

# 控制与决策

Control and Decision

## 一种融合信任和项目卷积描述信息的PMF算法

王建芳, 苗艳玲, 韩鹏飞, 司马海峰

引用本文:

王建芳, 苗艳玲, 韩鹏飞, 等. 一种融合信任和项目卷积描述信息的PMF算法[J]. *控制与决策*, 2020, 35(8): 1803–1812.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2018.1678>

---

## 您可能感兴趣的其他文章

Articles you may be interested in

### 社会网络环境下基于分布式信任的在线评价方法

A distributed trust based online evaluation under social network

*控制与决策*. 2020, 35(7): 1697–1702 <https://doi.org/10.13195/j.kzyjc.2018.1527>

### 基于信任关系和互补判断矩阵的群决策方法

Group decision making method for fuzzy complementary judgement matrices based on trust relationships

*控制与决策*. 2020, 35(5): 1240–1246 <https://doi.org/10.13195/j.kzyjc.2018.1288>

### 基于卷积神经网络的层级化智能故障诊断算法

A hierarchical intelligent fault diagnosis algorithm based on convolutional neural network

*控制与决策*. 2019, 34(12): 2619–2626 <https://doi.org/10.13195/j.kzyjc.2018.0253>

### 基于反卷积特征提取的深度卷积神经网络学习

Deep convolution neural network learning based on deconvolution feature extraction

*控制与决策*. 2018, 33(3): 447–454 <https://doi.org/10.13195/j.kzyjc.2017.0048>

### 基于信任机制的不完全信息大群体决策方法

Method for large group decision making with incomplete decision preference information based on trust mechanism

*控制与决策*. 2016, 31(4): 577–585 <https://doi.org/10.13195/j.kzyjc.2015.0104>

# 一种融合信任和项目卷积描述信息的PMF算法

王建芳<sup>†</sup>, 苗艳玲, 韩鹏飞, 司马海峰

(河南理工大学 计算机科学与技术学院, 河南 焦作 454000)

**摘要:** 针对现有概率矩阵分解(PMF)技术的个性化推荐系统在采用社交网络中信任信息时常常忽视项目相关描述文档信息的问题,提出一种融合用户信任和通过卷积网络以获取项目描述等信息的PMF模型. 首先,利用用户偏好信息和行为轨迹信息构建一种新的信任网络;然后,通过卷积神经网络从项目描述文档中提取项目潜在的特征向量;最后,在概率矩阵分解过程中同时利用评分数据、信任网络中用户的信任信息和项目的描述信息,计算用户和项目的潜在特征向量以预测评分并进行个性化推荐. 为验证算法的有效性,选择3种算法在4个数据集上进行对比,实验结果表明所提出的算法在推荐精确度和鲁棒性方面优于其他3种算法.

**关键词:** 推荐算法; 卷积神经网络; 社交网络; 信任; 概率矩阵分解

中图分类号: TP391

文献标志码: A

## A PMF algorithm based on trust and item convolutional description information

WANG Jian-fang<sup>†</sup>, MIAO Yan-ling, HAN Peng-fei, SIMA Hai-feng

(School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454000, China)

**Abstract:** The existing personalized recommendation algorithm for the probabilistic matrix factorization (PMF) usually ignores the description document information of the hidden items when using trust information in social networks. A PMF algorithm is proposed based on improved trust and item convolutional description information (ITC-PMF). Firstly, a new trust network is constructed by using user preferences and behavior trajectory information. Then, the item latent features are extracted from contextual documents by convolutional neural network (CNN). Furthermore, based on the PMF, the rating records, the trust information and the item description information are simultaneously used to calculate the latent feature vectors of the users and the items, so as to predict and make personalized recommendations. Finally, to verify effectiveness of the ITC-PMF algorithm, three other state-of-the-art algorithms are compared on four real-world datasets. Experimental results show that the proposed algorithm outperforms other three algorithms in terms of recommendation accuracy and robustness.

**Keywords:** recommender; convolutional neural network; social network; trust; probabilistic matrix factorization

## 0 引言

推荐系统主要针对的是信息过载问题,旨在为用户在不同领域提供与自身匹配的个性化信息. 在众多推荐技术中,协同过滤推荐算法由于仅利用评分信息而受到广泛关注<sup>[1-2]</sup>. 协同过滤算法主要分两类:基于内存的协同过滤和基于模型的协同过滤<sup>[3]</sup>. 经典的基于模型的协同过滤算法中,概率矩阵分解技术是应用最成功的模型之一,该方法从概率角度出发,将用户对项目(电影、音乐、美食和服务等)的评分信息以矩阵形式表示,通过对矩阵的概率分解操作挖掘低维潜在特征空间,并将用户与项目在该低维空间上进

行重表示,进而以用户及项目向量间的内积来刻画用户项目之间的关联性. 但其仍然存在数据稀疏性和冷启动问题,从而导致推荐精度降低<sup>[4]</sup>. 为了提高推荐质量,除了使用用户-项目评分信息外,融合社交网络中的信任信息或项目的描述文档信息等辅助信息源的推荐方案相继被提出.

社交网络中用户之间的信任作为人际关系的核心,将直接影响用户的决策过程,因此成为社交化推荐的研究重点内容之一<sup>[5-10]</sup>,信任信息的融入可有效缓解用户冷启动问题. 而作为推荐算法的另一信息源——项目相关的文档描述信息(如项目简介和项目

收稿日期: 2018-12-06; 修回日期: 2019-04-03.

基金项目: 国家自然科学基金项目(61602157); 河南省高等学校重点科研项目(15A520074).

责任编委: 曹进德.

<sup>†</sup>通讯作者. E-mail: wangjianfang@hpu.edu.cn.

评价等),近年来在提高推荐质量方面引起学术界的青睐<sup>[11-15]</sup>. Wang等<sup>[13]</sup>提出了协同深度学习算法将堆栈式降噪自编码器融入概率矩阵分解模型,结合项目文档信息和用户项目评分信息,获得更精确的用户和项目潜在因子矩阵; Kim等<sup>[15]</sup>将用于句子分类的卷积神经网络算法应用于文档推荐,利用卷积神经网络学习项目潜在特征向量,提高项目隐向量的精确度. 相关研究成果表明,有效利用基于文档信息的推荐技术可以提供更为精确的预测推荐结果. 为了清晰地描述本文所提出的算法,表1汇总了常用的符号及含义.

表1 本文常用符号及含义

符号	含义
$R$	用户-项目评分矩阵
$U_u, V_q$	用户 $u$ 和项目 $q$ 的潜在特征向量
$R_{uq}$	用户 $u$ 对项目 $q$ 的评分
$\hat{R}_{uq}$	用户 $u$ 对项目 $q$ 的预测评分
$PS_{uv}$	用户 $u$ 和用户 $v$ 之间的偏好相似度
$I_{uv}$	用户 $u$ 和用户 $v$ 共同评价的项目集合
$D$	项目描述文档数字化后的矩阵
$W_c^j$	第 $j$ 个卷积核矩阵
$c$	文档 $D$ 通过卷积得到的特征向量组
$W_{f1}, W_{f2}$	神经网络中全连接层的层间连接权重
$U, V$	用户和项目潜在特征矩阵
$N, M$	用户个数和项目个数
$T_{iv}$	用户 $i$ 对用户 $v$ 的信任值
$TS_{uv}$	用户 $u$ 和用户 $v$ 之间的行为轨迹相似度
$\bar{r}_u, \bar{r}_v$	用户 $u$ 和用户 $v$ 的评分均值
$ I_{uv} $	用户 $u$ 和用户 $v$ 共同评价的项目个数
$w_i$	单词 $i$ 的词向量
$c^j$	通过 $W_c^j$ 卷积得到的文档特征向量
$d_f$	文档 $D$ 提出的特征向量
$s_q$	通过 CNN 得到的项目 $q$ 的潜在向量

据可查资料,目前个性化推荐方案并没有同时将用户的社交信任信息和项目的描述信息作为研究对象. 本文提出一种融合信任和项目卷积描述信息的 PMF 算法(a PMF algorithm based on improved trust and item convolutional description information, ITC-PMF),在概率矩阵分解框架的基础上,针对包含项目相关文档描述信息的数据集,融入用户信任信息和项目描述文档信息,有效地提高了推荐质量.

## 1 相关工作

在推荐算法研究领域,基于信任的模型和基于项目描述文档的模型分别被广泛研究. 基于信任的模型中信任网络的构建是重点. 社交推荐中,用户之间的社交信任关系可以表示成矩阵形式,分为基于显式和隐式的社交关系<sup>[16]</sup>. 显式社交关系有利于用户好

友关系网络的建立,但是简单的连接关系并不能很好地反映用户之间的关联强度. 隐式社交关系通过深层信息计算得到关联强度,目前已有一些度量方法被提出,用于表示用户间社交关联强度,如皮尔逊相关系数(pearson correlation coefficient, PCC)、余弦相似度(cosine similarity)和修正的余弦相似度(modified cosine similarity)等,隐式社交关系为用户之间建立信任关系网络提供重要保证.

Ma等<sup>[9]</sup>提出的 SoRec 算法将信任关系与用户-项目评分信息融入概率矩阵分解模型中, SoRec 是第一个基于矩阵分解的社交推荐模型. Jamali等<sup>[10]</sup>基于社交网络中信任传播机制提出了一种新型社交推荐模型 SocialMF,将用户之间的信任值作为用户潜在特征向量的权重融入推荐过程. Bo等<sup>[17]</sup>提出基于信任与被信任关系的社交推荐模型 TrustMF,将每一个用户映射为信任者特征向量和被信任者特征向量. 但是,这些工作过度聚焦于数据中给出的显式信任关系,忽略了实际生活中用户之间信任关系建立的根本原因. 所以若数据中没有明确的信任关系,如何构建用户之间的信任网络则显得尤为重要. 本文基于用户相似度信息和行为轨迹相似度信息,重构用户信任矩阵,加强推荐算法的可信度和可解释性.

近年来,基于项目描述文档的模型在推荐技术中取得了一定成果. Wang等<sup>[18]</sup>提出协同主题回归算法,将 LDA(latent dirichlet allocation)文档分析模型与协同过滤以概率方法相融合,不仅弥补了原始 LDA 不能发现隐式主题的问题,而且很好地解决了冷启动问题. 但是,现有的算法模型并不能完全抓住文档的深层信息,因为词袋(bag of words)方法没有考虑语义、语法以及词与词之间的顺序,导致其只能停留在文档的字面意思,不能有效利用文本信息. 为了更好地捕获文本的深层信息, Kim<sup>[14]</sup>提出用于句子分类的卷积神经网络算法,该算法能够方便地将不定长的输入转换成定长输入,接入到卷积神经网络中,使得分类模型能够更好地捕捉文本局部的局部信息,提高了算法效率. 之后, Kim等<sup>[15]</sup>提出了卷积矩阵分解(convolutional matrix factorization, ConvMF)算法,采用卷积神经网络架构有效地捕获文本的局部特征,对于评分数据稀疏和文本信息利用不充分问题,卷积神经网络模型的使用有助于获得文本的深层信息,产生更合理的项目潜在模型. 虽然上述算法在项目描述文档的特征提取方面效果很好,但是这些算法均没有考虑用户之间的信任信息,所以如何有效地将用户之间的信任信息和项目描述文档信息更好地建模融合

是提高推荐质量的关键问题。

本文提出的ITC-PMF算法利用用户之间的隐式社交关系刻画用户潜在特征,通过卷积神经网络提取项目描述文档的局部特征信息,构造项目潜在特征,最后,基于概率矩阵分解模型融合用户之间的社交信息和项目描述文档信息,获得推荐结果。

## 2 相关理论

### 2.1 用户偏好相似度

协同过滤的核心是相似性度量<sup>[19-20]</sup>,PCC是最常见的相似性度量方法,用户间的相似度<sup>[21]</sup>计算为

$$PS_{uv} = \frac{\sum_{j \in I_{uv}} (r_{uj} - \bar{r}_u)(r_{vj} - \bar{r}_v)}{\sqrt{\sum_{j \in I_{uv}} (r_{uj} - \bar{r}_u)^2} \sqrt{\sum_{j \in I_{uv}} (r_{vj} - \bar{r}_v)^2}}. \quad (1)$$

其中: $I_{uv}$ 为用户 $u$ 和用户 $v$ 共同评价过的项目集合; $r_{uj}$ 和 $r_{vj}$ 分别为用户 $u$ 和用户 $v$ 对项目 $j$ 的评分; $\bar{r}_u$ 和 $\bar{r}_v$ 分别为用户 $u$ 与用户 $v$ 的评分均值; $PS_{uv}$ 为 $u_u$ 与 $u_v$ 之间的相似度,取值范围是 $[-1,1]$ ,负值表示不具备相关性,因此只取相似度矩阵的正值,且值越大表示用户之间的潜在偏好行为越相似。

### 2.2 行为轨迹相似度

行为轨迹相似性是指基于用户的历史行为记录预测用户的轨迹相似性。一般情况下用户-项目评分矩阵通常比较稀疏,数据集中评分数量占比一般小于5%,PCC不适用于稀疏矩阵,不能完全真实反映用户的评分情况,因为PCC仅估算数值距离,没有考虑用户的行为轨迹,然而行为轨迹对于用户相似性计算又具有非常重要的价值。本文采用Jaccard系数<sup>[22]</sup>计算用户行为轨迹的相似度,用于量化区分二元变量的不对称信息,呈现用户的行为轨迹。用户之间的行为轨迹相似度计算为

$$TS_{uv} = \frac{|I_{uv}|}{|I_u| + |I_v| - |I_{uv}|}. \quad (2)$$

其中: $|I_{uv}|$ 为 $u_u$ 和 $u_v$ 共同评分过的项目个数; $|I_u|$ 和 $|I_v|$ 分别为 $u_u$ 和 $u_v$ 各自评分的项目个数; $TS_{uv}$ 为用户 $u_u$ 和 $u_v$ 的行为轨迹相似度,取值范围是 $[0,1]$ ,值越大表示用户间行为轨迹越相似。

### 2.3 概率矩阵分解

矩阵分解的主要思想是从用户-项目评分矩阵中学习用户和项目在低维隐空间上的表示 $U$ 和 $V$ 。假设有 $N$ 个用户, $M$ 个项目,用户-项目评分矩阵为 $R \in R^{N \times M}$ , $R_{uq}$ 为用户 $u$ 对项目 $q$ 的评分,则用户潜在特征矩阵 $U = [U_1, U_2, \dots, U_N] \in R^{K \times N}$ , $U_u$ 为用户 $u$ 的偏好向量; $V = [V_1, V_2, \dots, V_M] \in R^{K \times M}$ 为项目

潜在特征矩阵, $V_q$ 为项目 $q$ 的潜在特征向量; $K$ 为分解维度;预测评分 $\hat{R}_{uq} = U_u^T V_q$ 。在矩阵分解基础上,Salakhutdinov等<sup>[23]</sup>从概率角度进行解释,提出概率矩阵分解模型(probabilistic matrix factorization, PMF),假设真实评分 $R_{uq}$ 与预测评分 $U_u^T V_q$ 的差值符合均值为0、方差为 $\sigma^2$ 的高斯分布,即 $p(R_{uq} - U_u^T V_q | 0, \sigma^2)$ ,所以真实评分值服从均值为 $U_u^T V_q$ 、方差为 $\sigma^2$ 的高斯分布。假设所有观测评分值均相互独立,可得

$$p(R|U, V, \sigma^2) = \prod_{u=1}^N \prod_{q=1}^M [N(R_{uq}|U_u^T V_q, \sigma^2)]^{I_{u,q}}. \quad (3)$$

其中: $N(x|0, \sigma^2)$ 为 $x$ 服从均值为0、方差为 $\sigma^2$ 的高斯分布; $I_{u,q}$ 为指示函数,如果用户 $u$ 对项目 $q$ 有评分,则 $I_{u,q} = 1$ ,否则 $I_{u,q} = 0$ 。同时假设用户和项目的潜在特征矩阵 $U$ 和 $V$ 均服从均值为0、方差分别为 $\sigma_U^2$ 和 $\sigma_V^2$ 的高斯分布,且用户潜在特征矩阵中的各个用户特征向量满足独立同分布,项目潜在特征矩阵中的各个项目特征向量满足独立同分布,有

$$p(U|\sigma_U^2) = \prod_{u=1}^N N(U_u | 0, \sigma_U^2 I), \quad (4)$$

$$p(V|\sigma_V^2) = \prod_{q=1}^M N(V_q | 0, \sigma_V^2 I). \quad (5)$$

由贝叶斯规则可知,用户和项目特征矩阵的后验分布计算如下:

$$\begin{aligned} p(U, V|R, \sigma^2, \sigma_U^2, \sigma_V^2) &\propto \\ p(R|U, V, \sigma^2) p(U|\sigma_U^2) p(V|\sigma_V^2) &= \\ \prod_{u=1}^N \prod_{q=1}^M [N(R_{uq}|U_u^T V_q, \sigma^2)]^{I_{u,q}} \times & \\ \prod_{u=1}^N N(U_u|0, \sigma_U^2 I) \prod_{q=1}^M N(V_q|0, \sigma_V^2 I). & \quad (6) \end{aligned}$$

最大化特征矩阵 $U$ 和 $V$ 的后验概率等价于最小化式(6)的负对数,可得目标函数为

$$\begin{aligned} L(R, U, V, \sigma^2, \sigma_U^2, \sigma_V^2) &= \\ \frac{1}{2} \sum_{u=1}^N \sum_{q=1}^M I_{u,q} (R_{uq} - U_u^T V_q)^2 + & \\ \frac{\lambda_U}{2} \sum_{u=1}^N \|U_u\|_{\text{Fro}}^2 + \frac{\lambda_V}{2} \sum_{q=1}^M \|V_q\|_{\text{Fro}}^2. & \quad (7) \end{aligned}$$

其中: $\lambda_U$ 和 $\lambda_V$ 为正则化系数,且 $\lambda_U = \sigma^2/\sigma_U^2$ , $\lambda_V = \sigma^2/\sigma_V^2$ ; $\|\cdot\|_{\text{Fro}}^2$ 为二范数。

### 2.4 卷积神经网络

目前卷积神经网络(convolutional neural network, CNN)的核心思想被成功用于信息检索和自然语言处理等领域,并在自然语言处理领域展现出出色的

效果<sup>[24]</sup>. CNN在分类问题方面效果突出,但在推荐算法方面的成果却不多,主要因为推荐是回归问题,两者的目标不同.为解决这一问题,Dieleman等<sup>[25]</sup>将CNN架构用于音乐推荐系统,其主要对歌曲的音频信号作处理.但是,如果将项目文档信息按照同样的方法应用于文档类型的推荐显然是行不通的,因为音频数据和文档数据存在本质上的区别,音频信号在每个时间点与其周围的信号有固定的震动频率类似,而文档信息中每个位置的单词与其周围的单词语义差异较大,这就需要设计能够处理文档信息的CNN框架.本文采用文献[15]中的ConvMF算法处理文档信息,将项目文档信息转换成数字矩阵,获得项目的潜在特征向量.

### 3 ITC-PMF算法描述

#### 3.1 构建信任矩阵

用户之间的信任关系是建立在用户之间具有社会相似性的基础上,即在社交关系中用户具有相似的兴趣偏好和行为等可以反映用户个体特征的指标.本文使用用户偏好相似度和行为轨迹相似度构建用户之间的信任矩阵.

用户偏好相似度:根据用户-项目评分矩阵,使用式(1)计算用户评分之间的相似度,从而获取用户的偏好信息.

行为轨迹相似度:根据行为轨迹矩阵,利用式(2)计算用户行为记录的相似度,从而获取用户行为相似度.

用户之间的信任值计算如下:

$$T_{uv} = \alpha PS_{uv} + (1 - \alpha) TS_{uv}. \quad (8)$$

其中: $\alpha$ 为平衡用户偏好相似度和行为轨迹相似度的参数,取值范围是[0, 1],值越大表示信任值的取值越依赖用户偏好相似度,相反,值越小表示信任值的取值越依赖用户的行为轨迹相似度,本文根据实验设置 $\alpha = 0.3$ ;  $T_{uv}$ 的取值范围是[0, 1],值越大表示用户 $u$ 与用户 $v$ 之间的兴趣和行为越相似,用户 $u$ 越可能相信用户 $v$ ,通过设置 $T_{uv}$ 的阈值确定用户 $u$ 的信任用户集 $N_u$ .

#### 3.2 处理项目描述文档信息的CNN框架

本文利用CNN框架处理文档信息. CNN框架的输入是项目描述文档,目标输出是项目文档的潜在特征向量,主要有嵌入层、卷积层和池化层.下面对每层进行详细分析.

##### 1) 嵌入层.

嵌入层将文档中的每个单词表示成向量,使每

个文档转变为由单词向量组成的数字矩阵.单词转换为向量时,本文采用GloVe<sup>[26]</sup>词向量模型进行初始化,再通过优化算法训练得到最终的词向量.假设项目描述文档矩阵用 $D$ 表示,每个文档由固定长度 $l$ 个的单词组成,每个单词的词向量维度为 $a$ ,则有

$$D = [\dots w_{(i-1)} w_i w_{(i+1)} \dots] \in R^{a \times l}. \quad (9)$$

其中: $w_i$ 为文档 $D$ 中第 $i$ 个单词的词向量, $w_i \in R^a$ .

##### 2) 卷积层.

由于文档信息与图像音频信息存在本质区别,本文采用文献[14-15]中的CNN框架处理文档.假设第 $j$ 个卷积核矩阵表示为 $W_c^j$ ,其窗口大小为 $h$ , $h$ 决定周围单词数量,有 $W_c^j \in R^{a \times h}$ ,则通过该卷积核得到的文档特征向量表示为 $c_j$ 且 $c_j \in R^{(l-h+1)}$ ,有

$$c_j = [c_1^j, c_2^j, \dots, c_p^j, \dots, c_{l-h+1}^j]. \quad (10)$$

其中 $c_p^j$ 为卷积核窗口滑过第 $p$ 个位置时的卷积结果,且有

$$c_p^j = f(W_c^j * D_{(:,p:(p+h-1))} + b_c^j), \quad (11)$$

$*$ 为卷积运算, $b_c^j$ 为 $W_c^j$ 的偏置, $b_c^j \in R$ , $f(\cdot)$ 为节点的激励函数,本文使用ReLU(线性整流单元)函数.

由于每个卷积核能够提取文档的一种特征,为了尽可能多地获取文档的基础特征,选取 $n_c$ ( $n_c \in R$ )个卷积核,卷积运算后得到文档 $D$ 的特征向量组 $c$ 如下所示:

$$c = \{c^1, c^2, \dots, c^j, \dots, c^{n_c}\}, \quad (12)$$

其中 $c^j$ 为由第 $j$ 个卷积核提取得到的相应的特征向量,由于卷积窗口的大小 $h$ 不一样,导致 $c^j$ 的维度随 $h$ 变化而不同.

##### 3) 池化层.

为了得到更能反映文档信息的特征和构建相同维度的特征向量,本文提取每个特征向量里的最大值来反映该特征属性,所以经过池化层的处理,每个文档均由一个长度为 $n_c$ 的特征向量来表示,即

$$d_f = [\max(c^1), \max(c^2), \dots, \max(c^{n_c})]. \quad (13)$$

其中 $d_f$ 为项目描述文档 $D$ 的特征向量.

##### 4) 输出层.

输出层是神经网络中的全连接层,由隐含层和输出层构成.通过这两层网络将文档的特征向量 $d_f$ 映射到一个 $K$ 维向量空间,得到最终的文档潜在向量 $s$ 为

$$s = \tanh(W_{f2}\{\tanh(W_{f1}d_f + b_{f1})\} + b_{f2}). \quad (14)$$

其中: $W_{f1}$ 为输入层到隐含层的连接权重, $W_{f1} \in$

$R^{g \times n_c}$ ,即输入层节点数为 $n_c$ ,隐含层节点数为 $g$ ;  $W_{f_2}$ 是隐含层到输出层的连接权重,  $W_{f_2} \in R^{K \times g}$ ,即输出层的节点数为 $K$ ,对应于推荐模型中的项目潜在向量的分解维度 $K$ ;  $f$ 为池化层的输出,作为全连接层的输入;  $b_{f_1}$ 和 $b_{f_2}$ 分别为 $W_{f_1}$ 和 $W_{f_2}$ 的偏置,  $b_{f_1} \in R^g$ ,  $b_{f_2} \in R^K$ ,  $s \in R^K$ ;  $\tanh$ 为网络中节点使用的激励函数.

项目描述文档信息通过CNN框架的4层得到该项目的潜在向量. CNN框架可以看作是一个函数,输入是项目描述文档信息 $D$ ,变量是全连接层的连接权重 $W$ ,输出是项目潜在向量,即

$$s_q = f_{\text{CNN}}(W, D^q). \quad (15)$$

其中:  $D^q$ 为项目 $q$ 对应的原始描述文档,  $W$ 为全连接层的连接权重,  $s_q$ 为项目 $q$ 的潜在向量.

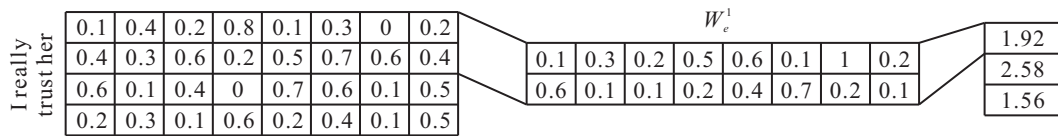


图1 CNN提取项目描述文档的潜在特征

### 3.3 ITC-PMF算法

ITC-PMF算法的核心思想如图2所示,利用用户之间的信任信息 $T$ 计算用户潜在特征表示 $U$ ,利用项目的描述文档信息 $D$ 学习项目潜在特征表示 $V$ ,然后将潜在特征表示融入PMF模型中拟合用户-项目评分矩阵 $R$ ,再根据损失函数 $L(R, U, V, T, D, W)$ 利用梯度下降法学习到最终的用户和项目潜在特征表示,从而对用户进行推荐.

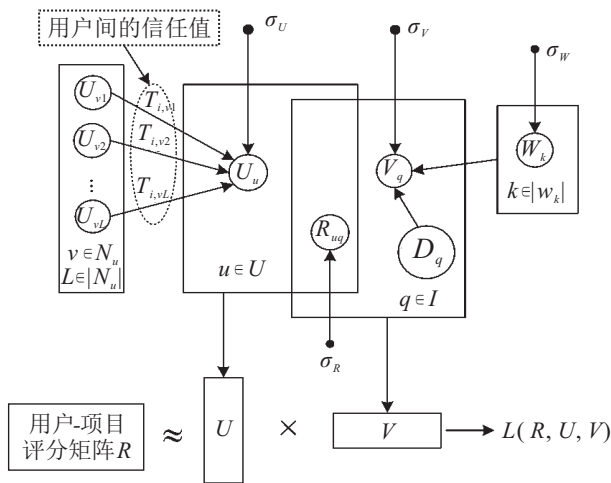


图2 ITC-PMF算法框架

如第2.3节所述,真实评分与预测评分的误差服从0均值高斯分布(如式(3)所示).若用户 $u$ 的信任用户集是 $N_u$ ,则用户 $u$ 的潜在特征向量受其信任用户

最后通过简单示例表明CNN提取项目描述文档深度信息的过程,如图1所示.例如“**I really trust her**”这句话通过嵌入层后可表示成一个二维矩阵,在卷积层使用不同的卷积核提取文档的不同特征,假设使用 $W_c^1$ 卷积后的结果如图1所示,是一个一维向量(1.92, 2.58, 1.56),这3个数分别代表(I, really)、(really, trust)和(trust, her)两两单词之间的关联.通过池化层后选出(1.92, 2.58, 1.56)中的最大值2.58代表抽取的文档潜在特征,这一特征在此可理解为英语语法中副词“really”修饰动词“trust”时,副词置于动词前.以上能够体现CNN可捕获句子中词序这一深层信息,而不同的卷积核可以提取到不同的文档特征,从而为项目潜在特征向量的生成提供更加精确的信息.

$v(v \in N_u)$ 的影响,有

$$\hat{U}_u = \sum_{v \in N_u} T_{uv} U_v. \quad (16)$$

与PMF模型不同的是:存在两个因素影响用户潜在特征向量的生成.一是用户潜在特征向量符合0均值的高斯分布;二是信任用户集对相应用户提供的潜在特征符合均值为 $\hat{U}_u$ 的高斯分布.用户特征向量的条件分布为

$$p(U|T, \sigma_U^2, \sigma_T^2) \propto p(U|\sigma_U^2) p(U|T, \sigma_T^2) = \prod_{u=1}^N N(U_u|0, \sigma_U^2 I) \times \prod_{u=1}^N N\left(U_u \mid \sum_{v \in N_u} T_{uv} U_v, \sigma_T^2 I\right). \quad (17)$$

加入项目描述文档信息后,项目潜在特征向量与PMF模型也有所不同.项目潜在特征向量的生成过程主要包含两个部分:一是项目的描述文档 $D$ 经过CNN架构处理过的文档潜在向量 $s$ ;二是项目潜在向量服从0均值高斯分布.所以项目 $q$ 的潜在向量表示为

$$V_q = s_q + \varepsilon_q. \quad (18)$$

其中: $s_q$ 为项目 $q$ 对应的描述文档的特征向量,  $\varepsilon_q$ 为项目潜在向量服从0均值高斯噪声.整理式(18)有 $V_q - s_q = \varepsilon_q$ ,所以 $V_q - s_q$ 服从均值为0的高斯分布,即 $V_q$ 服从均值为 $s_q$ 的高斯分布,如式(15)所示,  $s_q =$

$f_{\text{CNN}}(W, D^q)$ . 项目  $q$  潜在向量的条件分布为

$$p(V|W, D, \sigma_V^2) = \prod_q^M N(v_q | f_{\text{CNN}}(W, D^q), \sigma_V^2 I). \quad (19)$$

其中  $W$  中的每个  $w_k$  服从 0 均值高斯分布如下:

$$p(W | \sigma_W^2) = \prod_k N(w_k | 0, \sigma_W^2). \quad (20)$$

融入用户间的信任信息和项目的描述文档信息, 由贝叶斯规则可知用户和项目特征矩阵的后验分布为

$$\begin{aligned} & p(U, V | R, T, D, W, \sigma^2, \sigma_U^2, \sigma_V^2, \sigma_T^2, \sigma_W^2) \propto \\ & p(U, V | \sigma^2) p(U | \sigma_U^2) p(U | T, \sigma_T^2) \times \\ & p(V | W, D, \sigma_V^2) p(W | \sigma_W^2) = \\ & \prod_{u=1}^N \prod_{q=1}^M [N(R_{uq} | U_u^T V_q, \sigma^2)]^{I_{u,q}} \times \\ & \prod_{u=1}^N N\left(U_u \mid \sum_{v \in N_v} T_{uv} U_v, \sigma_U^2 I\right) \times \\ & \prod_q^M N(v_q | f_{\text{CNN}}(W, D^q), \sigma_V^2 I) \prod_k N(w_k | 0, \sigma_W^2). \end{aligned} \quad (21)$$

对式(21)取对数, 得到的目标函数形式为

$$\begin{aligned} L(R, U, V, T, D, W) = & \sum_{u=1}^N \sum_{q=1}^M \frac{I_{u,q}}{2} (R_{uq} - U_u^T V_q)^2 \frac{\lambda_U}{2} \sum_{u=1}^N \|U_u\|^2 + \\ & \frac{\lambda_T}{2} \sum_{u=1}^N \left\| U_u - \sum_{v \in N_v} T_{uv} U_v \right\|^2 \times \\ & \frac{\lambda_V}{2} \sum_{q=1}^M \|V_q - f_{\text{CNN}}(W, D^q)\|^2 + \frac{\lambda_W}{2} \sum_k \|w_k\|^2. \end{aligned} \quad (22)$$

其中:  $\lambda_U = \sigma^2/\sigma_U^2$ ,  $\lambda_T = \sigma^2/\sigma_T^2$ ,  $\lambda_V = \sigma^2/\sigma_V^2$ ,  $\lambda_W = \sigma^2/\sigma_W^2$ . 采用梯度下降法更新变量  $U_u$  和  $V_q$ , 有

$$\begin{aligned} \frac{\partial L}{\partial U_u} = & \sum_{q=1}^M I_{u,q} (U_u^T V_q - R_{uq}) V_q + \\ & \lambda_U U_u + \lambda_T \left( U_u - \sum_{v \in N_v} T_{uv} U_v \right), \end{aligned} \quad (23)$$

$$\begin{aligned} \frac{\partial L}{\partial V_q} = & \sum_{u=1}^N I_{u,q} (U_u^T V_q - R_{uq}) U_u + \\ & \lambda_V (V_q - f_{\text{CNN}}(W, D^q)). \end{aligned} \quad (24)$$

由于  $W$  的更新与 CNN 架构中的文档特征向量及全连接层的节点激励函数等因素有关, 关于  $W$  的损失函数为

$$\begin{aligned} l(W) = & \frac{\lambda_V}{2} \sum_{q=1}^M \| (V_q - f_{\text{CNN}}(W, D^q)) \|^2 + \\ & \frac{\lambda_W}{2} \sum_k \|w_k\|^2, \end{aligned} \quad (25)$$

使用经典神经网络中的误差反向传播算法更新  $W$  值.

#### 算法1 ITC-PMF算法.

输入: 用户评分矩阵  $R$  和项目描述文档  $D$ ;

输出: 用户潜在特征矩阵  $U$ , 项目潜在特征矩阵  $V$ , 损失函数值 **Loss** 和均方根误差 **RMSE**.

step 1: 初始化用户潜在特征矩阵  $U$  和项目潜在特征矩阵  $V$ .

step 2: 根据评分矩阵  $R$ , 利用式(1)计算用户之间的偏好相似度  $PS_{uv}$ .

step 3: 根据用户行为轨迹, 利用式(2)计算用户之间的行为相似度  $TS_{uv}$ .

step 4: 根据式(8)计算用户之间的信任值  $T_{uv}$ , 设置阈值  $t$ , 确定每个用户的可信用户集  $N_u$ , 为用户潜在特征向量构建提供重要信息.

step 5: 迭代开始:

step 5.1: 利用 GloVe 模型初始化文档的词向量  $D$ ;

step 5.2: 将词向量输入 CNN 模型, 根据式(11)、(13)和(14)得到项目文档特征向量  $s$ ;

step 5.3: 根据信任值  $T_{uv}$  和可信用户集  $N_u$ , 利用式(16)构建用户潜在特征矩阵  $U$ ;

step 5.4: 根据得到的项目文档特征向量  $s$ , 利用式(19)构建项目潜在特征矩阵  $V$ ;

step 5.5: 利用式(22)计算此次迭代对应的损失函数值 **Loss**, 根据式(23)~(25)更新变量  $U_u$ 、 $V_q$  和  $W$ ;

step 5.6: 如果损失函数值小于设定值或者迭代次数大于 100, 则停止迭代.

step 6: 得到用户潜在特征矩阵  $U$ , 项目潜在特征矩阵  $V$ , 损失函数值 **Loss**.

step 7: 根据得到的用户-项目评分预测矩阵, 利用均方根误差 **RMSE** 衡量推荐精度.

### 3.4 时间复杂度分析

算法的时间复杂度主要消耗在计算目标函数、更新用户、项目潜在特征矩阵  $U$  和  $V$  上, 针对每一次迭代过程, 假设记录的评分数目是  $n_R$ , 则其复杂度为  $O(K^2 n_R + K^3 N + K^3 M)$ . 由于  $W$  的更新与文档潜在向量的计算相关, 更新  $W$  的复杂度由卷积层的计算决定, CNN 中更新所有权重和偏置变量的复杂度为  $O(n_{cal} M)$ . 算法最终的时间复杂度为  $O(K^2 n_R + K^3 N + K^3 M + n_{cal} M)$ , 可以看出该算法与  $n_R$  线性相关.

## 4 实验结果及分析

### 4.1 数据集

为了验证本文所提出算法的有效性和推荐效果,选择4个真实数据集,分别是来自Amazon的Amazon Instant Video(AIV)数据集和来自MovieLens的MovieLens-100k(ML-100k)、MovieLens-1M(ML-1M)、MovieLens-10M(ML-10M)三个数据集.这些数据集均包含用户对项目的评分信息,评分值是[1,5]范围内的整数.此外,AIV数据集还提供项目描述文档信息(如用户评价和项目简介等信息),而MovieLens原始数据集并没有项目描述文档信息,与项目对应的描述信息从IMDB获取.4个数据集的详细信息如表2所示.

表2 所用数据集的统计信息表

数据集	用户数目	项目数目	评分数目	用户评价数		评分密度/%
				最小	最大	
ML-100k	943	1682	100000	20	737	6.305
ML-1M	6040	3952	1000209	20	2314	4.190
ML-10M	71567	10681	10000054	20	7359	1.308
AIV	426924	23965	583937	1	277	0.006

### 4.2 实验平台及数据预处理

实验环境为2.7GHZ×4 Intel Core i5,8G内存,Ubuntu 16.04 LTS操作系统.本文提出的ITC-PMF算法基于Python和Keras框架实现.项目描述文档预处理:根据MovieLens评分数据中的项目ID和项目链接等信息,通过Python2.7从IMDB数据库中提取该项目对应的描述信息.项目描述文档预处理与NLP领域中处理自然语言数据类似,根据文献[15]预处理设置如下:1)设置源文档最多由300个单词组成;2)移除停用词;3)根据每个词的TF-IDF,移除语料库中文档频率高于0.5的词,之后选取TF-IDF最高的前8000个单词组成单词库;4)移除源文档中不包含于单词库的词.评分数据预处理:移除评价项目数小于1的用户和没有描述文档的项目.

### 4.3 评估准则

为验证本文所提出算法的推荐精确度,实验中采用5折交叉验证法,每次随机选取80%作为训练集,其余20%作为测试集.评估准则使用均方根误差(RMSE),是学术界针对评分预测方向衡量推荐性能的常用准则之一,定义为

$$RMSE = \sqrt{\frac{\sum_{u,q} (R_{uq} - \hat{R}_{uq})^2}{total}} \quad (26)$$

其中: $R_{uq}$ 为用户 $u$ 对项目 $q$ 的真实评分,表示用户 $u$ 对项目 $q$ 的预测评分;total为测试集中评分记录总数.RMSE值越小表示算法的预测准确度越高,其单位与式(26)中的分母相同,本文中其单位是评分偏差值.

### 4.4 实验结果

为验证本文所提出算法的有效性,检验融入信任因子和项目描述文档后对推荐效果的影响,将ITC-PMF算法与其他算法进行对比,并分析各个参数对推荐的影响.对比算法采用PMF算法、ConvMF算法和ConvMF+算法等本领域经典算法.PMF算法仅利用了用户对项目的评分信息,是最原始经典的算法;ConvMF算法除了利用用户项目评分信息之外融入项目描述信息;ConvMF+算法加入项目描述信息的同时使用GloVe模型训练词向量.实验设置算法中用户和项目潜在特征矩阵的分解维度 $K = 50$ ,词向量维度 $a = 200$ ,根据实验结果得到参数 $\lambda_u$ 和 $\lambda_v$ 在不同数据集和不同算法上推荐效果最佳时的设置如表3所示,其中为降低模型复杂度PMF算法设置参数 $\lambda_u = \lambda_v$ .

表3 不同数据集上参数 $\lambda_u$ 和 $\lambda_v$ 的设置

算法	ML-100k		ML-1M		ML-10M		AIV	
	$\lambda_u$	$\lambda_v$	$\lambda_u$	$\lambda_v$	$\lambda_u$	$\lambda_v$	$\lambda_u$	$\lambda_v$
ML-100k	0.1	0.1	0.1	0.1	0.1	0.1	15	15
ML-1M	10	10	100	10	10	100	1	100
ML-10M	10	10	100	10	10	100	10	10
AIV	10	10	100	10	10	100	1	100

与表3对应的不同算法在各个数据集上的RMSE测试结果如表4所示.由表4可见,在各个数据集上,本文提出的算法对比其他3个算法都有不同程度的提高,表明同时融入信任和项目描述信息在某种程度上可以捕获用户的偏好信息,从而为其提供更精确的推荐.

表4 不同算法在不同数据集上的RMSE

算法	数据集			
	ML-100k	ML-1M	ML-10M	AIV
PMF	0.9844	0.8786	0.8293	1.1660
ConvMF	0.9661	0.8751	0.7960	1.1401
ConvMF+	0.9589	0.8561	0.7936	1.1338
ITC-PMF	0.9148	0.8514	0.7864	1.1206



1) 不同数据集上各种算法的推荐质量对比.

图3(a)~图3(d)为4个算法在不同数据集上随着迭代次数变化而变化的RMSE曲线图,图3(e)~图3(h)分别为与之对应的数据集评价数量分布图,柱状图由左至右分别为项目数=20, 21~25, 26~50, 51~100, 101~150, 151~200, 201~250, 251~300, 301~350, 351~400, >400. 例如:图3(e)是ML-100k数据集的评分数量分布,其中横轴的每个区间表示评价项目个数,纵轴表示用户个数.由图

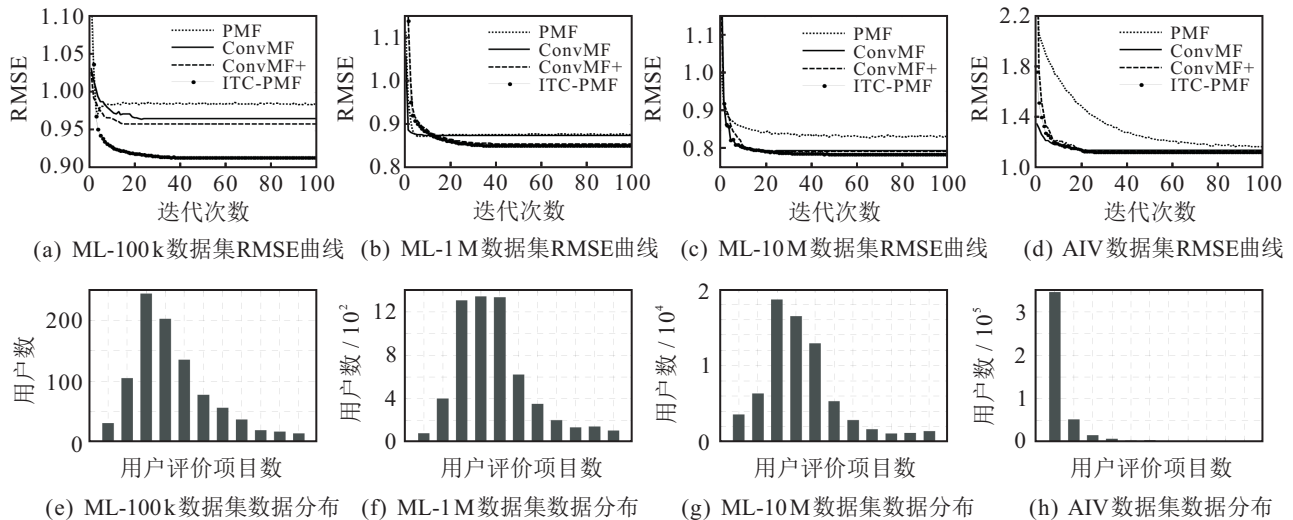


图3 不同数据集上的实验对比图

由图3(a)~图3(d)可见,在不同数据集上各个算法对应的RMSE曲线变化规律相似,随着迭代次数的增加, RMSE开始迅速下降,之后逐渐趋于收敛.而融入信任因子和项目文档信息的ITC-PMF算法较其他算法推荐效果均有提升.具体而言,对于MovieLens数据集,图3(a)是不同算法在ML-100k上的实验结果,可以看出与PMF算法相比,ConvMF算法和ConvMF+算法的推荐效果有所提高但并不突出.结合图3(e)和表2可知,ML-100k数据集的评分密度是6.305%,评分数目均匀,表明当评分数据足够多时,评分更能直观反映用户的偏好,而项目文档信息捕捉到的特征并没有为用户决策提供更加有效的推力.但是,ITC-PMF算法相对于PMF算法的提高,则表明在评分数据足够时,依靠用户评分相似度和行为相似度构建的信任因子能够挖掘用户深层的偏好特征,为用户决策时提供可信推荐.由图3(b)和图3(c)可见,随着评分密度降低,评分数据减少,对比PMF算法,ConvMF算法、ConvMF+算法和ITC-PMF算法的推荐效果提升较大,表明融入项目描述信息可以为项目潜在因子提供更加精确的特征,而ITC-PMF算法的推荐效果较之其他算法则体现出更好

3(e)可见,用户的最小评价数是20,评价数在26~50范围内的用户最多,用户评价数目区间主要集中在21~200之间.而ML-1M、ML-10M和AIV数据集的用户评价数目区间分别主要集中于21~400、20~400和1~10.另外,由图3(e)~图3(h)可见,来自MovieLens的3个数据集评分数目分布类似,用户评分数目最小值是20,分布较均匀;而相对MovieLens数据集,AIV数据集中用户评价数普遍较少,数据稀疏性显然更高.

的效果,表明信任因子和项目描述文档的同时加入能更好地提升推荐性能.由图3(h)和表2可知,AIV数据集的评分密度为0.006%,数据极其稀疏,且每个用户评分数目普遍较少,主要分布在1~10.结合图3(d)可知,在AIV数据集上ITC-PMF算法的推荐性能仍然稳定,表明即使在数据相当稀疏的情况下,ITC-PMF算法仍然可以通过信任因子捕获用户潜在的偏好信息,通过项目描述信息提取项目潜在特征,从而构成更加精确的用户项目潜在特征矩阵,为用户提供更加匹配的推荐.

2) 训练集密度对推荐质量的影响.

图4(a)为ITC-PMF算法在4个数据集上不同训练集密度下的实验结果.虽然数据集不同,但可以看出随着训练集密度的增加, RMSE值变化趋势类似.当密度由10%~50%变化时, RMSE值下降迅速,60%~80%之间RMSE值仍下降但变化幅度较小,表明当训练集密度很小时,即使算法融合了用户信任关系和项目描述信息,但训练后模型的推荐效果不敌训练集密度大时,这是因为训练集密度降低意味着用于构建模型的用户和项目数也在巨降,导致用户信任矩阵稀疏,项目描述信息可用率降低,推荐性能降低.

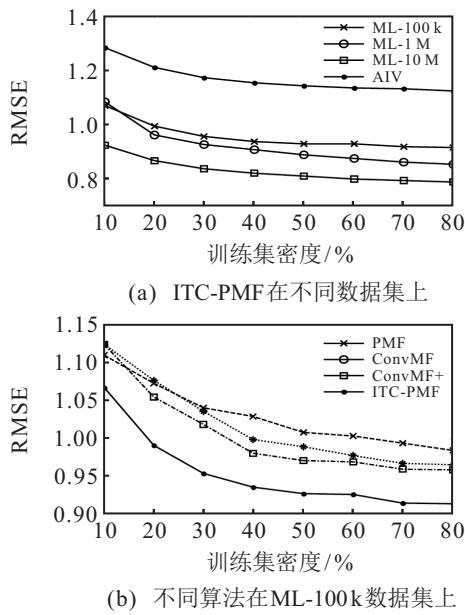


图4 训练集密度对推荐的影响

为了验证 ITC-PMF 算法在不同训练集密度下优于其他算法,在 ML-100k 数据集上进行对比实验,如图 4(b) 所示. 当训练集密度为 10% 时, PMF 优于 ConvMF 和 ConvMF+, ITC-PMF 效果最好. 表明当训练集密度很小时, 依靠评分和项目描述文档信息训练出的模型的推荐效果稍逊于 PMF, 因为当用户和项目很少时, 项目描述信息提取的项目潜在特征比重较大, 会引起轻微的过拟合现象; 而 ITC-PMF 较好的推荐效果则表明信任因子和项目描述信息的同时权衡会提高模型的鲁棒性. 同时, 可以看出, 在 ML-100k 数据集上不同训练集密度下, ITC-PMF 算法的推荐性能均高于其他算法, 表明 ITC-PMF 算法可以很好地缓解数据稀疏性和冷启动问题.

3) 信任因子  $\lambda_T$  对推荐的影响.

参数  $\lambda_T$  用来平衡信任因子对用户的影响,  $\lambda_T$  值越大表明信任用户的行为对用户的影响越大, 越小表明用户的偏好越依赖用户本身; 当然  $\lambda_T$  值并不是越大越好, 当  $\lambda_T$  值特别大时, 会导致过拟合现象, 即特征向量逼近训练阶段 RMSE 值较低时近邻用户的特征向量模型. 实验设置用来平衡用户偏好相似度和行为轨迹相似度的参数  $\lambda_T = 0.3$ , MovieLens 数据集上信任值阈值  $t = 0.6$ , 使得信任用户个数平均为 20; AIV 数据集的信任值阈值  $t = 0.3$ , 信任集用户个数平均为 10.

图 5 为不同数据集上不同  $\lambda_T$  对 RMSE 值的影响. 具体而言, 在 ML-100k 数据集上, 当  $\lambda_T = 1$  时, ITC-PMF 的推荐效果较好, 而在 ML-1M、ML-10M 和 AIV 数据集上,  $\lambda_T$  取值分别为 1、10 和 100 时, 对应的推荐效果较好. 可以看出, 针对不同数据集, 数据集越稀

疏,  $\lambda_T$  值越大, RMSE 值越低, 推荐效果越好; 而在相同数据集上却不一定. 实验结果表明, 信任因子通过平衡信任用户和用户自身偏好的影响可以缓解数据稀疏性问题.

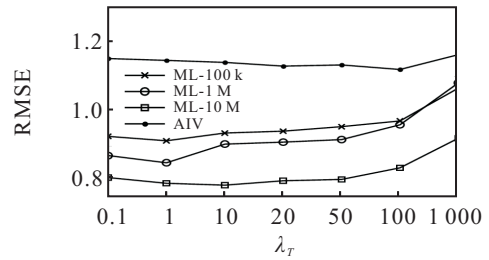


图5 不同数据集上参数  $\lambda_T$  的影响

5 结论

本文提出了一种融合信任和项目潜在的文档描述信息的 PMF 模型, 能够为用户提供更加精确和匹配的个性化推荐信息. 通过使用用户偏好相似度和行为轨迹相似度构建用户之间的信任关系, 使其更加符合社交网络的真实特征, 以缓解用户冷启动问题. 利用 CNN 提取项目描述文档的深度信息, 能够为项目潜在特征矩阵提供更加精确的特征, 缓解数据稀疏和项目冷启动问题. 实验结果表明, 将两种方法进行有效融合不仅缓解了数据稀疏性和冷启动问题, 而且提高了算法的鲁棒性. 为了更好地探索相关信息源对推荐的影响, 未来将进一步考虑融合多源异构数据(图片、视频和语音等), 挖掘有效信息, 提高推荐效果.

参考文献(References)

- [1] Peng F, Lu X, Ma C, et al. Multi-level preference regression for cold-start recommendations[J]. International Journal of Machine Learning and Cybernetics, 2017, 9(11): 1-14.
- [2] Ricci F, Rokach L, Shapira B. Recommender systems handbook[M]. New York: Springer, 2015: 1-34.
- [3] Park Y, Park S, Jung W, et al. Reversed CF: A fast collaborative filtering algorithm using a  $k$ -nearest neighbor graph[J]. Expert Systems with Applications, 2015, 42(8): 4022-4028.
- [4] Liu H F, Jing L P, Yu J. Survey of matrix factorization based recommendation methods by integrating social information[J]. Journal of Software, 2018, 29(2): 340-362.
- [5] Camacho L A G, Alves-Souza S N. Social network data to alleviate cold-start in recommender system: A systematic review[J]. Information Processing and Management, 2018, 54(4): 529-544.
- [6] Sedhain S, Braziunas D, Braziunas D, et al. Social collaborative filtering for cold-start recommendations[C]. Proceedings of the 8th ACM Conference on Recommender Systems. New York: ACM, 2014:

- 345-348.
- [7] Yan M, Sang J, Xu C, et al. A unified video recommendation by cross-network user modeling[J]. *ACM Transactions on Multimedia Computing Communications and Applications*, 2016, 12(4): 1-24.
- [8] Chen T, Zhu Q, Zhou M X, et al. Trust-based recommendation algorithm in social network[J]. *Journal of Software*, 2017, 28(3): 721-731.
- [9] Ma H, Yang H, Lyu M R, et al. SoRec: Social recommendation using probabilistic matrix factorization[C]. *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. Napa Valley: ACM, 2008: 931-940.
- [10] Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks[C]. *Proceedings of the 4th ACM Conference on Recommender Systems*. Barcelona: ACM, 2010: 135-142.
- [11] Ling G, Lyu M R, King I. Ratings meet reviews, a combined approach to recommend[C]. *Proceedings of the 8th ACM Conference on Recommender Systems*. Silicon Valley: ACM, 2014: 105-112.
- [12] McAuley J, Leskovec J. Hidden factors and hidden topics: understanding rating dimensions with review text[C]. *Proceedings of the 7th ACM Conference on Recommender Systems*. Hong Kong: ACM, 2013: 165-172.
- [13] Wang H, Wang N, Yeung D Y. Collaborative deep learning for recommender systems[C]. *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York: ACM, 2014: 1235-1244.
- [14] Kim Y. Convolutional neural networks for sentence classification[C]. *Proceedings of the 2014 Empirical Methods in Natural Language Processing (EMNLP)*. Eprint Arxiv, 2014: 1746-1751.
- [15] Kim D, Park C, Oh J, et al. Convolutional matrix factorization for document context-aware recommendation[C]. *Proceedings of the 10th ACM Conference on Recommender Systems*. Boston: ACM, 2016: 233-240.
- [16] Ma H, Zhou D, Liu C, et al. Recommender systems with social regularization[C]. *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*. New York: ACM, 2011: 287-296.
- [17] Bo Y, Yu L, Liu D, et al. Social collaborative filtering by trust[C]. *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*. Beijing: AAAI Press, 2013: 2747-2753.
- [18] Wang C, Blei D M. Collaborative topic modeling for recommending scientific articles[C]. *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Diego: ACM, 2011: 448-456.
- [19] Pan Y, Wu D, Olson D L. Online to offline (O2O) service recommendation method based on multi-dimensional similarity measurement[J]. *Decision Support Systems*, 2017, 103(11): 1-8.
- [20] Adomavicius G, Zhang J. Classification, ranking, and top- $K$  stability of recommendation algorithms[J]. *Inform Journal on Computing*, 2016, 28(1): 129-147.
- [21] Ahlgren P, Jarneving B, Rousseau R. Requirement for a cocitation similarity measure, with special reference to pearson's correlation coefficient[J]. *Journal of the American Society for Information Science and Technology*, 2003, 54(6): 550-560.
- [22] Seifoddini H, Djassemi M. The production data-based similarity coefficient versus Jaccard's similarity coefficient[J]. *Computers and Industrial Engineering*, 1991, 21(1/2/3/4): 263-266.
- [23] Salakhutdinov R, Mnih A. Probabilistic matrix factorization[C]. *Proceedings of the 20th International Conference on Neural Information Processing Systems*. Vancouver: Curran Associates Inc, 2007: 1257-1264.
- [24] Shen Y, He X, Gao J, et al. A latent semantic model with convolutional-pooling structure for information retrieval[C]. *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*. Shanghai: ACM, 2014: 101-110.
- [25] Dieleman S, Schrauwen B. Deep content-based music recommendation[C]. *International Conference on Neural Information Processing Systems*. Lake Tahoe: Curran Associates Inc., 2013: 2643-2651.
- [26] Pennington J, Socher R, Manning C. Glove: Global vectors for word representation[C]. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha: Association for Computational Linguistics, 2014: 1532-1543.

## 作者简介

王建芳(1979—),男,副教授,博士,从事推荐系统、深度学习、数据挖掘及大数据分析等研究, E-mail: wangjianfang@hpu.edu.cn;

苗艳玲(1992—),女,硕士生,从事数据挖掘和推荐算法的研究, E-mail: SkilaMiao@qq.com;

韩鹏飞(1992—),男,硕士生,从事深度学习和推荐算法的研究, E-mail: a106521ai@qq.com;

司马海峰(1982—),男,讲师,博士,从事模式识别和图像处理等研究, E-mail: smhf@hpu.edu.cn.

(责任编辑: 郑晓蕾)