加入标签迁移的跨领域项目推荐算法

葛梦凡 刘 真 王娜娜 田靖玉
(北京交通大学计算机与信息技术学院 北京 100044)

摘 要 大多数推荐算法常采用基于迁移学习的跨领域推荐技术,借助辅助领域的丰富数据信息来解决传统单域推荐中普遍存在的数据稀疏等问题。但若迁移的知识比较单一,没有结合用户行为,则往往会在目标领域导致负迁移、推荐结果不佳等问题。因此,考虑结合其他知识来辅助完成目标领域的学习任务。利用用户异构行为改善推荐结果,正是近年来的新兴研究热点之一。在用户数据中,标签与用户的真实偏好相关,通常能够反映用户或项目的部分隐式特征。通过结合迁移学习及用户标签数据,文中提出了基于标签迁移的跨领域项目推荐算法 ITTCF(Item-based Tag Transfer Collaborative Filtering)。该算法摒弃了在跨领域迁移推荐中仅对评分模式进行挖掘迁移的单一辅助方式,将用户行为反馈与数字评分相结合,融合了评分模式和标签这两种异构用户行为。在多个数据集中的实验结果均表明,ITTCF 具有更好的 *RMSE* 和 *MAE* 值,较传统算法分别提升了 1.61%～6.67%和 1.97%～8.83%。

关键词 迁移学习,跨领域推荐,标签,基于项目的协同过滤
中图法分类号 TP391.9 **文献标识码** A **DOI** 10.11896/jsjkx.180901792

Cross-domaing Item Recommendation Algorithm Including Tag Transfer

GE Meng-fan LIU Zhen WANG Na-na TIAN Jing-yu
(School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract Most recommendation algorithms often use cross-domain recommendation technology based on transfer learning and rich data in the auxiliary domain to solve the problems such as data sparse commonly existing in traditional single domain recommendation. However, if the transtered knowledge is relatively simple without combining user behavior, it will lead to the problems such as negative transfer and poor recommendation results. Therefore, it is possible to combine other knowledge to assist the learning tasks in target domain. Using user heterogeneous behavior to improve recommendation results is one of the emerging research hotspots in recent years. For user data, tags are related to the real user preferences, which can reflect some implicit features of user or item. In light of this, this paper proposed a cross-domain item recommendation algorithm ITTCF (Item-based Tag Transfer Collaborative Filtering) based on tag transfer. Instead of single auxiliary moded of performing mining and migration for rating pattern in cross-domain recommendation, this method combines user behavior feedback and numeric ratings, and fuses two typical user behaviors: rating patterns and tags. Experimental results on multiple datasets show that ITTCF has lower *RMSE* and *MAE* values, and its performance is 1.61% to 6.67% and 1.97% to 8.83% higher respectively than traditional algorithms.

Keywords Transfer learning, Cross-domain recommendation, Tag, Item-based collaborative

1 引言

现今,用户获取信息的渠道与日俱增,但同时也出现了信息过载等问题,因此推荐系统应运而生。顾名思义,推荐旨在捕获用户的兴趣点以完成相应反馈。常见的推荐算法包含协同过滤[1]、基于内容的推荐[2]等。

已有算法大多利用用户、项目的偏好信息在指定的单一领域内进行推荐。但是,传统单域推荐往往存在数据稀疏和冷启动等问题,推荐结果并不理想。因此,可以考虑结合其他领域的数据,利用迁移学习[3]、数据融合[4]等方法辅助完成目标域的学习任务,进行跨领域推荐。

除此之外,在推荐过程中,传统算法大多聚焦于利用用户对项目的显性偏好——数字评分数据。目前,利用诸如浏览记录、点击记录、标签信息等隐性偏好对推荐结果进行改善已成为热门技术之一,并且更能满足现实应用场景的需要。

作为用户和项目间的纽带,标签能够在一定程度上反映用户的兴趣。在现实场景中,用户给出的标签通常是无明确限制的,因此普遍存在标签冗余、模糊等问题。通过对源、目

到稿日期:2018-09-22 返修日期:2019-03-10 本文受国家重点研发计划(2016YFB1200100),中央高校基本科研业务费专项(2017JBM024)资助。

葛梦凡 女,硕士生,主要研究方向为推荐系统,E-mail:16120365@bjtu.edu.cn;**刘 真** 女,副教授,硕士生导师,E-mail:zhliu@bjtu.edu.cn(通信作者);**王娜娜** 硕士生,主要研究方向为推荐系统;**田靖玉** 硕士生,主要研究方向为推荐系统。

标领域数据进行聚类，能够汇聚具有相同特征的标签，从而有效缓解以上问题。因此，当目标领域内的评分、标签数据非常稀疏时，可通过聚合其他领域的标签数据来反映目标领域的项目特征信息，提高推荐的准确度。

本文将这一思想与传统基于项目的协同过滤算法(Item-CF)相结合，提出了一种基于标签迁移的跨领域项目推荐算法。该算法旨在通过迁移相关领域的标签及用户偏好，同时融合评分数据来解决传统单领域算法中推荐结果不佳等问题，缓解目标领域数据的稀疏程度。本文的主要贡献如下：

1)迁移源领域标签及用户偏好等隐式信息，一定程度地解决了仅迁移评分数据时因模式单一导致的负迁移问题。

2)提取了标签中的聚类信息，以此扩展标签数据，一定程度地解决了目标领域数据稀疏等问题。

3)通过两个相关的 UGC(User Generated Content)标签系统进行项目级别的跨领域实验，采用集体迁移模式提高领域间交互，而非仅通过分割一个数据集进行属性级别实验。

2 相关研究

跨领域推荐系统中的常用技术包括：基于知识的推荐、迁移学习、协同过滤等。Cremonesi 等[5]针对领域级别的不同，分别给出了属性、类型、项目、系统 4 种域的定义，旨在根据项目的不同属性和类型对领域概念进行划分。常见的跨领域推荐算法[6]往往基于属性级别，认为类型和属性均相同，而属性值不同的项目属于不同的领域。例如，电影、喜剧片和爱情片即属于不同领域。跨领域推荐的任务即为喜欢看喜剧片的用户推荐合适的爱情片。按照领域间知识处理的不同手段，可将现有技术划分为聚合式推荐[7-9]，以及迁移式[10-12]跨领域推荐。

跨领域推荐的出现，解决了单领域中普遍存在的数据稀疏、冷启动等问题，是目前推荐算法研究的流行趋势。然而，当且仅当满足训练数据和测试数据独立同分布的条件时，大部分机器学习算法才能取得较好的结果，这有悖于现实场景中的操作和应用。迁移学习则放宽了这两大基本假设。目前，迁移学习主要被运用在分类、聚类、协同过滤、人工智能规划等多个研究领域。Pan[3]根据“如何迁移”这一研究要点将现有的研究工作划分为：自适应迁移[13]、集体迁移[14]、聚合式迁移[15]3 类。

自适应迁移是一种有针对性的迁移方法。Li 等[16]提出的 CBT 算法，从辅助矩阵中提取用户-项目评分模式，以缓解目标领域中的稀疏问题。集体迁移是一种具有丰富交互性的双向迁移方法，通常对源、目标领域的知识进行共同学习。与 CBT 算法类似，RMGM[17]算法旨在通过集群级别的评分模型矩阵预测目标任务中的缺失值。聚合式迁移采用嵌入式迁移的技术，将源领域数据作为已知知识纳入目标领域中。其代表性算法 TCF[15]在推荐过程中结合了二进制评分，其目的同样是缓解目标领域的数据稀疏问题。

迁移学习的运用范围愈发广泛。然而，在推荐系统中，除算法常用的用户评分外，还能提取到诸如项目标签、用户行为等隐性偏好数据。目前，主流推荐算法倾向于利用用户偏好、项目偏好、项目特征 3 种方式将用户和项目关联起来[18]。其中，项目标签正是项目特征的主要表现方式之一。因此，越来越多的算法开始利用标签进行推荐，以缓解单一迁移模式下负迁移可能导致的推荐结果不佳等问题。

3 问题建模

定义 1(用户对项目的评分矩阵 $\boldsymbol{R}_t$) 考虑源领域中的用户集合为 $U_s=\{u_{s1},u_{s2},\cdots,u_{sm}\}$，目标领域中包含的用户集合为 $U_t=\{u_{t1},u_{t2}\cdots u_{tm}\}$，项目集合为 $P_t=\{p_{t1},p_{t2},\cdots,p_{tn}\}$。目标领域中的评分矩阵 $\boldsymbol{R}_t$ 如下所示：

$$\boldsymbol{R}_t=\begin{pmatrix} r_{u_1 1} & \cdots & r_{u_1 m} \\ \vdots & & \vdots \\ r_{u_t 1} & \cdots & r_{u_t m} \end{pmatrix}$$

其中，r_{ij} 代表用户 i 对项目 j 的历史评分。若当前不存在评分行为，则该值为空，表示用户尚未浏览关注或对该项目产生过评分操作。

定义 2(用户对项目的标签 tag) 在标签系统中，若用户对项目打上标签，则能同时描述用户兴趣和物品语义。定义源领域中包含的标签为 $T_s=\{t_{s1},t_{s2},\cdots,t_{sn}\}$，目标领域中包含的标签为 $T_t=\{t_{t1},t_{t2},\cdots,t_{tn}\}$，其标签个数分别为 d 和 e。不同领域间的标签可能有所重叠或互不相关。定义仅在目标领域中存在的 tag 为 $TransferTag=\{t\}$。

定义 3(表示相似特征 tag 的主题 $Topic$) 通过标签数据能够得到表示一类标签的主题：当标签 A 和标签 B 频繁用于相同的项目时，说明它们之间具有较大的相似性。本文将主题集合中任意两个主题之间的距离存储在距离矩阵 $\boldsymbol{D}_b$ 中。

如图 1 所示，本文算法利用迁移得到的用户偏好及标签数据计算项目相似度；再结合 $\boldsymbol{R}_t$ 中已有的评分，完成对目标领域评分缺失值的预测。

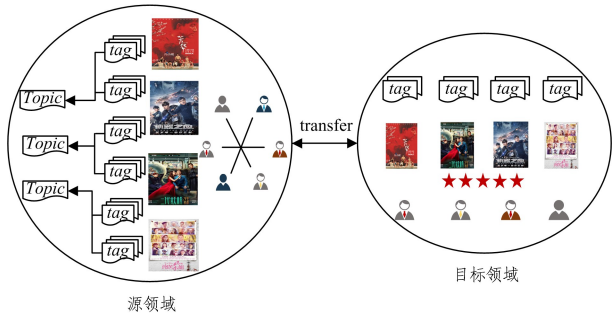


图 1 标签系统迁移模型

Fig. 1 Tag system transfer model

4 基于标签迁移的跨领域推荐

4.1 标签聚类

在标准数据集中，用户给项目打出的标签通常为文本形式的词语或短句。因此，若使用常见的 TFIDF 方法[19]进行处理，会得到非常稀疏的向量文件。为解决该问题，本文通过式(1)－式(3)对标签进行文本向量化处理，同时在 4.1 节、4.2 节中使用原本表示标签的符号来表示转化得到的标签向量。

$$p_z(t)=\sum_{u\in I} q(t|u)q(u|z) \tag{1}$$

$$q(t|u)=\frac{n(u,t)}{\sum\limits_{i\in T_u} n(u,i)} \tag{2}$$

$$q(u|z)=\frac{n(u,z)}{\sum_{i\in T_u} n(u,i)} \tag{3}$$

式(1)中，$P_z(t)$表示领域内部的任一标签 t 出现的情况下，标签 z 同时出现的概率；I 表示领域内的所有用户集合。式(2)和式(3)分别表示当前用户 u 打出该标签 t 和标签 z 在其打过标签总数中的占比；函数 $n(u,t)$表示用户 u 打出标签 t 的次数；T_u 则表示当前用户 u 打过的标签集合。本文在 4.1 节、4.2 节中均使用原本表示标签的符号来表示转化得到的标签向量。

本文首先对源领域标签进行聚类，并将聚类中心定义为标签主题。常用的聚类算法包括 k-means、分层聚类[20] 等。为使标签较为均匀地分布在不同类中，本文采用了自底向上的分层聚类算法，如算法 1 所示。

算法 1　源领域标签聚类算法

输入：源领域标签集合 T_s，初始聚类个数 num，聚类个数阈值 N

输出：主题集合 Topic

1. Topic←{$\mathbf{t}_{s1}$，$\mathbf{t}_{s2}$，…，$\mathbf{t}_{sn}$}
2. num←d
3. FOR EACH $\mathbf{I}\in T_s$ & JT_s
4. 　　$\mathbf{D}_b$=JSD($\mathbf{I}\parallel\mathbf{J}$)
5. ENDFOR
6. IF num＞N
7. 　　T_p T_q=MinDis(D_b)
8. 　　T_p+=T_q REMOVE T_q
9. 　　UPDATE D_b=Dis(T_p，T_q)
10. 　　num←num−1
11. ENDIF

算法 1 中，首先将源领域的每个标签定义为一个聚类中心，然后利用式(4)和式(5)计算任意两个标签向量 $\boldsymbol{I}$ 和 $\boldsymbol{J}$ 之间的 JSD 距离，以此初始化距离矩阵 $\boldsymbol{D}_b$。其中，$\boldsymbol{M}$ 表示向量 $\boldsymbol{I}$ 和 $\boldsymbol{J}$ 的均值。

$$JSD(\boldsymbol{I}\parallel\boldsymbol{J})=\frac{1}{2}D(\boldsymbol{I}\parallel\boldsymbol{M})+D(\boldsymbol{J}\parallel\boldsymbol{M}) \tag{4}$$

$$D(\boldsymbol{I}\parallel\boldsymbol{J})=\sum_{i=1}^{n}\boldsymbol{I}(i)\ln\frac{\boldsymbol{I}(i)}{\boldsymbol{J}(i)} \tag{5}$$

当现有聚类个数大于阈值 N 时，须通过函数 $MinDis$ 遍历矩阵 $\boldsymbol{D}_b$，找到距离最近的两个标签集合 T_p 和 T_q，并将 T_p 中的标签复制至 T_q 中，从而移除 T_p。进一步利用式(6)更新 $\boldsymbol{D}_b$ 矩阵中 T_p 和 T_q 之间的距离。

$$Dis(T_p,T_q)=\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}JSD(\boldsymbol{T}_p(\boldsymbol{i})\parallel\boldsymbol{T}_q(\boldsymbol{j}))}{m\cdot n} \tag{6}$$

其中，m 和 n 分别表示 T_p 和 T_q 中的标签个数。

4.2　标签迁移

完成源领域的标签聚类后，须进一步融合目标领域标签。利用式(6)计算目标领域中的任意标签 $\boldsymbol{t}_i$ 和任意主题 T_j 之间的距离，并将 $\boldsymbol{t}_i$ 迁移至距离最短的主题集合中。本文可以把 $\boldsymbol{t}_i$ 看作只有一个标签向量的主题集合，以此进行距离计算。

因源目标领域间的重叠标签已存在于主题集合中，所以仅需对只存在于目标领域中的独有标签进行进一步迁移，具体步骤如算法 2 所示。

算法 2　目标领域标签迁移算法

输入：最大距离 D_{max}，源领域标签集合 T_s，目标领域标签集合 T_t

输出：更新后的主题集合 Topic

1. transferTag←{$\mathbf{t}_i\in T_s$，$\mathbf{t}_i\notin T_s$}
2. FOR EACH $\mathbf{i}\in$ TransferTag $\in$ jTopic
3. 　　D_{min}=FindMin(Dis($\mathbf{t}_i$，T_j))
4. 　　IF(D_{min}＜D_{max})
5. 　　T_j+=$\mathbf{t}_i$
6. 　　ELSE Topic+=$\mathbf{t}_i$
7. ENDFOR

需要注意的是，不同领域间的标签可能会存在较大差异。因此，本文考虑设置阈值 D_{max}，使用函数 FindMin 获取标签 $\boldsymbol{t}_i$ 与 $Topic$ 中主题的最小距离 D_{min}。当 D_{min} 大于阈值时，以该标签为聚类中心生成新的类，以避免迁移过程中标签和主题间距离过大而造成不合理的情况。

完成上述步骤后，在原有集合的基础上得到了一个新的 $Topic$ 集合，其中包含了源、目标领域的全部标签；同时也初步回答了迁移学习中“迁移什么”这一重要问题，即完成了源领域至目标领域的标签及主题迁移。

4.3　主题偏好迁移

在推荐过程中，迁移源领域的用户偏好能在一定程度上缓解目标领域的数据稀疏问题，提高相似性计算的准确度，进而改善推荐结果。迁移后，目标领域中项目和主题的相关性，将基于源、目标领域中的标签偏好权重进行计算。迁移后主题偏好的计算框架如图 2 所示，算法伪代码如算法 3 所示。

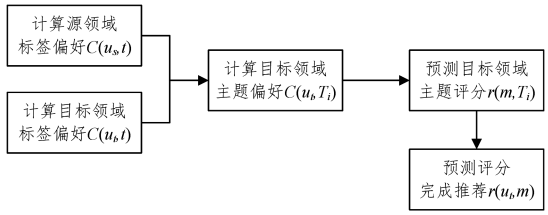


图 2　主题偏好计算框架

Fig. 2　Topic preference calculation framework

算法 3　主题偏好迁移算法

输入：源领域标签集合 T_s，目标领域标签集合 T_t，主题集合 Topic，聚类个数 num

输出：主题预测评分 $\hat{r}_{m,T}$

1. FOR EACH $u_s\in U_s$ $t\in T_s$
2. 　　souCor=C(u_s，t)
3. ENDFOR
4. FOR EACH $u_t\in U_t$ $t\in T_t$
5. 　　tarCor=C(u_t，t)
6. ENDFOR
7. FOR EACH $u_t\in U_{tTi}\in$ Topic
8. 　　topicCor=C(u_t，T_i)
9. ENDFOR
10. FOR EACH $m\in P_{tTi}\in$ Topic
11. 　　$\hat{r}_{m,T_i}$=R(m，T_i)
12. ENDFOR

如算法 3 所示，首先需要计算源、目标领域中用户 u 和标

签 t 的相关性，即用户标签偏好 $C(u,t)$。本文采用式(7)对其进行计算。

$$C(u,t)=\frac{\sum_{z\in T_u} n(u,z)p_z(t)}{\sum_{z\in T_u} n(u,z)} \tag{7}$$

根据上文计算得到的用户标签偏好，可进一步通过式(8)迁移源领域的隐式偏好信息，得出目标领域用户和主题间的相关偏好 $C(u,T_i)$。

$$C(u,T_i)=\sum_{t_t\in T_i}\sum_{t_s\in T_i}\lambda\cdot C(u,t_s)+\beta\cdot C(u,t_t) \tag{8}$$

其中，t_s 表示主题 T_i 中的源领域标签，t_t 表示目标领域标签，λ 和 β 分别代表源、目标领域标签偏好的权重因子。

利用上述结果，根据式(9)计算项目对主题的预测评分。

$$\hat{r}_{m,T_i}=\frac{\sum_{u\in I_m} C(u,T_i)r_{u,m}}{\sum_{u\in I_m} C(u,T_i)} \tag{9}$$

其中，I_m 表示所有对项目 m 有过评分行为的用户合集。

由上述公式可知，在计算预测评分时，并未使用源领域的评分数据，仅结合了迁移得到的隐式偏好信息。这有别于大多数迁移算法，使得实验中对源领域数据集格式要求较低。

4.4 ITTCF 推荐算法

综合算法 1－算法 3，本文提出结合标签迁移和传统协同过滤的跨领域项目推荐算法 ITTCF，该算法的框架如图 3 所示。

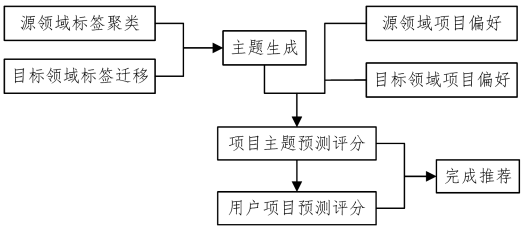


图 3 ITTCF 算法框架

Fig. 3 ITTCF algorithm framework

如果用 U_{mn} 表示曾对项目 m 和项目 n 都有过评分行为的用户集合，则传统的皮尔斯相似度计算方式如下：

$$sim(m,n)=\frac{\sum_{i\in U_{mn}}(r_{i,m}-\bar{r}_m)(r_{i,n}-\bar{r}_n)}{\sqrt{\sum_{i\in U_{mn}}(r_{i,m}-\bar{r}_m)^2\sum_{i\in U_{mn}}(r_{i,n}-\bar{r}_n)^2}} \tag{10}$$

其中，$\bar{r}_m$ 和 $\bar{r}_n$ 分别表示项目 m 和 n 的历史平均评分。

传统算法通过已知评分数据计算项目之间的相似度，完成对缺失值的预测，旨在为用户推荐与其已评分项目较为相似的商品。其只考虑了对两个项目有过共同评分的用户，可利用的推荐预测信息量较少。根据 4.3 节中式(9)计算的项目对主题的预测评分，对传统 ItemCF 算法中的相似度计算方法进行改进。本文引入式(9)中项目对主题的预测评分，改进后的相似度计算方法如式(11)所示：

$$S(m,n)=\frac{\sum_{T_i\in T_{m,n}}(\hat{r}_{m,T_i}-\bar{r}_{m,T_i})(\hat{r}_{n,T_i}-\bar{r}_{n,T_i})}{\sqrt{\sum_{T_i\in T_{m,n}}(\hat{r}_{m,T_i}-\bar{r}_{m,T_i})^2}\sqrt{\sum_{T_i\in T_{m,n}}(\hat{r}_{n,T_i}-\bar{r}_{n,T_i})^2}} \tag{11}$$

其中，$\hat{r}_{m,T_i}$ 表示式(9)中得到的项目 m 对主题 T_i 的预测评分，表示项目 m 相关主题的平均评分。

最后，根据式(12)计算目标领域中任一用户对项目的预测评分。

$$r_{u,m}=\bar{r}_m+\frac{\sum_{n\in m_u S(m,n)>0} s(m,n)(r_{u,n}-\bar{r}_n)}{\sum_{n\in m_u S(m,n)>0}|s(m,n)|} \tag{12}$$

其中，$\bar{r}_m$，$\bar{r}_n$ 分别为项目 m，n 的平均评分；m_u 表示对项目 m 有过评分的用户合集。当物品间的相似度大于 0 时，可推断两个物品有一定的近邻性，则为当前计算项赋予相似度权重。

5 实验结果及分析

5.1 实验数据及实验环境

本文采用了 MovieLens[21]、豆瓣电影、豆瓣图书 3 个数据集。其中，MovieLens 是最常用的电影推荐数据集。第 2 节曾提到，领域的定义可以是不同级别的。本文将根据项目属性级别首先对其进行划分，即进行同为电影属性的项目推荐。为得到标签数据较为稠密的源领域和较为稀疏的目标领域，经过多次实验，筛选得到源领域中每个项目使用的标签数大于 15，即每个项目都有超过 15 条标签记录。相应地，目标领域中每个项目使用的标签个数少于 8，以此保证目标领域的数据更为稀疏。

豆瓣网站中包含音乐、电影、图书 3 个领域的数据。其不同领域间虽存在一定重叠，但各自包含不同的项目集合，因此很适合用来进行项目级别的跨领域推荐。在实验中除了对豆瓣电影领域内部进行划分和推荐外，还将进行豆瓣图书至豆瓣电影领域的推荐，旨在更好地模拟真实世界的推荐场景。而在不同的数据集中，领域的划分也需要做出调整。由于豆瓣数据中标签的类型和数量相对较少，因此其与 MovieLens 数据集的处理稍有不同。在实验中将使用 3 个以上标签的项目划至源领域，而低于 3 个标签的项目则被划分至目标领域。

由于数据集中的标签质量参差不齐，为得到更好的实验效果，需对标签文件进行进一步去噪。经多次实验调整，删除少于 2 个用户使用及少于 5 个项目应用的无意义标签。

最后，实验中应用的数据集如表 1 所列，其中 S 和 T 代表源、目标领域。本文将分别进行 S_1 至 T_1、S_2 至 T_2、S_3 至 T_2 这 3 种不同领域级别的跨领域推荐。

表 1 本文使用的数据集

Table 1 Experimental data sets

数据集	领域	用户	项目	标签
MovieLens 10M	S_1	536	546	2178
MovieLens 10M	T_1	392	862	2058
豆瓣电影	S_2	1666	171	8651
豆瓣图书	S_3	161	654	1802
豆瓣电影	T_2	2299	251	6436

本文实验使用 Scala2.11.7 编写，环境为 macOS 操作系统；16GB 内存，2.9GHz Intel Core i5 CPU。

5.2 算法评价指标

实验中分别采用均方根误差 RMSE、平均绝对误差 MAE 对算法的推荐准确度进行对比，其计算公式分别如下：

$$RMSE=\sqrt{\sum_{i=1}^{n}\frac{1}{n}(observed_i-predict_i)^2} \tag{13}$$

$$MAE=\frac{\sum_{i=1}^{n}|observed_i-predict_i|}{n} \tag{14}$$

其中，$observerd_i$ 表示测试集中的真实评分；$predict_i$ 表示算法计算得到的预测评分。因 *RMSE* 与 *MAE* 代表预测值与真实值之间的差距，所以其值越小，推荐的结果越准确。

5.3　实验结果分析

在实验过程中，首先对相关参数进行选择，实验结果如表 2、表 3 所列。针对不同数据集，分别选取源、目标领域标签偏好权重的最优值。

表 2　β 不变而 λ 变化时的 *RMSE* 值

Table 2　*RMSE* value when only λ changes

数据集	λ=0.36	λ=0.38	λ=0.40	λ=0.42	λ=0.44
S_1-T_1	0.910	**0.906**	0.923	0.921	0.929
S_2-T_2	0.883	**0.882**	0.891	0.893	0.895
S_3-T_2	0.905	0.889	0.888	**0.885**	0.900

表 3　λ 不变而 β 变化时的 *RMSE* 值

Table 3　*RMSE* value when only β changes

数据集	β=0.58	β=0.60	β=0.62	β=0.64	β=0.66
S_1-T_1	0.918	0.910	0.919	**0.906**	0.921
S_2-T_2	0.896	**0.882**	0.903	0.896	0.904
S_3-T_2	0.891	0.901	**0.885**	0.886	0.893

由表 2、表 3 可知，针对不同数据集，各有相应的最优值。实验进一步观察 *MAE* 值随权重因子的变化情况，如表 4、表 5 所列。

表 4　β 不变而 λ 变化时的 *MAE* 值

Table 4　MAE value when only λ changes

数据集	λ=0.36	λ=0.38	λ=0.40	λ=0.42	λ=0.44
S_1-T_1	0.690	**0.688**	0.694	0.705	0.713
S_2-T_2	0.645	**0.634**	0.646	0.646	0.647
S_3-T_2	0.648	0.644	0.643	**0.642**	0.656

表 5　λ 不变而 β 变化时的 *MAE* 值

Table 5　*MAE* value when only β changes

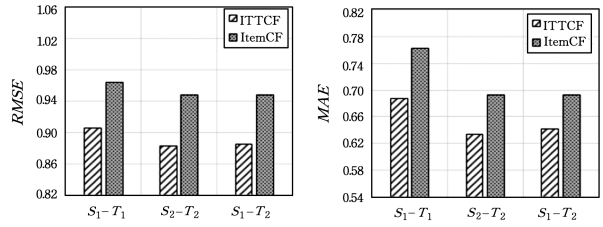
数据集	β=0.58	β=0.60	β=0.62	β=0.64	β=0.66
S_1-T_1	0.702	0.703	0.706	**0.688**	0.697
S_2-T_2	0.651	**0.634**	0.656	0.645	0.655
S_3-T_2	0.644	0.650	**0.642**	0.643	0.644

由表 4、表 5 可知，λ 和 β 的最优值始终维持在一定的变化范围内。特别地，因迁移学习中源领域数据只起到辅助作用，故在主题偏好计算中目标领域权重的占比仍较大。

在 ITTCF 中，初始主题是通过源领域已有标签训练得到的。因此，需要通过实验获取最优的主题个数。在预处理后得到的 MoviLens 10 M 数据集中，保持主题间的最大距离为 0.6 不变，每次增加 5 个类。最后设定 S_1 至 T_1 的源领域主题聚类个数为 20。算法中，$D_{\max}$ 也同样根据现有数据训练得到，其值取自 0.1 至 0.9，每次增加 0.1，保持主题个数为 20 不变。经实验验证，将 $D_{\max}$ 设置为 0.6。

与 MovieLens 不同，豆瓣数据集的标签冗余性低，是项目自身的属性偏好，因此更适合进行基于标签的推荐。由表 1—表 4 也可以看出，在不同参数下，豆瓣数据集的实验结果普遍优于 MovieLens 的实验结果。实验中依然保持 $D_{\max}$ 为 0.6，在豆瓣电影领域中，当主题个数为 3 时，实验结果最佳；而在豆瓣图书领域中，当主题个数为 10 时，其 *RMSE* 值最优。而保持主题个数不变时，最大距离依然在取 0.6 时结果最佳。

与传统 ItemCF 算法一样，ITTCF 也是基于项目的推荐。因此，本文首先将 ITTCF 与 ItemCF 算法进行对比，结果如图 4 所示。



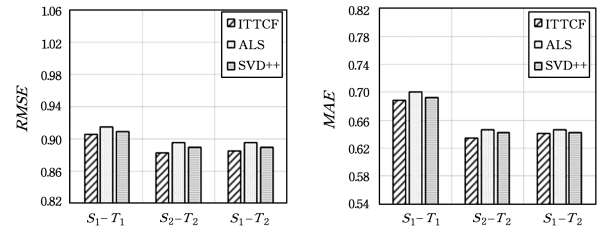
(a)*RMSE* 的对比　　(b)*MAE* 的对比

图 4　本文算法与 Item CF 的性能对比

Fig. 4　Performance comparison of this method and ItemCF

从图 4(a)可以看出，本文提出的算法在不同领域实验内的 *RMSE* 值均低于 ItemCF 算法的 *RMSE* 值：在 MovieLens 10 M 数据集上低 5.6%，在豆瓣电影数据集上低6.9%。而当进行项目级别的跨领域推荐时，ITTCF 的 *RMSE* 值依然大幅低于 ItemCF 算法。图 4(b)为不同数据集上 *MAE* 的对比结果，观察可知 ITTCF 的 *MAE* 值均低于对比算法的 *MAE* 值。以上结果证明了本文算法的推荐结果优于对比算法，进一步体现了利用迁移技术缓解目标领域数据稀疏性的有效性。

接下来，本文将 ITTCF 与基于模型的经典矩阵分解算法即 ALS[22] 算法和 SVD++[23] 算法进行对比，以验证其全面性。其中，SVD++在矩阵分解中进一步加入了相关因子，与 ITTCF 一样利用了用户的隐式偏好信息进行推荐。如图 5 所示，ITTCF 在 3 个数据集上的实验结果皆优于对比算法。其中，S_1-T_1，S_2-T_2 领域的效果提升较为明显，这得益于同领域内具有较高的用户重叠程度。在 S_3-T_2 领域中，本文算法较 SVD++在 *RMSE* 和 *MAE* 上的差别均在 1%以内，这从侧面反映了算法在应对项目级别的跨领域环境时需要做出进一步的优化，包括迁移内容的多项融合、迁移模式交替等，这也将成为算法后续的研究方向。

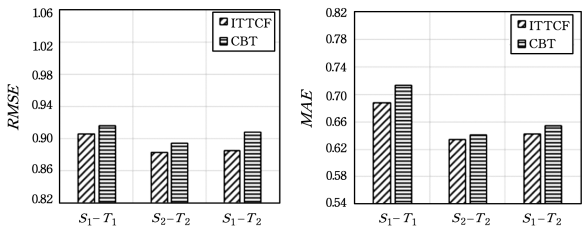


(a)*RMSE* 的对比　　(b)*MAE* 的对比

图 5　本文算法与 ALS 和 SVD++的性能对比

Fig. 5　Performance comparison of this method with ALS and SVD++

进一步将 ITTCF 与 CBT[16] 算法进行对比，结果如图 6 所示。CBT[16] 算法作为最早将迁移学习运用在跨领域推荐的基线算法之一，在实验过程中仅对评分模式进行迁移。有别于 ITTCF 利用异构行为的方式，其能够验证融合用户行为的标签迁移对推荐结果造成的影响。



(a)*RMSE* 的对比 (b)*MAE* 的对比

图 6 本文算法与 CBT 的性能对比

Fig. 6 Performance comparison of this method with CBT

由图 6 可知,不同数据集中,CBT 算法的 *RMSE* 及 *MAE* 值均高于 ITTCF 算法。但与图 4 中 ItemCF 算法的测评指标相比,CBT 的值仍较低。这既验证了当领域内部数据稀疏时,迁移技术能够在一定程度上缓解这个问题;也反映出了除数字评分外,标签文件对推荐结果有着积极的影响。

结束语 本文提出的 ITTCF 算法首先解决了“迁移什么”的问题,运用分层聚类技术对源领域的稠密标签数据进行了聚类,并将聚类结果作为知识迁移至稀疏的目标领域,结合传统 ItemCF 算法辅助完成学习任务。最后在 Movielens 10 M 数据集和豆瓣数据集上,将 ITTCF 算法与 4 种经典算法进行对比。实验结果表明,加入项目的标签属性偏好能够使推荐结果更加准确。算法在迁移模式上对源、目标领域进行共同学习,能够在一定程度上避免负迁移的发生,是一种具有双向性及丰富交互的集体式迁移。

未来,我们将试图找到一种更好的迁移算法与 ITTCF 进行项目级别的跨领域对比实验。另外,考虑融合更多的用户行为信息,如收藏、浏览记录等,以更加丰富的数据辅助完成预测评分,缓解数据稀疏等问题,从而得到更好的推荐结果。

参 考 文 献

[1] BELLOGÍN A, CANTADOR I, CASTELLS P. A comparative study of heterogeneous item recommendations in social systems [J]. Information Sciences, 2013, 221(1): 142-169.

[2] IVÁN C, VALLET D. Content-based recommendation in social tagging systems[C]// ACM Conference on Recommender Systems. ACM, 2010: 237-240.

[3] PAN W. A survey of transfer learning for collaborative recommendation with auxiliary data[J]. Neurocomputing, 2016, 177(C): 447-453.

[4] MA F, WANG W, DENG Z. TagRank: A New Tag Recommendation Algorithm and Recommender Enhancement with Data Fusion Techniques[M]// Social Media Retrieval and Mining. Berlin: Springer, 2013: 80-91.

[5] CREMONESI P, TRIPODI A. Cross-Domain Recommender Systems[C]// 11th IEEE International Conference on Data Mining Workshops. IEEE Computer Society, 2011: 496-503.

[6] CHEN L, ZHENG J, GAO M, et al. TLRec: Transfer Learning for Cross-Domain Recommendation[C]// 2017 IEEE International Conference on Big Knowledge (ICBK). IEEE Computer Society, 2017: 167-172.

[7] ZHUANG F, LUO P, XIONG H, et al. Cross-Domain Learning from Multiple Sources: A Consensus Regularization Perspective [J]. IEEE Transactions on Knowledge & Data Engineering, 2010, 22(12): 1664-1678.

[8] TIROSHI A, KUFLIK T. Domain Ranking for Cross Domain Collaborative Filtering[M]// User Modeling, Adaptation, and Personalization. Berlin: Springer, 2012: 328-333.

[9] LONI B, SHI Y, LARSON M, et al. Cross-domain collaborative filtering with factorization machines[C] //European Conference on Information Retrieval. Springer, 2014: 656-661.

[10] AZAK M. Crossing: A Framework to Develop Knowledge-based Recommenders in Cross Domains[D]. Middle East Technical University, 2010.

[11] KAMINSKAS M, RICCI F. A generic semantic-based framework for cross-domain recommendation[C]// International Workshop on Information Heterogeneity and Fusion in Recommender Systems. ACM, 2011: 25-32.

[12] SHI Y, LARSON M. Tags as bridges between domains: improving recommendation with tag-induced cross-domain collaborative filtering[C]// 19th International Conference on User Modeling, Adaption and Personalization. Springer, 2011: 305-316.

[13] WAN J, WANG X, YIN Y, et al. Transfer Learning in Collaborative Filtering for Sparsity Reduction Via Feature Tags Learning Model[C]// Computer Science and Technology. 2015: 56-60.

[14] ADAMS R P, DAHL G E, MURRAY I. Incorporating side information into probabilistic matrix factorization using Gaussian processes[C]// Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence. ACM, 2010: 1-9.

[15] PAN W, XIANG E W. Transfer learning in collaborative filtering with uncertain ratings[C]// Twenty-Sixth AAAI Conference on Artificial Intelligence. AAAI Press, 2012: 662-668.

[16] LI B, YANG Q, XUE X. Can Movies and Books Collaborate? Cross-Domain Collaborative Filtering for Sparsity Reduction [C]// Proceedings of 2009 International Joint Conference on Artificial Intelligence. Morgan Kaufmann, 2009: 2052-2057.

[17] LI B, YANG Q, XUE X. Transfer learning for collaborative filtering via a rating-matrix generative model[C]// International Conference on Machine Learning. ACM, 2009: 617-624.

[18] VIG J, SEN S, RIEDL J. Tagsplanations: explaining recommendations using tags[C]// International Conference on Intelligent User Interfaces. ACM, 2009: 47-56.

[19] SALTON G, BUCKLEY C. Term-weighting approaches in automatic text retrieval[J]. Information Processing & Management, 1988, 24(88): 513-523.

[20] SHEPITSEN A, GEMMELL J, MOBASHER B, et al. Personalized recommendation in social tagging systems using hierarchical clustering[C]// Proceedings of the 2008 ACM Conference on Recommender Systems(RecSys 2008). ACM, 2008: 259-266.

[21] HARPER F M, KONSTAN J A. The MovieLens Datasets: History and Context[J]. ACM Transactions on Interactive Intelligent Systems, 2016, 5(4): 1-19.

[22] WINLAW M, HYNES M B, CATERINI A, et al. Algorithmic Acceleration of Parallel ALS for Collaborative Filtering: Speeding up Distributed Big Data Recommendation in Spark[C]// IEEE International Conference on Parallel and Distributed Systems. IEEE, 2016: 682-691.

[23] BELL R M. Lessons from the Netflix prize challenge[J]. ACM SIGKDD Explorations Newsletter, 2007, 9(2): 75-79.