Book Recommendation System

Himanshu Mishra[1], Ashish Asthana[2]

[1,2] *Department of Information Technology, Galgotias College of Engineering and Technology, Greater Noida, Uttar Pradesh, India*

Abstract –*A recommender system can be defined as a system that produces individual recommendations as output, based on previous decisions that the system considers to be inputs. Book recommendation systems play an important role in book search engines, digital libraries, or book shopping sites. In the field of recommender systems, processing data, choosing the right data characteristics, and how to classify them are challenges in determining the performance of recommender systems. This paper presents several solutions for data processing capabilities to build efficient book recommender systems. Book Crossing datasets examined in many book recommender systems are considered case studies. Many of the products we use today are the result of recommender systems such as music, news, books and articles. However, in this paper we would be discussing about a book recommender system on Collaborative filtering and Content based filtering while comparing the accuracy of each recommendation system.*

Key Words: **Recommender System, Collaborative Filtering, Content Based Filtering, Hybrid Filtering, Nearest Neighbor.**

1.INTRODUCTION

The Current recommendation systems such as content based filtering and collaborative filtering use different information sources to make recommendations. Content based filtering, makes recommendations based on user preferences for product features in this case the genre of the Books, the authors, the ratings etc. Collaborative filtering impersonate user-to-user recommendations as a weighted and linear combination of other user preferences. Both methods have limitations.

Content-based filtering recommends new entities by using n or more custom data to absorb the best matches. Similarly, collaborative filtering requires a large dataset of active users who have previously evaluated the product in order to make accurate predictions. The combination of these different recommender systems is called a hybrid system and allows you to combine item properties with other users' preferences. Existing services such as Goodreads personalize recommendations and use weighted averages to provide an overall rating for a book. This greatly enhances the value of each recommendation as it takes into account the user's individual book preferences. However, our recommendation engine employs a collaborative social networking approach that mixes your tastes with the wider community to produce meaningful results.

The complete RS contains three main components: user resources, article resources, and recommended algorithms. The user model analyzes consumer interests, and the item model analyzes the properties of items in a similar way. It then collates the consumer's characteristics with the item's characteristics and uses the recommended algorithm to estimate the recommended item. The performance of this algorithm affects the performance of the entire system.

In memory-based CF, book ratings are used directly to rank unknown ratings for new books. This method can be divided into two types: a user-based approach and an item-based approach.

1) User-based approach: User-Based Collaborative Filtering is a technique used to predict the items which looks for similar users that might like on the basis of ratings given to that item by the other users who have positively interacted with that item [12,11,5].

2) Item-based approach: This method determines the similarity between a group of objects specifically evaluated by the buyer and the desired object. Items that are very similar will be selected. Recommended values are calculated by taking weighted average of user ratings for the same object.

3) Researchers have combined various recommendation techniques to obtain accurate and rapid recommendations, these are called hybrid recommender systems. Some Hybrid recommendation approaches are:

- Perform separate procedures and connect results.
- Use some content filtering directives with the CF communities
- Use some CF principles in content filtering recommendations.
- Use both content and collaborative filtering in the recommender

In addition, research on semantic-based, contextual, cross-language, cross-domain, and peer-to-peer methods in progress [1].

2.LITERATURE SURVEY

A recommendation system generally depends upon the inputs of a user and their relationship between the products.

2.1 Collaborative filtering

The collaborative filtering algorithm uses user's behaviour to recommend items. They exploit the behaviour of other users and factors of tracking history, ratings, and choices. Other users' behaviour and book preferences are used to recommend booksto new users. However, it can be difficult to includesecondary functionality, i.e. functions beyond the query or ISBN. For book recommendations, extras may include genre and rating. The inclusion of available extra features improves the quality of the model.

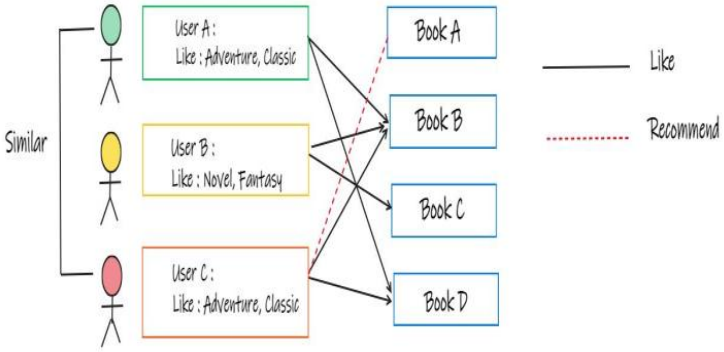

Fig-1: Example of Average weighted algorithm (Collaborative Filtering)

User A prefers book genres A, B, C, and user C prefers to read books B, D, hence we can conclude that the likings of user A and user C are very similar.Since user A likes book D as well, we can deduce that the user A may also like book D, therefore book D would be recommended to the user The general idea of the algorithm is based on a review of previous ratings provided by the users. Find the neighbor user as alpha who exhibits similar interest with target user beta, and then suggests the items which the neighbor user alpha preferred to target user beta, the predicted score which the target user amay give on the item is obtained by the score calculation of neighbor user alpha on the item.

2.1.1 Advantages of Collaborative Filtering

Memory-based collaborative filtering technology makes easier implementation of the recommender systems.

1) A memory-based collaborative filtering techniquethat allows you to add new data easily and in stages[10].

2) Prediction performance gets improved when using Model-Based Collaborative filteringtechniques.

2.1.2 Limitations of Collaborative Filtering

1) Cold Start problem: Collaborative filteringsystems often require a huge amount of existing data on which user can make exact recommendations [2].

2)Scalability: Collaborative filtering makes recommendations for various environments where billions of products and users exist. Hence, a huge amount of computation power is rather essential to compute recommendations.

3) Sparsity: On major websites like amazon or flipkart the number of items sold are enormously large. Because of that reason only a small subset of the entire database is rated by most active users. Therefore, very few ratings are given to the most popular items [3].

2.2 Nearest Neighbor

The most widely used algorithm for collaborative filtering is the k Nearest Neighbors (kNN) [6, 5, 4]. It is very simple algorithm that stores all the available cases and classifies the new data or case based on a similarity measure. It was first introduced in the GroupLens Usenet article recommender [13].

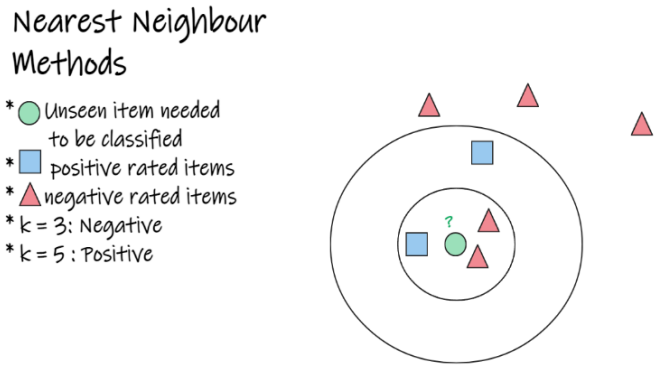

Fig-2: Example of Nearest Neighbor algorithm

There are two types of Nearest Neighbor algorithms:

1) User based Nearest Neighbor

2) Item based Nearest Neighbor.

2.2.1 User Based Nearest Neighbor

In user based nearest neighbor or user to user version, kNN executes three tasks to generate recommendation for a user:

(a) By using a selected similarity measure we can produce a set of k nearest neighbors for the active user alpha. The k neighbors for alpha are the nearest k, similar to the user beta [10].

(b) Once a set of k users (neighbors) similar to active user alpha has been calculated, one of the average, weighted sum, and adjusted weighted aggregation (deviation-from-mean) to get the prediction of item i for user alpha.

(c) To get the top n recommendation, we choose n items which provide most satisfaction to the active user.

2.2.2 Item Based Nearest Neighbor

The increase in the number of users increase the scalability problems in User to user based kNN. To overcome this drawback new method is introduced Item to item kNN, it is introduced by Sarwar et al.[7] and karypis. This item based nearest neighbor investigates the set of items rated by target users and calculates their similarity with the target item i and then chooses k most similar items {i1, i2, i3…., ik} and the representing similarities_{si1, si2, si3,……, sik} are also calculated at the same time. The most similar items are discovered early and then predictions are calculated by taking a weighted average of the target user's ratings for those similar items. Similarity computation and prediction generation are two key factors that make item-based recommendation more powerful. Different types of weighted sum, similarity measures and regression used for prediction computation for similarity computation.

2.3 Content Based Filtering

Content-based filtering uses books characteristic to recommend books that are similar to what users like,based on their previous actions or explicit comments. This makes scaling easier for a large number of users. The model can capture a user's specific interests and based on that, can recommendniche books that few other users are interested in. However, since the characteristic representation of elements is designed by hand in some respects, this methodology requires a lot of domains. That's why, this model can only be as good as manual features. The model can only recommend anything based on already existing user preferences. In general terms, the model has a limited ability to evolve existing user preferences [8]. When we talk about book recommender system, content based recommender system can recommend books based on explicitly provided user data, then user profile is created. This background knowledge is then used to make recommendations that become more accurate over time. In a content-based system, the concepts of (TF- IDF) term frequency and inverse document frequency are used for information retrieval and filtering systems. The prime use of these terms is to acquire the importance of any book. Term frequencycan be described as the number of times or the frequency of the word in a document.

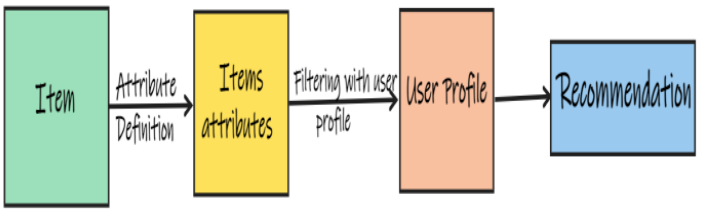

Fig-3: Example of Content Based Filtering algorithm

2.3.1 Advantages of Content Based Filtering

1) CBF recommender system provide user independence through exclusive ratings which are used by the active user to build their own profile.

2) CBF recommender system provide transparency to their active user by giving explanation how recommender system works.

3) CBF recommender system are adequate to recommend items not yet placed by any user. This will be a benefit for new user.

Here comes the most crucial step for your research publication. Ensure the drafted journal is critically reviewed by your peers or any subject matter experts. Always try to get maximum review comments even if you are well confident about your paper.

2.3.2 Limitations of Content Based Filtering

1) It is difficult to create item characteristics in some areas.

2) CBF recommend the same types of items because of that it suffers from an overspecialization problem.

3) It is harder to get feedback from users in CBF because users do not typically rank the items as compared to CF and hence, it is not possible to determine whether the recommendation is correct.

2.3 Hybrid Filtering

Hybrid based recommender system proved to be more effective in some cases. Basically content based filtering and Collaborative based filtering approaches are used more extensively in information filtering application. The main objective of hybrid approach is to aggregate content based and collaborative based filtering to improve recommendation accuracy. Hybrid approach can be implemented as follows:

I) Implement content and collaborative based methods separately and then aggregate their prediction.

2) Include some content based attributes into a collaborative approach,

3) Integrate some collaborative attributes into a content based approach

4) Build a general integration model to integrate both content-based and collaborative based features.

Sparsity and cold start are common problems in recommender system which are of some degree is resolved by using these methods. Good example of hybrid recommender system in amazon, they make recommendations by comparing the exploring habits of similar users (collaborative filtering) as well as by providing items that share features with items that a user has rated highly (content-based filtering). Some organizations, like Facebook, use this hybrid filtering method to display messages that are important to you and others in your network, and the same is used for LinkedIn.

CBF and CF can be accumulated in different ways [6].Below given figures shows the different choices for accumulating CB and CBF.

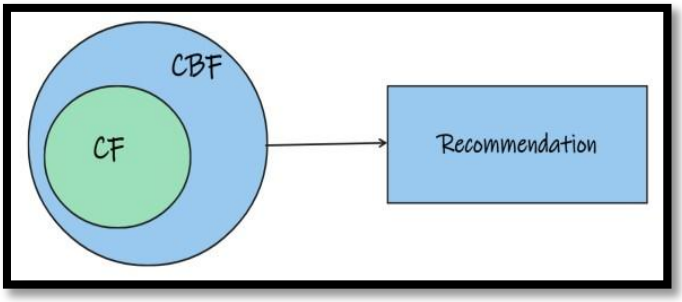

Fig-4: Shows the methods that estimate CBF and CF recommendations individually and afterwards combine them to yield better recommendations.

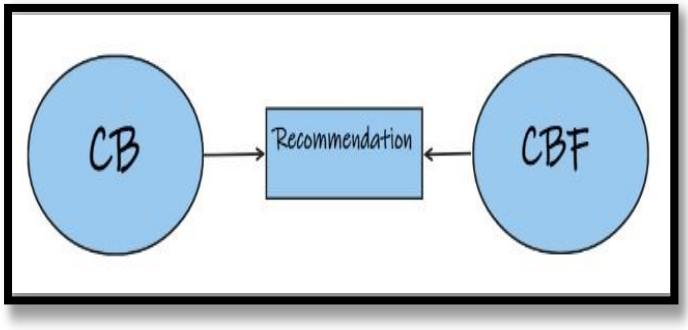

Fig-5: shows the methods that integrate CBF attributes into the CF approach, so that it will overcome the cold start problem in collaborativefiltering of content-based filtering.

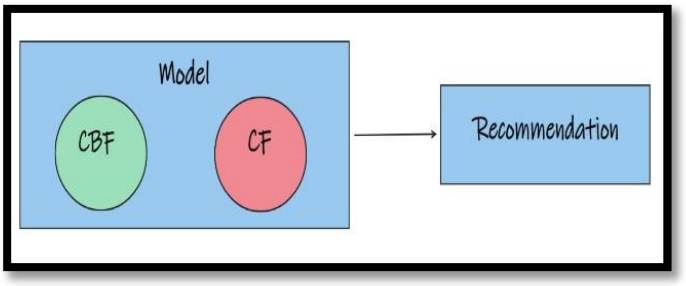

Fig-6: shows the methods for constructing a unified utility system with both CF and CBF attributes. In this method by combining somefeatures of CF and CBF one unified model is constructed that can improve performance of recommendation process.

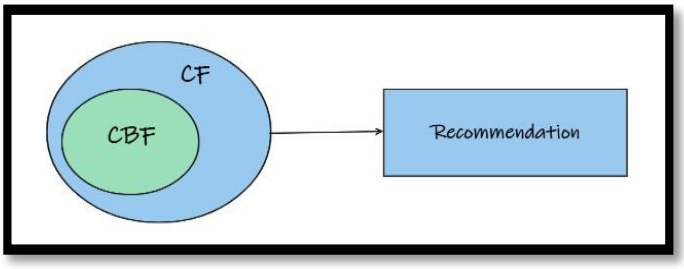

Fig-7: It shows the methods that incorporate CF attributes into a CBF approach

3. ARCHITECHTURE AND WORKFLOW

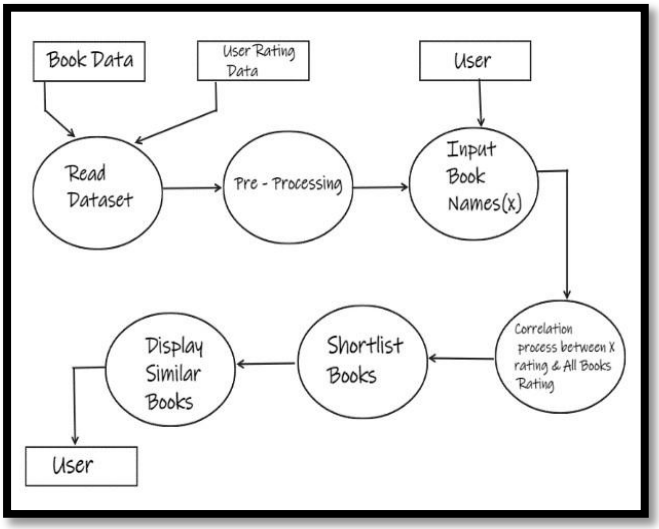

Fig-8: Architecture Diagram for BookRecommender System

4. ALGORITHM AND METHEDOLOGY

In a book recommender system, a book is that entity which is considered and recommendation is done using similar entities. Based on the user's query, you can make recommendations using the most relevant similar entities from a large number of datasets. Books are rated, which helps in retrieving other entities that are more relevant

based on popularity, relevance, etc. Below stating output, input and data of the recommender and problem.

4.1 Equation:

4.1.1 Average Weighted Rating Formula:

$$W = \frac{Rv + Cm}{v + m}$$

Where:

W = Weighted Rating

R = average for the book as a number from 0 to 10 (mean) = (Rating).

v = Number of votes for the books = (votes)

m = minimum votes required to be listed in the Top 200

C = the mean vote across the whole report

The above formula is a variation of the formula used by Goodreads to rate its Book lists. It's an effective formula that involves the consideration of a user's perspective to provide a clear recommendation thereafter.

4.2 Algorithm:

1) KNN is a supervised machine learning algorithmused for solving classification and regression problems. The K stands for the number of nearest neighbours. In this algorithm the number nearest neighbours to an unknown variable that needs prediction or classification is represented by k.

2) Algorithm tries to locate the closest neighbour around a new unknown data point in order to figure out which category the unknown data point belongs to. It assumed that the nearest neighbour to and by unknown data point possess the same characteristics.

3) Algorithm calculates the distance between the all the data point in the space around the unknown data point which are closest to the unknown data point and then tries to predict the similar data set.

4) In our book recommendation system we used this algorithm to find out the cluster of similar user based on common book ratings and make predictions based the top value of k-nearest neighbour.

5) In this model only those user have been considered only those rated minimum 200 books, this is done to make the recommendation highly relevant. The book ratings have been presented in a matrix with matrix having one row for each book and one column for each reader (user) and a pivot table is created.

6) The pivot table of the user rating is converted intoa sparse matrix meaning the missing values are filled with a 0 this is done because the distance between the rating vectors will be calculated. The sparse matrix makes the computation fast and calculation efficient.

7) To train the Nearest Neighbours model, we have created a compressed sparse row matrix taking ratings of each Book by each User individually. This matrix is used to train the Nearest Neighbours model and then to find n nearest neighbors using the cosinesimilarity metric.

5. TOOLS USED:

- Anaconda Navigator, Jupyter Notebook, PyCharm, Python 3.9
- Data cleaning by using Python libraries like NumPy and Pandas.
- Data and Output Visualization by Streamlit.
- Python Libraries like pickle are used to take data as cache and provide Streamlit with instantaneous result as the recommendations are provided by the backend.
- We have tested model using simulation of the following requirements.
- System with minimum requirement of 8 GB RAM and intel i5 core processor.

6. IMPLEMENTATION OF MODULE WITH CODE:

6.1 MODULE 1:

Under this module we are importing python librariesto Book Crossing dataset [9] has shown. The given book dataset is provided by Institut für Informatik,Universität Freiburg.

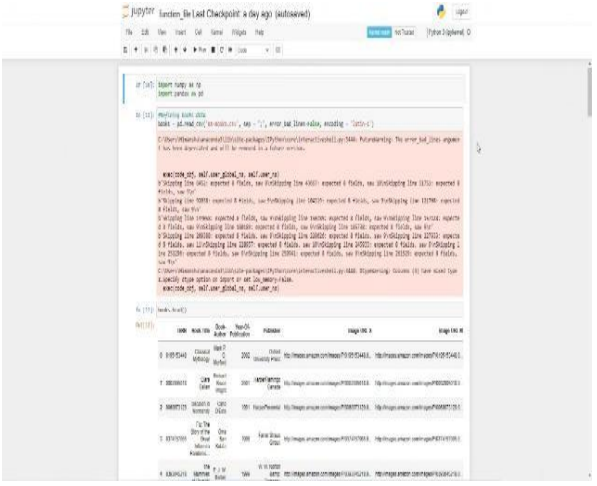

Fig-9

6.2 MODULE 2:

Data cleaning module comprises of the removal of those columns that are not required for calculation of our recommendations. In this module, as we can see the factors like Image-URL-M, Image-URL-L, Image-URL-S do not play any role nor give any idea about the interests of our users hence we have dropped such columns. Therefore we have separated those columns of use and merged with user's database which consists of columns like user-ID, Location, Age. The code is given below:

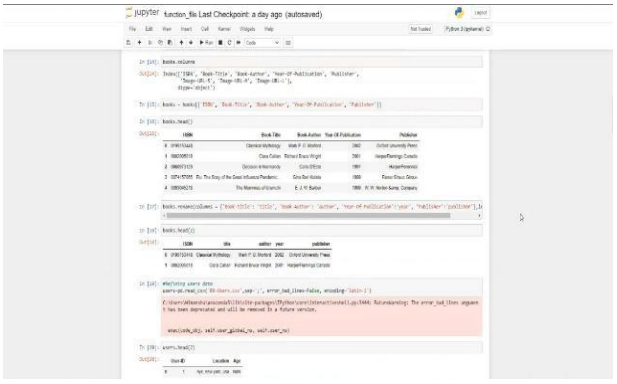

Fig-10

6.3 MODULE 3:

In this module we are cleaning book rating dataset which have duplicates and redundant data. We are cleaning the data by taking data of only users who have rated minimum of 200 books.

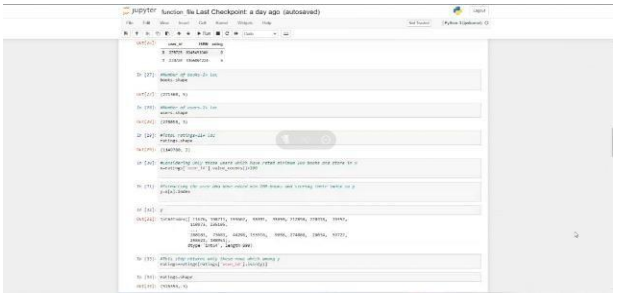

Fig-11

6.4 MODULE 4:

Merging those dataset and then we created pivot table between user-ID, index and ratings. It is also shown that using sparse matrix, by using it we can avoid computing time as the value with 0 is not considered and the remaining data creates a pivot table. Also, by using cosine similarity we are going to find the similarity in tastes of users by which our model get to know how similar one user's taste to another user is.

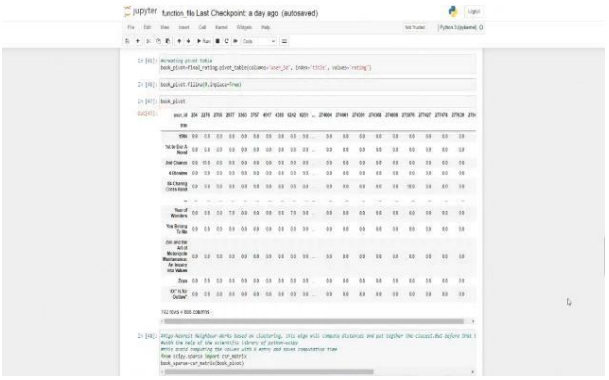

Fig-12

6.5 MODULE 5:

In this module we are training the model on nearest neighbour algorithm and when the model is trained, if asked the query by giving a name of a book, recommender system provides us with the names of recommended books.

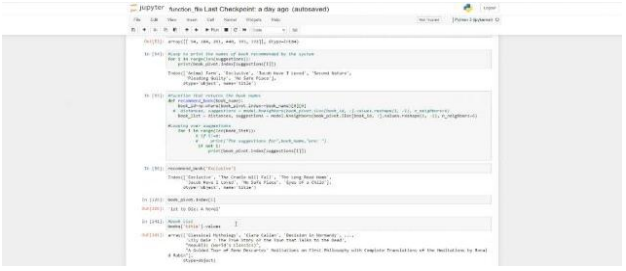

Fig-13

6.6 MODULE 6:

Under this module we are showing the recommendation produced by the model using visualization library Streamlit of Python.

Fig-14

7. CONCLUSION:

Under the condition of massive information availability, the requirements of Book recommendation system from book amateur are increasing. This article designs and implements a complete book recommendation system prototype based on the Nearest Neighbour classification, collaborative filtering algorithm and recommendation system technology.

8. REFRENCES:

[1] P.K. Singh, P.K.D. Pramanik, A.K. Dey and P. Choudhury, Recommender systems: an overview, research trends, and future directions, Int. J.Business Syst.Res. 15(1) (2021) 14–52.

[2] K. Heung-Nam, E.S. Abdulmotaleb, J. Geun- Sik, "Collaborative error-reflected models for cold-start recommender systems", Decision Support Systems 51 (3) (2011), pp. 519–531

[3] J. Bobadilla, F. Serradilla, "The effect of sparsity on collaborative filtering metrics", in: Australian Database Conference, 2009, pp. 9– 17.

[4] J. Bobadilla, A. Hernando, F. Ortega, J. Bernal, "A framework for collaborative filtering recommender systems", Expert Systems with Applications 38 (12) (2011) 14609–14623.

[5] J. B. Schafer, D. Frankowski, J. Herlocker, S.Sen,"Collaborative filtering recommender systems", in: P.Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web, 2007, pp. 291–324

[6] Adomavicius, G.; Tuzhilin, A., "Toward the next generation of recommender systems: a survey of the state of- the-art and possible extensions," Knowledge and Data Engineering, IEEE Transactions on , vol.17, no.6, pp.734,749, June 2005.

[7] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in ACM www "01, pp. 285–295, ACM, 2001.

[8] C. Basu, H. Hirsh, W. Cohen, "Recommendation as classification: using social and content-based information in recommendation", in: Proceedings of the Fifteenth National Conference on Artificial Intelligence, 1998, pp. 714–720.

[9] Book Crossing Dataset by apl. Prof. Dr. Cai- Nicolas Ziegler http://www.informatik.uni-freiburg.de/~cziegler/BX

[10] Thorat, Poonam B., R. M. Goudar, and Sunita Barve. "Survey on collaborative filtering, content-based filtering and hybrid recommendation system." International Journal of Computer Applications 110, no. 4 (2015): 31-36.

[11] M. Balabanovic, Y. Shoham, "Content-based, collaborative recommendation", Communications of the ACM 40 (3) (1997) pp.66–72.

[12] M. Pazzani, "A framework for collaborative, content based, and demographic filtering", Artificial Intelligence Review-Special Issue on Data Mining on the Internet 13 (5-6) (1999) pp.393–408.

[13] Konstan, Joseph & Miller, Bradley & Maltz, David & Herlocker, Jon & Gordon, Lee & Riedl, John. (2000). GroupLens: Applying collaborative filtering to Usenet news. Communications of the ACM. 40. 10.1145/245108.245126.