2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon)

Hybrid Recommendation System with Graph based and Collaborative Filtering Recommendation Systems

1st Shubham Khanduri
SRM Institute of Science and Technology
Kattankulathur, India
shubham140700@gmail.com

2nd S. Prabakeran
SRM Institute of Science and Technology
Kattankulathur, India
prabakes@srmist.edu.in

2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon) | 978-1-6654-9790-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/MysuruCon55714.2022.9972677

***Abstract*—Every online platform on the internet needs to have some kind of recommendation system. A recommendation system will suggest products or services on a platform to a user, based on their preference and history. This increases user engagement on the platform, as they help the user in their choices. Hence, there is a need for a simple recommendation system solution. Recommendation system works by creating relations between products or the users; and by evaluation of those relations, it can suggest relevant products related to any product. Hybrid Recommendation combines the predictions of two or more recommendation techniques. This system combines the predictions from product - based predictions; implemented using a Graph based recommendation system, and users - based predictions; implemented using Collaborative Filtering recommendation system.**

Keywords—Recommendation System, Network, Graph, Collaborative Filtering.

I. Introduction

Recommendation Systems are automated systems that filter out products, content or services for a customer based on their previous engagements and preferences. Recommendation System can be of two types: Personalized and Non - personalized. Personalized recommendation systems provide the users recommendations based on their liking or their history. Non - Personalized recommendation systems are simple systems that recommend based on general information of a product. For example, reviews from other users, popularity, or pricing. These are mainly used for a user trying to get a service or product on a new platform or in a new category, where no information of the user liking, or history is available. [9]

Recommendation System can also be categorized into several categories based on what data they use [2]:

- Content-based Filtering Systems: Uses the users' data and recommends based on data available for the products.
- Collaborative Filtering Systems: Uses a group of users and their relationship with the products to recommend products to a user.
- Demographic Filtering Systems: Filters products for users based on their demographic such as age, gender, education.
- Hybrid recommender Systems: It is the usage of multiple recommendation systems. It uses both content - based and collaborative approaches for recommendations.
- Graph Based recommender Systems: It uses graph to create relations between different products or content.

A. Network Recommendation System

1) Network Link Prediction: Networks can be created for any available entities, connecting one entity to another based on their relation. These networks are now being created everywhere. May it be the ever-growing social network, where we all are connected through our information, or may it be an e-commerce platform that links its products to one another and the customers, or may it be any online service platform. Using various link prediction techniques like Common neighbours, Jaccard coefficient and Adamic Adar measure [8], we can now predict the connection strength between two entities, which helps us to provide better service on every online platform available.

2) Adamic Adar: The formula takes in the nodes that have the common nodes in between them. Then it calculates the strength of the connection through the number of edges that are available between the nodes. [13]

$$similarity(A,B) = \sum_{shared\ items} \frac{1}{log[freq(shared\ item)]} \quad (1)$$

In (1), A and B are the nodes in question, similarity is the score, shared items are the common nodes

B. Collaborative Filtering

Collaborative Filtering is a recommendation technique that uses a group of users and their relationship with the products. This technique gives recommendation to a particular user by finding similarities with other users and then by analysing how those users interact with the products, the technique predicts products for the user.

1) Matrix Factorization: Matrix Factorization is used for decomposing a matrix of higher dimensionality into two or three matrices of lower dimensionality. This is useful for sparse matrices such as the user movie ratings matrix. These matrices have a lot of empty cells which makes them unsuitable as training data for a model.

2) KNN : K-Nearest Neighbours (KNN) is a supervised learning algorithm used for classification. The algorithm is trained with the data alongside its label. The trained data can be considered as points on a cartesian plane. The algorithm now classifies the data given to it based on K-nearest points.

The nearest trained data is found out by getting distance of input to all the points and getting the nearest points. The distance between points is calculated using Euclidean Distance formula.

978-1-6654-9790-9/22/$31.00 ©2022 IEEE

II. THE PROJECT

The hybrid recommendation system is a combination of the Graph based recommendation system and the collaborative recommendation system. Predictions for a user will be taken from both these systems and using these predictions we will be able to get the top recommendations for a user.

Network analysis or more specifically link analysis is used for identifying relationships in a network that cannot be seen from the raw data. We are using this to perform link prediction and to find strong connections in the network, to create our recommendation system. It is a graph-based recommendation system.

We have demonstrated our system as a movie recommendation system. We are using a dataset that contains details of the movies and ratings available on IMDB. [14] The system will get a movie list which belongs to the user's preference and based on the list the system will recommend movies to the user.

For the collaborative recommendation system, we are using a dataset containing every user's movie ratings. Using this database, we will be clustering the users based on their similarity in their interest. Using this, we can identify the users having similar interests to the user in question and based on the activities of these users we can predict the rating of a movie by the particular user.

A. *Literature Survey*

Recommendation systems are of various types and have various techniques to implement them. In the paper Research-paper recommender systems: a literature survey [7], the authors review multiple different methods to implement a recommender system for research papers. It also reviews graphs-based methods. The survey also shows that for text-filtering based systems, TF-IDF is the most.

For the Network recommendation system, we need to find the strongest links to a particular node. Link prediction methods are used for that purpose. For link prediction, multiple methods are available to use, such as, Common neighbours, Jaccard coefficient and Adamic Adar measure. Out of these, Adamic Adar yields the highest precision in link predictions for nodes in graphs.[3]

In the paper, A Graph-based Recommender System for Digital Library [14], the authors use a graph - based hybrid recommendation system for books in a library and test it out against the content based and collaborative systems. Aside from linking the books with each other, they also linked the customers with each other and with the books through purchase history.

Performing link predictions in a network for a node, yields result relevant in relation to the node. These results can be further classified using machine learning models if necessary. In the paper, Friend Recommendation Using Graph Mining on Social Media [10], the author uses graph and link predictions as a feature extraction method and then applies random forest algorithms for predictions.

Various clustering and classification models can be used to classify the products for recommendations. In the paper, Recommender System in Machine learning [5], the author applies machine learning models to implement a recommendation system.

The paper Movie recommendation system with Collaborative Filtering [1] using K-NN dwells into various methods and implementation to create a collaborative filtering recommendation system. The paper compares the accuracy of various models and concludes that collaborative filtering with matrix factorization using KNN algorithm has the highest accuracy.

Recommendation system is an ever-evolving technology. We can implement and integrate various types of recommender systems to fine tune our recommendations. As discussed in the paper, A Recommendation System: Trends and Future [4], we can see the various types of systems and the possibilities of the future variations of the recommendation system.

B. *Approach*

The system consists of the two types of recommendation systems: Graph based and Collaborative Filtering.

In Graph based, we create a network of the data. The network is represented by an undirected graph. Then the data is loaded in the graph as nodes and linking all the movie nodes with its detail nodes. We will be getting a user movie list. This can be the user's favourite movies or user's recently watched movies. Now, link prediction is performed for each of the movies in the list.

For link predictions for a movie, we need to find the movies connected to it. Here, a connection between does not mean direct linking, but represents connection through a common node (a detail node). Two connected movie nodes can have multiple connecting nodes. After getting all the connecting movies and the connecting nodes, we perform the Adamic Adar measure for link prediction. This gives us scores of connected movies which represent the strength of the connection.

So, for each of the movies from the user list, we get a predicted list of movies. For each of these lists, we sort them in descending order of their scores and give them a score based on their ranks. The ranked score of the movies from every list are all added and stored. The highest scoring movies will be given as a recommendation to the user.

Collaborative filtering will compare analyze the ratings of the user in question and the ratings given by other users. Using classification algorithms, we can create a model through which we try to predict how much rating a user will give to a movie, even though they might have never reviewed the movie.

Through both of these systems we can get a top recommendations list. We will then try to combine both lists, based on how a movie performed in both lists. The top movies after this will be the recommendations.

The database contains data for the movies and their details, and the users' ratings on these movies. The system has a usage log that stores the users' usage history on the platform. Based on this data, the user will get movies recommended by the system.

The top ranked movies after combining both of the lists will be the final recommendations. The final recommendations will be then displayed to the user.

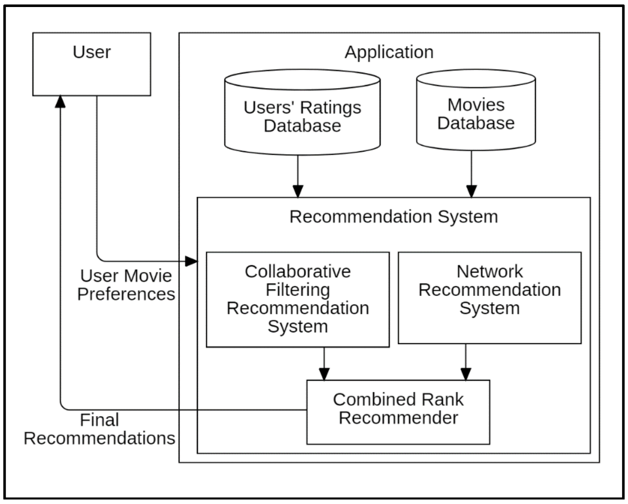

Fig. 1. Architecture Diagram

C. Implementation

1) Database: The data is imported from two datasets, Movie Titles from IMDB and Movie Ratings [15] into a Pandas DataFrame [11].

The first dataset is used in the Graph based recommendation system and contains details for the titles as follows: title, directors, cast, countries, categories and keyword. The second dataset is used in the Collaborative filtering recommendation system. It contains user Id for every user on the platform and ratings for every movie by the users. The ratings can be empty values, as each user cannot give rating to every movie.

2) Modules and Implementation: There are three modules implemented in this system as shown in Fig. 1.

D. Graph Recommendation System

(1) Network Creation: The Network is stored as a form of an undirected graph. The NetworkX module for python is used for creating a graph and entering nodes into it. [12]

After creating the graph, the nodes are to be inserted into it. Each of the movies will have its own main node; denoted by its title name and will be connected to the nodes of its details. We can call the movie node, the Major node and all the nodes of its details and similar movies as the sub-nodes.

(2) Finding Connections: To find the closely related movies for a given movie, we need to find connections between the movie nodes. These connections in the network are the common sub-nodes between two movies.

To find connections to a movie, we look for the second neighbours, i.e., neighbours of the neighbours. Neighbours of a movie are its sub-node and neighbours of these sub-nodes are the movies connected to it. The Fig. 2 shows a basic network between two movies.

(3) Evaluating Connections and Giving Recommendations: After getting the connections, we need to evaluate the strength of the connections. The method used in this project is Adamic Adar measure. This evaluates all the movie nodes with a connection with a particular single movie node. The formula gives out a score for each of the movies that represents the strength of the connection. The highest scored movies are the most closely related movies to the movie in question.

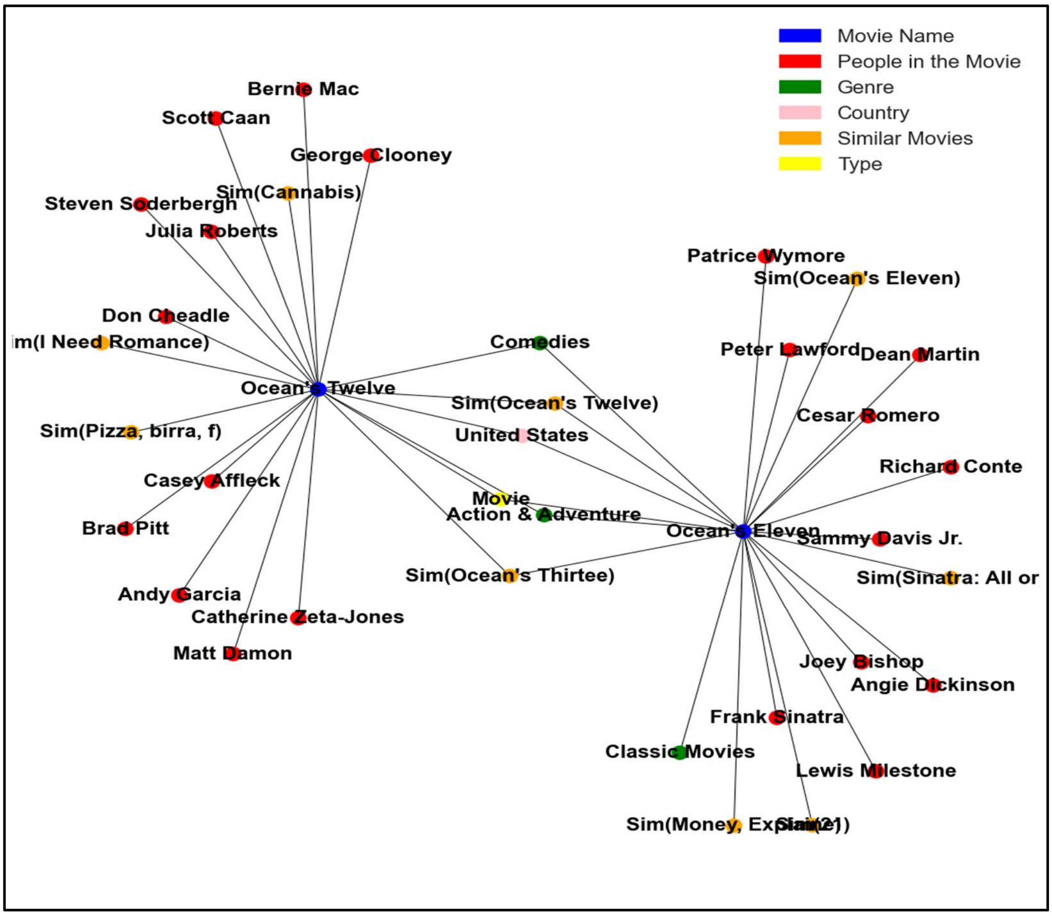

Fig. 2. Sample Image of Network for Two Movies

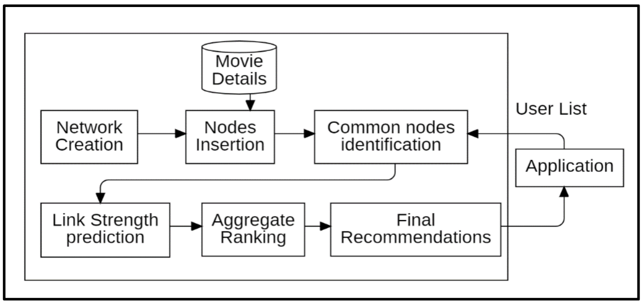

Fig. 3. Graph based Recommendation System

(4) For Multiple Movies: For giving recommendations for multiple movies, we first get the top predictions for each of the movies using the above method. The top scored movies for each of the predicted lists, are given a score based on their rank. The rank scores for every movie, in each of the predicted lists, are added to get the final score. The movies with high final scores are recommended.

For example, let's have a user list of movies [M1, M2]. We will now get predictions for each of them and rank in descending order of scores as shown in Table I.

TABLE. I. PREDICTIONS SCORES

Prediction for M1		**Prediction for M2**		**Final Scores**	
Predicted Movies	***Score Given***	***Predicted Movies***	***Score Given***	***Top Predicted***	***Total Score***
P1	3	P2	3	P2	5
P2	2	P3	2	P1	3
P3	1	P4	1	P3	3

So, the recommended movies are: P2, P1, P3. These movies with be used for recommendation to the user as shown in Fig. 3.

E. Collaborative Filtering

The collaborative filtering system makes recommendations based on users' movie ratings. The dataset of the ratings is a large dataset and contains a lot of empty cells as shown in Table 2. This will increase the root mean squared error. Hence, in order to train our model, we need to apply dimensionality reduction technique. This will get the correlations from the dataset and will remove the irrelevant data.

TABLE. II. RATINGS DATASET MATRIX REPRESENTATION

Matrix	**Movie A**	**Movie B**	**Movie C**	**Movie D**	**Movie E**
User A	3	4			5
User B	1		3	4	
User C	4	2			
User D	5			1	

The matrix factorization will be of singular value decomposition. SVD is used because it can decompose the matrix to its smallest form with the approximation of the actual matrix.

After the feature extraction of the matrix, we create a K nearest neighbour (KNN) model and train it using the matrix. The KNN model finds the closest K trained features and can give the top predictions from the feature space, shown in Fig. 4.

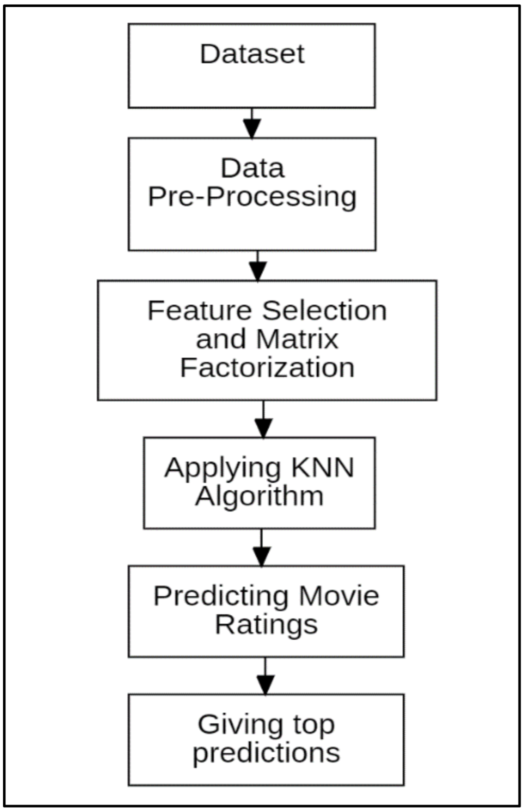

Fig. 4. Collaborative Filtering System

F. Combined Rank Recommender

After getting predicted recommendations from both the systems, we need to the combine them to get the final recommendations. For that we will be using weighted hybridization based on the rank of a movie in both lists. In this, we give a weightage on the ranks from recommendation list from each system. The weights can be modified to give preference on recommendations from one system.

$$Final\ Rank = W_n * R_n + W_c * R_c \tag{2}$$

Where in (2),

W_n = Weight for Network Recommendation Rank

R_n= Network Recommendation Rank

W_c = Weight for Collaborative Recommendation Rank

R_c = Collaborative Recommendation Rank

In our case, we are giving equal weight of 0.5 to both the recommendation systems' ranks.

III. RESULTS

A. Collaborative Filtering Accuracy

The accuracy of the collaborative filtering system can be evaluated by comparing predicted values to the already known user ratings. The rating matrix is split into training and test data, and then the accuracy is obtained.

The accuracy of the collaborative filtering system with matrix factorization using KNN algorithm is 0.88.

B. Sample Output

The given user list to the system => ['Avatar', 'Tangled', 'King Kong', 'Spectre']

C. Network Recommendation System

The recommendations from network recommendation systems are given in Fig. 5

```
Top Ranked Netword Recommendation List =>

Alice Through the Looking Glass
Armageddon
The Adventures of Tintin
Terminator Salvation
Southland Tales
Night at the Museum
Avengers: Age of Ultron
Warcraft
Sky Captain and the World of Tomorrow
Inkheart
```

Fig. 5. Network recommendation systems

D. Collaborative Filtering System

Slice of the highest rated predictions list sorted by their ratings predicted by the system is given in Fig. 6.

```
Top Rated Movies Prediction List =>

Pirates of the Caribbean: At World's End
Batman v Superman: Dawn of Justice
The Avengers
The Lone Ranger
Quantum of Solace
Pirates of the Caribbean: Dead Man's Chest
Superman Returns
The Dark Knight Rises
Harry Potter and the Half-Blood Prince
Spider-Man 3
```

Fig. 6. Slice of the highest rated predictions list

E. Final Recommendations

The top five final recommendations give to the user is presented in Fig. 7.

```
Final Recommendations =>

Alice Through the Looking Glass
Armageddon
The Adventures of Tintin
Avengers: Age of Ultron
Terminator Salvation
Southland Tales
Night at the Museum
Warcraft
The Chronicles of Narnia: Prince Caspian
Sky Captain and the World of Tomorrow

time taken:  1.3764684200286865  seconds
```

Fig. 7. The top final recommendations

F. Evaluation of results

The results of a recommendation system cannot be evaluated using the already stored data. The effectiveness of a recommendation system can only be evaluated against human behaviour. Hence, trying to predict the effectiveness of our recommendation system is a difficult task. However, we can evaluate the effectiveness of certain systems.

For Collaborative Filtering recommendation system, we can easily verify our results. This system is predicting the rating for a movie given by a certain user. As we have the data for ratings given by various users, this data can be split into testing and training data. Hence, we can evaluate the accuracy of our model. In this project, the accuracy of collaborative recommendation system is 0.88.

For the Network Recommendation System, which is a variant of the content-based recommendation system, effectiveness evaluation is a difficult task. Human factor plays a big role in the effectiveness of this system. Hence, these kind of recommendation systems are adjusted by recording the interaction of users with the content using different criterias. For example, Youtube uses a feature called Click through Rate (CTR) for their recommendation system. CTR measures the number of users clicking on a video after is has been recommended to them. [6] The system then pushes the videos with higher CTR more to the users.

G. Accuracy Comparison

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations is given in (3).

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (3)$$

The simulation values obtained for accuracy metric is shown in the following Table III and its graphical comparison is illustrated in the following Fig. 6.

TABLE. III. ACCURACY COMPARISON VALUES

Number of transaction	Accuracy	
	RS	CRS
5	75	86
10	76	89
15	77	80
20	75.5	86.5

The graphical comparison representation for the above simulation values are illustrated in the following Fig. 8.

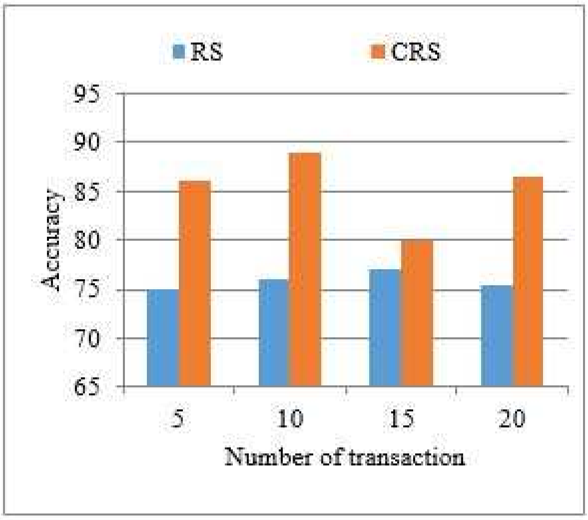

Fig. 8. Accuracy comparison

In Fig. 8, graphical comparison illustrated for the performance metrics accuracy between the proposed CRS, and the existing RS method. From this comparison it can be proved that the proposed method CRS is increased in its accuracy value which is 12% better than the RS

H. Precision Comparison

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations is described in (4). High precision relates to the low false positive rate

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

The simulation values obtained for precision metric is shown in the following Table IV and its graphical comparison is illustrated in the following Fig. 9.

TABLE. IV. PRECISION COMPARISON VALUES

Number of transaction	Precision	
	RS	CRS
5	43	74
10	43	75
15	46	61
20	43	54

The graphical comparison representation for the above simulation values are illustrated in the following Fig. 9.

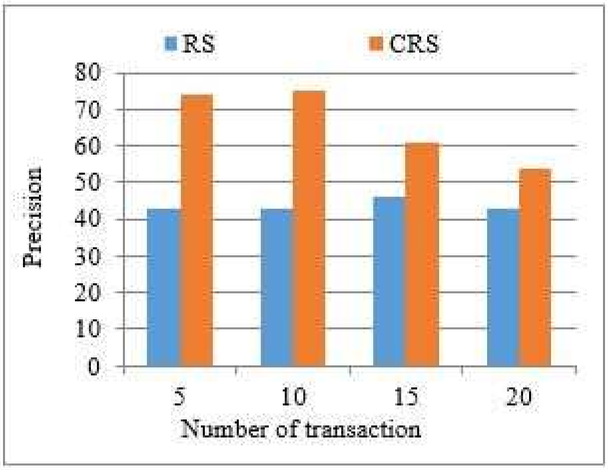

Fig. 9. Precision comparison

In Fig. 9, graphical comparison illustrated for the performance metrics precision between the proposed CRS, and the existing RS method. From this comparison it can be proved that the proposed method CRS is increased in its precision rate which is 75% better than previous work RS.

I. Run Time

The simulation values obtained for run time metric is shown in the following table V and its graphical comparison is illustrated in the following Fig. 10.

TABLE. V. RUN TIME COMPARISON VALUES

Minimum utility threshold	**Run time**	
	RS	***CRS***
1.5	1000	400
2	700	300
2.5	600	230
3	450	200

The graphical comparison representation for the above simulation values are illustrated in the following Fig. 10.

In Fig. 10, graphical comparison illustrated for the performance metrics run time between the proposed CRS method and the existing RS method. From this comparison it can be proved that the proposed method CRS is lesser run time which is 47% better than RS.

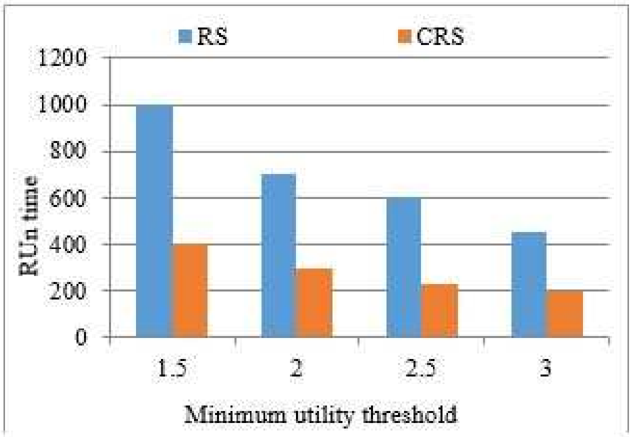

Fig. 10. Run time comparison

J. Memory Consumption Comparison

The simulation values obtained for memory consumption metric is shown in the following Table VI and its graphical comparison is illustrated in the following Fig. 11.

TABLE. VI. MEMORY CONSUMPTION COMPARISON VALUES

Number of transaction	Memory consumption	
	RS	CRS
1.5	40	36
2	38	34
2.5	34	30
3	32	28

The graphical comparison representation for the above simulation values are illustrated in the following Fig. 11.

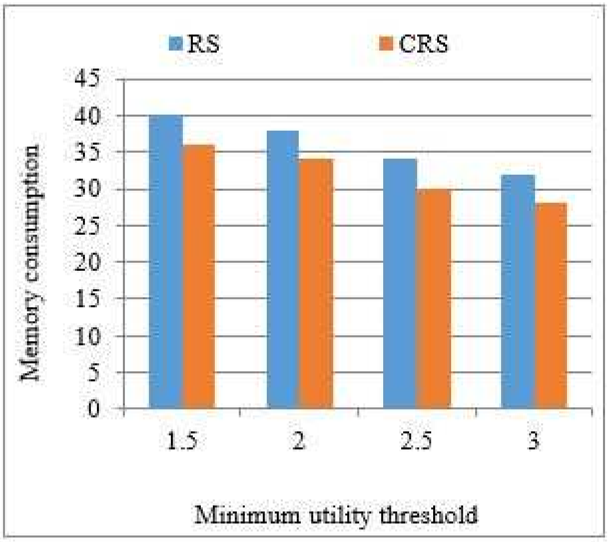

Fig. 11. Memory consumption comparison

In Fig. 11, graphical comparison illustrated for the performance metrics memory consumption between the proposed CRS and the existing RS method. From this comparison it can be proved that the proposed method CRS is increased in its F-Measure rate which is 13% better than RS.

IV. CONCLUSION

The recommendation system presented in this paper can be useful for many scenarios. The system can be modified for any kind of platform and product. For e.g., Online shopping, music streaming, etc. The system, paired with proper evaluation techniques, will be very useful for any platform to show relevant products to their customers.

The network recommendation system implemented was able to reliably predict movies with strong links to each other. The collaborative filtering recommendation system was able to predict ratings for a movie by a user with high accuracy. By combining both systems, we were able to find recommendations based on the products and the products' reviews by other users.

V. FUTURE ENHANCEMENTS

The system can be modified and implemented for other services and products. For e.g., E-retailers, Music streaming, etc.

More different types of recommendation systems can be incorporated. This will help to better refine the recommendations given to the user. Some types of recommender systems that be used alongside the systems used:

1 Demographic based filtering: Filtering products based on the demographic of the user.

2 Bayesian network model: Gets probability using naive bayes formula, of a user buying an item.

3 Sentimental product recommendation: Ranking products based on the sentiments of the user reviews.

The recommendation models suggested till now are based on the knowledge already known to the system. However, as a recommendation system needs to curate its results for a user, knowledge-based filtering techniques can be very effective. These systems can give importance to or try to avoid a certain type of product based on the feedback of the user.

Deep learning techniques like neural networks can be implemented on these kinds of systems that can actively learn about the users and their behaviour.

REFERENCES

[1] D. Bose, and M.B.D, "Analytics, Movie recommendation system with Collaborative Filtering using K-NN", 2020.

[2] Z. Fayyaz, M. Ebrahimian, D. Nawara, A. Ibrahim, and R. Kashef, "Recommendation systems: Algorithms, challenges, metrics, and business opportunités", Applied sciences, vol. 10, no. 21, pp. 7748, 2020.

[3] Parameshachari, B. D., and K. M. S. Soyjaudah. "A new approach to partial image encryption." In *Proceedings of International Conference on Advances in Computing*, pp. 1005-1010. Springer, New Delhi, 2013.

[4] S. Gupta, and M. Dave, "A Recommendation System: Trends and Future", International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no. 6S3, 2019.

[5] N. Karbhari, A. Deshmukh, and V.D. Shinde, "Recommender System in Machine Learning", International Conference on Emanations in Modern Technology and Engineering (ICEMTE-2017), vol. 5, no. 3, 2017.

[6] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations", In Proceedings of the 10th ACM conference on recommender systems, pp. 191-198, 2016.

[7] J. Beel, B. Gipp, S. Langer, and C. Breitinger, "Research-paper recommender systems: a literature survey", International Journal on Digital Libraries, vol. 17, no. 4, pp. 305–338, 2015.

[8] D. Schall, "Social Network-Based Recommender Systems", Ch - 2 Link prediction for Directed Graphs. Springer International Publishing, 2015.

[9] S. Jain, A. Grover, P.S. Thakur, and S.K. Choudhary, "Trends, problems and solutions of recommender system", In International Conférence on Computing, Communication & Automation, pp. 955-958, 2015.

[10] K.N. Kumar, and K. Vineela, "Friend Recommendation using graph mining on social media," 2010.

[11] W. McKinney, "Data structures for statistical computing in python", In Proceedings of the 9th Python in Science Conference, vol. 445, no. 1, pp. 51-56, 2010.

[12] A. Hagberg, P. Swart, and S.D. Chult, "Exploring network structure, dynamics, and function using Network (No. LA-UR-08-05495; LA-UR-08-5495)", Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 2008.

[13] BD, Parameshachari, and K. M. S. Soyjaudah. "Analysis and comparison of fully layered image encryption techniques and partial image encryption techniques." In *International Conference on Information Processing*, pp. 599-604. Springer, Berlin, Heidelberg, 2012.

[14] Z. Huang, W. Chung, T.H. Ong, and H. Chen, "A graph-based recommender system for digital library", In Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, pp. 65-73, 2002.

[15] https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset?select=credits.csv