Sparse Linear Methods with Side Information for Top-N Recommendations

Xia Ning
Computer Science & Engineering
University of Minnesota, Twin Cities
4-192 EE/CS Building, 200 Union Street SE
Minneapolis, MN 55455, USA
xning@cs.umn.edu

George Karypis
Computer Science & Engineering
University of Minnesota, Twin Cities
4-192 EE/CS Building, 200 Union Street SE
Minneapolis, MN 55455, USA
karypis@cs.umn.edu

ABSTRACT

This paper focuses on developing effective algorithms that utilize side information for *top-N* recommender systems. A set of Sparse Linear Methods with Side information (SSLIM) is proposed, that utilize a regularized optimization process to learn a sparse item-to-item coefficient matrix based on historical user-item purchase profiles and side information associated with the items. This coefficient matrix is used within an item-based recommendation framework to generate a size-N ranked list of items for a user. Our experimental results demonstrate that SSLIM outperforms other methods in effectively utilizing side information and achieving performance improvement.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous; J.7 [**Computer Applications**]: Computers in other systems—*Consumer products*

Keywords

Recommender Systems, Sparse Linear Methods, Side Information

1. INTRODUCTION

Top-N recommender systems are used in E-commerce applications widely to recommend ranked lists of items so as to help the users in identifying the items that best fit their personal tastes. Over the years, many conventional collaborative-filtering-based *top-N* recommendation algorithms [2] have been developed that primarily focus on utilizing user-item purchase profiles to generate recommendations. Recently, with the increasing availability of additional information associated with the items, referred to as *side information*, there is a greater interest in taking advantage of such information in order to improve the qualities of *top-N* recommender systems. As a result, a number of approaches have been developed for incorporating side information, including hybrid methods [3], matrix/tensor factorization [5], and other regression methods [1].

In this paper, we propose a set of Side-information-utilized Sparse LInear Methods (SSLIM) for *top-N* recommendation. SSLIM learns a sparse coefficient matrix for the items, which is used to do *top-N* recommendation, by leveraging both user-item purchase profiles and side information on items within a regularized optimization process. These methods learn recommendation models that explicitly incorporate the relation between the side information of an item and the historical purchase information of that item so as to improve recommendation accuracy. Sparsity is introduced into the coefficient matrix, which allows SSLIM to generate recommendations efficiently, and thus makes it better suitable for real-time applications. Our experimental results show that SSLIM produces better recommendations than the state-of-the-art methods.

WWW 2012 Companion, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

2. METHODS

2.1 SLIM: Sparse Linear Methods

SSLIM is an extension to the Sparse LInear Method [4] (SLIM). In SLIM, the recommendation score on an un-purchased item t_j of a user u_i is calculated as a *sparse* aggregation of items that have been purchased by u_i, that is,

$$\tilde{m}_{ij} = \mathbf{m}_i^\mathsf{T}\mathbf{s}_j, \tag{1}$$

where $\mathbf{m}_i^\mathsf{T}$ is the purchase profile vector for u_i on all the n items with $m_{ik} = 1$ if u_i has purchased item t_k and 0 otherwise. In Equation 1, $m_{ij} = 0$ and $\mathbf{s}_j$ is a sparse size-n column vector of aggregation coefficients. *Top-N* recommendation for u_i is done by sorting u_i's non-purchased items based on their recommendation scores in decreasing order and recommending the *top-N* items.

SLIM views the purchase activity of user u_i on item t_j (i.e., m_{ij}) as the ground-truth item recommendation score, and learns the sparse $n \times n$ matrix $S = [\mathbf{s}_1, \cdots, \mathbf{s}_j, \cdots, \mathbf{s}_n]$ as the minimizer for the following regularized optimization problem:

$$\begin{aligned} \underset{S}{\text{minimize}} \quad & \frac{1}{2}\|M - MS\|_F^2 + \frac{\beta}{2}\|S\|_F^2 + \lambda\|S\|_1 \\ \text{subject to} \quad & S \geq 0 \\ & \text{diag}(S) = 0, \end{aligned}$$

where $M = [\mathbf{m}_1, \cdots, \mathbf{m}_i, \cdots, \mathbf{m}_l]^\mathsf{T}$ (i.e., the user-item purchase profile matrix), $\|\cdot\|_F$ is the matrix Frobenius norm, and $\|S\|_1 = \sum_{i=1}^n \sum_{j=1}^n |s_{ij}|$ is the entry-wise ℓ_1-norm of S. The regularization on ℓ_1-norm (i.e., $\lambda\|S\|_1$) will introduce sparsity in the solution. The $S \geq 0$ constraint is applied such that the learned S represents non-negative relations between items. The constraint $\text{diag}(S) = 0$ is also applied so as to avoid trivial solutions (i.e., the optimal S is an identical matrix) and to ensure that m_{ij} is not used to compute $\tilde{m}_{ij}$.

Table 1: Comparison of *Top-N* Recommendation Algorithms

method \ dataset	ML100K		NF		CrossRef		Lib		Pubmed		DrugSE	
	HR	ARHR	HR	ARHR	HR	ARHR	HR	ARHR	HR	ARHR	HR	ARHR
SLIM	0.343	0.147	0.045	0.018	0.388	0.203	0.407	0.266	0.247	0.141	0.280	0.157
SSLIM1	**0.347**	0.150	**0.050**	0.021	0.401	0.213	**0.444**	0.287	0.245	0.134	0.263	0.137
SSLIM2	0.344	0.146	0.047	0.017	**0.412**	0.212	0.438	0.279	**0.250**	0.141	**0.283**	0.158
itemSI	0.284	0.124	0.045	0.016	0.397	0.209	0.418	0.264	0.152	0.077	0.090	0.034
CWRMF	0.331	0.144	0.045	0.015	0.208	0.089	0.385	0.224	0.129	0.065	0.193	0.084

Bold numbers are the best performance in terms of HR for each dataset. N in this table is equal to 10.

2.2 SSLIM: SLIM with Side Information

We have developed two SSLIM approaches. Common to these approaches is that they utilize the side information during learning to bias the sparse coefficient matrix S.

The first approach imposes an additional requirement on S, that is, if the item purchase profile vector $\mathbf{m}_j$ for item t_j from all the l users is estimated by $\tilde{\mathbf{m}}_j = M\mathbf{s}_j$, then the side information feature vector $\mathbf{f}_j$ of t_j should also be well approximated by

$$\tilde{\mathbf{f}}_j = F\mathbf{s}_j. \quad (2)$$

where $F = [\mathbf{f}_1, \cdots, \mathbf{f}_j, \cdots, \mathbf{f}_n]$. In order to satisfy this requirement, the corresponding S matrix can be learned as the minimizer of the following optimization problem,

$$\begin{aligned} \underset{S}{\text{minimize}} \quad & \frac{1}{2}\|M - MS\|_F^2 + \frac{\alpha}{2}\|F - FS\|_F^2 + \frac{\beta}{2}\|S\|_F^2 + \lambda\|S\|_1 \\ \text{subject to} \quad & S \geq 0 \\ & \text{diag}(S) = 0, \end{aligned}$$

where the term $\frac{\alpha}{2}\|F - FS\|_F^2$ regularizes the coefficient matrix S to be learned such that it also fits a model on the side information (i.e., Equation 2). The parameter α controls the importance of the side information F during learning. Larger α indicates more importance/emphasis on side information F. This method is denoted as SSLIM1.

The second approach also tries to reproduce the feature vector for each t_j, but it utilizes a different linear combination of the feature vectors via another aggregation coefficient vector $\mathbf{q}_j$, that is

$$\tilde{\mathbf{f}}_j = F\mathbf{q}_j. \quad (3)$$

However, it is required that $\mathbf{s}_j$ and $\mathbf{q}_j$ should not be very different. Note that the requirement in Equation 2 is a special case of the one in Equation 3, when the penalty associated with the difference of $\mathbf{s}_j$ and $\mathbf{q}_j$ is very high. The matrix S and the matrix $Q = [\mathbf{q}_1, \cdots, \mathbf{q}_j, \cdots, \mathbf{q}_n]$ in Equation 3 can be learned as the minimizer of the following optimization problem,

$$\begin{aligned} \underset{S,Q}{\text{minimize}} \quad & \frac{1}{2}\|M - MS\|_F^2 + \frac{\alpha}{2}\|F - FQ\|_F^2 \\ & + \frac{\beta}{2}\|S - Q\|_F^2 + \lambda(\|S\|_1 + \|Q\|_1) \\ \text{subject to} \quad & S \geq 0, \ Q \geq 0 \\ & \text{diag}(S) = 0, \ \text{diag}(Q) = 0. \end{aligned}$$

where the parameter β controls how S and Q are different from each other. This method is denoted as SSLIM2.

3. RESULTS

We evaluated the performance of SSLIM1 and SSLIM2 and compared them with other popular *top-N* recommendation methods with side information incorporated (i.e., itemSI and CWRMF, modified from [2] and [5], respectively). The performance was evaluated on six real datasets (Table 1) following 5-time Leave-One-Out cross validation protocol, and measured by the Hit Rate (HR) and the Average Reciprocal Hit-Rank (ARHR) [2]. Comparison of *top-N* recommendation algorithms is presented in Table 1. Comparing SLIM based methods (i.e., SSLIM1 and SSLIM2 vs SLIM), for all the datasets, SSLIM1 and SSLIM2 always improve the *top-N* recommendation performance over SLIM in terms of both HR and ARHR. Comparing SSLIM1 and SSLIM2 with the other methods, either SSLIM1 or SSLIM2 achieves the best performance for all the datasets. In term of HR, the best of SSLIM1 and SSLIM2 is on average 8.5% better than the best of any other comparison methods. For different values of N for *top-N* recommendation, SSLIM consistently outperforms the other methods for the different values of N. We also conducted a density study by keeping the testing set and side information unchanged, but randomly select a certain percentage of non-zero values from each user so as to construct training sets of different information density. The results on such training sets show that in general, SSLIM1 and SSLIM2 perform better than the other methods as the density decreases.

Acknowledgement

This work was supported in part by NSF (IIS-0905220, OCI-1048018, IOS-0820730), NIH (RLM008713A), and the Digital Technology Center at the University of Minnesota. Access to research and computing facilities was provided by the Digital Technology Center and the Minnesota Supercomputing Institute.

4. REFERENCES

[1] D. Agarwal and B.-C. Chen. Regression-based latent factor models. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining*, pages 19–28, 2009.

[2] M. Deshpande and G. Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems*, 22:143–177, January 2004.

[3] Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, and L. Schmidt-Thieme. Learning attribute-to-feature mappings for cold-start recommendations. *IEEE International Conference on Data Mining*, pages 176–185, 2010.

[4] X. Ning and G. Karypis. Slim: Sparse linear methods for top-n recommender systems. In *Proceedings of 11th IEEE International Conference on Data Mining*, pages 497–506, 2011.

[5] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *Proceeding of the 14th ACM International Conference on Knowledge Discovery and Data Mining*, pages 650–658, 2008.