**INDEX:**

**Abstract 2**

**Introduction 2**

**Dataset 2**

**Decision Parameters 2**

**Model 3**

**Popularity 3**

**Content based filtering 3**

**Collaborative based filtering 4**

**Hybrid model 4**

**Training and testing 4**

**Final Recommendation 5**

**Conclusion 7**

**Future Extension 7**

**References 7**

**Abstract:**

Recommender systems are highly used to recommend items to the end users that are most suitable. Online book vending websites presently are struggling with each other in a lot of means. Recommendation system is one of the stronger tools to enlarge yield and sustain the presence of buyer. The book recommendation system must recommend books that are of buyer’s interest. This project and report presents book recommendation system based on homogenized features of popularity, content filtering, collaborative filtering.

Keywords: Association rule, Collaborative filtering, Content based filtering,Recommendation system.

**Introduction:**

Recommendation systems were progressed as insightful algorithms, which can produce outputs in the form of recommendations to users. They minimize the elevated associated with making good choices among the plenty. Now, Recommender systems can be used in any domain from E-commerce to network. They give advantage to both the user and the manufacturer, by suggesting items to consumers, which can’t be requested until the recommendations. Every recommender system consists of two components, one is user and other is item. A user can be any product or items, who get the recommendations. Database is given as input to recommendation algorithm to user and items and output evidently will be the recommendations. As in our case, inputs from database consist users and books and results gives the book recommendations. This paper presents a new strategy for recommending books to the buyers. This system combines the characteristics of Popularity model, collaborative filtering and content filtering to generate efficient and effective recommendations.

**Dataset:**

4datasets:

FinalData: book\_id, authors, title, genres 1000 data items

AverageRatings: book\_id, rating 10000 data items

RatingCount: book\_id, rating 10000 data items

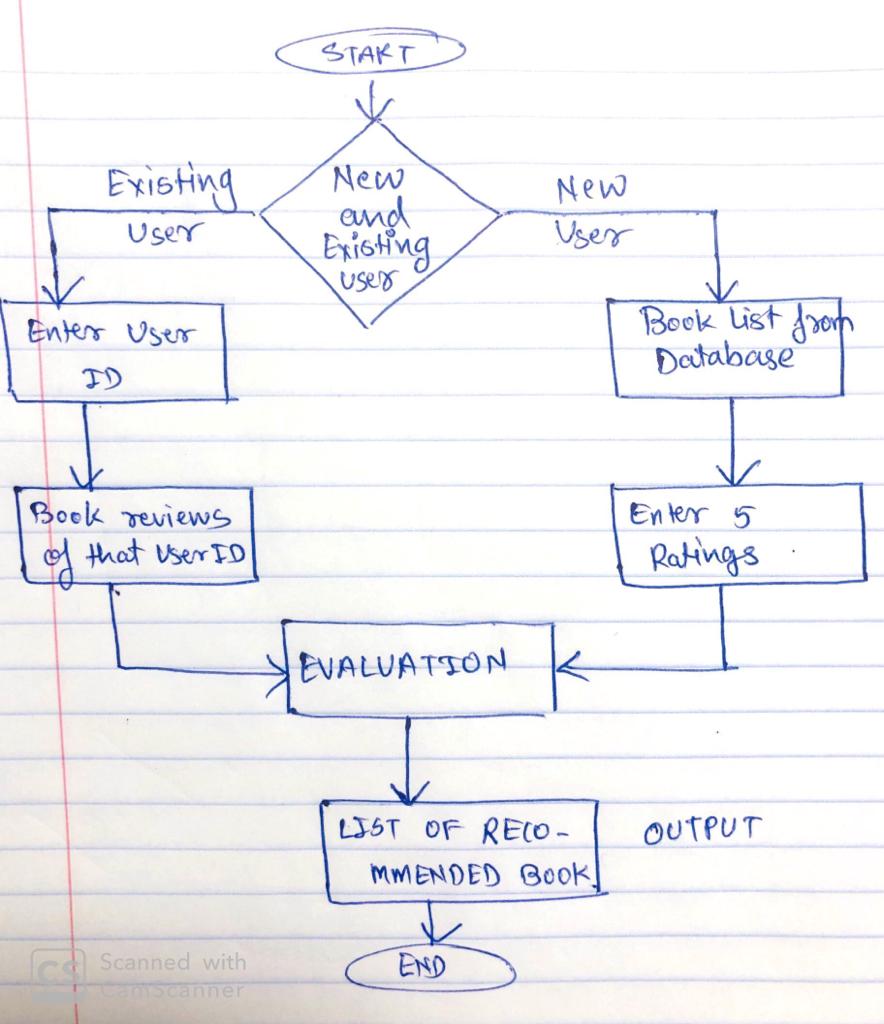
Ratings: user\_id, book\_id, rating 1048000 data items

**Decision Parameters:**

The genres included are: 'Fantasy', 'Fiction', 'Self-Help', 'Drama', 'Romance', 'Thriller', 'Biography', 'Erotic', 'Kids', 'Poetry', 'Horror', 'History', 'Academic', 'Comedy', 'Classic', ' Thriller', 'Domestic Fiction’, ‘SciFi’, ‘Crime Fiction’, ‘Psychological Fiction’, ‘Young-Adult Fiction’.

The final data contains book-id, authors, title and genres for 1000 books.

**Model:**



**Popularity model:**

The fundamental concept behind this recommender is that books that are more rated and more cruicially reviewd will have a greater probability of being read by the average users. Popularity model does not provide personalized recommendations based on the user. From the database of average ratings, ratings matrix and rating count for each book is measured. Mathematically, it is represented as follows:

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

where,

v is the number of votes for the book

m is the minimum votes required to be listed in the chart

R is the average rating of the book

C is the mean vote across the whole report

**Content Based Filtering:**

The model above is not really personal in that it doesn't recognize the personal interests of users and differences of a user. Anyone arguing this model for recommendations based on a book will get the similar recommendations for that books.

“The Content based Recommenders are built using:

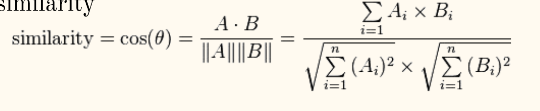
1. Authors

2. Genres

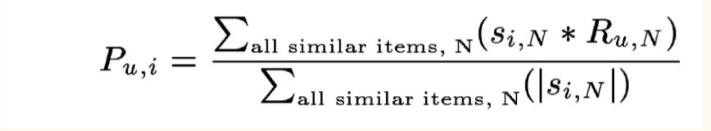
We have given twice the weigth to author of the book. Cosine Similarity is used to calculate a numeric quantity that denotes the similarity between two books. Mathematically, it is defined as follows:

cosine(x,y) = x.y⊺/ (||x||.||y||)”

Cosine Similarity:



Prediction:



**Collaborative based Filtering:**

The content based model is only able of recommending books which are close to a specific book. That is, it is not capable of capturing interests and suggesting recommendations throughout genres. Therefore, we have used Collaborative Filtering to make recommendations for book. Collaborative Filtering is based on the idea that users which are similar to a me can be used to predict how well I will be liking a particular product or service to those users which have used/experienced but I have not. Surprise library has been used that uses powerful algorithms like Singular Value Decomposition (SVD) to provide minimal RMSE (Root Mean Square Error) and give better recommendations. We get a RMSE of 0.8840.

**Hybrid model:**

I have merged the ratings from Popularity model, Collaborative based filtering and Content filtering to get more explicit output. It gives the predicted rating as weighted fusion of the above described techniques. Equal weigths have been given to collaborative and content rating.

Rhybrid = (1-2α)\* Rpopularity + α\* Rcollaborative + α\* Rcontent

α = 0.4

**Training and Testing:**

For training and testing purpose, 80:20 proportion of data was split into.

80% for training and

20% for testing.

I have used the RMSE technique to calculate accuracy of the predicted ratings by hybrid recommendation system in order with the actual rating given by user to a particular book. For the hybrid model, RMSE of 0.6960 is obtained which is eventually better than any other technique.

Ratings:3100000 Users:53000 Books:1000

Training Algorithms:

For Collaborative: Stochastic Gradient Descent

For Content Based: Countvectorizer’s output used for cosine similarity

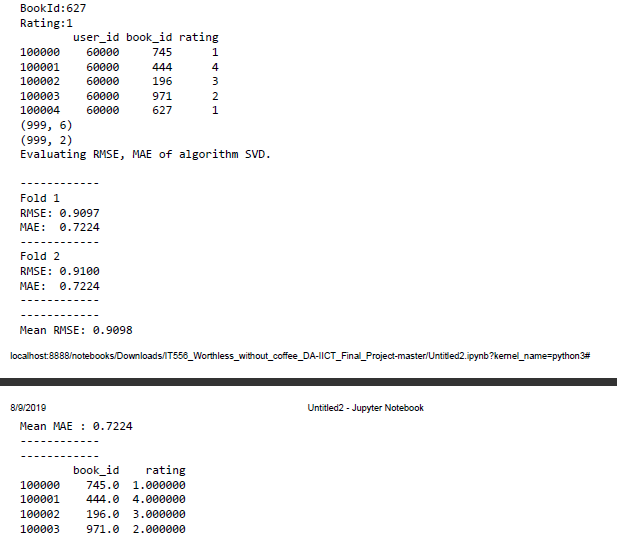
For hybrid: SGD can be used to minimize RMSE and find optimal value of a.

Accuracy metric: RMSE

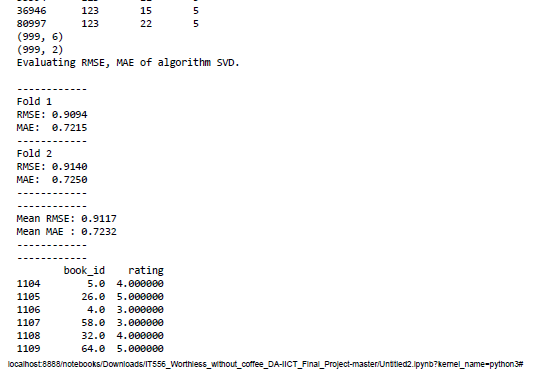
**Final Recommendation:**

Final result or output of the system looks as follows:

Output when User is new:



**Output when User is Existing:**



**Conclusion:**

The grail of the many of the recommendation systems is to anticipate the buyer’s engrossment and recommends the books accordingly. This book recommendation system has taken into consideration many features like content of the book and caliber of the book by doing collaborative filtering of ratings by the different buyers. This hybrid book recommender system gives stronger recommendations. There is no performance problem because it works offline. Consequently, almost all of the recommender systems are evaluated and marked based on one of the strategies mentioned above. But these recommender systems do not provide the accurate efficiency which is the most critical prospect of recommender systems. Inaccurate results can experience significant loss of resources. Some ways to engross accuracy like:

### “-Get more data about user.

### -Use a better algorithm or switch to another technique.

### -Train models using different techniques and ensemble them together to improve results”

As my dataset is static and I already have implemented hybrid technique for predictions, so I chose to amalgamate models trained distinctly and output cumulative predictions. The four models that I have used are:

### -Popularity based model

### -Collaborative Filtering model

### -Content based filtering model

### -Hybrid Model

**Future Extension:**

If had given more time for execution, I would have used Stochastic Gradient Descent to absorb the weights given to all of the ratings of popularity model, Content based model and collaborative model.

**References:**

[1] http://www.ijirst.org/articles/IJIRSTV1I11135.pdf

[2] <https://www.machinelearningplus.com/nlp/cosine-similarity/>

[3] https://en.wikipedia.org/wiki/Collaborative\_filtering

[4]FOLTZ, P. W. AND DUMAIS, S. T. 1992.Personalized information delivery: an analysis of information filtering methods. Comm. ACM35(12), pp. 51–60.

[5] http://mathworld.wolfram.com/SingularValueDecomposition.html

[6]Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl. J., 1994. GroupLens: An Open Architecture for Collaborative Filtering of Netnews, Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work, Chapel Hill, NC, pp.175-186.

[7]SARWAR, B., KARYPIS, G., KONSTAN, J., AND REIDL, J. 2001. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web (WWW’01). ACM, New York, NY, pp.285–295.

[8]J. Han, M. Kamber, Data Mining: Concepts and Techniques, The Morgan Kaufmann Series, 2001.

[9]Agrawal, R., Imielinski, T., Swami, A. N. "Mining association rules between sets of items in large databases". In Proceedingsof the 1993 ACM SIGMOD International Conference on Management of Data, 207216, 1993.