Improving Recommendations with Collaborative Factors

Penghua Yu[1], Lanfen Lin[1], Feng Wang[1,2], Jing Wang[1], and Meng Wang[1]

[1] Zhejiang University, Hangzhou, China, 310027
[2] Hangzhou Dianzi University, Hangzhou, China, 310018
{yph719,llf,wangfeng,cswangjing,wangmeng}@zju.edu.cn

Abstract. Collaborative filtering has become the most popular technique in the field of recommender system to deal with the information overload problem. Most collaborative filtering approaches either based on the intuitive nearest neighbor methods or the scalable latent factor models. In order to benefit from the advantages of these two paradigms, some hybrid strategies are proposed by taking weighted averages on near neighbors' ratings as effects, or factorizing neighborhood to model interactions and relationships directly. However, these methods usually assume that the latent factors of users/items are independent of each other. Yet in fact, there are relationships among latent factors would affect the performance of recommendations. Motivated by this, in this paper, we introduce the collaborative factors, which are smoothed by near neighbors' factors, to better capture the intrinsic features for users and items. We further propose a novel collaborative matrix factorization (CoMF) model in order to elaborately incorporate these collaborative factors into latent factor models. Finally, experimental results on two datasets show that our CoMF significantly outperforms some state-of-the-art methods in prediction accuracy.

Keywords: Recommender Systems, Collaborative Filtering, Collaborative Factors, Matrix Factorization.

1 Introduction

Recommender system is an indispensable tool used to produce item recommendations based on users' preferences. In many applications, degrees of users' preferences are presented by the explicit scores provided by users, or these implicit feedbacks inferred from users' behaviors [3]. We refer to all types of interactions as ratings. The ratings expressed by users on items are stored in a rating matrix, which is usually extremely sparse [1]. Existing recommendation algorithms are mainly based on two strategies: content based filtering versus collaborative filtering (CF). The former methods require gathering content information that might not be easy to collect. The latter only rely on users' historical records [10]. The CF methods are further adopted into two directions: the nearest neighbor based method and the latent factor model. The latter gains much more attention since the success of the Netflix competition, and many variants [2], [3], [8], [11] have been proposed to improve the prediction accuracy.

In order to take full advantages of these two strategies, the task on combining these CF approaches has been studied recently. Existing research can be classified into the

F. Li et al. (Eds.): WAIM 2014, LNCS 8485, pp. 30–33, 2014.

following categories: (1) incorporating the neighbors' preferences into latent factor approaches[2], [5]; (2) factorizing the neighborhood to directly model neighbors, e.g., a factorized neighborhood method [4] is presented; (3) making use of external data to model these relationships more realistically and precisely [6], [7]. However, both users and items are not independent of each other in the real world. The latent factors of similar entities are not directly utilized into latent factor models in previous studies. In this paper, we aim at improving the accuracy by injecting the dependent latent factors rather than the individual user and item factor.

The remainder of this paper is structured as follows. In Section 2, we introduce the collaborative factors and then elaborate our proposed model. The performances of our approach is reported in Section 3. Finally, we draw some conclusions and discuss the future work in Section 4.

2 Collaborative Matrix Factorization

Given an active user and a target item, we refer to factors of user' neighbors as the collaborative user factors, so as for collaborative item factors. The collaborative user factors and collaborative item factors are named together as collaborative factors:

$$p_{uc} = \alpha p_u + (1 - \alpha) \sum_{v \in N(u)} s_{uv} p_v \tag{1}$$

$$q_{ic} = \beta q_i + (1 - \beta) \sum_{j \in N(i)} s_{ij} q_j \tag{2}$$

where p_{uc} represents the factor of user u smoothed by the user' neighbors $N(u)$, q_{ic} is the factor for item i smoothed by its neighboring items $N(i)$. s_{uv} is similarity between user u and v, and s_{ij} is the similarity between item i and j. Then α and β are constants to determine weights between individual factor and collaborative factors.

Model Representation. We denote b_{ui}as the bias term effect, which makes up of global effect, user effect and item effect. Then we replace the factor of user and item in traditional latent factor model with collaborative factors of user and item, thus leading to a novel formulation represented as:

$$\hat{r}_{ui} = b_{ui} + \left(\alpha p_u + (1 - \alpha) \sum_{v \in N(u)} s_{uv} p_v \right)^T \left(\beta q_i + (1 - \beta) \sum_{j \in N(i)} s_{ij} q_j \right) \tag{3}$$

This model is referred to as the Collaborative Matrix Factorization (CoMF), and it is utilized to better infer the latent features of users and items.

Learning Algorithm. To find the optimization of the CoMF, we perform the minimization problem on the sum of squared error between the actual observed ratings and their predicted values. The objective function minimizing the regularized squared error on the set (denoted as S) of known ratings is then defined as:

$$L = \min_{p,q} \sum_{(u,i)\in S} (r_{ui} - \hat{r}_{ui})^2 + \lambda_1 \left(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2 \right) + \lambda_2 \left(\sum_u \|b_u\|^2 + \sum_i \|b_i\|^2 \right) \tag{4}$$

where $\hat{r}_{ui}$ obeys the rule formulated in Eq. (3). To learn optimization (b_*, p_* and q_*), we apply the stochastic gradient descent algorithm, which loops through all known ratings. The derivations are omitted as they are easy for readers to perform.

3 Experiments

We evaluate the methods on two data sets: the Movielens dataset[1] and the Epinions dataset [9]. In addition, we adopt the RMSE [1] on 5-fold cross validation to measure the result. We further compare our proposed CoMF with the following methods:

- **UserMean.** It utilizes the mean value of other users to predict the missing values.
- **ItemMean.** It applies the mean value of every item to predict the missing values.
- **Biased SVD.** The regularization term is set to 0.05, and the learning rate is 0.005.
- **SVD++.** It is proposed in [4] and regularization is 0.055, and learning rate 0.07.
- **RSTE.** It is proposed in [6] and α is set to 0.6 with the regularization term 0.04 and the learning rate is 0.01.

On the Movielens dataset, the regularization coefficients are set to 0.005and 0.001 with learning rate set to 0.005. In the Epinions dataset, all of them are all set to 0.01.

In the CoMF model, α and β play very important roles. To determine the sensitivity of α and β, we carried out an experiment where we varied the value of α and β from 0.0 to 1.0 in an increment of 0.2. We finally selected 0.6 for both α and β on both Movielens dataset and Epinions dataset as an optimum value.

Table 1. Performance comparisons of the CoMF with other approaches

Datasets	D	Metric	UserMean	ItemMean	BSVD	SVD++	RSTE	CoMF
Epinions	5	RMSE	1.1988	1.0942	1.0380	1.0408	1.0480	**1.0094**
		Improve	15.80%	7.75%	2.76%	3.02%	3.68%	
	10	RMSE	1.1988	1.0942	1.0378	1.0411	1.0406	**1.0021**
		Improve	16.41%	8.42%	3.44%	3.75%	3.70%	
Movielens	5	RMSE	1.036	0.983	0.9001	0.8932	0.8875	**0.8423**
		Improve	18.70%	14.31%	6.42%	5.70%	5.09%	
	10	RMSE	1.0360	0.9830	0.8980	0.8831	0.8702	**0.8291**
		Improve	19.97%	15.66%	7.67%	6.11%	4.72%	

Table 1 reports the results of our CoMF method compared with several state-of-the-art methods. "D" means dimensionality of factors. It can be observed that our CoMF method significantly outperforms the other methods both on the Movielens and the Epinions dataset.

[1] `http://www.grouplens.org/datasets/movielens/`

4 Conclusions

In this paper, we focus on improving recommendation accuracy of the latent factor models. Motivated by the fact that both users and items are not independent of each other in the real world, we introduce the collaborative factors to capture factors of the neighbors of users and items, respectively. We further investigate integrating these collaborative factors into the matrix factorization models to improve the prediction accuracy, and propose a novel collaborative matrix factorization (CoMF) model.

With the explosive increase of information in the Web, the modalities and types of information evolve at the same time. Such heterogonous information is useful for recommender systems. Therefore, it is important to consider the heterogeneous data to improve performance of recommendations. Meanwhile, scalable methods to address other recommendation tasks, such as the Top-N task, and other performance measures should be considered under this circumstance.

Acknowledgments. The work described in this paper is supported by grants from Programs under Grant No: 2012BAD35B01-3, 20110101110065, 2013BAF02B10. In addition, the authors thank the anonymous reviewers for their helpful comments.

References

1. Adomavicius, G., Tuzhilin, A.: Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE TKDE 17(6), 734–749 (2005)
2. Bell, R., Koren, Y.: Scalable Collaborative Filtering with Jointly Derived Neighborhood Interpolation Weights. In: IEEE ICDM, pp. 43–52 (2007)
3. Hu, Y., Koren, Y., Volinsky, C.: Collaborative Filtering for Implicit Feedback Datasets. In: IEEE ICDM, pp. 263–272 (2008)
4. Koren, Y.: Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model. In: ACM SIGKDD, pp. 426–434 (2008)
5. Koren, Y.: Factor in the Neighbors: Scalable and Accurate Collaborative Filtering. TKDD 4(1), 1 (2010)
6. Ma, H., King, I., Lyu, M.R.: Learning to Recommend with Social Trust Ensemble. In: ACM SIGIR, pp. 203–210 (2009)
7. Ma, H., King, I., Lyu, M.R.: Learning to Recommend with Explicit and Implicit Social Relations. ACM TIST 2(3), 29 (2011)
8. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In: Proceedings of KDD Cup and Workshop, pp. 5–8 (2007)
9. Paolo, M., Avesani, P.: Trust-aware Bootstrapping of Recommender Systems. In: Workshop on Recommender Systems, pp. 29–33 (2006)
10. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based Collaborative Filtering Recommendation Algorithms. In: ACM WWW, pp. 285–295 (2001)
11. Takács, G., Pilászy, I., Németh, B., Tikk, D.: A unified approach of factor models and neighbor based methods for large recommender systems. In: IEEE Applications of Digital Information and Web Technologies (ICADIWT), pp. 186–191 (2008)