

基于评分子空间和信任机制的协同过滤推荐算法

张 莉,孙晓寒,郑晓晗

(苏州大学计算机科学与技术学院,江苏 苏州 215006)

[摘要] 互联网技术的快速发展导致了互联网上数据信息的爆炸式增长. 推荐系统作为解决互联网信息过载问题的关键技术,其核心思想是通过用户历史行为数据挖掘出用户的个性化偏好,为用户推荐其感兴趣的物品. 然而,稀疏的评分数据会导致相似度计算不够准确,进而影响相似用户集的质量. 为了提高相似用户搜索的可靠性,引入信任机制和评分子空间,提出基于评分子空间和信任机制的协同过滤推荐算法. 创新点主要包括以下两点:首先,算法引入基于用户显式声明的关系数据所构建的信任机制,该关系数据能够对稀疏的评分数据进行补充. 其次,利用评分子空间和信任关系,设计了一种基于隐式和显式相似度的混合相似度度量方式,并将之引入到多阶近邻的相似用户搜索方法和迭代评分预测方案中. 实验结果表明,所提算法提高了推荐的准确度,具备较好的预测能力.

[关键词] 推荐系统,协同过滤,信任机制,用户评分子空间,迭代评分预测

[中图分类号] TP391 **[文献标志码]** A **[文章编号]** 1672-1292(2023)03-0027-09

Collaborative Filtering Method Based on User Rating Subspace and Trust Mechanism for Recommendation System

Zhang Li, Sun Xiaohan, Zheng Xiaohan

(School of Computer Science and Technology, Soochow University, Suzhou 215006, China)

Abstract: The rapid growth of the internet has led to the explosive growth of information on the internet. To solve the issue of information overload, recommendation system has been proposed. The core idea behind recommendation system is to explore users' personalized preferences based on users' historical behavior data, and recommend items that match users' interests to users. However, sparse rating data leads to poor accurate similarity calculations, which in turn affect the quality of similar user sets. To improve the reliability of similar user set, this paper proposes a collaborative filtering algorithm based on user rating subspace and trust mechanism (URSTM) for recommendation system. The innovation of this paper contains the following two main points. Firstly, URSTM introduces a trust mechanism constructed based on a trust relationship explicitly declared by users, which can supplement the sparse rating data. Secondly, by using the rating subspace and trust relationship, this paper designs a hybrid similarity measurement based on explicit and implicit similarities, and then integrates it into the multi-order nearest neighbor search method and iterative rating prediction method. Experimental results show that URSTM can improve the accuracy of recommendation performance and has a better prediction ability.

Key words: recommendation system, collaborative filtering, trust mechanism, user rating subspaces, iterative rating prediction

互联网技术的变革与普及为人们提供了一种新的生活方式. 随之产生的海量数据也使得人们难以选择自己真正需要的信息. 针对人与信息不对称的问题, Xerox 公司设计的邮件系统将推荐系统带入了人们的视野^[1]. Resnick 等^[2]在 1997 年对推荐系统的学术概念进行了规范化定义. 作为一种信息处理技术,推荐系统通过用户的历史信息判断用户喜好,在不需要与用户发生过多互动的情况下,提高用户筛选信息的效率^[3-4]. 推荐系统近年来在工业界的应用十分广泛. 就用户而言,当用户需求比较模糊时,推荐系统能够利用用户的各类历史数据预测其可能需要的信息. 就工业用户而言,推荐系统可以帮助公司更限度地

收稿日期:2023-04-24.

基金项目:江苏省高校自然科学研究项目(19KJA550002)、江苏省六大大人才高峰项目(XYDXX-054)、江苏高校优势学科建设工程资助项目.

通讯作者:张莉,博士,教授,研究方向:机器学习,模式识别,神经网络和智能信息处理. E-mail:zhangliml@suda.edu.cn

吸引用户,增加用户留存率和用户粘性,提高公司收益.推荐系统在诸如电商、社交、广告等互联网服务中都有着广泛的应用^[5-8].

不同的应用场景衍生出了各类推荐算法.在众多推荐系统算法中,协同过滤是目前应用得最广泛的推荐算法.其核心思想是参考大家的意见一起对海量信息进行过滤和处理,从中筛选出目标用户可能感兴趣的信息^[9-11].协同过滤算法能够为推荐列表提供良好的可解释性,并能获得较好的推荐效果.然而,面对海量物品,每个用户浏览物品的能力是有限的.这将导致传统的协同过滤算法无法根据历史信息准确地勾画用户画像,进而影响推荐质量^[12-13].另一方面,现有的一些协同过滤算法从用户的历史评分数据中挖掘用户之间的潜在关系,但用户本身对这种潜在关系并不一定赞同.

为了提高相似用户搜索的可靠性,本文通过引入信任机制,提出了基于评分子空间和信任机制(user rating subspace and trust mechanism,URSTM)的协同过滤推荐算法.该算法涉及到的信任关系源于用户显式声明过的社交关系数据,该数据能够对稀疏的评分数据进行评分补充.利用用户的信任关系,本文设计了一种基于评分子空间和信任机制的相似性度量方式,并将之引入到多阶近邻的相似用户搜索方式和迭代预测方法中.实验验证了所提算法的有效性.

1 相关工作

作为推荐算法的典型代表,协同过滤算法根据相似度搜索目标用户的相似用户集,同保持可以分为基于邻域的协同过滤^[14-15]和基于模型的协同过滤^[16].本文主要关注的是基于邻域的协同过滤算法.

基于邻域的协同过滤算法,首先需要根据用户之间的相似性搜索目标用户的相似用户集,然后利用相似用户集提供的信息来预测目标用户对物品的评分.由于历史评分数据规模十分庞大,为了提高计算效率,在相似用户搜索阶段,一般采用形式比较简单的相似度度量算法来计算用户之间的相似性,如欧氏距离^[17]、余弦相似度^[18]、杰卡德系数^[19]等.这些相似度度量的计算涉及到用户之间的共同历史评分记录.

然而,在实际应用中的评分数据存在稀疏性,用户之间的共同评分物品可能非常少.因此,上述相似度度量算法计算得到的相似度会存在一定偏差,而且相似度的不准确性又会使得相似用户集的质量变差.为了提高相似用户集的质量,研究人员引入“朋友的朋友也是朋友”的想法,提出了一些迭代的协同过滤算法.如 Margaritis 等^[20]提出的协同过滤算法(collaborative filtering based on friend of a friend,CFfoaf),Sun 等^[21]提出的多阶近邻预测(multi-order nearest neighbor prediction,MNNP)方法.但是这些方法注重了相似用户的质量,忽略了用户的评分习惯和多样性问题.

另外,相似用户集的搜索过程除了利用用户之间的相似性,也可以基于子空间的概念.Ramezani 等^[22]提出一种基于子空间近邻树(users' tree accessed on subspace,UTAOS)的推荐算法.该算法将评分数据划分为感兴趣子空间和不感兴趣子空间,然后旨在感兴趣子空间中构造近邻树.UTAOS 忽略了不感兴趣子空间中的信息.为此,Koohi 等^[23]提出一种基于子空间聚类的近邻用户(neighbor users by subspace clustering on collaborative filtering,NUSCCF)推荐算法.NUSCCF 将用户评分数据划分为感兴趣子空间、中立子空间以及不感兴趣子空间.该算法在 3 个子空间中分别构造近邻树,然后分别搜索目标用户的相似用户集.UTAOS 和 NUSCCF 都是基于用户兴趣模式构造近邻树.这类方法的缺点是在近邻树的构造过程中需要大量的内存空间来存储兴趣模式结果.基于 3 个子空间,孙晓寒等^[24]提出了基于评分区域子空间(rating region subspace,RRS)的协同过滤推荐算法.该算法考虑到评分用户的惰性习惯,只认定在 3 个子空间都有高相似度的用户对才是真正相似的.上述和相似用户集相关的方法都从用户的历史评分数据中挖掘出用户之间的潜在关系,用户本身对这种潜在关系并不一定赞同,也就是说用户并没有声明这种相似关系.

为了缓解评分数据稀疏性和评分行为多样性问题,我们在评分子空间基础上,引入了通过用户显示声明的关系数据所构建信任机制,该关系数据能够对稀疏的评分数据进行补充.基于评分子空间和信任机制,我们设计了一种新的相似度度量方式.通过将该相似性度量方式引入到多阶近邻的相似用户搜索和迭代预测方法中,也可以在一定程度上缓解评分行为多样性问题.

2 本文算法

先给出问题定义.令集合 $U = \{u_1, u_2, \dots, u_n\}$ 是用户集合, $T = \{t_1, t_2, \dots, t_m\}$ 是物品集合, n 和 m 分别代

表用户总数和物品总数. 用户对物品的评分矩阵可表示为 $\mathbf{R} = [r_{ij}]^{n \times m}$, r_{ij} 表示用户 u_u 对物品 t_j 的评分值. 一般来说, r_{ij} 值越高表示用户 u_u 越喜欢物品 t_j . 下面, 我们基于子空间和信任机制定义了一种新的相似度计算方式, 其中子空间提供隐式相似度, 社交关系数据提供显式相似度.

2.1 隐式相似度

为了定义本文的隐式相似度, 构造文献[24]所提供的评分子空间. 在评分矩阵 \mathbf{R} 中, 利用评分最高值 r_{\max} 和评分最低值 r_{\min} 计算阈值 α 和 β . 即 $\alpha = [(r_{\max} + r_{\min})/2]$, $\beta = [(r_{\max} + r_{\min})/2]$. 然后根据 α 和 β 将评分划分为 3 个级别. 评分值大于 α 的为高级别评分, 小于 β 的为低级别评分, 在区间 $[\beta, \alpha]$ 内的为中级别评分.

按照不同的评分级别为每个用户搜索其评分过的物品子空间. 对于用户 u_u 来说, T_u^H 、 T_u^M 和 T_u^L 分别表示该用户高评分、中评分以及低评分的物品集合, 可分别表示为:

$$T_u^H = \{t_j | r_{uj} > \alpha, j = 1, \dots, m\}, \quad (1)$$

$$T_u^M = \{t_j | \beta \leq r_{uj} \leq \alpha, j = 1, \dots, m\}, \quad (2)$$

$$T_u^L = \{t_j | r_{uj} < \beta, j = 1, \dots, m\}. \quad (3)$$

通过式(1)-式(3), 定义所有用户高评分子空间的并集为高评分子空间, 即 $T^H = T_1^H \cup \dots \cup T_n^H$. 所有用户中评分子空间的并集为中评分子空间, 即 $T^M = T_1^M \cup \dots \cup T_n^M$. 同样地, 所有用户低评分子空间的并集为低评分子空间, 即 $T^L = T_1^L \cup \dots \cup T_n^L$. 在这里, 评分子空间 T^H 、 T^M 和 T^L 都是物品集合 T 的子集, $T = T^H \cup T^M \cup T^L$.

在各评分子空间中, 可以计算用户 u_u 和 u_v 之间的评分支持度, 即:

$$S^*(u_u, u_v) = \frac{|T_u^* \cap T_v^*|}{|T^*|}. \quad (4)$$

式中, 上标 * 在 $\{H, M, L\}$ 中取值, 分别代表不同的子空间. 合并不同评分子空间中的支持度, 并确定用户 u_u 和 u_v 之间的隐式相似度 t_{implicit} 为:

$$t_{\text{implicit}}(u_u, u_v) = S^H(u_u, u_v) + S^M(u_u, u_v) + S^L(u_u, u_v). \quad (5)$$

2.2 显式相似度

从社交关系数据中, 我们可建立一个社交信任关系矩阵 \mathbf{E} . 若用户 u_u 对用户 u_v 显式声明过信任关系, 表示用户 u_u 信任用户 u_v . 反之, 则表示用户 u_u 还不信任用户 u_v . 用户 u_u 和 u_v 之间的信任关系可以表示为:

$$\mathbf{E}(u_u, u_v) = \begin{cases} 1, & u_u \text{ 信任 } u_v, \\ 0, & \text{其他.} \end{cases} \quad (6)$$

注意, 由于用户之间的信任关系具备不对称性, 因而有 $\mathbf{E}(u_u, u_v) \neq \mathbf{E}(u_v, u_u)$.

用户之间的显式相似度与信任传播路径的长度相关. 用户之间信任传播路径越短表示显式相似度越高, 反之显式相似性越低. 简单来说, 用户之间的相似性随着信任传播路径长度的增加而逐渐减少. 因而, 用户之间的显式相似性 t_{explicit} 可以被度量为:

$$t_{\text{explicit}}(u_u, u_v) = \frac{d_{\max} - d_{u,v} + 1}{d_{\max}}. \quad (7)$$

式中, $d_{u,v}$ 表示用户 u_u 到用户 u_v 的最短传播路径的长度, d_{\max} 表示用户之间最大允许传播路径的长度, 其定义为^[25]:

$$d_{\max} = \left\lceil \frac{\ln(n)}{\ln(h)} \right\rceil. \quad (8)$$

式中, n 表示用户信任关系图中用户节点的总数, h 表示用户之间出度和入度的平均值. 根据用户信任关系图, d_{\max} 近似等于用户之间的平均路径长度.

2.3 相似性用户搜索

在相似用户搜索阶段, URSTM 综合考虑用户社交关系数据和评分数据中的信息, 通过将显式相似度与隐式相似度进行融合, 更好地表征了用户之间的相似性. 在 URSTM 中, 用户 u_u 与用户 u_v 的相似度 $\text{Sim}(u_u, u_v)$ 计算公式为:

$$\text{Sim}(u_u, u_v) = \lambda t_{\text{explicit}}(u_u, u_v) + (1 - \lambda) t_{\text{implicit}}(u_u, u_v). \quad (9)$$

式中, 权重 $\lambda \in (0, 1)$.

在此基础上,我们结合 MNNP 中提供的多阶相似用户搜索方式来生成目标用户的相似用户集. 具体的搜索方式,这里就不再叙述,详情请参考文献[21].

2.4 迭代评分预测方法

URSTM 在多阶近邻的相似性用户搜索方式的基础上设计了相应的迭代评分预测方案. 随着迭代次数的增加,评分矩阵逐渐得到回溯和修改,重复这一过程直至算法收敛.

假设 \mathbf{R}^0 是原始评分矩阵, \mathbf{R}^k 是第 k 次迭代得到的评分预测矩阵. \mathbf{R}^k 中的评分值可以表示为:

$$r_{uj}^k = \begin{cases} \hat{r}_{uj}^k, & \text{若 } r_{uj}^0 = 0 \wedge N_u^K \neq \emptyset, \\ r_{uj}^{k-1}, & \text{若 } r_{uj}^0 = 0 \wedge N_u^K = \emptyset, \\ r_{uj}^0, & \text{其他.} \end{cases} \quad (10)$$

式中, k 为当前迭代次数或近邻阶数, N_u^k 是用户 u_u 的第 k 阶近邻集, \hat{r}_{uj}^k 表示第 k 次迭代时用户 u_u 对物品 t_j 的预测评分值. 若用户对物品没有产生过评分行为,即 $r_{uj}^0 = 0$ 时,则根据如下公式预测用户 u_u 对物品 t_j 的评分值,具体表示为:

$$\hat{r}_{uj}^k = \bar{r}_u^{k-1} + \frac{\sum_{u_v \in N_u^k} \text{Sim}(u_u, u_v) (r_{uj}^{k-1} - \bar{r}_v^{k-1})}{\sum_{u_v \in N_u^k} |\text{Sim}(u_u, u_v)|}. \quad (11)$$

式中, \bar{r}_u^{k-1} 表示第 $k-1$ 次迭代后用户 u_u 的评分均值.

2.5 算法描述

基于上述讨论,我们提出了 URSTM,其框架结构如图 1 所示. 在图 1 中,URSTM 算法根据评分矩阵构造用户的高评分、中评分以及低评分子空间,利用不同评分子空间中的信息计算用户之间的隐式相似度. 同时根据社交信任关系矩阵得到用户之间的显式相似度. 然后基于用户之间显式和隐式相似度设计一种新的相似性度量来衡量用户之间的相似度,并结合多阶近邻的相似用户搜索方式选择相似用户集,进而迭代预测用户对物品的评分.

具体的算法步骤如算法 1 所示. URSTM 算法的时间复杂度为 $O(n^2m)$.

算法 1 基于评分子空间和信任机制的协同过滤推荐算法

输入:原始评分矩阵 \mathbf{R} 、社交关系矩阵 \mathbf{E} 、最高近邻阶数 K 、容误差 θ 、用户集 U 、物品集 T .

输出:评分预测矩阵 $\hat{\mathbf{R}}$.

①初始化 $k=1, \mathbf{R}^0=\mathbf{R}$.

②对任意 $\forall u_u \in U$ 通过式(7)计算用户之间的显式相似度.

③构造评分子空间,通过式(5)计算用户之间的隐式相似度.

④基于显式和隐式相似度,采用式(9)衡量用户之间的相似性.

⑤while $k \leq K$ do

⑥ for $\forall u_u \in U$ do

⑦ 搜索用户 u_u 的第 k 阶近邻集 N_u^k ;

⑧ for $t_j \in T$ do

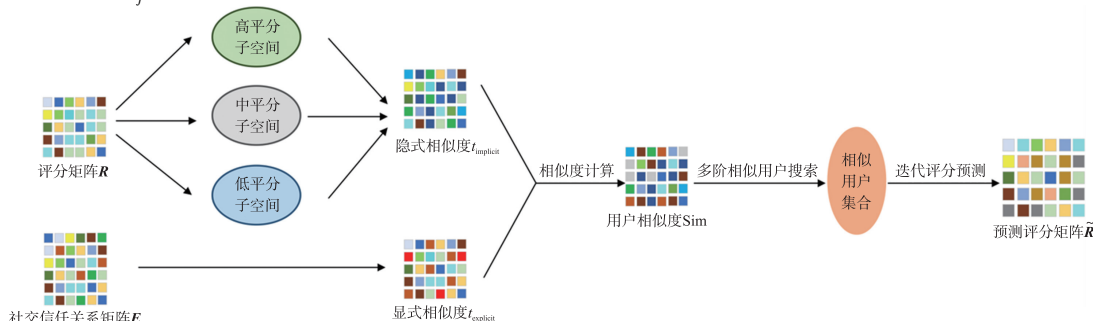


图 1 基于评分子空间和信任机制的协同过滤推荐算法框架

Fig. 1 The frame of the URSTM method for collaborative filtering recommendation

- ⑨ 利用式(11)迭代预测用户对物品的评分;
- ⑩ end for
- ⑪ end for
- ⑫ if $\| \mathbf{R}^k - \mathbf{R}^{(k-1)} \| \leq \theta$ then
- ⑬ $\tilde{\mathbf{R}} = \mathbf{R}^k$;
- ⑭ else
- ⑮ 令 $k = k + 1$;
- ⑯ end if
- ⑰ end while
- ⑱ 返回:评价预测矩阵 $\tilde{\mathbf{R}}$.

3 实验分析

本节通过实验验证所提算法性能. 所有实验均在配备有英特尔 i5 处理器和 16G 内存的个人电脑上完成. 通过采用五折交叉验证方式,使用了 FilmTrust^[26]、Ciao^[27] 和 Epinions^[28] 数据集进行实验. 表 1 给出了这些数据集的统计信息.

3.1 评估指标

在推荐系统中,为了评估算法在实验数据集中的表现,一般采用离线评估指标进行衡量. 常见的离线评估指标大致分为两类,即准确度指标和非准确度指标. 准确度指标衡量的是推荐算法能够以多大程度准确地预测用户对物品的评分偏好. 预测分类准确度和预测评分准确度都属于准确度指标. 预测分类准确度指标包括准确率 (Accuracy)、精确率 (Precision)、召回率 (Recall) 以及 F1-score 等. 预测评分准确度指标包括平均绝对误差 (mean absolute error, MAE)、均方误差 (mean squared error, MSE) 和均方根误差 (root mean squared error, RMSE). 非准确度指标衡量的是推荐列表的多样性、新颖性、覆盖率等.

为了衡量算法的预测分类准确度,我们采用 Recall 进行评估. Recall 计算的是正确预测用户喜欢的物品在用户真正喜欢的物品集中所占的比例. Recall 值越大表示算法性能越好. 在图 2 中,真反例 (true negative, TN) 表示实际上用户不喜欢且预测为不喜欢的物品数,真正例 (true position, TP) 表示实际上用户喜欢且预测为喜欢的物品数,假正例 (false position, FP) 表示实际上用户不喜欢但预测为喜欢的物品数,假反例 (false negative, FN) 表示实际上用户喜欢但预测为不喜欢的物品数. 如图 2 所示,Recall 的计算公式为:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (12)$$

为了衡量算法的预测评分准确度,采用 MAE 和 RMSE 进行评估. MAE 和 RMSE 计算的是预测评分值与用户真实评分值之间的误差,它们的值越小表示算法的预测性能越好. MAE 和 RMSE 的计算公式为

$$\text{MAE} = \frac{1}{|V|} \sum_{(u_u, t_j) \in V} |\hat{r}_{uj} - r_{uj}|. \quad (13)$$

$$\text{RMSE} = \sqrt{\frac{1}{|V|} \sum_{(u_u, t_j) \in V} (\hat{r}_{uj} - r_{uj})^2}. \quad (14)$$

式中,测试数据集 $V = \{(u_u, t_j) \mid u_u \in U, t_j \in T\}$, U 是用户集合, T 是物品集合, \hat{r}_{uj} 和 r_{uj} 分别表示用户 u_u 对物品 t_j 的预测评分值和真实评分值.

另外,我们采用覆盖率 (Coverage) 来衡量推荐物品集与目标用户评分过的物品集的比例,其值越高表示算法的推荐覆盖率越强. Coverage 的计算式为

表 1 数据集信息统计情况
Table 1 Statistic information of data sets

数据集	用户数	物品数	评分数	社交关系数
FilmTrust	1 508	2 071	35 497	1 853
Ciao	7 375	99 746	278 483	111 781
Epinions	49 290	139 738	664 824	478 183

		预测值	
		0	1
真实值	0	TN (Ture Negative)	FP (False Positive)
	1	FN (False Negative)	TP (True Positive)

图 2 混淆矩阵

Fig. 2 Confusion matrix

$$\text{Coverage} = \frac{\sum_{(u_u, v) \in V} |R(u_u) \cap P(u_v)|}{\sum_{(u_u, v) \in V} |P(u_v)|}.$$

(15)

式中, $R(u_u)$ 是为目标用户 u_u 产生的推荐列表, $P(u_v)$ 是在测试集中目标用户 u_u 评分过的物品集合.

3.2 参数分析

在 URSTM 算法中, 主要包括权重 λ 、近邻阶数 K' 和相似用户数 ℓ 共 3 个参数. 根据文献[21], 将近邻阶数 K' 设置为 2. 另外, 为了研究参数 λ 和 ℓ 对算法性能的影响, 采用五折交叉验证的方式在 FilmTrust 数据集上进行实验.

在分析参数 ℓ 对算法性能的影响时, 设置权重 $\lambda = 0.1$. 当 ℓ 取不同值时, URSTM 算法的平均性能指标变化趋势如图 3 所示. 图 3 是一个双纵坐标图, 左边纵坐标表示 MAE、RMSE、Recall 和 Coverage 共 4 个性能指标, 右边纵坐标表示算法的运行时间 t . 观察图 3 可以发现, 当相似用户数 $\ell = 30$ 时, MAE 和 RMSE 曲线出现最小拐点. 在拐点之后, 其变化幅度都不太大. 另外, 虽然 Recall 和 Coverage 随着相似用户数 ℓ 的增加而逐渐上升, 但是可以看到当 $\ell = 30$ 时, Recall 和 Coverage 曲线上升的趋势比较明显. 除此之外, 算法的运行时间与相似用户数 ℓ 成线性关系. 综合考虑算法性能和时间成本问题, 在实验中, 我们将 ℓ 设置为 30.

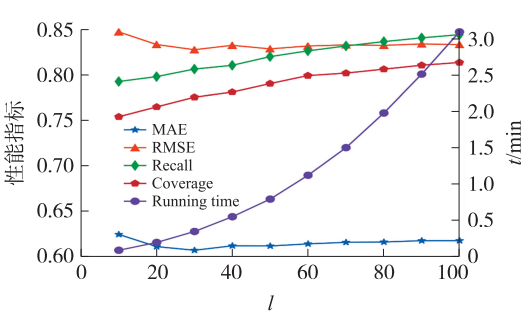


图 3 URSTM 在不同 ℓ 值下的性能

Fig. 3 The performance of URSTM under different ℓ

在分析参数 λ 对算法性能的影响时, 先设置相似用户数 $\ell = 30$. 令 λ 的取值范围为 $\{0.05, 0.1, \dots, 0.95\}$. 图 4 展示了参数 λ 取不同值时的本文方法预测性能变化趋势. 可见, 当 $\lambda = 0.1$ 时, MAE 和 RMSE 曲线出现最小拐点. 当 $\lambda > 0.2$ 时, MAE 曲线趋于稳定. 实验将参数 λ 设置为 0.1.

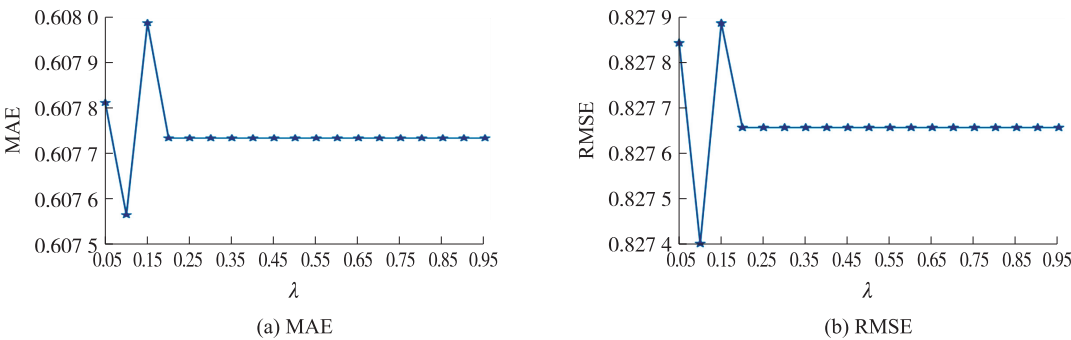


图 4 URSTM 在不同 λ 值下的性能

Fig. 4 The performance of URSTM under different λ

3.3 实验结果对比

为了验证 URSTM 算法的性能, 在 3 个数据集上将之与其他相关算法方法进行对比, 这些算法包括: UTCF^[29]、TRP2M^[28]、TrustASVD++^[30]、PCC^[12]、TARS^[31]、RRS^[24] 和 MNNP^[21]. 上述 8 种算法在 FilmTrust、Ciao 和 Epinions 数据集上取得的结果分别如表 2—表 4 所示, 其中最优结果用粗体表示, 次优结果用下划线表示.

在 FilmTrust 数据集中, URSTM 算法获得最佳 MAE、RMSE 和 Recall. 从表 2 中可以看出: URSTM 算法的 MAE、RMSE 和 Recall 值比次优算法 RRS 分别提高了 0.98%、1.73% 和 1.12%. 虽然 URSTM 没有获得最佳 Coverage, 但从综合排名来看, URSTM 算法排名第一.

表 2 FilmTrust 数据集上的实验结果				
Table 2 Results on the FilmTrust data set				
方法	MAE	RMSE	Recall	Coverage
PCC	0.786 3	1.108 7	0.596 9	0.539 0
TARS	0.701 7	0.942 6	0.278 6	0.261 8
UTCF	0.715 3	0.933 5	0.510 6	0.464 9
TRP2M	0.657 9	0.968 7	0.670 9	0.762 3
TrustASVD++	0.607 4	0.832 1	0.807 2	<u>0.781 2</u>
MNNP	0.661 6	0.960 1	0.726 7	0.688 6
RRS	<u>0.606 3</u>	<u>0.830 0</u>	<u>0.810 0</u>	0.791 0
URSTM	0.600 3	0.815 6	0.819 1	0.774 0

在 Ciao 数据集上,URSTM 方法的预测性能 MAE 和 RMSE 在对比方法中是最优的,见表 3. 另外,URSTM 算法取得次优 Recall 和 Coverage. 与 TrustASVD++方法相比,URSTM 算法的 MAE 和 RMSE 比其分别提高了 0.78%和 0.66%. 就总体而言,URSTM 算法性能相对较好.

在 Epinions 数据集上,URSTM 方法取得最优的 RMSE 和 Recall 结果,次优的 MAE 和 Coverage,见表 4. URSTM 算法的 RMSE 和 Recall 值比 TrustASVD++方法分别提高了 1.37%和 1.11%.

表 3 Ciao 数据集上的实验结果

Table 3 Results on the Ciao data set

方法	MAE	RMSE	Recall	Coverage
PCC	0.851 6	1.184 5	0.601 3	0.540 6
TARS	0.797 2	1.138 2	0.450 4	0.402 1
UTCF	0.795 3	1.173 5	0.644 3	0.621 0
TRP2M	0.785 1	1.051 4	0.701 4	0.699 8
TrustASVD++	<u>0.657 4</u>	<u>0.983 2</u>	0.765 3	0.621 8
MNNP	0.718 9	0.985 4	0.636 7	0.598 3
RRS	0.689 1	1.090 9	0.729 1	0.655 9
URSTM	0.652 3	0.976 7	<u>0.739 8</u>	<u>0.688 2</u>

表 4 Epinions 数据集上的实验结果

Table 4 Results on the Epinions data set

方法	MAE	RMSE	Recall	Coverage
PCC	0.870 3	1.165 1	0.593 1	0.536 7
TARS	0.888 6	1.248 6	0.325 6	0.319 3
UTCF	0.793 3	1.056 4	0.630 8	0.618 3
TRP2M	0.805 5	1.053 7	0.618 6	0.762 6
TrustASVD++	0.689 6	<u>0.985 2</u>	<u>0.713 4</u>	0.682 3
MNNP	0.750 3	1.044 9	0.645 6	0.639 2
RRS	0.727 9	0.989 1	0.714 9	0.653 0
URSTM	<u>0.715 3</u>	0.971 5	0.724 5	<u>0.685 9</u>

另外,我们根据 4 个性能指标结果,采用显著水平为 0.5 的双尾 t -test 检验来分析 URSTM 方法与其他方法之间是否存在显著差异,进而说明 URSTM 方法的有效性. 表 5 列出了其他方法与 URSTM 方法之间的 t -test 检验结果,其中赢-输-平(W-L-T)对 t -test 结果进行了总结.

表 5 URSTM 与其他方法性能指标的 t -test 检验结果比较Table 5 Comparison of t -test results of performance indicators of URSTM and other methods

Dataset	Performance	PCC	TARS	UTCF	TRP2M	TrustASVD++	MNNP	RRS
FilmTrust	MAE	9.26×10^{-8}	1.70×10^{-5}	5.46×10^{-7}	0.007 3	0.581 5	0.000 3	0.538 4
	RMSE	1.24×10^{-8}	9.98×10^{-6}	4.25×10^{-6}	0.002	0.097 4	3.95×10^{-6}	0.305 7
	Recall	3.09×10^{-10}	3.39×10^{-14}	1.52×10^{-11}	8.70×10^{-6}	0.199	4.98×10^{-8}	0.004 9
	Coverage	6.21×10^{-12}	8.60×10^{-15}	5.43×10^{-13}	0.288 6	0.473 7	7.26×10^{-10}	5.25×10^{-7}
Ciao	MAE	9.12×10^{-8}	0.001 6	3.84×10^{-7}	1.51×10^{-5}	0.810 6	0.009 7	0.002 6
	RMSE	0.001 9	0.000 9	7.44×10^{-6}	0.049 4	0.796 7	0.787 3	0.147 3
	Recall	6.75×10^{-5}	1.83×10^{-8}	7.66×10^{-5}	0.025	0.130 6	0.000 1	0.509 7
	Coverage	6.14×10^{-5}	2.87×10^{-7}	0.000 6	0.434 2	0.005 2	0.000 7	0.016 8
Epinions	MAE	1.12×10^{-8}	5.17×10^{-9}	1.25×10^{-7}	1.30×10^{-8}	9.58×10^{-5}	2.01×10^{-5}	0.044 2
	RMSE	1.61×10^{-11}	4.34×10^{-10}	6.36×10^{-8}	6.18×10^{-11}	1.05×10^{-7}	0.020 5	3.14×10^{-8}
	Recall	6.25×10^{-9}	4.92×10^{-16}	2.34×10^{-7}	1.55×10^{-10}	0.007	3.49×10^{-9}	0.015 3
	Coverage	6.70×10^{-8}	1.44×10^{-16}	8.46×10^{-8}	3.04×10^{-8}	0.012 4	6.80×10^{-10}	0.012 1
W-L-T		0-12-0	0-12-0	0-12-0	1-10-1	2-6-4	0-11-1	1-8-3

从表 5 中可以看到只有 RRS、TRP2M、TrustASVD++方法在某个指标上胜过 URSTM 以外,URSTM 在 4 个性能指标上明显优于其他对比方法. 总的来说,URSTM 算法显著优于其他方法的情况是占大多数的,因此 URSTM 算法在性能方面是可取的. 这说明了引入用户显示声明关系能在一定程度上提高算法预测的精准性,所构造的相似性度量方法的确缓解了评分行为的多样性问题.

4 结论

引入信任机制,提出了基于评分子空间和信任机制的协同过滤推荐算法 URSTM. 该算法利用用户的显式信任度和隐式信任度来衡量用户之间的相似性,并将之引入到多阶近邻的相似用户搜索方法和迭代评分预测方案中. 在 3 个公开的推荐系统数据集 FilmTrust、Ciao 以及 Epinions 上,本文进行了大量的实验. 实验结果表明 URSTM 算法在这 3 个数据集上取得最佳的 RMSE 预测性能,在两个数据集中取得最优 MAE 和 Recall 结果,在一个数据集上取得次优 MAE 和 Recall 结果. 此外,通过显著性检验验证了 URSTM 算法明显优于其他对比方法. 总之,URSTM 算法通过社交关系数据对稀疏的评分数据进行补充,可以取得

良好的推荐性能. 基于用户之间的关系对协同过滤算法展开了研究,实验结果表明,本文工作具有较好的预测性能.

[参考文献] (References)

- [1] GOLDBERG D, NICHOLS D, OKI B M, et al. Using collaborative filtering to weave an information tapestry[J]. *Communications of the ACM*, 1992, 35(12): 61–70.
- [2] RESNICK P, VARIAN H R. Recommender systems[J]. *Communications of the ACM*, 1997, 40(3): 56–58.
- [3] LE Q H, VU S L, LE T X. A state-of-the-art survey on context-aware recommender systems and applications[J]. *International Journal of Knowledge and Systems Science*, 2021, 12(3): 1–20.
- [4] HE C, PARRA D, VERBERT K. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities[J]. *Expert Systems with Applications*, 2016, 56: 9–27.
- [5] CEN Y K, ZHANG J W, ZOU X, et al. Controllable multi-interest framework for recommendation[C]//The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York, NY, USA, 2020.
- [6] TOLLON F. Designed to seduce: epistemically retrograde ideation and youtube’s recommender system[J]. *International Journal of Technoethics*, 2021, 12(2): 60–71.
- [7] MEDEL D, GONZÁLEZ-GONZÁLEZ C S, ACIAR S V. Social relations and methods in recommender systems: A systematic review[J]. *International Journal of Interactive Multimedia and Artificial Intelligence*, 2022, 7(4): 7.
- [8] DU C, GAO Z, YUAN S, et al. Exploration in online advertising systems with deep uncertainty-aware learning[C]//Proceedings of the 27th ACM SIGKDD Conference Discovery and Data Mining. New York, NY, USA, 2021.
- [9] ZANGERLE E, BAUER C. Evaluating recommender systems: survey and framework[J]. *ACM Computing Surveys*, 2023, 55(8): 1–38.
- [10] HERLOCKER J L, KONSTAN J A, BORCHERS A, et al. An algorithmic framework for performing collaborative filtering[J]. *ACM SIGIR Forum*, 2017, 51(2): 227–234.
- [11] AFSAR M M, CRUMP T, FAR B H. Reinforcement learning based recommender systems: a survey[J]. *ACM Computing Surveys*, 2023, 55(7): 1–38.
- [12] JIN R M, LI D, GAO J, et al. Towards a better understanding of linear models for recommendation[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. New York, NY, USA; Association for Computing Machinery, 2021.
- [13] ZHANG Z, ZHANG Y, REN Y. Employing neighborhood reduction for alleviating sparsity and cold start problems in user-based collaborative filtering[J]. *Information Retrieval Journal*, 2020, 23: 449–72.
- [14] WANG R, JIANG Y, LOU J. Attention-based dynamic user preference modeling and nonlinear feature interaction learning for collaborative filtering recommendation[J]. *Applied Soft Computing*, 2021, 110: 107652.
- [15] RAMEZANI M, TAB F A, ABDOLLAHOPOURI A, et al. A new generalized collaborative filtering approach on sparse data by extracting high confidence relations between users[J]. *Information Sciences*, 2021, 570: 323–41.
- [16] PAPADAKIS H, PAPAGRIGORIOU A, PANAGIOTAKIS C, et al. Collaborative filtering recommender systems taxonomy[J]. *Knowledge and Information Systems*, 2022, 64(1): 35–74.
- [17] SUN H, PENG Y, CHEN J, et al. A new similarity measure based on adjusted euclidean distance for memory-based collaborative filtering[J]. *J Softw*, 2011, 6(6): 993–1000.
- [18] KHOJAMLI H, RAZMARA J. Survey of similarity functions on neighborhood-based collaborative filtering[J]. *Expert Systems with Applications*, 2021, 185: 115482.
- [19] BAG S, KUMAR S K, TIWARI M K. An efficient recommendation generation using relevant Jaccard similarity[J]. *Information Sciences*, 2019, 483: 53–64.
- [20] MARGARIS D, VASSILAKIS C. Improving collaborative filtering’s rating prediction coverage in sparse datasets by exploiting the ‘friend of a friend’ concept[J]. *International Journal of Big Data Intelligence*, 2020, 7(1): 47–57.
- [21] SUN X, ZHANG L. Multi-order nearest neighbor prediction for recommendation systems[J]. *Digital Signal Processing*, 2022, 127: 103540.
- [22] RAMEZANI M, MORADI P, AKHLAGHIAN F. A pattern mining approach to enhance the accuracy of collaborative filtering in sparse data domains[J]. *Physica a Statistical Mechanics & Its Applications*, 2014, 408: 72–84.

- [23] KOOHI H, KIANI K. A new method to find neighbor users that improves the performance of collaborative filtering[J]. Expert Systems with Applications, 2017, 83: 30–39.
- [24] 孙晓寒, 张莉. 基于评分区域子空间的协同过滤推荐算法[J]. 计算机科学, 2022, 49(7): 50–56.
- [25] MORADI P, AHMADIAN S. A reliability-based recommendation method to improve trust-aware recommender systems[J]. Expert Systems with Applications, 2015, 42(21): 7386–98.
- [26] GUO G, ZHANG J, YORKE-SMITH N. A Novel Evidence-Based Bayesian Similarity Measure for Recommender Systems[J]. ACM Transactions on the Web, 2016, 10(2): 1–30.
- [27] GUO G, ZHANG J, THALMANN D, et al. ETAF: An extended trust antecedents framework for trust prediction[C]//2014 IEEE International Conference on Advances in Social Networks Analysis and Mining. Beijing, China: IEEE, 2014.
- [28] MASSA P, SOUREN K, SALVETTI M, et al. Trustlet, Open Research on Trust Metrics[J]. Scalable Computing Practice Experience, 2008, 9(4): 31–44.
- [29] SALEEM F, ILTAF N, AFZAL H, et al. Using trust in collaborative filtering for recommendations[C]//IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises. Kyoto, Japan, 2007.
- [30] RAHIM A, DURRANI M Y, GILLANI S A, et al. An efficient recommender system algorithm using trust data[J]. The Journal of Supercomputing, 2022, 78(3): 3184–204.
- [31] YUAN W W, GUAN D H, LEE Y K, et al. The small-world trust network[J]. Applied Intelligence, 2011, 35(3): 399–410.

[责任编辑:陈 庆]