Collaborative Filter Based Recommender Systems

Sabarish Gopalakrishnan

Abstract— **Recommender systems have been used quite successfully by companies to nudge customers and users into consuming content based on their past behavior. The objective and an desired outcome of a good recommender system would be an enhanced user experience. These systems are widely used to suggest a book or movie after a user has made a similar purchase. There are two ways to build a recommender system - Collaborative filtering and content based filtering. In this project we describe in detail two of the methods used to build a collaborative recommender system; the user based and the item based collaborative filtering. The user based collaborative filtering technique recommends a new or unused product to a user based on his similarity to other users who are already using the product. The item based collaborative filtering on the other hand tries to find a relation between products and then goes on to recommend a new or unused product to a user based on the products that the user is already using. We will be evaluating our model on the MovieLens[8] 1M data set made available by GroupLens[8] at the University of Minnesota. This dataset contains nearly 1 million ratings by 6000 users for around 3000 movies.**

I. INTRODUCTION

The age of internet has paved the way to a more data driven approach to more and more things that shape our world. The ability to store vast amounts of data without having to restrict it to local disks means that a abundant data is at our disposal for using it to improve productivity and make systems more efficient.

Recommender systems suggest products and items to users based on their previous history[9], [10], [12]. The history, in this context, is the rating to the products that have previously been used by the user. There are two broad categories that recommender systems fall into - collaborative filtering based[15], [2], [3] and content-based[16]. Computational speed is one of the most critical aspects of a recommender system. A good recommender should be able to traverse through millions of rows of ratings and calculate the predicted rating for the user in real time. The ratings that users give to their products are used as input to a system that predicts the rating the user would give to an unseen or unused product. The higher the predicted rating, the greater is the incentive for recommending the item to the user.

In this paper, we explore the user based and item based collaborative filtering technique on the MovieLens[1] data set. MovieLens is a on-line community where a user can rate movies that the user has previously watched and get recommendations in return. The implementation in this paper will be restricted to the MovieLens 1M data which contains 1 million ratings for 6000 users for around 3000 movies.

II. RELATED WORK

Recommender system is one of the many positives that has come from the internet age where historic data is available to us to traverse and make suitable suggestions to users. Recommendation systems attempt to shape the user experience in a more personalized way. Due to sudden availability of a lot of data and an almost inexhaustible supply of computation power, recommender systems are being deployed almost in every domain.They are especially very prominently visible in the field of online shopping and internet based video streaming services. The GroupLens organization built a system that sifted through Usenet news data.[5] Music aficionados used Ringo to browse new music as per their likings.[11] The survey by Beel et al.[17] inspected work on recommender systems in the last sixteen years. They found that 55% of the papers published were on the content based technique while 16% was based on the collaborative filtering technique. The user based collaborative filtering[7], [13] is based on finding similarity between a pair of users and then suggesting an unknown item to a user. The item based collaborative filtering[6], [7] compares two items and then predicts whether a user will like to use a previously unused item or not. The similarity between two users or items are calculated using either the Pearson correlation[4] or the cosine similarity[4]

III. COLLABORATIVE FILTERING

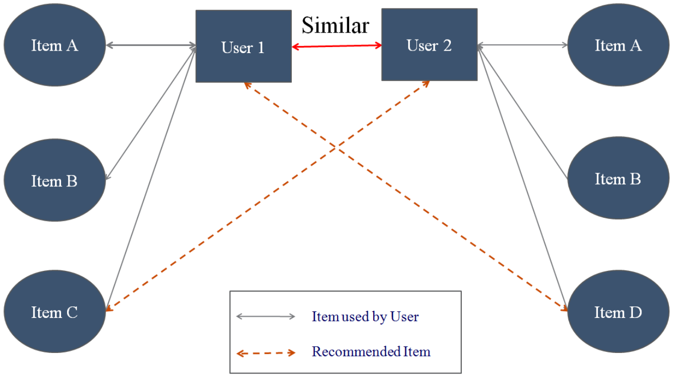
Collaborative filtering predicts the item preferences of a user based on his past ratings. The heart of the collaborative filtering technique lies in calculating the similarity between two users or items. The greater the similarity, the more the chance that two users might like the same item. There are two types of collaborative filtering techniques - user based collaborative filtering[7], [13] and item based collaborative filtering[6], [7]. They can be used in different scenarios for recommending items to the user.

A. User Based Collaborative Filtering

The user based collaborative filtering technique finds the similarity between the user who has never used an item and all the users who have used and rated the item. It then predicts the rating that the unused item will be given by the user.

[1] https://grouplens.org/datasets/movielens/

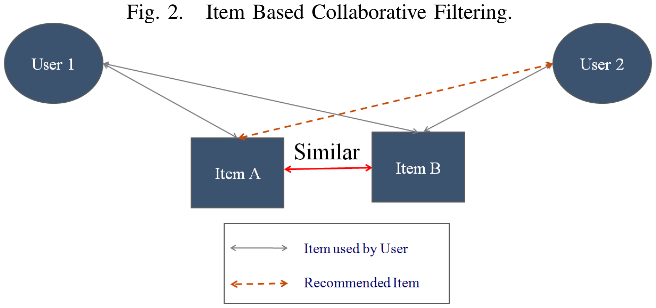
Fig. 1. User Based Collaborative Filtering.



B. Item Based Collaborative Filtering

The item based collaborative filtering technique is quite similar to the user based collaborative filtering technique. The difference is that here instead of comparing the similarity of two users, the similarity of two items is compared. If the user uses a range of items and a majority of them are similar to a particular unused item, then the unused item would be recommended to the user.

Fig. 2. Item Based Collaborative Filtering.



IV. CALCULATIONS

A. Similarity Metric

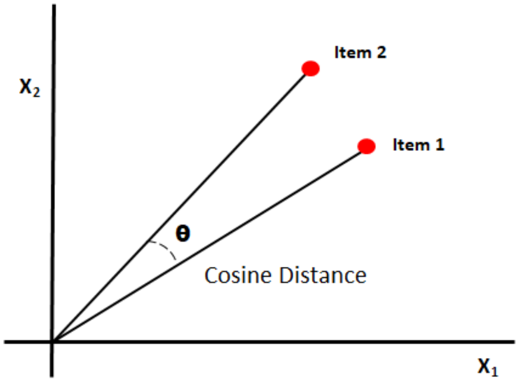
This paper will describe two ways to ways to calculate the similarity between a pair of users and items.

1) Pearson's Correlation: The Pearson's correlation measures the correlation between two users or items. This correlation ranges from -1 to 1 and gives a good measure of the similarity between the pair of users or items. Negative correlations indicate opposite user behavior whereas positive correlation indicates similar user behavior. The Pearson coefficient of correlation(r) is given below:

$$r = \frac{\sum(x-\overline{x}).(y-\overline{y})}{\sqrt{\sum(x-\overline{x})^2.(y-\overline{y})^2}} \tag{1}$$

2) Cosine Similarity: The cosine similarity projects the users and items on to a vector space where the distance is the rating metric of the user of the item. Two users or items are similar if the cosine of the angle between them is small.

Fig. 3. Cosine similarity



$$similarity(x,y) = cos(\theta) = \frac{(x).(y)}{||x||.||y||} \tag{2}$$

B. Recommendation Condition

To recommend any item for a user, the predicted rating of the user must be greater than the average rating of the user. The average rating for all the items used by the user is μ. Given a set of ratings for items by *n* users, the item is recommended to the user if

$$\frac{\sum_{i=1}^{n-1} r_i.x_i}{\sum_{i=1}^{n-1} r_i} > \mu \tag{3}$$

In 3, the numerator will contain all possible user pairs for a particular user. This results in a lot of time being used by the algorithm for traversing through the similarity matrix and also for the calculation. Sarwar et al.[1] show that restricting the calculations to the top n nearest neighbor gives a much enhanced performance by converging to low error in minimal amount of time.

V. RESULTS

This paper implements the collaborative filtering technique on the Movie Lens dataset. Movie Lens[8] is a online community where a user can go and rate movies and get movie recommendations in return. The MovieLens 1M data set has around 6000 users rating over 3000 movies resulting in over a million ratings. Every user gives a rating between 1 and 5 where 1 is the lowest rating while 5 indicates the best rating. 5 ratings of 20% of the data is used for testing the model. The rest of the data is used as the training data set for the model.

The Pearson Correlation and the Cosine distance are the two metrics used in this implementation of the model. The Root Mean Squared Error(RMSE) is used as a metric to measure the model.

Fig. 4. Distribution of User Ratings.

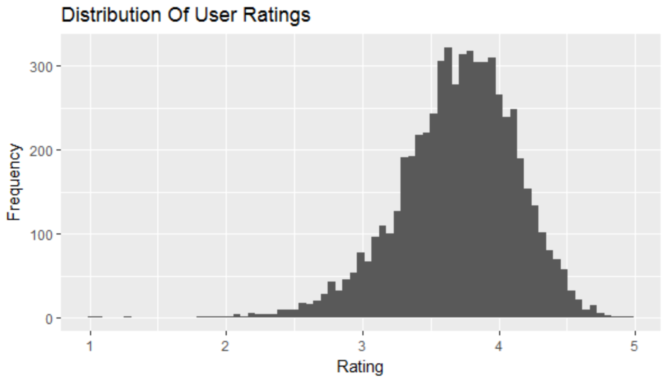
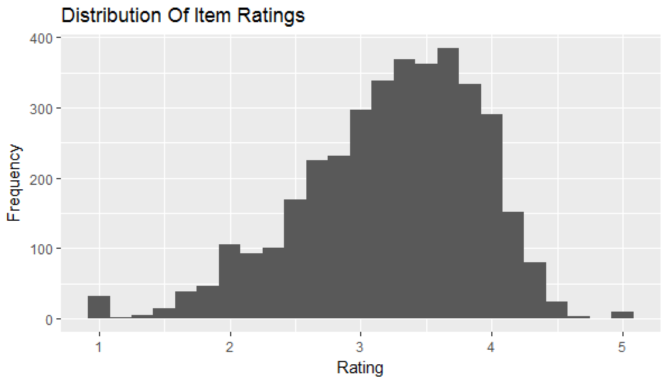


Fig. 5. Distribution of Item Ratings.



It is first necessary to test the distribution of the data of the ratings of given by users to items and also the ratings that each item has received to check for any inherent bias in the data. The plots in Fig. 4 and Fig. 5 show a histogram of the users' average rating item's average rating. Both the histograms resemble a nearly normal distribution and hence one can arrive at the conclusion that there is not much bias in the data.

Fig. 6. User Based RMSE Plot.

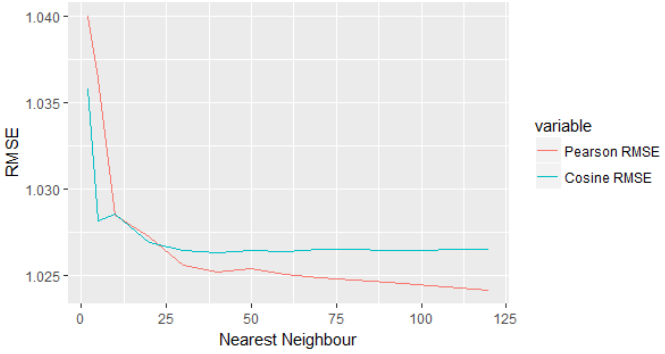
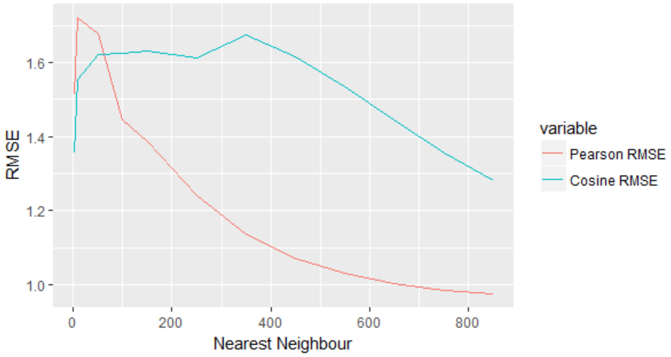


Fig. 7. Item Based RMSE Plot.



The RMSE for the user based collaborative filtering shown in Fig. 6 converges at 25 nearest neighbors. The Pearson correlation similarity metric performs much better than the cosine similarity beyond 25 nearest neighbors. The RMSE for the item based filtering shown in Fig. 7 returns a RMSE value similar to the user based technique but at a nearest neighbor count of over 700. This shows that the user based filtering technique is performing better in terms of memory required to build the model. The Pearson similarity based item filtering technique outperforms the cosine similarity based item based filtering system as can be seen in Fig. 7.

TABLE I

RMSE - User Based Method

Nearest Neighbor	Pearson	Cosine
2	1.040048	1.035777
5	1.03642	1.028149
10	1.02848	1.028579
20	1.027212	1.026925
30	1.025589	1.026445
40	1.025192	1.02632
50	1.02539	1.026451
60	1.025049	1.026398
70	1.024858	1.026496
80	1.024711	1.026527
90	1.024621	1.02647
120	1.024125	1.026484

TABLE II

RMSE - Item Based Method

Nearest Neighbor	Pearson	Cosine
2	1.512965	1.354006
10	1.721195	1.554024
50	1.676879	1.621224
100	1.444051	1.623866
150	1.385073	1.629426
250	1.241984	1.612857
350	1.137481	1.674524
450	1.071849	1.613632
550	1.030929	1.534524
650	1.002076	1.443563
750	0.9835508	1.356667
850	0.9746966	1.280283

VI. CONCLUSIONS

This paper demonstrates the working of a user and item based collaborative filtering. The Pearson and cosine similarity were used to measure how closely two users and items are related. The user based collaborative filtering requires very less nearest neighbors to converge to a low value of error as compared to the item based filtering technique. The time taken for building the similarity matrix of users and items is time and memory intensive and a small count nearest neighbors that gives the least error is a model that is more preferable. In the experiments that we ran, the user based collaborative filtering technique converged to a low error much faster than the the item based filtering technique as it was using only a fraction of the nearest neighbors as compared to the item based filtering technique.

REFERENCES

[1] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, Analysis of Recommendation Algorithms for e-Commerce, in Proceedings of the 2Nd ACM Conference on Electronic Commerce, New York, NY, USA, 2000, pp. 158167.

[2] R. Zhang, Q. d Liu, Chun-Gui, J. X. Wei, and Huiyi-Ma, Collaborative Filtering for Recommender Systems, in 2014 Second International Conference on Advanced Cloud and Big Data, 2014, pp. 301308.

[3] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, Collaborative Filtering Recommender Systems, Found. Trends Hum.-Comput. Interact., vol. 4, no. 2, pp. 81173, Feb. 2011.

[4] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems (TOIS), vol. 22, no. 1, pp. 553, Jan. 2004.

[5] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, GroupLens: Applying Collaborative Filtering to Usenet News, Commun. ACM, vol. 40, no. 3, pp. 7787, Mar. 1997.

[6] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, Item-based Collaborative Filtering Recommendation Algorithms, in Proceedings of the 10th International Conference on World Wide Web, New York, NY, USA, 2001, pp. 285295.

[7] Peng Yu: ”User-Based and Item-Based Collaborative Filtering Recommendation Algorithms Design”, International Conference on Computer Science and Network Technology (ICCSNT 2015)

[8] B. N. Miller, I. Albert, S. K. Lam, J. A. Konstan, and J. Riedl, MovieLens Unplugged: Experiences with an Occasionally Connected Recommender System, in Proceedings of the 8th International Conference on Intelligent User Interfaces, New York, NY, USA, 2003, pp. 263266.

[9] L. Niu, J. Wu, and Y. Shi, Second-order Mining for Active Collaborative Filtering, Procedia Computer Science, vol. 4, no. Supplement C, pp. 17261734, Jan. 2011.

[10] X. Su and T. M. Khoshgoftaar, A Survey of Collaborative Filtering Techniques, Adv. in Artif. Intell., vol. 2009, p. 4:24:2, Jan. 2009.

[11] U. Shardanand and P. Maes, Social Information Filtering: Algorithms for Automating Word of Mouth, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 1995, pp. 210217.

[12] Ekstrand M. , Riedl J. and Konstan J. (2011), ”Collaborative Filtering Recommender Systems”, Foundations and Trends in HumanComputer Interaction: Vol. 4: No. 2, pp 81-173

[13] Cai, L., Yao, G. (2015). User-Based and Item-Based Collaborative Filtering Recommendation Algorithms Design

[14] Ricci F., Rokach L., Shapira B. (2011) Introduction to Recommender Systems Handbook. Springer, Boston, MA

[15] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, Using Collaborative Filtering to Weave an Information Tapestry, Commun. ACM, vol. 35, no. 12, pp. 6170, Dec. 1992.

[16] M. J. Pazzani and D. Billsus, Content-Based Recommendation Systems, in The Adaptive Web, Springer, Berlin, Heidelberg, 2007, pp. 325341.

[17] Beel, J., Gipp, B., Langer, S. et al. Int J Digit Libr (2016) 17: 305. https://doi.org/10.1007/s00799-015-0156-0