International Journal of Advanced Research in Engineering Technology & Science

ISSN: 2349-2819

Email: editor@ijarets.org Volume-6, Issue-4 April- 2019 www.ijarets.org

AN ANALYTICAL APPROACH ON SOCIAL RECOMMENDATION SYSTEMUSING DIFFERENT ALGORITHMS AND RECOMMENDS THE MOST RELEVANT ITEMS TO USERS

Gaurav Agarwal[1], Himanshu Bahuguna[2] and Ajay Agarwal[3]

[1]Research Scholar, Uttarakhand Technical University, Dehradun, (U.K.), INDIA

[2]Professor, Shivalik College of Engineering, Dehradun, (U.K.), INDIA

[3]Professor, Krishna Institute of Engineering & Technology, Ghaziabad, (U.P.), INDIA

ABSTRACT:

People are generally tending to buy products recommended to them by their colleagues, relatives and friends or the people they trust. When there was any doubt about the product; people try to use this as primary method of purchasing. Now the advantage of the digital era, the circle has enlarged to include online websites that utilize the recommendation system.Recommender systems use to analyze the data for different algorithms and suggest the most relevant items to users. It collects the past experience of a customer and depends on that, recommends the relevant products to users to buy.If the recommendation system recommends a more items to a user based on their interests, it will make a positive impact on the user experience and lead to incessant visits. Consequently, organizations/groupsthese days are building smart and intelligent recommendation systems by studying the past attitude of their users.

INTRODUCTION:

Todaymultiple online businesses rely on customer past experience, feedback and ratings. The specified feedback is particularly significant in the amusement and electronic commerce industry where all customers oriented servicearehighly impacted by these feedback and experience. Amazon relies on such evaluating information to power its recommendation systems to provide the best film and TV series suggestions that are customized and most relevant to the user.

How does a Recommendation System work?

(a) We can recommend items to a user according to popularity among all the users

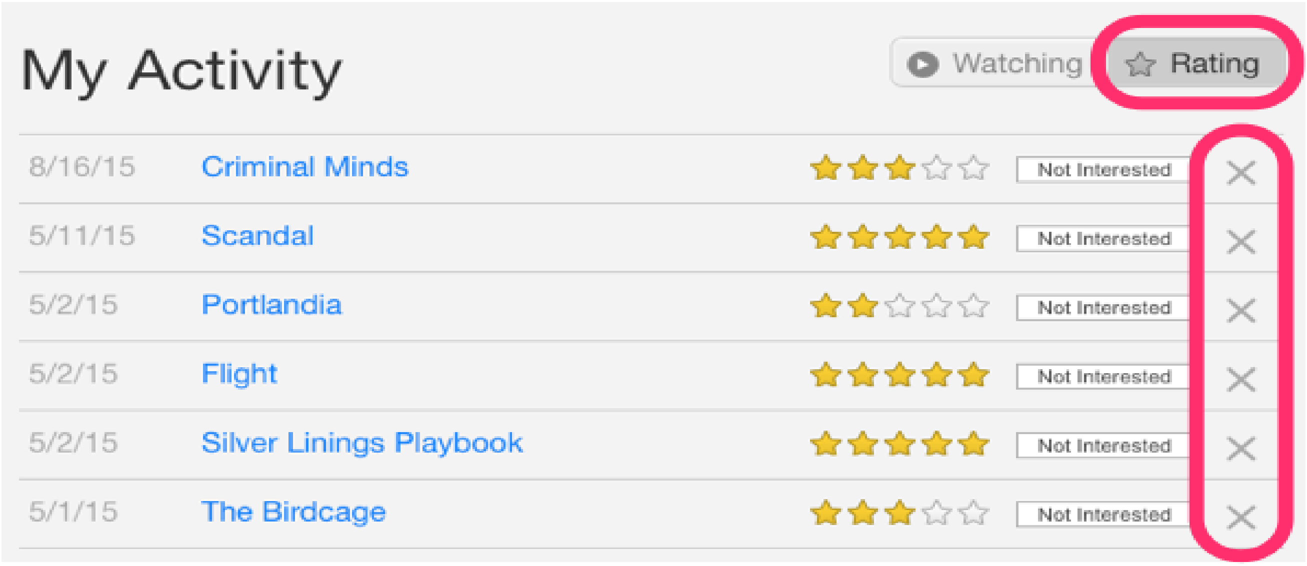
(b) We can intersect the users into various segments based on their desires.

The main issue is that we are unable to couturier recommendations based on the particular relevance of the users. It seems as Amazon is recommending you buy a laptop just because it's been bought by the most of the customers. But thankfully, Amazon (or any other big firm) does not recommend products using the above stated procedure. They use some customized approaches which leads them in recommending products more closely.

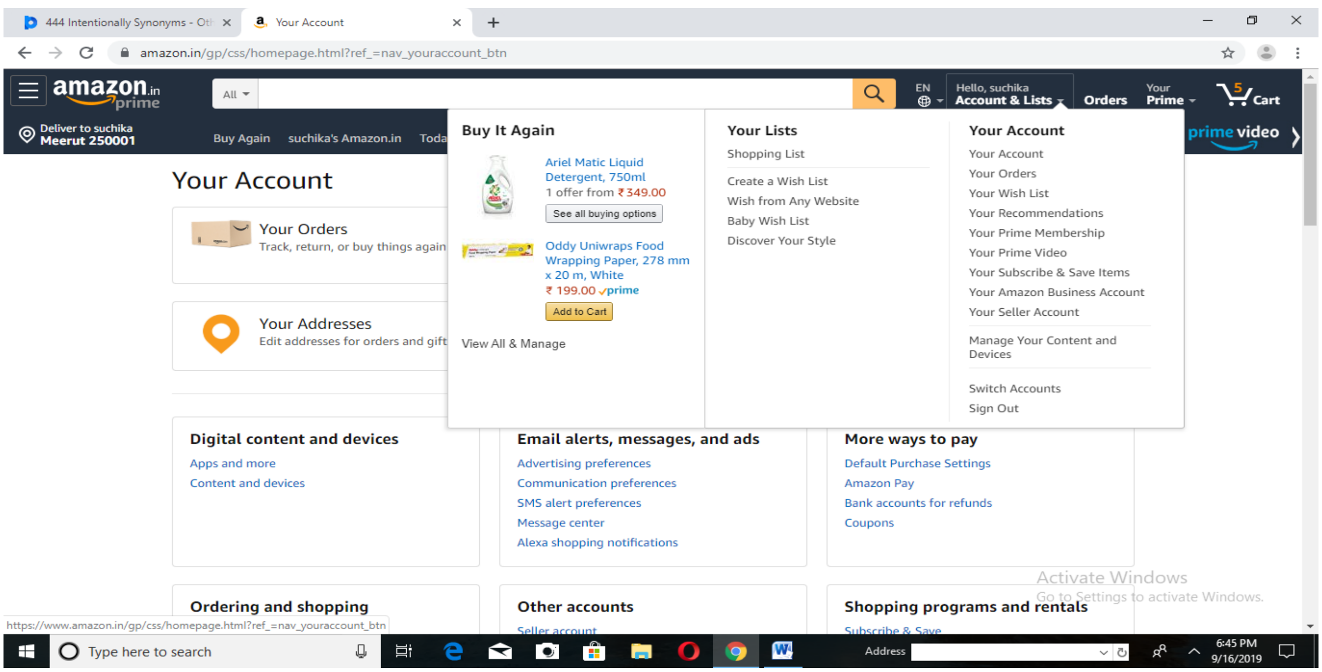
Now let's focus on how a recommendation system doings by going through the following steps.

1.1 Collection of Data

The first and most preferred step for building a recommendation system. Thereare two methods by the data can be collected: explicitly and implicitly. Explicit data is that information which providespurposely, i.e. input from the users such as movie ratings. Implicit data is information that is not provided intentionally but gathered from available data streams like search history, clicks, order history, etc.



In the above image, Netflix is collecting the data explicitly in the form of ratings given by user to different movies [3].



Here the orders and other things history of a user which was recorded by Amazon,this is an example of implicit mode of data collection.

DATA STORAGE

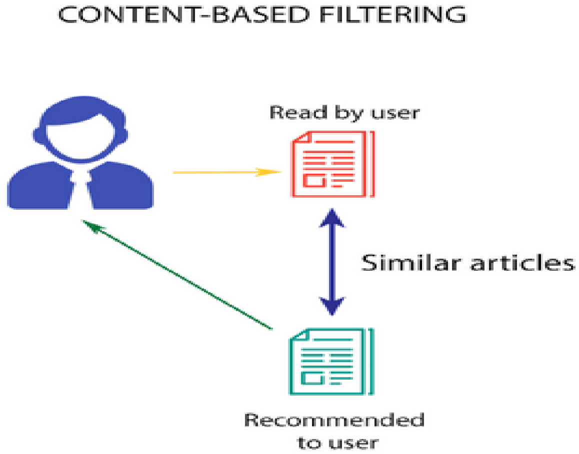
Data storage commands about the good recommendations of the model. For example, in a mobile phone recommender system, the more ratings users give to mobile phones, the best recommendations get for other users. The data plays an important role for deciding the type of storage that has to be used.

FILTERING THE DATA

When collecting and storing the data, we have to filter it and pull out the significantinformation required for final recommendations. There are lots of algorithms that help us make the filtering process simpler [2].

FILTERING BASED ON CONTENTS

This algorithm recommends products which are similar to the ones that a user has liked in the past.



Source: Medium

If people have alike the movie Initiate, this algorithm will recommend movies that fall under the same generation. Now how algorithm recognizes which generation to select and recommend movies from[2] [3]. **Consider the example of Netflix.** They are saving information associated to each user in vector form. These vectors contain the past behavior of the user, i.e. the movies liked/disliked by the user and the ratings given by them. This vector is known as the *profile vector*. All movies related information stored in another vector called the *item vector*. Item vector contains the details of every movie, like generation, editor, casting, etc. The algorithm based on content-based filtering finds the cosine angle between the item vector and profile vector [3]. Suppose A is the item vector and B is the profile vector, then the relation can be calculated as:

$$sim(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|}$$

Here on the bases of cosine value, which ranges from-1 to 1, the movies are arranging in descending order.The two below approaches is used for recommendations:

TOP-M APPROACH: where the top m movies are recommended.

RATING SCALE APPROACH: Where a threshold is set and all the movies above that threshold are recommended.

COLLABORATIVE FILTERING

Let if person P likes 3 movies, say Angoor, Hera Pheri and Andazapnaapna, and person Q likes Bawarchi, Angoor and GolMaal, and then they have almost similar interests. We can say with some certainty that P should like Gol Mall and B should like Hera Pheri. The collaborative filtering algorithm purpose is user behaviour for recommending items. Collaborative filtering algorithm is most commonly used algorithm in the businesses. It is not depend on other additional information. The other collaborating filtering techniques are:

USER-USER COLLABORATIVE FILTERING

This type of algorithm primarily finds the similar score between the users. Based on the similarity record, it then picks up the most similar users and recommends products to similar users which have liked or bought in past. *Source: Medium*

This algorithm finds the similarity between each user based on the ratings they have previously given to different movies.

Here we are showing the user's ratingswithin profile vector and based on that we are predicting the other user's ratings.

1. First we find the items rated by the users and based on the ratings, correlation between the users are calculated.
2. This algorithm, first calculates the similarity between every user and then based on similarity calculates the predictions.
3. Based on above prediction values, recommendations are made. for example:
   Consider the user-movie rating matrix:

User/Movie	x1	x2	x3	x4	x5	Mean User Rating
X	5	2	–	5	–	4
Y	–	5	–	2	4	4
Z	–	2	–	5	5	4

We have a user movie rating matrix. The similarities are showing between users (X, Z) and (Y, Z) in the above matrix. Common movies rated by Xand Z are movies x2 and x4 and by Y and Z are movies x2, x4 and x5.

The correlation between user X and Z is more than the correlation between Y and Z. Hence users X and Z have more similarity and the movies liked by user X will be recommended to user Z and vice versa.

This type of algorithm is time consuming as it involves calculating the similarity for each user and then calculating prediction for each similarity score [2] [8].

ITEM-ITEM COLLABORATIVE FILTERING

In this algorithm we computethe similarity between every pair of items.The similarity between each movie pair and based on that, we will recommend similar movies which are liked by the users in the past. This algorithm works collaborative filtering similar to user to user with minor change stead of taking the weighted sum of ratings of "user-neighbors", we take the weighted sum of ratings of "item-neighbors" [2]. The prediction is given by:

$$P_{u,i} = \frac{\sum_N (s_{i,N} * R_{u,N})}{\sum_N (|s_{i,N}|)}$$

Now we will find the similarity between items.

$$sim(i,j) = cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{||\vec{i}||_2 * ||\vec{j}||_2}$$

The similarity between each movie and the ratings, predictions are made and based on predictions, similar movies are recommended. For example

User/Movie	x1	x2	x3	x4	x5
X	5	2	3	5	5
Y	3	5	5	3	2
Z	–	2	–	4	5
Mean Item Rating	4	3	4	4	4

Here the rating of main item is the average of all the ratings given to a particular item. Instead of finding the user to user similarity as we saw earlier, we find the item-item similarity [2] [9].

Now we need to find such users who have rated those items and based on the ratings; similarity between the items is calculated. Let us find the similarity between movies (x1, x4) and (x1, x5). Common users who have rated movies x1 and x4 are X and Y while the users who have rated movies x1 and x5 are also X and Y. The similarity between movie x1 and x4 is more than the similarity between movie x1 and x5. Therefore based on these similar values, if any user searches for movie x1, they will be recommended movie x4 and vice versa [2].

CONCLUSION AND FUTURE WORK

In this work we present a framework for finding demographic attributes available in recommender systems datasets to be used for recommending relevant items to new users. Thedataset results showed that all attributes have almost the same influence. Conclusively, it seems that the demographic does not influence differently on ratings of users. Further research can be performed for enhance the results. A higher level of movie recommendation can be obtained by relating the movie genres to demographic attributes.

REFERENCES

1. Pan Zhou, Yingxue Zhou, Dapeng Wu, Hai Jin, Seni, “Differentially Private Online Learning for Cloud-Based Video Recommendation With Multimedia Big Data in Social Networks”, IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 18, NO. 6, JUNE 2016, pp. 1217-1219.
2. https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/
3. Hanhua Chen, Hai Jin, and Shaoliang Wu, “Minimizing Inter-Server Communications by Exploiting Self-Similarity in Online Social Networks”, IEEE TRANSACTIONS PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 27, NO. 4, APRIL 2016, pp. 1116 -1125.
4. R. Burke, “Hybrid web recommender systems,” in The Adaptive Web: Methods and Strategies of Web Personalization, P. Brusilovsky, A. Kobsa, and W. Nejdl, ed., Springer, 2007, pp. 377-408.
5. LailaSafoury and Akram Salah, “Exploiting User Demographic Attributes for Solving Cold Start Problem in Recommender System,” Lecture Notes on Software Engineering, Vol. 1, No. 3, pp. August 2013.
6. G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-ofthe-art and possible extensions,” IEEE Trans. on Knowledge and Data Engineering, vol. 17, pp. 734-749, June 2005.
7. J. B. Schafer, D. Frankowski, and J. Herlocker, “Collaborative Filtering Recommender Systems,” in The Adaptive Web: Methods and Strategies of Web Personalization, P. Brusilovsky, A. Kobsa, and W. Nejdl, ed., Springer, 2007, pp. 291-324.
8. D. Almazro, G. Shahatah, L. Albdulkarim, M. Kherees, R. Martinez, and W. Nzoukou. “A Survey Paper on Recommender Systems,” arXiv preprint arXiv: 1006.5278, Dec. 2010.
9. A. Said, T. Plumbaum, W. E. De Luca, and S. Albayrak, “A comparison of how demographic data affects recommendation,” in Proc. 19th international conference on User modeling, adaption, and personalization, 2011.
10. Wikipedia, “Recommender System” [online] available at:https://en.wikipedia.org/wiki/Recommender_system Hybrid_Recommender_Systems.
11. Ruchika, Singh Ajay Vikram, Sharma Dolly, “Evaluation Criteria for Measuring the Performance of Recommender Systems, IEEE 2015.
12. Xiao-Lin Zheng, Senior, Chao-Chao Chen, Jui-Long Hung, Wu He, Fu-Xing Hong, and Zhen Lin, “A Hybrid Trust-Based Recommender System for Online Communities of Practice”,IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, VOL. 8, NO. 4, OCTOBER-DECEMBER 2015.
13. A. Sridharan, Y. Gao, K. Wu, and J. Nastos, “Statistical behavior of embeddedness and communities of overlapping cliques in online social networks,” in Proc. INFOCOM, 2011, pp. 546–550
14. M. Jones, “Introduction to approaches and algorithms for Recommendation systems”December 12, 2013 [online] available: https://www.ibm.com/developerworks/library/os-recommender.
15. LinkeGuo, Chi Zhang, and Yuguang Fang, “A Trust-Based Privacy-Preserving Friend Recommendation Scheme for Online Social Networks”, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, JULY/AUGUST 2015, pp. 413-415.
16. S. Wen, W. Zhou, J. Zhang, Y. Xiang, W. Zhou, and W. Jia , “Modeling propagation dynamics of social network worms,” IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 8, pp. 1633–1643, Aug. 2013.
17. R. Chen, E. K. Lua, and Z. Cai, “Bring order to online social networks,” in Proc. IEEE INFOCOM, 2011, pp. 541–545..
18. Amazon.com [Online]. Available: http://en.wikipedia.org/wiki/Amazon.com
19. Esther Palomar, Lorena González-Manzano, AlmudenaAlcaide, Álvaro Galán,“Implementing a privacy-enhanced attribute based credential system for online social networks with co-ownership management”, The Institution of Engineering and Technology 2016, pp. – 60- 68.
20. A. Samuel, M. I. Sarfraz, and H. Haseeb, “A framework for composition and enforcement of privacy-aware and context-driven authorization mechanism for multimedia big data,” IEEE Trans. Multimedia, vol. 17, no. 9, pp. 1484– 1494, Sep. 2015.
21. Yuan Cheng, Jaehong Park, and Ravi Sandhu, “An Access Control Model for Online Social Networks Using Userto-User Relationships”, IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, VOL. 13, NO. 4, JULY/AUGUST 2016, pp. 424-425