A comparative study of collaborative recommendation systems

Abstract:

Recommendation systems are considered integral part of any online portal today. Use of recommender system is widespread in online shopping today. In this this paper few important recommender systems have been studied and compared. That has been followed by number of experiments to do a comparative study of different recommender systems. User experience is one important criteria for success of a recommendation system. A good recommendation system will enhance customer engagement and experience by providing relevant recommendation thus helping in navigation or selection. In contrary irrelevant information can led to user frustration. In this paper another objective is to come up with a book recommender system that has shown best result in experiment.

Keywords: User-based, Item-based, Collaborative Filtering, SVD, Recommender System

1. Introduction:

Personalization of information displayed to Customer has become one of the critical aspect of doing online business. Successful ecommerce site needs to understand Customer and recommend product and services that suits Customer taste and needs. One such application, book recommender systems has been very helpful for users. Recommender filters through lot of information and save lot of time, effort to find right book user will enjoy. Recommendation systems are considered important research area as they play very important role in current and future state of the information systems.
Using historical purchases made by Customer or customer rating, recommendation systems apply various quantitative methods to display list of related product and services to Customer. Measuring performance of the recommender system is also important to understand effectiveness of the system and if the method is the best one for given scenario. Such study on performance of recommender system has been one evolving area.

As part of this project objective is to design a book recommender system based on different recommendation techniques. Then carry out a comparative study on performance. This study will help to create a framework that can be used to design effective book recommender system. Also, this study can be extended to similar areas.

Key objectives that will drive the study are following:

1. Study of various recommendation systems
2. Comparative study of different types of Collaborative recommender systems
3. Comparative study and evaluation of three different methods

1.1 RESEARCH OBJECTIVE:

1. Research objective of this project was to do comparative study off the different recommendation systems and come up with a best recommendation.
2. Design reusable framework reusable for further study and enhancement.

1.2 METHODOLOGY:

Following study has been three key sections- a. Theoretical study b. Conduct experiments c. Comparative analysis and evaluation d. Recommend

Theoretical study: Looking into the problem from prospective of research already done and identifying the approach for comparative study

Conduct experiment: This section starts with doing data collection from source and data clean up. Data exploration is part of this process. Data is transformed to conduct the experiments. Data is split into train and test set and then various collaborative models are applied to conduct different types of experiments.

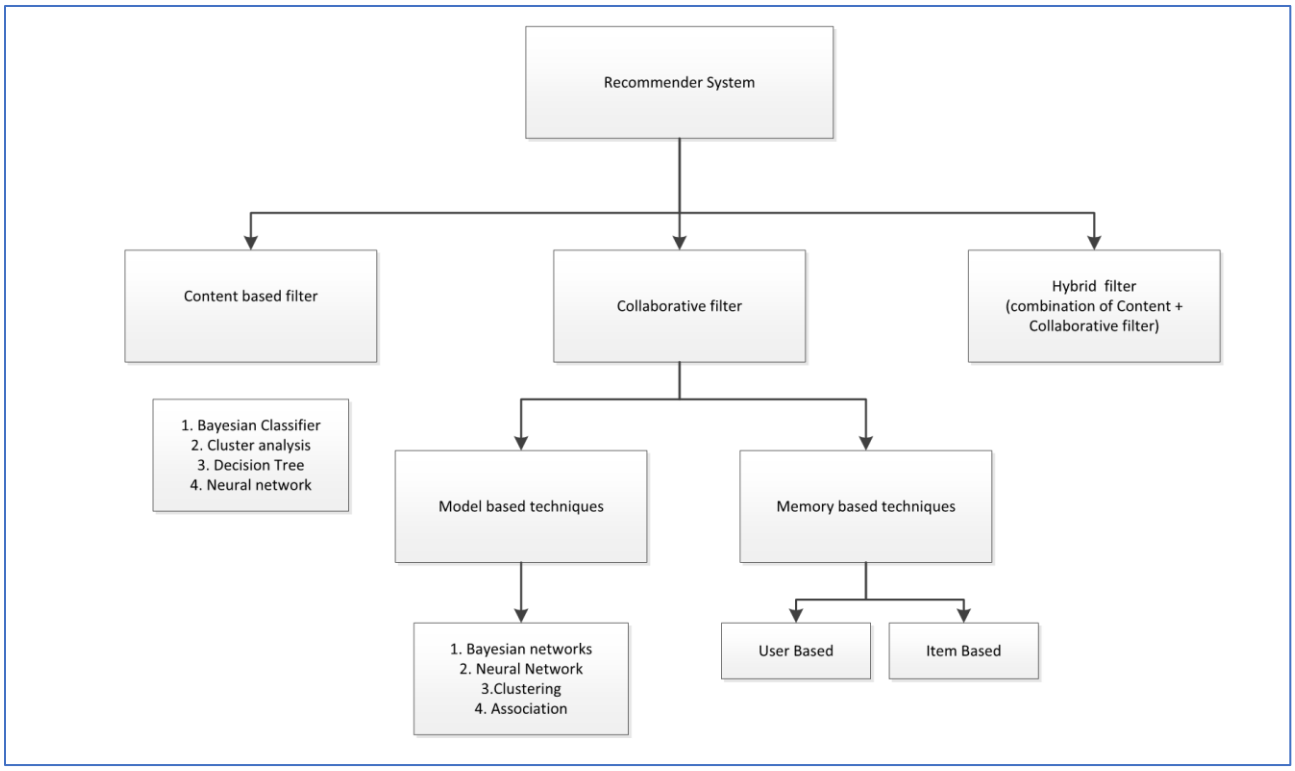
Comparative analysis and evaluation: Using outcomes from different experiments results (RMSE) are compared. Based on performance best model was selected

Recommend: Best model selected above will be used to recommend the top 10 books

2. Theoretical study:

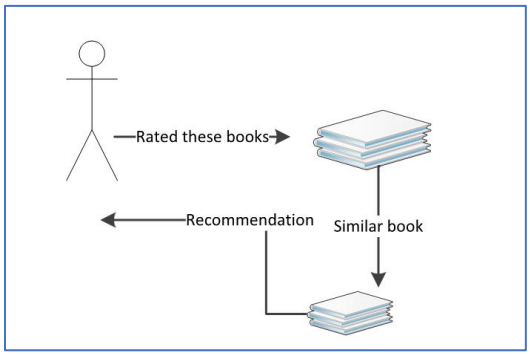
2.1 Exploration on Different types of recommendation systems:

Below diagram depicts the different types of recommendation systems in use today-



Content Based filtering technique:

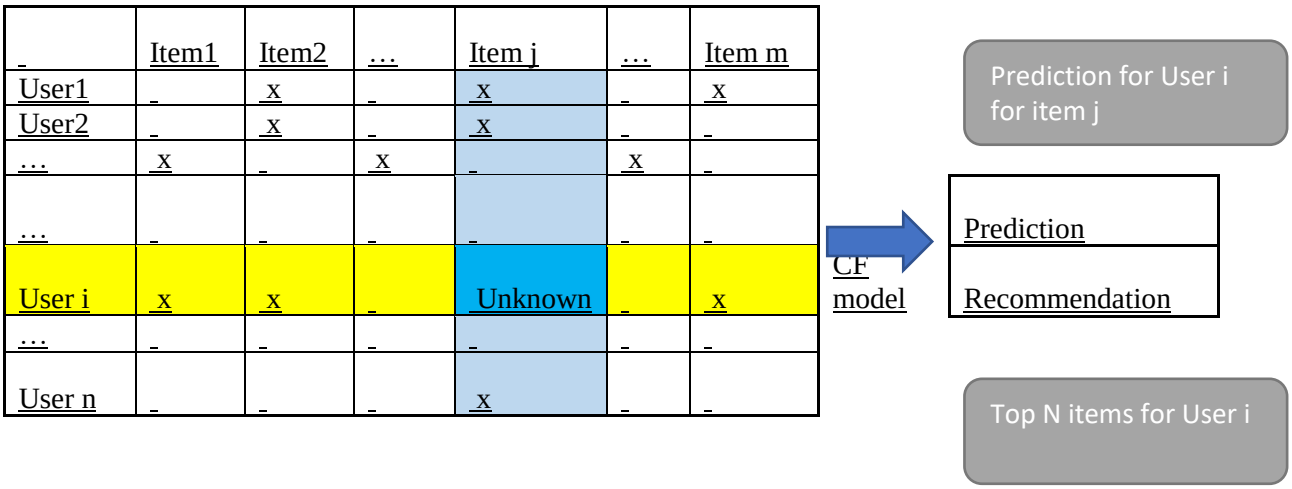
Content based recommendation system uses Individual user data (like previous ratings of book) and attributes of books to come up with recommendation for the user. Different profiling techniques are used such as Bayesian classifier, cluster analysis, decision trees, artificial neural networks, etc.



Pros and cons:
This type of recommendation system does not need prior User data. System can recommend books to user solely based on book attributes or user preferences shared by user. Recommendation engine is not influenced by user bias such as ratings scale used by different users. On flip side system does not learn over the period and system will recommend same set of books for same input.

Collaborative filtering:

Collaborative filtering (CF) works based on historical user ratings. Based on existing user ratings on books one matrix is designed keeping User and Book in two dimensions(shown below). In this approach similarity between one User to other Users calculated that is called neighborhood. User then gets recommendation for those books that he has not rated but has been rated by other users in same neighborhood. This process is mainly divided into memory based and model based techniques.



Memory based approach:

In this approach the recommendation engine depends on similarity measures (Cosine similarity, Pearson correlation etc.) between two Users. This approach leverages User and item matrix to come up with nearest neighbors by using similarity measures and doing prediction for unknown combination between User and item.

Model based approach:

In this approach algorithm is used to predict unknown ratings of items. This is a probabilistic approach to compute expected value of User-item combination. There are different machine learning algorithms that are used for this purpose like- Bayesian Network, Rule- based approach. Bayesian approach treats this as a probabilistic approach. Clustering approach treats this as classification approach by clustering user in same group and probability within that group. Association rule is used to find association between items that User has purchased to other items. Some examples of model-based techniques include Dimensionality Reduction technique such as Singular Value Decomposition (SVD), Matrix Completion Technique, Latent Semantic methods, and Regression and Clustering.

Hybrid filtering techniques:

In this approach different recommendation techniques are used. This is to avoid some bias or inaccuracies tied to the content or collaborative approach. Independent results from content and collaborative methods can be tied together to create a hybrid recommendation engine. Different approaches are used to combine the results from

content and collaborative approaches. One such approach is giving weight to each method and then come up with a weighted score. Another option is cascading down approach. Based on situation one approach gets priority and then other one is applied before final recommendations are given.

2.2 Advantage and disadvantage:

CF Type	Advantage	Shortcoming
Memory Based	Implementation is easy o new data can be added easily o Use knowledge of other users to recommend items to user	Depends on user's ratings o performance issue for sparse data scenario o startup issue for new users and items o Limited scalability for large datasets
Model Based	Model takes into consideration sparsity, scalability and other problems o Predictions performance is improved o Easy to understand for recommendation	Expensive model building o For dimensionality reduction technique useful information is lost
Hybrid	This model is used to prevail over limitations other recommenders o Better forecast performance o Can handle problem like sparsity and cold start	Implementation is costly and increased complexity

2.3 Similarity Measures:

Several types of similarity measures are used to compute similarity between Users. The two most popular similarity measures are correlation-based and cosine-based.

Pearson Correlation coefficient

Pearson Correlation coefficient is one widely used similarity measure. This score quantifies how two objects are fit in a line. Its value varies between -1 to +1. 1 indicates perfect correlation and -1 indicates objects are not correlated.

$$Pearson(x,y) = \frac{\sum xy - \frac{\sum x \sum y}{N}}{\sqrt{(\sum x^2 - \frac{(\sum x)^2}{N})(\sum y^2 - \frac{(\sum y)^2}{N})}}$$

In this equation x , y refers to the to users and N total number of items

Euclidean Distance Similarity

This method based on distance between Users. Distance is calculated based on matrix that has User ratings for each item. This metric calculates Euclidean distance between 2 users by taking into considerations ratings for each item. Distance is small when users have similarity and increases as the similarity between users fades.

$$E(x,y) = \sqrt{\sum_{i=0}^{n}(x_i - y_i)^2}$$

In this equation x ,y refers to the to users and n total number of items

Cosine similarity

This measure is little different than above two and measures the angle between User vector. In this method Individual users are treated as vectors and ratings given by user are coordinates of those vectors. Similarity between two users are determined by angle between the user vectors. Values can change from -1 to +1. 1 being User are very similar and -1 being User are completely dissimilar

$$similarity(x,y) = cos(\theta) = \frac{x \cdot y}{||x|| * ||y||}$$

In this equation x, y refers to the to users

Cosine method for similarity will be used in this study. The reason is range of rating is integer between 1 to 10 and large number of user has only rated few number of books.

2.4 Research on Current book recommendation systems-

There are some sites out in the internet which provides recommendation for books. Some of the sites which are popular are following-

Whichbook.net site provides user recommendation on books based on information provided by user. The site provides recommendation based on attributes stored for book and content of the book. Based on user mood site can recommends books. This type of recommender system tends to provide almost similar book over time if search criteria remain same.

WhatshouldIreadnext.com is another site that uses collaborative filtering based on a reference list in its data base. User can type in Author name or title of the book. Based on existing preference list site recommends books. Performance of this system is based on the preference list that is stored and that needs constant update for the system to perform better.

Site **librarything.com** uses information of its online users and this is one of the first site that has used online user community feedback to recommend books. This site calls itself world's largest book club. It is a great user-powered book ratings, review, and recommendation site.

One of the best recommendation system for book is **Amazon.com**. ecommerce giant uses item to item collaborative filtering techniques to provide recommendation. Site uses user information captured directly

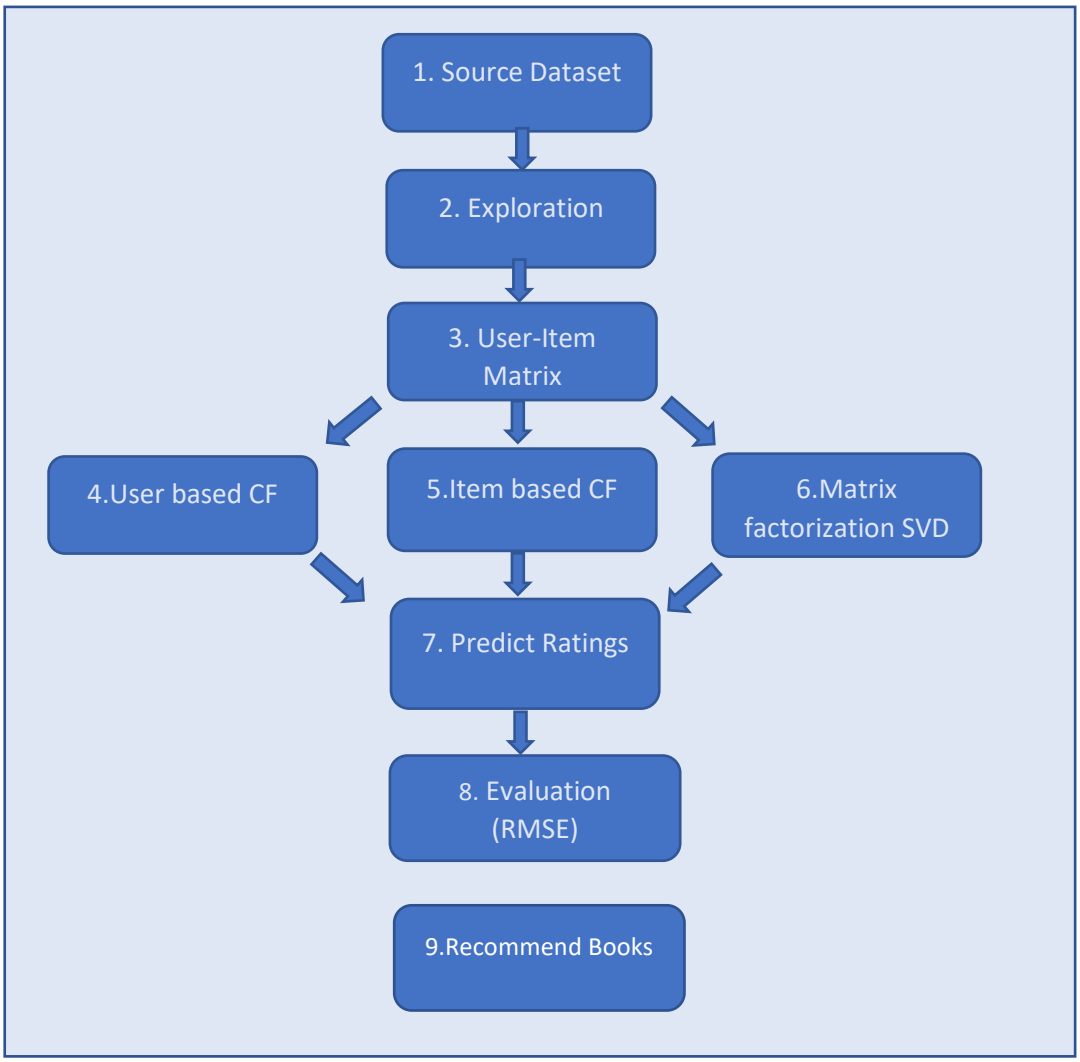
through the site to enhance recommendation over time. Site predicts books that users may like based on book that user has rated or purchased.

Libra is another book recommendation system based on content. It uses web data to learn about User and then use Naïve Bayes classifier to create a rank of books that can be recommended to user. Good thing about the site is that it provides the information on features that has produced the result.

3. Conduct Experiment:

Below framework was used for carrying out different experiments.

3.1 Framework used for the experiment:



I. Source data set:

First step in this study was to get a data set that has historical user rating information that can be used to evaluate the performance of the mode. Following were the key criteria that was used to get a data set

a) Data set that has User rating information and Item details
b) Good quality data that can be used after cleaning up as training and test data set.

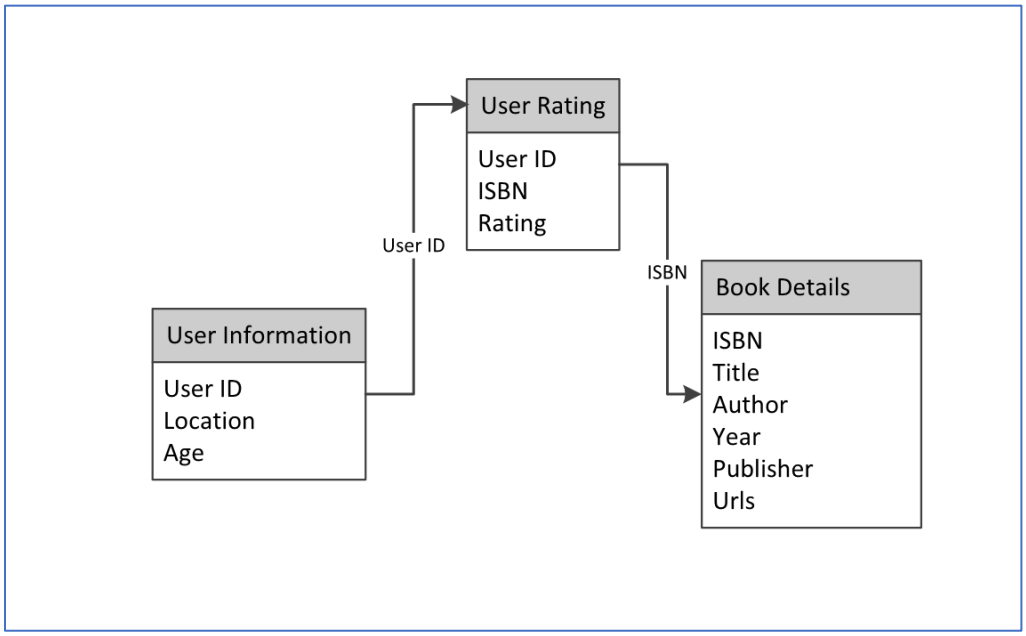
For this study available dataset from internet has been considered. Following data source was used for Book recommendation dataset.

Book Crossing data set. This data set has three key files-

http://www2.informatik.uni-freiburg.de/~cziegler/BX/

1. User Information file
2. User Rating file
3. Book Details file

Below is the data file structure:



Data files structure:

User Information file:

	UserID	Location	Age
0	1	nyc, new york, usa	NaN
1	2	stockton, california, usa	18.0
2	3	moscow, yukon territory, russia	NaN

User Rating file:

	UserID	ISBN	Rating
0	276725	034545104X	0
1	276726	0155061224	5
2	276727	0446520802	0

Book details file:

	ISBN	Title	Autho	Year	Publisher	Urls	Urlm
0	0195153448	Classical Mythology	Mark P. O. Morford	2002	Oxford University Press	http://images.amazon.com/images/P/0195153448.0...	http://images.amazon.com/images/P/019
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/000
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/006

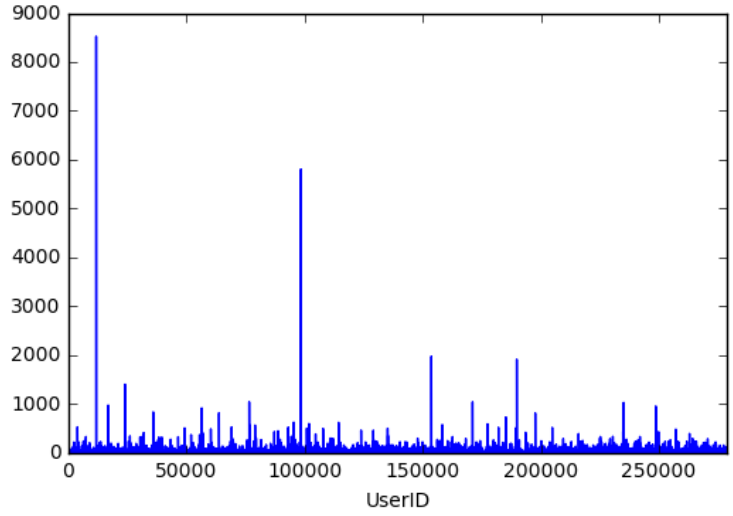
II. Data Exploration:

Key objective of data exploration exercise is to understand the underline data around User, User rating behavior, ratings associated with different books. Then transform the data set to a format that can be used for creating the model**.**

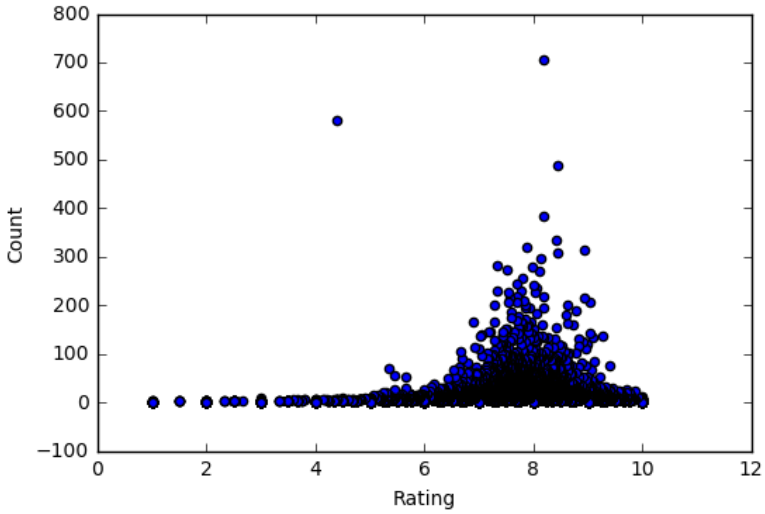
Focus of data exploration and transformation is to understand and explore following points-

1. How Users have rated different books
2. How many ratings each book has
3. Distribution of average ratings
4. Transforming data to a matrix form to use that for building model

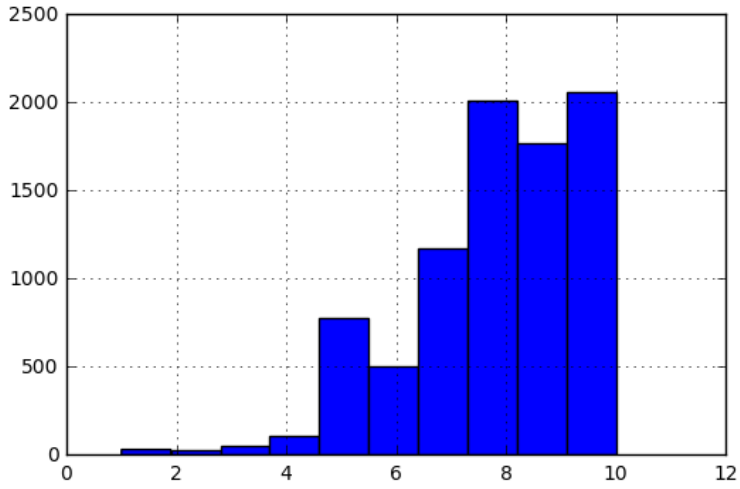
Data set has unique 77805 users who has provided ratings for 185973 books. User ratings varies from 0 to 10. Below is the distribution of total number User ratings. It can be seen there are few users who has provided very high number of ratings in this data set.



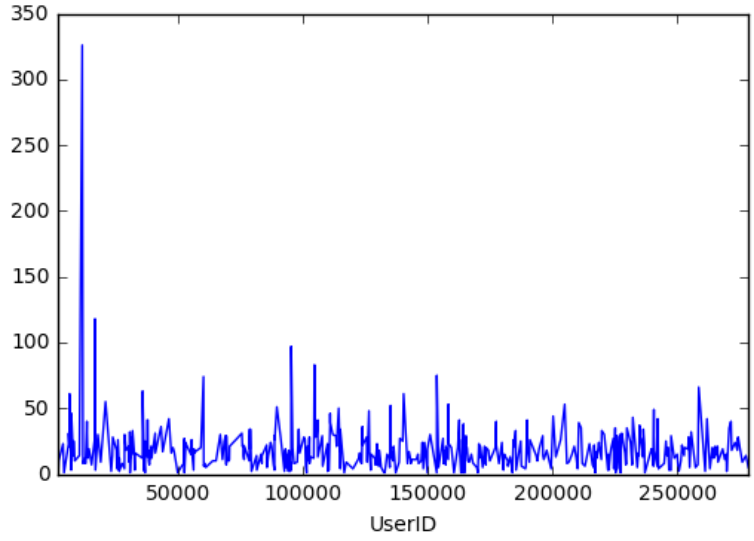
In this diagram data has been plotted by count of Users who have rated a book and ratings for the book. This indicates popular books tends to have more ratings from Users.



Data has been filtered to only keep those books having at least 50 user ratings. Also, there were very high number of ratings with zero values. Zero could be misleading as very low rating or not and data has been plotted below. It shows most of the books



After the data has been transformed wide variations has been reduced to large extent. Below chart is derived out of derived dataset.



III. User item matrix:

After data has been cleaned and sorted user item matrix was created. This matrix has user in one dimension and item in another. Cell values denotes the ratings by users for books. As all users do not rate all the books. This matrix is high dimension matrix with very high sparsity. Modeling data with NA values will result in very skewed outcome and high error. Following two approaches were used for the data:

a. Using mean rating values for NA values

b. Using Median values for missing values

Next data set was split between test and train dataset. Test dataset will be used for evaluating the performance for this study.

IV. User-based Collaborative Filter:

In this approach similarity between two users is derived. Prediction for an item for a user u is calculated by weighted average of sum of different users ratings for item i. This method is used to, predict the unknown user ratings for books. In this case for user u and ith item-

$$P_{(u,i)} = \sum((R_{(v,i)} * S_{(u,v)}) / \sum (S_{(u,v)}$$

Here R(v,i) is the rating of user v on item i.
S(u,v) similarity matrix for u*v

User based Collaborative model was applied to the User Item matrix and similarity values were calculated. Using pairwise distance from sklearn package Cosine similarity was calculated. Similarity values are between 0-1 as all ratings are positive. In the data set large number of Users have only few ratings and values are between 0-10. Based on this Cosine similarity measures were selected. As different Users tends to rate different way adjustment to rating variance is required by using weighted average of User ratings.

V. Item based Collaborative Filter:

In this approach same process was followed as above for User based CF. Only difference is the way Cosine similarity was calculated by changing the dimension of User Item matrix.

VI. Matrix factorization (SVD):

Singular value decomposition (SVD) is well known matrix factorization method. Collaborative filtering can be done by deriving a matrix by using singular value decomposition. Rating matrix X can be decomposed into following components.

$$X = U \times S \times V$$

Where X= rating matrix (mxn dimension)

U= m orthogonal matrix (mxr) representing feature vector for user

S= r x r diagonal matrix (singular values)

V= r x n orthogonal matrix feature vector for Item

Prediction can be made for unknown values taken dot product of = U,S V

Matrix factorization is known for dealing better way in terms of scalability and sparsity than Memory-based CF. The goal of Matrix factorization is to identify or learn the latent preferences of users and the latent attributes of items then predict the unknown ratings using the dot product of the features of users and items.

VII. Predict Ratings

Using the similarity matrix prediction matrix was calculated for User based and Item based Collaborative method. Example is shown below unknown values were derived.

User-Item Matrix	Book 1	Book 2	Book 3	….	Book N
User 1	**7**	**8**			**6**
User 2	**6**	**5**	**8**		**9**
User 3		**4**	**9**		
…..					
User M	**5**	**8**			**7**

Similarity Matrix	Book 1	Book 2	Book 3	….	Book N
User 1	**0.81**	**0.001**	**0.0123**	**0.75**	**0.001**
User 2	**0.22**	**0.091**	**0.345**	**0.31**	**0.96**
User 3	**0.001**	**0.92**	**0.034**	**0.23**	**0.001**
…..	**…**				
User M	**0.012**	**0.023**	**0.88**	**0.41**	**0.023**

Using the similarity matrix and User Item matrix similarity prediction matrix was created as shown below. One key outcome of the matrix is unknown rating.

Predicted values	Book 1	Book 2	Book 3	….	Book N
User 1			**2.1**		
User 2					
User 3	**1**				**1**
…..					
User M			**8.5**		

VIII. Evaluation

There are different metrics used for evaluation of recommendation systems. Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are two well know metrics. For this study RMSE has been selected as it gives higher weightage to higher defects by square hence penalize undesirable higher errors.

RMSE= sqrt[(1/N) ∑ (Ai−Xi^)2]

Ai= Actual rating given by user
Xi= Predicted rating
N = Number of observations

IX. Recommend Books

Best model has been selected based on the lower value of RMSE. Model will be used to recommend list of books to user based on user Id.

4. RESULTS AND CONCLUSIONS:

4.1 Handling sparsity:

Sparsity is one of the issue in building a recommendation model. Recommender model created without treating the Null values will result in very high RMSE. It was important to handle the Null values for an effective model. Following two strategies were used to handle the null values:

1. Using mean values for each book. In many cases due to Null values mean value has resulted in skewed values. Due to that 2nd method was also used.
2. Using Median value for a book.

4.2 Training and test data set:

Data for the experiment was split to between a test and training set. Training set was used to come up with a model and test set was used to evaluate the performance of model. Data was split in two ways to compare the performance of the outcome.

1. Splitting train and test set by 80/20 rule. With this approach required amount of data is available to do the validation. But less amount of actual data is available for model building.
2. Splitting train and test by 90/10 rule. This approach issue is completely opposite. Both the approaches were used to balance out the outcome and get comparative outcome.

4.3 Results from experiments:

In first approach data set was split 80/20 between train and test. Following experiments were conducted in the train/test set and RMSE value was compared:

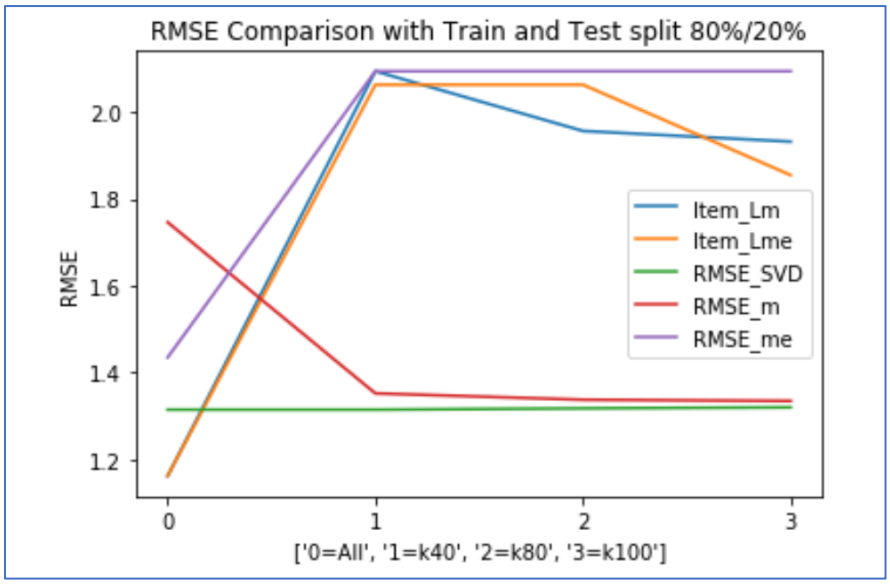
- Item_Lm= Data set was replaced with mean value and Item CF model created
- Item_Lme= Data set was replaced with median value and Item CF model created
- RMSE_SVD= Data set was replaced with mean value and SVD model created
- RMSE_m= Data set was replaced with mean value and user CF model created

- RMSE_me= Data set was replaced with median value and user CF model created

For each of this experiment different observations were made with different values

1. Full train dataset without K value
2. K value =40
3. K value=80
4. K value =100

From experiment 1, SVD method came out as clear winner with less and consistent level of RMSE.



Outcome with 90/10% data split:

In second approach data set was split 90/10 between train and test. Following experiments were conducted in the train/test set and RMSE value was compared:

Definition of the different lines are given below:

- RMSE_m_10= Data set was replaced with mean value and user CF model created
- RMSE_SVD10= Data set was replaced with mean value and SVD model created
- Item_Lm_10= Data set was replaced with mean value and user CF model created

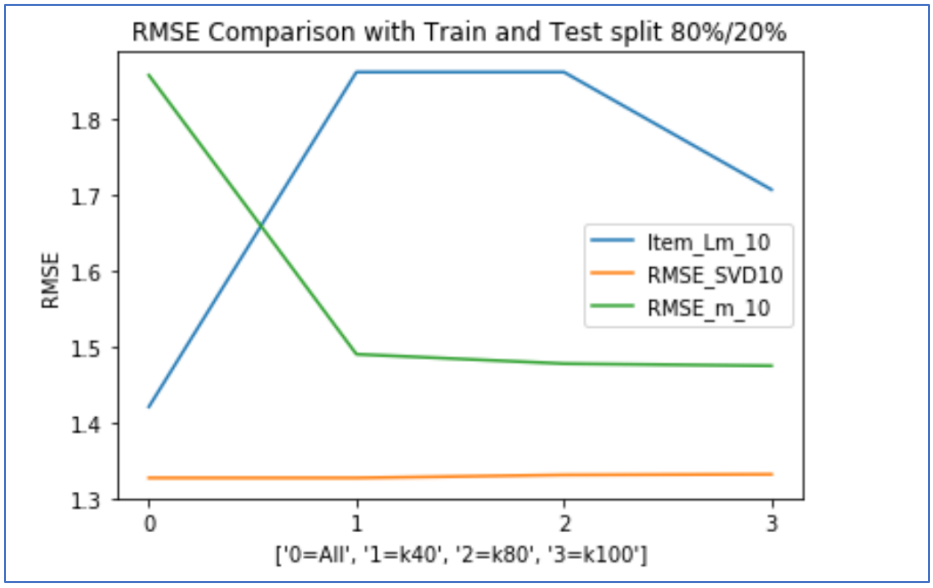
Each Observation was run for – following scenarios

1. Full train dataset without K value
2. K value =40

3. K value=80
4. K value =100

In second approach was also SVD came out with better result.

Based on the comparison from both the scenarios matrix factorization method has performed far better than other two methods. One key reason that was identified was sparsity in the data. Based on that final matrix can be derived that can be filter by both User and Item to provide list of recommended books.



4.4 Top 10 Book recommendation to User:

Using the selected model SVD model one sample case has been demonstrated how model responds to real life scenario. Based on the logged in user, user value can be picked up and recommender can recommend book. In this case study, User Id is = ''35857' . Recommender, identifies and sorts out predicted score for different books for that user. Next top 10 books with high score are displayed to the user.

```
Recommendation_for_user('35857')
```

	ISBN	Title	Autho	Year	Publisher
388	0156528207	The Little Prince	Antoine de Saint-ExupÃ©ry	1968	Harcourt
1080	0394800133	One Fish Two Fish Red Fish Blue Fish (I Can Re...	DR SEUSS	1960	Random House Books for Young Readers
2785	0618002219	The Hobbit: or There and Back Again	J.R.R. Tolkien	1999	Houghton Mifflin Company
3847	0064400557	Charlotte's Web (Trophy Newbery)	E. B. White	1974	HarperTrophy
5898	0553274295	Where the Red Fern Grows	Wilson Rawls	1984	Random House Children's Books

Further analysis on this reveals the User ''35857' has not read the recommended books earlier and all the books are very highly rated and hence the model is working very effectively.

```
ISBN
0836218051 9.000000
0156528207 8.978723
0618002219 8.972222
0064400557 9.166667
0553274295 8.592593
0140143505 9.560000
0618002235 9.708333
0836218051 9.000000
006440188X 9.157895
```

5. REFERENCES

[1] comparison of collaborative filtering algorithms with various similarity measures for movie recommendation International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.6, No.3, June 2016

[2] User-Based and Item-Based Collaborative Filtering Recommendation Algorithms Design- Guanwen Yao

[3] Recommender Systems - Comparison of Content-based Filtering and Collaborative Filtering- Bhavya SanghaviÀ*, Rishabh RathodÀ and Dharmeshkumar MistryÀ- Computer Science, Dwarkadas J.Sanghvi College of Engineering, Vile Parle(W),Mumbai-400056,India

[4] Building a Book Recommender system using time based content filtering- CHHAVI RANA Department of Computer Science Engineering, University Institute of Engineering and Technology, MD University, Rohtak, Haryana, 124001, INDIA.

[5] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. "Research Paper Recommender Systems: A Literature Survey." International Journal on Digital Libraries (2015):1–34. doi:10.1007/s00799-015-0156-0.

[6] https://www.quora.com/How-do-I-compute-Precision-and-Recall-in-Recommender-Systems

[7] http://fastml.com/evaluating-recommender-systems/

[8] http://aimotion.blogspot.com/2011/05/evaluating-recommender-systems.html

[9] https://cambridgespark.com/content/tutorials/implementing-your-own-recommender-systems-in-Python/index.html
[10] Matrix Factorization and Collaborative Filtering, Daryl Lim, University of California, San Diego,2013
[11] Recommendation System Based on Collaborative Filtering, Zheng Wen, December 12, 2008
[12] Recommender Systems for Learning, Nikos Manouselis • Hendrik Drachsler Katrien Verbert • Erik Duval,Springer
[13]Dataset http://www2.informatik.uni-freiburg.de/~cziegler/BX/
[14] https://medium.com/ai-society/a-concise-recommender-systems-tutorial-fa40d5a9c0fa
[15] http://blog.untrod.com/2016/06/simple-similar-products-recommendation-engine-in-python.html

6. Appendix

4.1 Github link to Python code & data files

https://github.com/arindambarman/Datascience