Combining User-Based and Item-Based Collaborative Filtering Using Machine Learning

Priyank Thakkar, Krunal Varma, Vijay Ukani, Sapan Mankad and Sudeep Tanwar

Abstract Collaborative filtering (CF) is typically used for recommending those items to a user which other like-minded users preferred in the past. User-based collaborative filtering (UbCF) and item-based collaborative filtering (IbCF) are two types of CF with a common objective of estimating target user's rating for the target item. This paper explores different ways of combining predictions from UbCF and IbCF with an aim of minimizing overall prediction error. In this paper, we propose an approach for combining predictions from UbCF and IbCF through multiple linear regression (MLR) and support vector regression (SVR). Results of the proposed approach are compared with the results of other fusion approaches. The comparison demonstrates the superiority of the proposed approach. All the tests are performed on a large publically available dataset.

Keywords User-based collaborative filtering · Item-based collaborative filtering Machine learning · Multiple linear regression · Support vector regression

P. Thakkar (✉) · K. Varma · V. Ukani · S. Mankad · S. Tanwar
Institute of Technology, Nirma University, Ahmedabad 382481, India
e-mail: priyank.thakkar@nirmauni.ac.in

K. Varma
e-mail: krunalvarma@nirmauni.ac.in

V. Ukani
e-mail: vijay.ukani@nirmauni.ac.in

S. Mankad
e-mail: sapanmankad@nirmauni.ac.in

S. Tanwar
e-mail: sudeep.tanwar@nirmauni.ac.in

S. C. Satapathy and A. Joshi (eds.), *Information and Communication Technology for Intelligent Systems*, Smart Innovation, Systems and Technologies 107,
https://doi.org/10.1007/978-981-13-1747-7_17

1 Introduction

Recommender systems have become more and more important due to increasing use of web for business and e-commerce transactions [2]. Movie recommender systems [7, 8], Book recommender systems [9], Tag recommender systems [13], Facebook friend recommender systems are some examples of recommender systems. CF models are among the basic models of recommender systems which can exploit user–item interaction data such as ratings. UbCF and IbCF are two flavours of collaborative filtering which are widely used in both industry and academia to address the information overload problem.

The first step in UbCF is to find the set of users that are most similar to the target user. Target user's rating for the target item is then predicted using the ratings given to the target item by these nearest/most similar neighbours/users. On the other hand, nearest neighbours in IbCF are the items which are most alike the target item. Ratings that are assigned by the target user to these most similar items are then used to compute his/her rating for the target item. The number of nearest neighbours is a design parameter and should be tuned properly.

This paper focuses on combining predictions of UbCF and IbCF to arrive at the final prediction. The novel contribution of this paper is the use of MLR and SVR to combine the predictions from UbCF and IbCF. Results of our proposed approach are compared with other fusion approaches in addition to the results of UbCF and IbCF when implemented individually.

The organization of the rest of the paper is as follows. Review of existing the literature is presented in Sect. 2 while Sect. 3 discusses fundamentals of collaborative filtering. Proposed fusion approach and results of experiments are discussed in Sects. 4 and 5 respectively. Paper ends with concluding remarks in Sect. 6.

2 Related Work

There have been some attempts to combine predictions from different recommender systems. In [5], content-based and collaborative filtering were merged with the help of a hybrid approach. An approach utilizing singular value decomposition and hybridization of content-based and IbCF for recommending programs on TV was proposed in [3].

One of the first attempts to combine UbCF and IbCF approaches is described in [12]. This approach was based on similarity fusion and the fusion framework was probabilistic in nature. The approach described in this paper is inspired from the work carried out in [11] and [6].

Thakkar et al. [11] also attempted to fuse predictions from UbCF and IbCF. However, their approach was simple and relied on weighted averaging of predictions to come up with final predictions. They figured out weights for averaging through fivefold crossvalidation of the training dataset.

Authors in [6] have combined the predictions from UbCF and IbCF using stacked regression. To the best of our understanding, they solved the regression problem of combining predictions from UbCF and IbCF as constrained quadratic optimization problem. We have attempted to solve the problem using simple approach of multiple linear regression (without any constraints) in addition to the approach involving support vector regression. The dataset used and experimental methodology adopted is also different.

3 Collaborative Filtering

UbCF and IbCF are discussed in this section. If we assume m users and n items, dimensionality of user–item rating matrix X is $m \times n$. Element $x_{i,j} = r$ indicates that ith user has assigned rating r to jth item, where $r \in R$. $x_{i,j} = \phi$ depicts that jth item has not been rated by ith user. Rows and columns in X correspond to users and items profile, respectively.

3.1 User-based Collaborative Filtering (UbCF)

As mentioned earlier, the first step in UbCF is to figure out target user's nearest neighbours. This can be achieved by finding similarity between the target user and all other users. The N most like-minded users can then be selected to form a set of N nearest neighbours. There are a few ways for finding similarity between users and Pearson correlation which is one such method is used in this paper. Pearson correlation between users u_1 and u_2 as discussed in [1, 11] is:

$$sim(u_1, u_2) = \frac{\sum_{i \in I_{u_1 u_2}} (x_{u_1,i} - \overline{x}_{u_1})(x_{u_2,i} - \overline{x}_{u_2})}{\sqrt{\sum_{i \in I_{u_1 u_2}} (x_{u_1,i} - \overline{x}_{u_1})^2 \sum_{i \in I_{u_1 u_2}} (x_{u_2,i} - \overline{x}_{u_2})^2}} \tag{1}$$

Here, $I_{u_1 u_2}$ is used to designate a set of items corated by u_1 and u_2. $\overline{x}_{u_1}$ indicates the average rating of user u_1.

There are several ways in which user i's rating for the item j can be worked out. In this paper, we have used Eq. 2 to accomplish this task [1, 11].

$$x_{i,j} = \overline{x}_i + \frac{\sum_{u' \in \hat{U}} sim(i, u') \times (x_{u',j} - \overline{x}_{u'})}{\sum_{u' \in \hat{U}} |sim(i, u')|} \tag{2}$$

Here, $\hat{U}$ expresses set of N nearest neighbours/users of user i who have rated item j.

3.2 Item-based Collaborative Filtering (IbCF)

In IbCF, items that are having similar profiles to the target item are considered as the nearest neighbours of the target item. As in UbCF, Pearson correlation as mentioned in Eq. 3 [10, 11] can be used to find similarity between items.

$$sim(i_1, i_2) = \frac{\sum_{u \in U} (x_{u,i_1} - \overline{x}_{i_1})(x_{u,i_2} - \overline{x}_{i_2})}{\sqrt{\sum_{u \in U} (x_{u,i_1} - \overline{x}_{i_1})^2 \sum_{u \in U} (x_{u,i_2} - \overline{x}_{i_2})^2}} \tag{3}$$

Here, U represents a set of users who have rated both i_1 and i_2. $\overline{x}_{i_1}$ depicts average rating of item i_1. There are several ways to calculate user i's rating for the item j. This paper employs Eq. 4 to accomplish the task [1, 11].

$$x_{i,j} = \overline{x}_j + \frac{\sum_{i' \in \hat{I}} sim(j, i') \times (x_{i,i'} - \overline{x}_{i'})}{\sum_{i' \in \hat{I}} |sim(j, i')|} \tag{4}$$

Here, $\hat{I}$ represents set of N items which are most similar to item j and have been rated by user i.

4 Proposed Approach

This paper proposes to fuse predictions from UbCF and IbCF through multiple linear regression and support vector regression models. The approach is inspired from the work done in [6, 11]. The idea is depicted in Fig. 1.

It is vital to note that a training set is needed to learn UbCF and IbCF models. The training set which is used for learning UbCF and IbCF is denoted as $Train_{CF}$

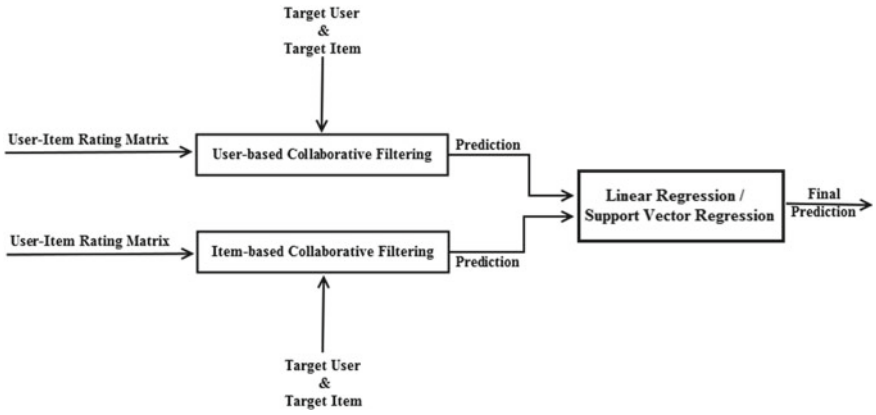

Fig. 1 Proposed approach

henceforth in this paper. It is easy to understand that $Train_{CF}$ for UbCF and IbCF consist of user and item profiles, respectively.

Mutliple linear regression and support vector regression models also need training set to get trained. We denote this training set as $Train_{ML}$ henceforth in this paper. It is evident that $Train_{ML}$ consists predictions from UbCF and IbCF as the training data. These predictions which constitute $Train_{ML}$ are made by UbCF and IbCF through fivefold cross validation of $Train_{CF}$.

Once the $Train_{ML}$ is formed, it is used to train multiple linear regression and support vector regression models. These trained models are then used to make the final prediction based on prediction from UbCF and IbCF.

5 Experimental Evaluation

This section begins with the discusion on dataset. Methodology used for various experiments and results are also discussed.

5.1 Dataset

All the experiments are carried out on Hetrec2011-movielens-2k dataset [4], (http://www.imdb.com/, http://www.rottentomatoes.com/) which was made public by a research group known as GroupLens (http://www.grouplens.com/). The dataset is summarized in Table 1. User–movie rating matrix was constructed by preprocessing this dataset.

5.2 Evaluation Measures

To evaluate performance of UbCF, IbCF and fusion approaches, mean absolute error (MAE), mean absolute percentage error (MAPE) and mean squared error (MSE) were used as discussed in [11].

Table 1 Dataset summary

Number of users	2113
Number of movies/items	10197
Number of ratings	855,598
Range of rating	0.5, 1.0, …, 5.0
Average number of ratings per user	405
Average number of ratings per movie/item	85

5.3 Experimental Methodology

For all the experiments, the users who had rated between 100 and 120 movies were selected as target users. There were 87 such users in the dataset. For each of the target users, randomly selected 25 movies–items were selected as target items. This gave us a test set of 87 users and 25 movies. Predictions were made for each of the user–movie pairs in the test dataset. In actual user–item rating matrix, 87 × 25 ratings of testset were masked to construct $Train_{CF}$. Experiments conducted included UbCF, IbCF and four fusion approaches. Fusion approach 1 used simple averaging while fusion approach 2 utilized weighted averaging as discussed in [11]. Fusion approaches 3 and 4 are the novel contributions of this paper, and they employ linear regression and support vector regression to combine predictions of UbCF and IbCF as discussed in Sect. 4.

5.4 Results and Discussion

Results of different techniques are depicted in Table 2. For each of the techniques, experiments were carried out for 12 different values of nearest neighbours (NN). It can be seen that only MAPE is reported in the results. It is worth mentioning that MAE and MSE were also measured but they have not been reported due to space limitations.

Minimum MAPE achieved by different techniques is summarized in Fig. 2. Minimum MAPE achieved by fusion through simple and weighted averaging approaches

Table 2 MAPE (reported values × 100%) in UbCF, IbCF and fusion approaches

Sr.	NN	UbCF	IbCF	Fusion using simple averaging [11]	Fusion using weighted averaging [11]	Fusion using multiple linear regression	Fusion using support vector regression
1	1	0.337	0.320	0.278	0.278	0.181	0.173
2	2	0.229	0.305	0.263	0.263	0.175	0.169
3	5	0.270	0.285	0.250	0.250	0.170	0.165
4	10	0.260	**0.277**	**0.246**	**0.246**	0.167	0.162
5	20	0.256	0.278	0.247	0.247	**0.166**	**0.161**
6	30	**0.255**	0.278	0.248	0.248	**0.166**	**0.161**
7	50	**0.255**	0.281	0.249	0.249	**0.166**	**0.161**
8	60	0.256	0.282	0.250	0.250	0.167	**0.161**
9	70	0.257	0.284	0.251	0.251	0.167	0.162
10	80	0.258	0.285	0.252	0.252	0.168	0.163
11	90	0.258	0.285	0.252	0.252	0.168	0.163
12	100	0.258	0.285	0.253	0.253	0.168	0.163

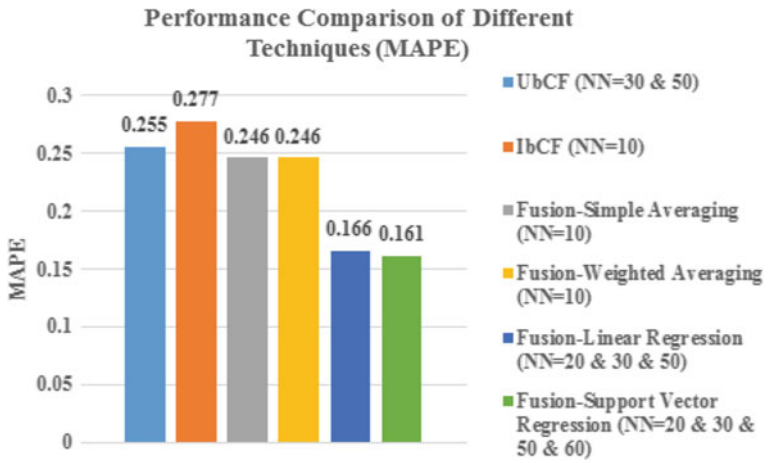

Fig. 2 Performance comparison of different techniques

is 0.246. This is definitely better when compared with UbCF and IbCF, with minimum MAPE of 0.255 and 0.277, respectively.

It can be seen that fusion through multiple linear regression and support vector regression has further improved the MAPE to 0.166 and 0.161, respectively, which is significantly better than all other approaches. This is because in these two approaches, optimal weights required for weighted averaging predictions from UbCF and IbCF are learnt through LR and SVR and the problem is handled as the learning problem.

6 Conclusion

The paper focused on fusing predictions from UbCF and IbCF with an intent of minimizing error in prediction. Experiments performed included UbCF, IbCF and four fusion approaches. First two fusion approaches relied on simple and weighted averaging of predictions from UbCF and IbCF to come up with the final predictions. The main contribution of the paper is an approach that combines predictions from UbCF and IbCF through multiple linear regression and support vector regression. Superiority of this approach is evident from the result. It can be seen that improvement in the performance is approximately 8% when compared to fusion through simple and weighted averaging. This raise is approximately 9% and 11% when compared to UbCF and IbCF, respectively. The boost in the performance is encouraging and presents a future direction where the robustness of the proposed approach can be validated by means of tests on other datasets.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
2. Aggarwal, C.C.: Recommender Systems. Springer, Berlin (2016)
3. Barragáns-Martínez, A.B., Costa-Montenegro, E., Burguillo, J.C., Rey-López, M., Mikic-Fonte, F.A., Peleteiro, A.: A hybrid content-based and item-based collaborative filtering approach to recommend tv programs enhanced with singular value decomposition. Inf. Sci. **180**(22), 4290–4311 (2010)
4. Cantador, I., Brusilovsky, P., Kuflik, T.: Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In: RecSys. pp. 387–388 (2011)
5. De Campos, L.M., Fernández-Luna, J.M., Huete, J.F., Rueda-Morales, M.A.: Combining content-based and collaborative recommendations: a hybrid approach based on bayesian networks. Int. J. Approx. Reason. **51**(7), 785–799 (2010)
6. Liu, Q., Xiong, Y., Huang, W.: Combining user-based and item-based models for collaborative filtering using stacked regression. Chin. J. Electron. **23**(4), 712–717 (2014)
7. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J.: Movielens unplugged: experiences with an occasionally connected recommender system. In: Proceedings of the 8th International Conference on Intelligent User Interfaces, pp. 263–266. ACM (2003)
8. Patel, R., Thakkar, P., Kotecha, K.: Enhancing movie recommender system. In: International Journal of Advanced Research in Engineering and Technology (IJARET), ISSN pp. 0976–6499 (2014)
9. Rich, E.: User modeling via stereotypes. Cogn. Sci. **3**(4), 329–354 (1979)
10. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM (2001)
11. Thakkar, P., Varma, K., Ukani, V.: Outcome fusion-based approaches for user-based and item-based collaborative filtering. In: International Conference on Information and Communication Technology for Intelligent Systems, pp. 127–135. Springer (2017)
12. Wang, J., De Vries, A.P., Reinders, M.J.: Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 501–508. ACM (2006)
13. Yagnik, S., Thakkar, P., Kotecha, K.: Recommending tags for new resources in social bookmarking system. Int. J. Data Min. Knowl. Manag. Process **4**(1), 19 (2014)