

# 一种基于信任关系隐含相似度的社会化推荐算法

潘一腾 何发智 于海平

(武汉大学软件工程国家重点实验室 武汉 430072)

(武汉大学计算机学院 武汉 430072)

**摘 要** 推荐算法已经成为许多电子商务网站必不可少的组成部分. 基于用户历史评价数据的协同过滤推荐算法通常面临着数据稀疏的问题, 即用户评分过于稀疏导致推荐质量下降. 为了解决这一问题, 结合辅助数据成为一种必然的趋势. 因此, 随着社交媒体的发展, 基于信任关系的社会化推荐算法被证明为一种有效的解决方法. 这些算法利用社交网络信息对用户偏好进行建模, 并进行推荐. 然而, 目前大部分算法直接利用社交网络的二值信任关系来提高推荐质量, 从而没有考虑用户对每个好友信任强度的差异. 为了解决这一问题, 该文提出了一种新的基于信任关系隐含相似度的度量方法, 并与协同推荐算法相结合, 获得更高的推荐质量. 与之前的方法不同, 在考虑评分相似度的基础上, 该文专注于研究利用社交信息来估计信任强度并提出了信任关系隐含相似度. 首先, 该文考虑了用户间的间接影响, 即通过分解社交矩阵得到隐含间接影响的社交偏好, 并基于此得到了信任关系隐含相似度; 其次, 鉴于用户在作为信任者和被信任者时的偏好并不相同, 该文提出的信任关系隐含相似度分别考虑了这两种情况; 进一步, 考虑到评分和社交数据都非常稀疏, 文章同时考虑了评分相似和信任相似对每组用户间信任强度的影响, 得到一个更加精确的社会化推荐模型; 最后, 不同于直接计算信任强度的算法, 该文基于评分和社交数据, 提出了一种自适应相似度计算的模型. 该文在 Epinions 和 Ciao 数据集上进行了丰富的实验, 并与多种前沿的算法进行了性能对比. 文中同时采用基于误差的指标 (MAE 和 RMSE) 和排序类指标 (精度、召回率和 NDCG) 对算法性能的性能进行度量, 结果表明该文算法对于评分预测和 Top-N 项目推荐任务都能得到鲁棒的表现. 文中还展示了对于评分和信任数据稀疏用户的性能表现, 结果仍优于以往的算法. 概括来说, 文中算法充分挖掘了用户在评分和社交数据中的隐含信息, 从而有效提高了社会化推荐算法的精度.

**关键词** 推荐系统; 协同过滤算法; 社会推荐算法; 潜在矩阵分解; 信任关系传播; 隐含相似度

**中图法分类号** TP18 **DOI 号** 10.11897/SP.J.1016.2018.00065

## Social Recommendation Algorithm Using Implicit Similarity in Trust

PAN Yi-Teng HE Fa-Zhi YU Hai-Ping

(State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072)

(School of Computer Science, Wuhan University, Wuhan 430072)

**Abstract** Recommendation system is becoming one of the most important components in many e-commerce websites and web applications. In the past decades, collaborative filtering (CF) is the most success approach to handle the recommendation task, which leverage users' historical rating data to model users and items preference. As an early approach, the CF algorithm is seriously affected by the data spare problem, which means that most users rate only a few items. This problem distinctly degenerates the performance of CF-based methods. To address this problem, introducing auxiliary information for CF algorithm becomes a necessary trend to improve recommendation accuracy. Therefore, with the development of social media, social recommendation algorithms have been proposed and presented as a potential way to solve this problem. These

收稿日期:2016-02-26; 在线出版日期:2016-11-25. 本课题得到国家自然科学基金(61472289)、湖北省自然科学基金(2015CFB254)资助. 潘一腾, 男, 1988 年生, 博士研究生, 主要研究方向为推荐系统、深度学习. E-mail: panyiteng@whu.edu.cn. 何发智 (通信作者), 男, 1968 年生, 博士, 教授, 中国计算机学会 (CCF) 会员, 主要研究领域为 CAD 图形图像、协同计算、数据挖掘. E-mail: fzhe@whu.edu.cn. 于海平, 女, 1978 年生, 博士研究生, 主要研究方向为图像处理、图像分割.

methods utilizing users' social information to help modeling user preference for recommendation. However, most of them simply make use of the social network information with binary format directly, and thus ignore the fact that the trust degrees between each pair of users are full of variety. In this paper, we propose a new method to measure trust degrees, i. e., how much degrees will users trust in other users, and to integrate these degrees for social recommendation. Advancing previous works, in addition of rating similarity, we focus on the social information to estimate trust degrees and propose the implicit similarity in trust. Specially, we take the indirect social influence of users into account to estimate trust degrees between each pair of users. We obtain the social preferences which contains indirect influence by factorizing the social matrix. Based on these preferences, we propose the implicit similarity in trust for users, which exactly reflect social relations. In addition, based on the intuition that a user plays different preferences with role of truster and trustee, our proposed implicit similarity in trust is consist of two parts: users as truster and as trustee. Moreover, on account of the fact that both rating and social data are very spare, we take both rating and social data into account to enhance the evaluation of trust degrees. In this way, after synthesizing both rating similarity and implicit similarity in trust for each pair of users, we establish an accurate social recommendation model. Furthermore, instead of directly estimating the trust degrees, we propose an adaptive similarity learning model based on both rating similarity and implicit similarity in trust to evaluate trust degrees for recommender system. We conduct several experiments on both Epinions and Ciao datasets, and compared our model with several state-of-the-art algorithms. The experiments demonstrate that the proposed approach fully exploits implicit information in rating data and social network, and efficiently improves the performance of social recommendation. Specially, the results are evaluated in terms of two error based metrics (*MAE* and *RMSE*) and rank-sensitive metrics (*precision*, *recall* and *NDCG*), and indicate that our approach is robust in both tasks of rating prediction and top-*n* recommendation tasks. We also demonstrate the results in case of sparse rating data or spare social data, which outperforms previous works. In summary, our approach, fully exploiting implicit information in rating and social data, can efficiently improve recommendation accuracy.

**Keywords** recommendation system; collaborative filtering; social recommendation; latent matrix factorization; trust propagation; implicit similarity

## 1 引 言

在日常生活中,随着网络信息的不断发展,信息过载问题变得越来越严重.如何从海量的数据中获得有效的信息,对于普通用户来说是一个巨大的挑战.推荐算法是解决这一问题的重要手段之一,通过对用户的历史行为建模,主动向用户提供满足用户潜在偏好的产品.对于用户来说,推荐系统能够帮助他们在海量的信息中快速找到满意的信息;对于商家来说,推荐系统不仅能帮助决定向特定用户推销那些产品,同时能够通过更满意的服务增加用户的忠诚度.推荐系统广泛应用于大量的电子商务网站中,比如淘宝、京东以及亚马逊等,已经成为网络

应用的重要组成部分之一<sup>[1-5]</sup>.

尽管近年来推荐算法在各种应用中都取得了巨大的成功<sup>[6-11]</sup>,然而数据稀疏问题仍然是影响算法性能的重要瓶颈之一.推荐算法通常基于用户的历史数据对用户偏好进行建模,然后根据用户的偏好从海量的数据中找到合适的产品推荐给用户.而当这些数据并不足以反映用户或产品的属性时,推荐性能就会受到影响.具体来说,在一般情况下,大多数用户由于精力有限或个人隐私等原因,并不会对所有的产品进行评价,相反,只会选择少量的产品进行评分,从而所获得的评分矩阵只有少量是已知的,而其它大量的未知评分则是推荐算法的预测目标.此时,这少量的评分数据只能反映用户非常少的有用信息,而基于此所建立的模型也难以准确把握用

户真正的喜好,进而导致推荐精度下降,这一问题被称为数据稀疏问题。

围绕着数据稀疏的问题,许多学者提出了有效的解决方法<sup>[7,12-17]</sup>。包括引入辅助数据、综合跨域信息或挖掘数据中的更多的隐含规律,例如,利用项目特征辅助提高推荐算法处理数据稀疏问题的能力<sup>[6-7,12,18]</sup>,考虑将不同信息源的数据进行结合得到跨域推荐结果<sup>[19-20]</sup>,挖掘数据中的隐含反馈信息<sup>[17,21]</sup>,充分利用用户提供的信息<sup>[22]</sup>或者结合社交关系信息<sup>[23-25]</sup>等。

其中,结合社交信息的推荐算法是解决数据稀疏问题的有效手段之一。一方面,在生活中,人们在做出决定之前通常会向他的好友咨询,并受到好友的影响;另一方面,人们总是倾向于与他爱好相似的人结为好友关系<sup>[26-27]</sup>。因此,社交网络数据在一定程度上揭示了用户的偏好相似性,这对提高推荐算法的质量提供了新的机会。特别是在处理数据稀疏问题上,社会化推荐算法能显著提高稀疏用户的推荐性能,从而提升用户体验和忠诚度<sup>[3]</sup>。近年来,社会化推荐算法吸引了越来越多研究人员的关注,成为推荐算法的热门研究领域之一<sup>[6,10,13,23-24,28-33]</sup>。

基于用户与好友偏好相似的假设,为了更有效的对用户之间的社交关系进行建模,充分挖掘社交好友对用户偏好的影响,很多工作对此做出了有益的探索。例如, Ma 等人<sup>[23]</sup>采用概率矩阵分解的方法对社交信息建模, Jamali 等人<sup>[25]</sup>将信任传播引入到社会化推荐算法当中, Yang 等人<sup>[34]</sup>则对用户作为信任者和被信任者的情况进行了建模。

考虑到用户对每个好友的信任强度并不相同,如何估计用户之间的信任度,成为影响社会化推荐算法进一步提高的重要问题之一。该相似度可以从评分或信任数据的角度进行度量,即基于具有相似的评分行为或信任关系的用户之间具有相似的兴趣偏好的假设,可以用评分或信任相似度量用户之间的信任度。例如, Ma 等人<sup>[35]</sup>利用评分相似度函数计算用户之间的信任强度, Yao 等人<sup>[36]</sup>从信任关系的角度对信任强度进行估计, Wang 等人<sup>[37]</sup>则考虑了用户的评分和信任相似度并进行加权。

这些对信任强度进行探索的工作,很好的挖掘了社交关系中的隐含信息,然而这一问题并没有完全解决,不能很好的处理评分和信任数据稀疏的情况。具体来说,文中主要针对这些工作的以下两个不足之处进行研究。

首先,这些工作对信任相似的估计都是基于共

同好友集合的相似函数进行计算,这在信任数据稀疏的情况下,不能得到有效的结果。例如,在只有零个或很少共同好友时,得到零或很小的相似度,而这并不能反映用户之间的真实关联情况。

其次,已有的工作表明评分相似和信任相似都能在一定程度上揭示用户之间的信任强度,然而,大部分工作<sup>[35-36]</sup>都仅考虑了其中一方面的影响。虽然 Wang 等人<sup>[37]</sup>从评分和信任相似的角度进行了建模,然而,没有对每组用户间的信任强度综合两方面的信息。

为此,文中算法在社会化推荐算法 SocialMF 的基础上,提出一种基于信任关系隐含相似度的推荐算法。本文的贡献点主要包括:(1)提出一种信任关系隐含相似的度量方法,不仅包含用户间的直接关联信息,而且隐含了用户间的间接社交关联,进而提高了推荐精度;(2)提出分别计算用户作为信任者和被信任者时的信任关系隐含相似度,从而更为全面的利用了信任关系中的隐含信息;(3)综合考虑了评分相似和信任关系隐含相似对每组用户之间信任度的影响,得到了更精确的信任度量和推荐模型。

## 2 相关工作

传统的推荐算法主要专注于充分利用用户的历史评价数据进行推荐,这类算法称为协同过滤推荐算法。然而,这些算法始终面临着用户历史数据非常稀疏的问题,导致推荐质量下降。

为了解决这一问题,在算法中引入辅助数据或挖掘数据中更多规律是有效的方法。例如, Wang 等人<sup>[7]</sup>利用 LDA 算法提取文本特征提高推荐算法处理冷启动和稀疏用户的能力, Koren<sup>[17]</sup>通过挖掘数据中的隐含反馈从而提高推荐精度,黄创光等人<sup>[16]</sup>则动态结合用户和产品的相似性从而提高算法处理稀疏数据的能力,以及其他许多学者提出一系列结合社交关系的推荐算法解决冷启动和稀疏问题<sup>[15,23-25,29,35,38]</sup>。

其中结合了社交关系的协同过滤算法称为社会化推荐算法。得益于社交媒体应用的发展,用户之间在网络上的交流越来越频繁,逐渐成为人们日常生活的一部分,使得社交关系数据的获取越来越方便快捷。因此,基于社交的推荐也逐渐成为推荐算法不可或缺的一部分。社会化推荐算法研究重点在于如何更有效的将社交信息与评价信息有机融合。

大多数社会化推荐算法主要专注于解决两个问

题:如何将社交信息有效的集成到推荐算法中?如何估计用户之间的信任度以提高算法精度?

针对第一个问题,许多工作尝试从不同的角度对社交信息进行建模.例如,Jamali 等人<sup>[38]</sup>提出一种在评分网络和社交网络中随机游走的 TrustWalker 算法.该算法在不需要预先训练的情况下,通过在历史数据中游走得到结果,但是在数据量比较大的时候,查询时间往往过大.Ma 等人<sup>[29]</sup>提出 SoRec 算法,假设用户在评价网络和社交网络中共享相同的偏好向量,并采用概率矩阵分解的方式进行约束,最终获得考虑用户好友因素的推荐结构.相比于之前的工作,他们的算法获得更高的精度.随后他们又提出将评价信息和社交网络进行集成的推荐算法 RSTE<sup>[24]</sup>,该算法假设用户的评价由个人偏好和好友影响共同决定,并存在一个平衡.Jamali 等人<sup>[25]</sup>将信任传播的方法引入到推荐算法中(SocialMF),通过约束用户与其好友的平均偏好相似来传播信任关系,从而得到更准确的结果.

以上这些工作假设每个好友对用户的影响是相同的.而现实中,用户对每个好友的信任度并不相同<sup>[39]</sup>,甚至有可能完全不信任(即有可能仅仅出于礼貌而添加好友关系).实际上,用户间的信任关系受多种因素的影响,有的是由于爱好相似,有的是由于相同的社交圈子,有的则仅仅是出于礼貌.简单的二值信任网络并不能反映出用户之间的影响力大小,也不能充分挖掘社交网络中隐含的用户偏好信息,由此引出了第 2 个问题的研究,即如何估计用户之间的信任度?

对此, Ma 等人<sup>[35]</sup>考虑了用户评分相似度对信任强度的影响,将社交网络作为正则约束项来学习用户偏好,得到更加精确的结果.郭磊等人<sup>[40]</sup>利用对评分矩阵进行概率矩阵分解得到的潜在向量计算用户与好友之间的相似性,提高了算法精度.他们同时提出充分利用对象间的关系提高推荐精度<sup>[41]</sup>.

但是,当用户评分数据稀疏时,根据评分计算得到的相似度并不能准确反映用户的喜好,而基于此得到的用户信任度也不可靠,从而对推荐算法的精度产生不利的影响.

所以,人们继续从社会关系的角度进行挖掘.Yao 等人<sup>[36]</sup>则考虑了用户作为信任者和被信任者时的不同情况,分别计算两种情况的信任相似度,以此约束用户偏好向量的学习,然后对这两种情况下的用户向量进行加权并对评分结果产生影响.不过,

这没有考虑到用户评分偏好和社交偏好并不共享相同的潜在空间,基于此,Wang 等人<sup>[42]</sup>提出通过线性映射的方法建立用户评分偏好和社交偏好的关系,并采用修正的信任强度约束用户之间的关系.为了进一步提高精度,Wang 等人<sup>[37]</sup>则同时从评分和信任数据的角度对用户之间的信任强度进行估计,分别提出了用户的能力度量函数和可信赖度量函数,并通过自适应加权的方式与评分相似函数结合得到估计的用户偏好,并在此基础上构建了一个更为精确推荐模型.

以上这些算法从评分或信任数据的角度对用户之间的信任强度进行了研究,相比于将每个好友对用户的影响视为等同的算法,取得了明显的提高.然而,对这一问题的研究仍然存在一些问题.针对引言中提到的具体问题,我们提出了一种基于信任关系隐含相似度的社会化推荐算法.

首先,我们采用概率矩阵分解的方法获得用户作为信任者和被信任者时的潜在向量,并以此来计算用户分别作为信任者和被信任者时的相似度.与之前的工作<sup>[36-37,42]</sup>相比,我们的信任相似度计算并不依赖于用户之间的共同好友集合或用户好友数量,即使在信任数据比较稀疏或共同好友集合为零的情况也能得到稳定可靠的计算结果.同时,这一方法通过将用户社交行为映射到低维子空间,得到的用户向量不仅包含了用户之间的直接关联,而且隐含了用户之间的间接联系.具体来说,如果两个用户之间没有建立社交关系,而这两个用户的好友集合之间有很多关联,那么他们的用户向量之间也会具有很强的联系.因此,我们得到了一个更为精确的信任度量模型.

其次,我们综合考虑了评分和信任相似对用户间信任强度的影响.大部分已有的工作都仅单独考虑了这两者对信任强度的影响<sup>[35-36]</sup>,而他们的工作表明,这两者对于信任强度的估计都是有益的,因此,有必要综合两者信息从而得到一个更精确的模型.基于此,Wang 等人在<sup>[37]</sup>与我们的工作比较接近,也同时考虑了评分和信任数据中的信息.他们从信任近邻和评分近邻集合的角度对两者进行建模,然而,这两个集合并不是相等的,则对于只存在于其中一个集合的近邻而言,仅利用了其中一方面的信息;同时,他们也没有考虑两个用户都作为信任者或被信任者时的信任关系隐含相似度.与他们的工作<sup>[37]</sup>相比,我们对于每一组用户间的信任估计都综

合了两方面的信息,并且考虑了信任关系隐含相似度,进而得到更为精确的信任度量。

最终,我们得到一个综合考虑评分和信任相似的推荐法模型,不仅提高了算法处理稀疏数据的能力,而且通过挖掘数据中的隐含关联信息,进一步提高了模型的精度。

### 3 社会化推荐算法

本节主要介绍与本文直接相关的两个经典算法.首先对社会化推荐的问题进行形式化定义和描述,然后分别回顾两个已有的相关工作,包括基于概率矩阵分解的推荐算法 PMF<sup>[43]</sup>和基于信任传播的社会化推荐算法 SocialMF<sup>[25]</sup>.

#### 3.1 问题描述

我们假设推荐系统中含有  $N$  个用户组成的集合  $U = \{u_1, \dots, u_n\}$  和  $M$  个产品组成的集合  $I = \{i_1, \dots, i_m\}$  以及用户对产品的评价矩阵  $\mathbf{R} = [\mathbf{R}_{u,i}]_{N \times M}$ . 在评价矩阵中  $\mathbf{R}_{u,i}$  表示用户  $u$  对产品  $i$  的评分. 评分矩阵中通常隐含了用户的潜在偏好信息,是推荐算法采用的主要数据。

在社交网络中,每个用户有  $N_u$  个好友,  $t_{u,v}$  表示用户  $u$  和  $v$  的好友关系,0 表示非直接好友,1 表示直接好友,从而得到用户的信任矩阵  $\mathbf{T} = [\mathbf{T}_{u,v}]_{N \times N}$ . 用户的社交关系通常只能获得二值的数据,然而并非所有的好友对用户的影响都是一样的,因此,本文主要研究如何结合信任关系隐含相似度来提高推荐算法的质量。

社会化推荐算法的基本目标是,在给定评价矩阵  $\mathbf{R}$  和信任关系  $\mathbf{T}$  的情况下,预测用户  $u$  对未评价项目  $i$  的评分情况. 研究表明,结合社交信息的推荐算法能够有效缓解数据稀疏的问题. 目前,如何高效的挖掘社交关系中隐藏的用户偏好信息是社会化推荐算法的研究重点。

#### 3.2 概率矩阵分解

传统的协同过滤推荐算法通常只采用用户历史评价数据对用户建模,进而对用户未来的评价和选择进行预测,包括基于记忆<sup>[44]</sup>和基于模型<sup>[43,45]</sup>两种类型. 其中,基于概率矩阵分解的协同过滤算法 PMF<sup>[43]</sup> (Probabilistic Matrix Factorization) 是一种经典的推荐算法模型,该算法假设每个用户和产品的潜在属性都用一维向量表示,而用户对于特定产品的评价则由对应的两个向量的点积获得. 与基于记忆的推荐算法相比,该模型可以更有效的处理大

规模数据和稀疏数据,不需要花费大量的查询时间,从而很快的生成推荐结果。

假设  $\mathbf{U} \in \mathbf{R}^{K \times N}$  和  $\mathbf{V} \in \mathbf{R}^{K \times M}$  分别表示用户和产品的潜在矩阵,其中  $\mathbf{U}_u$  和  $\mathbf{V}_i$  分别表示用户  $u$  和产品  $i$  的  $K$  维潜在向量. 概率矩阵分解的目的就是从评分中学习到了这些向量。

那么,评分矩阵  $\mathbf{R}$  的条件概率为

$$p(\mathbf{R}|\mathbf{U},\mathbf{V},\sigma_R^2) = \prod_{u=1}^N \prod_{i=1}^M [N(\mathbf{R}_{u,i} | g(\mathbf{U}_u^T \mathbf{V}_i), \sigma_R^2)]^{I_{u,i}^R} \quad (1)$$

其中:  $N(x|\mu, \sigma^2)$  是以  $\mu$  为均值,  $\sigma^2$  为方差的正态分布;  $I_{u,i}^R$  是指示函数,当用户  $u$  对  $i$  有评价时为 1, 否则为 0. 而函数  $g(x)$  是逻辑斯蒂函数  $g(x) = 1/(1 + e^{-x})$ , 将用户评分映射到  $[0, 1]$  范围内. 同时,为了防止过拟合,算法假设用户和产品的潜在向量满足高斯分布:

$$p(\mathbf{U}|\sigma_U^2) = \prod_{u=1}^N N(\mathbf{U}_u | 0, \sigma_U^2 I),$$

$$p(\mathbf{V}|\sigma_V^2) = \prod_{i=1}^M N(\mathbf{V}_i | 0, \sigma_V^2 I) \quad (2)$$

最终,通过贝叶斯推断,用户和产品的潜在特征  $\mathbf{U}$  和  $\mathbf{V}$  的后验概率可以由式(3)获得:

$$p(\mathbf{U}, \mathbf{V} | \mathbf{R}, \sigma_R^2, \sigma_U^2, \sigma_V^2) \propto p(\mathbf{R} | \mathbf{U}, \mathbf{V}, \sigma_R^2) p(\mathbf{U} | \sigma_U^2) p(\mathbf{V} | \sigma_V^2) \\ = \prod_{u=1}^N \prod_{i=1}^M [N(\mathbf{R}_{u,i} | g(\mathbf{U}_u^T \mathbf{V}_i), \sigma_R^2)]^{I_{u,i}^R} \times \prod_{u=1}^N N(\mathbf{U}_u | 0, \sigma_U^2 I) \times \prod_{i=1}^M N(\mathbf{V}_i | 0, \sigma_V^2 I) \quad (3)$$

在式(3)的基础上,利用用户评分矩阵,可以学习到用户和产品的潜在向量,从而进行预测。

#### 3.3 SocialMF 推荐算法

Ma 等人<sup>[25]</sup>提出的 SocialMF 算法将信任传播与概率矩阵分解模型相结合,充分利用了社交网络中隐含的信息。

由于社交关系的影响,用户的行为受到它的直接好友的影响. 换句话说,用户的潜在偏好向量与他的所有相关好友相似,即用户的潜在向量可以用他的所有好友偏好的平均值表示:

$$\hat{\mathbf{U}} = \frac{\sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v}{\sum_{v \in N_u} \mathbf{T}_{u,v}} = \frac{\sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v}{N_u} \quad (4)$$

其中  $\hat{\mathbf{U}}$  表示在给定好友的情况下估计的用户偏好向量. 同时,将信任矩阵进行归一化处理,使得  $\sum_{v \in N_u} \mathbf{T}_{u,v} = 1$ , 于是得到

$$\hat{\mathbf{U}} = \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v \quad (5)$$

与式(3)相似,通过贝叶斯推断,可以得到如下关于用户和产品的潜在向量的后验概率:

$$\begin{aligned} p(\mathbf{U}, \mathbf{V} | \mathbf{R}, \mathbf{T}, \sigma_R^2, \sigma_T^2, \sigma_U^2, \sigma_V^2) \\ \propto p(\mathbf{R} | \mathbf{U}, \mathbf{V}, \sigma_R^2) p(\mathbf{U} | \mathbf{T}, \sigma_U^2, \sigma_T^2) p(\mathbf{V} | \sigma_V^2) \\ = \prod_i \prod_u [N(\mathbf{R}_{u,i} | g(\mathbf{U}_u^T \mathbf{V}_i), \sigma_R^2)]^{I_{u,i}^R} \times \\ \prod_{u=1}^N N(\mathbf{U}_u | \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v, \sigma_U^2 I) \times \\ \prod_{u=1}^N N(\mathbf{U}_u | 0, \sigma_U^2 I) \times \prod_{i=1}^M N(\mathbf{V}_i | 0, \sigma_V^2 I) \quad (6) \end{aligned}$$

其概率模型如图 1 所示,为了便于求解,取 log 对数后验概率,如式(7)所示:

$$\begin{aligned} \ln p(\mathbf{U}, \mathbf{V} | \mathbf{R}, \mathbf{T}, \sigma_R^2, \sigma_T^2, \sigma_U^2, \sigma_V^2) = \\ -\frac{1}{2\sigma_R^2} \sum_{u=1}^N \sum_{i=1}^M I_{u,i}^R (\mathbf{R}_{u,i} - g(\mathbf{U}_u^T \mathbf{V}_i)) - \\ \frac{1}{2\sigma_U^2} \sum_{u=1}^N \mathbf{U}_u^T \mathbf{U}_u - \frac{1}{2\sigma_V^2} \sum_{i=1}^M \mathbf{V}_i^T \mathbf{V}_i - \\ \frac{1}{2\sigma_T^2} \sum_{u=1}^N ((\mathbf{U}_u - \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v)^T (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v)) - \\ \frac{1}{2} \left( \sum_{u=1}^N \sum_{i=1}^M I_{u,i}^R \right) \ln \sigma_R^2 - \frac{1}{2} ((N \times K) \ln \sigma_U^2 + \\ (M \times K) \ln \sigma_V^2 + (N \times K) \ln \sigma_T^2) + C \quad (7) \end{aligned}$$

最大化式(7)相当于最小化如下的损失函数:

$$\begin{aligned} L(\mathbf{R}, \mathbf{T}, \mathbf{U}, \mathbf{V}) = \frac{1}{2} \sum_{u=1}^N \sum_{i=1}^M I_{u,i}^R (\mathbf{R}_{u,i} - g(\mathbf{U}_u^T \mathbf{V}_i))^2 + \\ \frac{\lambda_U}{2} \sum_{u=1}^N \mathbf{U}_u^T \mathbf{U}_u + \frac{\lambda_V}{2} \sum_{i=1}^M \mathbf{V}_i^T \mathbf{V}_i + \\ \frac{\lambda_T}{2} \sum_{u=1}^N ((\mathbf{U}_u - \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v)^T \cdot \\ (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v)) \quad (8) \end{aligned}$$

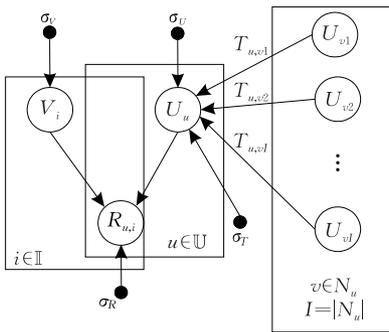


图 1 基于信任传播的社会化推荐算法模型图

在式(8)中,  $\lambda_U = \sigma_R^2 / \sigma_U^2$ ,  $\lambda_V = \sigma_R^2 / \sigma_V^2$  以及  $\lambda_T = \sigma_R^2 / \sigma_T^2$ . 我们可以通过梯度下降法求得式(8)的局部最优解:

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{U}_u} = \sum_{i=1}^M I_{u,i}^R \mathbf{V}_i g'(\mathbf{U}_u^T \mathbf{V}_i) (g(\mathbf{U}_u^T \mathbf{V}_i) - \mathbf{R}_{u,i}) + \\ \lambda_U \mathbf{U}_u + \lambda_T (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{T}_{u,v} \mathbf{U}_v) - \\ \lambda_T \sum_{\{v | u \in N_v\}} \mathbf{T}_{v,u} (\mathbf{U}_v - \sum_{w \in N_v} \mathbf{T}_{v,w} \mathbf{U}_w) \quad (9) \\ \frac{\partial L}{\partial \mathbf{V}_i} = \sum_{i=1}^N I_{u,i}^R \mathbf{U}_v g'(\mathbf{U}_u^T \mathbf{V}_i) (g(\mathbf{U}_u^T \mathbf{V}_i) - \mathbf{R}_{u,i}) + \lambda_V \mathbf{V}_i \quad (10) \end{aligned}$$

其中:  $g'(x)$  是逻辑斯蒂函数的导数, 相当于  $g'(x) = e^{-x} / (1 + e^{-x})^2$ .  $\mathbf{U}$  和  $\mathbf{V}$  的初始值是由均值为 0 的标准正态分布生成, 在每次迭代中根据式(9)和式(10)计算更新步长, 最终得到局部最优解.

## 4 一种基于信任关系隐含相似度的社会化推荐算法

本节主要介绍文中所提出算法的模型推导和理论分析. 首先从信任者和被信任者的角度对用户进行建模, 然后分析提出我们的信任关系隐含相似度, 最终通过概率方法综合到同一代价函数当中. 除此之外, 我们还给出了算法的梯度求解方法、伪代码以及算法复杂度分析.

### 4.1 信任者和被信任者

社会学的相关研究表明<sup>[46]</sup>, 用户之间的信任是非对称的, 即如果 A 信任用户 B, 并不能保证 B 也对 A 具有同样程度的信任. 也就是说, 用户在作为信任者和被信任者时所表现出的偏好并不相同.

如图 2 中所示, A 和 B 表示两个用户, I 表示产品. A 作为信任者接受其他用户的影响时, 与 B 的信任关系强度为 0.2, 而他作为被信任者影响其他用户时, 与 B 的信任关系强度为 0.8. 如图 2(a) 所示, 当已知 B 对 I 的评价为 5 时, 从信任关系强度的角度, 我们可以根据 A 受 B 的影响程度, 估计 A 对 I 的评价为相似度较低的 1 分, 反之, 如图 2(b) 则估计 B 对 A 的评价为较相似的 4 分.

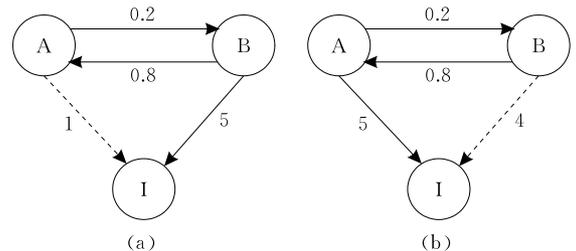


图 2 用户的信任者和被信任者示意图

我们可以看到用户作为信任者和被信任者时表现出不同的特性, 因此, 文中采用  $\mathbf{P} \in \mathbf{R}^{K \times M}$  和  $\mathbf{W} \in$

$R^{K \times M}$  分别表示用户作为信任者和被信任者时的  $K$  维偏好矩阵。

#### 4.2 信任关系隐含相似度

社会学和计算机科学共同研究表明<sup>[36,42,47]</sup>, 如果用户信任相同的对象或被相同的对象信任, 那么他们之间就存在一定的潜在关联. 例如, 关注或信任同一个健身专家的两个用户, 有可能都在学习健身的相关知识, 那么他们之间的经验就可以相互学习并进行推荐.

尽管已有文献对用户作为信任者和被信任者的情况进行建模分析<sup>[36-37,42]</sup>, 并且根据信任关系中的信息对用户偏好向量进行约束. 然而, 这些算法仍然存在一些不足之处需要进一步研究.

这些研究对该隐含信息的挖掘都是基于共同的信任与被信任集合, 这没有考虑到间接社交关系的影响. 导致在这些集合的数量非常少 ( $< 5$ ) 或者为 0 时, 计算得到的信任关系隐含相似度不能准确反映用户之间的信任强度; 另外, 这些算法对该信息的研究没有综合考虑用户分别作为信任者和被信任者时的关联, 而这使得对于信任关系的挖掘不够全面.

如图 3 所示, A、B、C、D 和 E 表示社交关系网络中的 5 个用户. 我们要从信任者的角度估计 A 对 B 的信任强度, 那么, 在只考虑直接的信任集合时, 图 3(a) 和图 3(b) 中得到的估计值是相同的; 而如果我们同时考虑了间接的社交影响, 则在图 3(a) 中, 用户 A 和 B 通过 C 和 E 的连接产生相互影响, 从而 AB 在图 3(a) 中的信任强度要大于图 3(b).

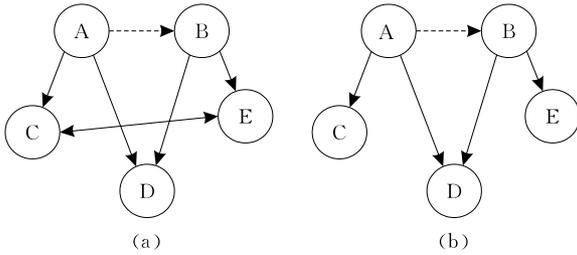


图 3 间接信任的影响示意图

当然, 现实中的连接情况更复杂多样. Ma 在文献<sup>[47]</sup>中的研究表明, 用户之间的关联强度与他们之间的子图连接情况有关, 而子图连接反映了用户的间接连接强度. 该研究中展示了许多复杂拓扑连接时的统计分析, 虽然可以直接通过分析关系子图的情况来统计分析信任强度, 然而如 Ma<sup>[47]</sup>所述, 各种子图的情况非常多, 而且每一种连接图的信任强度都不同, 这不利于对子图情况的精确建模.

因此, 为了充分挖掘信任关系中的信息, 我们提

出一种信任关系隐含相似度的计算方法, 不仅充分考虑用户同时或分别作为信任者和被信任者时的关联, 而且隐含了用户之间的间接社交关联.

首先, 我们采用概率矩阵分解的方法, 对信任关系数据  $T$  进行分解, 建立用户分别作为信任者和被信任者时的关系. 然后, 根据获得的向量计算用户同时作为信任者或被信任者时的相似度. 值得注意的是, 这些向量反映了社交网络中用户之间相互关系情况, 不仅包含了直接联系, 同时也隐含了全体社交网络的间接影响. 具体如图 3(a), 在目标子空间中, A 被约束与 C 的相关, 而 C 与 E 相关, 同样 E 与 B 的距离也是相关的, 那么通过这样的约束, 隐含的建立了 A 与 B 的间接关联. 基于此, 我们计算得到了更为精确的信任关系隐含相似度.

为了建立用户分别作为信任者和被信任者时的关系, 假设信任关系  $T$  由关于用户的信任向量  $P$  和  $W$  的正态分布生成. 则  $T$  的条件概率分布如下:

$$p(T|P, W, \sigma_F^2) = \prod_{u=1}^N \prod_{v \in N_u} [N(T_{u,v} | g(P_u^T W_v), \sigma_F^2)] \quad (11)$$

其中, 我们采用逻辑斯蒂函数  $g(x) = 1/(1 + e^{-x})$  将  $P^T W$  映射到  $[0, 1]$  的范围内, 以符合信任关系  $T$  的取值范围.  $N(T_{u,v} | g(P_u^T W_v), \sigma_T^2)$  表示  $T_{u,v}$  是由以  $g(P_u^T W_v)$  为均值,  $\sigma_T^2$  为方差的高斯分布生成.

同时, 为了防止过拟合, 假设信任向量和被信任向量分别满足均值为 0, 方差为  $\sigma_P^2$  和  $\sigma_W^2$  的高斯分布:

$$p(P|\sigma_P^2) = \prod_{u=1}^N N(P_u | 0, \sigma_P^2 I),$$

$$p(W|\sigma_W^2) = \prod_{v=1}^N N(W_v | 0, \sigma_W^2 I) \quad (12)$$

从而, 根据贝叶斯公式, 可以得到信任向量和被信任向量的后验概率分布如下:

$$p(P, W|T, \sigma_F^2, \sigma_P^2, \sigma_W^2) \propto p(T|P, W, \sigma_F^2) p(P|\sigma_P^2) p(W|\sigma_W^2)$$

$$= \prod_{u=1}^N \prod_{v \in N_u} [N(T_{u,v} | g(P_u^T W_v), \sigma_F^2)] \times \prod_{u=1}^N N(P_u | 0, \sigma_P^2 I) \times \prod_{v=1}^N N(W_v | 0, \sigma_W^2 I) \quad (13)$$

我们对上式取对数, 可以得到对数后验概率分布:

$$\ln p(P, W|T, \sigma_F^2, \sigma_P^2, \sigma_W^2) = -\frac{1}{2\sigma_T^2} \sum_{u=1}^N \sum_{v \in N_u} (T_{u,v} - P_u^T W_v)^2 - \frac{1}{2\sigma_P^2} \sum_{u=1}^N P_u^T P_u - \frac{1}{2\sigma_W^2} \sum_{v=1}^N W_v^T W_v -$$

$$\frac{1}{2} \sum_{u=1}^N \sum_{v \in N_u} \ln \sigma_F^2 - \frac{1}{2} ((N \times K) \ln \sigma_P^2 + (N \times K) \ln \sigma_W^2) + C \quad (14)$$

在保持方差参数不变的情况下,最大化对数后验概率分布相当于最小化如下的损失函数:

$$L_F(\mathbf{P}, \mathbf{W}, \mathbf{T}) = \frac{\lambda_F}{2} \sum_{u=1}^N \sum_{v \in N_u} (\mathbf{T}_{u,v} - \mathbf{P}^T \mathbf{W})^2 + \frac{\lambda_P}{2} \sum_{u=1}^N \mathbf{P}_u^T \mathbf{P}_u + \frac{\lambda_W}{2} \sum_{v=1}^N \mathbf{W}_v^T \mathbf{W}_v \quad (15)$$

通过梯度下降法求解得到用户的信任向量  $\mathbf{P}$  和被信任向量  $\mathbf{W}$ .

根据上面获得的结果,我们可以计算用户  $u$  和用户  $v$  的信任关系隐含相似度. 为了全面的利用用户的社交信息,我们分别考虑用户作为信任者和被信任者时的信任关系隐含相似度,并且综合起来提升推荐算法的精度.

用户  $u$  和用户  $v$  都作为信任者时的信任关系隐含相似度:

$$\mathbf{S}_{u,v}^P = \mathbf{P}_u^T \mathbf{P}_v \quad (16)$$

以及都作为被信任者时的信任关系隐含相似度:

$$\mathbf{S}_{u,v}^W = \mathbf{W}_u^T \mathbf{W}_v \quad (17)$$

由此,我们得到了用户分别同时作为信任者和被信任者时,用户  $u$  和  $v$  之间的信任关系隐含相似度,包括  $\mathbf{S}_{u,v}^P$  和  $\mathbf{S}_{u,v}^W$ .

从公式可以看出,我们的信任关系隐含相似度计算具有以下优点:(1)综合考虑了用户同时及分别作为信任者和被信任者时的关联,更加全面的利用了信任关系中的信息;(2)并不依赖于用户之间的共同好友集合,即使在没有共同好友的情况下也能计算出可靠的结果;(3)所有用户向量在计算中隐含考虑了整体的信息,即包含了间接的社交关联. 因此,我们得到了一个精确的信任关系隐含相似度的方法.

### 4.3 信任强度估计

已有的工作表明<sup>[34,36-37,42]</sup>,评分和信任数据能够在一定程度上揭示用户之间的信任强度. 然而,这些工作并没有综合考虑到两者信息对于每组用户信任强度的影响. 虽然 Wang 等人<sup>[37]</sup>进行了一些相关的探索,然而,他们并没有对每组信任强度都综合考虑两者信息. 为了更加精确的估计用户间的信任强度,我们提出一种综合考虑评分相似度和信任关系隐含相似度的信任强度估计方法.

我们首先计算用户  $u$  和  $v$  之间的评分相似度  $\mathbf{S}_{u,v}^R$ ,

由用户的评分向量  $\mathbf{U}_u$  和  $\mathbf{U}_v$  计算得到

$$\mathbf{S}_{u,v}^R = \mathbf{U}_u^T \mathbf{U}_v \quad (18)$$

我们假设用户间的信任强度矩阵为  $\mathbf{S}$ ,由关于评分相似度  $\mathbf{S}_{u,v}^R$ 、信任关系隐含相似度  $\mathbf{S}_{u,v}^P$  和  $\mathbf{S}_{u,v}^W$  的正太分布生成,则  $\mathbf{S}$  的条件概率分布为

$$p(\mathbf{S} | \mathbf{S}_{u,v}^R, \mathbf{S}_{u,v}^P, \mathbf{S}_{u,v}^W, \sigma_S^2) = \prod_{u=1}^N \prod_{v \in N_u} [N(\mathbf{S}_{u,v} | g(\mathbf{S}_{u,v}^R + \mathbf{S}_{u,v}^P + \mathbf{S}_{u,v}^W), \sigma_S^2)] \quad (19)$$

其中  $\sigma_S^2$  表示该高斯分布的方差,也可以理解为估计值的噪声情况. 对式(19)取对数,得到对数后验概率分布:

$$\begin{aligned} \ln p(\mathbf{S} | \mathbf{S}_{u,v}^R, \mathbf{S}_{u,v}^P, \mathbf{S}_{u,v}^W, \sigma_S^2) = & -\frac{1}{2\sigma_S^2} \sum_{u=1}^N \sum_{v \in N_u} (\mathbf{S}_{u,v} - g(\mathbf{S}_{u,v}^R + \mathbf{S}_{u,v}^P + \mathbf{S}_{u,v}^W))^2 - \\ & \frac{1}{2} \sum_{u=1}^N \sum_{v \in N_u} \ln \sigma_S^2 \end{aligned} \quad (20)$$

为了最大化式(20),我们需要最小化如下的损失函数:

$$\begin{aligned} L_S(\mathbf{S}, \mathbf{U}, \mathbf{P}, \mathbf{W}) & = \frac{\lambda_S}{2} \sum_{u=1}^N \sum_{v \in N_u} (\hat{\mathbf{S}}_{u,v} - g(\mathbf{S}_{u,v}^R + \mathbf{S}_{u,v}^P + \mathbf{S}_{u,v}^W))^2 \\ & = \frac{\lambda_S}{2} \sum_{u=1}^N \sum_{v \in N_u} (\hat{\mathbf{S}}_{u,v} - g(\mathbf{U}_u^T \mathbf{U}_v + \mathbf{P}_u^T \mathbf{P}_v + \mathbf{W}_u^T \mathbf{W}_v))^2 \end{aligned} \quad (21)$$

显然,当  $\hat{\mathbf{S}}_{u,v}$  等于  $g(\mathbf{S}_{u,v}^R + \mathbf{S}_{u,v}^P + \mathbf{S}_{u,v}^W)$  时,取得最小值 0. 但是为了同时兼顾其它因素,我们将式(21)作为关于信任强度矩阵  $\hat{\mathbf{S}}$  的正则化约束项,与式(8)进行结合,得到最终有效的估计值. 值得注意的是,式(21)作为正则化项时,当  $\lambda_S \rightarrow +\infty$ ,  $\hat{\mathbf{S}}_{u,v}$  将完全等于  $g(\mathbf{S}_{u,v}^R + \mathbf{S}_{u,v}^P + \mathbf{S}_{u,v}^W)$ ;但考虑到即使融合了评分和信任关系隐含相似度,得到的信任强度估计仍然存在噪声误差,我们将在实验中选择表现最好的参数作为最终结果.

### 4.4 算法模型

在以上分析的基础上,为了充分挖掘社交关系中的隐含信息,本文提出一种基于信任关系隐含相似度的社会化推荐算法 SocialIT (Social Recommendation Algorithm based on Implicit Similarity in Trust).

首先,采用概率矩阵分解 PMF<sup>[43]</sup>的方法对评分进行建模,并且考虑了用户和产品偏置的影响<sup>[17]</sup>:

$$\hat{\mathbf{R}}_{u,i} = g(\mathbf{U}_u \mathbf{V}_i + \mathbf{b}_u^U + \mathbf{b}_i^V) \quad (22)$$

其中,  $\mathbf{b}_u^U$  和  $\mathbf{b}_i^V$  分别表示用户  $u$  和产品  $i$  的评分偏置.

然后,本文采用与 SocialMF<sup>[25]</sup>算法相同的假

设,即用户偏好与他的信任好友的平均值相似,如式(4)所示.不同之处在于,文中使用估计的信任强度矩阵  $\hat{S}$  代替信任矩阵  $T$ :

$$\hat{U}_u = \frac{\sum_{v \in N_u} \mathbf{s}_{u,v} \mathbf{U}_v}{\sum_{v \in N_u} \mathbf{s}_{u,v}} = \sum_{v \in N_u} \hat{S}_{u,v}^* \mathbf{U}_v \quad (23)$$

其中,  $\hat{S}_{u,v}^* = \hat{S}_{u,v} / \sum_{v \in N_u} \hat{S}_{u,v}$  表示归一化的信任强度.

因此,基于 SocialMF<sup>[25]</sup> 算法的式(8),我们综合考虑了评分相似和信任关系隐含相似度的影响,结合式(15)、式(21)、式(22)和式(23),得到我们的代价函数为

$$\begin{aligned} E = & \frac{1}{2} \sum_{u=1}^N \sum_{i=1}^M I_{u,i}^R (\mathbf{R}_{u,i} - \hat{\mathbf{R}}_{u,i})^2 + \\ & \frac{\lambda_U}{2} \sum_{u=1}^N \mathbf{U}_u^T \mathbf{U}_u + \frac{\lambda_V}{2} \sum_{u=1}^N \mathbf{V}_i^T \mathbf{V}_i + \\ & \frac{\lambda_T}{2} \sum_{u=1}^N ((\mathbf{U}_u - \sum_{v \in N_u} \mathbf{s}_{u,v}^* \mathbf{U}_v)^T (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{s}_{u,v}^* \mathbf{U}_v)) + \\ & \frac{\lambda_S}{2} \sum_{u=1}^N \sum_{v \in N_u} (\mathbf{s}_{u,v} - g(\mathbf{U}_u^T \mathbf{U}_v + \mathbf{P}_u^T \mathbf{P}_v + \mathbf{W}_u^T \mathbf{W}_v))^2 + \\ & \frac{\lambda_F}{2} \sum_{u=1}^N \sum_{v \in N_u} (\mathbf{T}_{u,v} - g(\mathbf{P}_u^T \mathbf{W}_v))^2 + \\ & \frac{\lambda_P}{2} \sum_{u=1}^N \mathbf{P}_u^T \mathbf{P}_u + \frac{\lambda_W}{2} \sum_{u=1}^N \mathbf{W}_u^T \mathbf{W}_u \end{aligned} \quad (24)$$

通过最小化代价函数(24),我们可以得到用户和产品的评分向量  $\mathbf{U}$  和  $\mathbf{V}$ ,然后根据式(22)对用户评分进行预测.

SocialIT 算法的代价函数(24)对应的概率图模型如图4所示,文中算法 SocialIT 与 SocialMF<sup>[25]</sup> 算法相比,主要有以下特点:(1) SocialIT 并不是将每个好友对用户的影响视为等同,而是采用估计的信任强度矩阵  $\hat{S}$  代替信任矩阵  $T$ ; (2) SocialIT 利用概率矩阵分解的方法得到信任者和被信任者向量,并

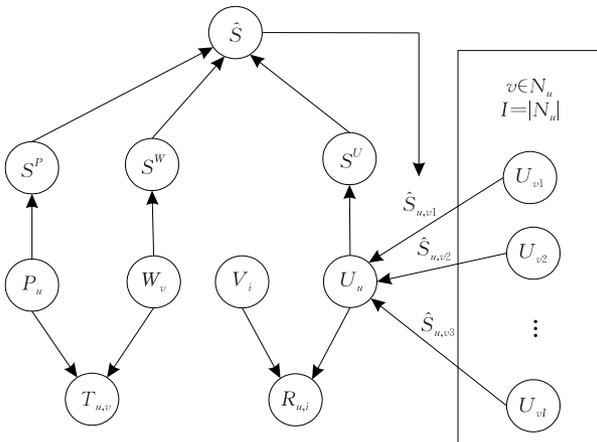


图4 SocialIT 算法示意图

给予此计算信任关系隐含相似度,从而隐含了用户间的间接社交影响;(3) SocialIT 同时考虑了评分相似和信任关系隐含相似度对每组用户间信任强度的影响,得到更为精确的估计结果,进而提高了推荐精度.

#### 4.5 问题求解

文中采用梯度下降法求解得到式(24)的局部最优解,具体的计算过程如算法1所示,各变量的梯度计算如下:

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{U}_u} = & \sum_{i=1}^M I_{u,i}^R \mathbf{V}_i g'(\hat{\mathbf{R}}_{u,i}) (g(\hat{\mathbf{R}}_{u,i}) - \mathbf{R}_{u,i}) + \\ & \lambda_U \mathbf{U}_u + \lambda_T (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{s}_{u,v}^* \mathbf{U}_v) - \\ & \lambda_T \sum_{\{v|u \in N_v\}} \mathbf{s}_{v,u}^* (\mathbf{U}_v - \sum_{w \in N_v} \mathbf{s}_{w,v}^* \mathbf{U}_w) + \\ & \lambda_S \sum_{v \in N_u} [(\mathbf{s}_{u,v} - g(\hat{\mathbf{S}}_{u,v})) g'(\hat{\mathbf{S}}_{u,v}) \mathbf{U}_v] + \\ & \lambda_S \sum_{\{v|u \in N_v\}} [(\mathbf{s}_{v,u} - g(\hat{\mathbf{S}}_{v,u})) g'(\hat{\mathbf{S}}_{v,u}) \mathbf{U}_v] \end{aligned} \quad (25)$$

$$\frac{\partial E}{\partial \mathbf{V}_i} = \sum_{i=1}^N I_{u,i}^R \mathbf{U}_v g'(\mathbf{U}_u^T \mathbf{V}_i) (g(\mathbf{U}_u^T \mathbf{V}_i) - \mathbf{R}_{u,i}) + \lambda_V \mathbf{V}_i \quad (26)$$

$$\frac{\partial E}{\partial \mathbf{s}_{u,v}} = \lambda_S (\mathbf{s}_{u,v} - g(\hat{\mathbf{S}}_{u,v})) + \lambda_T (\mathbf{U}_u - \sum_{v \in N_u} \mathbf{s}_{u,v}^* \mathbf{U}_v) \left[ \frac{\mathbf{U}_v \sum_{v \in N_u} \mathbf{s}_{u,v} - \sum_{v \in N_u} \mathbf{s}_{u,v} \mathbf{U}_v}{(\sum_{v \in N_u} \mathbf{s}_{u,v})^2} \right] \quad (27)$$

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{P}_u} = & \lambda_S (\mathbf{s}_{u,v} - g(\hat{\mathbf{S}}_{u,v})) g'(\hat{\mathbf{S}}_{u,v}) \mathbf{P}_v + \\ & \lambda_S (\mathbf{s}_{v,u} - g(\hat{\mathbf{S}}_{v,u})) g'(\hat{\mathbf{S}}_{v,u}) \mathbf{P}_v + \\ & \lambda_F \sum_{v \in N_u} (\mathbf{T}_{u,v} - g(\mathbf{P}_u^T \mathbf{W}_v)) g'(\mathbf{P}_u^T \mathbf{W}_v) \mathbf{W}_v + \lambda_P \mathbf{P}_u \end{aligned} \quad (28)$$

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{W}_u} = & \lambda_S (\mathbf{s}_{u,v} - g(\hat{\mathbf{S}}_{u,v})) g'(\hat{\mathbf{S}}_{u,v}) \mathbf{W}_v + \\ & \lambda_S (\mathbf{s}_{v,u} - g(\hat{\mathbf{S}}_{v,u})) g'(\hat{\mathbf{S}}_{v,u}) \mathbf{W}_v + \\ & \lambda_F \sum_{v \in N_u} (\mathbf{T}_{u,v} - g(\mathbf{P}_u^T \mathbf{W}_v)) g'(\mathbf{P}_u^T \mathbf{W}_v) \mathbf{P}_v + \lambda_W \mathbf{W}_u \end{aligned} \quad (29)$$

其中,为了简洁起见,我们用  $\hat{S}_{u,v}$  表示  $\mathbf{U}_u^T \mathbf{U}_v + \mathbf{P}_u^T \mathbf{P}_v + \mathbf{W}_u^T \mathbf{W}_v$ ,  $\mathbf{s}_{u,v}^*$  表示归一化的信任强度  $\mathbf{s}_{u,v}^* = \mathbf{s}_{u,v} / \sum_{v \in N_u} \mathbf{s}_{u,v}$ .

#### 算法1. SocialIT 算法.

输入:用户-产品评分数据  $\mathbf{R}$ ,用户-用户社交数据  $\mathbf{T}$ ,潜在特征维数  $K$ ,正则化系数  $\lambda_U, \lambda_V, \lambda_T, \lambda_S, \lambda_P$  和  $\lambda_W$ ,学习率  $\alpha$ ,迭代次数  $L$

输出:用户评分偏好向量  $\mathbf{U}$ ,产品特征向量  $\mathbf{V}$ ,用户和产品偏置  $\mathbf{b}^U$  和  $\mathbf{b}^V$ ,用户信任者向量  $\mathbf{P}$ ,用户被信任者向量  $\mathbf{W}$ ,用户信任强度矩阵  $\mathbf{S}$

BEGIN

1. 初始化:随机初始化  $U, V, P, W, b^U$  和  $b^V$  (我们采用方差为 0.1 的标准正态分布进行初始化), 信任强度矩阵  $S$  的元素全部初始化为 1.
  2. WHILE  $l < L$  DO:
  3. FOR ALL  $r_{u,i} \in R$ , DO:
  4.  $U_u \leftarrow U_u - \alpha \frac{\partial E}{\partial U_u}, V_i \leftarrow V_i - \alpha \frac{\partial E}{\partial V_i}$
  5.  $b_u^U \leftarrow b_u^U - \alpha \frac{\partial E}{\partial b_u^U}, b_i^V \leftarrow b_i^V - \alpha \frac{\partial E}{\partial b_i^V}$
  6.  $P_u \leftarrow P_u - \alpha \frac{\partial E}{\partial P_u}, W_u \leftarrow W_u - \alpha \frac{\partial E}{\partial W_u}$
  7.  $S_{u,v} \leftarrow S_{u,v} - \alpha \frac{\partial E}{\partial S_{u,v}}$
  8. END FOR
  9.  $l \leftarrow l + 1$
  10. End WHILE
- END

#### 4.6 SocialIT 算法复杂度分析

文中算法 SocialIT 的求解过程中, 最主要的计算代价在于损失函数  $E$  以及每个变量对应梯度的计算.

对于损失函数  $E$ , 根据文献[25]的分析, 其对应的时间复杂度为  $O(N\bar{r}K + N\bar{t}K)$ . 其中,  $N$  表示用户数量,  $K$  表示算法维度,  $\bar{r}$  表示用户平均评分数量, 表示用户平均信任关系数量. 由于评分和信任数据通常是稀疏的, 比如表 1 所示,  $\bar{r}, \bar{t}$  和  $K$  的值都比较小. 因此, 损失函数  $E$  的计算是非常快的, 且与总的用户数量  $N$  和算法维度  $K$  成线性正比关系.

表 1 Epinions 数据集的统计特性

| 数据集      | Epinions | Ciao    |
|----------|----------|---------|
| 用户数量     | 22166    | 7375    |
| 产品数量     | 296277   | 106797  |
| 评分记录     | 922267   | 284086  |
| 信任记录     | 300548   | 111781  |
| 用户平均评分数量 | 41.61    | 38.52   |
| 商品平均评分数量 | 3.11     | 2.66    |
| 用户平均信任数量 | 13.56    | 15.16   |
| 评分稀疏度    | 0.99986  | 0.99964 |
| 信任稀疏度    | 0.9988   | 0.9959  |

相对应的, 梯度计算的复杂度的时间复杂度为  $O(N\bar{r}K + M\bar{r}K + N\bar{t}^2K + N\bar{t}K)$ , 其中  $M$  表示产品数量. 由于  $\bar{r}, \bar{t}$  和  $K$  的值都比较小, 因此, 梯度的计算复杂度也不算高, 且与  $N$  成线性正比关系.

综上, 文中算法 SocialIT 的算法复杂度主要受用户数量  $N$  和产品数量  $M$  的影响, 适合大数据情况下的应用.

## 5 实验结果和分析

本节主要展示 SocialIT 算法的与其它相关算法的实验结果. 我们首先介绍实验采用了数据集和评价方法, 然后设计了 4 组实验从不同的角度对比分析本文算法的性能情况.

### 5.1 数据集

为了避免实验的偏向性, 我们选择两个独立的数据集<sup>①</sup>进行算法验证, Epinions 和 Ciao. 这两个公开数据集同时包含了评分数据和在线社交数据, 该数据集由 Tang 等人<sup>[48]</sup>在 2011 年收集并公开.

Epinions<sup>②</sup> 和 Ciao<sup>③</sup> 是两个著名的大众消费点评网站, 主要市场分别在美国和欧洲. 在这两个网站上, 任何用户都可以在注册后对所有的商品进行评分和评论, 同时也可以浏览其他用户对商品的评分和评价, 进而帮助用户做出更加有利的决定. 与此同时, 用户可以与信任的用户建立好友关系, 从而建立社交关系网络.

Tang 等人<sup>[48]</sup>在这两个网站上爬取了从 1999 年至 2011 年的评分和社交数据, 时间范围非常广泛, 是最新的公开社交数据集之一<sup>[48-50]</sup>. 这两个网站的评分范围为 1~5 (1 表示“不喜欢”, 5 表示“喜欢”), 用户之间的信任关系是二值的, 即只能取 1 或 0, 分别对应信任或非信任关系. 文中研究的目的是估计用户之间的潜在信任度, 从而提高推荐精度.

在我们的实验中, 采用与 Wang 等人<sup>[37]</sup>类似的预处理. 即过滤掉评分数量小于 5 的用户和产品, 并且在信任数据也只保留相对应的用户.

### 5.2 评价方法

本文算法的任务之一是预测用户对产品的评分, 通常希望算法越精确越好. 为了评估本文算法的性能, 我们选择两种流行的精确度指标<sup>[51]</sup>, 平均绝对误差 (Mean Absolute Error, MAE) 和均方根误差 (Root Mean Squared Error, RMSE). MAE 和 RMSE 适合于预测的场景, 反映了算法预测评分的准确度. 在仅考虑准确度的情况下, 它们的值越低, 算法的性能越好.

平均绝对误差 MAE 的计算公式为

$$MAE = \frac{1}{N} \sum_{(i,j) \in I} |\hat{R}_{i,j} - R_{i,j}| \quad (30)$$

① <http://www.jiliang.xyz/trust.html>

② <https://en.wikipedia.org/wiki/Epinions>

③ [https://en.wikipedia.org/wiki/Ciao\\_%28website%29](https://en.wikipedia.org/wiki/Ciao_%28website%29)

其中,  $N$  表示预测评分的数量;  $I$  表示所有测试集的数据;  $\hat{R}_{i,j}$  表示预测评分;  $R_{i,j}$  表示真实评分.

相对应的, 均方根误差  $RMSE$  的计算公式为

$$RMSE = \sqrt{\frac{\sum_{(i,j) \in I} (\hat{R}_{i,j} - R_{i,j})^2}{N}} \quad (31)$$

很多推荐系统领域的研究表明<sup>[21]</sup>, 反映评分预测能力的误差率指标 ( $MSE, RMSE$ ) 并不能完全地表征算法的推荐性能. 而与实际场景更为相关的是 Top- $N$  项目推荐性能. 因此, 为了更加全面的验证本文算法的性能, 我们将 SocialIT 算法与面向 Top- $N$  项目推荐任务的 BPR 算法相结合<sup>[21]</sup>, 并采用 3 种反映击中率性能的指标进行度量, 包括精确率 ( $Precision$ )、召回率 ( $Recall$ ) 和排序指标  $NDCG$  (Normalized Discounted Cumulative Gain).

采用  $LI'_u$  表示用户  $u$  的项目推荐序列以及  $LI_u$  表示用户  $u$  在测试数据中预测的项目推荐序列,  $U_{Test}$  表示待测试的用户集合, 则可以得到  $Precision$  和  $Recall$  的计算公式分别为

$$Precision@n = \sum_{u \in U_{Test}} |LI'_u \cap LI_u| / |LI'_u| \quad (32)$$

$$Recall@n = \sum_{u \in U_{Test}} |LI'_u \cap LI_u| / |LI_u| \quad (33)$$

相对应的, 排序指标  $NDCG@n$  的计算公式表示为

$$NDCG@n = \sum_{u \in U_{Test}} \frac{1}{Y_u} \sum_{i=1}^k \frac{2^{t_i} - 1}{\log_2(i+1)} \quad (34)$$

其中,  $Y_u$  表示用户  $u$  的最大  $NDCG@n$  值.  $t_i = 1$  表示击中,  $t_i = 0$  则表示未击中.

### 5.3 实验设计

为了全面地测试算法在一般情况和稀疏数据时的表现, 我们的实验设计与 Yao 等人<sup>[36]</sup> 的类似, 采用以下 3 种不同的策略从不同的角度对算法进行测试, 以验证本文算法的有效性.

(1) 全局测试. 我们验证算法在一般情况下的表现. 采用交叉验证的方法测试算法性能. 具体来说, 就是将数据集随机分成 5 等份, 每次测试将其中 4 份作为训练数据, 剩下一份作为测试数据. 总共进行 5 次测试, 保证所有的数据都被测试一次. 最终得到的平均精度作为算法的评估标准, 从而保证算法验证的可靠性.

(2) 不同评分稀疏度的测试. 采用与 (1) 相同的设定, 但在验证测试集时, 我们分别验证具有不同评分数量 (稀疏程度) 的用户表现, 从而得到算法在不

同评分稀疏度时的性能情况.

(3) 不同信任稀疏度的测试. 采用与 (1) 相同的设定, 但在验证测试集时, 分别验证具有不同信任关系数量用户的表现, 从而得到算法在不同信任稀疏度时的性能情况.

(4) Top- $N$  项目推荐测试: 验证算法预测的最符合用户偏好的项目与用户真实偏好项目的差别, 为此, 我们采用  $Precision$ 、 $Recall$  和  $NDCG$  对算法性能进行度量. 并且也采用 (1) 中描述的交叉验证方法, 进行 5 次测试, 保证测试的可靠性.

### 5.4 比较算法及参数设定

为了对比说明本文算法在评分预测和项目推荐中的性能, 我们选择两组算法分别进行对比与分析.

对于评分预测任务, 我们选择了经典的推荐算法 PMF<sup>[43]</sup> 以及其它社会化推荐算法:

(1) PMF<sup>[43]</sup>. 采用概率矩阵分解的方法, 将用户和产品映射到低维向量空间, 提高了算法处理大量数据的能力和精度.

(2) SoRec<sup>[23]</sup>. 由 Ma 等人<sup>[23]</sup> 提出的采用概率矩阵分解的方法建立用户偏好与社交好友之间的关联.

(3) SocialMF<sup>[25]</sup>. 由 Jamali 等人<sup>[25]</sup> 提出的基于信任传播的社会化推荐算法. 该算法假设每个好友对用户的影响力相同, 本文算法在此基础上, 综合考虑了用户评分和信任关系隐含相似度的影响.

(4) SoReg<sup>[35]</sup>. 由 Ma 等人<sup>[35]</sup> 提出, 采用评分相似作为用户间的信任度量, 并且改进了 SocialMF 中的均值传播方式.

(5) TrustMF<sup>[36]</sup>. 由 Yao 等人<sup>[36]</sup> 提出, 对于用户作为信任者和被信任者时的情况进行分别建模, 并进行集成得到推荐结果.

(6) TS\_MF<sup>[37]</sup>. 由 Wang 等人<sup>[37]</sup> 提出, 考虑了用户评分相似和信任关系中隐含信息的影响.

对于项目推荐任务, 我们选择 PMF 算法以及其它面向排序的推荐算法进行对比:

(1) PMF<sup>[43]</sup>. 概率矩阵分解方法, 将用户和产品映射到低维空间, 并以此预测用户对其它产品的喜好程度.

(2) BPR<sup>[21]</sup>. Rendle 等人将排序学习算法应用于推荐系统, 通过学习用户对一对产品间的偏好关系, 从而预测用户最有可能偏好的产品.

(3) GBPR<sup>[52]</sup>. Pan 等人在这项工作中结合了社交群体对于用户偏好的影响, 从而提高了项目推荐质量.

(4) SBPR<sup>[53]</sup>, Zhao 等人考虑了用户与社交好友选择的产品之间的直接关联,有效利用了社交关系对于项目推荐的影响。

为了得出更为可靠的对比结果,我们根据作者推荐的参数值以及多次参数调优的结果,分别选择最优的参数作为最终的对比结果,并且验证了算法在特征向量维度分别为 5、10 和 20 时的表现. 本文实验代码在公开的算法库 Librec<sup>①</sup> 的基础上进行改

进,该算法库对 SocialMF 进行了实现,文中运行结果与其给出的实验结果一致。

## 5.5 实验结果与分析

### 实验 1. 全局测试.

为了使测试不失偏向性,我们在 Epinions 和 Ciao 数据集上进行了测试,并且验证了算法在维度为 5、10 和 20 精度. 在表 2 中展示了各种算法在 Epinions 和 Ciao 数据集中的实验结果,从中我们可以看出:

表 2 在 Epinions 和 Ciao 数据集的精度表现

| 数据集              | 度量   | PMF    | SoRec  | SocialMF | SoReg          | TrustMF        | TS_MF          | SocialIT      | 提高/% |
|------------------|------|--------|--------|----------|----------------|----------------|----------------|---------------|------|
| Epinions<br>K=5  | MAE  | 0.9095 | 0.8406 | 0.8714   | 0.8332         | <b>0.8282*</b> | 0.8467         | <b>0.8046</b> | 2.85 |
|                  | RMSE | 1.1347 | 1.0758 | 1.0809   | <b>1.0674*</b> | 1.0722         | 1.0811         | <b>1.0403</b> | 2.54 |
| Epinions<br>K=10 | MAE  | 0.9098 | 0.8463 | 0.8549   | <b>0.8396*</b> | 0.8428         | 0.8463         | <b>0.8086</b> | 3.69 |
|                  | RMSE | 1.1491 | 1.1047 | 1.0904   | 1.1109         | 1.1208         | <b>1.0804*</b> | <b>1.0420</b> | 3.55 |
| Epinions<br>K=20 | MAE  | 0.9164 | 0.8829 | 0.8730   | 0.8839         | 0.8934         | <b>0.8461*</b> | <b>0.8156</b> | 3.60 |
|                  | RMSE | 1.1581 | 1.2260 | 1.1932   | 1.2399         | 1.2893         | <b>1.0806*</b> | <b>1.0485</b> | 2.97 |
| Ciao<br>K=5      | MAE  | 0.8214 | 0.7987 | 0.7831   | 0.7781         | <b>0.7534*</b> | 0.7555         | <b>0.7370</b> | 2.08 |
|                  | RMSE | 1.0483 | 1.0308 | 1.0087   | 0.9924         | <b>0.9794*</b> | 0.9819         | <b>0.9605</b> | 1.93 |
| Ciao<br>K=10     | MAE  | 0.8179 | 0.8273 | 0.7849   | 0.7632         | <b>0.7520*</b> | 0.7559         | <b>0.7414</b> | 1.41 |
|                  | RMSE | 1.0438 | 1.0643 | 1.0100   | 1.0011         | 1.0030         | <b>0.9823*</b> | <b>0.9729</b> | 0.96 |
| Ciao<br>K=20     | MAE  | 0.8196 | 0.8482 | 0.7815   | 0.7665         | 0.7736         | <b>0.7550*</b> | <b>0.7440</b> | 1.46 |
|                  | RMSE | 1.0455 | 1.1620 | 1.0060   | 1.0970         | 1.1452         | <b>0.9812*</b> | <b>0.9773</b> | 0.40 |

(1) 社会化推荐算法(如 SocialMF 和 SoReg 等)在 MAE 和 RMSE 上都优于仅依赖于用户评分的概率矩阵分解方法 PMF,这表明充分利用社交关系中隐含的用户偏好信息能够有效提升推荐算法的精度。

(2) 本文算法 SocialIT 相比于其它社会化推荐算法,在 MAE 和 RMSE 上有明显的提高. 值得注意的是,考虑了用户间信任强度的算法(SoReg 和 TS\_MF)以及考虑了信任者和被信任者关系的算法(TrustMF 和 TS\_MF)都优于 SocialMF 算法,可见对用户间信任强度以及用户角色(信任者和被信任者)进行建模都能提高社会化推荐算法的结果. 而文中算法更进一步,利用了信任者和被信任者向量对用户间信任强度进行修正,从而得到了更优的推荐质量。

(3) 本文算法在 Epinions 和 Ciao 数据集上都能有效提升推荐精度. 可见,文中算法对于独立收集的不同数据集,也能保持良好的性能,进而验证了文中算法提出的信任关系隐含相似度具有可靠的鲁棒性,并不对特定的数据集有明显的偏向性。

(4) 本文算法 SocialIT 在维度增加的时候,性能有一定程度的退化,但相比于其它算法仍然具有明显的竞争优势. 这是由于文中算法对每个用户从 3 个维度进行了建模,包括(评分者、信任者和被信任者),从而实际上每个用户的维度是单个向量维度的 3 倍,这增强了算法的拟合能力的同时也增加了

过拟合的风险. 从表 2 也可以看出,在维度为 5 时的性能比其它维度都更优. 因此,本算法更为适合采用较低的向量维度。

### 实验 2. 不同评分稀疏度的测试.

为了测试算法对于评分稀疏度的鲁棒性,图 5 中展示了各种算法对于不同评分稀疏度(评分数量)的表现. 不失一般性,我们分别统计了 Epinions 和 Ciao 的结果,其中,横轴表示用户评分数量,纵轴表示 MAE 和 RMSE. 从图 5 中我们可以看出:

(1) 在不同评分稀疏的情况下,SocialIT 算法的精度始终明显优于其它算法. 这表明,对于不同评分数量的用户,文中算法都能够有效挖掘社交数据中的隐含信息,从而提高算法的推荐精度,也使得算法对于各种评分数量用户都具有良好的鲁棒性。

(2) 对于评分数据稀疏(评分数量 < 20)的情况,SocialIT 算法相比于其它算法有明显的提高. 随着评分数量的增加,SocialIT 在 MAE 上与其它算法的差距有所减小,但依然优于其它算法. 这表明,在评分稀疏的情况下,文中算法利用信任关系隐含相似度能够有效的修正推荐结果,得到更好的推荐精度。

(3) 对于评分数据稠密(评分数量 > 100)的情况,相比于仅采用评分数据的 PMF 算法, SocialIT

① <http://www.librec.net>

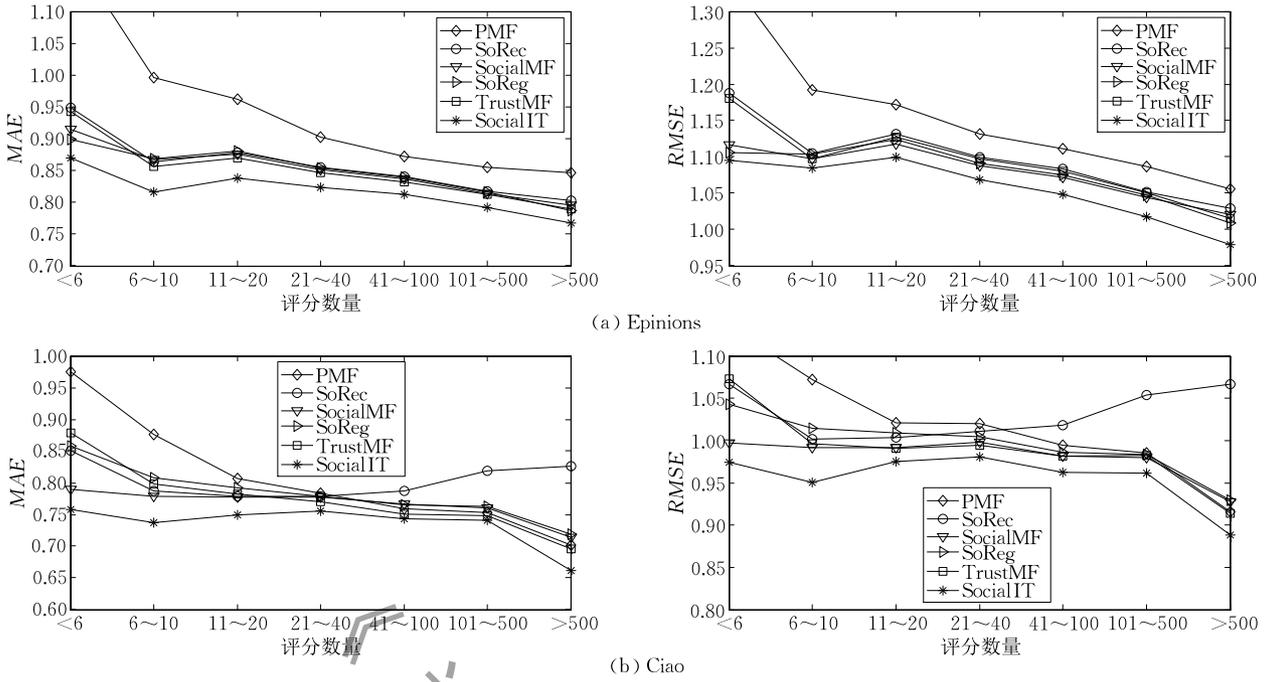


图5 各算法在不同评分稀疏度情况下的精度

有明显的提高,而其它社会化推荐算法则提高较少. 这表明,随着数据量的增多,SocialIT 算法依然能够提取社交网络中的有效信息,从而提高推荐质量.

实验 3. 不同信任稀疏度的测试.

为了测试算法对于信任稀疏度的鲁棒性,图 6 中展示了各种算法对于不同信任稀疏度(信任数量)的表现. 从图中可以看出:

(1) 在不同信任稀疏的情况下,SocialIT 算法的

精度始终明显优于其它算法. 这表明,通过融合评分相似和信任关系隐含相似度,能够有效的提高社会化推荐算法对于不同信任稀疏度时的鲁棒性.

(2) 在信任数据稀疏(信任数量<10)时,SocialIT 算法在 MAE 和 RMSE 精度上明显好于其它算法. 这表明,在信任稀疏的情况下,SocialIT 算法通过评分相似度和信任关系隐含相似度修正对信任强度的估计,得到了更为精确的信任强度估计以及推荐结束.

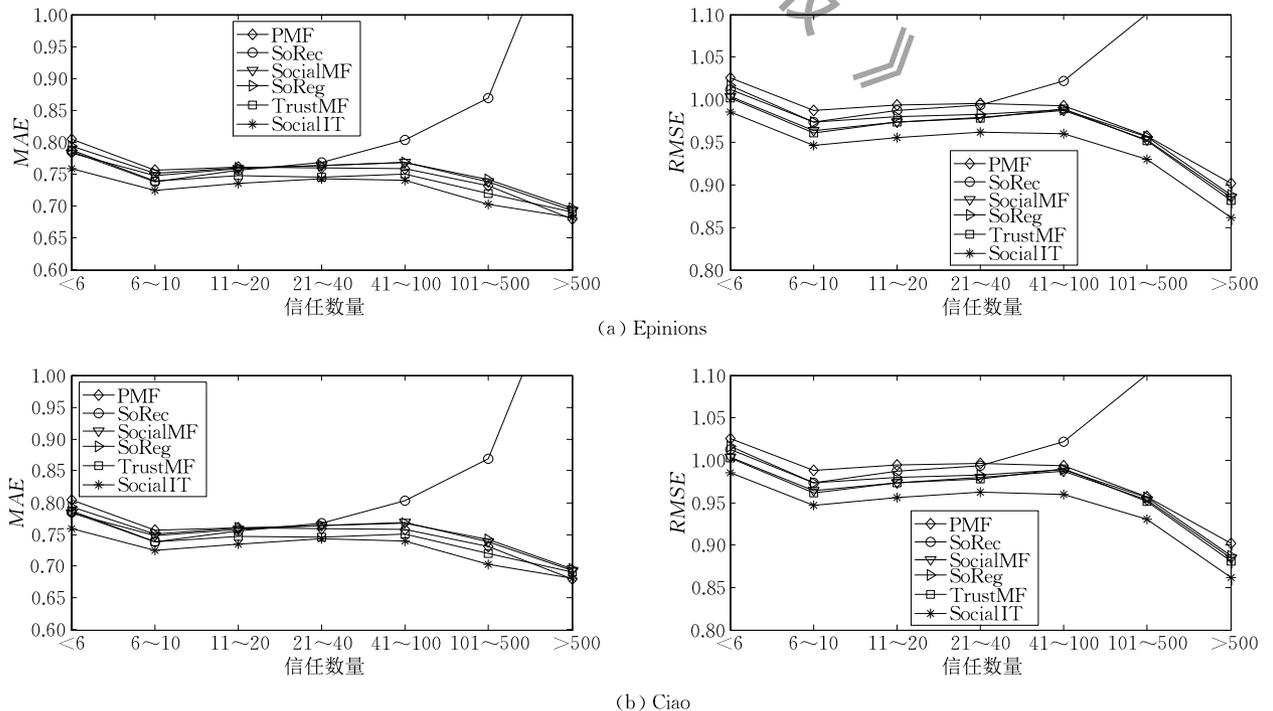


图 6 各算法在不同信任稀疏度情况下的精度

(3)随着信任数量的增加, SocialIT 的 MAE 精度与其它算法的差距有所减小, RMSE 仍然明显优于其它算法. 特别是在信任数量大于 100 时, 其它社会化推荐算法的 RMSE 精度与 PMF 相比差别非常小, 而 SocialIT 依然能够有效的提高算法精度. 对于稠密社交数据, 用户间的直接和间接交互信息更加丰富. 相比于其它算法, SocialIT 算法通过信任关系隐含相似度, 能够更为有效地挖掘和提取, 进而提

高推荐精度.

#### 实验 4. Top-N 项目推荐测试.

为了测试算法对于项目推荐任务的表现, 我们将本文算法 SocialIT 与 BPR 算法进行结合, 得到一个面向项目推荐任务的推荐算法, 并与其它算法进行对比. 在表 3 中展示了各算法在精确率 (Precision)、召回率 (Recall) 和排序指标 NDCG (Normalized Discounted Cumulative Gain) 上的表现.

表 3 在 Epinions 和 Ciao 数据集的 Top-N 项目推荐测试

| 算法           | Epinions |          |                 |          |                 |      | Ciao     |          |                  |                  |                 |       |
|--------------|----------|----------|-----------------|----------|-----------------|------|----------|----------|------------------|------------------|-----------------|-------|
|              | PMF      | BPR      | GBPR            | SBPR     | BPR+ SocialIT   | 提高/% | PMF      | BPR      | GBPR             | SBPR             | BPR+ SocialIT   | 提高/%  |
| Precision@5  | 0.007704 | 0.011383 | <b>0.011437</b> | 0.010388 | <b>0.011558</b> | 1.06 | 0.019619 | 0.023968 | <b>0.024999*</b> | 0.022064         | <b>0.025329</b> | 1.32  |
| Precision@10 | 0.006017 | 0.008993 | <b>0.009058</b> | 0.007971 | <b>0.009128</b> | 0.78 | 0.012283 | 0.019158 | <b>0.019932*</b> | 0.017557         | <b>0.020961</b> | 5.16  |
| Recall@5     | 0.006905 | 0.011025 | <b>0.011133</b> | 0.010206 | <b>0.011277</b> | 1.30 | 0.017029 | 0.020578 | 0.020382         | <b>0.020544*</b> | <b>0.022685</b> | 10.40 |
| Recall@10    | 0.010602 | 0.016731 | <b>0.016878</b> | 0.015143 | <b>0.016936</b> | 0.94 | 0.021918 | 0.032457 | <b>0.033383*</b> | 0.032184         | <b>0.036399</b> | 9.03  |
| NDCG@10      | 0.009350 | 0.015029 | <b>0.015046</b> | 0.013100 | <b>0.015127</b> | 0.81 | 0.021108 | 0.027824 | <b>0.028091*</b> | 0.027405         | <b>0.031064</b> | 10.60 |

从中我们可以看出:

(1)在这些指标上直接面向排序预测的算法 (BPR、GBPR、SBPR 及 BPR + SocialIT) 明显好于面向评分预测的算法 (PMF). 这是由于排序类算法学习的目标就是获得更好的排序预测, 这与 Top-N 项目预测任务的目标更加吻合. 而评分预测类算法则先预测评分, 再根据评分预测排序, 这样一个间接预测的过程使得该类算法的排序预测表现远不如排序类算法. 因此, 本文采用排序算法 BPR 与 SocialIT 算法相结合, 进而产生 Top-N 项目推荐结果.

(2)我们的算法 BPR + SocialIT 相对于其它利用社会关系信息的算法 (GBPR 和 SBPR) 有明显的提升, 尤其在 Ciao 数据集中提升明显. 这表明, 文中算法综合考虑评分相似和信任关系隐含相似度, 能够有效的提升 Top-N 项目推荐的准确率.

(3)各算法在 Epinions 数据集的表现远不如 Ciao 数据集. 这是由于 Epinions 数据集的候选项目数量远大于 Ciao (见表 1), 使得在该数据集进行 Top-N 项目推荐任务更为困难.

(4)BPR + SocialIT 算法在 Epinions 中的提高程度相比 Ciao 有所下降, 但仍然优于其它算法. 这说明, 文中算法在候选项目数量非常大的情况下 (大于 2 万), 对于算法质量的提高有一定的影响, 但是通过挖掘社交关系中的隐含信息, 也能够一定程度上提高排序推荐质量. 因此, 本算法更为适合于候选数量较小的情况.

## 6 结论与工作展望

为了更精确的度量用户与社交好友的相似度,

本文提出了一种基于信任关系隐含相似度的社会化推荐算法. 与现有算法相比, 本文首先对信任矩阵进行分解, 避免了皮尔逊相关或余弦距离必须要有共同对象的要求, 同时, 充分考虑信任关系隐含相似对用户相似度的直接影响, 提高了算法对于数据稀疏问题的精度. 最后, 本文共同考虑了评分相似和信任关系隐含相似对用户相似度的影响, 进一步提高了推荐精度.

在 Epinions 数据集上的实验表明, 本文算法能够有效提高推荐精度, 特别是, 对于稀疏用户, 能够取得明显的提高.

本文重点考虑了信任关系隐含相似度对社会化推荐算法的提升. 其它相关问题, 例如有向信任传递、时间漂移和产品特征等, 将在今后的工作中进一步探索.

## 参 考 文 献

- [1] Tang J, Hu X, Liu H. Social recommendation: A review. *Social Network Analysis and Mining*, 2013, 3(4): 1113-1133
- [2] Ambulkar H D, Pathan A. Recommender system challenges and methodologies in social network: Survey. *International Journal of Science and Research*, 2015, 4(11): 286-289
- [3] Yang X, Guo Y, Liu Y, et al. A survey of collaborative filtering based social recommender systems. *Computer Communications*, 2014, 41: 1-10
- [4] Koren Y, Bell R. *Advances in Collaborative Filtering*. *Recommender Systems Handbook*. Berlin, Germany: Springer, 2015: 77-118
- [5] Ning X, Karypis G. Recent advances in recommender systems

- and future directions//Proceedings of the International Conference on Pattern Recognition and Machine Intelligence. Warsaw, Poland, 2015: 3-9
- [6] Wang X, Wang Y. Improving content-based and hybrid music recommendation using deep learning//Proceedings of the ACM International Conference on Multimedia. Orlando, USA, 2014: 627-636
- [7] Wang C, Blei D M. Collaborative topic modeling for recommending scientific articles//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, USA, 2011: 448-456
- [8] Hu L, Cao J, Gu Z. Service discovery and recommendation in rough hierarchical granular computing//Proceedings of the 2010 IEEE Asia-Pacific Services Computing Conference (APSCC). Hangzhou, China, 2010: 191-198
- [9] Pan W, Xia S, Liu Z, et al. Mixed factorization for collaborative recommendation with heterogeneous explicit feedbacks. *Information Sciences*, 2016, 332: 84-93
- [10] Ji K, Sun R, Li X, et al. Improving matrix approximation for recommendation via a clustering-based reconstructive method. *Neurocomputing*, 2016, 173: 912-920
- [11] Sun Z. Exploiting item and user relationships for recommender systems//Proceedings of the International Conference on User Modeling, Adaptation, and Personalization. Dublin, Ireland, 2015: 397-402
- [12] Wang H, Wang N, Yeung D-Y. Collaborative deep learning for recommender systems//Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Sydney, Australia, 2015: 1235-1244
- [13] Guo G. Improving the performance of recommender systems by alleviating the data sparsity and cold start problems//Proceedings of the 23rd International Joint Conference on Artificial Intelligence. Beijing, China, 2013: 3217-3218
- [14] Chen C C, Wan Y-H, Chung M-C, et al. An effective recommendation method for cold start new users using trust and distrust networks. *Information Sciences*, 2013, 224: 19-36
- [15] Chen Ke-Han, Han Pan-Pan, Wu Jian. User clustering based social network recommendation. *Chinese Journal of Computers*, 2013, 36(2): 349-359(in Chinese)  
(陈克寒, 韩盼盼, 吴健. 基于用户聚类的异构社交网络推荐算法. *计算机学报*, 2013, 36(2): 349-359)
- [16] Huang Chuang-Guang, Yin Jian, Wang Jing, et al. Uncertain neighbors' collaborative filtering recommendation algorithm. *Chinese Journal of Computers*, 2010, 33(8): 1369-1377(in Chinese)  
(黄创光, 印鉴, 汪静等. 不确定近邻的协同过滤推荐算法. *计算机学报*, 2010, 33(8): 1369-1377)
- [17] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model//Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Las Vegas, USA, 2008: 426-434
- [18] Lu Yi-Jing, Cao Jian. Topic model based hierarchical service recommendation. *Journal of Chinese Computer Systems*, 2015, 36(11): 2457-2461(in Chinese)  
(陆怡静, 曹健. 基于主题模型的多层次服务推荐. *小型微型计算机系统*, 2015, 36(11): 2457-2461)
- [19] Tang J, Wu S, Sun J, et al. Cross-domain collaboration recommendation//Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Beijing, China, 2012: 1285-1293
- [20] Gao S, Luo H, Chen D, et al. Cross-domain recommendation via cluster-level latent factor model//Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Prague, Czech Republic, 2013: 161-176
- [21] Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback//Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. Montreal, Canada, 2009: 452-461
- [22] Li S, Kawale J, Fu Y. Deep collaborative filtering via marginalized denoising auto-encoder//Proceedings of the 24th ACM International Conference on Information and Knowledge Management. Melbourne, Australia, 2015: 811-820
- [23] Ma H, Yang H, Lyu M R, et al. SoRec: Social recommendation using probabilistic matrix factorization//Proceedings of the 17th ACM Conference on Information and Knowledge Management. Napa Valley, USA, 2008: 931-940
- [24] Ma H, King I, Lyu M R. Learning to recommend with social trust ensemble//Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval. Boston, USA, 2009: 203-210
- [25] Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks//Proceedings of the 4th ACM Conference on Recommender Systems. Barcelona, Spain, 2010: 135-142
- [26] McPherson M, Smith-Lovin L, Cook J M. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 2001, 27: 415-444
- [27] Lazer D, Pentland A S, Adamic L, et al. Life in the network: The coming age of computational social science. *Science*, 2009, 323(5915): 721
- [28] Chen C, Li D, Zhao Y, et al. WEMAREC: Accurate and scalable recommendation through weighted and ensemble matrix approximation//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. Santiago, Chile, 2015: 303-312
- [29] Moradi P, Ahmadian S. A reliability-based recommendation method to improve trust-aware recommender systems. *Expert Systems with Applications*, 2015, 42(21): 7386-7398
- [30] Fang H, Guo G, Zhang J. Multi-faceted trust and distrust prediction for recommender systems. *Decision Support Systems*, 2015, 71: 37-47
- [31] Guo G, Zhang J, Yorke-Smith N. Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems. *Knowledge-Based Systems*, 2015, 74: 14-27
- [32] Fazeli S, Loni B, Bellogin A, et al. Implicit vs. Explicit trust in social matrix factorization//Proceedings of the 8th

- ACM Conference on Recommender Systems. Foster City, USA, 2014: 317-320
- [33] Choo E, Yu T, Chi M, et al. Revealing and incorporating implicit communities to improve recommender systems// Proceedings of the 15th ACM Conference on Economics and Computation. Palo Alto, USA, 2014: 489-506
- [34] Yang B, Lei Y, Liu D, et al. Social collaborative filtering by trust//Proceedings of the 23rd International Joint Conference on Artificial Intelligence. Beijing, China, 2013: 2747-2753
- [35] Ma H, Zhou D, Liu C, et al. Recommender systems with social regularization//Proceedings of the 4th ACM International Conference on Web Search and Data Mining. Hong Kong, China, 2011: 287-296
- [36] Yao W, He J, Huang G, et al. Modeling dual role preferences for trust-aware recommendation//Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval. Gold Coast, Australia, 2014: 975-978
- [37] Wang M, Ma J. A novel recommendation approach based on users' weighted trust relations and the rating similarities. *Soft Computing*, 2015, 20(10): 3981-3990
- [38] Jamali M, Ester M. Trustwalker: A random walk model for combining trust-based and item-based recommendation// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, France, 2009: 397-406
- [39] Au Yeung C-M, Iwata T. Strength of social influence in trust networks in product review sites//Proceedings of the 4th ACM International Conference on Web Search and Data Mining. Hong Kong, China, 2011: 495-504
- [40] Guo Lei, Ma Jun, Chen Zhu-Min. Trust strength aware social recommendation method. *Journal of Computer Research and Development*, 2015, 50(9): 1805-1813(in Chinese)  
(郭磊, 马军, 陈竹敏. 一种信任关系强度敏感的社会化推荐算法. *计算机研究与发展*, 2015, 50(9): 1805-1813)
- [41] Guo Lei, Ma Jun, Chen Zhu-Min, et al. Incorporating item relations for social recommendation. *Chinese Journal of Computers*, 2014, 37(1): 219-228(in Chinese)  
(郭磊, 马军, 陈竹敏等. 一种结合推荐对象间关联关系的社会化推荐算法. *计算机学报*, 2014, 37(1): 219-228)
- [42] Wang T, Jin X, Ding X, et al. User interests imbalance exploration in social recommendation: A fitness adaptation// Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. Shanghai, China, 2014: 281-290
- [43] Mnih A, Salakhutdinov R. Probabilistic matrix factorization// Proceedings of the Advances in Neural Information Processing Systems. Vancouver, Canada, 2007: 1257-1264
- [44] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms//Proceedings of the 10th International Conference on World Wide Web. Hong Kong, China, 2001: 285-295
- [45] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer*, 2009, 42(8): 30-37
- [46] Castelfranchi C, Falcone R. Trust Theory: A Socio-Cognitive and Computational Model. New York, USA: John Wiley & Sons, 2010
- [47] Ma H. On measuring social friend interest similarities in recommender systems//Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval. Gold Coast, Australia, 2014: 465-474
- [48] Tang J, Gao H, Liu H. mTrust: Discerning multi-faceted trust in a connected world//Proceedings of the 5th ACM International Conference on Web Search and Data Mining. Seattle, USA, 2012: 93-102
- [49] Massa P, Avesani P. Trust-aware recommender systems// Proceedings of the 2007 ACM Conference on Recommender Systems. Minneapolis, USA, 2007: 17-24
- [50] Richardson M, Domingos P. Mining knowledge-sharing sites for viral marketing//Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Canada, 2002: 61-70
- [51] Shani G, Gunawardana A. Evaluating Recommendation Systems. *Recommender Systems Handbook*. Berlin, Germany: Springer, 2014: 257-297
- [52] Pan W, Chen L. GBPR: Group preference based Bayesian personalized ranking for one-class collaborative filtering// Proceedings of the 23rd International Joint Conference on Artificial Intelligence. Beijing, China, 2013: 2691-2697
- [53] Zhao T, McAuley J, King I. Leveraging social connections to improve personalized ranking for collaborative filtering// Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. Shanghai, China, 2014: 261-270



**PAN Yi-Teng**, born in 1988, Ph. D. candidate. His research interests include recommender system and deep learning.

**HE Fa-Zhi**, born in 1968, Ph. D., professor. His research interests include, CAD graphics, collaborative computing and data mining.

**YU Hai-Ping**, born in 1978, Ph. D. candidate. Her research interests include image processing and image segmentation.

## Background

With the rapid development of internet technology, information overloading is becoming a serious problem, which makes adverse impact on user experience. As one of the potential solutions, Recommendation system has been wildly used in many web applications and e-commerce web sides, aims to recommend items that fit user's preference by modeling user's historical behaviour. These algorithms always suffer from the cold start and data sparsity problem, which means that most users rates for only a few items and causing performance degradation. Utilizing users' social information is one of the efficient ways to alleviate this problem and attract increasing attention in recent years. However, most of them direct make use of the social network information on binary format, and ignore the fact that a user trusts in his friends with different degrees.

In this paper, we focus on the problem of estimating trust degrees between each pair of users for recommendation systems. Naturally, both ratings and social links could be used to answer this question. Based on the assumption that users may trust some social friends with similar ratings or social links in high degree, there are several methods have been proposed. To further obtains a more accurate

recommendation model, we proposed a new method to tackle this problem, and focus on two aspects: (1) Previous works estimated trust degrees in social links based on the common sets of social friends, but doesn't model the indirect relation between users in social networks; (2) Moreover, they didn't make full use of both ratings and social links for each pair of users, while they both show great potential for estimating trust degree. We first factored the matrix social links, and obtain the social latent vector of users which imply contain indirect relation between users. Based these latent vectors, we calculated the implicit similarity in trust, and then integrated it into traditional social recommendation algorithm. Finally, we achieved our accurate social recommendation model with implicit similarity in trust, named as SocialIT. The experiment on both rating prediction and item recommendation demonstrated that this model could achieve a more accurate result compared to previous works, especially for users with sparse rating data.

This work is supported by the National Natural Science Foundation of China (No. 61472289), the Natural Science Foundation of Hubei Province (No. 2015CFB254).