Proceedings of 2017 Tenth International Conference on Contemporary Computing (IC3), 10-12 August 2017, Noida, India

An Improved Content Based Collaborative Filtering Algorithm For Movie Recommendations

Ashish Pal, Prateek Parhi and Manuj Aggarwal
Department of Computer Science
ARSD College, University of Delhi
New Delhi, India
iamashish121@outlook.com, prateekparhi936@gmail.com, mmanuj.aggarwal@gmail.com

Abstract— **Recommender system comprises of two prime methods which help in providing meaningful recommendations namely, Collaborative Filtering algorithm and Content-Based Filtering. In this paper, we have used a hybrid methodology which takes advantage of both Content and Collaborative filtering algorithm into account. The algorithm discussed in this article is different from the previous work in this field as it includes a novel method to find the similar content between two items. The paper incorporates an analysis that justifies this new methodology and how it can provide practical recommendations. The above approach is tested on existing user and objects data and produced improved results when compared with other two favourite methods, Pure Collaborative Filtering, and Singular Value Decomposition.**

Keywords—RecommenderSystems, Sparsity, Collaborative Filtering, Content Based Filtering, Singular Value Decomposition, Hybrid Approach

I. INTRODUCTION

Recommender systems (RS) are the software machinery that recommends or suggest appropriate items to users. Three important stages are involved in the tasks of RS, object data collection, similarity decision and prediction computation [1]. Also, Recommendation systems are based on three major approaches [3,4]. Content-based approaches [12] makes use of the content of the items and even the users. In our methodology, we have used the Genres and Tags. Thus using this method one can find the similarities between the content of one movie and the other which are liked by the users [4,5]. To predict the likes of a target user, Collaborative Filtering takes into consideration, the neighbors of that target user, and finds the similarity between the neighbors and target user such that the most similar users are selected and their ratings and likes are recommended to the target users [3,5,8]. Thus, user preferences are dependent on the others users present in the active user's neighborhood. Also, the domain dependence nature of CF can make it vulnerable to sparsity and cold start. Further, this class of recommender can be classified into memory based ,model based [15] and the hybrid of the two [3,5]. Since Collaborative filtering is largely dependent on the ratings of the users, therefore, if the number of users in the domain is low as compared to items then it can lead to cold start [3,4,11,16].

Hybrid approaches are the amalgam of content based and collaborative filtering [4,6,7,10,12]. In the later sections, we will discuss that how a hybrid CF algorithm is superior to CF and CB.

II. RELATED WORK

Melville et al. [12] proposed a Content Boosted Collaborative Filtering Algorithm that combines both content and collaborative approaches to providing recommendations. The sparsity of the user-item rating matrix is 97.4%. Pseudo rating matrix is calculated using the content-based filtering method that learns the user profiles. The simple, naive Bayesian text classifier used in Content Boosted Collaborative Filtering compares and classifies the content of different movies and the pseudo-ratings matrix generated from the above with the help of collaborative filtering makes predictions. For similarity between two movies, the authors used Pearson correlation [2]. In our approach, we have modified the content based algorithm, and instead of using Naive Bayes for text matching, we have used a simple comparator which compares and matches the tags and genres of two movies which are tested on the movielens dataset and compared with the SVD and Pure CF. Also, we have made sure that the initial sparsity in out user-item rating matrix is high as compared to the previous models.

III. DATASET DESCRIPTION

To test our Modified Hybrid content recommendation approach, we have used Movielens Dataset [16]. The dataset comprises individual ratings provided by users for a particular movie. This dataset consists of total 100004 ratings where, rating varies from 0 to 5 for 9125 movies, given by 671 users. The total number of genres is 20. The row indicates users and the columns indicate the movies which lead to a formation of 671 × 9125 user-item rating matrix. The Movielens dataset consists of the following slots userId, movieId, title, ratings, tags, and genres. The Dataset is later filtered, and the user-rating matrix with sparsity 98.36% is tested with our algorithm. The number of ratings initially provided in the dataset was 100004. Out of these ratings, 2000 ratings were separated which were later used to test the accuracy of our algorithm. After separating these ratings, we were left with our

training dataset constituting 98004 ratings whose sparsity was 98.399%. Now for the further set of readings we randomly removed some percentage of rating from training dataset in the direction of increased sparsity and tested with our predicted ratings. In this way, we took a total of six readings with sparsity ranging from 98.399 to 99.8%.

IV. PROPOSED METHODOLOGY

Our proposed algorithm takes into considerations the tags and genres specified in the dataset, and for the content-based prediction, we have applied a set matching comparator. This comparator returns the number of common objects between two movies. The term object here refers to tags and genres. For each particular movie, the tags and genres are merged into a single set. This gives us a bulky content for each movie, and more the content better is the predictions. After getting the set of common objects, the weight of each set for a movie is calculated. Once the weights are assigned to each of the set, they are then used to provide the ratings of the unrated movies using the rated movies which were previously compared. In our methodology first, the tags for each movie assigned by different users are used and converted them into a single list. The genres for each movie are appended to the same list of tags. This final list is referred as the objects for a particular movie. The object set for each active movie is compared with the object set of every other movie in the dataset and the number of matching objects are assigned to a set. The length of this set is used to predict the ratings as shown below:

$$R = M * (H_r/M')$$

where, R is the rating for an active movie, M indicates the number of common objects, M' is the maximum number of matching objects between any two movies in the dataset and highest_rating (H_r) is the maximum rating that we can assign to a movie, which in our case is 5. If the rating is greater than 2.5(a threshold value which is equal to the average of the lowest and the highest rating possible), then we can assign that movie and its computed ratings to the similar movie-set of an active movie. Next, we have constructed the user rating matrix using the dataset. This matrix is at 98.36% sparsity. Using the similar movie's list formed we have reduced the sparsity of the user rating matrix. For every non-zero entry in user rating matrix, we find the movies similar to it, using the list formed in the above steps. Once the sparsity from the user rating matrix is reduced we have applied CF by using Pearson Correlation and hence generated final predictions for the users.

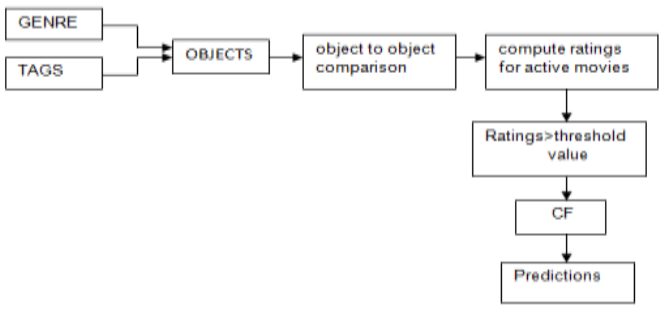

Fig. 1. Flowchart of Hybrid Collaborative Filtering algorithm for improved movie recommendations

V. EVALUATION

While evaluating our proposed methodology we have taken two traditional recommender approaches which are Pure CF and SVD. The above two methods are tested on the same Movielens dataset and at different sparsity levels. Also, the reason behind choosing these two methods for evaluation and testing is that the previous works in this field [12] have compared their approach of content boosted CF with the same two methods. Since our algorithm is designed in order to compare the improvements with the work done by Melville et al. [12] hence, we took the same evaluation methods for testing purpose which the former had in its work. Also, by virtue of this evaluation the difference in the results and the efficacy of our proposed approach can be clearly seen.

Our model is evaluated using the Mean absolute error (MAE) metric. Mean Absolute error is a robust evaluation model. It is a more natural measure of average error. Also, the Dimensioned Evaluations and inter-comparison models should use MAE as an evaluation metrics [14]. It is the deviation of the predicted values from the actual values. To calculate the MAE we have taken into account the predicted ratings and the actual ratings. The MAE values are calculated at a different level of the sparsity of the user-item matrix and are calculated separately for all three algorithms. Also, the dataset used here for testing and evaluation is better than the ones used in similar kind of approaches as mentioned in Section III. Although the total number of users is less but, the number of movies in our case suppresses this fact and provides an added advantage since the number of movies are much more, such that it allows the algorithm to run on a more sparsed user-rating matrix hence the results provided in the next sections are justified.

VI. RESULTS

We compared our results with two popular approaches, Pure CF, and SVD. The results of which are shown in the table given below. Also, the graph in the figure given below shows how hybrid movie recommendation performs slightly better than Pure CF and outperforms SVD. On comparing with Pure CF, we found that for high sparsity levels our approach works better than Pure CF and the MAE values were better than the one's generated from the Pure CF. The reason is that Pure CF algorithm is dependent on the data available through the user-rating matrix. At high sparsity, the data available is less and hence Pure CF doesn't perform better. On the other hand, the primary area of interest of our algorithm was to reduce the sparsity level by applying item-item comparison. Thus the CF used after content based filtering in our case works better, in the case of higher sparsity than the pure CF. In Fig.2 it is clear that at sparsity levels around 98.5%, there is little difference between the results of Pure CF and our approach but on further increase in the sparsity around 99%, the difference in the results increases and our algorithm out- performs the Pure CF. Fig. 3 shows that when our algorithm is compared with SVD then, from sparsity levels starting at 98% to 100%, Hybrid CF algorithm for improved recommendations performs well and we found that our approach is way more efficient than SVD. It can be seen quite clearly seen from Table-I, at 98.399% of sparsity MAE is 2.5199 and as the sparsity is

increased the MAE values are also increasing. The differences in the MAE values for both the algorithms are huge. SVD fails to perform efficiently because data sparsity is less. Table-II shows that how the proposed methodology successfully reduces the sparsity of a given user-item rating matrix.

TABLE I. MAE VALUES OF HYBRID CF, PURE CF AND SVD

Sparsity	Hybrid CF	Pure CF	SVD
98.399%	0.77323	0.76928	2.51995
98.5%	0.77436	0.77326	2.592
98.72%	0.77826	0.77483	2.76146
99%	0.78139	0.77708	2.97789
99.5%	0.81075	0.82098	3.29984
99.8%	0.90207	0.92239	3.57369

TABLE II. IMPROVMENT IN SPARSITY AFTER APPLYING HYBRID CF

Initial sparsity	Improved sparsity
98.399%	97.13%
98.5%	97.22%
98.72%	97.44%
99%	97.75%
99.5%	98.365%
99.8%	98.937%

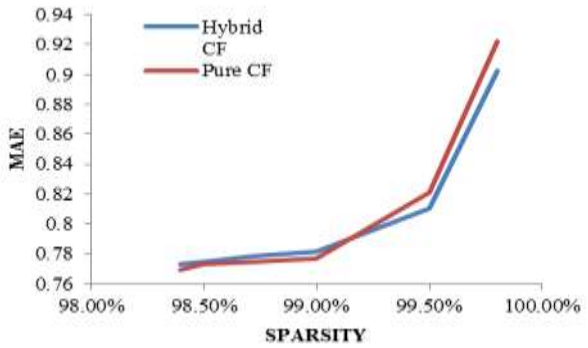

Fig. 2. Comparison of MAE values with increasing sparsity for Hybrid CF andPure CF

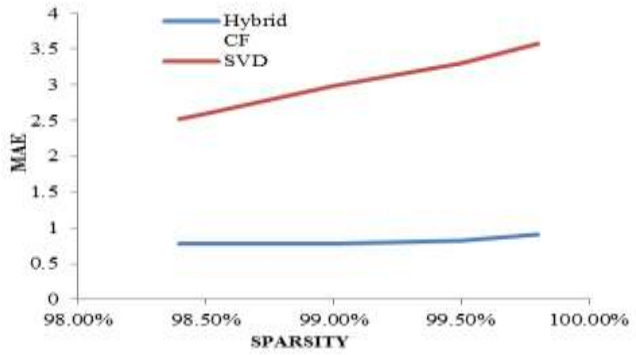

Fig. 3. Comparison of MAE values with increasing sparsityfor Hybrid CF and SVD.

VII. CONCLUSIONS

This paper is a novel alternative which depicts a simple approach to find the correlation between two features using set intersection in content-based filtering and finds the similarity between two items and predicts them for recommendations using CF. Previously related approaches have used text classifiers like Naive Bayes in the content based algorithm. The algorithm is further tested and compared with Pure CF and SVD. MAE values generated after evaluation provides successful comparisons. Although Hybrid content recommendation produced better MAE values and improved the sparsity of the dataset between 1%-2%, the results may vary when tested with a larger dataset.

REFERENCES

[1] Chen,Anne Yun-An, and Dennis McLeod."Collaborative filtering for information recommendation systems." Encyclopedia of E-Commerce, E-Government, and Mobile Commerce 1, 2005,pp. 118-123.

[2] Ricci, Francesco,LiorRokach,and BrachaShapira. "Introduction to recommender systems handbook".Springer US, 2011.

[3] Su,Xiaoyuan, and Taghi M. Khoshgoftaar. "A survey of collaborative filtering techniques." Advances in artificial intelligence,2009.

[4] Melville,PremandVikasSindhwani."Recommendersystems"

Encyclopedia of machine learning.Springer US, 2011, pp.829-838.

[5] Desrosiers,Christian, and George Karypis. "A comprehensive survey of neighborhood based recommendation methods." Recommender systems handbook.Springer US, 2011, pp.107-144.

[6] Bobadilla, Jesús, Fernando Ortega,Antonio Hernando, and Abraham Gutièrrez."Recommender systemssurvey." Knowledge-Based Systems 46,2013 : pp.109-13

[7] Pennock,David M.,Eric Horvitz, Steve Lawrence, and C.Lee Giles."Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach." In Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence,pp. 473-480. Morgan Kaufmann Publishers Inc.,2000.

[8] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J., 2001, April. "Item-based collaborative filtering recommendation algorithms." In Proceedings of the 10th international conference on World Wide Web (pp. 285-295). ACM.

[9] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42.8, 2009,pp. 30-37.

[10] Burke, Robin."Hybrid web recommender systems".The adaptive web.Springer Berlin Heidelberg, 2007, pp.377-408.

[11] Zhang, Z.K., Liu, C., Zhang, Y.C. and Zhou, T., 2010. "Solving the cold-start problem in recommender systems with social tags." EPL (Europhysics Letters), 92(2), p.28002.

[12] Melville,Prem,Raymond J. Mooney,and Ramadass Nagarajan. "Content-boosted collaborative filtering for improved recommendation"In Aaai/iaai, pp. 187-192. 2002.

[13] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J., 2002, December. Incremental singular value decomposition algorithms for highly scalable recommender systems. In Fifth International Conference on Computer and Information Science (pp. 27-28). Citeseer.

[14] Willmott, Cort J., and Kenji Matsuura. "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance." Climate research 30.1 (2005): 79-82.

[15] Hu,Yifan,Yehuda Koren, and Chris Volinsky. "Collaborative filtering for implicit feedback datasets."Eighth IEEE International Conference on Data Mining. IEEE,2008, pp. 263-272.

[16] MovieLens 100K Dataset(4/1998) permalink: https://grouplens.org/dataset/movielens/100K/