Matrix Completion with Preference Ranking for Top-N Recommendation

Zengmao Wang[1,2] and **Yuhong Guo**[2] and **Bo Du**[1]

[1] School of Computer Science, Wuhan University, Wuhan 430072, China
[2] School of Computer Science, Carleton University, Ottawa, Canada
wangzengmao@whu.edu.cn, yuhong.guo@carleton.ca, gunspace@163.com

Abstract

Matrix completion has become a popular method for top-N recommendation due to the low rank nature of sparse rating matrices. However, many existing methods produce top-N recommendations by recovering a user-item matrix solely based on a low rank function or its relaxations, while ignoring other important intrinsic characteristics of the top-N recommendation tasks such as preference ranking over the items. In this paper, we propose a novel matrix completion method that integrates the low rank and preference ranking characteristics of recommendation matrix under a self-recovery model for top-N recommendation. The proposed method is formulated as a joint minimization problem and solved using an ADMM algorithm. We conduct experiments on E-commerce datasets. The experimental results show the proposed approach outperforms several state-of-the-art methods.

1 Introduction

Living in the information era, how to search valuable information from thousands and millions of online items to satisfy requests of various people is a challenge we face today. To address this challenge, recommender systems have drawn industry and academia's common attention in recent years [Covington *et al.*, 2016; Linden *et al.*, 2003; Xu *et al.*, 2017]; a top-N recommender system generates a short preference ranking list with length N for a user. In the past decades, recommender systems have been successfully applied in business applications, and many E-commerce companies have reported profit growth by integrating recommender systems into their applications [Covington *et al.*, 2016; Linden *et al.*, 2003].

Many works have developed top-N recommender systems to improve people's experience of online services, such as online shopping [Linden *et al.*, 2003; Mirbakhsh and Ling, 2015]. One popular technique for top-N recommendation is collaborative filtering (CF), which predicts people's preference by finding the relations in an observed user-item matrix. In general, CF methods can be divided into two categories, model-based methods and nearest-neighborhood-based methods. For model-based methods, matrix factorization is the

most popular technique, which projects users and items into a latent space and predicts the rating scores of a user-item matrix with the inner products of users' and items' latent factors. For example, [Cremonesi *et al.*, 2010] produces the user and item latent factors with singular value decomposition (SVD). [He *et al.*, 2017] performs matrix factorization with a deep learning technique by learning the nonlinearities of user and item latent factors with multi-layer perceptions. Different from model-based CF, which can be treated as user-item models, nearest-neighborhood-based methods mainly focus on the similarities between either items or users. The most representative approach is the item-based k-nearest-neighbor method, which is the first item-based approach for identifying a set of similar items [Deshpande and Karypis, 2004]. However, since the user-item matrix is very sparse, the recommendation quality can be very limited solely based on the similarity measurement. To improve the nearest-neighborhood-based collaborative filtering, a new type of item-item based model has been developed to learn a sparse coefficient matrix from data with linear item-item aggregations, which has demonstrated effective performance for top-N recommendation [Ning and Karypis, 2011; Kabbur *et al.*, 2013].

The above methods perform learning based on the pointwise scores of user-item pairs. However, preference ranking between each pair of items for a user is an intrinsic characteristic of top-N recommender systems for assisting users. Recently, many works have exploited preference ranking learning to improve the performance of top-N recommendation [Zhao and Guo, 2016; Rendle *et al.*, 2009; Man *et al.*, 2017; Park *et al.*, 2015]. These methods learn users' preference information between items from large number of ranking triplets, each containing a pair of items and a user. The preference ranking information can naturally benefit top-N recommendations due to the selective nature of the task. Nevertheless, such preference ranking methods mainly capture local item comparison information for each user, while ignoring global structural information of the recommendation matrix such as the inherent low-rank property. The low rank property is an intrinsic characteristic of recommendation matrices due to the limited available rating scores for each single user [Kang *et al.*, 2016; Cheng *et al.*, 2014].

In this paper, we propose a novel matrix completion ap-

proach that integrates both the low rank property and the preference ranking objective into a joint nonconvex learning problem for top-N recommendation. The objective function combines the low rank regularization and the preference ranking loss under a linear self-recovery model to enforce both the global low rank consistency and the local ranking order consistency over the user-item scores in the recovered recommendation matrix. To facilitate an easy optimization procedure over the joint nonconvex minimization, we use an alternating direction method of multipliers (ADMM) to reformulate the problem into convex sub-problems over each individual variable. To validate the effectiveness of our proposed approach, we test it on several real world datasets. The experimental results show that our approach achieves the best performance compared with several state-of-the-art top-N recommendation methods.

2 Related Works

As top-N recommender systems have significant commercial impact, various top-N recommender systems have been developed over the past decades. We briefly review the works that are most closely related to the proposed method. [Deshpande and Karypis, 2004] proposed the item-based top-N recommendation method based on items' similarities. The similarity between two items are measured with all the ratings on the two items, and then the top-N most similar items to the current items of the users are recommended. In [Cremonesi *et al.*, 2010], pureSVD was proposed by using the principle singular vectors to represent the users' and items' latent factors and reconstruct the rating matrix with the inner products of user-item latent factors. In [Ning and Karypis, 2011], the authors proposed a sparse linear method (SLIM) to recover the recommendation matrix by learning a sparse aggregation coefficient matrix that captures item-item correlations. SLIM achieved promising performance for top-N recommendation.

The observed records of a recommendation matrix typically demonstrate the preference of each user over the item set; in particular a user prefers the purchased items more than the unpurchased items. Hence the preference structure is one intrinsic characteristic of top-N recommendation tasks. To improve top-N recommendation performance, many works have exploited the preference structure in the top-N recommendation task [Rendle *et al.*, 2009; Man *et al.*, 2017; Yun *et al.*, 2014; Park *et al.*, 2015; Zhao and Guo, 2016]. The work in [Rendle *et al.*, 2009] proposed a ranking strategy that maximizes the posterior probability between a ranking pairs. [Man *et al.*, 2017] proposed a cross domain recommendation method and improved the method with the ranking strategy in [Rendle *et al.*, 2009]. [Yun *et al.*, 2014] proposed a ranking algorithm that is motivated by observing a close connection between evaluation metrics and loss functions. [Park *et al.*, 2015] proposed a large-scale collaborative ranking method from pairwise comparisons. [Zhao and Guo, 2016] integrated the collaborative ranking loss and the pointwise loss of SLIM to gain better recommendation performance.

Besides the preference structure, the low rank property of the recommendation matrix have yield another line of research study [Kang *et al.*, 2016; Cheng *et al.*, 2014;

Zhao *et al.*, 2017]. The work in [Cheng *et al.*, 2014] extended SLIM by using a nuclear norm to enforce low-rank regularization on the coefficient matrix. [Kang *et al.*, 2016] proposed to use a log-determinant function, $logdet$, to directly enforce the low-rank property of the recovered recommendation matrix, which produced impressive results on top-N recommendation tasks. [Zhao *et al.*, 2017] proposed a social local model by incorporating social connections into a local low rank framework.

Although many works have been done on top-N recommendation tasks, most of these existing methods have not fully considered the intrinsic characteristics of the task, e.g., exploiting the low rank structure and preference ranking structure simultaneously. In this paper, we complement the literature by developing a parametric top-N recommendation model that integrates low-rank and preference ranking together under a linear self recovery model.

3 Methodology

In this section, we present the proposed matrix completion approach. We consider the following settings. We assume a partially observed user-item matrix $M \in R^{n \times m}$ with implicit feedbacks, i.e., $\{0, 1\}$-values, over n users and m items. Each entry of the matrix, M_{ui}, represents whether user u has implicit feedback (e.g., a purchase record) on item i. If M_{ui} is equal to 1, it indicates the user u has a purchase record on item i; otherwise M_{ui} is 0. The main goal of the top-N recommendation task is to recommend the top-N item list that the user is most likely to purchase in the future by recovering a fully "observed" matrix X from the partially observed sparse matrix M.

3.1 Matrix Completion with Preference Ranking

Low-rank matrix completion methods and preference ranking methods perform matrix recovery for recommender systems by utilizing different characteristics of recommender matrix. We propose to combine them into a coherent joint model to integrate their complementary strengths.

Low-rank matrix completion methods have mainly pursued different approximations of the rank function. We adopt a nonconvex rank approximation from [Kang *et al.*, 2016], which uses the log-determinant approximation below

$$\min_{X} \quad logdet((X^T X)^{\frac{1}{2}} + I)$$
$$s.t. \quad X_{ui} = M_{ui}, \forall (u, i) \in \Omega; \ X \geq 0 \tag{1}$$

where Ω is a index set of the observed user-item entries and I is an identity matrix. This low-rank based learning method apparently did not consider any local user preference information. We assume a proper recovered matrix X will not only possess the global low-rank property, but also maintain a consistent local preference ranking structure in the observed matrix M. For example, if user u has purchased item i but not item j, it indicates that user u prefers item i more than item j. In fact, many methods in the literature have demonstrated that preference ranking over items is necessary and scientific for top-N recommendation. Hence, we propose to integrate the preference ranking advantage with the low-rank matrix

completion to further improve top-N recommendation performance. In particular, we formulate the minimization problem of matrix completion with preference ranking as follows:

$$\min_{X} \quad logdet((X^TX)^{\frac{1}{2}} + I) + \frac{\alpha}{2} \sum_{(u,i,j)\in\mathcal{T}} \mathcal{L}(X_{ui}, X_{uj}) \quad (2)$$

$$s.t. \quad X_{ui} = M_{ui}, \forall(u,i) \in \Omega; \; X \geq 0$$

where α is a trade-off parameter that balances the low-rank regularization and the preference ranking loss; $\mathcal{T}$ denotes a set of triplet instances $\{(u,i,j)\}$ which are derived from the observed user-item matrix, and $\mathcal{L}(\cdot)$ denotes a preference ranking loss for each triplet instance (u,i,j) from the set $\mathcal{T}$. A triplet instance (u,i,j) denotes user u prefers item i more than item j, namely $M_{ui} > M_{uj}$ in the observed matrix **M**. This also should be preserved as $X_{ui} > X_{uj}$ in the recovery matrix X.

Although the minimization problem in (2) meets our goal, the preference ranking loss in it mainly serves as a direct ranking regularization without being statistically and tractably learned from data, and the problem is hardly scalable. To solve this problem, we define the recovery matrix X using a linear self-recovery model $X = MW$ with parameter matrix W. Sparse linear self-recovery models have been very effectively exploited in a few literature works, including SLIM [Ning and Karypis, 2011]. Here we ignore the sparsity but keep constraints $W \geq 0$ and $diag(W) = 0$ on the parameter matrix to avoid trivial solutions. Moreover, we use a max-margin squared hinge loss function, $\mathcal{L}_r(x) = max(0, 1 - x)^2$, as the ranking loss, since hinge loss is particularly suitable for encoding pairwise ranking orders due to its comparative form. This yields the joint learning problem below:

$$\min_{X,W} \quad logdet((X^TX)^{\frac{1}{2}}+I) + \frac{\alpha}{2} \sum_{(u,i,j)\in\mathcal{T}} \mathcal{L}_r(M_{u,:}(W_{:,i}-W_{:,j}))$$

$$s.t. \; X_{ui} = M_{ui}, \forall(u,i) \in \Omega; \; X \geq 0; \quad (3)$$

$$X = MW; \; W \geq 0, diag(W) = 0$$

To increase the feasibility of the learning problem, instead of keeping the equality constraint $X = MW$, we encode it as a regularization term in the objective function instead and obtain the following learning problem:

$$\min_{X,W} \quad logdet((X^TX)^{\frac{1}{2}} + I) + \frac{\alpha}{2}||X - MW||_F^2$$

$$+ \frac{\beta}{2} \sum_{(u,i,j)\in\mathcal{T}} \mathcal{L}_r(M_{u,:}(W_{:,i} - W_{:,j})) \quad (4)$$

$$s.t. \; X_{ui} = M_{ui}, \forall(u,i) \in \Omega; \; X \geq 0;$$

$$W \geq 0, diag(W) = 0$$

where α and β are the trade-off parameters. Note in the optimization problem (4), the linear self-recovery parameter matrix W is learned mainly based on preference ranking loss, while the regularizer $||X - MW||_F^2$, as a relaxation of the equality constraint $X = MW$, builds connections between the low-rank recovery matrix X and the preference ranking recovery matrix MW and propagates the preference ranking information to produce the low-rank recovery matrix X.

Hence the trade-off parameter α actually controls how much the recovery MW from the preference ranking component affects the final recovery matrix X.

3.2 Optimization Algorithm

We adopt the alternating direction method of multipliers (ADMM) technique to solve the joint minimization problem (4). To obtain easy solutions for each subproblem of the ADMM algorithm, we first separate the low rank recovery and the preference ranking recovery by introducing an auxiliary variable Y and reformulate the problem as follows:

$$\min_{Y,X,W} logdet((X^TX)^{\frac{1}{2}} + I) + \frac{\alpha}{2}||Y - MW||_F^2$$

$$+ \frac{\beta}{2} \sum_{(u,i,j)\in\mathcal{T}} \mathcal{L}_r(M_{u,:}(W_{:,i} - W_{:,j})) \quad (5)$$

$$s.t. \; X_{ui} = M_{ui}, \forall(u,i) \in \Omega; \; Y \geq 0, X = Y;$$

$$W \geq 0, diag(W) = 0$$

Then we use the augmented Lagrangian dual formulation to encode the equality constraint $X = Y$, which produces

$$\min_{X,Y,W} logdet((X^TX)^{\frac{1}{2}} + I) + \frac{\rho}{2}||X - Y + \frac{Z}{\rho}||_F^2 \quad (6)$$

$$+ \frac{\alpha}{2}||Y - MW||_F^2 + \frac{\beta}{2} \sum_{\{u,i,j\}\in\mathcal{T}} \mathcal{L}_r(M_{u,:}(W_{:,i} - W_{:,j}))$$

$$s.t. \; X_{ui} = M_{ui}, \forall(u,i) \in \Omega; Y \geq 0, W \geq 0, diag(W) = 0$$

where ρ is a penalty parameter and Z is a dual variable matrix.

The ADMM algorithm will then solve (6) iteratively in an alternating manner by solving a sequence of simpler subproblems. In the $(t + 1)-$th iteration, we first update W given Y^t from the previous iteration by minimizing the following subproblem:

$$\min_{W} \frac{\alpha}{2}||Y^t - MW||_F^2 + \frac{\beta}{2} \sum_{\{u,i,j\}\in\mathcal{T}} \mathcal{L}_r(M_{u,:}(W_{:,i} - W_{:,j}))$$

$$s.t. \; W \geq 0, diag(W) = 0 \quad (7)$$

This subproblem can be solved using projected stochastic gradient descent, and the updating rules are as follows

$$W_{:,i}^{t+1} = W_{:,i}^t - \eta\{\nabla\mathcal{L}_r(W_{:,i}^t) + \frac{\alpha}{\beta|\mathcal{T}_i|}M^T(MW_{:,i}^t - Y_{:,i}^t)\}$$

$$W_{:,j}^{t+1} = W_{:,j}^t - \eta\{\nabla\mathcal{L}_r(W_{:,j}^t) + \frac{\alpha}{\beta|\mathcal{T}_j|}M^T(MW_{:,j}^t - Y_{:,j}^t)\}$$

$$(8)$$

where $|\mathcal{T}_i|$ and $|\mathcal{T}_j|$ represent the number of triplets which contain the item i and item j in $\mathcal{T}$ respectively; η is learning rate. $\nabla\mathcal{L}_r(W_{:,i}^t)$ and $\nabla\mathcal{L}_r(W_{:,j}^t)$ are gradients of $W_{:,i}^t$ and $W_{:,j}^t$ based on $\mathcal{L}_r(x) = max(0, 1 - x)^2$ respectively. If $M_{u,:}(W_{:,i}^t - W_{:,j}^t) < 1$,

$$\nabla\mathcal{L}_r(W_{:,i}^t) = 2M_{u,:}^T(M_{u,:}(W_{:,i}^t - W_{:,j}^t) - 1)$$

$$\nabla\mathcal{L}_r(W_{:,j}^t) = -2M_{u,:}^T(M_{u,:}(W_{:,i}^t - W_{:,j}^t) - 1) \quad (9)$$

If $M_{u,:}(W_{:,i}^t - W_{:,j}^t) \geq 1$, $\nabla\mathcal{L}_r(W_{:,i}^t) = 0$ and $\nabla\mathcal{L}_r(W_{:,j}^t) = 0$. Moreover, we project W^{t+1} onto the constraints by setting $W^{t+1} = max(0, W^{t+1})$ and $diag(W^{t+1}) = 0$.

Next we update Y with fixed X^t, Z^t and W^{t+1} by solving the following subproblem

$$\min_{Y \geq 0} \frac{\rho^t}{2} ||X^t - Y + \frac{Z^t}{\rho}||_F^2 + \frac{\alpha}{2} ||Y - MW^{t+1}||_F^2 \quad (10)$$

which has a closed form solution:

$$Y^{t+1} = \max\left(\left(\alpha MW^{t+1} + \rho^t \left(X^t - \frac{Z^t}{\rho^t} \right) \right)/(\alpha + \rho^t), 0 \right) \quad (11)$$

Then X can be updated with fixed Y^{t+1} by solving

$$\min_X \; logdet((X^TX)^{\frac{1}{2}} + I) + \frac{\rho^t}{2} ||X - Y^{t+1} + \frac{Z^t}{\rho^t}||_F^2$$
$$s.t. \; X_{ui} = M_{ui}, \; \forall (u, i) \in \Omega \quad (12)$$

This subproblem can be converted to scalar optimization problems following the following proposition.

Proposition 1 *Given a matrix $B \in R^{n \times m}$ with SVD of $B = Udiag(\sigma(B))V^T$, consider the minimization problem below*

$$\min_A \sum_{i=1}^{min(m,n)} g(\sigma_i(A)) + \frac{\delta}{2} ||A - B||_F^2 \quad (13)$$

where $\sigma_i(A)$ denotes the ith singular values of A and $g(\cdot)$ denotes a unitary invariant function with $g(A) = g(\sigma(A))$. This problem has a solution of $A^* = U(diag(\sigma^*(A)))V^T$, where $\sigma_i^*(A)$ can be obtained by solving the minimization problem below:

$$\min_{\sigma(A)} \sum_{i=1}^{min(m,n)} \left(g(\sigma_i(A)) + \frac{\delta}{2}(\sigma_i(A) - \sigma_i(B))^2 \right) \quad (14)$$

Proposition 1 can be easily proved following [Kang *et al.*, 2015; Lu *et al.*, 2015]. Based on this proposition, Solution for equation (12) can be obtained by solving a series of scalar problems for $\sigma^*(X)$ with:

$$\sigma_i^*(X) = \min_{\sigma_i(X) \geq 0} log(\sigma_i(X) + 1) + \frac{\rho}{2}(\sigma_i(X) - \sigma_i)^2 \quad (15)$$

where σ_i is the ith singular value of $(Y^{t+1} - \frac{Z^t}{\rho^t})$. By setting the derivative of the objective function of (15) wrt $\sigma_i(X)$ as zero, we can obtain its optimal solution $\sigma_i^*(X)$, which is the unique minimizer. Then X can be updated by

$$X^{t+1} = Udiag(\sigma^*(X))V^T, \; X_{ui}^{t+1} = M_{ui}, \forall (u, i) \in \Omega \quad (16)$$

where U and V are the singular vector decomposition $(Y^{t+1} - \frac{Z^t}{\rho^t}) = Udiag(\sigma)V^T$.

Finally, following the standard ADMM, we update the dual variables Z and the penalty parameter ρ with rules below

$$Z^{t+1} = Z^t + \rho^t(X^{t+1} - Y^{t+1}), \; \rho^{t+1} = \gamma \rho^t, \gamma > 1 \quad (17)$$

The overall procedure is summarized in Algorithm 1.

Algorithm 1 Matrix Completion with Preference Ranking

Input:
　　Initial parameters at $t = 0$: $X^t = Y^t = M$, the parameter matrix W^t with $W^t \geq 0$, $diag(W^t) = 0$, the penalty parameter ρ^t, the dual matrix $Z^t = 0$, and $\gamma > 1$.
1: **repeat**
2:　　Given Y^t, update the linear self-recover parameter W^t by (8);
3:　　Given X^t, Z^t, W^{t+1}, update the auxiliary matrix Y^t by (11);
4:　　Given Y^{t+1}, Z^t, ρ^t, update the completion matrix X^t by (16);
5:　　Given Y^{t+1}, X^{t+1}, update the Lagrangian dual matrix Z^t and ρ^t by (17);
6:　　Set $t = t + 1$
7: **until** $\frac{||X^{t+1} - X^t||_F^2}{||X^t||_F^2} \leq \epsilon$
Output:　　The completion matrix $X^* = X^{t+1}$.

---

4 Experiments

In this section, we report our experimental setting and results on six real world recommendation datasets.

4.1 Settings

To investigate the effectiveness of the proposed approach for top-N recommendation, we conducted experiments on six Amazon datasets: Automotive, Musical, Video, Office, Sports and Clothes. The datasets are downloaded from website[1]. We preprocessed the datasets by filtering out the users and items with less than 5 ratings. This leads to the following datasets: Automotive (2928 users, 1835 items, 20473 transactions), Musical (1429 users, 900 items, 10261 transactions), Office (4905 users, 2420 items, 53258 transactions), Video (5130 users, 1685 items, 37126 transactions), Sports (2000 users, 1961 items, 27209 transactions) and Clothes (2000 users, 1389 items, 12959 transactions). We transformed the matrices into implicit feedbacks by keeping value 1 if there is a transaction/activity between the user and item pair, and setting the value as 0 otherwise. The ranking pairs are randomly selected with one hundred times of user number.

We adopted the leave-one-out technique to evaluate the performances of different recommendation methods by following the setting of previous literature work [He *et al.*, 2017]. On each dataset, we selected the latest transaction item for each user as the test set and used the remaining items as the training set. For efficient test evaluation, we randomly sampled 100 items without transaction records as part of the test set and produced the top-N rank list from only the test set, which is a common strategy used in [He *et al.*, 2017; Koren, 2008; Elkahky *et al.*, 2015]. We adopted two widely used popular evaluation metrics, Hit Ratio (HR) and Normalized Discounted Gain (NDCG) [He *et al.*, 2017], to evaluate the top-N recommendation performance of each method.

We compared the proposed matrix completion with preference ranking method, *PrMC*, with several state-of-the-art

---

[1]http://jmcauley.ucsd.edu/data/amazon/

method	Automotive				HR@10	NDCG@10	Musical				HR@10	NDCG@10
	params						params					
BPR	200	-	-	-	0.4607	0.2992	500	-	-	-	0.4352	0.2764
SLIM	200	0.001	-	-	0.4095	0.2729	10	0.001	-	-	0.3765	0.2461
MC	1.5	0.03	-	-	0.4372	0.2816	1.5	0.03	-	-	0.4066	0.2612
NeuMF	64	4	8	4	0.4501	0.2891	64	4	8	4	0.4024	0.2563
$PrMC_{nopr}$	0.03	0.001	-	-	0.4348	0.2775	0.03	0.001	-	-	0.4038	0.2617
$PrMC_{norank}$	0.01	5000	-	-	0.4440	0.2786	0.01	5000	-	-	0.4059	0.2539
PrMC	0.03	1	5000	-	**0.4826**	**0.3037**	0.03	1	5000	-	**0.4423**	**0.2855**

method	Video				HR@10	NDCG@10	Office				HR@10	NDCG@10
	params						params					
BPR	100	-	-	-	0.5904	0.3949	200	-	-	-	0.3127	0.1718
SLIM	100	0.01	-	-	0.5723	0.4139	200	0.001	-	-	0.3172	0.1888
MC	1.5	0.01	-	-	0.6031	0.4383	1.5	0.02	-	-	0.3678	0.2169
NeuMF	64	4	8	4	0.5821	0.3876	64	4	8	4	0.3252	0.1764
$PrMC_{nopr}$	0.01	0.001	-	-	0.5926	0.4310	0.03	0.001	-	-	0.3672	0.2152
$PrMC_{norank}$	0.01	5000	-	-	0.6008	0.4174	3	5000	-	-	0.3370	0.1947
PrMC	0.01	0.01	5000	-	**0.6612**	**0.4552**	0.02	0.001	5000	-	**0.3702**	**0.2171**

method	Sports				HR@10	NDCG@10	Clothes				HR@10	NDCG@10
	params						params					
BPR	200	-	-	-	0.4400	0.2835	100	-	-	-	0.3060	0.1927
SLIM	100	0.001	-	-	0.4255	0.2734	100	0.001	-	-	0.3290	0.2118
MC	1.5	0.02	-	-	0.4580	0.2867	1.5	0.03	-	-	0.3510	0.2173
NeuMF	64	4	8	4	0.4515	0.2664	64	4	8	4	0.2995	0.1759
$PrMC_{nopr}$	0.02	0.001	-	-	0.4470	0.2801	0.03	0.001	-	-	0.3505	0.2163
$PrMC_{norank}$	0.001	5000	-	-	0.4450	0.2739	3	5000	-	-	0.3400	0.2071
PrMC	0.02	1	5000	-	**0.5065**	**0.3099**	0.03	1	5000	-	**0.3710**	**0.2262**

Table 1: Top-N recommendation results for comparison methods with N=10. The params columns contain the parameter settings for each approach. BPR: the latent factor dimension. SLIM: l_1 and l_2 norm regularization parameters. MC: the penalty μ and the update parameter γ, NeuMF: the dimension of latent factors, the number of negative pairs, the convergence epoches and the number of layers. PrMC: the penalty parameter ρ, the balance parameter α and β. $PrMC_{nopr}$: parameters γ, α. $PrMC_{norank}$: parameters α, β.

methods, *BPR, SLIM, MC* and NeuMF. *BPR* is a pairwise ranking method which is proposed in [Rendle *et al.*, 2009] based on a matrix factorization model. SLIM is a most popular and effective method on top-N recommendation [Ning and Karypis, 2011]. We used the librec code for these two methods[2]. MC [Kang *et al.*, 2016] and NeuMF [He *et al.*, 2017] are two state-of-the-art methods for top-N recommendation, which are based on low-rank matrix completion and deep learning technique respectively. For these two methods we used the released codes in the original papers. To further obtain ablation analysis for the proposed method, we also directly compared with two variants of the methods, $PrMC_{nopr}$ and $PrMC_{norank}$. $PrMC_{nopr}$ is a variant of the proposed method without the preference ranking component, which simply integrates the *logdet* function and the linear self construction. $PrMC_{norank}$ is another variant of the proposed method without the low rank regularization.

We performed the following parameter selection for the trade-off parameters and algorithm running parameters. The latent dimension of BPR is selected from $\{50, 100, 200, 500\}$. For SLIM, we chose the l_1 norm parameter from $\{1, 10, 100\}$, and the l_2 norm parameter from $\{0.001, 0.01, 0.1\}$. For MC, the Lagrangian penalty parameter is selected from $\{0.01,$

0.02, 0.03$\}$, and the γ for updating the Lagrangian penalty is selected from $\{1.5, 2, 2.5, 3, 3.5, 4\}$. For NeuMF, we used the most effective settings according to the original paper. For the proposed method, we chose the penalty parameter ρ from the same set as MC, and selected the weight parameter α from $\{0.001, 0.01, 0.1, 1, 3, 5\}$. We fixed γ as 1.5 and β as 5000. The best results are reported for each method.

4.2 Experimental Results

We reported the results in terms of HR@10 and NDCG@10 and the parameter setting for each method in Table 1. Meanwhile, we also reported results with different N values from $\{5, 10, 15, 20, 25\}$ in Figure 1.

Overall we can see that the proposed PrMC is the best-performing method, and it consistently outperforms the other comparison methods on all the six datasets across different N values. The other comparison methods though have strengths in different scenarios, none of them consistently performs better than the other. For example, though MC performs slightly worse than BPR on Automotive and Musical datasets, it outperforms BPR on the other four datasets. For the comparisons between SLIM and NeuMF, although NeuMF achieves better performance than SLIM in terms of HR, SLIM outperforms NeuMF in terms of NDCG in most cases
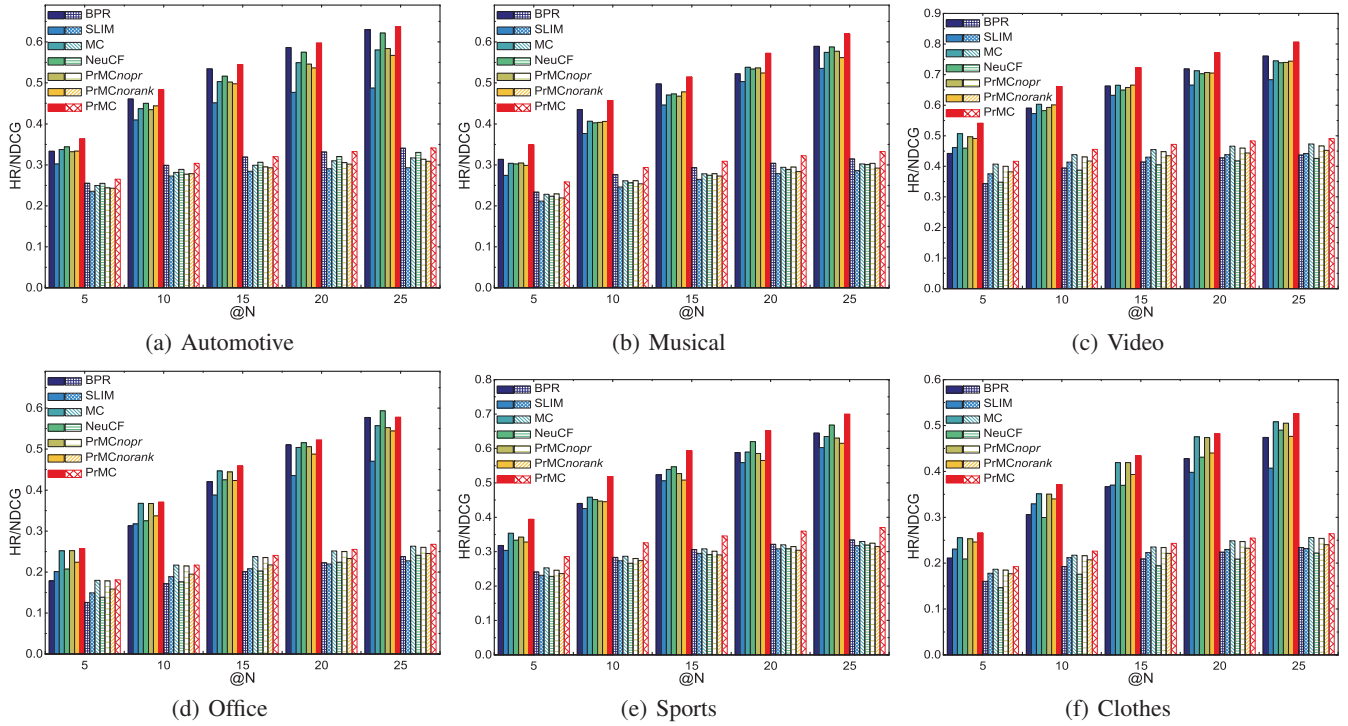
Figure 1: Top-N recommendation results in terms of HR and NDCG at different N levels on Amazon datasets: the solid bars represent HR@N, and the grid bars represent NDCG@N

(on four datasets). The two variants of the proposed method, PrMC$_{nopr}$ and PrMC$_{norank}$, both produce inferior performance comparing to PrMC. This suggests both the global low-rank regularization and local preference loss ranking on each user are effective for top-N recommendation and critical for the proposed model. The proposed model demonstrates its capacity for successfully integrating the two fundamental and complementary characteristics of recommender systems.

4.3 Parameters Analysis

For the proposed approach, two main parameters are tuned: the penalty parameter ρ and the trade-off parameter α that controls the affecting of preference ranking in the recovery matrix X. The sensitivity analysis are conducted on office and video datasets by using a range of different values, and the results in terms of HR@10 are reported in Figure 2.

From the results we can see that the performance of the approach is not very sensitive to the parameter value of ρ. Given fixed α value, the results have no obvious change with different ρ values when both α and ρ are not too small. This is reasonable since ρ is iteratively upscaled within the ADMM optimization procedure and the initial ρ value should not cause any dramatic difference. Comparing to ρ, the top-N recommendation performance of the proposed method is very sensitive to the α value. When α is large ($\alpha > 0.1$), the performance is very poor. This might be due to the fact that large α allows the preference ranking component to dominate the recovery matrix X and hence diminish the low-rank information. With a proper value $\alpha \in (0.001, 0.1)$, the low-rank and preference ranking can be better balanced and the performance can be more reasonable.
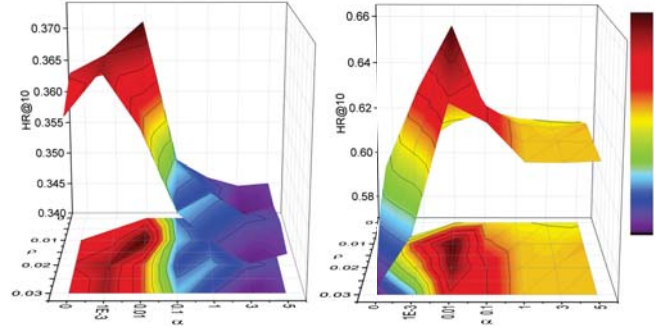

Figure 2: Parameter sensitivity analysis for α and ρ on Office (left) and Video (right) with metric HR@10. The colors represent the performance, with low values (dark color) to high values (red color).

5 Conclusion

In this paper, we proposed a novel top-N recommendation method based on matrix completion by integrating the low rank regularization and the preference ranking into a joint learning model. The proposed method maintains both the global low-rank structure and the local preference ranking structure of the recommendation matrix by combining their complementary strengths. We developed an ADMM algorithm to solve the proposed joint learning problem. We conducted the experiments on six E-commerce real world datasets. The results demonstrated that the proposed approach can outperform a number of state-of-the-art top-N recommendation methods. The ablation comparison results suggested both the low-rank and the preference ranking components are useful within our proposed model.

Acknowledgements

References

[Cheng *et al.*, 2014] Yao Cheng, Liang Yin, and Yong Yu. Lorslim: low rank sparse linear methods for top-n recommendations. In *Data Mining (ICDM), 2014 IEEE International Conference on*, pages 90–99. IEEE, 2014.

[Covington *et al.*, 2016] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 191–198. ACM, 2016.

[Cremonesi *et al.*, 2010] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 39–46. ACM, 2010.

[Deshpande and Karypis, 2004] Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.

[Elkahky *et al.*, 2015] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*, pages 278–288. International World Wide Web Conferences Steering Committee, 2015.

[He *et al.*, 2017] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*, pages 173–182. International World Wide Web Conferences Steering Committee, 2017.

[Kabbur *et al.*, 2013] Santosh Kabbur, Xia Ning, and George Karypis. Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 659–667. ACM, 2013.

[Kang *et al.*, 2015] Zhao Kang, Chong Peng, Jie Cheng, and Qiang Cheng. Logdet rank minimization with application to subspace clustering. *Computational intelligence and neuroscience*, 2015:68, 2015.

[Kang *et al.*, 2016] Zhao Kang, Chong Peng, and Qiang Cheng. Top-n recommender system via matrix completion. In *AAAI*, pages 179–185, 2016.

[Koren, 2008] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

[Linden *et al.*, 2003] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.

[Lu *et al.*, 2015] Canyi Lu, Changbo Zhu, Chunyan Xu, Shuicheng Yan, and Zhouchen Lin. Generalized singular value thresholding. In *AAAI*, pages 1805–1811, 2015.

[Man *et al.*, 2017] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. Cross-domain recommendation: An embedding and mapping approach. In *Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 2464–2470, 2017.

[Mirbakhsh and Ling, 2015] Nima Mirbakhsh and Charles X Ling. Improving top-n recommendation for cold-start users via cross-domain information. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(4):33, 2015.

[Ning and Karypis, 2011] Xia Ning and George Karypis. Slim: Sparse linear methods for top-n recommender systems. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 497–506. IEEE, 2011.

[Park *et al.*, 2015] Dohyung Park, Joe Neeman, Jin Zhang, Sujay Sanghavi, and Inderjit Dhillon. Preference completion: Large-scale collaborative ranking from pairwise comparisons. In *International Conference on Machine Learning*, pages 1907–1916, 2015.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, pages 452–461. AUAI Press, 2009.

[Xu *et al.*, 2017] Jingwei Xu, Yuan Yao, Hanghang Tong, Xianping Tao, and Jian Lu. Rapare: A generic strategy for cold-start rating prediction problem. *IEEE Transactions on Knowledge and Data Engineering*, 29(6):1296–1309, 2017.

[Yun *et al.*, 2014] Hyokun Yun, Parameswaran Raman, and S Vishwanathan. Ranking via robust binary classification. In *Advances in Neural Information Processing Systems*, pages 2582–2590, 2014.

[Zhao and Guo, 2016] Feipeng Zhao and Yuhong Guo. Improving top-n recommendation with heterogeneous loss. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 2378–2384. AAAI Press, 2016.

[Zhao *et al.*, 2017] Huan Zhao, Quanming Yao, James T Kwok, and Dik Lun Lee. Collaborative filtering with social local models. In *Proceedings of the 16th International Conference on Data Mining*, pages 645–654, 2017.