

融合显式反馈与隐式反馈的协同过滤推荐算法*

陈碧毅¹, 黄玲¹, 王昌栋¹, 景丽萍²

¹(中山大学 数据科学与计算机学院, 广东 广州 510006)

²(北京交通大学 计算机与信息技术学院, 北京 100044)

通讯作者: 黄玲, E-mail: huanglinghl@hotmail.com



摘要: 显式反馈与隐式反馈相结合,可以有效提升推荐性能.但是现有的融合显式反馈与隐式反馈的推荐系统存在未能发挥隐式反馈数据缺失值反映用户隐藏偏好的能力,或者未能保留显式反馈数据反映用户偏好程度的能力的局限性.为了解决这个问题,提出了一种融合显式反馈与隐式反馈的协同过滤推荐算法.该算法分为两个阶段:第1阶段利用加权低秩近似处理隐式反馈数据,训练出隐式用户/物品向量;第2阶段引入了基线评估,同时将隐式用户/物品向量作为补充,通过显隐式用户/物品向量结合,训练得出用户对物品的预测偏好程度.该算法与多个典型算法在标准数据集上进行了实验比较,其可行性和有效性得到验证.

关键词: 推荐系统;协同过滤;矩阵分解;显式反馈;隐式反馈

中图法分类号: TP18

中文引用格式: 陈碧毅,黄玲,王昌栋,景丽萍.融合显式反馈与隐式反馈的协同过滤推荐算法.软件学报,2020,31(3):794-805.
<http://www.jos.org.cn/1000-9825/5897.htm>

英文引用格式: Chen BY, Huang L, Wang CD, Jing LP. Explicit and implicit feedback based collaborative filtering algorithm. Ruan Jian Xue Bao/Journal of Software, 2020,31(3):794-805 (in Chinese). <http://www.jos.org.cn/1000-9825/5897.htm>

Explicit and Implicit Feedback Based Collaborative Filtering Algorithm

CHEN Bi-Yi¹, HUANG Ling¹, WANG Chang-Dong¹, JING Li-Ping²

¹(School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China)

²(School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract: The combination of explicit and implicit feedback can effectively improve recommendation performance. However, the existing recommendation systems have some disadvantages in integrating explicit feedback and implicit feedback, i.e., the ability of implicit feedback to reflect hidden preferences from missing values is ignored or the ability of explicit feedback to reflect users' preferences is not fully utilized. To address this issue, this study proposes an explicit and implicit feedback based collaborative filtering algorithm. The algorithm is divided into two stages, where the first stage deals with implicit feedback data by weighted low rank approximation to train implicit user/item vectors, and the second stage introduces a baseline estimate and uses the implicit user/item vectors as supplementaries to the explicit user/item vectors. Through the combination of explicit and implicit user/item vectors, the

* 基金项目: 国家自然科学基金(61876193, 61822601, 61773050, 61632004); 广东省自然科学基金-杰出青年基金(2016A030306014); 广东特支计划“科技创新青年拔尖人才”(2016TQ03X542); 北京市自然科学基金(Z180006); 国家重点研发计划(2017YFC1703506); 中央高校基本科研业务费专项资金(2019JBZ110)

Foundation item: National Natural Science Foundation of China (61876193, 61822601, 61773050, 61632004); Guangdong Natural Science Funds for Distinguished Young Scholar (2016A030306014); Tip-top Scientific and Technical Innovative Youth Talents of Guangdong Special Support Program (2016TQ03X542); Beijing Natural Science Foundation of China (Z180006); National Key Research and Development Program of China (2017YFC1703506); Fundamental Research Funds for the Central Universities (2019JBZ110)

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: 2019-05-30; 修改时间: 2019-09-10, 2019-11-25; 采用时间: 2019-12-18; jos 在线出版时间: 2020-01-10

CNKI 网络优先出版: 2020-01-10 13:34:25, <http://kns.cnki.net/kcms/detail/11.2560.TP.20200110.1333.001.html>

predictions of users' preferences for items can be obtained by training. The proposed algorithm is compared with several typical algorithms on standard datasets, and the results confirm its feasibility and effectiveness.

Key words: recommendation system; collaborative filtering; matrix factorization; explicit feedback; implicit feedback

推荐系统作为解决信息过载问题的主要技术之一,近年来受到了学术界和工业界的广泛关注^[1-7].推荐系统中最常用的方法是协同过滤推荐方法^[3],如图 1 所示,仅利用用户历史行为信息,分析用户与物品之间的关系,从而给出满足用户个性化需求的推荐.用户的历史行为信息包括显式反馈与隐式反馈^[5]:显式反馈是指用户对物品的评分或者评价,可以直接反映用户偏好,并且细化了偏好的程度;隐式反馈指的是用户的购买、点击、收藏等行为^[7],无法直接反映用户偏好,但可用于挖掘用户偏好.现实生活中,用户往往很少主动对物品进行评分或者评价,很多场景下,甚至只能收集到隐式反馈数据,因此显式反馈数据十分稀疏,而隐式反馈数据十分丰富,并且避免了用户操作负担,较为容易获取.许多研究人员逐渐意识到隐式反馈数据的价值,将目光转到了基于隐式反馈的推荐方法上^[6-16].一些研究发现:显式反馈与隐式反馈之间存在一种互补关系^[5,17-19],将二者同时应用将有很大可能改善推荐效果.然而,现有的研究未能充分考虑显式反馈与隐式反馈的特性,未能充分发挥二者的作用,例如仅利用隐式反馈的正样本而忽视缺失值的作用,或者将显隐式反馈数据都转化为 0-1 之间的值,破坏了其原本结构.

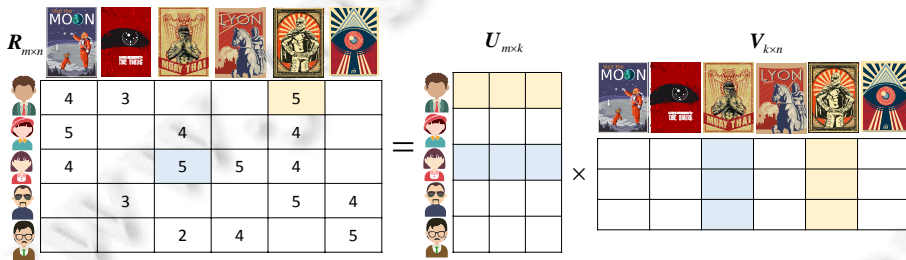


Fig.1 Collaborative filtering recommendation system

图 1 协同过滤推荐系统

针对上述问题,本文提出一种融合显式反馈与隐式反馈的推荐算法,算法分为两个阶段:第 1 阶段利用加权低秩近似处理隐式反馈数据,训练得出隐式用户/物品向量;第 2 阶段引入基线评估,同时将隐式用户/物品向量作为补充,通过显隐式用户/物品向量结合,训练得出用户对物品的预测偏好程度.实验结果表明,本方法显著地提高了推荐性能.

1 相关工作

1.1 基于隐因子模型的推荐

隐因子模型(latent factor model,简称 LFM)是十分常用的推荐方法,其思想是将用户和物品特征化,映射到一个共有的隐因子空间,使用矩阵分解方法将稀疏的用户-物品评分矩阵补全,由此预测用户对物品的兴趣度,具有很高的精确度^[5,6,20-28].由于评分矩阵具有高度稀疏性,通常是低秩的,因此隐因子空间的维度可以设置得很低而不会影响精确度,从而极大减小了存储复杂度.同时,特征矩阵的尺寸与用户和物品的数量呈线性相关,因此具有良好的可扩展性.由于上述优越性,隐因子模型受到了广泛的关注.Hofmann^[23]利用概率隐语义分析建立一个协同过滤模型.Yin 等人^[6]提出了潜在要素模型,利用降维技术提高了模型抗噪声能力.Salakhutdinov 和 Mnih^[24]提出了概率矩阵分解模型(probabilistic matrix factorization,简称 PMF).

1.2 基于深度学习的推荐

近几年来,由于深度学习具有强大的学习能力,深度学习方法逐渐被引入到推荐系统中,用以学习用户对物品的偏好^[7,29,30].神经网络被用于学习用户-物品隐因子表示与匹配函数之间复杂的映射关系.He 等人^[29]提

出了3种方法:广义矩阵分解(generalized matrix factorization,简称GMF)、多层感知机(multi-layer perceptron,简称MLP)和神经网络矩阵分解(neural matrix factorization,简称NeuMF),分别用不同的方式对用户-物品交互进行建模.Deng等人^[7]结合表示学习与匹配方程学习的优点,提出了DeepCF框架.然而,基于深度神经网络的模型面临严重的过拟合问题,并且具有高计算复杂度和高存储复杂度,无法适应大规模数据^[30].

1.3 隐式反馈推荐算法

相比于显式反馈数据,隐式反馈数据具有数据量大、数据更稠密、更稳定的特点,是由用户自然产生的,通常会真实地反映用户态度,具有很高的应用价值^[6-8].然而,隐式反馈数据也存在显著的缺陷^[8].

- 1) 缺少负反馈.隐式反馈数据只能反映用户是否对物品进行过操作,操作的频率可用于推测出用户喜欢的物品,却无法反映用户不喜欢哪些物品;
- 2) 无法表达偏好程度.隐式反馈的数值通常表示用户的操作频率,比如节目的观看次数、歌曲的播放次数、点击率等,数值越高,并不代表用户更喜欢该物品,而是表示该行为的可信度,因为一次性事件可能是偶然的,而高频事件则有更高的可信度;
- 3) 数据噪声.显式反馈数据都存在噪声数据,然而相比于评分明确的显式反馈数据,隐式反馈数据更容易包含噪声,只能用于猜测用户的偏好和真实动机.

针对隐式反馈中存在的显著缺陷,研究人员提出了许多基于隐式反馈的推荐方法,主要可以分为以下3类:基于评分预测的单类协同过滤算法、引入辅助信息的推荐以及基于排序的推荐.

- 1) 基于评分预测的单类协同过滤.Pan等人^[9]提出了加权的ALS算法(weighted alternating least squares,简称wALS),其核心思想是:将所有的缺失数据当作负样本,对所有样本进行加权.用户对物品有操作,这本身就反映了用户对物品有偏好,具有很高的可信度,因此其相关权重应设置为1.而用户对物品没有操作的情况,并不代表用户真的不喜欢这个物品,很有可能是用户没有见过这个物品,因此其可信度较低,相关权重应设置为小于1的值.wALS^[9]算法可以有效地从缺失值中挖掘出潜在的正、负样本.Hu等人^[8]提出了改进的因子分解模型,其核心思想是:给隐式反馈数据集中的正样本和负样本分别分配一个变化的信任权重;
- 2) 引入外部信息的推荐.由于隐式反馈具有显著的缺陷,因此仅使用隐式反馈进行推荐效果不佳.Li等人^[10]在传统的单类协同过滤模型的基础上引入用户的购买、搜索和浏览记录,利用增加的用户搜索特征,有效提升了推荐的准确度.Sindhwani等人^[11]引入了辅助矩阵,运用非负矩阵分解技术进行分解建模,提出了ldNFM.Wang等人^[12]在传统的矩阵分解模型上融入推荐对象的内容信息或者文本描述信息,提出了CTR模型,进一步缓解隐式反馈数据的类不平衡问题以及新对象的冷启动问题;
- 3) 基于排序的推荐.在信息检索领域,排序学习得到广泛的关注与快速发展,一些学者尝试将排序的思想应用到单类协同过滤问题上.Rendle等人^[13]提出了BPR模型,运用贝叶斯成对损失函数,研究了基于排序的单类协同过滤问题.Pan和Chen^[14]放弃了BPR^[13]模型中提出的用户之间相互独立的假设,提出了GBPR.Shi等人^[15]提出了CLiMF,通过直接优化平均倒数排名(mean reciprocal rank,简称MRR)指标,实现对推荐对象进行排序.Qiu等人^[16]通过研究不同种类的隐式反馈之间的相关关系,提出了基于排序技术的BPRH模型.

这些算法皆有效地缓解了隐式反馈的类不平衡问题以及缺少负反馈问题,可以有效地从缺失值中分类出正样本和负样本,一些研究可以根据隐式反馈的数值或者辅助信息对推荐对象进行排序.然而,隐式反馈的数值只能反映动作的可信度,而无法真实反映用户的偏好程度.因此,本文算法在利用隐式反馈数据挖掘用户隐藏偏好的同时,结合显式反馈数据准确反映用户偏好程度的优势,以给出更准确的推荐结果.

1.4 融合显式反馈的推荐算法

许多研究发现显式反馈与隐式反馈之间存在一种互补关系^[5,17-19],将二者同时应用有很大可能改善推荐效果.Koren^[5]提出将用户是否评过这个行为作为隐式反馈,利用用户评过所有物品的隐因子来代表用户,

将显式反馈与隐式反馈同时用于推荐任务,提出了非对称 SVD(asymmetric-SVD,简称 ASVD)和 SVD++。Li^[22]结合了 SVD++^[5]和 xCLiMF^[19]算法,通过最优化预期倒数排名(expected reciprocal rank,简称 ERR)指标,提出了融合显隐式反馈的个性化排序模型 MERR_SVD++。然而,上述算法仅利用了评分数据和隐式反馈的正样本进行训练,未能考虑到隐式反馈中缺失值的价值,未能缓解隐式反馈的缺少负反馈问题,因此未能充分发挥隐式反馈的作用。用户的意见不仅包括其评过分的物品,同时也由他没有评价或者操作过的内容组成,即缺失值也有其价值,以上算法皆采用了直接将缺失值当作未知的策略。与现有算法不同的是,本文算法对隐式反馈数据进行加权低秩处理,从缺失值中挖掘出隐藏的正负样本,能够有效缓解隐式反馈的缺少负反馈问题,从而更好地发挥隐式反馈数据反映用户隐藏偏好的能力。

Liu 等人^[18]考虑到了显隐式反馈的异质性,即显式反馈多为数值型的,而隐式反馈多为二元型的,为了消除二者的数值差异,将二者都转化为 0-1 之间的数值,并分别给它们设置不同的权重,提出了基于评分预测的矩阵分解模型 Co_rating。然而,将显式反馈数据转化为 0-1 之间的数值的做法,削弱了显式反馈数据反映用户偏好程度的能力,未能考虑到显式反馈数据的重要特性。与上述工作不同,本文算法在保留显式反馈与隐式反馈各自特性的基础上,对二者分别采用合适的方法进行训练,充分发挥了隐式反馈数据反映用户隐藏偏好的能力以及显式反馈数据反映用户偏好程度的能力,同时将从隐式反馈数据学习得出的信息作为显式反馈数据的补充,更大程度地提升了挖掘用户对物品隐藏偏好程度的能力。

2 融合显式反馈与隐式反馈的协同过滤推荐算法

2.1 问题定义

为了便于形式化描述,本文的符号标记见表 1。

Table 1 Symbol definition

表 1 符号定义

符号	描述
$A=[A_{ui}]_{m \times n}$	用户-物品操作矩阵
$P_{m \times k}$	隐式用户矩阵
$Q_{n \times k}$	隐式物品矩阵
$W_{m \times n}$	隐式权重矩阵
$X_{m \times d}$ $Y_{n \times d}$	由隐式权重矩阵 W 分解得到的低秩矩阵
$R=[r_{ui}]_{m \times n}$	用户-物品评分矩阵
$U_{m \times k}$	显式用户矩阵
$V_{n \times k}$	显式物品矩阵
$\kappa=\{(u,i) r_{ui} \neq 0\}$	评分已知的用户-物品对集合
$N=\{1, \dots, n\}$	矩阵 A 中的列索引集合
N_r	用户 r 操作过的物品在矩阵 A 中的列索引集合
b_{ui}	用户 u 对物品 i 的评分基线
μ	所有物品的评分均值
b_u	用户 u 评分的偏置
b_i	物品 i 评分的偏置
$\hat{R}=[\hat{r}_{ui}]_{m \times n}$	用户-物品预测偏好程度矩阵

假定有 m 个用户和 n 个物品。隐式反馈数据中,用户对物品的操作信息用矩阵 $A=[A_{ui}]_{m \times n}$ 表示, A_{ui} 代表用户 u 对物品 i 的操作信息。本文选用用户对物品进行评分的行为表示隐式反馈,因为用户对物品有评分的情况恰好反映了用户对该物品有一定偏好。因此,当用户 u 对物品 i 有评分时, $A_{ui}=1$;否则, $A_{ui}=0$ 。隐式用户偏好信息用隐式用户矩阵 $P_{m \times k}$ 表示, P_u 代表用户 u 的隐式因子表示;隐式物品信息用隐式物品矩阵 $Q_{n \times k}$ 表示, Q_i 代表物品 i 的隐式因子表示,实际的用户-物品操作矩阵 A 可由 PQ^T 近似表示。 $W_{m \times n}$ 为隐式权重矩阵,用于反映操作信息 $A_{m \times n}$ 的可信度。低秩矩阵 $X_{m \times d}$ 和 $Y_{n \times d}$ 的乘积 XY^T 用于近似表示稠密的隐式权重矩阵 $W_{m \times n}$ 。显式反馈数据中,用户对物品的评分信息用矩阵 $R=[r_{ui}]_{m \times n}$ 表示, r_{ui} 代表用户 u 对物品 i 的评分信息,评分越高,代表用户对该物品越

满意.例如,在五分制的电影评分系统下, $r_{ui} \in \{1,2,3,4,5\}$,评分越接近 5,代表用户越喜欢该电影;评分越接近 1,代表用户对该电影越不感兴趣.显式用户偏好信息用显式用户矩阵 $U_{m \times k}$ 表示, U_i 代表用户 u 的显式隐因子表示;显式物品信息用显式物品矩阵 $V_{n \times k}$ 表示, V_i 代表物品 i 的显式隐因子表示.评分 r_{ui} 已知的用户-物品对 (u,i) 存储于集合 $\kappa = \{(u,i) | r_{ui} \neq 0\}$ 中.用户-物品操作矩阵 A 中的列索引存储于集合 N 中.用户 r 操作过的物品在用户-物品操作矩阵 A 中的列索引存储于集合 N_r 中.用户 u 对物品 i 的评分特异性用评分基线 b_{ui} 表示,其由所有物品的评分均值 μ 、用户 u 评分的偏差 b_u 和物品 i 评分的偏差 b_i 之和构成,即 $b_{ui} = \mu + b_u + b_i$.推荐系统的目标是:给定一个用户 u 以及一个商品 i ,如果用户 u 对商品 i 的偏好程度未知,则系统会根据已知的一些信息来预测出用户 u 对商品 i 的偏好程度 \hat{r}_{ui} ;接着,根据每个用户的预测偏好程度由高到低对商品进行排序,将前 K 个商品推荐给用户.

2.2 EIFCF 的模型

隐因子模型作为协同过滤推荐中的一种常见方法,具有挖掘数据多维特征的潜力.因此,为了充分发挥隐式反馈数据反映用户隐藏偏好的能力,以及显式反馈数据反映用户对各种特征的偏好程度的能力,本文在考虑显隐式反馈数据的显著差异条件下,针对显隐式反馈数据各自的特性,对隐式反馈数据和显式反馈数据分别进行矩阵分解处理,同时将从隐式反馈数据中学习得出的用户隐藏偏好信息作为显式反馈数据的补充,提出了一种融合显式反馈与隐式反馈的协同过滤推荐算法(explicit and implicit feedback based collaborative filtering algorithm,简称 EIFCF).

EIFCF 算法主要分为两个阶段:第 1 阶段利用 gap-ALS(GALS)^[20] 算法快速处理隐式反馈数据,得到隐式用户向量 P 和隐式物品向量 Q ;第 2 阶段引入了基线评估,同时将第一阶段学习得到的隐式用户向量 P 和隐式物品向量 Q 作为补充,由隐式用户向量 P 和显式用户向量 U 共同组成用户向量,由隐式物品向量 Q 和显式物品向量 V 共同组成物品向量,利用显式反馈数据训练模型,预测得出用户对物品的偏好程度.

2.2.1 第 1 阶段

第 1 阶段对用户-物品操作矩阵采用矩阵分解方法,其思想是任何用户都有自己偏好的特征,每种物品包含的特征也各不相同,用户对各个特征的偏好程度构成了用户的隐向量,而物品在各个特征上的概率分布构成了物品的隐向量.实际的用户-物品操作矩阵 $A_{m \times n}$ 可以用低秩的隐式用户矩阵 $P_{m \times k}$ 和隐式物品矩阵 $Q_{n \times k}$ 的乘积 PQ^T 近似表示,由此可以得到损失函数如下:

$$Loss = \sum_{ui} (A_{ui} - P_u Q_i^T)^2 + \lambda_1 (\|P\|^2 + \|Q\|^2) \quad (1)$$

其中, k 代表隐式用户/物品向量的特征维度, λ_1 为正则化参数, P 的行向量代表一个用户对各个特征的偏好程度, Q 的行向量代表一个物品在各个特征上的概率分布.

在解决隐式反馈数据缺少负反馈的问题上,常用的思路是将所有的缺失值当作负样本,或者忽视所有缺失样本,仅使用正样本.而这两种策略都与实际情况有很大偏差,因为缺失值中可能存在正样本.因此,隐式反馈数据的处理学习 wALS^[9] 算法的思想从缺失值中挑选出潜在的正负样本.首先,设置 $A_{ui}=1$ 为正样本,由于用户对物品评过分恰好代表用户对该物品存在一定偏好,因此 $A_{ui}=1$ 代表正样本具有很强的可信度,设置其相关权重 $W_{ui}=1$;对于缺失值,定义它们有可能为负样本,对应的 $A_{ui}=0$,然而,用户没有对物品进行操作并不代表用户不喜欢该物品,很有可能是用户从未见过这个物品,因此 $A_{ui}=0$ 代表负样本的可信度较低,其相关权重应该较低,设置为 $W_{ui}=\delta \in [0,1]$,可得相关权重矩阵:

$$W_{ui} = \begin{cases} 1, & A_{ui} = 1 \\ \delta \in [0,1], & A_{ui} = 0 \end{cases} \quad (2)$$

当 $\delta=0$ 时,相当于将所有的缺失值当作未知,仅使用正样本的情况;当 $\delta=1$ 时,相当于将所有的缺失值当作负样本.因此,公式(1)可改写为如下形式:

$$Loss = \sum_{ui} W_{ui} ((A_{ui} - P_u Q_i^T)^2 + \lambda_1 (\|P_u\|^2 + \|Q_i\|^2)) \quad (3)$$

由于 $W_{m \times n}$ 和 $A_{m \times n}$ 皆为稠密的,损失函数(3)的计算复杂度为 $O(m \cdot n)$.为了解决该问题,采用 GALS^[20] 算法分解加权矩阵 $W_{m \times n}$,使用两个低秩矩阵的乘积 XY^T 近似加权矩阵 W ,其中正样本的权重为 1,得到:

$$\begin{aligned}
Loss &= \sum_{ui} \mathbf{X}_u \cdot \mathbf{Y}_i^T (\mathbf{P}_u \cdot \mathbf{Q}_i^T - A_{ui})^2 + \sum_{ui} \mathbf{X}_u \cdot \mathbf{Y}_i^T \lambda_1 (\|\mathbf{P}_u\|^2 + \|\mathbf{Q}_i\|^2) \\
&= \underbrace{\sum_{ui} \mathbf{X}_u \cdot \mathbf{Y}_i^T (\mathbf{P}_u \cdot \mathbf{Q}_i^T)^2}_{=:A} + \underbrace{\sum_{ui \in \mathcal{K}} ((\mathbf{P}_u \cdot \mathbf{Q}_i^T - 1)^2 - \mathbf{X}_u \cdot \mathbf{Y}_i^T (\mathbf{P}_u \cdot \mathbf{Q}_i^T)^2)}_{=:B} + \underbrace{\sum_{ui} \mathbf{X}_u \cdot \mathbf{Y}_i^T \lambda_1 (\|\mathbf{P}_u\|^2 + \|\mathbf{Q}_i\|^2)}_{=:C}
\end{aligned} \quad (4)$$

对公式(4)各项分别求偏导,同时令其导数为0,可求出隐式用户向量 \mathbf{P} ,见算法1中第1行~第15行.

算法 1. EIFCF.

输入:隐式物品向量 $\mathbf{Q}_{n \times k}$, $\mathbf{X}_{m \times d}$ 和 $\mathbf{Y}_{n \times d}$,用户-物品评分矩阵 $\mathbf{R}_{m \times n}$;

输出:用户-物品预测偏好程度矩阵 $\hat{\mathbf{R}}_{m \times n}$.

```

1:  $\mathbf{Y}_2 = \mathbf{Y}^T \mathbf{1}$ 
2: for all  $\tilde{d} \in \{1, \dots, |d|\}$  do
3:    $\hat{A}_{jk}^{(\tilde{d})} := \sum_{i \in N} Q_{ij} \cdot Q_{ik} \cdot Y_{i\tilde{d}}$ 
4: for rows  $r \in \{1, \dots, m\}$  do
5:   读取  $\mathbf{Q}$  中的  $r \in N_r$  列的数据到  $\pi_r(\mathbf{Q})$ 
6:    $\hat{\mathbf{B}} := \pi_r(\mathbf{Q})^T \pi_r(\mathbf{Q})$ 
7:    $\mathbf{q} := \pi(\mathbf{Q}) \mathbf{1}$ 
8:   读取  $\mathbf{Y}$  中的  $r \in N_r$  列的数据到  $\pi_r(\mathbf{Y})$ 
9:    $\hat{c} := \mathbf{X}_r (\mathbf{Y}_2 - \pi_r(\mathbf{Y})^T) \mathbf{1}$ 
10:  for all  $\tilde{d} \in \{1, \dots, |d|\}$  do
11:     $\tilde{\mathbf{B}}_{jk}^{(\tilde{d})} := \sum_{i \in N_r} Q_{ij} \cdot Q_{ik} \cdot Y_{i\tilde{d}}$ 
12:     $\mathbf{A}' := \sum_d \mathbf{X}_{rd} \cdot \hat{\mathbf{A}}^{(d)}$ 
13:     $\mathbf{B}' := \hat{\mathbf{B}} - \sum_d \mathbf{X}_{rd} \cdot \tilde{\mathbf{B}}^{(d)}$ 
14:     $\mathbf{C}' := \lambda (\hat{c} + |N_r|) \cdot \mathbf{E}$ 
15:     $\mathbf{P}_r := [(\mathbf{A}' + \mathbf{B}' + \mathbf{C}')^{-1} \mathbf{q}_r]^T$ 
16:  end for
17: 进行上述流程的对称实验,得隐式物品向量  $\mathbf{Q}$ 
18: 设定步长  $\gamma$ 
19: for  $i \in \{1, \dots, iteration\}$  do
20:   预测误差为  $e_{ui} = r_{ui} - \hat{r}_{ui}$ 
21:   for  $(u, i)$  in  $\kappa$  do
22:      $b_u \leftarrow b_u + \gamma (e_{ui} - \lambda_2 \cdot b_u)$ 
23:      $b_i \leftarrow b_i + \gamma (e_{ui} - \lambda_2 \cdot b_i)$ 
24:      $\mathbf{V}_i \leftarrow \mathbf{V}_i + \gamma (e_{ui} \cdot (\mathbf{P}_u + \mathbf{U}_u) - \lambda_2 \cdot \mathbf{V}_i)$ 
25:      $\mathbf{U}_u \leftarrow \mathbf{U}_u + \gamma (e_{ui} \cdot (\mathbf{Q}_i + \mathbf{V}_i) - \lambda_2 \cdot \mathbf{U}_u)$ 
26:   end for
27: end for
28: return  $\hat{\mathbf{R}}$ 

```

2.2.2 第2阶段

第2阶段对用户-物品评分矩阵采用矩阵分解方法,实际的评分矩阵 $\mathbf{R}_{m \times n}$ 可以由低秩的显式用户矩阵 $\mathbf{U}_{m \times k}$ 和显式物品矩阵 $\mathbf{V}_{n \times k}$ 的乘积 \mathbf{UV}^T 近似表示,其目标是使得对所有 \mathbf{R} 中的元素 $r_{ui} \neq 0$ 有:

$$Loss = \sum_{r_{ui} \neq 0} (r_{ui} - \mathbf{U}_u \cdot \mathbf{V}_i^T)^2 + \lambda_2 (\|\mathbf{U}\|^2 + \|\mathbf{V}\|^2) \quad (5)$$

事实上,当用户给物品评分或者进行评价时,除了考虑物品多大程度符合自身兴趣外,通常还会受到自身的评分习惯和该物品的平均评分的影响.例如:有些用户给分比较宽松,通常会比给分比较严格的人给出的评分要高,或者当某物品的平均评分较低时,用户也会倾向于给该物品较低的评分.因此引入评分基线评估^[21],用以衡量用户间和物品间的差异.假设用户 u 对物品 i 的评分的基线为 b_{ui} ,其可由所有物品的评分均值 μ 、用户 u 评分的偏置 b_u 和物品 i 评分的偏置 b_i 之和构成,即:

$$b_{ui} = \mu + b_u + b_i \quad (6)$$

例如:在五分制的电影评分系统中,整体的电影评分均值为 2.5 分,用户 U 给分比较严格,通常评分会比均值低 1 分,而电影 I 比较受欢迎,通常收到的评分会比均值高 1.5 分,那么用户 U 对电影 I 的评分的基线为 $2.5 - 1 + 1.5 = 3$ 分.

因此,对用户偏好程度的预测可表示为

$$\hat{r}_{ui} = \mu + b_u + b_i + \mathbf{U}_u \cdot \mathbf{V}_i^T \quad (7)$$

其中: μ 是直接由训练集统计得出的,代表全局评分偏置项,与具体的用户和物品无关; b_u 由机器学习训练得来,只与用户 u 有关,体现了用户特异性; b_i 同样由机器学习训练得来,只与物品 i 有关,体现了物品特异性.

将隐式训练部分得出的隐式用户向量 \mathbf{P} 和隐式物品向量 \mathbf{Q} 作为显式用户向量 \mathbf{U} 和显式物品向量 \mathbf{V} 的补充,公式(7)可改写为如下形式:

$$\hat{r}_{ui} = \mu + b_u + b_i + (\mathbf{P}_u + \mathbf{U}_u) \cdot (\mathbf{Q}_i + \mathbf{V}_i)^T \quad (8)$$

隐式用户向量 \mathbf{P} 和隐式物品向量 \mathbf{Q} 可以有效提供用户隐藏偏好信息,而显式用户向量 \mathbf{U} 和显式物品向量 \mathbf{V} 提供用户对物品的偏好程度信息.显式用户向量 \mathbf{U} 和显式物品向量 \mathbf{V} 的具体值将由显式反馈数据训练得出.

将公式(8)代入公式(5)中,则最终目标函数形式为

$$Loss = \sum_{(u,i) \in \kappa} (r_{ui} - \mu - b_u - b_i - (\mathbf{P}_u + \mathbf{U}_u) \cdot (\mathbf{Q}_i + \mathbf{V}_i)^T) + \lambda_2 (b_u^2 + b_i^2 + \|\mathbf{U}\|^2 + \|\mathbf{V}\|^2) \quad (9)$$

采用梯度下降法求解公式(9),见算法 1 中第 22 行~第 25 行.重复步骤 21~步骤 26,直至达到迭代次数.

2.3 算法复杂度分析

算法 1 的时间复杂度很大程度上受 $\hat{A}_{jk}^{(d)}$ 计算的影响,其在整个初始化过程中的时间复杂度为 $\mathcal{O}(n \cdot k^2 \cdot |d|)$.每一行计算的时间复杂度由 $\tilde{B}_{jk}^{(d)}$ 决定,其时间复杂度为 $\mathcal{O}(|N_i| \cdot k^2 \cdot |d|)$.相比之下, $\hat{B}, \hat{q}, \hat{c}, \hat{A}, \hat{B}', \hat{C}'$ 和 \hat{P}_r 对时间复杂度的影响几乎可以不计,因此,所有行的时间复杂度为 $\mathcal{O}(m \cdot |N_i| \cdot k^2 \cdot |d|)$.相比之下,第 2 阶段的训练时间复杂度对整体时间复杂度影响很小,因此算法 1 的时间复杂度为 $\mathcal{O}(m \cdot |N_i| \cdot k^2 \cdot |d|)$.可以看到,算法 1 的时间复杂度与数据集的规模呈线性相关,因此,算法具有良好的可扩展性.

3 实验结果与分析

在开源的 MovieLens 数据集上,融合显式反馈与隐式反馈的协同过滤推荐算法 EIFCF 与其他典型算法进行对比,实验结果验证了 EIFCF 的可行性和有效性.同时,本文还分析比较了一些重要参数对结果的影响.

3.1 数据集

实验采用 MovieLens 的数据集:ml-100k,ml-latest-small(简称 ml-ls),ml-1m 和 ml-10m,数据集的具体信息见表 2.所有数据集中每名用户至少拥有 20 条评分记录.

Table 2 Data set description

表 2 数据集描述

数据集	用户总数	物品总数	评分总数	稀疏度(%)
ml-100k	943	1 682	100 000	93.69
ml-ls	610	9 742	100 836	98.30
ml-1m	6 040	3 900	1 000 209	95.75
ml-10m	71 567	10 681	10 000 054	98.69

3.2 评价指标

推荐系统的任务是将用户最感兴趣的物品推荐给用户,用户偏好程度越高的物品,在推荐列表中的位置应该越靠前^[6],因此,实验采用 $F1(F1\text{-measure})$ 、平均精度均值(mean average precision,简称 MAP)和归一化折损累计增益(normalized discounted cumulative gain,简称 NDCG)这 3 种评价指标来评估推荐算法的质量。

3.3 实验结果与分析

本文比较了提出的 EIFCF 与以下 4 种推荐方法。

- 1) UserCF^[31]:一个经典的推荐算法,通过与目标用户相似的用户偏好信息,预测目标用户的偏好程度.通常被用作推荐系统的基线;
- 2) Bias LFM^[21]:经典的仅利用显式反馈的基于矩阵分解的推荐算法,没有引入隐式向量的 EIFCF 的第 2 阶段即为该算法;
- 3) GALS^[20]:该算法利用加权的思想,从隐式反馈数据缺失值中挖掘出隐藏的正、负样本.EIFCF 的第 1 阶段采用了该算法;
- 4) SVD++^[5]:该算法同时利用显、隐式反馈数据,将用户评过分的物品隐因子集合作为用户隐因子的补充.

本文根据各个对比算法的参考文献或者实验结果设置合适的参数,使得各对比算法达到最佳性能.本文分别在 4 个数据集上都随机抽取了 50% 的训练数据和 50% 的测试数据,分别表示为 ml-100k(0.5),ml-ls(0.5),ml-1m(0.5)和 ml-10m(0.5);以及随机抽取了 70% 的训练数据和 30% 的测试数据,分别表示为 ml-100k(0.3),ml-ls(0.3),ml-1m(0.3)和 ml-10m(0.3),按照各数据集的特点,设定了合适的 TopN 值。

由表 3 可以观察到,所提算法 EIFCF 明显优于 4 种对比算法.这是由于相比于 UserCF, Bias LFM 和 GALS 仅考虑显式反馈数据或隐式反馈数据,以及 SVD++ 仅考虑显式反馈数据和隐式反馈数据中的正样本,EIFCF 在考虑显式反馈数据反映用户偏好程度能力的基础上,同时考虑到了隐式反馈数据中缺失值的作用,能够从缺失值中挖掘出隐藏的正负样本,实现了对两种数据的有效融合,显著提升了推荐性能. EIFCF 的第 1 阶段采用了 GALS 算法,同时, Bias LFM 相当于没有引入隐式向量的 EIFCF 的第 2 阶段.而 EIFCF 算法明显优于上述两种算法,这表明了算法两个阶段对总体性能起到较大的提升作用。

Table 3 Performance comparison of EIFCF and baselines

表 3 EIFCF 与对比算法推荐效果比较

数据集	TopN	评价指标	UserCF	Bias LFM	GALS	SVD++	EIFCF	最小提升(%)	平均提升(%)
ml-100k (0.3)	5	F1	0.169 9	0.328 3	0.345 1	0.336 6	0.355 2	2.93	31.43
		MAP	0.108 8	0.311 2	0.330 0	0.314 4	0.334 5	1.36	55.67
		NDCG	0.108 7	0.220 4	0.220 6	0.222 6	0.230 3	3.46	31.05
	3	F1	0.119 1	0.193 0	0.209 3	0.197 6	0.224 1	7.07	31.19
		MAP	0.088 5	0.152 2	0.165 4	0.158 8	0.173 6	4.96	31.11
		NDCG	0.092 2	0.134 2	0.148 3	0.145 8	0.179 5	21.04	43.15
ml-100k (0.5)	10	F1	0.284 1	0.395 9	0.396 1	0.396 0	0.408 3	3.08	13.26
		MAP	0.169 9	0.319 3	0.320 1	0.325 5	0.329 2	1.14	25.21
		NDCG	0.138 0	0.195 3	0.194 3	0.195 6	0.197 5	0.97	11.72
	5	F1	0.168 8	0.189 6	0.212 3	0.189 9	0.214 8	1.18	13.71
		MAP	0.108 6	0.164 1	0.175 7	0.165 5	0.190 4	8.37	28.69
		NDCG	0.108 1	0.152 3	0.151 1	0.153 3	0.173 0	12.85	25.24
ml-ls (0.3)	5	F1	0.277 0	0.301 5	0.305 9	0.306 9	0.308 9	0.65	3.90
		MAP	0.186 1	0.280 1	0.284 5	0.287 3	0.297 1	3.41	18.39
		NDCG	0.171 7	0.220 9	0.212 6	0.224 1	0.228 0	1.74	11.25
	3	F1	0.182 0	0.195 6	0.192 5	0.196 9	0.205 5	4.37	7.27
		MAP	0.134 3	0.141 2	0.151 3	0.145 8	0.161 8	6.94	13.24
		NDCG	0.136 1	0.140 8	0.143 4	0.149 3	0.158 1	5.89	11.15
ml-ls (0.5)	10	F1	0.340 8	0.375 0	0.363 8	0.375 2	0.376 3	0.29	3.62
		MAP	0.221 7	0.305 3	0.294 3	0.306 9	0.310 2	1.08	12.00
		NDCG	0.168 2	0.178 7	0.178 3	0.178 9	0.181 8	1.62	3.35
	5	F1	0.160 3	0.184 3	0.193 4	0.186 9	0.194 4	0.52	7.82
		MAP	0.142 7	0.157 3	0.168 5	0.165 1	0.171 8	1.96	8.91
		NDCG	0.143 6	0.141 7	0.141 0	0.147 7	0.158 9	7.58	10.77

Table 3 Performance comparison of EIFCF and baselines (Continued)

表 3 EIFCF 与对比算法推荐效果比较(续)

数据集	TopN	评估指标	UserCF	Bias LFM	GALS	SVD++	EIFCF	最小提升(%)	平均提升(%)
ml-1m (0.3)	5	F1	0.180 2	0.271 9	0.256 7	0.272 4	0.286 7	5.25	20.37
		MAP	0.126 7	0.240 3	0.226 1	0.240 6	0.251 6	4.57	29.78
		NDCG	0.122 3	0.186 9	0.173 7	0.187 6	0.195 9	4.44	20.55
	3	F1	0.148 1	0.161 3	0.151 2	0.162 0	0.171 3	5.74	10.22
		MAP	0.118 8	0.116 0	0.115 2	0.116 8	0.130 4	9.76	11.75
		NDCG	0.119 0	0.115 7	0.111 0	0.116 1	0.126 8	6.55	9.90
ml-1m (0.5)	10	F1	0.269 1	0.316 2	0.319 4	0.324 3	0.334 6	3.18	9.52
		MAP	0.168 8	0.241 2	0.237 9	0.245 1	0.248 1	1.22	13.84
		NDCG	0.136 1	0.153 8	0.154 9	0.157 2	0.162 5	3.37	8.33
	5	F1	0.187 4	0.171 2	0.165 5	0.177 1	0.195 0	4.06	11.47
		MAP	0.132 9	0.138 0	0.128 9	0.140 0	0.156 6	11.86	16.16
		NDCG	0.127 6	0.123 7	0.116 3	0.126 4	0.140 3	9.95	13.75
ml-10m (0.3)	5	F1	0.193 4	0.328 3	0.315 0	0.334 9	0.337 5	0.78	21.31
		MAP	0.117 9	0.301 0	0.291 1	0.307 3	0.316 4	2.96	46.28
		NDCG	0.118 3	0.221 8	0.230 6	0.228 4	0.232 6	0.87	26.05
	3	F1	0.141 8	0.214 6	0.198 0	0.215 1	0.216 3	0.56	15.78
		MAP	0.097 9	0.165 7	0.151 3	0.168 3	0.169 5	0.71	22.04
		NDCG	0.103 0	0.158 2	0.145 2	0.159 7	0.160 5	0.50	17.08
ml-10m (0.5)	10	F1	0.286 8	0.381 5	0.368 4	0.381 5	0.383 3	0.47	9.66
		MAP	0.135 9	0.305 2	0.290 9	0.305 2	0.307 5	0.75	33.37
		NDCG	0.138 1	0.185 2	0.177 0	0.185 2	0.194 1	4.81	14.96
	5	F1	0.192 1	0.204 2	0.187 0	0.208 4	0.209 1	0.34	5.85
		MAP	0.117 2	0.171 4	0.154 6	0.172 6	0.173 8	0.70	15.70
		NDCG	0.117 6	0.147 4	0.135 0	0.149 5	0.152 2	1.81	11.81

3.4 重要参数的影响

EIFCF 算法中包含两个重要的超参数:缺失值的相关权重 δ 和用户/物品向量的维度 k ,实验进一步研究了这两个超参数对推荐结果的影响.

在 EIFCF 的第 1 阶段, $A_{ij}=0$ 的相关权重被定义为 $W_{ij}=\delta\in[0,1]$.正如前面所述,将该权重设置为 0,表示将所有缺失值当作未知值,仅利用正样本进行训练;将该权重设置为 1,表示将所有缺失值当作负样本.可见,该权重值的取值对实验结果会产生重要影响.实验在数据集 ml-1s,ml-100k 和 ml-1m 的不同测试集切割比条件下进行,其中:在 ml-1s(0.5),ml-100k(0.5)和 ml-1m(0.5)数据集下,TopN 取 5;在 ml-1s(0.3),ml-100k(0.3)和 ml-1m(0.3)数据集下,TopN 取 3.实验结果如图 2 所示.

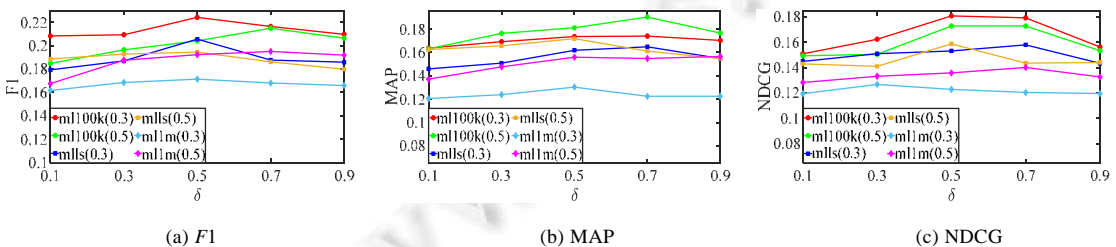


Fig.2 Effect of δ on performance

图 2 δ 对性能的影响

由图 2 可以得到如下结论:绝大多数情况下,缺失值的相关权重取值偏小或者偏大,推荐效果都不好.正如前面所述:当缺失值的相关权重为 0 时,相当于将所有的缺失值当作未知,仅利用已知项来进行计算,对比的算法 SVD++和 Bias LFM 正是采用了这种策略;而当缺失值的相关权重为 1 时,相当于将所有的缺失值当作负样本,显然这是不符合实际情况的.同时,绝大多数情况下, $\delta=0.9$ 的结果优于 $\delta=0.1$ 的结果,这是因为用户评过分的物品数量是远远小于物品总量的,用户喜欢的物品种类也远远小于所有物品种类,因此缺失样本中有很大一部分属

于负样本.另外,在数据集 ml-100k(0.5)和 ml-1m(0.5)中, $\delta=0.7$ 时推荐效果最佳;在余下的数据集中, $\delta=0.5$ 时推荐效果最佳.由此可以推断出,对缺失值进行加权的策略是优于将所有的缺失值当作未知和将所有的缺失值当作负样本的策略的.

参数 k 表示用户向量和物品向量的维度,其取值也会对实验结果产生一定影响.实验同样在数据集 ml-ls, ml-100k 和 ml-1m 的不同测试集切割比条件下进行,其中:在 ml-ls(0.5),ml-100k(0.5)和 ml-1m(0.5)数据集下,TopN 取 5;在 ml-ls(0.3),ml-100k(0.3)和 ml-1m(0.3)数据集下,TopN 取 3.实验结果如图 3 所示.

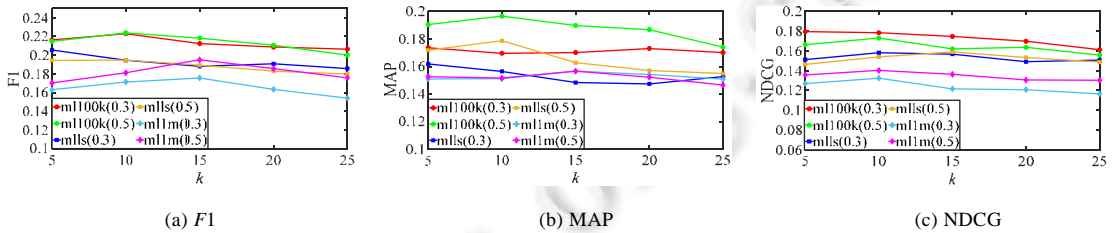


Fig.3 Effect of k on performance

图 3 k 对性能的影响

由图 3 可以看到,绝大多数情况下,在一定范围内,随着用户和物品向量的维度 k 的增大,推荐性能有所提升.这是因为隐向量的维度越高,能储存的信息越多,从而能更大程度地挖掘到隐含信息.然而 k 越大,越容易受到个别误差的干扰,从而导致推荐性能的下降.而实验所用的数据是高度稀疏的,即用户评过分的物品数量很少,因此 k 不宜取得过高.在 ml-ls(0.3)数据集下, k 取 5 时推荐性能最佳;在 ml-100k(0.3),ml-100k(0.5)以及 ml-ls(0.5)数据集下, k 取 10 时推荐性能最佳;在 ml-1m(0.3)和 ml-1m(0.5)数据集下, k 取 15 时推荐性能最佳.

4 总 结

针对现有融合显式反馈与隐式反馈的推荐算法存在的局限性,即未能发挥隐式反馈数据缺失值反映用户隐藏偏好的能力,或者未能保留显式反馈数据反映用户偏好程度的能力,本文提出了融合显式反馈与隐式反馈的协同过滤推荐算法 EIFCF.该算法充分考虑显式反馈数据与隐式反馈数据存在的显著差异,首先利用加权矩阵分解的思想从隐式反馈数据缺失值中挖掘出隐藏正负样本,克服了隐式反馈数据缺少负样本的问题,更大程度发挥隐式反馈数据的反映用户隐藏偏好信息的能力;接着,将训练得出的信息作为显式反馈数据的补充,缓解了显式反馈数据的高度稀疏性问题,同时增强模型从显式反馈数据中学习用户偏好程度信息的能力.实验结果验证了本文算法的可行性与有效性.

本文在融合显式反馈与隐式反馈的推荐场景下,为了更好地发挥两种反馈数据的不同作用,即隐式反馈数据反映用户隐藏偏好的作用以及显式反馈数据反映用户偏好程度的作用,针对两种反馈数据的特性,分别对其采用了合适的训练方法.下一步,我们将尝试对隐式反馈数据采用更先进的加权策略,并尝试引入更先进的方法来训练模型,探索更好地促进隐式反馈与显式反馈数据协同合作的方法.此外,现实生活中隐式反馈的种类十分多样,我们将尝试利用更多种类的隐式反馈数据,进一步缓解数据稀疏问题的影响.

References:

- [1] Liu LY, Du XZ, Zhu L, Shen FM, Huang Z. Learning discrete hashing towards efficient fashion recommendation. *Data Science and Engineering*, 2018,3(4):307–322. [doi: 10.1007/s41019-018-0079-z]
- [2] Gao DW, Tong YX, She JY, Song TS, Chen L, Xu K. Top- k team recommendation and its variants in spatial crowdsourcing. *Data Science and Engineering*, 2017,2(2):136–150. [doi: 10.1007/s41019-017-0037-1]
- [3] Wang CD, Deng ZH, Lai JH, Yu PS. Serendipitous recommendation in e-commerce using innovator-based collaborative filtering. *IEEE Trans. on Cybernetics*, 2019,49(7):2678–2692. [doi: 10.1109/TCYB.2018.2841924]

- [4] Srivastava R, Palshikar GK, Chaurasia S, Dixit A. What's next? A recommendation system for industrial training. *Data Science and Engineering*, 2018,3(3):232–247. [doi: 10.1007/s41019-018-0076-2]
- [5] Koren Y. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In: *Proc. of the 14th ACM Int'l Conf. on Knowledge Discovery and Data Mining*. New York: ACM, 2008. 426–234. [doi: 10.1145/1401890.1401944]
- [6] Yin J, Wang ZS, Li Q, Su WJ. Personalized recommendation based on large-scale implicit feedback. *Ruan Jian Xue Bao/Journal of Software*, 2014,25(9):1953–1966 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/4648.htm> [doi: 10.13328/j.cnki.jos.004648]
- [7] Deng ZH, Huang L, Wang CD, Lai JH, Yu PS. DeepCF: A unified framework of representation learning and matching function learning in recommender system. In: *Proc. of the 33rd AAAI Conf. on Artificial Intelligence*. Palo Alto: AAAI, 2019. 61–68. [doi: 10.1609/aaai.v33i01.330161]
- [8] Hu YF, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets. In: *Proc. of the 2008 8th IEEE Int'l Conf. on Data Mining*. Piscataway: IEEE, 2008. 263–272. [doi: 10.1109/ICDM.2008.22]
- [9] Pan R, Zhou YH, Cao B, Liu NN, Lukose R, Scholz M, Yang Q. One-class collaborative filtering. In: *Proc. of the 2008 8th IEEE Int'l Conf. on Data Mining*. Piscataway: IEEE, 2008. 502–511. [doi: 10.1109/ICDM.2008.16]
- [10] Li YN, Zhai CX, Chen Y. Improving one-class collaborative filtering by incorporating rich user information. In: *Proc. of the 19th ACM Int'l Conf. on Information and Knowledge Management*. New York: ACM, 2010. 959–968. [doi: 10.1145/1871437.1871559]
- [11] Sindhvani V, Bucak SS, Hu JY, Mojsilovic A. One-class matrix completion with low-density factorizations. In: *Proc. of the 2010 IEEE Int'l Conf. on Data Mining*. Piscataway: IEEE, 2010. 1055–1060. [doi: 10.1109/ICDM.2010.164]
- [12] Wang H, Chen BY, Li WJ. Collaborative topic regression with social regularization for tag recommendation. In: *Proc. of the 23rd Int'l Joint Conf. on Artificial Intelligence*. Palo Alto: AAAI, 2013. 2719–2725. [doi: 10.13140/2.1.3004.7689]
- [13] Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L. BPR: Bayesian personalized ranking from implicit feedback. In: *Proc. of the 25th Conf. on Uncertainty in Artificial Intelligence*. Arlington: AUAI, 2009. 452–461.
- [14] Pan WK, Chen L. GBPR: Group preference based Bayesian personalized ranking for one-class collaborative filtering. In: *Proc. of the 23rd Int'l Joint Conf. on Artificial Intelligence*. Palo Alto: AAAI, 2013. 2691–2697.
- [15] Shi Y, Karatzoglou A, Baltrunas L, Larson M, Oliver N, Hanjalic A. CLiMF: Learning to maximize reciprocal rank with collaborative less-is-more filtering. In: *Proc. of the 6th ACM Conf. on Recommender Systems*. New York: ACM, 2012. 139–146. [doi: 10.1145/2365952.2365981]
- [16] Qiu HH, Liu Y, Guo GB, Sun Z, Zhang J, Nguyen HT. BPRH: Bayesian personalized ranking for heterogeneous implicit feedback. *Information Sciences*, 2018,453:80–98. [doi: 10.1016/j.ins.2018.04.027]
- [17] Wang D, Chen Z, Yue WJ, Gao X, Wang F. Probabilistic matrix factorization recommendation with explicit and implicit feedback. *Journal of Computer Applications*, 2015,35(9):2574–2578 (in Chinese with English abstract). [doi: 10.11772/j.issn.1001-9081.2015.09.2574]
- [18] Liu NN, Xiang EW, Zhao M, Yang Q. Unifying explicit and implicit feedback for collaborative filtering. In: *Proc. of the 19th ACM Int'l Conf. on Information and Knowledge Management*. New York: ACM, 2010. 1445–1448. [doi: 10.1145/1871437.1871643]
- [19] Shi Y, Karatzoglou A, Baltrunas L, Larson M, Hanjalic A. xCLiMF: Optimizing expected reciprocal rank for data with multiple levels of relevance. In: *Proc. of the 7th ACM Conf. on Recommender Systems*. New York: ACM, 2013. 431–434. [doi: 10.1145/2507157.2507227]
- [20] Pan R, Scholz M. Mind the gaps: Weighting the unknown in large-scale one-class collaborative filtering. In: *Proc. of the 15th ACM Int'l Conf. on Knowledge Discovery and Data Mining*. New York: ACM, 2009. 667–676. [doi: 10.1145/1557019.1557094]
- [21] Koren Y. Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Trans. on Knowledge Discovery from Data*, 2010,4(1):1–24. [doi: 10.1145/1644873.1644874]
- [22] Li G, Chen Q. Exploiting explicit and implicit feedback for personalized ranking. *Mathematical Problems in Engineering*, 2016, 1–11. [doi: 10.1155/2016/2535329]
- [23] Hofmann T. Latent semantic models for collaborative filtering. *ACM Trans. on Information Systems*, 2004,22(1):89–115. [doi: 10.1145/963770.963774]

- [24] Salakhutdinov R, Mnih A. Probabilistic matrix factorization. In: Proc. of the 20th int'l Conf. on Neural Information Processing Systems. New York: Curran Associates Inc., 2007. 1257–1264.
- [25] Wu B, Lou ZZ, Ye YD. Co-Regularized matrix factorization recommendation algorithm. Ruan Jian Xue Bao/Journal of Software, 2018,29(9):2681–2696 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/5274.htm> [doi: 10.13328/j.cnki.jos.005274]
- [26] Liu HF, Jing LP, Yu J. Survey of matrix factorization based recommendation methods by integrating social information. Ruan Jian Xue Bao/Journal of Software, 2018,29(2):340–362 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/5391.htm> [doi: 10.13328/j.cnki.jos.005391]
- [27] Lin CY, Wang LC, Tsai KH. Hybrid real-time matrix factorization for implicit feedback recommendation systems. IEEE Access, 2018,6:21369–21380. [doi: 10.1109/ACCESS.2018.2819428]
- [28] He XN, Zhang HW, Kan MY, Chua TS. Fast matrix factorization for online recommendation with implicit feedback. In: Proc. of the 39th Int'l Conf. on Research and Development in Information Retrieval. New York: ACM, 2016. 549–558. [doi: 10.1145/2911451.2911489]
- [29] He XN, Liao LZ, Zhang HW, Nie LQ, Hu X, Chua TS. Neural collaborative filtering. In: Proc. of the 26th Int'l World Wide Web Conf. New York: ACM, 2017. 173–182. [doi: 10.1145/3038912.3052569]
- [30] Xi WD, Huang L, Wang CD, Zheng YY, Lai JH. BPAM: Recommendation based on BP neural network with attention mechanism. In: Proc. of the 28th Int'l Joint Conf. on Artificial Intelligence. California: IJCAI, 2019. 3905–3911. [doi: 10.24963/ijcai.2019/542]
- [31] Herlocker JL, Konstan JA, Borchers A, Riedl J. An algorithmic framework for performing collaborative filtering. In: Proc. of the 22nd Annual Int'l Conf. on Research and Development in Information Retrieval. New York: ACM, 1999. 230–237.

附中文参考文献:

- [6] 印鉴,王智圣,李琪,苏伟杰.基于大规模隐式反馈的个性化推荐.软件学报,2014,25(9):1953–1966. <http://www.jos.org.cn/1000-9825/4648.htm> [doi: 10.13328/j.cnki.jos.004648]
- [17] 王东,陈志,岳文静,高翔,王峰.基于显式与隐式反馈信息的概率矩阵分解推荐.计算机应用,2015,35(9):2574–2578. [doi: 10.1609/aaai.v33i01.330161]
- [25] 吴宾,姜铮铮,叶阳东.联合正则化的矩阵分解推荐算法.软件学报,2018,29(9):2681–2696. <http://www.jos.org.cn/1000-9825/5274.htm> [doi: 10.13328/j.cnki.jos.005274]
- [26] 刘华锋,景丽萍,于剑.融合社交信息的矩阵分解推荐方法研究综述.软件学报,2018,29(2):340–362. <http://www.jos.org.cn/1000-9825/5391.htm> [doi: 10.13328/j.cnki.jos.005391]



陈碧毅(1996—),女,广东汕头人,硕士生,主要研究领域为推荐系统,机器学习.



王昌栋(1984—),男,博士,副教授,博士生导师,CCF专业会员,主要研究领域为数据挖掘,机器学习.



黄玲(1985—),女,博士生,主要研究领域为数据挖掘,机器学习.



景丽萍(1978—),女,博士,教授,博士生导师,CCF专业会员,主要研究领域为机器学习,高维数据表示及其在人工智能领域中的应用.