Recent Advances in Recommender Systems and Future Directions

Xia Ning[1] and George Karypis[2](✉)

[1] Department of Computer and Information Science, IUPUI, Indianapolis, IN 46202, USA
xning@cs.iupui.edu

[2] Department of Computer Science and Engineering, University of Minnesota, Twin Cities, MN 55455, USA
karypis@cs.umn.edu

Abstract. This article presents an overview of recent methodological advances in developing nearest-neighbor-based recommender systems that have substantially improved their performance. The key components in these methods are: (i) the use of statistical learning to estimate from the data the desired user-user and item-item similarity matrices, (ii) the use of lower-dimensional representations to handle issues associated with data sparsity, (iii) the combination of neighborhood and latent space models, and (iv) the direct incorporation of auxiliary information during model estimation. The article will also provide illustrative examples for these methods in the context of item-item nearest-neighbor methods for rating prediction and Top-N recommendation. In addition, the article will present an overview of exciting new application areas of recommender systems along with the challenges and opportunities associated with them.

1 Introduction

Recommender systems [1] are designed to identify the items that a user will like or find useful based on the user's prior preferences and activities. These systems have become ubiquitous and are an essential tool for information filtering and (e-)commerce [2]. Over the years, collaborative filtering (CF) [3], which derives these recommendations by leveraging past activities of groups of users, has emerged as the most prominent approach in recommender systems. Among the multitude of CF methods that have been developed, user- and item-based nearest-neighbor approaches are the simplest to understand and are easy to extend to capture different user behavioral models and types of available information. However, in their classical forms [3–8], the performance of these methods is worse than that of latent-space based approaches [9–14].

In this article, we present an overview of recent methodological advances in developing nearest-neighbor-based CF methods for recommender systems that have substantially improved their performance. In specific, we overview the methods that (i) use statistical learning to estimate from the data the desired

M. Kryszkiewicz et al. (Eds.): PReMI 2015, LNCS 9124, pp. 3–9, 2015.
DOI: 10.1007/978-3-319-19941-2_1

user-user and item-item similarity matrices, (ii) use lower-dimensional representations to handle issues associated with sparsity, (iii) combine neighborhood and latent space models, and (iv) directly incorporate auxiliary information during model estimation. We provide illustrative examples for these methods in the context of item-item nearest-neighbor methods for rating prediction and Top-N recommendation. We also briefly discuss the reasons as to why such methods achieve superior performance and derive insights from there for further development. In addition, we present an overview of exciting new application areas of recommender systems along with the challenges and opportunities associated with them.

2 Review of Previous Research

In the conventional nearest-neighbor-based CF methods [3–8,15–17], the user-item ratings stored in the system are directly used to predict ratings or preferences for a user on certain items. This has been done in two ways known as *user-based* recommendation and *item-based* recommendation. In user-based recommendation methods such as those used in GroupLens [6], Bellcore video [15] and Ringo [17], a set of *nearest user neighbors* for a target user is first identified as the users who have most similar preference patterns as the target user over a set of common items. Then the preferences from such neighboring users on a certain item are leveraged to produce a recommendation score of that item to the target user. In item-based approaches [3,5,7], on the other hand, a set of *nearest item neighbors* for a certain item is first identified as those that have been preferred in a most similar fashion as the item of interest by a set of common users. Then the recommendation score of the item to a user is generated by incorporating the user's preferences on the neighboring items.

The fact that the conventional nearest-neighbor-based CF methods work well in practice is largely due to that the available information of user-item preferences is typically very sparse, but such CF methods can capture and utilize the most important signals among the sparse data using simplistic and non-parametric approaches. Nearest-neighbor-based CF methods are intuitive, easy in computation, and very scalable to large e-commerce datasets and thus suitable for really applications. Although there have been numerous other recommendation methods developed over the years, particularly the latent-space-based methods [9–14], which involve more complicated modeling, demand much more computational resources and could achieve better recommendation performance, nearest-neighbor-based CF methods still remain as a strong baseline particularly when the trade-off between computational costs and performance is a major consideration.

3 CF from Data-Driven Nearest Item Neighbors: SLIM

Conventionally, the item-item similarities used in CF methods are calculated using a pre-defined similarity function, typically cosine similarity, correlation coefficient or their variations. A drawback of using pre-defined similarity functions is that they cannot adapt to different datasets and therefore may lead to poor neighborhood structures and thus sub-optimal recommendation results. A recent advance is to derive the similarity matrices from data rather than use any pre-defined similarity functions. A representative neighborhood-learning recommendation method is the Sparse LInear Methods (SLIM) [18]. In SLIM, the recommendation score $\tilde{r}_{ui}$ for a user u on an item i is predicted as a sparse aggregation of existing ratings in a user's profile, that is,

$$\tilde{r}_{ui} = \mathbf{r}_u \mathbf{w}_i^{\mathsf{T}},$$

where $\mathbf{r}_u$ is the user u's rating profile over items and $\mathbf{w}_i^{\mathsf{T}}$ is a sparse row vector of item similarities with respect to item i. The non-zero entries in $\mathbf{w}_i^{\mathsf{T}}$ correspond to the nearest items neighbors of item i. The item neighborhood matrix $W = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_n]$ is learned by minimizing the reconstruction error of the user-item data R using item-based CF with item neighbors represented in W. In specific, the optimization problem is formulated as follows,

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

where both the non-negativity constraint and the ℓ_1 regularization on W enforce a sparse and positive neighborhood for each item. Extensive experiments as in [18] demonstrate that SLIM outperforms the state-of-the-art latent-space-based methods in terms of recommendation performance. Meanwhile, SLIM is scalable to large datasets, which makes SLIM much more applicable in real applications. The success of SLIM validates CF as a fundamental framework for recommendation problems, and meanwhile demonstrates the advantage of data-driven item neighborhoods over the conventional hand-crafted similarity metrics in real problems.

4 CF from Factorized Item Similarities: FISM

A remaining issue for SLIM is that when the use-item data is very sparse, it is difficult to well estimate W. The data sparsity issue has substantially challenged almost all the CF based recommendation methods, while latent-space-based (LS) methods provide an appropriate remedy that consequently inspires the combination of CF and LS. The Factorized Item Similarity Method (FISM) [19] represents a recent effort along this line. In FISM, the recommendation score $\tilde{r}_{ui}$ for a user u on an item i is calculated from an aggregation of the items that have been rated by u and that are also similar to item i, where the item-item similarity between two items i and j is factorized and calculated as a dot product of two latent item factors $\mathbf{p}_j$ and $\mathbf{q}_i$. In specific, $\tilde{r}_{ui}$ is calculated as follows,

$$\tilde{r}_{ui} = b_u + b_i + (n_u^+)^{-\alpha} \sum_{j \in \mathcal{R}_u^+} \mathbf{p}_j \mathbf{q}_i^{\mathsf{T}}, \tag{1}$$

where $\mathcal{R}_u^+$ is the set of items that have been rated by user u, $n_u^+ = |\mathcal{R}_u^+|$, and b_u and b_i are user and item bias, respectively. The learning of $\mathbf{p}_j$ and $\mathbf{q}_i$ can be done by minimizing the reconstruction error or by minimizing the ranking divergence using Eq. 1 on the training data. The experiments in [19] show that when the user-item data is sparse, FISM outperforms SLIM in the recommendation performance. FISM provides a general framework that combines neighborhood-based CF and LS-based factorization of data-driven item-item similarities so as to effectively handle the data sparsity issues and achieve good recommendation performance.

5 CF from User-Specific Feature-Based Similarities: UFSM

In addition to leveraging advanced modeling and learning techniques as in FISM to accommodate for data sparsity, an alternative is to leverage additional information sources. The increasing amount of auxiliary information associated with the items in E-commerce applications has provided a very rich source of information that, once properly exploited and incorporated, can significantly improve the performance of the conventional CF methods. Thus, a recent trend is to incorporate auxiliary information to improve nearest-neighbor-based CF methods [20–22]. For example, in the User-specific Feature-based Similarity Models (UFSM) [20], the recommendation score $\tilde{r}_{ui}$ for a user u on an item i is calculated as the aggregation of multiple user-specific item-item similarities (i.e., l different similarity functions $\text{sim}(i, j)$), that is,

$$\tilde{r}_{ui} = \sum_{j \in \mathcal{R}_u^+} \sum_{d=1}^{l} m_{u,d} \; \text{sim}_d(i, j),$$

where $\text{sim}_d(i, j)$ is the d-th similarity between item i and item j, and it is estimated from the feature vectors $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ of items i and j, respectively, as follows,

$$\text{sim}_d(i, j) = \boldsymbol{w}_d(\boldsymbol{f}_i \odot \boldsymbol{f}_j)^{\mathsf{T}},$$

The Feature-based factorized Bilinear Similarity Model (FBSM) proposed in [21] extends UFSM by modeling the item-item similarity $\text{sim}(i, j)$ as a bilinear function of their features, that is,

$$\text{sim}(i, j) = \boldsymbol{f}_i^{\mathsf{T}} W \boldsymbol{f}_j$$

where W is the weight matrix which captures correlation among item features, and it is further factorized as follows so as to deal with data sparsity issues during learning,

$$W = D + V^{\mathsf{T}} V,$$

where D is a diagonal matrix and V is low-rank.

UFSM and FBSM calculate item-item similarities only from item features. This characteristics enables them to conduct cold-start recommendations for new items when there is no existing rating information for the new items. As demonstrated in [20] and [21], the performance of UFSM and FBSM for cold-start recommendations is superior to that of the state-of-the-art methods.

A different way to leverage auxiliary information is to use such information to bias the learning of an existing CF method. For example, SLIM is extended to incorporate item features in such a way [22]. In the collective SLIM method (cSLIM), it is imposed that the item-item similarities calculated from user-item information and the item-item similarities calculated from item features are identical, while in the relaxed collective SLIM (rcSLIM), the item-item similarities calculated from both aspects are close. In these methods, the item features are used to bias the learning of item neighbors so that the neighbor structures conform to and also encode the information from item features. It is demonstrated in [22] that when the user-item information is sparse, item features can play an important role for CF methods that use such information to achieve good recommendation performance.

6 Future Directions on Nearest-Neighbor-Based CF

There have existed other methods that have substantially improved conventional CF methods. Such methods include the ones that can capture high-order relations among item similarities [23], the methods that learn and utilize non-linear relations among items [24], etc. However, to make CF methods fully personalized, highly scalable and sufficiently robust against data sparsity and meanwhile produce high-quality recommendations, significant efforts from recommender system communities have been continuously dedicated. It has been recognized [25] that items may fall into clusters and thus item-item similarities may have local structures that may be sufficiently different from other local structures and from global structures, which leads to potential future research that discovers and incorporates local item neighbors into conventional CF methods. Fast and scalable learning algorithms are demanded for such methods once non-linear similarity structures are involved. On the other hand, dynamic components (e.g., user preferences change over time) have become ubiquitous among recommender systems, which may result in dynamically evolving user/item neighborhood structures. Such evolvement may exhibit interesting signals from which novel knowledge can be derived and used to predict future user preference/needs and make recommendations correspondingly (e.g., to recommend TV shows, to recommend courses). Another interesting research topic would be to develop scalable and efficient methods that can effectively incorporate heterogeneous auxiliary information from various static/dynamic sources in a systematical way into CF methods.

References

1. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.): Recommender Systems Handbook. Springer, New York (2011)

2. Schafer, J.B., Konstan, J., Riedl, J.: Recommender systems in e-commerce. In: Proceedings of the 1st ACM Conference on Electronic Commerce, EC 1999, pp. 158–166. ACM, New York (1999)
3. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. In: WWW 2001, Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM, New York, NY, USA (2001)
4. Delgado, J., Ishii, N.: Memory-based weighted majority prediction for recommender systems. In: Proceedings of the ACM SIGIR 1999 Workshop on Recommender Systems (1999)
5. Deshpande, M., Karypis, G.: Item-based top-N recommendation algorithms. ACM Trans. Inf. Syst. **22**, 143–177 (2004)
6. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to usenet news. Commun. ACM **40**, 77–87 (1997)
7. Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Comput. **7**, 76–80 (2003)
8. Nakamura, A., Abe, N.: Collaborative filtering using weighted majority prediction algorithms. In: ICML 1998, Proceedings of the 15th International Conference on Machine Learning, pp. 395–403. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
9. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 263–272. IEEE Computer Society, Washington, DC, USA (2008)
10. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, pp. 426–434. ACM, New York, NY, USA (2008)
11. Pan, R., Zhou, Y., Cao, B., Liu, N.N., Lukose, R., Scholz, M., Yang, Q.: One-class collaborative filtering. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pp. 502–511. IEEE Computer Society, Washington, DC, USA (2008)
12. Rennie, J.D.M., Srebro, N.: Fast maximum margin matrix factorization for collaborative prediction. In: Proceedings of the 22nd International Conference on Machine learning, ICML 2005, pp. 713–719. ACM, New York, NY, USA (2005)
13. Sindhwani, V., Bucak, S.S., Hu, J., Mojsilovic, A.: One-class matrix completion with low-density factorizations. In: Proceedings of the 2010 IEEE International Conference on Data Mining, ICDM 2010, pp. 1055–1060, IEEE Computer Society, Washington, DC, USA (2010)
14. Weimer, M., Karatzoglou, A., Smola, A.: Improving maximum margin matrix factorization. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part I. LNCS (LNAI), vol. 5211, p. 14. Springer, Heidelberg (2008)
15. Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: CHI 1995, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 194–201. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995)
16. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In: CSCW 1994, Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, pp. 175–186. ACM, New York, NY, USA (1994)

17. Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "word of mouth". In: CHI 1995, Proceedings of the SIGCHI Conference on Human factors in Computing Systems, pp. 210–217. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (1995)
18. Ning, X., Karypis, G.: Slim: sparse linear methods for top-n recommender systems. In: Proceedings of 11th IEEE International Conference on Data Mining. pp. 497–506 (2011)
19. Kabbur, S., Ning, X., Karypis, G.: 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, KDD 2013, pp. 659–667. ACM, New York, NY, USA (2013)
20. Elbadrawy, A., Karypis, G.: User-specific feature-based similarity models for top-n recommendation of new items. ACM Trans. Intell. Syst. Technol. 6(3) (2015)
21. Sharma, M., Zhou, J., Hu, J., Karypis, G.: Feature-based factorized bilinear similarity model for cold-start top-n item recommendation (2015)
22. Ning, X., Karypis, G.: Sparse linear models with side-information for top-n recommender systems. In: RecSys 2012 (2012)
23. Christakopoulou, E., Karypis, G.: HOSLIM: higher-order sparse linear method for top-N recommender systems. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) PAKDD 2014, Part II. LNCS, vol. 8444, pp. 38–49. Springer, Switzerland (2014)
24. Kabbur, S., Karypis, G.: Nlmf: Nonlinear matrix factorization methods for top-n recommender systems. In: 2014 IEEE International Conference on Data Mining Workshop (ICDMW), pp. 167–174 (2014)
25. Lee, J., Bengio, S., Kim, S., Lebanon, G., Singer, Y.: Local collaborative ranking. In: Proceedings of the 23rd International Conference on World Wide Web, WWW 2014, pp. 85–96. ACM, New York, NY, USA (2014)