Trustworthy Recommender System using Machine Learning Techniques

Submitted as a partial fulfilment of Bachelor of Technology in Computer Science & Engineering
Of
Maulana Abul Kalam Azad University of Technology
(*Formerly known as West Bengal University of Technology)*

Project Report

Submitted by

Supriya Kundu	**11600116015**
Shalmoli Neogi	**11600116030**
Sarbajit Nandy	**11600116034**
Vishnu Pandey	**11600116011**

Under the supervision of

Mrs. Sasmita Subhadarsinee Choudhury

Assistant Professor, Computer Science and Engineering

Department of Computer Science & Engineering,
MCKV Institute of Engineering
243, G.T. Road(N)
Liluah, Howrah - 711204

Department of Computer Science & Engineering
MCKV Institute of Engineering
243, G. T. Road (N),
Liluah, Howrah-711204

CERTIFICATE OF RECOMMENDATION

I hereby recommend that the thesis prepared under my supervision by **Supriya Kundu, Shalmoli Neogi, Sarbajit Nandy, Vishnu Pandey** entitled Trustworthy Recommender System using Machine Learning Techniques be accepted in partial fulfilment of the requirements for the degree of Bachelor of Technology in Computer Science & Engineering Department.

Dr. S. S.Thakur
Associate Professor & Head of the Department,
Computer Science & Engineering Department.
MCKV Institute of Engineering, Howrah

Mrs. Sasmita Subhadarsinee Choudhury
Assistant Professor,
Computer Science & Engineering Dept.
MCKV Institute of Engineering, Howrah

MCKV Institute of Engineering
243, G. T. Road (N), Liluah
Howrah-711204

Affiliated to

Maulana Abul Kalam Azad University of Technology
(Formerly known as West Bengal University of Technology)

CERTIFICATE

This is to certify that the project entitled Trustworthy Recommender System using Machine Learning Techniques and submitted by

Supriya Kundu	11600116015
Shalmoli Neogi	11600116030
Sarbajit Nandy	11600116034
Vishnu Pandey	11600116011

has been carried out under the guidance of myself following the rules and regulations of the degree of Bachelor of Technology in Computer Science & Engineering of **Maulana Abul Kalam Azad University of Technology (**Formerly West Bengal University of Technology).

____

(Signature of the project guide)
Mrs. Sasmita Subhadarsinee Choudhury,
Assistant Professor,
Computer Science & Engineering,

1. ___________________________
2. ___________________________
3. ___________________________
4. ___________________________

MCKV Institute of Engineering
243, G. T. Road (N), Liluah
Howrah-711204
Affiliated to
Maulana Abul Kalam Azad University of Technology
(Formerly known as West Bengal University of Technology)

CERTIFICATE OF APPROVAL
(B.Tech Degree in Computer Science & Engineering)

This project report is hereby approved as a creditable study of an engineering subject carried out and presented in a manner satisfactory to warrant its acceptance as a prerequisite to the degree for which it has been submitted. It is to be understood that by this approval, the undersigned do not necessarily endorse or approve any statement made, opinion expressed and conclusion drawn therein but approve the project report only for the purpose for which it has been submitted.

COMMITTEE ON FINAL	1. ______________________
EXAMINATION FOR	2. ______________________
EVALUATION OF	3. ______________________
PROJECT REPORT	4. ______________________
	5. ______________________

ACKNOWLEDGEMENT

It is our proud privilege and duty to acknowledge the kind of help and guidance received from several people in preparation of this report. We express our gratitude and respect towards all those who guided us throughout the completion of this project and report.

First and foremost, we wish to record our sincere gratitude to our project guide **Mrs. Sasmita Subhadarsinee Choudhury**, Assistant Professor, CSE for his constant support and encouragement in preparation of this project and report. We would also like to thank all the people who have provided their valuable suggestions and feedbacks to make this project.

Our sincere thanks to **Prof. (Dr.) S. S. Thakur**, Associate Professor & Head of the Department, Department of Computer Science and Engineering, MCKVIE, for his valuable suggestions and encouragement.

CONTENTS

PROJECT ABSTRACT

Recommendation systems are one of the most successful and widespread applications of machine learning technologies in E-commerce. These techniques are used to predict the rating that one individual will give to an item or social entity. It uses the opinions of members of a community to help individuals in that community to identify the information most likely to be interesting to them or relevant to their needs. These systems use the similarity between the user and recommenders or between the items to form the recommendation list for the user. These preferences are being predicted using different approaches, namely content-based approach, collaborative filtering approach, etc. In a highly dynamic and decentralized environment, where data are uncertain, Trust has become a key factor in the process of decision making. Trust-based recommendation is based on **trust** between users. Applying trust consideration in item recommendations, which is aiming to suggest the optimal items to users. Herein, the trust is considered from similarity among users within the social network and accuracy of item predictions.

INTRODUCTION

In today's world, every customer is faced with multiple choices. For example, If I'm looking for a book to read without any specific idea of what I want, there's a wide range of possibilities how my search might pan out. I might waste a lot of time browsing around on the internet and trawling through various sites hoping to strike gold. I might look for recommendations from other people.

But if there was a site or app which could recommend me books based on what I have read previously, that would be a massive help. Instead of wasting time on various sites, I could just log in and get 10 recommended books tailored to my choice.

This is what recommendation systems do and its power is being harnessed by most businesses these days. From Amazon to Netflix, Google to Goodreads, recommendation engines are one of the most widely used applications of machine learning techniques.

3.Recommender system

A recommendation engine filters the data using different algorithms and recommends the most relevant items to users. It first captures the past behaviour of a customer and based on that, recommends products that the users might be likely to buy.

3.1 Working principle of recommender system:

- ✓ We can recommend items to a user which are most popular among all users.
- ✓ We can divide users into multiple classes based on their preferences and recommend items to them based on the class they belong to.
- ✓ We can divide the products into multiple classes based on types or features and recommend the same type of product to the user.

The above processes have some drawbacks. In the first case, the most popular items would be the same for each user so everybody will see the same recommendations. While in other cases, as the number of users increase, the number of features will also increase. So classifying the users into various classes will be very difficult.

3.2. Types of recommender system:

There are four broad types of recommender system:

1. Collaborative filtering
2. Content-based filtering
3. Knowledge-based recommendation
4. Hybrid recommendation

1) **Collaborative filtering:** Collaborative filtering, also referred to as social filtering, filters information by using the recommendations of other people. It is based on the idea that people who agreed in their evaluation of certain items in the past are likely to agree again in the future. A person who wants to see a movie, for example, might ask for recommendations from friends. The recommendations of some friends who have similar interests are trusted more than recommendations from others. This information is used in the decision on which movie to see.
2) **Content-based filtering:** This type of filter does not involve other users if not ourselves. Based on what we like, the algorithm will simply pick items with similar content to recommend us. In this case, there will be less diversity in the recommendations, but this will work either the user rates things or not.
3) **Knowledge-based filtering:** These types of recommender systems are employed in specific domains where the purchase history of the users is smaller. In such systems, the algorithm takes into consideration the knowledge about the items, such as features, user preferences asked explicitly, and recommendation criteria, before giving recommendations. The accuracy of the model is judged based on how useful the recommended item is to the user. Take, for example, a scenario in which you are building a recommender system that recommends household electronics, such as air conditioners, where most of the users will be first-timers. In this case, the system considers features of the items, and user profiles are generated by obtaining additional information from the users, such as specifications, and then recommendations are made. These types of systems are called constraint-based recommender systems, which we will learn more about in subsequent chapters.
4) **Hybrid recommendation:** Most recommender systems now use a hybrid approach, combining collaborative filtering, content-based filtering, and other approaches. There is no reason why several different techniques of the same type could

not be hybridized. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model (see for a complete review of recommender systems). Several studies that empirically compare the performance of the hybrid with pure collaborative and content-based methods and demonstrated that hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as the cold start and the sparsity problem, as well as the knowledge engineering bottleneck in knowledge-based approaches.

4. Similarity measure

The similarity measure is the measure of how much alike two data objects are. Similarity measure (in a data mining context) is a distance with dimensions representing features of the objects. If this distance is small, it will be a high degree of similarity where large distance will be the low degree of similarity.

The similarity is subjective and is highly dependent on the domain and application. For example, two fruits are similar because of color or size or taste. Care should be taken when calculating distance across dimensions or features that are unrelated. The relative values of each element must be normalized, or one feature could end up dominating the distance calculation. Similarity are measured in the range 0 to 1 [0,1].

Two main consideration about similarity:

- Similarity = 1 if X = Y , (Where X, Y are two objects)
- Similarity = 0 if X ≠ Y

4.1 Reasons of Similarity measure:

Collaborative filtering has become the most frequently used method to suggest items for users. It makes suggestion in accordance to similar users with the active user or similar item with the items which are rated by the active user.

Collaborative filtering mainly consists of two methods:

- Model-based method
- Memory-based method

The model-based method first defines/explains the interest of users and consequently forecast the rating of items.

The memory-based method first defines the similarity among users and then selects the most similar users as neighbouring recommenders. So, here, for finding out the best possible similar users, different similarity measures are used.

4.2 Popular Types of Similarity measure:

- **Euclidean Distance:** Euclidean distance is the most common use of distance. In most cases when people said about distance, they will refer to Euclidean distance. Euclidean distance is also known as simply distance. When data is dense or continuous, this is the best proximity measure. The Euclidean distance between two points is the length of the path connecting them. The Pythagorean theorem gives this distance between two points.

- **Manhattan Distance:** Manhattan distance is a metric in which the distance between two points is the sum of the absolute differences of their Cartesian coordinates. In a simple way of saying it is the total sum of the difference between the x-coordinates and y-coordinates. Suppose we have two points A and B if we want to find the Manhattan distance between them, just we have, to sum up, the absolute x-axis and y-axis variation means we have to find how these two points A and B are varying in X-axis and Y-axis. In a more mathematical way of saying Manhattan distance between two points measured along axes at right angles. In a plane with p1 at (x1, y1) and p2 at (x2, y2).
  Manhattan distance = |x1 – x2| + |y1 – y2|

- **Minkowski Distance:** The Minkowski distance is a generalized metric form of Euclidean distance and Manhattan distance. In the equation, d^MKD is the Minkowski

$$d^{MKD}(i,j) = \sqrt[\lambda]{\sum_{k=0}^{n-1} \left| y_{i,k} - y_{j,k} \right|^{\lambda}}$$

  distance between the data record i and j, k the index of a variable, n the total number of variables y and λ the order of the Minkowski metric. Although it is defined for any λ > 0, it is rarely used for values other than 1, 2 and ∞. The way distances are measured by the Minkowski metric of different orders between two objects with three variables (In the image, it displayed in a coordinate system with x, y, z-axes).

Synonyms of Minkowski: Different names for the Minkowski distance or Minkowski metric arise from the order:

- $\lambda = 1$, is the Manhattan distance. Synonyms are L1-Norm, Taxicab or City-Block distance. For two vectors of ranked ordinal variables, the Manhattan distance is sometimes called Foot-ruler distance.
- $\lambda = 2$, is the Euclidean distance. Synonyms are L2-Norm or Ruler distance. For two vectors of ranked ordinal variables, the Euclidean distance is sometimes called Spear-man distance.
- $\lambda = \infty$, is the Chebyshev distance. Synonyms are Lmax-Norm or Chessboard distance.

❖ **Cosine Similarity:** The cosine similarity metric finds the normalized dot product of the two attributes. By determining the cosine similarity, we would effectively try to find the cosine of the angle between the two objects. The cosine of 0° is 1, and it is less than 1 for any other angle. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors at 90° have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1]. One of the reasons for the popularity of cosine similarity is that it is very efficient to evaluate, especially for sparse vectors.

$$\cos\theta = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\ \sqrt{\sum_{i=1}^{n} b_i^2}}$$

❖ **Jaccard Similarity:** Jaccard similarity is computed using the following formula:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

The library contains both procedures and functions to calculate the similarity between sets of data. The function is best used when calculating the similarity between small numbers of sets. The procedures parallelize the computation and are therefore more appropriate for computing similarities on bigger datasets.

4.3 Drawbacks of similarity measure:

The core of collaborative filtering in case of memory-based method is to compute similarities between users or items. The generic traditional similarity measure such as Pearson Correlation Coefficient, mean square distance or cosine are not enough to capture the most similar neighbours, particularly for those who are cold user, i.e. who have rated very small number of items. Data Sparsity is also a big issue in case of collaborative filtering.

5.Trust-aware Recommender System

With time, researchers have shifted their attention towards incorporating trust in RSs because of the problem of sparsity, malicious attacks etc. In traditional CF based RS, motivated by the fact the people generally agree upon the choices made by their friends, colleagues whom they trust. Trust information helps in overcoming the said problems.

5.1 Requirement of Trust:

Trust plays an important role across many disciplines and forms an important feature of our everyday lives. In addition, trust is a property associated with people in the real world as well as social media. In recommender systems, it is defined based on the other user's ability to provide a valuable recommendation.

5.2 Types of Trust:

Trust can be either of two types:

- **Explicit trust**: Explicit trust is the trust value explicitly provided by the users.
- **Implicit trust**: It refers to the trust information implicitly inferred from user behaviour in the system e.g. user-item ratings.

5.3 Advantages of implicit trust over explicit trust:

1) Explicit trust puts an extra burden on user for providing trust information apart form rating information.

2) Some popular datasets available for research purposes lack explicit trust information.

3) Most of the publicly available explicit trust data set contain trust scores in binary form.

4) Sometime explicit trust could be noisy. Users may trust each other due to various offline relations.
5) The amount of explicit trust information is relatively less than the number of ratings.
6) Implicit trust metrics are based on the instinct that users with similar ratings tend to be trustworthy.

5.4 Trust Matrix Calculation: Trust is defined as one's belief towards the ability of others in providing valuable ratings. Conceptually trust is different from finding similar users. It is possible that two highly similar users may have low trust between them and therefore may not like items recommended by the other one. Trust can be provided explicitly by the user or generated implicitly from various metrics. Explicit trust data is very sparse and not frequently available. To compute the implicit trust various authors suggested different metrics and analyze them based on certain properties.

There are different processes to compute trust metrics.

1) **Lathia et al** calculated trust using mean of absolute difference of ratings between two users.

$$t_{u,v} = \frac{1}{|I_{u,v}|} \sum_{i \epsilon I_{u,v}} (1 - \frac{|r_{u,i} - r_{v,i}|}{r_{max}})$$

It is trivial to analyse that the above equation gives a value of trust that is same for both users ***u*** and ***v*** towards the one. Thus, trust is given by this equation is symmetric.

2) **Hwang and Chen** used Resnick's prediction formula for generating predicted rating.

$$p_{u,i} = \bar{r}_u + (r_{v,i} - \bar{r}_v)$$

The trust value is derived as follows:

$$t_{u,v} = \frac{1}{|I_{u,v}|} \sum_{i \epsilon I_{u,v}} (1 - \frac{|p_{u,i} - r_{u,i}|}{r_{max}})$$

3) **Papagelis et al** derived trust via user similarity computed as Pearson correlation coefficient (PCC). It basically treats similarity values generated by PCC as trust and

propagates this trust values over the network to generate inferred trust values. In order to eliminate data sparsity.

$$t_{u,v} = \frac{\sum_i (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_i (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_i (r_{v,i} - \bar{r}_v)^2}}$$

Papagelis et al considers propagation of implicit trust generated. Trust values can be propagated only if they are positive in nature. Negative trust (distrust) is not propagated in network.

4) **Shambour and Lu** also used Resnick's prediction formula for trust computation. Computed trust is based on men squared distance (MSD).

$$t_{u,v} = \frac{|I_{u,v}|}{|I_u \cup I_v|} \left(1 - \frac{1}{|I_{u,v}|} \sum_{i \in I_{u,v}} \left(\frac{|p_{u,i} - r_{u,i}|}{r_{max}} \right)^2 \right)$$

$t_{u,v}$ = Trust between user u and user v

$I_{u,v}$ = no of items rated by both user u and v

$r_{u,i}$ = rating of ith item given by user u

$r_{v,i}$ = rating of ith item given by user u

r_{max} = maximum rating possible for an item

RESULT AND DISCUSSION

- We used collaborative filtering for building a book-recommender system.
- We used 3 datasets: users.csv, books.csv, ratings.csv

Sample of User Dataset

	userID	Location	Age
0	1	nyc, new york, usa	NaN
1	2	stockton, california, usa	18.0
2	3	moscow, yukon territory, russia	NaN
3	4	porto, v.n.gaia, portugal	17.0
4	5	farnborough, hants, united kingdom	NaN

Sample of Book Dataset

	ISBN	bookTitle	bookAuthor	yearOfPublication	publisher	imageUrlS	imageUrlM	imageUrlL
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/0195153448.0...	http://images.amazon.com/images/P/0195153448.0...
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/0002005018.0...
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/0060973129.0...
3	0374157065	Flu: The Story of the Great Influenza Pandemic...	Gina Bari Kolata	1999	Farrar Straus Giroux	http://images.amazon.com/images/P/0374157065.0...	http://images.amazon.com/images/P/0374157065.0...	http://images.amazon.com/images/P/0374157065.0...
4	0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton & Company	http://images.amazon.com/images/P/0393045218.0...	http://images.amazon.com/images/P/0393045218.0...	http://images.amazon.com/images/P/0393045218.0...

Sample of Ratings Dataset

	userID	ISBN	bookRating
0	276725	034545104X	0
1	276726	0155061224	5
2	276727	0446520802	0
3	276729	052165615X	3
4	276729	0521795028	6

- User-Book Rating matrix is created using the ratings given to each book by each user

User-Book Rating Matrix

us_canada_user_rating_pivot - DataFrame

Index	1984	lst to Die: A Nove	2nd Chance	4 Blondes	Charing Cross Ro	\ Bend in the Roa	A Case of Need	\": One Child's Co	A Civil Action	\ Cry In The Night	kness More Than	Late and a Dollar	A
254	9	nan	nan	nan	nan	0	nan	nan	nan	nan	nan	nan	nar
2276	nan	nan	10	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
2766	nan	nan	nan	nan	nan	7	0	nan	nan	nan	nan	nan	nar
2977	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
3363	nan	nan	nan	nan	nan	nan	nan	nan	0	nan	nan	nan	nar
3757	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
4017	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
4385	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
6242	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
6251	nan	nan	nan	0	nan	nan	0	nan	nan	nan	nan	nan	nar
6323	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	7	nan	nar
6543	nan	9	0	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
6563	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
6575	nan	nan	nan	nan	0	1	nan	nan	0	nan	nan	nan	0
7158	nan	0	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
7286	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
7346	8	nan	nan	nan	nan	nan	nan	0	0	nan	nan	nan	nar
7915	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
8067	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nar
8245	nan	nan	nan	nan	nan	nan	nan	nan	0	nan	nan	nan	9
8681	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	nan	0	nar
8936	nan	0	nan	nan	nan	nan	0	nan	nan	nan	nan	nan	nar

Format Resize Background color Column min/max Save and Close Close

- Trust matrix is made, which depicts how much trust one user has on each of the other users. It is basically an user vs user matrix.

Trust Matrix

	0	1	2	3	4	5	6	7	8	9	10	11	12
0	1	0.733333	0.633333	0.466667	1	1	0.383333	0	0.266667	0.578947	0.7	1	0.738462
1	0.733333	1	0.7	1	0.333333	0.3	0	1	0	0.6	0.7	0	0.65
2	0.633333	0.7	1	0.7	0.644444	0	0.544444	0	0.5	0.58	0.95	0.45	0.51
3	0.466667	1	0.7	1	0.7	0.3	0.3	0	1	0.766667	0.3	0.6	0.466667
4	1	0.333333	0.644444	0.7	1	0	0.62	0	0.416667	0.829167	0.685714	0.7875	0.58
5	1	0.3	0	0.3	0	1	0	0	0	0	0	0.2	0
6	0.383333	0	0.544444	0.3	0.62	0	1	1	0.62	0.521053	0.666667	0.533333	0.253333
7	0	1	0	0	0	0	1	1	0	0.5	0	0	0
8	0.266667	0	0.5	1	0.416667	0	0.62	0	1	0.56	0	0.52	0.633333
9	0.578947	0.6	0.58	0.766667	0.829167	0	0.521053	0.5	0.56	1	0.925	0.65625	0.651852
10	0.7	0.7	0.95	0.3	0.685714	0	0.666667	0	0	0.925	1	0	0.575
11	1	0	0.45	0.6	0.7875	0.2	0.533333	0	0.52	0.65625	0	1	0.9
12	0.738462	0.65	0.51	0.466667	0.58	0	0.253333	0	0.633333	0.651852	0.575	0.9	1
13	0.568421	0.2	0.521053	0.6	0.816667	0	0.426667	0	0.5875	0.675	0.516667	0.592593	0.541935
14	0.525	0.6	0.618182	1	0.806667	0	0.4	0	0.366667	0.688889	0.7	0.6	0.685714
15	0.5	1	0.466667	1	0.766667	0	0.4	0	0.2	0.8	1	0.625	0.55
16	0.716	0.6875	0.48125	0.433333	0.717647	0.3	0.373684	0.6	0.4125	0.706667	0.622222	0.56087	0.728571
17	0.6	0	0.75	0.8	0.72	0	0.5875	0	0.425	0.672222	0.98	0.578571	0.569231
18	0.5	1	0.2	1	0.54	0	0.566667	0	0.5	0.333333	0	0.5125	0.833333
19	0.46	0.3	0.2	0.48	0.84	0	0.446154	1	0.65	0.576923	0.2	0.47	0.430769

- A target user is chosen from the trust matrix, and 5 similar users, i.e.: on whom the target user has much trust is found out.
- A book-title vs user matrix created where only trusted users' ratings for each book is considered

Results after getting 5 similar users of a random target user & their book rating information

userID	227447	135149	35859	225763	87746	Average
BookTitle						
The Little Prince	10.0	NaN	NaN	NaN	NaN	10.0
Tara Road	NaN	10.0	NaN	NaN	NaN	10.0
Sudden Prey	10.0	NaN	NaN	NaN	NaN	10.0
The Red Tent (Bestselling Backlist)	NaN	NaN	10.0	NaN	NaN	10.0
Left Behind: A Novel of the Earth's Last Days (Left Behind No. 1)	NaN	NaN	9.0	NaN	NaN	9.0
The Mummy or Ramses the Damned	NaN	NaN	9.0	NaN	NaN	9.0
Midnight in the Garden of Good and Evil	NaN	9.0	NaN	NaN	NaN	9.0
The Eight	NaN	NaN	NaN	9.0	NaN	9.0
Nicolae: The Rise of Antichrist (Left Behind No. 3)	NaN	NaN	9.0	NaN	NaN	9.0
The Book of Ruth (Oprah's Book Club (Paperback))	NaN	9.0	NaN	NaN	NaN	9.0

- The book which has the highest average rating and has not been read by the target user is recommended to him.

CONCLUSION

Collaborative filtering has become the most frequently used method to suggest items for users. It makes suggestion in accordance to similar users with the active user or similar item with the items which are rated by the active user.

Collaborative filtering mainly consists of two methods:

- Model-based method
- Memory-based method

The model-based method first defines/explains the interest of users and consequently forecast the rating of items.

The memory-bases method first defines the similarity among users and then selects the most similar users as neighbouring recommenders. But the computing time grows rapidly with the increasing number of items and users, which makes it hard to take action in real time.

The model-based technique tends to be faster in prediction time then the memory-based technique, because the creation of the model can be completed in a considerable amount of time and we can make use of this technique off-line.

FUTURE SCOPE OF THE WORK

- We used collaborative filtering, but using content-based filtering it can be done also
- Here we only built it for books, but would like to build a generic recommender system, which can be used for any platform, like e-commerce website etc.
- This recommender system can be also built for not only recommending similar movies, books or gadgets, but also things which trusted people usually buys together. For example, if an user buys a mobile phone, then the recommender system finds that his trusted users who are buying phones are also buying temper glass for that. So it can recommend that too to the target user.
- We can also make use of both content-based filtering along with collaborative filtering to make a hybrid structure that will make this strategy more accurate.
- A comparative study on different similarity measure will be made.
- Here we used only one machine learning algorithm, but want to make use of more than one.

REFERENCES

[1] Trust Aware Recommender Systems: A Survey on Implicit Trust Generation Techniques, Swati Gupta, Sushama Nahpal.

[2] Trust Aware Recommender System using swarm intelligence, Sambhav Yadav, Vikesh Kumar, Shreyam Sinha, Sushama Nahpal.

[3] Guibing Guo, Jie Zhang, Daniel Thalmann, Anirban Basu and Neil Yorke-Smith, From ratings to trust: an empirical study of implicit trust in recommender system, 2014.

[4] N. Lathia , S. Hiles and L Capra, trust based collaborative filtering, in trust management II.

[5] F Ricci, L. Rokach, B. Shapira and P.B. Kantor, Recommender System Handbook, 2011

[6] Stephen Paul Marsh, Formalising trust as a computational concept,2014

[7] G. Guo, J. Zhang, N.y. Smith, A novel recommendation model regularized with user trust and ratings.

[8] The International Federation for information processing, N. Lathia, S Hailes.