The experimental result show (was 0.757 Mean Absolute Error (MAE) and 0.959 Root Mean Square Error (RMSE) that the capability of the RS can be much enhanced without compromising the quality of the recommendation.

Sappadla et.al., [9] in 2017, presented a system based on C and CBF using K-Nearest-Neighbour (KNN) method and experiments were performed on the Movielens dataset. The performance of user-based CF and CBF techniques is 0.885 and 0.9554 respectively, according to mean squared error (MSE), in this experiment the data type 100k was used. The user-based CF technique gives the best performance.

Mishra, N., et al.,[10] proposed a new approach in the same year of 2007 that can solve the problem of sparsity in movie recommendation to a great extent by applying to cluster and using IMDB dataset.

Srivastava, R. et al., [11] proposed a k-means approach o alleviating the sparsity problem. When examining this method with other methods, (k-mean clustering-based methods) presented a better and easier performance in applying and managing comparison of other methods also in high sparsity level. In this approach, the correctness of sparsity reduction depends on the performance of clustering.

T. Matsuo et al. [12] presented research related to the hybrid recommendation based mainly on CF (using Non-Personalized Recommendation) and CBF through using machine learning. This system was designed in the form of its construction in the rating frequency form, and lastly finding inverse ratings frequency method combined with the Cosine Technique which used to find similarities (In other words, achieved through Term Frequency-inverse Rating Frequency (Referred as TF-IRF). The results of the two methods showed an effective result through a minimum of web page load time (the time factor was used to determine the effectiveness of the system), where the time of web page load was between (0.32 to 0.41 seconds).

Zhouen, P., L. Chen [50] applied item-based CF in a cloud computing form and used a big data environment to handle the large size of the rating matrix, where Movie lens dataset 10M was used (they selected 10681 moves to form a 71567 x 106 1 matrix). For the experimental at, the first 1000 rows were used for testing purposes. They used the Hadoop model to perform recommendations based on CF I and framework MapReduce. According to the experimental statistics, the device can put in force excessive reliability and performance at the massive statistics set. O. P. Verma et al.

O.P. Verma et al., [21] suggested a novel RS-based cuckoo-search optimization and K-means. The method is efficient and innovative, employing an algorithm for an efficient recommendation for Dataset of Movie lens. The mode has (0.68) based on MAE, the proposed approach also has an efficient improvement in their previous work (which was 0.78 MAE).

S.H. Hashim et al., [15] proposed a collaborative filtering-based non-sparse rating matrix, and a Movie lens dataset was used for this experiment. The results show that the prediction has been enhanced by (10% to1 5%). They got the best result at 0.60 according to MAE.

M.T. Imtiaz., et al [16] presented a movies recommendation by using clustering and pattern recognition network. For separating users that have related taste, K-means has been done and the neural network (NN) has been employed to analyze the users' behavior.

Jain, K.N., et al [17] built a website to implement a recommendation system, and they used the hybrid system approach (consisting of CF and CBF). which was implemented through the application of CBF over CF and using the Movie lens dataset (10k ratings). The CF employs the Pearson correlation coefficient and the CBF uses the genres correlated with the movies. The result from the experiment was 0.980763 according to (MSE) measurement.

Wu, H., et al., [25] suggested dual-regularized Matrix Factorization (MF) with deep neural networks (DNN), and they used for MF the probabilistic matrix factorization denoted as PMF. They used four real data sets for experiments. "Kindle Store (KS), App-Android (AA), and Amazon-Instant Video (AIV, " are obtained from data of Amazon products while Yelpdatasetst are collected from the website Yelp. The results showed that the KS dataset gives the best accuracy, which was 0.5507 and 0.7736 based on (MAE) and (RMSE) respectively.

Zhang, W., et al., [26], a new approach deep variational MF recommendation is suggested for a sparse dataset of big scale. It is used to expect the ratings based on latent factors. For the users and item, the features are obtained via a deep-nonlinear form. The experiments were on three actual datasets collected from different domains, the datasets were 10M Movie lens, Book-Crossing dataset, and Job dataset. The results showed that DVMF can achieve higher accuracy compared with other recommendation algorithms based on deep-learning or MF separately on big sale sparse datasets. The performance according to Movie lens, BookCrossing, and Job dataset was 0.692738,2.611982 and 0.470229 respectively based on MAE and was 0.914313,3.271782, and 0.657864 on RMSE.

Ahuja R [27] proposed an approach to use the user information in the recommendation system. MovieLens dataset that contains 1000 ratings was used in this approach, they selected information from more than 900 users and they used it for this large data principle of clustering. The implementation of the recommendation algorithm was as follow: First, user information such as ID, occupation, age, gender, and pin code was taken, then the K-means clustering algorithm was applied, then the similarity was calculated between users using the Pearson method, finally, K-Nearest-Neighbor was implemented to find the prediction. The performance was 1.08 1648 according to the RMSE. I

Singh S.P et al. [28] applied movie recommendation by using a nature-inspired algorithm and data clustering Because of its simple nature, the k-mean algorithm is widely used for clustering, handling a huge amount of data and low running time, but it locates in the local Optima because of the random initial centroids. They proposed that the algorithm can obtain the global optimum solution via combining this algorithm with the nature-inspired algorithm. This paper incorporates nature-inspired algorithms (modified cuckoo search (MCs). Cuckoo (C-cuckoo), bat (C-bat and firefly(C-firefly) with k -means and Movie lens dataset was used for this experiment 100k) This algorithm k-mean MCS) achieved better results compared to other algorithms (C-cuckoo, C-bat, and C- firefly, got lower results, which are 0.712984,5.479074 and 3.079919 respectively), while the proposed approach was 0.690459 according to objective function value.

.

III. PROPOSED APPROACH

The proposed approach being used takes into account the content and collaborative filtering and makes use of strength of each of these algorithms to provide the recommendations in the optimal time and to perform the computation in the movie recommender system we implement the cosine similarity measurer with the necessary machine learning algorithms to perform and predict the accuracy of each of our model

A. Proposed Algorithm

Content-Based Filtering

For the implementation of a content-based filtering system following steps to be done:

- Terms Allocation
- Terms Representation
- Learning Algorithm Selection
- Provide Recommendations.

Collaborative Filtering

Collaborative Filtering takes into account the user's rating and reviews regarding the movie and therefore allows us to approach with two types of approaches.

Item-based Nearest Neighbour:
This approach is used to generate recommendations based on the similarity between the items. The prediction for the item should be based on similar items with a base criteria,for eg:-Vote Count..Then it takes into account the weighted average of the target item,based on ratings on the similar items and makes usage of cosine similarity function to filter out the results

User-based Nearest Neighbour:
This algorithm generates predictions based on the ratings on the item by the similar users.So if a user n1 has the same taste as that of n2 the predictions,n1 is said to be a nieghbour of n2,so ratings will be generated in the neighbourhood of user's of similar taste for any item.

Cosine Similarity
Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction.

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Hybrid Recommendation System
Hybrid Filtering Recommendation System: The hybrid filtering recommendation system works by employing the working principles of both the systems such as content-based and collaborative recommendation system. It implements a item-item model by integrating content-based filtering technique into the collaborative filtering model. It is quite effective in its usage and can be made used of in many fields involving Music, Movies, e-commerce, social networking etc. The hybrid filtering technique is classified into following seven classes based on Burke's taxonomy, which are as follows Weighted, Switching, Mixed, Feature Combination, Feature augmentation, Cascade and Metalevel. In term of accuracy in prediction and recommendation the hybrid method outperforms the conventional methods. The hybrid method can integrate with other advanced machine learning techniques such as clustering involving the process of grouping large amount of data using sequential and Linear combination.

K-Nearest Neighbour
K-nearest neighbour algorithm(abbreviated as KNN) is one of the most popular Machine Learning algorithm used for regression as well as classification purposes KNN algorithm assumes the similarity between the available data and the new data and then classifies the new data into the closest resemblance with the data provided.It is a non-parametric algorithm which means there are no assumptions about the data from the algorithm.It comes with its advantages like it is able to deal with noisy data with ease and is of great usage when working with large datasets.

Distance functions

Euclidean $\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2}$

Manhattan $\sum_{i=1}^{k} |x_i - y_i|$

Minkowski $\left(\sum_{i=1}^{k} (|x_i - y_i|)^q \right)^{1/q}$

SVD(Singular Value Decomposition):

SVD is a matrix factorization method that generalizes the eigen decomposition of square matrix(n*n) to any matrix n*m.It is a method derived from linear algebra that has been used as a dimensionality reduction technique in machine learning.It reduces the dimensions of dataset with n dimension to k-dimension(where k<n).Here each row represents a user,each column an item and the values are the ratings It decomposes a matrix into 3 matrices via:

A=USV^t

- A is a m*n utility matrix,
- U is a m*r orthogonal left singular matrix which represents relationship between user and latent factors
- S is a r*r diagonal matrix representing strength of each latent factor
- V is a r*n diagonal right singular matrix representing relationship between items and latent factors
- Latent items refer to characteristic of an item.For Eg:genre of a movie

SVD++

This algorithm follows the same procedure as of its predecessor SVD alongside with an added feature of implicit feedback capability which helps it in achieving a better predictive accuracy compared to SVD,The core of the proposed method which is to increase the low efficiency of calculating the recommendation is to conduct a backtracking line search in the SVD++ algorithm and the effectiveness of the proposed method is demonstrated by comparing the root mean square error, absolute mean error and recall rate simulation results

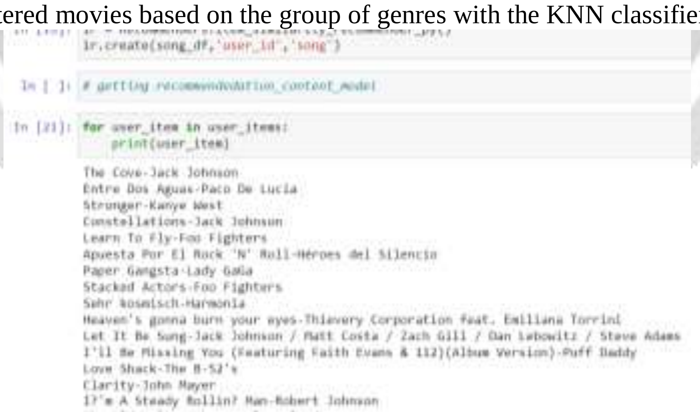
IV. IMPLEMENTATION

To develop this Machine Learning based recommendation system we are employing various technologies such as Google Collapse and Jupyter notebook, a web-based interactive python computing platform. We have coded our project in python and stored and processed the data using pandas data frame. We have used the surprise library for analyzing different prediction algorithms and evaluating the model.

Models

A. Content-Based Recommendation System

Predicts the most similar movies based on the genre choices of the user using cosine similarity. For evaluating the model we have clustered movies based on the group of genres with the KNN classifier.



Hit Ratio and Accuracy of the Model

```
evaluate_content_based_model()
total = true_count + false_count
print("Hit:"+ str(true_count/total))
print("Fault:" + str(false_count/total))
print("Accuracy:"+str(true_count/total))
```

```
Hit:0.8840986336367161
Fault:0.11590136636328384
Accuracy:0.8840986336367161
```

B. Collaborative Based Recommendation System

Collaborative Filtering uses ratings provided by users to set similarities among items and users' choices. We have used Memory-Based Collaborative Filtering and Model-Based Collaborative Filtering.

- Memory-Based Collaborative Filtering

Memory-Based Collaborative Filtering is of two types item-based and user-based.

i) Item Based

We build the item-similarity matrix that will measure the similarity between any two pairs of items.

```
In [8]: # cumulative songs of listencount
        song_grouped = song_df.groupby(['song']).agg({'listen_count':'count'}).reset_index()
        song_grouped.head()
```

Out[8]:

	song	listen_count
0	#!*@ You Tonight [Featuring R. Kelly] (Explic...	78
1	#40-DAVE MATTHEWS BAND	338
2	& Down-Boys Noize	373
3	' Cello Song-Nick Drake	103
4	'97 Bonnie & Clyde-Eminem	93

ii) User Based

We build the user-similarity matrix that will consist of some distance metrics that measure the similarity between any two pairs of users.

```
## song recommended for user_id 100

user_items = ir.get_user_items(song_df['user_id'][100])

for user_item in user_items:
    print(user_item)

Undo-Björk
Dog Days Are Over (Radio Edit)-Florence + The Machine
High Life-Daft Punk
You're The One-Dwight Yoakam
Where Did You Sleep Last Night-Nirvana
Come As You Are-Nirvana
Hey_ Soul Sister-Train
Horn Concerto No. 4 in E flat K495: II. Romance (Andante cantabile)-Barry Tuckwell/Academy of St Martin-in-the-Fields/Sir Neville Marriner
Rhyme & Reason-DAVE MATTHEWS BAND
Sehr kosmisch-Harmonia
Someone Else's Arms-Mae
Cry For Help (Album Version)-Shinedown
Lady In Black-Ensiferum
For You (Amended/Radio Edit LP)-Staind
```

Novel Approach using SVD

Hit Ratio= recommended songs rating>3/total recommended songs

Hit Ratio for User-Based Collaborative filtering: 0.778

Hit Ratio for Item-Based Collaborative filtering: 0.889

```
print("Hit ratio of User-user collaborative filtering")
print(evaluation_collaborative_svd_model(user_id,True))
print("Hit ratio of Item-Item collaborative filtering")
print(evaluation_collaborative_svd_model(user_id,False))
```

```
Hit ratio of User-user collaborative filtering
0.7777777777777778
Hit ratio of Item-Item collaborative filtering
0.8888888888888888
```

- Model-Based Collaborative Approach

We have chosen Single Valued Decomposition (SVD) model to predict user-item ratings. The Sparsity level of SongLens data set is 98.3%.

```
## top ten songs that user_id[5] will enjoy
```

```
pr.recommend(song_df['user_id'][5])
```

	user_id	song	score	Rank
7127	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Sehr kosmisch-Harmonia	8277	1.0
9084	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Undo-Björk	7032	2.0
2068	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Dog Days Are Over (Radio Edit)-Florence + The ...	6949	3.0
9877	b80344d063b5ccb3212f76538f3d9e43d87dca9e	You're The One-Dwight Yoakam	6412	4.0
6774	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Revelry-Kings Of Leon	6145	5.0
7118	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Secrets-OneRepublic	5841	6.0
3613	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Horn Concerto No. 4 in E flat K495: II. Romanc...	5385	7.0
2717	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Fireflies-Charttraxx Karaoke	4795	8.0
3485	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Hey_ Soul Sister-Train	4758	9.0
8847	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Tive Sim-Cartola	4548	10.0

Accuracy for SVD and SVD++Model
For SVD and SVD++ models accuracy is calculated using RMSE (Root Mean Square Error).
Accuracy for SVD: 87.4%
Accuracy for SVD++: 94.3%

```
cross_validate(svd, data, measures=['RMSE', 'MAE'], cv=5, verbose=True)
```

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

                  Fold 1  Fold 2  Fold 3  Fold 4  Fold 5  Mean    Std
RMSE (testset)    0.8736  0.8732  0.8715  0.8729  0.8750  0.8732  0.0011
MAE (testset)     0.6718  0.6733  0.6690  0.6691  0.6702  0.6707  0.0017
Fit time          4.88    5.07    5.08    4.82    4.85    4.94    0.11
Test time         0.13    0.13    0.19    0.13    0.13    0.14    0.02
```

```
algo_svdpp = SVDpp(n_factors=160, n_epochs=10, lr_all=0.005, reg_all=0.1)
algo_svdpp.fit(trainset)
test_pred = algo_svdpp.test(testset)
print("SVDpp : Test Set")
accuracy.rmse(test_pred, verbose=True)
```

```
SVDpp : Test Set
RMSE: 0.9429

0.9428872306340533
```

C. Hybrid Recommendation System

After developing individual models, we stack them to get better results and develop our own recommendation system using content and the SVD approach.

Run Content-based filtering and determine the movies which we want to recommend to the user.

Filter and sort the recommendations of CF using SVD predicted ratings.

```
In [12]: pr.recommend(song_df['user_id'][5])
Out[12]:
```

	user_id	song	score	Rank
7127	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Sehr kosmisch-Harmonia	8277	1.0
9084	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Undo-Bjork	7032	2.0
2068	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Dog Days Are Over (Radio Edit)-Florence + The ...	6949	3.0
9877	b80344d063b5ccb3212f76538f3d9e43d87dca9e	You're The One-Dwight Yoakam	6412	4.0
6774	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Revelry-Kings Of Leon	6145	5.0
7115	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Secrets-OneRepublic	5841	6.0
3613	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Horn Concerto No. 4 in E flat K495: II. Romanc...	5385	7.0
2717	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Fireflies-Charttraxx Karaoke	4795	8.0
3486	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Hey_ Soul Sister-Train	4758	9.0
8847	b80344d063b5ccb3212f76538f3d9e43d87dca9e	Tive Sim-Cartola	4548	10.0

V. RESULT

We have implemented cosine similarity to evaluate the content-based model by using the KNN classifier and hit ratio as a metric. The hit ratio for the content-based model is 0.884. We have implemented a novel technique to evaluate collaborative filtering algorithms by using SVD and hit ratio as a metric. In Memory based collaborative approaches, the hit ratio of user-user filtering is **0.778**, and for Item-Item filtering **is 0.889**.

We attempted to build a model-based Collaborative Filtering movie recommendation system based on latent features from a low-rank matrix factorization method called SVD and SVD++. As it captures the underlying features driving the raw data, it can scale significantly better to massive datasets as well as make better recommendations based on users' tastes. In Model-based collaborative approaches, the RMSE for SVD and SVD++ models are **0.87** and **0.943** respectively.

VI. CONCLUSION, LIMITATIONS, FUTURE SCOPE

A. Conclusions

Under the condition of massive information[21], the requirements of movie recommendation systems from film amateurs are increasing. This project designs and implements a complete movie recommendation system prototype based on the KNN algorithm, collaborative filtering algorithm, content-based filtering algorithm, and recommendation system technology. We give a detailed design and development process and test the stability and high efficiency of the experiment system through professional tests. This project has reference significance for the development of personalized recommendation technology. Various uses, advantages, disadvantages are also discussed. To build an efficient recommender system a hybrid combination of different methods of recommendation is a must. It is concluded that by using a combination of similarity measures a better user similarity can be generated rather than using a single similarity measure and the efficiency of the system is also increased. One of the facts is that similarity measures like RJMSD are evolved by the author and up till now it is only used in movie recommendations. The author also showed that this similarity measure is better than the other in terms of efficiency parameters. In this project, to improve the accuracy, quality, and scalability of movie recommendation system, a Hybrid approach by unifying content-based filtering and collaborative filtering; using Singular Value Decomposition (SVD) as a classifier and Cosine Similarity is presented in the proposed methodology. Existing pure approaches and a proposed hybrid approach are implemented on three different Movie datasets and the results are compared among them. Comparative results depict that the proposed approach shows an improvement in the accuracy, quality, and scalability of the movie recommendation system than the pure approaches. Also, the computing time of the proposed approach is lesser than the other two pure approaches.

B. Limitations

a) **The cold-start problem:** Collaborative filtering systems are based on the action of available data from similar users. If we are building a brand new recommendation system, we would have no user data to start with. We can use content-based filtering first and then move on to the collaborative filtering approach.

b) **Scalability:** As the number of users grows, the algorithms suffer scalability issues. If you have 10 million customers and 100,000 movies, you would have to create a sparse matrix with one trillion elements.

c) **The lack of right data:** Input data may not always be accurate because humans are not perfect at providing ratings. User behavior is more important than ratings. Item-based recommendations provide a better answer in this case.

C. Future Scope

In the proposed approach, It has considered Genres of movies[15] but, in the future, we can also consider the age of user as according to the age movie preferences also changes, like for example, during our childhood we like animated movies more as compared to other movies. There is a need to work on the memory requirements of the proposed approach in the future. The proposed approach has been implemented here on different movie datasets only. It can also be implemented on the Film Affinity and Netflix datasets and the performance can be computed in the future

ACKNOWLEDGMENT

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend my sincere thanks to all of them. We are highly indebted to Dr. Shashi Kant Singh for his guidance and constant supervision. Also, we are highly thankful to them for providing necessary information regarding the project & also for their support in completing the project. We are extremely indebted to Dr. Vishnu Sharma, HOD, Department of Computer Science and Engineering, GCET, and Dr. Jaya Sinha, Project Coordinator, Department of Computer Science and Engineering, GCET for their valuable suggestions and constant support throughout my project tenure. We would also like to express our sincere thanks to all faculty and staff members of the Department of Computer Science and Engineering, GCET for their support in completing this project on time. We also express gratitude towards our parents for their kind co-operation and encouragement which helped us in the completion of this project. Our thanks and appreciations also go to our friends in developing the project and all the people who have willingly helped us out with their abilities.

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, “Title of paper if known,” unpublished.

[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[8] Aggarwal, C.C., Recommender systems. 2016: Springer.

[9] Lin, C.-Y., L.-C. Wang, and K.-H. Tsai, Hybrid real-time matrix factorization for implicit feedback recommendation systems. IEEE Access, 2018. 6: p. 21369-21380.

[10] Yi, B., et al., Deep matrix factorization with implicit feedback embedding for recommendation system. IEEE Transactions on Industrial Informatics, 2019. 15(8): p. 4591-4601.

[11] Chen, J., et al. An Implicit Information Based Movie Recommendation Strategy. in 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). 2018. IEEE.

[12] Dhruv, A., et al., Artist Recommendation System Using Hybrid Method: A Novel Approach, in Emerging Research in Computing, Information, Communication and Applications. 2019, Springer. p. 527-542.

[13] Kluver, D., M.D. Ekstrand, and J.A. Konstan, Rating-based collaborative filtering: algorithms and evaluation, in Social Information Access. 2018, Springer. p. 344-390.

[14] Mahata, A., et al. Intelligent movie recommender system using machine learning. in International Conference on Intelligent Human Computer Interaction. 2016. Springer.

[15] Liao, C.-L. and S.-J. Lee, A clustering based approach to improving the efficiency of collaborative filtering recommendation. Electronic Commerce Research and Applications, 2016. 18: p. 1-9.

[16] Sappadla, P., Y. Sadhwani, and P. Arora, Movie Recommender System. Search Engine Architecture, Spring, 2017.

[17] Mishra, N., et al., Solving Sparsity Problem in Rating-Based Movie Recommendation System, in Computational Intelligence in Data Mining. 2017, Springer. p. 111-117.

[18] Shrivastava, R. and H. Singh, K-means clustering based solution of sparsity problem in rating based movie recommendation system. International Journal of Engineering and Management Research (IJEMR), 2017. 7(2): p. 309-314.

[19] Subramaniam, R., R. Lee, and T. Matsuo. Movie Master: Hybrid Movie Recommendation. in 2017 International Conference on Computational Science and Computational Intelligence (CSCI). 2017. IEEE

[20] Zhou, T., L. Chen, and J. Shen. Movie Recommendation System Employing the User-Based CF in Cloud Computing. in 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC). 2017. IEEE.

[21] Katarya, R. and O.P. Verma, An effective collaborative movie recommender system with cuckoo search. Egyptian Informatics Journal, 2017. 18(2): p. 105-112.

[22] Al-Bakri, N.F. and S.H. Hashim, Reducing Data Sparsity in Recommender Systems. Al-Nahrain Journal of Science, 2018. 21(2): p. 138-147.

[23] Ahmed, M., M.T. Imtiaz, and R. Khan. Movie recommendation system using clustering and pattern recognition network. in 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC). 2018. IEEE.

[24] Jain, K.N., et al., Movie recommendation system: hybrid information filtering system, in Intelligent Computing and Information and Communication. 2018, Springer. p. 677-686.

[25] Wu, H., et al., Dual-regularized matrix factorization with deep neural networks for recommender systems. Knowledge-Based Systems, 2018. 145: p. 46-58.

[26] Zhang, W., et al., A deep variational matrix factorization method for recommendation on large scale sparse dataset. Neurocomputing, 2019. 334: p. 206-218.

[27] Ahuja, R., A. Solanki, and A. Nayyar. Movie Recommender System Using K-Means Clustering AND K-Nearest Neighbor. in 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). 2019. IEEE.

[28] Singh, S.P. and S. Solanki, A Movie Recommender System Using Modified Cuckoo Search, in Emerging Research in Electronics, Computer Science and Technology. 2019, Springer. p. 471-482.

[29] Isinkaye, F., Y. Folajimi, and B. Ojokoh, Recommendation systems: Principles, methods and evaluation. Egyptian Informatics Journal, 2015. 16(3): p. 261-273.

[30] Mustafa, N., et al. Collaborative filtering: techniques and applications. in 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE). 2017. IEEE.

[31] Raghuwanshi, S.K. and R. Pateriya, Collaborative Filtering Techniques in Recommendation Systems, in Data, Engineering and Applications. 2019, Springer. p. 11-21.

[32] Amin, S.A., J. Philips, and N. Tabrizi. Current Trends in Collaborative Filtering Recommendation Systems. in World Congress on Services. 2019. Springer.

[33] Ortega, F., et al., Recommending items to group of users using matrix factorization based collaborative filtering. Information Sciences, 2016. 345: p. 313-324.

[34] FadhelAljunid, M. and D. Manjaiah, A survey on recommendation systems for social media using big data analytics. International Journal of Latest Trends in Engineering and Technology, 2017: p. 48-58.

[35] Wu, Z., et al. Optimization matrix factorization recommendation algorithm based on rating centrality. in International Conference on Data Mining and Big Data. 2018. Springer.

[36] Bokde, D., S. Girase, and D. Mukhopadhyay, Matrix factorization model in collaborative filtering algorithms: Asurvey. Procedia Computer Science, 2015. 49: p. 136-146.

[37] Subramaniyaswamy, V., et al., A personalised movie recommendation system based on collaborative filtering. International Journal of High Performance Computing and Networking, 2017. 10(1-2): p. 54-63.

[38] Jain, S., et al. Trends, problems and solutions of recommender system. in International Conference on Computing, Communication & Automation. 2015. IEEE.

[39] Jain, A., V. Jain, and N. Kapoor, A literature survey on recommendation system based on sentimental analysis. Advanced Computational Intelligence, 2016. 3(1): p. 25-36.

[40] Barjasteh, I., et al., Cold-start recommendation with provable Guarantees: A decoupled approach. IEEE Transactions on Knowledge and Data Engineering, 2016. 28(6): p. 1462-1474.

[41] Wei, J., et al., Collaborative filtering and deep learning based recommendation system for cold start items. Expert Systems with Applications, 2017. 69: p. 29-39.

[42] Yu, Y., et al., Attributes coupling based matrix factorization for item recommendation. Applied Intelligence, 2017. 46(3): p. 521-533.

[43] AL-Bakri, N.F. and S.H. Hashim, Collaborative Filtering Recommendation Model Based on k-means Clustering. Al-Nahrain Journal of Science, 2019. 22(1): p. 74-79.

[44] SPnchez-Moreno, D., et al., A collaborative filtering method for music recommendation using playing coefficients for artists and users. Expert Systems with Applications, 2016. 66: p. 234-244

[45] Ghorbani, S. and A.H. Novin, An introduction on separating gray-sheep users in personalized recommender systems using clustering solution. International Journal of Computer Science and Software Engineering, 2016. 5(2): p. 14-18.

[46] Badsha, S., et al. Privacy preserving user-based recommender system. in 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). 2017. IEEE.

[47] Karthikeyan, P., et al. Prevention of shilling attack in recommender systems using discrete wavelet transform and support vector machine. in 2016 Eighth International Conference on Advanced Computing (ICoAC). 2017. IEEE.

[48] Si, M. and Q. Li, Shilling attacks against collaborative recommender systems: a review. Artificial Intelligence Review, 2020. 53(1): p. 291-319.

[49] Bathla, G., H. Aggarwal, and R. Rani, Improving Recommendation Techniques by Deep Learning and Large Scale Graph Partitioning. INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS, 2018. 9(10): p. 403-409.

[50] Zhou, X., et al., SVD-based incremental approaches for recommender systems. Journal of Computer and System Sciences, 2015. 81(4): p. 717-733.