

基于注意力机制的规范化矩阵分解推荐算法*

张青博, 王斌, 崔宁宁, 宋晓旭, 秦婧

(东北大学 计算机科学与工程学院, 辽宁 沈阳 110189)

通讯作者: 王斌, E-mail: binwang@mail.neu.edu.cn



摘要: 近年来,矩阵分解(MF)技术因其有效性和简便性在推荐系统中得到广泛应用.但是,数据稀疏和冷启动问题导致 MF 学习到的用户特征向量不能准确地代表用户的偏好以及反映用户间的相似关系,影响了模型的性能.为了解决该问题,规范化矩阵分解(RMF)技术引起了研究者的关注.挖掘用户间可靠的相似关系,是 RMF 需要解决的问题.此外,MF 将目标用户特征向量和目标项目特征向量的内积作为目标用户对目标项目的评分,这种简单的线性关系忽略了用户对项目各个属性特征不同的关注度.如何分析用户对项目属性特征的关注度,获取用户更准确的偏好,仍然是一个挑战.针对上述问题,提出了基于注意力机制的规范化矩阵分解模型(ARMF).具体地,为了获取用户间可靠的相似关系解决数据稀疏和冷启动问题,该模型同时依据用户信任网络和评分记录构建用户-项目异构网络,并基于该异构网络挖掘用户间的相似关系;为了进一步提升模型性能,通过在 MF 中引入注意力机制,分析用户对项目各个属性特征不同的关注度来获取用户更准确的偏好.最后,在两个真实数据集上对比 ARMF 与现有工作,实验结果证明,ARMF 有更好的准确性和健壮性.

关键词: 推荐系统;矩阵分解;数据稀疏;冷启动;社交网络;注意力机制

中图法分类号: TP18

中文引用格式: 张青博,王斌,崔宁宁,宋晓旭,秦婧.基于注意力机制的规范化矩阵分解推荐算法.软件学报,2020,31(3): 778-793. <http://www.jos.org.cn/1000-9825/5913.htm>

英文引用格式: Zhang QB, Wang B, Cui NN, Song XX, Qin J. Attention-based regularized matrix factorization for recommendation. Ruan Jian Xue Bao/Journal of Software, 2020,31(3):778-793 (in Chinese). <http://www.jos.org.cn/1000-9825/5913.htm>

Attention-based Regularized Matrix Factorization for Recommendation

ZHANG Qing-Bo, WANG Bin, CUI Ning-Ning, SONG Xiao-Xu, QIN Jing

(School of Computer Science and Engineering, Northeastern University, Shenyang 110189, China)

Abstract: In recent years, matrix factorization (MF) has been exploited commonly in recommender system because of its capability and simplification. However, data sparsity and cold-start problems make the latent feature of users learned by MF cannot represent the users' preferences and the similarity relation among users exactly, which limits the performance of MF. To remedy it, the regularized matrix factorization (RMF) draws researchers' attention. And the problem demanding prompt solution in RMF is capturing the reliable similarity relation among users. Besides, MF simply regards the inner product between the latent features of both target user and target item as the score that target user may rate the target item, ignoring the user's different attentions on various features of the item. How to analyze the user's attention on item's features and capture more accurate preference of the user is still a challenge. To address these issues, a model is

* 基金项目: 国家重点研发计划(2018YFB1700404); 国家自然科学基金(U1736104, 61572122, 61532021); 中央高校基本科研专项资金(N171602003)

Foundation item: National Key Research and Development Program of China (2018YFB1700404); National Natural Science Foundation of China (U1736104, 61572122, 61532021); Fundamental Research Funds for the Central Universities (N171602003)

本文由人工智能赋能的数据管理、分析与系统专刊特约编辑李战怀教授、于戈教授和杨晓春教授推荐.

收稿时间: 2019-08-13; 修改时间: 2019-09-10; 采用时间: 2019-11-25; jos 在线出版时间: 2020-01-10

CNKI 网络优先出版: 2020-01-10 13:34:56, <http://kns.cnki.net/kcms/detail/11.2560.TP.20200110.1334.013.html>

put forward named attention-based regularized matrix factorization, abbreviated as ARMF. Specifically, to settle the problems of data sparsity and cold-start and obtain reliable similar relationships among users, the model builds a user-item heterogeneous network according to the social network and the rating record, and the similarities among users can be obtained based on it. Incorporating attention mechanism into MF allows us to analyze the attention of users on different item's features and capture more accurate preferences of users, which improves the precision of MF further. At last, the proposed model is compared with the state-of-the-art models on two real-world datasets and the result demonstrates the better precision and robustness of ARMF.

Key words: recommender system; matrix factorization; data sparsity; cold-start; social network; attention mechanism

随着互联网技术的发展以及智能手机的普及,信息超载问题也亟待解决.推荐系统^[1]作为解决信息超载问题的有效工具,已被成功应用于各个领域,包括电子商务、电影、音乐和基于位置的服务等^[2,3].推荐系统通过分析用户与系统交互的历史数据来获取用户偏好,使不同的用户进入系统时能够得到个性化推荐结果.由于推荐系统需要依赖于用户的历史数据,所以一般是作为一个应用存在于不同的网站中^[4].为方便系统获取用户偏好,大多数网站都允许用户对自己购买或体验过的项目(商品、电影、音乐等)评分,推荐系统则可以根据用户-项目评分矩阵(如图 1(a)所示,其中, u 代表用户, i 代表项目)分析用户的偏好,并为其推荐可能感兴趣的项目.

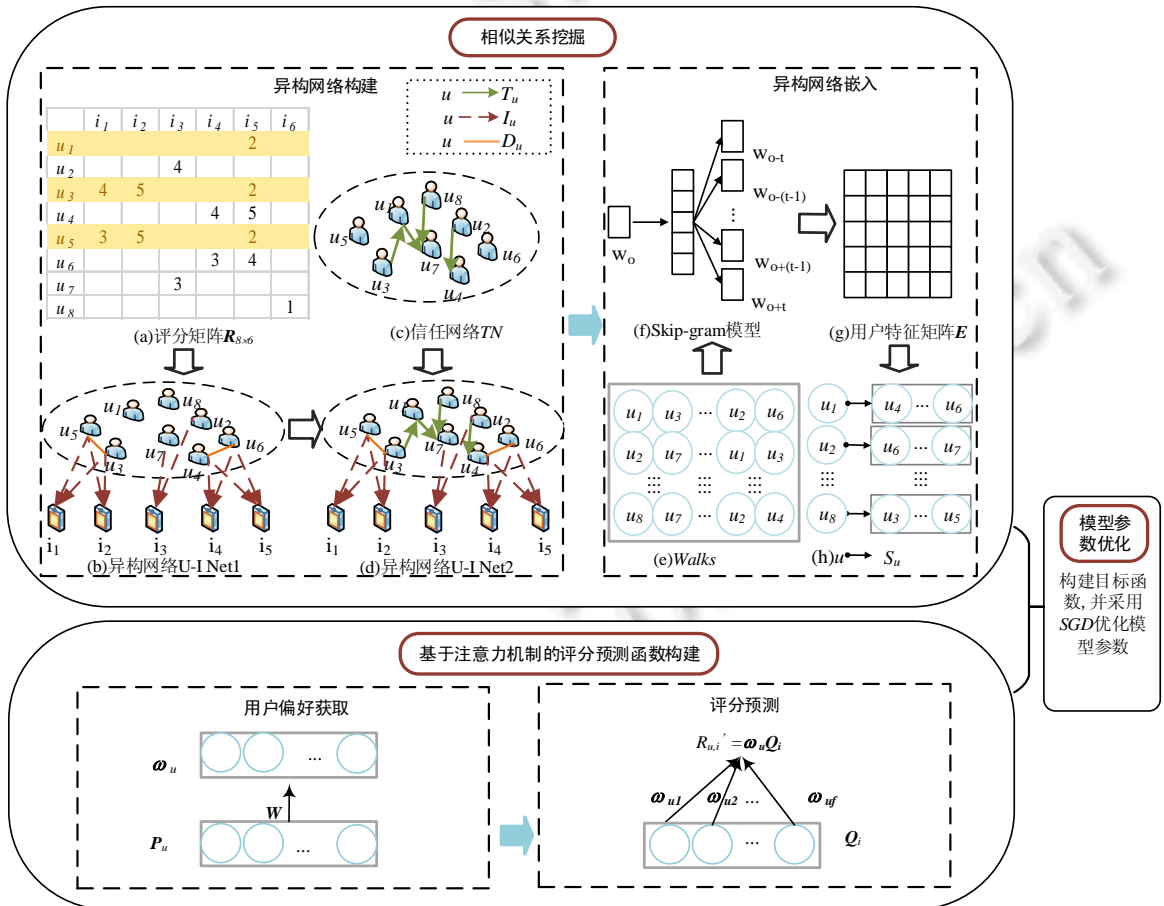


Fig.1 Architecture of our proposed attentive matrix factorization with regulation

图 1 基于注意力机制的规范化矩阵分解模型框架

近年来,推荐系统的实用价值使其在工业界和学术界都得到了广泛的研究.在众多推荐方法中,协同过滤算法^[5]应用最为普遍,其主要思想是:基于评分矩阵中的已有项来预测未知项,继而为用户推荐预测分值较高的项

目.协同过滤算法可以分为基于记忆(memory-based)的^[2,3]和基于模型的(model-based)^[6,7]两类算法,其中, memory-based 算法又包括基于用户的(user-based)^[8,9]和基于项目(item-based)的^[2,10].虽然协同过滤算法已经取得了巨大成功,但是依旧存在数据稀疏和冷启动问题:数据稀疏指由于用户只能体验系统中极小一部分项目,所以评分矩阵中存在很多未知项;冷启动问题则指对于评分记录很少甚至没有的用户,系统很难获取其偏好并作出推荐.

Memory-based 算法的核心在于寻找与目标用户(待推荐用户)或目标项目(待评分项目)有较高相似度的最近邻.现有 memory-based 算法中,用户(项目)间的相似度主要通过计算用户评分向量(项目得分向量)之间的皮尔逊相关系数(Pearson coefficient of correlation,简称 PCC)或余弦相似度(cosine similarity,简称 COS)获取^[9,10].这两种方式都需要参与计算的用户(项目)间存在公共评分项目(打分用户),数据稀疏导致由这两种方法获取的相似度无法准确代表用户(项目)间的相似关系.对于没有评分记录的冷启动用户(项目),系统无法基于此方法为其找到最近邻.Memory-based 算法在分析用户偏好时只用到了局部评分记录;model-based 算法则依据评分矩阵中所有已知项来训练预测模型,通过预测模型获取用户偏好和项目的属性特征,并利用学习到的参数预测评分矩阵中的未知项.在众多 model-based 算法中,矩阵分解(MF)技术^[6]因其有效性和简便性而被广泛采用.MF 模型认为,用户的偏好和项目的属性特征都可以用低维的特征向量表示,其中,用户特征向量每一维的元素代表了用户对项目某种属性特征的偏好程度,项目特征向量对应维的元素则代表该项目拥有此属性特征的程度,用户对项目的评分则用二者特征向量的内积表示^[11].显然,具有相似偏好用户的特征向量也应该是相似的.然而,数据稀疏导致 MF 学习到的用户特征向量不能准确代表用户的偏好,也无法反映用户间的相似关系,这同时降低了推荐准确度和模型在训练阶段的收敛速度.此外,MF 无法为冷启动用户学习特征向量获取其偏好.

为了提高 MF 的推荐准确度,带约束条件的规范化矩阵分解(RMF)技术引起了研究者的关注^[12,13].在模型的训练阶段,RMF 通过在 MF 目标函数的基础上增加约束项,使相似偏好用户的特征向量尽可能有较高的相似度.挖掘用户间的相似关系,并为用户找到与其有相似偏好的用户(相似朋友)是 RMF 需要解决的问题之一.为了解决数据稀疏和冷启动问题,当前大部分 RMF 算法都是基于信任关系的^[7,14].引入信任关系的 RMF 认为用户与其信任人有相似的偏好,直接将用户的信任人看作其相似朋友.Zhang 等人在文献[15]中指出:用户与其信任人不一定有相似的偏好,直接将用户的信任人看作其相似朋友可能会导致非个性化的推荐结果.他们提出的 CUNE 模型试图依据评分矩阵挖掘用户间的相似关系,但是该方法无法解决冷启动问题.如何利用评分矩阵和信任关系挖掘用户间可靠的相似关系,解决 MF 中的数据稀疏和冷启动问题,是本文重点解决的问题之一.

MF 中,用户对项目的评分是二者特征向量间的内积.He 等人^[16]认为,这种简单的线性方式无法捕捉用户和项目间复杂的交互关系.他们提出了基于神经网络的协同过滤模型框架(NCF),该框架利用多层感知机将用户和项目的特征向量以非线性方式结合预测评分,并通过实验证明了该框架的有效性.DMF 模型将评分矩阵作为深度神经网络的输入,以非线性方式学习用户和项目的特征向量,在神经网络的输出层,通过计算用户和项目特征向量间的余弦相似度预测评分^[17].虽然深度神经网络的引入能够有效提升 MF 的准确度,但是与此同时,模型的复杂度也大幅提升.如何在提高模型准确度的同时避免模型复杂度的大幅升高,是极具挑战的任务.

MF 将用户特征向量和项目特征向量所对应的维度元素相乘,并将乘积结果等权重相加后的和作为用户对项目的评分,忽略了用户对项目不同属性特征的关注度.我们希望通过学习用户对项目各个属性特征不同的关注度来获取用户更准确的偏好,进而提高模型性能.获取用户对项目属性关注度最直观的方法是为用户学习一个 f 维的关注度向量(f 是项目属性特征向量的维度),其中,第 k 维元素代表用户对项目第 k 个属性特征的关注度($k \in [0, f-1]$).这样会使模型多引入 $M * f$ 个参数(M 是系统中用户数量).由于系统中用户数量通常非常大,所以这种方式同样会使模型复杂度大幅度升高.如何在避免模型复杂度大幅提升的前提下分析用户对项目各个属性特征不同的关注度,获取用户更准确的偏好,是本文的另一个研究内容.

基于上述工作的问题,本文主要面临如下挑战:(1) 如何利用评分矩阵和信任关系挖掘用户间可靠的相似关系,解决 MF 中的数据稀疏和冷启动问题;(2) 如何在避免模型复杂度大幅提升的前提下,分析用户对项目属性特征的关注度,获取用户更准确的偏好.为了解决这些挑战,本文提出融合信任关系和评分矩阵的基于注意力

机制^[18]的规范化矩阵分解模型(ARMF),模型框架如图 1 所示.该模型主要包括 3 部分.

- 第 1 部分是用户间相似关系的挖掘:首先,依据评分矩阵和信任关系构建异构网络;然后,利用网络嵌入技术^[19]挖掘用户间的相似关系;
- 第 2 部分是基于注意力机制的评分预测函数构建:通过公共权重矩阵,以双线性方式在 MF 中引入注意力机制,分析用户对项目各个属性的关注度,获取用户更准确的偏好;
- 第 3 部分是模型参数优化:以准确地重构评分矩阵同时使有相似偏好的用户具有相似的特征向量为目的构建目标函数,并采用随机梯度下降法(SGD)^[20]优化模型参数.

本文主要贡献如下.

- (1) 为了挖掘用户间可靠的相似关系,提出了伪相似朋友的概念用于构建异构网络,并在网络嵌入阶段提出了伪相似传播机制和一种新的随机游走算法 VDWalk;
- (2) 为了分析用户对项目各个属性特征不同的关注度,获取用户更准确的偏好,同时保证模型复杂度不会大幅提升,首次以双线性方式将注意力机制引入 MF 中;
- (3) 在真实数据集 FilmTrust^[21]和 CiaoDVD^[22]上进行了大量实验,实验结果表明,与现有的矩阵分解模型相比,ARMF 模型具有更好的推荐准确性和健壮性.

1 相关工作

1.1 传统协同过滤算法

协同过滤算法包括 memory-based 和 model-based 两类算法.

Memory-based 算法的关键在于最近邻的查找过程.

- 文献[23]计算目标用户与其他用户间的属性(年龄、性别和职业)相似度和评分向量相似度并将二者线性相加的和作为最终的相似度,然后将与目标用户有较高相似度的用户定义为其最近邻;
- 文献[24]在找用户的最近邻时考虑了地理位置的影响:首先,依据评分矩阵得到目标用户与其他用户评分向量之间的皮尔逊相关系数;然后找出与目标用户在同一区域内的用户,并将他们与目标用户间的相似度增加一个常数项,以此提高了在依据给定相似度阈值选择目标用户最近邻时这些用户的被选择权;
- 文献[3]认为依据评分矩阵计算音乐间的相似度会存在数据稀疏问题,所以提出了依据音乐特征间的相似度找最近邻的算法.在找到目标音乐最近邻后,依据目标用户对其最近邻的偏好预测对目标音乐的评分.

由于 memory-based 算法需要计算系统中所有用户(项目)间的相似度,导致该类算法的扩展性较差,所以 model-based 算法逐渐成为研究重点.Model-based 算法包括基于图模型的^[25]、基于聚类的^[26]和基于 MF 的^[7]等,其中,MF 是目前应用最为普遍的算法.假设系统中有 M 个用户(用户集合用 U 表示)和 N 个项目(项目集合用 I 表示),MF 的主要思想是:利用评分矩阵 $R_{M \times N}$ 中的已有评分记录学习能够代表用户偏好和项目属性特征的用户特征 $P \in R^{f \times M}$ 和项目特征 $Q \in R^{f \times N}$,用户对项目的评分则用二者特征向量的内积表示.为了尽可能准确地重构评分矩阵并预测未知项,MF 通过最小化公式(1)中所示的目标函数并采用 SGD 学习模型参数 P 和 Q :

$$L(R, P, Q) = \frac{1}{2} \sum_{u=1}^m \sum_{i=1}^n I_{ui} \left(\sum_{j=0}^{f-1} P_u^j * Q_i^j - R_{u,i} \right)^2 + \frac{\lambda_1}{2} \|P\|_F^2 + \frac{\lambda_2}{2} \|Q\|_F^2 \quad (1)$$

其中, $I_{u,i}$ 是一个指示函数,如果用户 u 对项目 i 进行了评分, $I_{u,i}$ 等于 1,否则为 0; P_u^j 表示用户 u 对项目第 j 个属性特征的偏好程度; Q_i^j 表示项目 i 拥有第 j 个属性特征的程度; λ_1 和 λ_2 是正则项系数,引入正则项是为了防止模型过拟合; $\|\cdot\|_F^2$ 代表向量的二范式.RMF 在 MF 目标函数的基础上增加约束项,使用户的特征向量与其相似朋友的特征向量之间的距离尽可能小,以此提升了模型性能.挖掘用户间的相似关系为用户找到其相似朋友,是 RMF 的任务之一.CUNE^[15]通过将已评分项目有交集的用户连接的方式构建用户同质网络,在利用网络嵌入技术学

习网络中用户的隐向量后,依据用户隐向量间的相似度挖掘用户间的相似关系.然而,数据稀疏和冷启动问题导致系统中只有小部分用户能够连接到网络,稀疏的网络使得通过网络嵌入得到的用户隐向量反映不出用户间可靠的相似关系.如何挖掘用户间可靠的相似关系,是 RMF 需要解决的问题.

1.2 引入信任关系的协同过滤算法

信任关系作为评分矩阵的辅助信息,被证明能够有效缓解协同过滤算法中的数据稀疏和冷启动问题^[8,9]. Guo 等人^[9]提出了融合信任关系的 user-based 算法——Merge:该算法依据用户信任人的评分记录预测用户对未评分项目的评分,以此来对评分矩阵中的未知项进行填补,用户之间的相似度依据新的评分矩阵得到.实验结果证明,该方法得到的用户间相似关系更可靠,推荐结果的准确度也更高.文献^[8]提出了 MoleTrust 算法:在信任网络中,通过深度优先遍历进行信任传播,并计算用户之间的信任度.在利用目标用户信任人的评分记录预测目标用户的偏好时,信任人对目标用户偏好的影响程度取决于目标用户对其的信任度.与 MoleTrust 相似的 TidalTrust 算法^[27]则是利用广度优先遍历法推断和计算用户间信任值.Jamali 和 Ester 提出了融合 item-based 和 trust-based 的 TrustWalker 随机游走算法^[2],该算法分别连接用户与其评分项目、信任人构建用户-项目异构网络,在预测目标用户对目标项目的评分时,从目标用户出发在网络中随机游走,为在准确度和效率间取得平衡,在游走过程中,按条件要么返回目标用户对与目标项目相似项目的评分,要么返回目标用户信任人对目标项目的评分.

Ma 等人^[28]认为,用户对项目的评分不仅依赖于自己的偏好,还容易受其信任人的影响.他们提出了基于 MF 的 STE 模型,STE 将用户依据个人偏好对项目的评分与用户信任人对项目的评分线性相加作为用户对项目的最终评分.该模型只考虑了信任人对用户评分的影响,忽略了信任人对用户特征向量的影响.引入信任关系的 RMF 基于用户与其信任人有相似偏好的假设,将用户的信任人看作其相似朋友,所以信任人直接影响着用户的特征向量.其中:SocialRec^[12]通过共享用户特征向量矩阵,同时分解评分矩阵和信任矩阵来学习用户特征向量,这样,用户的特征向量不仅取决于自己的评分,还受到其信任人的影响;SocialMF^[13]认为用户与其信任人有相似的特征向量,特征向量间的相似度取决于用户对信任人的信任度;SocialReg^[14]计算用户与其信任人二者评分向量之间的相似度来决定他们特征向量之间的相似度;Parvina 等人^[29]在学习用户特征向量时,同时考虑用户信任人和非信任人的影响——用户特征向量与其信任人特征向量间的距离应尽可能近,同时与其非信任人特征向量间的距离应尽可能远.这些算法都认为用户与其所有信任人有相似的偏好,同时忽略了用户与其他用户间的相似关系.然而,用户在定义信任人时会受到复杂因素的影响,用户与其信任人不一定有相似的偏好.为找到与用户有相似偏好的信任人,Li 等人^[7]提出了信任关联度的概念.他们认为,只有与目标用户有相似评分记录的信任人是目标用户的相似朋友.同时还引入了信任传播机制解决数据稀疏和冷启动问题.虽然引入了信任传播机制,但是目标用户的相似朋友候选集依旧很小,可能会丢失真实存在的相似关系.与现有引入信任关系的 RMF 不同的是,本文将用户间的信任关系作为评分矩阵的辅助信息,挖掘系统中所有用户间的相似关系,并为用户找到其相似朋友.

2 基于注意力机制的规范化矩阵分解模型

为了能够找到用户之间可靠的相似关系并分析出用户复杂的偏好,本文提出了基于注意力机制的规范化矩阵分解模型——ARMF.该模型的框架如图 1 所示,整个框架由 3 部分构成.

- (1) 用户间相似关系的挖掘,主要包含两部分:
 - ① 异构网络构建:依据评分矩阵和用户间信任关系构建用户-项目异构网络(详见第 2.3 节);
 - ② 异构网络嵌入:利用网络嵌入技术学习步骤①所构建的网络中用户的隐向量,通过计算用户隐向量间的余弦相似度,挖掘用户间的相似关系(详见第 2.4 节);
- (2) 基于注意力机制的评分预测:以双线性方式在 MF 中引入注意力机制,从用户的历史数据中获取用户更准确的偏好(详见第 2.5 节);
- (3) 模型参数优化:依据步骤(1)中挖掘到的相似关系和步骤(2)中对评分的预测,构造目标函数并采用

SGD 优化模型参数(详见第 2.6 节).ARMF 的执行过程可总结如图 2 所示.

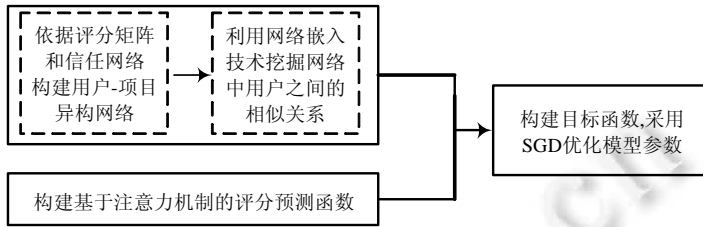


Fig.2 Execution of ARMF
图 2 ARMF 的执行过程

2.1 问题描述

在推荐系统中,假设有 M 个用户 $U=\{u_1, u_2, \dots, u_M\}$ 和 N 个项目 $I=\{i_1, i_2, \dots, i_N\}$, 用户通过对体验过的项目评分, 系统会生成一个 M 行 N 列的评分矩阵 $R_{M \times N}$ (图 1(a)评分矩阵中, $M=8, N=6$), 其中, 第 u 行 i 列的 $R_{u,i}$ 表示用户 u 对项目 i 的评分(评分通常是 1~5 的整数). 在本文中, 用 I_u 表示用户 u 评分大于 $\sigma(\sigma \in [0, 5])$ 的项目集合; C_u 表示和用户 u 共同评分项目个数不少于 $\delta(\delta \in \mathbb{Z})$ 的用户集合; D_u 表示和用户 u 至少有一个公共评分项目, 且对项目评分都大于 $\sigma(\sigma \in [0, 5])$ 的用户集合.

在引入信任关系的推荐系统中, 用户可以定义自己的信任人, $t_{u,v}$ 代表用户 u 对用户 v 的信任度. 在大多数系统中, $t_{u,v} \in \{0, 1\}$ (0 表示不信任, 1 表示信任), 本文用 $T_u = \{v | t_{u,v} = 1\}$ 表示用户 u 的信任人集合. 依据用户之间的信任关系, 可以构建对应的信任网络(TN), 如图 1(c)所示.

当 $\delta=1$ 和 $\sigma=2$ 时, 图 1(a)中每个用户 u 的 I_u, C_u, D_u 以及 T_u 见例 1. 在第 2.3 小节异构网络构建和第 2.4 小节异构网络嵌入阶段, 会以例 1 为基础介绍模型每个模块的实现步骤, 以便理解模型原理.

例 1:

用户	I_u	C_u	D_u	T_u
u_1	\emptyset	\emptyset	\emptyset	$\{u_7\}$
u_2	$\{i_3\}$	\emptyset	$\{u_7\}$	$\{u_4\}$
u_3	$\{i_1, i_2\}$	$\{u_5\}$	$\{u_5\}$	$\{u_1\}$
u_4	$\{i_4, i_5\}$	$\{u_6\}$	$\{u_6\}$	\emptyset
u_5	$\{i_1, i_2\}$	$\{u_3\}$	$\{u_3\}$	\emptyset
u_6	$\{i_4, i_5\}$	$\{u_4\}$	$\{u_4\}$	\emptyset
u_7	$\{i_3\}$	\emptyset	$\{u_2\}$	\emptyset
u_8	\emptyset	\emptyset	\emptyset	$\{u_7\}$

本文利用 $R_{M \times N}$ 和 TN 挖掘用户间可靠的相似关系, 并分析用户对项目各个属性特征不同的关注度. 在尽可能准确地预测评分矩阵中未知项的同时, 保证有相似偏好用户的特征向量间有较高的相似度.

2.2 引入信任关系的异构网络

推荐系统中, 用户和用户之间的信任关系以及用户和项目之间的交互关系可以通过不同的方式构建用户-项目异构网络展现出来. 最直接的方式就是用户与其已评分项目连接, 同时与其信任人连接, 并假设用户与其信任人有相似偏好^[2]. 由于用户与其信任人不一定完全相似, 直接将信任关系看作相似关系可能导致非个性化的推荐结果. 因此, 本文将用户的信任人看作其伪相似朋友(伪相似朋友定义详见第 2.3 节), 而非相似朋友, 来挖掘用户间更可靠的相似关系. 此外, 与文献[2]不同的是: 本文中, 用户 u 只与 I_u 中的项目连接, 项目在 network 中只是起到媒介作用, 将可能相似的用户连接起来. 比如, 例 1 中的 i_3 可以使 u_2 与 u_7 相互连接. 本文通过将用户与其伪相似朋友连接的方式构建异构网络, 并基于若两个用户有相似的伪相似朋友则他们有相似偏好, 即有相似隐向量的假设, 利用网络嵌入技术挖掘用户间可靠的相似关系. 下面将在第 2.3 节和第 2.4 节分别介绍如何利用评分矩阵和信任网络构建异构网络和如何通过网络嵌入获取用户间的相似关系.

2.3 引入信任关系的异构网络构建

为了构建稠密的用户-项目异构网络,提出了伪相似朋友的概念.将可能与用户 u 有相似偏好的用户定义为其伪相似朋友,同时假设:如果两个用户有较多公共的伪相似朋友,则他们一定有相似偏好.具体地,用户 u 的伪相似朋友 F_u 定义如下.

定义 1(伪相似朋友 F_u).

(1) 对称伪相似

i. $u \in F_v$ 且 $v \in F_u | v \in C_u$;

ii. $u \in F_v$ 且 $v \in F_u | v \in D_u$;

(2) 非对称伪相似

$v \in F_u | v \in T_u$.

基于定义 1,首先根据评分矩阵找到每个用户 u 的 I_u, C_u, D_u 以及 T_u (例 1),然后直接连接每个用户 u 与其伪相似朋友 $v(v \in C_u), u$ 和 D_u 中的伪相似朋友则通过 I_u 中的项目间接连接,这样得到图 1(b)中的异构网络 U-I Net1.由于数据稀疏问题导致大部分用户有很少甚至没有伪相似朋友,所以 U-I Net1 中会存在许多孤立的节点(比如 u_1 和 u_8)和处于小圈中的用户节点(比如 u_5 和 u_3).如果直接在稀疏的 U-I Net1 上学习用户的隐向量,不仅无法为孤立的节点学习其隐向量,也会使得为处于小圈中的节点学习到的隐向量不准确,最终会导致依据隐向量挖掘的用户相似关系不可靠.

为了解决数据稀疏问题,ARMF 在评分矩阵的基础上融入了用户间的信任关系,认为用户的信任人可以看做其伪相似朋友.通过在 U-I Net1 的基础上连接用户与其信任人,可得到图 1(d)中的异构网络 U-I Net2.可以看出:在引入信任关系后,孤立用户 u_1 和 u_8 都通过 u_7 连接到网络中,处于小圈中的其他用户也可以连接到更多其他伪相似朋友,这就使得通过网络嵌入挖掘到的相似关系更加可靠.此外,通过在用户和信任人之间添加连边的操作,可以将携带信任信息的新用户加入网络中,并为其找到相似朋友,成功解决了冷启动问题.

2.4 异构网络嵌入

为了获取 U-I Net2 中用户间可靠的相似关系,提出了伪相似传播机制,并设计了 VDWalk 随机游走算法.在利用网络嵌入学习用户隐向量时,我们做了如下假设:如果两个用户有较多公共的伪相似朋友,则他们有相似偏好,即有相似的隐向量.数据稀疏问题会导致每个用户的伪相似朋友数量很少,虽然在构建异构网络时引入了用户间的信任关系,但是由于信任关系也存在冷启动问题,即有很少评分记录的用户通常也有非常少的信任人^[15],所以用户的伪相似朋友集合间很难有比较大的交集.鉴于此,我们提出了伪相似传播机制,伪相似传播概念见定义 2.

定义 2(伪相似传播). $u \in F_v$ 且 $v \in F_r$, 则 $u \in F_r$.

伪相似传播机制使每个用户能够拥有更多的伪相似朋友,进而可以挖掘到用户间更可靠的相似关系.由于引入了伪相似传播机制,网络 U-I Net2 中每个用户附近的用户都可以看做其伪相似朋友.基于如果两个用户有相似的伪相似朋友则他们有相似的偏好,即有相似的隐向量的假设,在 U-I Net2 上以每个非孤立用户(由于网络中的孤立节点没有邻居节点,没有办法为他们采集到路径,不能通过网络嵌入方法学习他们的隐向量,所以下文提到的 U-I Net2 中的用户均指非孤立用户)为起始节点选取 ρ 条长度为 l 的路径,把这些路径放入 Skip-gram 模型^[30]学习用户的隐向量.

为了能够为每个用户节点采集到的 ρ 条路径尽可能不同,我们提出了基于 DeepWalk^[31]的一种变形随机游走算法——VDWalk.

算法 1. VDWalk.

输入:U-I Net2;路径条数: ρ ;路径长度: l ;

输出:以 U-I Net2 中每个用户为起始节点的 ρ 条长 l 的路径.

1. 初始化路径集合 $Walks$ 为空;

```

2. For U-I Net2 每个非孤立用户节点  $u$ 
3.   初始化 U-I Net2 中所有节点的邻居状态为‘False’;
4.   While 路径条数 $<\rho$ 
5.     初始化当前节点路径  $path=[u]$ ,父节点  $last=u$ ;
6.     While 路径长度 $<l$ 
7.       下一个节点  $next$ =随机从  $last$  未被访问的用户邻居中选择一个
8.       改变  $next$  的状态为‘True’;
9.       If  $last$  所有用户邻居的状态都为‘True’
10.        If  $last$  有项目邻居
11.          下一个节点  $next$ =随机从  $last$  的项目邻居中选择一个;
12.        Else 下一个节点  $next$ =随机从  $last$  的用户邻居中选择一个;
13.        If 下一个节点  $next$  是用户节点
14.          将  $next$  加入  $path$ , $path$  长度加 1;
15.     将长度为  $l$  的路径  $path$  加入  $Walks$ ,路径条数加 1;
16. Return  $Walks$ 

```

DeepWalk 在每次采集下一个节点时会在当前节点邻居节点中随机选择一个,这样可能导致一些节点被重复选择,而一些节点从不被选择,也可能为同一个节点采集到相同的路径,从而降低了通过网络嵌入学习到的节点隐向量的准确度.为避免这些情况的出现,VDWalk 定义不同的邻居节点有不同的优先选择权,在每次采集下一个节点时,优先选择当前节点的邻居节点中优先级较高的节点.邻居节点的优先选择权定义如下.

定义 3(优先选择权).

- (1) $u>v|u\in U,v\in U,state(u)=False,state(v)=True$
- (2) $u=v|u\in U,v\in U,state(u)=True,state(v)=True$
- (3) $u=v|u\in U,v\in U,state(u)=False,state(v)=False$
- (4) $u>i|u\in U,i\in I,state(u)=False,state(i)=False$
- (5) $u>i|u\in U,i\in I,state(u)=True,state(i)=True$
- (6) $i>u|u\in U,i\in I,state(u)=True,state(i)=False$
- (7) $i=j|i\in I,j\in I$

其中: $a>b$ 表示 a 的优先选择权大于 b , $a=b$ 表示 a 的优先选择权等于 b , $state(a)=False$ 表示 a 节点未被访问过, $state(a)=True$ 表示 a 节点已被访问过.

VDWalk 在为每个用户节点采集路径时,会初始化 U-I Net2 中所有节点的邻居节点状态为‘False’(算法 1 第 1 行~第 3 行),在选择下一个节点时,会根据定义 3 优先选择父节点邻居节点中优先级较高的节点(第 6 行~第 11 行).基于用户 u 和 T_u 和 C_u 中的用户比 D_u 中的用户更为相似的假设,父节点邻居节点中的用户节点比项目节点有较高的选择优先权(第 7 行~第 10 行),除非所有的用户节点已经被访问过,才会选择项目节点(第 8 行~第 10 行).项目节点在 VDWalk 中只是起到媒介作用,使能够为用户 u 采集到 D_u 中的伪相似朋友,所以它不被放入路径中(第 12 行、第 13 行).这样,路径中除了用户 u 本身之外的其他节点都是其伪相似朋友,记作 N_u .例 2 是以 U-I Net2 中以 u_3 为起始节点选取第 1 条长度为 5 的路径时的选择过程.

例 2:下图中的黄色节点表示路径中每一步的节点选择过程:以 u_3 为源节点,最初 u_3 的邻居节点中既有项目节点 i_1 和 i_2 ,又有用户节点 u_1 和 u_5 ,因为用户节点优先级高于项目节点,并且此时两个用户节点都未被访问过,所以从 u_1 和 u_5 中随机选择一个作为子节点.如果选择了 u_1 , u_1 的邻居节点只有 u_7 , u_7 作为下一个子节点,然后 u_7 通过 i_3 选择 u_2 , u_2 的信任人 u_4 作为终止节点,最终的路径为 (u_3,u_1,u_7,u_2,u_4) , $N_{u_3}=\{u_1,u_7,u_2,u_4\}$.

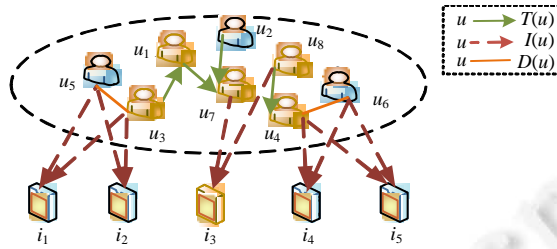


Fig.3 Example of VDWalk
图3 VDWalk 算法例子

Skip-gram 模型针对句中给定单词,最大化窗口大小为 t 内周围单词的共现概率,用函数 Φ 将该单词映射到一个 d 维的低维空间中.其形式化地定义为:若给定句子 $S=(w_1, w_2, \dots, w_o, w_{o+1}, \dots)$ 中的单词 w_o ,Skip-gram 通过最小化目标函数 L 来学习 w_o 的低维隐向量:

$$L = -\text{lbPr}(\{w_{o-t}, \dots, w_{o+t}\} | \Phi(w_o)) \tag{2}$$

从 Skip-gram 的目标函数可以看出,Skip-gram 模型将会对具有相似上下文的单词学习到相似的隐向量.因为我们提出了类似的假设,即:如果两个用户有相似的伪相似朋友,他们就有相似的隐向量,所以可以利用 Skip-gram 模型学习 U-I Net2 中用户的隐向量,并以此来获取用户之间隐藏的相似关系,此时目标函数为

$$L = -\text{lbPr}(N_u | \Phi(\mathbf{E}_u)) \tag{3}$$

将在 U-I Net2 上收集到的所有路径放入 Skip-gram 模型中,学习用户的 d 维隐向量 \mathbf{E} ,根据公式(4)计算用户隐向量之间的余弦相似度,取和用户 u 有最大相似度的前 K 个用户作为他的相似朋友,记为 S_u :

$$\text{Sim}(u, v) = \frac{\sum_{j=0}^{d-1} \mathbf{E}_u^j \cdot \mathbf{E}_v^j}{\sqrt{\sum_{j=0}^{d-1} (\mathbf{E}_u^j)^2} \sqrt{\sum_{j=0}^{d-1} (\mathbf{E}_v^j)^2}} \tag{4}$$

其中, \mathbf{E}_u^j 代表用户 u 隐向量的第 j 维.

2.5 基于注意力机制的评分预测

ARMF 希望分析用户对项目不同属性特征的关注度,获取用户更准确的偏好.最直观的做法就是为每个用户学习一个关注度向量,此时,用户 u 对项目 i 的评分预测函数见公式(5):

$$R'_{u,i} = \sum_{j=0}^{f-1} \varepsilon_u^j \times P_u^j \times Q_i^j \tag{5}$$

其中, ε_u^j 表示用户 u 对项目第 j 个特征的关注度.但为系统中每个用户学习 ε_u 需多引入 $M \times f$ 个参数,这样不仅使模型复杂度升高,也不能保证相似用户对项目属性特征有相似的关注度.鉴于此,引入用户共享矩阵 $\mathbf{W} \in \mathbf{R}^{M \times f}$,用户新的偏好矩阵 ω 即可通过 $\mathbf{P}^T \mathbf{W}$ 得到,这样只需多引入 f^2 个参数.由于 f 一般比较小,所以模型复杂度的提升特别小.此时, $R'_{u,i}$ 可通过公式(6)得到:

$$R'_{u,i} = \mathbf{P}_u^T \mathbf{W} \mathbf{Q}_i \tag{6}$$

虽然只是用了双线性计算方式,却能挖掘出用户项目间复杂的交互关系,获取用户更准确的偏好.同时,由于所有用户共享矩阵 \mathbf{W} ,所以保证了相似用户对项目属性特征也有相似的关注度.

2.6 模型参数优化

在模型训练阶段,首先随机初始化模型参数,并以保证评分预测准确度的同时使得相似用户有相似的特征向量为目的,构建公式(7)中的目标函数:

$$L(R, \mathbf{P}, \mathbf{Q}, \mathbf{W}) = \frac{1}{2} \sum_{u=1}^m \sum_{i=1}^n I_{ui} (R'_{u,i} - R_{u,i})^2 + \frac{\lambda}{2} \sum_{u=1}^m \left\| \mathbf{P}_u - \frac{1}{|S_u|} \sum_{v \in S_u} \mathbf{P}_v \right\|_F^2 + \frac{\lambda_1}{2} \|\mathbf{P}\|_F^2 + \frac{\lambda_2}{2} \|\mathbf{Q}\|_F^2 + \frac{\lambda_3}{2} \|\mathbf{W}\|_F^2 \quad (7)$$

其中, λ 是规范化函数的系数. 规范化函数约束用户的特征向量与其相似朋友特征向量的平均值距离尽可能小, 以此来保证用户与其相似朋友有相似的特征向量, λ_1, λ_2 和 λ_3 是正则项系数. 目标函数构建后, 采用 SGD 对模型参数进行优化, $\mathbf{P}_u, \mathbf{Q}_i$ 和 \mathbf{W} 的一阶求导公式如下所示:

$$\left. \begin{aligned} \frac{\partial L}{\partial \mathbf{P}_u} &= (R'_{u,i} - R_{u,i}) \mathbf{W} \mathbf{Q}_i + \lambda \left(\mathbf{P}_u - \frac{1}{|S_u|} \sum_{v \in S_u} \mathbf{P}_v \right) + \lambda_1 \mathbf{P}_u - \sum_{\{v|v \in S_u\}} \frac{\lambda}{|S_v|} \left(\mathbf{P}_v - \frac{1}{|S_v|} \sum_{w \in S_v} \mathbf{P}_w \right), \\ \frac{\partial L}{\partial \mathbf{Q}_i} &= (R'_{u,i} - R_{u,i}) \mathbf{P}_u^T \mathbf{W} + \lambda_2 \mathbf{Q}_i, \\ \frac{\partial L}{\partial \mathbf{W}} &= (R'_{u,i} - R_{u,i}) \mathbf{P}_u^T \mathbf{Q}_i + \lambda_3 \mathbf{W} \end{aligned} \right\} \quad (8)$$

3 实验

3.1 数据集

实验采用文献[21,22]中的 FilmTrust 和 CiaoDVD 这两个真实数据集验证 ARMF 模型和对比模型的性能, 两个数据集的详细信息见表 1.

Table 1 Experimental datasets statistics

表 1 实验数据集统计数据

名称	Filmtrust	CiaoDVD
用户	1 508	17 615
项目	2 071	16 121
评分	35 497	72 665
信任关系	1 632	22 483

为了验证各个模型的健壮性, 每次测试时, 预先将测试数据集划分为 Warm 集以及 Cold 集, 这主要根据训练数据集中每个用户的评分记录所得. 具体过程如下: 训练数据集中包含了用户及其评分记录, 若训练集中单个用户的评分记录大于等于 5, 则该用户对应测试集中的评分记录为 Warm 集; 若训练集中单个用户的评分记录小于 5, 那么该用户对应测试集中的评分记录为 Cold 集.

3.2 评估方法

为了评估各个模型的准确性和健壮性, 本文采用了 5 交叉验证法^[32]. 并在每次验证时采用均方根误差 (root mean square error, 简称 RMSE) 和平均绝对误差 (mean absolute error, 简称 MAE) 来对各模型预测评分的准确度性进行评估. 假设测试集中的评分记录个数为 X , RMSE 和 MAE 计算方式见公式(9)和公式(10).

$$RMSE = \sqrt{\frac{(R'_{u,i} - R_{u,i})^2}{X}} \quad (9)$$

$$MAE = \frac{|R'_{u,i} - R_{u,i}|}{X} \quad (10)$$

模型在训练阶段的收敛速度越快, 系统就可以更快地依据训练结果做出响应. 由于所有模型的目标函数都是非凸函数, 只有局部最优解^[11], 所以为了比较各个模型在训练阶段的收敛速度, 在本文中约定: 当模型的准确度不再提升时, 认为模型收敛达到局部最优解. 模型参数优化的具体步骤见算法 2.

算法 2. 模型参数优化.

输入: 随机初始化的模型参数;

输出: 优化的模型参数.

1. 根据模型的目标函数在训练集上以给定学习率采用 SGD 更新参数;

2. 在测试集上,根据公式(10)和公式(11)分别计算 RMSE 和 MAE;
3. $Pre_RMSE=RMSE$;
4. **While** $Pre_PMSE>RMSE$
5. 执行第 1 行~第 3 行

由于模型的最终目的是使预测误差最小,所以将 RMSE 的值作为模型收敛的指标:模型参数通过随机初始化得到后(算法输入),在模型参数优化阶段,首先在训练集上采用 SGD 更新模型参数(第 1 行),然后在测试集上计算 RMSE 和 MAE,验证模型参数的可行性(第 2 行).不断重复这个步骤,直至 RMSE 不再变小,完成对参数的优化(第 4 行、第 5 行).我们将模型在训练集上更新参数的迭代次数作为模型收敛速度的指标,对比各个模型的收敛速度.

3.3 对比实验及实验参数设置

本文采用以下几个对比实验.

- ① BasicMF^[6]:最简单形式的 MF,通过分解评分矩阵学习模型参数;
- ② SocialRec^[12]:共用用户特征矩阵同时分解评分矩阵和信任矩阵学习模型参数;
- ③ SocialMF^[13]:约定用户与其信任人有相似的特征向量,特征向量间的相似度取决于他们评分向量的 PCC;
- ④ SocialReg^[14]:规定用户与其信任人有相似的特征向量,特征向量的相似度取决于用户对其信任人的信任度;
- ⑤ CUNE^[15]:依据评分矩阵构建用户同质网络挖掘用户间的相似关系,在模型训练阶段约定具有相似偏好的用户有相似的特征向量.

为保持和其他对比实验的一致性,分别在特征向量维度 f 为 5 和 10 时验证各模型性能,并进行大量实验找出各模型的最优参数,除学习率均为 0.01 外,其他参数配置见表 2.

Table 2 Parameter settings of different models

表 2 模型参数设置

模型名称	参数设置
SocialMF	$\lambda_r=1, \lambda_u=\lambda_v=0.001$
SocialRec	$\lambda_c=0.1, \lambda_u=\lambda_v=\lambda_z=0.001$
SocialReg	$\beta=0.1, \lambda_1=\lambda_2=0.001$
CUNE	$\lambda_u=\lambda_v=\lambda_z=0.001, L=20, T=30, l=20, \tau=5, k=50$
BasicMF	$\lambda_1=\lambda_2=0.01$
ARMF	$\lambda=0.5, \lambda_1=\lambda_2=\lambda_3=0.001, \tau=20, \rho=30, t=2, \delta=\sigma=2, k=10, d=f$

3.4 实验结果分析

3.4.1 准确性分析

我们首先分析了所有模型在准确性方面的表现,各模型在两个数据集的 Warm 集和 Cold 集上的准确度见表 3 和表 4.

Table 3 Accuracy of ARMF comparing with other models over Warm users

表 3 在 Warm 集上各算法准确度对比

DataSet	Dim	Metrics	BasicMF	SocialRec	SocialReg	SocialMF	CUNE	ARMF
Filmtrust	5	RMSE	0.884 3	0.851 5	0.836 5	0.844 9	0.831 5	0.813 4
		MAE	0.718 8	0.654 9	0.641 2	0.650 8	0.652 1	0.628 4
	10	RMSE	0.883 9	0.856 5	0.843 2	0.843 7	0.829 9	0.811 1
		MAE	0.719 0	0.660 6	0.648 7	0.649 6	0.651 1	0.626 4
CiaoDVD	5	RMSE	1.064 2	1.140 1	1.067 9	1.098 1	1.022 0	1.008 4
		MAE	0.830 8	0.850 0	0.787 6	0.810 0	0.756 7	0.735 9
	10	RMSE	1.063 7	1.142 5	1.066 0	1.133 7	1.060 0	1.008 0
		MAE	0.830 6	0.853 7	0.787 7	0.828 6	0.785 3	0.733 7

Table 4 Accuracy of ARMF comparing with other models over *Cold* users**表 4** 在 *Cold* 集上各算法准确度对比

Data set	Dim	Metrics	BasicMF	SocialRec	SocialReg	SocialMF	CUNE	ARMF
Filmtrust	5	RMSE	0.984 8	1.003 1	0.945 0	0.937 4	0.949 7	0.877 7
		MAE	0.801 5	0.818 1	0.735 6	0.736 8	0.746 0	0.670 3
	10	RMSE	0.991 5	0.996 3	0.965 2	0.947 3	0.950 2	0.865 9
		MAE	0.810 1	0.807 7	0.763 3	0.742 5	0.754 2	0.661 1
CiaoDVD	5	RMSE	1.147 1	1.237 4	1.113 4	1.136 8	1.130 1	1.041 4
		MAE	0.863 4	0.940 9	0.800 5	0.831 4	0.828 8	0.715 1
	10	RMSE	1.147 2	1.240 6	1.110 4	1.134 1	1.131 3	1.035 3
		MAE	0.865 5	0.943 5	0.803 9	0.831 5	0.836 1	0.709 0

从两个表中可以看到,所有模型在特征向量维度 $f=5$ 和 $f=10$ 时性能保持相对一致.故我们以 $f=5$ 为例对各个模型进行分析,但分析结果同样适用于 $f=10$.由于所有模型的目标函数都是最小化评分预测值与真实值之间的差平方,故在实验中主要采用 RMSE 衡量模型的准确性,并以此分析和比较各个模型.

如表 3 所示,各个模型在两个数据集的 *Warm* 集上的性能分析如下.

- 在 FilmTrust 数据集上,由于 BasicMF 忽略了用户间的相似关系,数据稀疏问题导致其学习到的用户特征向量不够准确,所以在所有模型中表现最差;SocialRec, SocialReg, SocialMF 和 CUNE 都是通过分解评分矩阵学习模型参数,同时约束相似用户有相似的特征向量,所以它们的表现结果相差不大;
- 但各模型在 CiaoDVD 数据集上的性能并没有和在 FilmTrust 数据集上的性能保持完全一致,此时, SocialRec, SocialReg 和 SocialMF 准确度不如 BasicMF.这是因为这几个模型在训练模型参数时使用用户与其信任人有相似特征向量, FilmTrust 数据集中的信任关系可以近似看作相似关系,而 CiaoDVD 数据集上直接将信任关系看作相似关系会引入比较多的噪音,最终导致学习到的用户特征向量不能很好代表用户偏好;
- 由于 CUNE 挖掘的用户相似关系比直接将信任人看做相似朋友可靠,所以 CUNE 在两个数据集中都较其他几个模型取得更好的结果;
- ARMF 相比其他模型准确度得到了明显提升,主要因为 ARMF 在保证相似用户具有相似特征向量的同时分析用户对项目各个特征的关注度,从而获取了用户更准确的偏好.

从表 4 可发现, BasicMF 和 SocialRec 在两个 *Cold* 集上的准确性均不如其他模型,其中, SocialRec 最差.这是因为 *Cold* 集中用户评分记录较少, BasicMF 很难学习到能够代表他们偏好的特征向量.此外,这些用户的信任人也相对较少,这就使得需要同时分解评分矩阵和信任矩阵的 SocialRec 准确性最差. SocialReg 和 SocialMF 在学习模型参数时借助少量的信任关系缓解了冷启动问题. CUNE 在 *Cold* 集上较其他模型的表现不如其在 *Warm* 集上优越,这主要因为 CUNE 构建的网络中 *Cold* 集中用户不能与较多其他用户建立联系,导致 CUNE 学习到的隐向量不能很好反映出他们与其他用户间的相似关系,在学习最终参数时会引入噪音.这表明,仅依据评分矩阵挖掘用户间的相似关系对冷启动问题不敏感. ARMF 在评分矩阵的基础上引入信任关系构建相对较稠密的网络,在一定程度上解决了冷启动问题.网络嵌入阶段挖掘到的用户间可靠的相似关系,保证了训练阶段学习到的用户特征向量的准确性,使得 ARMF 在所有模型中取得最高的推荐准确度.

从表 3 和表 4 综合看来:无论是 *Warm* 集还是 *Cold* 集,所有模型在 FilmTrust 数据集上都取得比 CiaoDVD 数据集上高的准确度.这主要因为 FilmTrust 数据集相比 CiaoDVD 数据集中每个用户平均评分个数更多,而用户评分记录个数直接影响模型学习到的参数质量.

3.4.2 收敛速度分析

图 4 和图 5 分别是各个模型在 FilmTrust 数据集和 CiaoDVD 数据集上的收敛速度对比图,两幅图中均为左侧 $f=5$,右侧 $f=10$.

从图中可以看出,在 $f=5$ 和 $f=10$ 两种情况下, BasicMF 在两个数据集上的收敛速度都最慢,而 ARMF 的收敛速度是最快的.这主要是由于 BasicMF 在学习模型参数时忽略了用户之间的相关性,数据稀疏导致其在每次循环中学到的参数变动较大,难以达到平稳状态,收敛速度比较缓慢;其他模型通过约束用户向量之间的相似性缓

解了数据稀疏问题,使模型的收敛速度加快;ARMF 收敛速度之所以能够大幅度优于其他 RMF 模型,是因为注意力机制使 ARMF 能够快速分析出用户更准确的偏好.

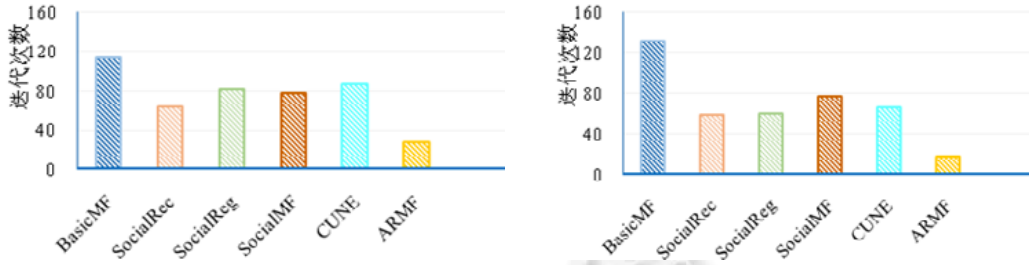


Fig.4 Rate of convergence of models on FilmTrust

图 4 各模型在 FilmTrust 数据集上的收敛速度

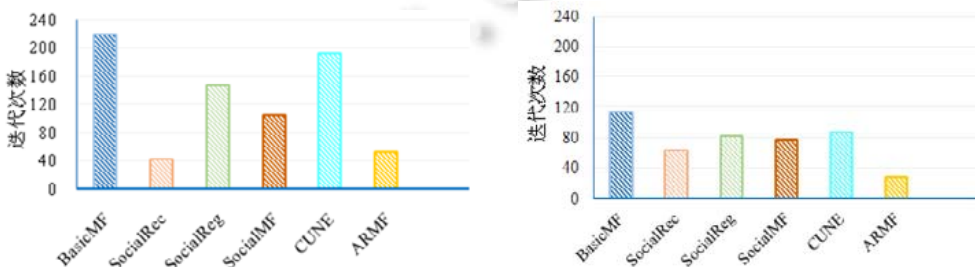


Fig.5 Rate of convergence of models on CiaoDVD

图 5 各模型在 CiaoDVD 数据集上的收敛速度

此外,SocialRec,SocialReg,SocialMF 和 CUNE 的收敛速度在两个数据集上的收敛速度表现不同.由于这 4 个模型都在训练模型参数时约束用户与其相似朋友有相似的特征向量,根据表 3 和表 4 中的实验结果已得出,FlimTrust 数据集中的信任关系可近似看作相似关系;又因为 CUNE 能够为用户找到可靠的相似朋友,这 4 个模型引入的噪音都较少,所以当 $f=5$ 和 $f=10$ 时,它们在 FilmTrust 数据集上收敛速度相差都不大.但当 $f=5$ 时,它们在 CiaoDVD 数据集上的收敛速度相差较大.这主要是因为将 CiaoDVD 数据集中信任关系看作相似关系会引入较多噪音,同时,数据稀疏导致 CUNE 挖掘到的相似关系中也会存在较多噪音.噪音的存在,使得收敛速度会对模型目标函数比较敏感,不同的目标函数会有不同的优化过程,使得它们的收敛速度也相差较大.但是当 $f=10$ 时,它们在 CiaoDVD 数据集上又有相近的收敛速度.这是因为特征向量维度的增加使得模型健壮性较好,使得模型不再因为噪音的出现而使收敛速度对模型目标函数敏感.

对比 4 副图还可以发现,所有模型在 FlimTrust 数据集上的收敛速度比其在 CiaoDVD 上的收敛速度快.这主要是因为 FlimTrust 数据集中用户平均评分记录个数比 CiaoDVD 数据集中的要多,用户评分记录越多,就更容易分析出其偏好,也就加快了模型收敛速度.

3.4.3 注意力机制分析

为验证 ARMF 中注意力机制的有效性,将 ARMF 去掉注意力机制之后的模型 NEMF 在两个数据集的 Warm 集和 Cold 集上做实验比较和其他模型的准确性.当 $f=5$ 时,所有模型在 4 个数据集上的 RMSE 实验结果如图 6 所示.从图中可以看出:ARMF 在 4 个数据集上的准确度最优,证明注意力机制的引入确实可以得到用户更准确的偏好,提高模型的准确度;NEMF 的准确度虽然不如 ARMF 却优于其他模型,说明 ARMF 为用户找到的相似朋友是最可靠的,通过约束用户与其相似朋友特征向量之间的相似性,能够学习到比较准确的用户偏好.当 $f=10$ 时能够得到同样的结论,所以此处省略了对比实验图.

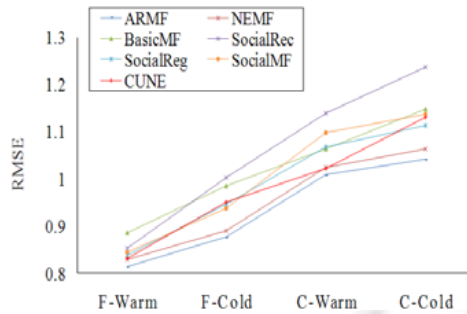


Fig.6 RMSE of models on four dataset

图 6 各模型在 4 个数据集上的 RMSE

3.4.4 模型特点总结

在综合比较了各个模型的准确性以及收敛速度后,为对实验中所有对比模型有更直观的了解,对各模型的特点进行了如下总结(见表 5).

- 由于 BasicMF 不依赖社交网络,所以对数据稀疏和冷启动敏感度都比较强,继而导致模型的推荐准确度很低,收敛速度很慢.综合看来,BasicMF 的实用性很弱;
- SocialRec 通过共享用户特征矩阵同时分解评分矩阵和信任关系矩阵,虽然使模型收敛速度加快,但是数据稀疏和冷启动问题导致其准确性与 BasicMF 接近,其实用性也较弱;
- SocialMF 和 SocialReg 通过引入信任关系缓解数据稀疏和冷启动问题,不仅加快了模型收敛速度,也使模型准确度提升.但这两个模型将用户的信任人看作其相似朋友,在引入噪音的同时,丢失了可能存在的相似关系,导致模型性能提升并不明显,实用性不是很强;
- CUNE 依据评分矩阵构建网络并挖掘用户间的相似关系,解决了数据稀疏问题,提高了模型准确度和模型的收敛速度.但对于冷启动用户,该算法无法学习到其准确的特征向量,所以实用性不是很高;
- ARMF 利用信任关系作为评分矩阵的辅助信息挖掘用户间的相似关系,解决了数据稀疏和冷启动问题,同时具有最高的准确度和最快的收敛速度,所以有较强的实用性.

Table 5 Summary of different models' characteristics

表 5 各模型特点总结

模型	数据稀疏敏感度	冷启动敏感度	社交网络依赖性	模型收敛速度	推荐准确度	实用性
BasicMF	强	强	弱	慢	低	弱
SocialRec	强	强	强	快	低	弱
SocialMF	中	强	强	中	中	中
SocialReg	中	强	强	中	中	中
CUNE	中	强	弱	中	中	中
ARMF	弱	弱	强	快	高	强

4 结束语

为解决数据稀疏和冷启动问题给 MF 在训练模型时带来的困扰,本文提出了融合用户信任关系和评分矩阵的基于注意力机制的规范化矩阵分解算法. ARMF 在挖掘用户间可靠相似关系的同时,能够分析出用户对项目不同特征的关注度,获取用户更准确的偏好.在利用网络嵌入技术找用户之间的相似关系时,为解决数据稀疏问题带来的困扰,ARMF 在依据评分矩阵构建用户-项目异构网络时,考虑了用户之间的信任关系对网络的影响,同时解决了冷启动问题.另外,ARMF 通过双线性方式引入了注意力机制,在没有大幅增加模型复杂度的情况下,获取了用户准确的偏好,同时保证了相似用户之间有相似的特征向量,提高了推荐准确度;注意力机制的引入也使模型收敛速度大大加快.在两个真实数据集上的大量实验结果,验证了 ARMF 的准确性和健壮性.

References:

- [1] Schafer JB, Konstan J, Riedl J. Recommender systems in e-commerce. In: Proc. of the 1st ACM Conf. on Electronic Commerce. ACM, 1999. 158–166.
- [2] Jamali M, Ester M. Trustwalker: A random walk model for combining trust-based and item-based recommendation. In: Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. 2009. 397–406.
- [3] Su JH, Chiu TW. An item-based music recommender system using music content similarity. In: Proc. of the Asian Conf. on Intelligent Information and Database Systems. Berlin, Heidelberg: Springer-Verlag, 2016. 179–190.
- [4] Xiang L. The Practice of Recommended System. Beijing: People's Posts and Telecommunications Press, 2012. 39–44 (in Chinese).
- [5] Su XY, Khoshgoftaar TM. A survey of collaborative filtering techniques. In: Proc. of the Advances in Artificial Intelligence. 2009.
- [6] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer*, 2009,42(8):30–37.
- [7] Li WM, Zhou XK, Shimizu S. Personalization recommendation algorithm based on trust correlation degree and matrix factorization. *IEEE Access*, 2019. [doi: 10.1109/ACCESS.2018.2885084]
- [8] Massa P, Avesani P. Trust-aware recommender systems. In: Proc. of the 2007 ACM Conf. on Recommender Systems. 2007. 17–24.
- [9] Guo GB, Zhang J, Thalmann D. Merging trust in collaborative filtering to alleviate data sparsity and cold start. *Knowledge-Based Systems*, 2014,57:57–68.
- [10] Sarwar BM, Karypis G, Konstan JA. Item-based collaborative filtering recommendation algorithms. In: Proc. of 10th Int'l Conf. on World Wide Web, 2001. 285–295.
- [11] Deng SG, Huang LT, Xu GD. On deep learning for trust-aware recommendations in social networks. *IEEE Trans. on Neural Networks and Learning Systems*, 2016,28(5):1164–1177.
- [12] Ma H, Yang H, Lyu MR. Sorec: Social recommendation using probabilistic matrix factorization. In: Proc. of the 17th ACM Conf. on Information and Knowledge Management. 2008. 931–940.
- [13] Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks. In: Proc. of the 4th ACM Conf. on Recommender Systems. 2010. 135–142.
- [14] Ma H, Zhou DY, Liu C. Recommender systems with social regularization. In: Proc. of the 4th ACM Int'l Conf. on Web Search and Data Mining. 2011. 287–296.
- [15] Zhang CX, Yu L, Wang Y. Collaborative user network embedding for social recommender systems. In: Proc. of the 2017 SIAM Int'l Conf. on Data Mining. 2017. 381–389.
- [16] He XL, Liao LZ, Zhang HW, *et al.* Neural collaborative filtering. In: Proc. of the 2017 Int'l World Wide Web Conf. Committee (IW3C2), 2017. 173–182. <http://dx.doi.org/10.1145/3038912.3052569>
- [17] Xue HJ, Dai XY, Zhang JB. Deep matrix factorization models for recommender systems. In: Proc. of the IJCAI. 2017. 3203–3209.
- [18] Vaswani A, Shazeer N, Parmar N. Attention is all you need. In: Proc. of the Advances in Neural Information Processing Systems. 2017. 5998–6008.
- [19] Cui P, Wang X, Pei J. A survey on network embedding. *IEEE Trans. on Knowledge and Data Engineering*, 2018,31(5):833–852.
- [20] Poggio T, Voinea S, Rosasco L. Online learning, stability, and stochastic gradient descent. arXiv:1105.4701, 2011.
- [21] Guo GB, Zhang J, Yorke-Smith N. A novel bayesian similarity measure for recommender systems. In: Proc. of the 23rd Int'l Joint Conf. on Artificial Intelligence. 2013.
- [22] Guo GB, Zhang J, Thalmann D. ETAF: An extended trust antecedents framework for trust prediction. In: Proc. of the 2014 IEEE/ACM Int'l Conf. on Advances in Social Networks Analysis and Mining. 2014. 540–547.
- [23] Zhang L, Liu X, Cao Y. O-recommend: An optimized user-based collaborative filtering recommendation system. In: Proc. of the IEEE 24th Int'l Conf. on Parallel and Distributed Systems (ICPADS). 2018. 212–219.
- [24] Madadipouya K. A location-based movie recommender system using collaborative filtering. arXiv: 1508.01696, 2015.
- [25] Fouss F, Pirotte A, Renders J. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation. *IEEE Trans. on Knowledge and Data Engineering*, 2007,19(3):355–369.
- [26] Ma JW, Wen JH, Zhong MY, Chen WT, Li X. MMM: Multi-source multi-net micro-video recommendation with clustered hidden item representation learning. *Data Science and Engineering*, 2019,4(3):240–253.
- [27] Golbeck JA. Computing and applying trust in web-based social networks. 2005. <http://hdl.handle.net/1903/2384>

- [28] Ma H, King I, Lyu MR. Learning to recommend with social trust ensemble. In: Proc. of the 32nd Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. 2009. 203–210.
- [29] Parvina H, Moradi P, Esmaili S. An efficient recommender system by integrating non-negative matrix factorization with trust and distrust relationships. In: Proc. of the IEEE Data Science Workshop (DSW). 2018. 135–139.
- [30] Guthrie D, Allison B, Liu W. A closer look at skip-gram modelling. In: Proc. of the LREC. 2006. 1222–1225.
- [31] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations. In: Proc. of the 20th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. 2014. 701–710.
- [32] Zhang P. Model selection via multifold cross validation. In: Proc. of the Annals of Statistics. 1993. 299–313.

附中文参考文献:

- [4] 项亮.推荐系统实践.北京:人民邮电出版社,2012.39–44.



张青博(1994—),女,河南三门峡人,博士生,主要研究领域为推荐算法,机器学习.



宋晓旭(1984—),男,讲师,主要研究领域为社交网络.



王斌(1972—),男,博士,副教授,博士生导师,主要研究领域为查询优化处理,图数据管理,隐私保护,推荐系统.



秦婧(1981—),女,硕士,CCF 专业会员,主要研究领域为推荐系统,数据挖掘.



崔宁宁(1988—),男,硕士,主要研究领域为数据安全,隐私保护.