Cross-Domain Recommendation for Cold-Start Users via Neighborhood Based Feature Mapping

Xinghua Wang[1], Zhaohui Peng[1(✉)], Senzhang Wang[2], Philip S. Yu[3,4], Wenjing Fu[1], and Xiaoguang Hong[1]

[1] School of Computer Science and Technology, Shandong University, Jinan, China
wang.xingh@foxmail.com, {pzh,hxg}@sdu.edu.cn, fuwenjing@mail.sdu.edu.cn
[2] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China
szwang@nuaa.edu.cn
[3] Department of Computer Science, University of Illinois at Chicago, Chicago, USA
psyu@uic.edu
[4] Institute for Data Science, Tsinghua University, Beijing, China

Abstract. Traditional Collaborative Filtering (CF) models mainly focus on predicting a user's preference to the items in a single domain such as the movie domain or the music domain. A major challenge for such models is the data sparsity problem, and especially, CF cannot make accurate predictions for the cold-start users who have no ratings at all. Although Cross-Domain Collaborative Filtering (CDCF) is proposed for effectively transferring users' rating preference across different domains, it is still difficult for existing CDCF models to tackle the cold-start users in the target domain due to the extreme data sparsity. In this paper, we propose a Cross-Domain Latent Feature Mapping (CDLFM) model for cold-start users in the target domain. Firstly, the user rating behavior is taken into consideration in the matrix factorization for alleviating the data sparsity. Secondly, neighborhood based latent feature mapping is proposed to transfer the latent features of a cold-start user from the auxiliary domain to the target domain. Extensive experiments on two real datasets extracted from Amazon transaction data demonstrate the superiority of our proposed model against other state-of-the-art methods.

Keywords: Cross-domain recommendation · Cold start
Feature mapping

1 Introduction

With the quick development of Internet and Web techniques, e-commerce has become increasingly popular. In order to help consumers find what they really desire from the massive amounts of products, recommender systems become indispensable in most e-commerce websites. Collaborative Filtering (CF) is a

J. Pei et al. (Eds.): DASFAA 2018, LNCS 10827, pp. 158–165, 2018.
https://doi.org/10.1007/978-3-319-91452-7_11

widely used technique in recommender systems due to the fact that it requires little domain-specific knowledge. Traditional CF models focus on single-domain user preference prediction and suffer from the data sparsity problem. Although Cross-Domain Collaborative Filtering (CDCF) is proposed to enrich the knowledge in the target domain by taking advantage of multi-domain ratings, and most CDCF models, e.g. TCF [1], CST [2], CMF [3] are designed to alleviate the single-domain data sparsity problem, while how to effectively recommend for the cold-start users is still not fully explored.

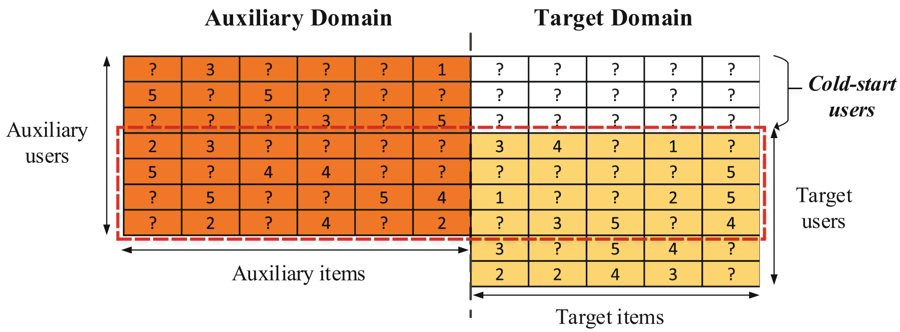


Fig. 1. Illustration of the cross-domain recommendation for cold-start users (Color figure online)

In real world, the cold-start users of an item domain may have ratings in another item domain. Thus, the problem setting about cross-domain recommendation for cold-start users studied in this paper is illustrated in Fig. 1. One can see from Fig. 1 that cold-start users only have ratings in the auxiliary domain, which is different from most previous works [1,2] that only assume the auxiliary domain data is relatively denser than the target domain data. Users who have ratings in both domains are called linked users whose rating data is marked with dashed red box in Fig. 1. It is challenging to make recommendations for the cold-start users in the target domain. First, rating matrices in different item domains are usually sparse, which makes it difficult to characterize users. Second, there is no rating data for cold-start users in the target domain, and user rating behavior and preference in different domains are varied.

To address the above challenges, we propose a Cross-Domain Latent Feature Mapping (CDLFM) model. Firstly, we handle the rating matrices in different domains separately with Matrix Factorization by incorporating User Similarities (MFUS) in order to gain domain-specific user latent features in sparse domains. Next, to transfer the knowledge of user characteristics across domains, we propose a neighborhood based gradient boosting trees method to learn the cross-domain user latent feature mapping function for each cold-start user. Due to space limit, the details of CDLFM are addressed in our technical report [4]. Our major contributions are summarized as follows:

- An improved rating matrix factorization model is proposed which is the first to consider users' similarity relationship reflected from their rating behaviors.
- A neighborhood based gradient boosting trees method is proposed for more accurately performing cross-domain latent feature mapping.
- We conduct extensive experiments on the Amazon rating data to evaluate the proposed model and make comparisons with other state-of-the-art models.

2 Related Work

The existing research related to our work mainly includes matrix factorization and cross-domain recommendation. Compared with existing rating matrix factorization models [5–7], besides taking the user rating behavior into consideration, our MFUS also capitalizes on the advantages of both the neighborhood and latent factor models by incorporating user similarities into the Matrix Factorization (MF) [5] process. TagiCoFi [7] also aims to improve the performance of MF via user similarities, but it relies on tagging information.

For cross-domain recommendation, transfer learning has been used extensively for alleviating the data sparsity problem. For the cold start problem, there have been tag-based and review-based cross-domain factorization models [8,9], and Hu [10] mentions the unacquainted world for users. EMCDR [11] and [12] try to use the Multi-Layer Perceptron (MLP) and a transformation matrix to map the user feature vector across domains, but they take all the linked users into consideration which may introduce noise. On social networks [13], Zhao [14] maps users' social networking features to another feature representation for product recommendation. In our work, no text information is available and we make cross-domain feature mapping in a more explicable way.

3 Matrix Factorization by Incorporating User Similarities

The user rating behaviors in different item domains can be quite different. Firstly, we handle the rating matrices of different domains separately. In order to better characterize users in sparse domains, we take users' rating behaviors into consideration and an improved rating matrix factorization model named MFUS (**M**atrix **F**actorization by incorporating **U**ser **S**imilarities) is proposed.

3.1 Rating Behavior Based User Similarity Measures

Similarity Based on Common Ratings. Given two users u and v, if they have commonly rated products C_{uv}, we can compute their similarity based on their rating similarity on C_{uv}. In our experiments, we first compute the average of squared rating difference over the users' ratings on C_{uv}, and then a monotonously-decreased exponential function is used to transform the difference into a similarity value.

Similarity Based on the Estimations of Having No Interest. A user's potential preference can be also reflected by the products that he/she does not give ratings to, and we can not arbitrarily conclude that a user does not like the unrated products. We use P_{ui} to represent the probability of user u having no interest on the product i. If u dose not rate i, P_{ui} can be estimated by $P_{ui} = [1 - f_1(n_u) \times f_2(n_i)] \times [1 - f_3(n_i/n) \times f_3(n_{Hi}/n_i)]$, where $f_1(n_u) = \sqrt{1 - \frac{n_u^2}{m^2}}$, $f_2(n_i) = \sqrt{1 - \frac{n_i^2}{n^2}}$, $f_3(x) = \frac{2}{1+e^{-\sigma x}} - 1$. n and m denote the numbers of users and products in a domain. n_u and n_i represent the total rating numbers of user u and product i. n_{Hi} is the number of high ratings on product i (the high rating is 4 or 5 in our experiments). When user u rates product i, P_{ui} can be estimated from the rating score R_{ui}. For example, if $R_{ui} = 1$, $P_{ui} = 1$; if $R_{ui} = 2$, $P_{ui} = 0.8$; if $R_{ui} = 3$, $P_{ui} = 0.5$ and so on.

Given two users and the products which have not been rated by both of them, we can compute a similarity value based on the computed probability values.

Similarity Based on Rating Biases. We observe that users' rating values are usually unevenly distributed. For example, high ratings, 4 and 5, usually account for a large proportion, while low ratings hold a small proportion. We call this as the rating biases of users. Here, we adopt the idea of TF-IDF to measure the users' rating biases, and the rating bias of user u to rating score $r \in \{1, 2, 3, 4, 5\}$ can be calculated via $rf(u, r) \times log_{base}(n/uf(r))$, where $uf(r)$ represents the number of users who have given the rating score r, $rf(u, r) = n_{ur} / \left(\sum_{z=1}^{5} n_{uz}\right)$, n_{ur} represents the frequency of rating score r used in u's rating history, and *base* is a predefined parameter (*base* $= 2$ in our experiments). Then we can compute the third similarity measure based on the users' rating biases.

Finally, for users u and v, the weighted average of the three similarity measures is used as the final similarity value between them.

3.2 Rating Matrix Factorization

We embed the user similarities in Sect. 3.1 into the matrix factorization model as a new regularization term and our goal is solving the following minimization problem:

$$\begin{aligned} \min_{\mathbf{U},\mathbf{V}} \frac{1}{2} \sum_{u=1}^{n} \sum_{i=1}^{m} Y_{ui} \left(R_{ui} - \mathbf{U}_{u*}\mathbf{V}_{i*}^T\right)^2 + \frac{\alpha}{2} tr\left(\mathbf{U}\mathbf{U}^T\right) \\ + \frac{\alpha}{2} tr\left(\mathbf{V}\mathbf{V}^T\right) + \frac{\beta}{2} \sum_{u=1}^{n} \sum_{v=u+1}^{n} S_{uv} \parallel \mathbf{U}_{u*} - \mathbf{U}_{v*} \parallel^2 \end{aligned} \tag{1}$$

where $\mathbf{U}_{u*}$ and $\mathbf{V}_{v*}$ represent the latent features of user u and product i, $(\cdot)^T$ and $tr(\cdot)$ denote the transposition and trace of a matrix, and Y_{ui} is 1 if user u rated product i and 0 otherwise. α and β are the regularization parameter to prevent over-fitting. Equally, we transform (1) to

$$\min_{\mathbf{U},\mathbf{V}} \frac{1}{2}\sum_{u=1}^{n}\sum_{i=1}^{m} Y_{ui}\left(R_{ui}-\mathbf{U}_{u*}\mathbf{V}_{i*}^{T}\right)^{2}+\frac{\alpha}{2}tr\left(\mathbf{V}\mathbf{V}^{T}\right)+\frac{1}{2}tr\left[\mathbf{U}^{T}\left(\alpha\mathbf{I}+\beta\mathbf{L}\right)\mathbf{U}\right] \quad (2)$$

where $\mathbf{L} = \mathbf{D} - \mathbf{S}$ with $\mathbf{D}$ being a diagonal matrix whose diagonal element is $D_{uu} = \sum_{v=1}^{n} S_{uv}$, $\mathbf{S}$ is the users' similarity matrix, and $\mathbf{I}$ is an identity matrix. We apply the alternating gradient descent to optimize one column of $\mathbf{U}$ or one row of $\mathbf{V}$ at a time. If we use F to represent the objective function in (2), the gradients can be computed as follows:

$$\frac{\partial \mathrm{F}}{\partial \mathbf{U}_{*k}} = \left(\alpha\mathbf{I}+\beta\mathbf{L}\right)\mathbf{U}_{*k}-\mathbf{x}, \quad \mathbf{x}\in R^{n\times 1} \text{ with } x_u = \textstyle\sum_{i=1}^{m} Y_{ui}\left(R_{ui}-\mathbf{U}_{u*}\mathbf{V}_{i*}^{T}\right)V_{ik}$$

$$\frac{\partial \mathrm{F}}{\partial \mathbf{V}_{i*}} = -\textstyle\sum_{u=1}^{n} Y_{ui}\left(R_{ui}-\mathbf{U}_{u*}\mathbf{V}_{i*}^{T}\right)\mathbf{U}_{u*}+\alpha\mathbf{V}_{i*}.$$

4 Neighborhood Based Latent Feature Mapping

The proposed MFUS can learn the domain-specific latent features of users in different domains. However, for the cold-start users U_T, we can only obtain their latent features in the auxiliary domain which cannot be used directly for making recommendation in the target domain due to the different semantic meanings of latent features in different domains. However, the same user's latent features in different domains can be highly correlated. Therefore, we try to use the linked users U_L as a bridge to learn the function $\mathcal{F}$ which can map the user's latent features from the auxiliary domain to the target domain. The input of the mapping function $\mathcal{F}$ is a user's latent features in the auxiliary domain and the output is the same user's latent features in the target domain.

We adopt the Gradient Boosting Trees (GBT) method [15] to learn the mapping function $\mathcal{F}$ since it is powerful to capture higher-order transformation relationship between the input and output. Assuming the dimension of the latent features in the target domain is K_t, we can use GBT K_t times and learn the mapping function $\mathcal{F} = \left\{f^{(k)}\left(\boldsymbol{x}\right)\right\}_{k=1}^{K_t}$, where the jth subfunction $f^{(j)}\left(\boldsymbol{x}\right)$ takes the user's latent features in the auxiliary domain as input and returns the jth mapped latent feature in the target domain. Moreover, for each cold-start user, we use the similar linked users to learn the mapping function. Thus in the last step of our model, for each user $u \in U_T$, we use $\mathbb{N}_u$ to denote the similar linked users to u with each $v \in \mathbb{N}_u$, $v \in U_L$ and $S_{uv}^{a} > sim$. Here, sim is a predefined similarity threshold value and S_{uv}^{a} is the user similarity in the auxiliary domain computed in MFUS. Latent feature pairs $\{\mathbf{U}_{v*}^{a}, \mathbf{U}_{v*}^{t}\}_{v\in\mathbb{N}_u}$, where $\mathbf{U}_{v*}^{a}$ and $\mathbf{U}_{v*}^{t}$ represent user v's latent features in the auxiliary domain and target domain, are used to learn the mapping function $\mathcal{F}_u = \left\{f_u^{(k)}\left(\boldsymbol{x}\right)\right\}_{k=1}^{K_t}$ via GBT. According to the latent features $\mathbf{U}_{u*}^{a}$ and the mapping function $\mathcal{F}_u$, we can compute the user mapped latent features $\boldsymbol{u}$ in the target domain with the element $u_k = f_u^{(k)}\left(\mathbf{U}_{u*}^{a}\right)$. Based on the latent feature matrix of products in the target domain, the rating predictions of the cold-start user u can be computed by $\hat{\mathbf{r}} = \mathbf{V}^{t}\boldsymbol{u}^{T}$.

5 Experiments

5.1 Experiment Setup

We extract two datasets from the Amazon rating data [16]. The first extracted dataset consists of the ratings about movies and books, and the second one consists of the ratings about movies and electronic products. We first filter out the linked users and items with very small number of ratings. Besides the linked users, some active users who have given a large number of ratings in a certain domain are also included. Finally, we have 16926 linked users in the first dataset and 12004 linked users in the second one. The statistics of the two datasets are given in Table 1. RMSE and MAE are used as the evaluation metrics, and we compare our model CDLFM with the following baselines AF [1], CDCF-U [10], CDCF-I [10], CMF [3], TMatrix [12] and EMCDR [11], where EMCDR is one state-of-the-art cross-domain recommendation method for cold-start users.

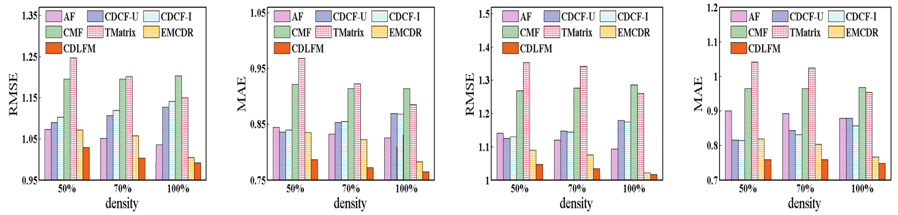
Table 1. Statistics of the two datasets used for evaluation

Dataset 1	Rating value			Density
Movie	$\{1, 2, 3, 4, 5\}$	#users	17926	0.00225
		#movies	4595	
		#ratings	185421	
Book	$\{1, 2, 3, 4, 5\}$	#users	17426	0.00149
		#books	8935	
		#ratings	231564	
Dataset 2				
Movie	$\{1, 2, 3, 4, 5\}$	#users	12203	0.00307
		#movies	3625	
		#ratings	135587	
Electronics	$\{1, 2, 3, 4, 5\}$	#users	12728	0.00212
		#electronics	4302	
		#ratings	115955	

5.2 Experimental Results

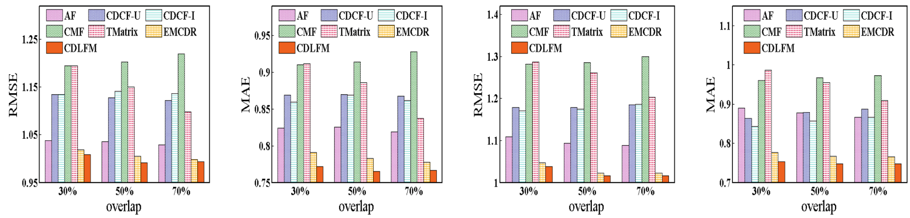
Experiments with different auxiliary and target domains are denoted as MB and ME for brevity. For example, MB denotes the experiments on Dataset 1 with **M**ovies as the auxiliary domain and **B**ooks as the target domain. The dimension of latent features is set to 15 and *sim* in CDLFM is set to 0.45.

Impact of Data Density. To evaluate the impact of data density, we randomly select 50% of the total linked users as the cold-start users and construct three



(a) MB (b) MB (c) ME (d) ME

Fig. 2. Performance of methods under different data density levels



(a) MB (b) MB (c) ME (d) ME

Fig. 3. Performance of methods under different overlap levels

different training sets denoted as density levels 50%, 70% and 100%. Taking the density level 70% for example, the training set consists of 70% of the total ratings in the auxiliary domain and 70% of the remaining ratings (after removing the cold-start users' ratings) in the target domain. Figure 2 report the results on different datasets and CDLFM performs best under all different data density levels. For EMCDR, MLP are learned based on all linked users which may introduce noise. Besides, from Fig. 2, one can see that the sparser the dataset is, the improvement of our model compared to EMCDR is more obvious.

In our model, MFUS can learn more accurate domain-specific latent features in sparse domains and neighborhood based GBT can learn more appropriate feature mapping function for each cold-start user. Therefore, we can predict cold-start users' latent features and preference accurately in the target domain.

Impact of the Size of Linked Users. we also experiment with three different user overlap levels, namely 30%, 50% and 70%. Taking overlap level 30% for example, we randomly select 70% of the total linked users as the cold-start users, and the remaining ratings in the dataset compose the training set. The results are reported in Fig. 3. One can see that our model achieves the best performance under all user overlap levels. Similarly, the less users overlap between two domains, the improvement of our model compared to EMCDR is more obvious.

6 Conclusions

In this paper, we present a novel model CDLFM for more effective cross-domain recommendation for cold-start users. Firstly, a new rating matrix factorization model is proposed to learn more accurate latent features of users in sparse domains. Then, a neighborhood based feature mapping method is used to learn more appropriate latent feature mapping function across domains. The experimental results demonstrate the superiority of our model.

Acknowledgements. This work is supported by NSF of China (No. 61602237, No. 61672313), 973 Program (No. 2015CB352501), NSF of Shandong, China (No. ZR2017MF065), NSF of Jiangsu, China (No. BK20171420). This work is also supported by US NSF through grants IIS-1526499, and CNS-1626432.

References

1. Pan, W., Yang, Q.: Transfer learning in heterogeneous collaborative filtering domains. Artif. Intell. **197**, 39–55 (2013)
2. Pan, W., Xiang, E.W., Liu, N.N., Yang, Q.: Transfer learning in collaborative filtering for sparsity reduction. In: AAAI, pp. 230–235 (2010)
3. Singh, A.P., Gordon, G.J.: Relational learning via collective matrix factorization. In: KDD, pp. 650–658 (2008)
4. Wang, X., Peng, Z., Wang, S., Yu, P.S., Fu, W., Hong, X.: Cross-domain recommendation for cold-start users via neighborhood based feature mapping. https://arxiv.org/
5. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: NIPS, pp. 1257–1264 (2007)
6. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD, pp. 426–434 (2008)
7. Zhen, Y., Li, W.J., Yeung, D.Y.: TagiCoFi: tag informed collaborative filtering. In: RecSys, pp. 69–76 (2009)
8. Fernández-Tobías, I., Cantador, I.: Exploiting social tags in matrix factorization models for cross-domain collaborative filtering. In: CBRecSys@RecSys, pp. 34–41 (2014)
9. Song, T., Peng, Z., Wang, S., Fu, W., Hong, X., Yu, P.S.: Review-based cross-domain recommendation through joint tensor factorization. In: Candan, S., Chen, L., Pedersen, T.B., Chang, L., Hua, W. (eds.) DASFAA 2017. LNCS, vol. 10177, pp. 525–540. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-55753-3_33
10. Hu, L., Cao, J., Xu, G., Cao, L., Gu, Z., Zhu, C.: Personalized recommendation via cross-domain triadic factorization. In: WWW, pp. 595–606 (2013)
11. Man, T., Shen, H., Jin, X., Cheng, X.: Cross-domain recommendation: an embedding and mapping approach. In: IJCAI, pp. 2464–2470 (2017)
12. Kazama, M., Varga, I.: Cross domain recommendation using vector space transfer learning. In: RecSys Posters (2016)
13. Wang, S., Hu, X., Yu, P.S., Li, Z.: MMRate: inferring multi-aspect diffusion networks with multi-pattern cascades. In: KDD, pp. 1246–1255 (2014)
14. Zhao, W.X., Li, S., He, Y., Chang, E.Y., Wen, J.R., Li, X.: Connecting social media to e-commerce: cold-start product recommendation using microblogging information. IEEE Trans. Knowl. Data Eng. **28**(5), 1147–1159 (2016)
15. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Statist. **29**, 1189–1232 (2000)
16. He, R., McAuley, J.: Ups and downs: modeling the visual evolution of fashion trends with one-class collaborative filtering. In: WWW, pp. 507–517 (2016)