

# 引入集群效应的跨领域推荐新方法\*

翟浩然, 张三国<sup>†</sup>

(中国科学院大学数学科学学院, 北京 100049)

(2023 年 1 月 5 日收稿; 2023 年 4 月 18 日收修改稿)

Zhai H R, Zhang S G. A new cross-domain recommendation method with cluster effect[J]. Journal of University of Chinese Academy of Sciences, 2025, 42(2): 153-158. DOI:10.7523/j.ucas.2023.040.

**摘要** 近年来,推荐系统在网络平台上得到了广泛的应用,它可以从巨量的数据中提取有用的信息,并根据用户的喜好向用户推荐合适的项目。基于此,提出一种利用相似用户对不同项目的评分数据作为源域对目标域的跨领域推荐表现方法,在研究的目标域中引入项目集群效应,提取与某个项目相关的具有相似特征的信息。该方法可有效解决数据稀疏性的问题,由于目标域的稀疏性,目标域的测试集中多数项目是拥有很少评分的,它们的信息难以从训练集中获得。所提模型的一个优点是,能够通过基于每个用户对项目的评分和缺失情况相关的变量的聚类,将来自缺失机制和特定项目集群特征的信息结合起来。MovieLens 数据分析表明,与现有推荐方法和跨领域推荐方法相比,所提出的引入集群效应跨领域推荐新方法在预测精度上有着有效的提升。

**关键词** 跨领域推荐;奇异值分解算法;集群效应;数据稀疏性

**中图分类号:** O212.1 **文献标志码:** A **DOI:** 10.7523/j.ucas.2023.040

## A new cross-domain recommendation method with cluster effect

ZHAI Haoran, ZHANG Sanguo

(School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China)

**Abstract** In recent years, the recommender system has been widely used in online platforms, which can extract useful information from giant volumes of data and recommend suitable items to the user according to user preferences. In this article, we put forward a crossdomain recommendation method based on the rating data of the different projects from similar users, introducing project cluster effect in the target domain of the study, the use of this specific group of singular value decomposition with the method of extracting information associated with a project with similar characteristics. This method could effectively solve the problem of data sparsity. Due to the sparsity of the target domain, most items in the test set of the target domain have few scores, and their information is challenging to obtain from the training set. A strictly related problem is the one of collaborative filtering in recommender systems, where an algorithm tries to extrapolate missing

\* 广西科技厅重点研发计划(2020AB10023)和国家自然科学基金(U19B2040)资助

<sup>†</sup> 通信作者, E-mail: sgzhang@ucas.ac.cn

information about the items from the rating activity of the users in order to provide a specific ad-hoc ranking for each user also on the items that have not been rated (on this see discuss how to aggregate the information from multilayer networks, while showing the importance of centrality measures for this issue). MovieLens data analysis indicated that, compared with the existing recommendation methods and cross-domain recommendation methods, the proposed new method of cross-domain recommendation with cluster effect has a significant improvement in the prediction accuracy.

**Keywords** cross-domain recommendation; singular value decomposition algorithm; cluster effect; data sparsity

随着信息量的大规模增长,如何高效地识别哪些信息与自己的兴趣相关变得越来越重要,这就产生了推荐系统,它可以通过帮助用户定位所需的物品有效地消除在线信息过载。电子商务平台、博客、电影网站、音乐软件、社交网站平台都不同程度上应用了推荐系统。此外,推荐系统还可应用于餐厅、酒店的选择以及基因识别治疗、金融服务等领域。

但是在提高推荐系统的推荐性能上面临许多挑战,随着数据量增长和信息量扩大,出现更多的用户和项目,用户对项目的评分矩阵变得越来越高维和稀疏,使得推荐系统的准确性下降。

目前比较成熟的推荐系统有 2 种,分别是协同过滤推荐和基于内容的推荐。协同过滤方法是基于用户的历史评分记录,假设具有相似历史评分的用户对于其他项目也会有相似的评分。主流的协同过滤推荐算法有以下几种:奇异值分解(singular value decomposition,SVD)算法<sup>[1]</sup>是对用户对项目的评分矩阵进行矩阵分解,获得用户和项目的嵌入向量来进行推荐评分的预测;最近邻方法(K-nearest neighbor,KNN)<sup>[2]</sup>是通过计算用户之间的相似度,根据相似用户对目标项目的评分来预测目标用户对目标项目的推荐评分;受限玻尔兹曼机(restricted Boltzmann machine,RBM)<sup>[3]</sup>通过二值化的分析方法判断用户对项目喜欢或是不喜欢,之后通过分析项目间隐藏因子决定用户对其余项目喜欢或是不喜欢。

基于内容的推荐算法不需要依据用户对项目的评分数据,认为用户对项目的评分主要取决于项目自身特征(类型、功能等),主要考察项目与用户偏好的匹配程度,根据历史信息提取用户偏好的属性特征,再计算用户偏好与项目特征的匹配度得到最终预测评分值。如 Blanco-Fernandez 等<sup>[4]</sup>通过比较项目内容和用户信息来推荐物品,这种推荐方法的优点是可将新项目推荐给用

户。但这种方法也面临着挑战,对每个用户建立档案需要进行预处理工作<sup>[5]</sup>,对于新用户的推荐也难以实现,面临冷启动问题<sup>[6]</sup>。

面对新用户或是新项目的冷启动问题仍是推荐算法面临的重大挑战,Park 等<sup>[7]</sup>在数据中加入特定特征的项目和用户来解决冷启动问题,Zhu 等<sup>[8]</sup>提出一种使用其他内容的辅助信息来提高推荐效果,但冷启动问题仍没有得到有效解决。而跨领域推荐算法迁移源域的信息来提高目标域的推荐性能对解决此问题和目标域数据稀疏的问题有着良好效果。CMF<sup>[9]</sup>跨领域推荐算法是将 2 个领域的数据联合起来进行矩阵分解,通过共享用户或项目的中间变量来提高目标域的推荐效果。目前跨领域推荐比较主流的方法是 CBT<sup>[10]</sup>算法,通过将源域矩阵评分拓展为满秩后进行三分解,将分解的中间矩阵作为评分模式共享到目标域中来提高目标域的推荐性能。Ren 等<sup>[11]</sup>提出一种基于概率集群的跨领域推荐算法 PCLF,假定每个用户和项目集群具有相同的效应,每个用户和项目的嵌入是用户属于不同集群的概率向量,通过源域中学习的概率分布迁移至目标域中来提升目标域推荐效果。

本文提出一种基于矩阵分解的跨领域推荐算法,不需要源域和目标域中的项目一致,通过对源域进行矩阵分解,得到用户效应。通过假定同一用户对于不同的项目有着类似的偏好,将源域中分解得到的用户效应迁移至目标域,并在目标域的分解中引入项目集群效应<sup>[12]</sup>。在实际应用上,对于没有足够信息来进行推荐的新的项目类别,可以通过对应用户对其他项目的评分情况来提升新项目类别的推荐效果。

具体是假定同一个集群中的项目是相关的,不同集群中的项目是独立的,即来自同一集群的项目除了自身的项目效应外还有着相同的项目集群效应。具体操作是通过源域进行矩阵分解,

将分解得到的用户效应迁移至目标域中,并在目标域的矩阵分解中同时分解项目效应和项目中潜在的项目集群效应,以此来提高推荐性能。本文贡献在:通过使用源域信息和在目标域中考虑集群效应有效解决数据稀疏性问题;其次对于没有信息的新的项目,通过在目标域中的集群效应来提高预测准确性,有效解决冷启动问题。

## 1 模型与方法

### 1.1 新模型

此方法通过假定来自同一集群的项目除了自身的项目效应外还有着相同的项目集群效应。具体做法是在源域中进行矩阵分解,学习得到用户效应,将用户效应迁移至目标域,在目标域的矩阵分解中引入项目集群效应,将项目的影响解释为用户效应和项目集群效应的共同作用结果。

首先使用矩阵分解将用户对项目的评分矩阵分解成用户效应矩阵和项目效应矩阵。

$R = (r_{ui})_{n \times m}$  为评分矩阵,  $n$  是用户数量,  $m$  是项目数量,  $r_{ui}$  是用户  $u$  对项目  $i$  的评分 ( $u = 1, \dots, n, i = 1, \dots, m$ ), 将其分解为如下形式

$$R = PQ'$$

其中:  $P = (p_1, \dots, p_n)'$  为  $n \times k$  维的用户效应矩阵,  $p_i$  是用户  $i$  的  $k$  维的用户嵌入向量,  $k$  为预先设定的向量维数。同理,  $Q = (q_1, \dots, q_m)'$  为  $m \times k$  维的项目效应矩阵,  $q_j$  是项目  $j$  的  $k$  维项目嵌入向量。

预测评分

$$\hat{r}_{ui} = \hat{p}_u' \hat{q}_i. \quad (1)$$

$\hat{p}_u$  和  $\hat{q}_i$  用交替最小二乘法进行估计, 估计时使用 L2 惩罚<sup>[12]</sup>, 不需要进行参数选择

$$\hat{p}_u = \operatorname{argmin}_{p_u} \sum_{i \in I_u} (r_{ui} - p_u' q_i)^2 + \lambda \|p_u\|_2^2, \quad (2)$$

$$\hat{q}_i = \operatorname{argmin}_{q_i} \sum_{u \in U_i} (r_{ui} - p_u' q_i)^2 + \lambda \|q_i\|_2^2. \quad (3)$$

其中:  $I_u$  是所有用户  $u$  打过评分的项目集合,  $U_i$  是所有给项目  $i$  打过评分的用户集合。

在源域中使用上述方法进行分解, 得到源域中的用户效应和项目效应, 将得到的用户效应迁移至目标域中。

在目标域中, 令  $\phi_{ui} = E(r_{ui})$  为评分的期望, 对  $\phi_{ui}$  做如下分解

$$\phi_{ui} = p_u' (q_i + c_{j_i}). \quad (4)$$

其中: 用户效应  $p_u$  是分解源域得到的,  $c_{j_i}$  是  $k$  维

的项目集群效应, 来自第  $j$  个集群的项目  $J_j = \{i: j_i = j\}$  ( $j = 1, \dots, M$ ),  $M$  为项目集群数量, 所有集群的项目数量和为项目总数  $m$ 。

$C = (c_1, \dots, c_M)'$  为项目集群效应矩阵,  $C_{M \times k}$  和  $Q_{m \times k}$  维数不同, 需要把矩阵  $C_{M \times k}$  进行拓展, 令

$$C_r = (c_1 I'_{|J_1|}, \dots, c_M I'_{|J_M|})'. \quad (5)$$

其中:  $I'_{|J_i|}$  为集群  $i$  中项目个数维数的 1 向量。预测评分矩阵  $\Phi = (\phi_{ui})_{n \times m}$ , 可以得到预测评分的矩阵形式

$$\Phi = P(Q + C_r)'. \quad (6)$$

来自同一集群的项目是相互关联的, 不同集群之间的项目是相互独立的。不同项目形成一个集群是源于它们相似的特征, 评分的数量也是考虑因素之一, 所以此模型不止通过汲取源域信息, 也通过缺失有关的信息进行聚类,  $c_{j_i}$  可用于对新项目的聚类, 从这两方面解决冷启动问题, 提升推荐效果。

### 1.2 参数估计

在目标域中需要估计的参数是  $Q$  和  $C$ , 令  $\varphi$  为向量化的  $(Q, C)$ ,  $O$  是所有有评分的用户-项目组合组成的集合,  $R^0 = \{r_{ui}: (u, i) \in O\}$ , 所以可以定义损失函数如下

$$\mathcal{L}(\varphi | R^0) = \sum_{(u,i) \in O} (r_{ui} - \phi_{ui})^2 + \lambda \left( \sum_{i=1}^m \|q_i\|_2^2 + \sum_{j=1}^M \|c_j\|_2^2 \right). \quad (7)$$

最小化损失函数来估计参数

$$\hat{\varphi} = \operatorname{argmin}_{\varphi} \mathcal{L}(\varphi | R^0). \quad (8)$$

估计出参数后即可进行评分预测, 按评分高低进行推荐

$$\hat{\phi}_{ui} = p_u' (\hat{q}_i + \hat{c}_{j_i}). \quad (9)$$

使用最小二乘法即可估计出项目效应与项目集群效应, 目标函数如下

$$\operatorname{argmin}_{q_i, c_j} \sum_{i \in J_j} \sum_u (r_{ui} - \phi_{ui})^2 + \lambda \left( \sum_{i \in J_j} \|q_i\|_2^2 + \|c_j\|_2^2 \right). \quad (10)$$

对于每个项目集群  $J_j$ , 估计参数

$$\hat{q}_i = \operatorname{argmin}_{q_i} \sum_{u \in U_i} (r_{ui} - \phi_{ui})^2 + \lambda \|q_i\|_2^2, \quad (11)$$

$$\hat{c}_j = \operatorname{argmin}_{c_j} \sum_{i \in J_j} \sum_{u \in U_i} (r_{ui} - \phi_{ui})^2 + \lambda \|c_j\|_2^2. \quad (12)$$

1.3 算法框架

算法主要分为 2 个步骤进行,第 1 步是分解源域,第 2 步是将得到的信息迁移至目标域进行矩阵分解。

算法框架如下:

步骤 1:对源域进行矩阵分解

输入:源域的用户-项目评分数据

设置:

用户和项目的向量维数  $k$ ;

最大迭代次数  $N_1$ ;

学习率  $l_1$  和惩罚系数  $\lambda_1$

1:以随机数初始化用户效应矩阵  $\mathbf{P}$  和项目效应矩阵  $\mathbf{Q}$

2:设置迭代次数  $s = 1$

3: $s < N$  时

4:对每个用户  $u = 1, \dots, n$ , 计算  $\mathbf{p}_u^s$

5:对每个项目  $i = 1, \dots, m$ , 计算  $\mathbf{q}_i^s$

6:设置  $\mathbf{P}^s \leftarrow \mathbf{P}^{s-1}, \mathbf{Q}^s \leftarrow \mathbf{Q}^{s-1}$

输出:估计的  $\hat{\mathbf{P}}$  和  $\hat{\mathbf{Q}}$

步骤 2:在目标域中使用引入集群效应的矩阵分解算法

输入:目标域的用户-项目评分数据和步骤 1 得到的用户效应  $\hat{\mathbf{P}}$

设置:

最大迭代次数  $N_2$ ;

学习率  $l_2$  和惩罚系数  $\lambda_2$ ;

终止条件  $x$ ;

项目组数  $M$

1:以随机数初始化项目效应矩阵  $\mathbf{Q}$  和根据组数生成的项目集群效应矩阵  $\mathbf{C}$

2:估计  $\hat{\mathbf{Q}}$  和  $\hat{\mathbf{C}}$

3:设置迭代次数  $s = 1, s < N$  时

4:设置  $\mathbf{Q}^s \leftarrow \mathbf{Q}^{s-1}, \mathbf{C}^s \leftarrow \mathbf{C}^{s-1}$

5:对每个项目  $i = 1, \dots, m$ , 计算  $\mathbf{q}_i^s$

6:对每个项目集群  $j = 1, \dots, M$ , 计算  $\mathbf{c}_j^s$

7:如果  $\frac{1}{mk} \|\mathbf{Q}^{s_{\text{new}}} - \mathbf{Q}^s\|_F^2 + \frac{1}{Mk} \|\mathbf{C}^{s_{\text{new}}} - \mathbf{C}^s\|_F^2 < x$  或  $s \geq N$  则

停止循环,否则设置  $\mathbf{Q}^s \leftarrow \mathbf{Q}^{s_{\text{new}}}, \mathbf{C}^s \leftarrow \mathbf{C}^{s_{\text{new}}}, s \leftarrow s + 1$  并回到 4

8:得到估计的  $\hat{\mathbf{Q}}$  和  $\hat{\mathbf{C}}$ , 计算预测评分  $\hat{\mathbf{\Phi}} = \mathbf{P}(\hat{\mathbf{Q}} + \hat{\mathbf{C}}_r)$

输出:预测评分  $\hat{\mathbf{\Phi}}$

在集群数量  $M$  是设定的参数,在参数选择上,不同集群数目对预测效果的影响十分微小,如果数据中存在类别的话可以根据现有类别进行分组,数据不存在类别可以根据项目的评分数量按项目个数均分对项目进行预分组<sup>[13]</sup>。

2 实证与结果分析

2.1 实验数据与测评标准

实验使用的是 MovieLens 100K 数据<sup>[14]</sup>,该数据集是由 GroupLens Research 收集的,此数据集包含 943 个用户对 1 682 部电影的 100 000 个评分,评分范围为 1~5,评分充裕度为 6.305%。根据电影的热门程度即评分数量排序,将数据分为

源域和目标域,评分较多的电影作为源域数据,评分较少的电影作为目标域数据,源域包含 943 个用户对 841 部电影的 86 993 个评分,评分充裕度为 10.969%,目标域包含 898 个用户对 841 部电影的评分,评分充裕度为 1.722%。

使用目标域的测试集上预测评分与真实评分的均方误差作为算法性能评价标准<sup>[9]</sup>,展示预测评分与真实评分之间的误差

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{u,i} (\hat{\phi}_{ui} - r_{ui})^2}.$$

(13)

其中  $n$  为目标域中测试集的评分个数。

2.2 算法性能与结果对比

将源域与目标域按照 8 : 2 随机分为训练集与测试集,在选取不同向量维数的参数  $k$  下,各次迭代的 RMSE 结果如图 1 所示。

在不同  $k$  下,算法的预测准确性均在迭代 60 次左右后趋于收敛,之后预测效果保持稳定,表现出良好的收敛性。

在选取循环终止条件  $x = 10^{-30}$  时,不同  $k$  下的迭代次数如表 1 所示。

在  $k$  的选择上,不论是 RMSE 值还是达到收敛的循环次数上,选择向量维数  $k = 3$  均为最优,收敛后的 RMSE = 1.283 2。

不同集群数目  $M$  的选择对预测效果只会造成十分微小的影响,在选择向量维数  $k = 3$  时,不同集群数目的实验结果如表 2 所示。

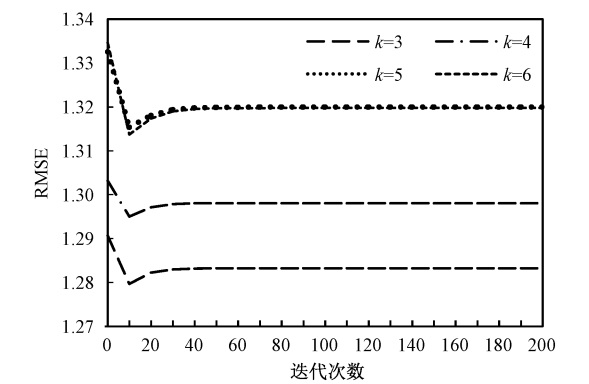


图 1 不同向量维数下的算法 RMSE

Fig. 1 Algorithm RMSE in different vector dimensions

表 1 不同向量维数下的迭代次数

Table 1 Number of iterations with different vector dimensions

向量维数 $k$	跳出循环迭代次数
3	206
4	207
5	239
6	259



表 2 不同集群数目下的 RMSE

Table 2 RSME with different number of clusters

集群数目 $M$	RMSE
5	1.297 2
10	1.283 2
15	1.300 9
50	1.305 1
200	1.308 4

在选择集群数目  $M$  为 10 时具有相对优异的推荐性能。

在算法对比上分别对比 CBT 跨领域推荐算法<sup>[10]</sup>、基于集群分解的 GSSVD 算法<sup>[12]</sup> 和 SVD 算法<sup>[1]</sup>。

CBT 算法各次迭代的 RMSE 如图 2 所示。

GSSVD 算法各次迭代的 RMSE 如图 3 所示。

SVD 算法各次迭代的 RMSE 如图 4 所示。

3 种算法的性能情况如图所示, CBT 算法在极多次迭代后预测效果震荡, 但震荡幅度较小; GSSVD 算法在迭代 2 次后预测效果极好, RSME 达到 1.226, 但第 3 次迭代后就回升到 1.883, 但

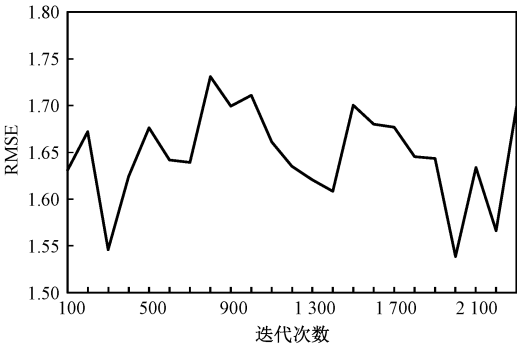


图 2 CBT 算法在不同迭代次数下的 RMSE

Fig. 2 RMSE of CBT algorithm with different number of iterations

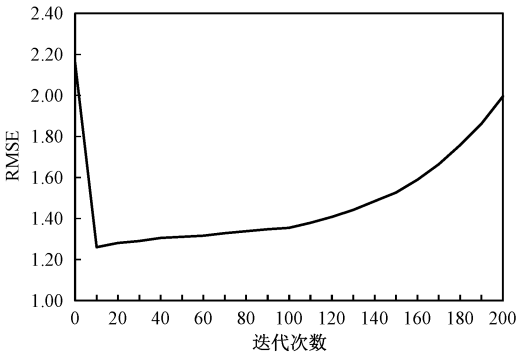


图 3 GSSVD 算法在不同迭代次数下的 RMSE

Fig. 3 RMSE of GSSVD algorithm with different number of iterations

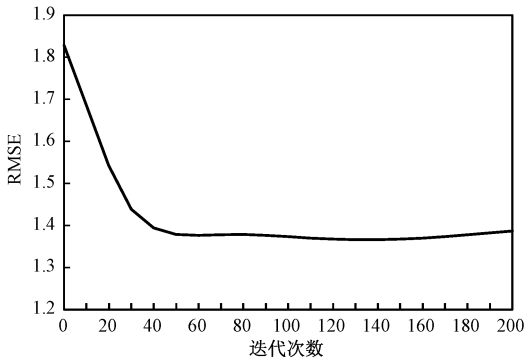


图 4 SVD 算法在不同迭代次数下的 RMSE

Fig. 4 RMSE of SVD algorithm under different number of iterations

随着迭代次数增加预测效果缓慢下降后开始变差; SVD 算法在迭代 60 次左右趋于平稳。

从收敛后的预测稳定性来看, 本文提出的方法在收敛性上具有良好的性能。

除去欠拟合时预测效果不稳定, 几种算法收敛后的 RMSE 和对应的迭代次数如表 3 所示。

除收敛性外, 在预测性能上本文提出的算法也有着比较优异的性能。

表 3 不同算法结果对比

Table 3 Comparison of the results of different algorithms

算法	RMSE	迭代次数
本文方法	1.283 2	42
CBT	1.538 6	2 000
GSSVD	1.284 4	41
SVD	1.365 7	135

3 总结

现实中存在很多具有相似用户的评分数据, 而项目在不同领域内却十分不同, 对于这种情况的一个评分缺失较多的数据, 只需找到一个具有相似用户的信息足够丰富的数据作为源域即可提高该域的推荐性能。

针对此情况, 本文提出一种新的跨领域推荐方法, 在源域中学习用户效应的基础上, 将用户效应迁移至目标域, 并在目标域的分解中引入项目集群效应从而提高推荐性能。与已有算法相比, 它在算法的迭代次数收敛性、收敛后预测效果的稳定性和推荐性能上都有着比较优异的表现。

参考文献

[ 1 ] Mazumder R, Hastie T, Tibshirani R. Spectral regularization algorithms for learning large incomplete matrices[ J ]. The

- Journal of Machine Learning Research, 2010, 11: 2287-2322.
- [ 2 ] Bell R M, Koren Y. Scalable collaborative filtering with jointly derived neighborhood interpolation weights [ C ] // Seventh IEEE International Conference on Data Mining (ICDM 2007). October 28-31, 2007, Omaha, NE, USA. IEEE, 2008; 43-52. DOI: 10.1109/ICDM.2007.90.
- [ 3 ] Salakhutdinov R, Mnih A, Hinton G. Restricted Boltzmann machines for collaborative filtering [ C ] // Proceedings of the 24th international conference on Machine learning. June 20-24, 2007, Corvallis, Oregon, USA. New York: ACM, 2007; 791-798. DOI: 10.1145/1273496.1273596.
- [ 4 ] Blanco-Fernandez Y, Pazos-arias J J, Gil-Solla A, et al. Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems [ J ]. IEEE Transactions on Consumer Electronics, 2008, 54(2): 727-735. DOI: 10.1109/TCE.2008.4560154.
- [ 5 ] Lops P, de Gemmis M, Semeraro G. Content-based recommender systems: State of the art and trends [ M ] // Recommender Systems Handbook. Boston, MA: Springer, 2011; 73-105. 10.1007/978-0-387-85820-3\_3.
- [ 6 ] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions [ J ]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734-749. DOI: 10.1109/TKDE.2005.99.
- [ 7 ] Park S T, Pennock D, Madani O, et al. Naïve filterbots for robust cold-start recommendations [ C ] // Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. August 20-23, 2006, Philadelphia, PA, USA. New York: ACM, 2006; 699-705. DOI: 10.1145/1150402.1150490.
- [ 8 ] Zhu Y Z, Shen X T, Ye C Q. Personalized prediction and sparsity pursuit in latent factor models [ J ]. Journal of the American Statistical Association, 2016, 111(513): 241-252. DOI: 10.1080/01621459.2014.999158.
- [ 9 ] Zhao L L, Pan S J, Xiang E W, et al. Active transfer learning for cross-system recommendation [ C ] // Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence. July 14-18, 2013, Bellevue, Washington. New York: ACM, 2013; 1205-1211. DOI: 10.5555/2891460.2891628.
- [ 10 ] Li B, Yang Q, Xue X Y. Can movies and books collaborate?: cross-domain collaborative filtering for sparsity reduction [ C ] // Proceedings of the 21st International Joint Conference on Artificial Intelligence. July 11-17, 2009, Pasadena, California, .
- [ 11 ] Ren S T, Gao S, Liao J X, et al. Improving cross-domain recommendation through probabilistic cluster-level latent factor model [ C ] // Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. January 25-30, 2015, Austin, Texas. New York: ACM, 2015; 4200-4201. DOI: 10.5555/2888116.2888327.
- [ 12 ] Bi X, Qu A N, Wang J H, et al. A group-specific recommender system [ J ]. Journal of the American Statistical Association, 2017, 112(519): 1344-1353. DOI: 10.1080/01621459.2016.1219261.
- [ 13 ] Wang J H. Consistent selection of the number of clusters via crossvalidation [ J ]. Biometrika, 2010, 97(4): 893-904. DOI: 10.1093/biomet/asq061.
- [ 14 ] Miller B N, Albert I, Lam S K, et al. MovieLens unplugged: experiences with an occasionally connected recommender system [ C ] // Proceedings of the 8th international conference on Intelligent user interfaces. January 12-15, 2003, Miami, Florida, USA. New York: ACM, 2003; 263-266. DOI: 10.1145/604045.604094.