COSTOMIZATION OF RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING ALGORITHM ON CLOUD USING MAHOUT

Swati Pandey[1], T. Senthil Kumar[2]

[1]*M.Tech Student, Computer Science & Engineering, Amrita University, Tamilnadu, India*
[2]*Assistant Professor, Computer Science & Engineering, Amrita University, Tamilnadu, India*

Abstract

Recommendation System helps people in decision making regarding an item/person. Growth of World Wide Web and E-commerce are the catalyst for recommendation system. Due to large size of data, recommendation system suffers from scalability problem. Hadoop is one of the solutions for this problem. Collaborative filtering is a machine learning algorithm and Mahout is an open source java library which favors collaborative filtering on Hadoop environment.

Keywords: *Collaborative Filtering, Distributed System, Hadoop, Machine Learning, Mahout, Map-Reduce*

---***---

1. INTRODUCTION

In fast growing world, since time is on its heel, people do not want to go shop by shop and buy the best item according to their requirement. To save time, everyone wants to buy things in home in reasonable cost. They prefer online shopping, online suggestion for an item, so that they can take a decision on a particular item which may be suitable for a particular. In such scenario Recommendation System plays a vital role. When the questions are "Whether I will like this item", "I want to buy an item of a particular type which suits according to my taste", "Can you suggest me an item which we may like?", the feasible answers can be obtained through recommendation system. It helps in recommending items of a similar type as well as predicting an item, whether it will be liked by user or not.

For recommendation, our proposed system uses collaborative filtering machine learning algorithm. Collaborative filtering (CF) is a machine learning algorithm which is widely used for recommendation purpose. Collaborative filtering finds nearest neighbor based on the similarities. The metric of collaborative filtering is the rating given by the user on a particular item. Different users give different ratings to items. Users, who give almost same rating to items, are the nearest neighbors. In case of User based collaborative filtering, based on the ratings given by the users, nearest neighbors has been find. Item based collaborative filtering predicts the similarity among items. To recommend an item, items which are liked by the user in his past have been found. Item which is similar to those items has been recommended [5].

Internet contains a huge volume of data for recommendation purpose. Due to size of data, if recommendation computation has been done in single system, then performance may degrade, and we cannot find an efficient solution. Hence we require distributed environment so that computation can be increased and performance of recommendation system gets improve. An open source cloud environment Hadoop provides distributed environment [3]. Due to Map-Reduce programming, it provides result efficiently and effectively in less amount of time [2]. Proposed system has been modeled on Hadoop. Mahout is an open source java library which favors Collaborative Filtering. Mahout favors Hadoop for recommendation.

Rest of the paper has been arranged in different sections. Section 2 briefly describe about related work that has been done. Section 3 elaborates the proposed model for Recommendation System. Section 4 describes data set and results. Section 5 focuses on conclusion and future enhancement.

2. RELATED WORK

Zhi-Dan Zhao and Ming-Sheng Shang [9] have used based Collaborative Filtering using Hadoop as distributed framework. The approach is scalable but the response time taken for a single user could not be reduced. Carlos E. Seminario a David C. Wilson [10] use Mahout for recommendation. Use of mahout for Collaborative Filtering has enhanced the accuracy.

Xiao Yan Shi, Hong Wu Ye and Song Jie Gong [23] integrated user based and item based collaborative filtering algorithm to increase performance. Manos Papagelis and Dimitris Plexousakis [11] did qualitative analysis on user based and item based collaborative filtering with implicit and explicit ratings. According to their analysis, prediction based on explicit rating is better than implicit rating. Trouong Khanh Quan, Ishikawa Fuyuki, Honiden Shinichi [12] clustered the items on stability of user similarity and applied Collaborative Filtering. Final clustering of the items may be locally optimal but the prediction accuracy has been increased.

Yanhong Guo, Xuefen Cheng, Dahai Dong, Chunyu Luo and Rishuang Wang [16] used CF based on trust factor. For calculating similarities Cosine correlation and Pearson correlation has been used. They found that CF based on trust factor is better than traditional CF. Mustansar Ali Ghazanfar and Adam Prugel Bnnett [19] had built hybrid recommendation system which combines the rating, feature and demographic information of item. Yajie Hu, Ziqi Wang, Wei Wu, Jianzhong Guo and Ming Zhang [22] used Semantic distance measurement and considered the features of movie. They are able to give a list of recommended movie along with stars of that movie. Dhoha Almazro, Ghadeer Shahatah, Lamia Albbulkarim, Mona Kherees, Romy Martinez and William Nzoukou [13] collected Demographic information of user, clustered all items and then combined both user-based and item-based.

Hee Choon Lee, Seok Jun Lee, Young Jun Chung [14] did a study on improved Collaborative filtering algorithm. They have used Neighborhood CF (NBCFA), Correspondence mean algorithm (CMA). They found that the preference prediction performance of CMA is better than NBCFA. Kai Yu, Xiaowei Xu, Jianhua Tao, Martin Aster and EansPeter Kriegel [17] proposed instance selection techniques for memory based collaborative filtering. It reduces the storage requirement of training data, but there is scalability problem for big data. Alexandros Karatzoglou, Alex Smola and Markus Weimer [21] proposed CF with hashing using Ït-intensive loss function, Huber loss function. This model can be scaled to bigger data-set on large server and to still data-set on small machines, but time complexity is the problem. Aristomenis S. Lampropoulos and George A. Tsihrintzis [7] did survey on recommendation machine learning algorithms. Sarwar B. et al.[5], Mukund Deshpande et al.[6] and Badrul Sarwar et al.[20] worked on item based collaborative filtering for recommendation.

3. PROPOSED SYSTEM

This paper is focusing to customize a recommendation system using Collaborative Filtering (CF) and clustering techniques. For our approach, we use Apache Hadoop (a widely used open source platform which implements Map-Reduce programming model) and Apache Mahout (An Open source library of scalable data mining algorithms, where it forms a core of the distributed recommender module).

The proposed algorithm is worked in two parts. In first phase, we obtain the results for recommendation by applying User-based CF and Item-based CF separately. In second phase, we combine the results obtain from user-based CF and item-based CF.

3.1 User Based CF

The dataset is firstly loaded into Hadoop distributed file system (HDFS). Then we perform User-based CF using Mahout.

We take rating matrix, in which each row represents user and column represents item, corresponding row-column value represents rating which is given by a user to an item. Absence of rating value indicates that user has not rated the item yet. There are many similarity measurement methods to compute nearest neighbors. We have used Pearson correlation coefficient to find similarity between two users. Hadoop is used to calculate the similarity. The output of the Hadoop Map phase i.e. userid and corresponding itemid are passed to reduce phase. In reduce phase, output has been generated and sorted according to userid. Output again has been stored in HDFS. The architecture diagram for User-based CF can be shown in diagram 1.

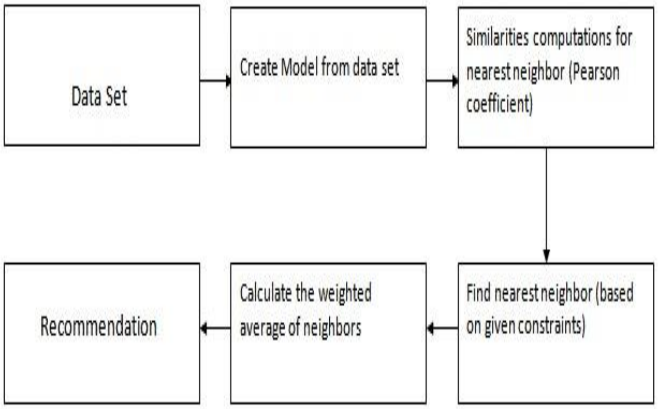
Data Set
Create Model from data set
Similarities computations for nearest neighbor (Pearson coefficient)
Find nearest neighbor (based on given constraints)
Calculate the weighted average of neighbors
Recommendation

Fig-1: User-based Collaborative Filtering

3.2 Item Based CF

Dataset is loaded into HDFS, then using Mahout we performs Item-based CF.

Past information of the user, i.e. the ratings they gave to items are collected. With the help of this information the similarities between items are builded and inserted into item to item matrix. Algorithm selects items which are most similar to the items rated by the user in past. In next step, based on top-N recommendation, target items are selected. The processing of Item-based CF can be described through figure 2.

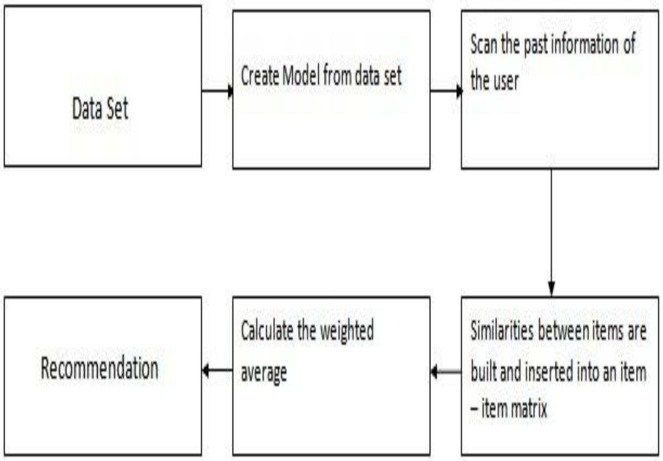
Data Set
Create Model from data set
Scan the past information of the user
Similarities between items are built and inserted into an item – item matrix
Calculate the weighted average
Recommendation

Fig-2: Item-based Collaborative Filtering

3.3 Combined Result

In case of user-based CF, if nearest neighbors (similar users) are not in enough number, i.e. taste of target user is not similar to many users then the recommending an item for that particular user may be not accurate. Item-based CF is based on the past information of the user, so it works well in such cases. The user-based and item-based results that are stored in HDFS are taken and then we combine these results based on threshold value. Suppose the threshold by increasing threshold value, probability of recommending correct item get increases because it has been liked by many users. Flow can be seen in figure 3.

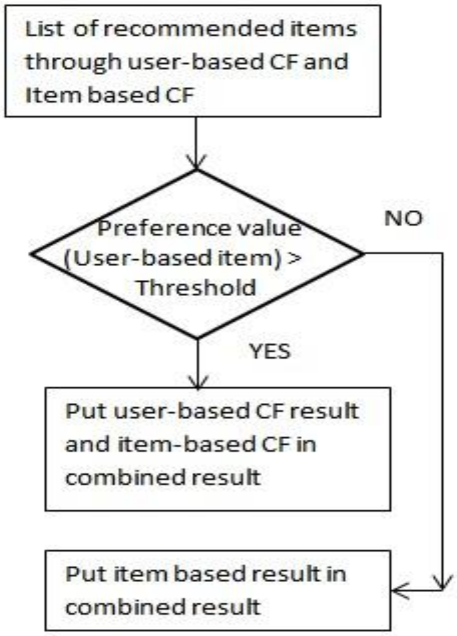
List of recommended items through user-based CF and Item based CF
Preference value (User-based item) > Threshold
NO
YES
Put user-based CF result and item-based CF in combined result
Put item based result in combined result

Fig-3: Item-based Collaborative Filtering

4. EXPERIMENT AND RESULT

4.1 Dataset

For experiment, we have used MovieLens dataset of size 1M. The dataset contains 10000054 ratings and 95580 tags applied to 10681 movies by 71567 users. There are three files, movies.dat, ratings.dat and tags.dat. Ratings data file has atleast three columns; those are UserId, given by user to movie.

4.2 Result Analysis

For movie recommendation, important factor is the list of recommended items as soon as possible. Since we are using Hadoop, speedup and efficiency varies as number of nodes varies. To analyze this we have obtained the number of movies which are recommended as threshold changes, Speedup and efficiency according to number of nodes.

4.3 Movies versus Threshold

Threshold value indicates minimum number of users who like an item which has to be recommended. It means when we increase threshold, accuracy in recommending an item increases because more number of users like that item. When the threshold value increases, the number of items recommended by the users will be reduced. Even though that value is too small, it is the relevant one. Hence such items can be readily recommended to users without any further processes. Comparison graph based on threshold value is given by Figure 4.

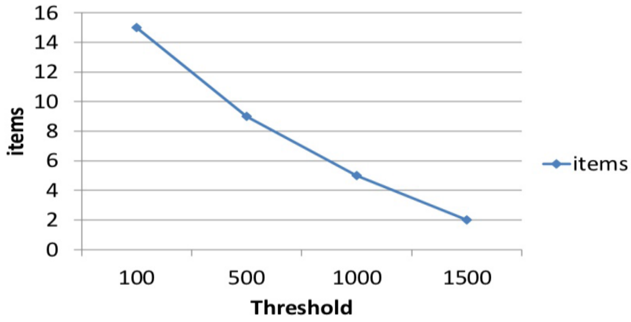

Fig-4: Comparison graph based on Threshold value

4.4 Speed Up

Speedup is the measure of the performance. It is defined as the ratio between sequential execution time and the parallel execution time.

$$\text{Speed Up} = T(1) / T(b)$$

Where T(1) states the execution time taken by single processor. T(b) is the execution time taken by 'b' no. of processor. While running algorithm in Hadoop framework, speedup varies as numbers of nodes vary. As we increase no of nodes, speedup increases. It can be seen in figure 5.

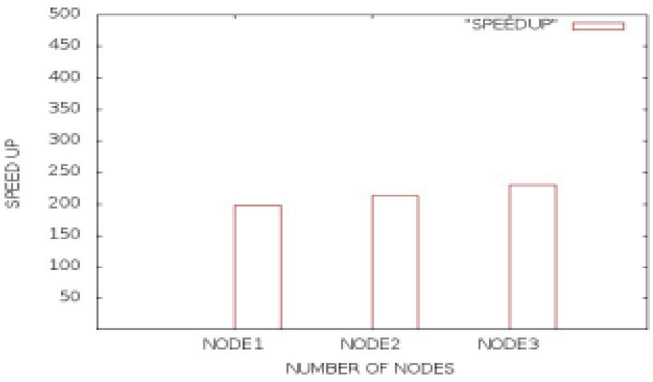

Fig-5: Speed Up with respect to no. of nodes

4.5 Efficiency

It is denoted as the usage of the computational resources. This is the ration between the speedup and no. of processors.

Efficiency Ratio = T(1) / b.T (b)

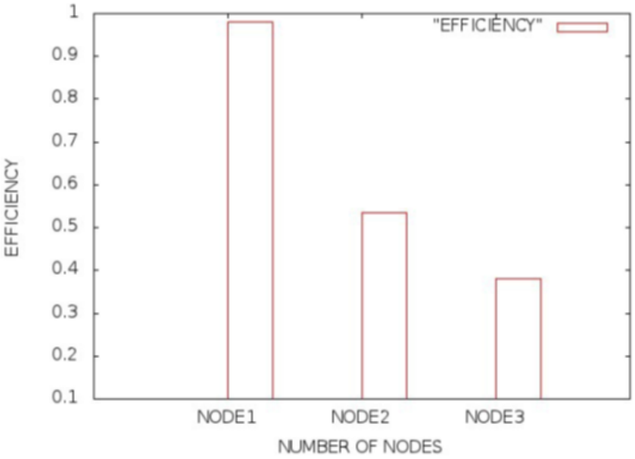

Fig-6: Efficiency with respect to no. of nodes

From the above graph (figure 6) it is visible that if the number of processors is one, the efficiency increases because that processor is fully utilized for the particular program. When the processors increases the efficiency decreases, that means we can utilize the processors for different purposes. This is the one main advantage of the distributed system.

5. CONCLUSIONS

This paper presented combined Collaborative Filtering using Mahout on Hadoop for movie recommendation. By combining User-based and Item-based CF, accuracy of the results gets improve. Hadoop has increased throughput. Because of multiple computer nodes, time taken for solving problem has been reduced. Mahout is feasible for handling large amount of structured data. But now a day's data are more unstructured. Hadoop using Mahout can handle big data statistically. To handle real time data randomly, HBASE, HIVE are the better solutions.

REFERENCES

[1]. Andrew S Tanenbaum and Maarten van Steen," Distributed Systems: Principle and Paradigms", in Pearson Prentice Hall, 2nd edition, May 2005.
[2]. Jeffrey Dean and Sanjay Ghemawat, "Map-Reduce: Simplified data processing on large clusters", to appear in OSDI 2004.
[3]. Chuck Lam, "Hadoop in Action", Manning publication, 2010.
[4]. J. B. Schafers, J. Konstan and J. Riedi, "Recommendation Systems in e-commerce", 1st ACM conference on Electronic Commerce ACM press, pp. 158-166, 1999.
[5]. Sarwar B. and Karypis, "Item based collaborative filtering algorithms" in 10th International World Wide Web conference, pp 285-295, 2001.
[6]. Mukund Deshpande and George Karypis, "Item based top-N recommendation algorithm", in ACM Transactions Information Systems, volume 22, no. 1, pp 143-177, 2004.
[7]. Aristomenis S. Lampropoulos and George A. Tsihrintzis, "A survey approach to designing recommendation system", Springer 2013.
[8]. Loren Terveen and Will Hill, "Beyond recommender systems: Helping people help each other", In HCI in The New Millennium, Jack Aarrdl, Addison Wesley, 2001 page 2 of 21.
[9]. Zhi-Dan Zhao and Ming-Sheng Shang, "User based collaborative filtering recommendation algorithm an hadoop", IEEE 2012.
[10]. Carlos E. Seminario and David C. Wilson, "Case study evaluation of mahout as a recommender plateform", presented in workshop on recommendation utility evaluation: Beyond RMSE, held in conjunction with ACM in Ireland, 2012.
[11]. Manos Papagelis and Dimitris Plexousakis, "Qualitative analysis of user based and item based prediction algorithms for recommendation agents", Science Direct 2005.nb
[12]. Trouong Khanh Quan, Ishikawa Fuyuki, Honiden Shinichi, "Improving accuracy of recommendation system by clustering item based on stability of user similarity", IEEE 2006.
[13]. Dhoha Almazro, Ghadeer Shahatah, Lamia Albbulkarim, Mona Kherees, Romy Martinez, William Nzoukou, "A survey paper on recommendation system", ACM 2010.
[14]. Hee Choon Lee, Seok Jun Lee, Young Jun Chung, "A

study on improved collaborative filtering algorithm for recommendation system", IEEE 2007.

[15]. Wu Yueping and Zheng Jianguo, "A research of recommendation algorithm based on cloud model", IEEE 2010.

[16]. Yanhong Guo, Xuefen Cheng, Dahai Dong, Chunyu Luo, Rishuang Wang, "An improved collaborative filtering algorithm based on trust in e-commerce recommendation system", IEEE 2010.

[17]. Kai Yu, Xiaowei Xu, Jianhua Tao, Martin Aster, Eans-Peter Kriegel, "Instance selection techniques for memory based collaborative filtering", SIAM.

[18]. Dilek Tapucu, Seda Kasap, Fatih Tekbacak, "Performance comparision of combined collaborative filtering algorithm for recommender system", IEEE 2012.

[19]. Mustansar Ali Ghazanfar and Adam Prugel Bnnett, "A scalable, accurate hybrid recommender system".

[20]. Badrul Sarwar, George Karypis, Joseph Konstan, John Riedl, "Item based collaborative filtering recommendation algorithms", www10, in Hong Kong, ACM 2001.

[21]. Alexandros Karatzoglou, Alex Smola, Markus Weimer, "Collabrative filtering on a budget", Appear in proceedings of the 13th international conference on Artificial Intelligence and Statistics 2010, Italy.

[22]. Yajie Hu, Ziqi Wang, Wei Wu, Jianzhong Guo, Ming Zhang, "Recommendation for movies and stars using YOGA and IMDB", IEEE 2010.

[23]. Xiao Yan Shi, Hong Wu Ye, Song Jie Gong, "A personalized recommender integrating item based and user based collaborative filtering", IEEE 2008

BIOGRAPHIES

Swati Pandey is pursuing M.Tech (CSE) in Amrita University, Coimbatore. She holds first rank in M.Tech. Her area of interest is Machine Learning and Distributed Computing.

Dr. T. Senthil Kumar is Assistant Professor in Amrita University. His area of interest is Image Processing and Distributed Computing.