

联合成对排序的物品推荐模型

吴宾, 陈允, 孙中川, 叶阳东

(郑州大学信息工程学院, 河南 郑州 450001)

摘要: 现有的推荐模型大多仅从用户角度进行建模, 忽略了物品的功能关系对用户购买决策的影响。从用户和物品这2个角度, 同时考虑用户-物品之间的交互关系和物品-物品之间的功能关系, 提出了联合成对排序的推荐模型。考虑正样本的排名位置和负采样策略直接影响模型收敛速度, 构建一种排序感知的学习算法, 用于求解所提模型的参数。实验结果表明, 与当前主流推荐算法相比, 该算法在多个评价指标上具有明显的性能优势。

关键词: 物品推荐; 成对排序; 协同过滤; 隐式反馈; 矩阵分解

中图分类号: TP181

文献标识码: A

doi: 10.11959/j.issn.1000-436x.2019137

Co-pairwise ranking model for item recommendation

WU Bin, CHEN Yun, SUN Zhongchuan, YE Yangdong

School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China

Abstract: Most of existing recommendation models constructed pairwise samples only from a user's perspective. Nevertheless, they overlooked the functional relationships among items--A key factor that could significantly influence user purchase decision-making process. To this end, a co-pairwise ranking model was proposed, which modeled a user's preference for a given item as the combination of user-item interactions and item-item complementarity relationships. Considering that the rank position of positive sample and the negative sampler had a direct impact on the rate of convergence, a rank-aware learning algorithm was devised for optimizing the proposed model. Extensive experiments on four real-world datasets are conducted to evaluate of the proposed model. The experimental results demonstrate that the devised algorithm significantly outperforms a series of state-of-the-art recommendation algorithms in terms of multiple evaluation metrics.

Key words: item recommendation, pairwise ranking, collaborative filtering, implicit feedback, matrix factorization

1 引言

随着网络信息的爆炸式增长, 如何从海量数据中帮助用户快速获取所需信息, 已成为信息检索领域亟待解决的问题。相比于传统的搜索引擎技术, 推荐技术因不需要用户的明确需求便能主动向用户提供个性化服务而成为缓解这一问题的重要工具^[1-3]。近些年, 众多网站已将推荐技术作为系统的必备组

件, 如亚马逊商品推荐、Netflix 电影推荐等。推荐技术不仅可以增加用户的参与度及对系统的信任度, 而且能够为商家带来可观的收益^[4]。

在现有的推荐技术中, 协同过滤^[5]技术是当前电子商务系统中应用最为广泛的技术, 其核心思想是根据目标用户的历史行为记录来寻找兴趣相投的近邻用户, 综合考虑这些用户的兴趣爱好, 帮助目标用户发现可能感兴趣的物品。用户的历史行为

收稿日期: 2018-11-22; 修回日期: 2019-05-14

通信作者: 叶阳东, yeyd@zzu.edu.cn

基金项目: 国家重点研发计划基金资助项目(No.2018YFB1201403); 国家自然科学基金资助项目(No.61772475, No.61502434)

Foundation Items: The National Key Research and Development Program of China (No.2018YFB1201403), The National Natural Science Foundation of China (No.61772475, No.61502434)

记录通常可分为两类：1) 显式反馈，即评分信息，例如亚马逊网站用户对所购买的物品打出 1~5 的评分，该评分能够明确地表达用户对物品的喜好程度；2) 隐式反馈，例如购买、点击、浏览等信息，其不能明确表达用户对物品的喜好程度，但在一定程度上反映了用户感兴趣的物品。早期的协同过滤技术主要基于显式反馈来处理推荐系统中的评分预测任务^[6]。相比不易收集的评分信息，隐式反馈的形式多样性和易获取的特性使基于隐式反馈的推荐模型更具实用性。近些年，基于隐式反馈信息的建模方法通常可分为两类^[7]：单值排序 (pointwise ranking)^[8-9] 和成对排序 (pairwise ranking)^[10-12]。单值排序方法通常潜在地假设用户已交互的物品为正样本且所有未交互物品为负样本。与单值排序方法不同，成对排序方法基于一个更弱的假设^[10]，即用户对已交互物品的喜好强于未交互的物品。相比单值排序方法优化用户对物品的绝对偏好，成对排序方法优化物品之间相对排名位置的建模方式显得更加合理。

在众多基于成对排序的推荐模型中，贝叶斯个性化排序 (BPR, Bayesian personalized ranking)^[10] 因具有理论完备、精度较高、易于实现等优点受到了学术界和工业界的广泛关注。当前已有不少推荐算法扩展了该模型^[13-15]，如融合社会关系的 SBPR (social Bayesian personalized ranking)^[13]、融合多级反馈信息的 BPR++^[14]、融合用户浏览信息的 BPR+view^[11] 等。上述模型仅从用户角度来构建用户-物品样本对，忽略了从物品角度建模物品之间内在的功能关系。在现实生活中，用户在做出购买决策时，不仅受自身喜好的影响，还受物品在功能上互补关系的影响因素。在图 1 手机及配件推荐场景中，Lily 已经购买了手机、耳机、自拍杆等商品，亚马逊网站向其推荐与已购买物品存在功能互补关系的手机壳可能更为合适；在宠物用品推荐场景中，Jason 的历史购买列表中有狗粮、宠物牵引绳、宠物垫子、宠物项圈等商品，此时系统向用户推荐宠物沐浴露更为合适。图 1 所示的 2 个推荐场景通过考虑商品之间的互补关系，这不仅避免了产生重复性推荐，同时满足了用户的当前需求。

对于成对排序方法而言，推荐模型的学习通常需要选取合适的正负样本对。在现实商业推荐场景中，用户与物品之间的历史交互数据通常具

有长尾分布的特性^[15]，这使负样本的选取对于模型的精度和收敛性都有很大影响^[16]。在采样过程中，需要选取一个比正样本具有更高排名的负样本，从而能够有效地更新参数并训练模型^[17]。原始 BPR 模型基于均匀采样策略构建正负样本对，该策略未考虑物品流行度和上下文信息的影响，同时忽略了正负样本的相对排名位置。当选取的正样本比负样本的排名位置靠前时，该正负样本对用于模型训练的贡献甚微。文献[16]提出了自适应采样策略，该策略每次迭代时根据当前正样本排名选取负样本，每隔固定的迭代次数更新一次排名信息。相比均匀采样策略，该策略的收敛更快，但忽略了正样本当前排名位置对于学习模型梯度的影响。

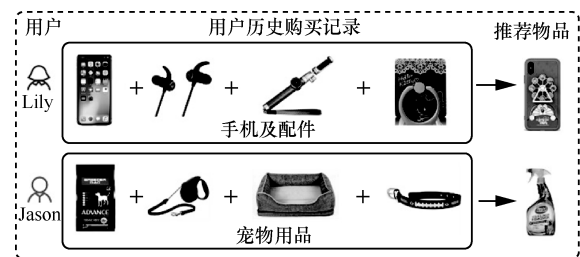


图 1 亚马逊网站的商品推荐案例

针对上述问题，本文从用户和物品 2 个角度，同时考虑用户-物品之间的交互关系和物品之间在功能上的互补关系，提出一种联合成对排序的推荐模型。考虑到正样本的排名位置和负样本的选取将直接影响模型精度及收敛速度，本文进一步构建了一种新颖的排序感知策略，使每次迭代均能合理选取正负样本对并根据正样本的当前排名位置动态控制模型学习梯度。最后基于该策略设计了一种高效的协同过滤推荐 (CPR, co-pairwise ranking) 学习算法，用于求解所提出模型的参数。

本文主要贡献如下。

- 1) 依据用户与物品之间的交互关系与物品和物品之间在功能上的互补关系分别从用户和物品 2 个角度构建正负样本对，提出一种联合成对排序的推荐模型。
- 2) 构建一种新颖的排序感知策略，该策略不仅能够合理选取负样本，而且可以根据正样本当前排名位置动态控制模型学习梯度。
- 3) 设计一种高效的协同过滤推荐算法——CPR。实验结果表明，所设计的算法在推荐精度及收敛速度上均优于当前主流推荐算法。

2 相关工作

单值排序方法通常把物品排序任务转化为回归或分类问题，将观测数据视为正样本，并将所有未观测数据视为负样本。系统从观测数据和未观测数据中学习一个回归或分类函数，并基于该函数预测用户对物品的喜好程度，最后根据喜好程度进行排序获得推荐列表。Hu 等^[8]提出一种加权正则化矩阵分解 (WRMF, weighted regularized matrix factorization) 模型，该模型通过学习一个加权平方损失函数来建模用户和物品之间的交互关系。尽管该模型基于权重策略缓解了正负样本不平衡的问题，但存在所有负样本使用相同权重的问题。为了解决该问题，He 等^[9]设计了一种用于负样本的基于物品流行度的可变权重策略，使改进后的模型性能明显提升。He 等^[17]又提出了一种神经矩阵分解 (NeuMF, neural matrix factorization) 模型，该模型统一了矩阵分解模型的线性优势和多层感知机的非线性优势。

相比于单值排序方法，成对排序方法更加关注物品之间的相对排名位置。该类方法通常假设用户对已交互物品的喜好要强于未交互物品。贝叶斯个性化排序^[10]由于理论完备、可扩展性强、易于实现等优势已成为成对排序方法中的经典模型。近些年，不少学者以此为出发点，提出了许多扩展的模型并取得了不错的推荐效果。例如，Zhao 等^[13]融入用户社交关系，构建社会化贝叶斯个性化排序 (SBPR, social Bayesian personalized ranking) 模型，该模型潜在地满足 2 个假设：1) 用户对自身购买物品的偏好要强于其朋友购买过的物品；2) 用户对自身或朋友购买过物品的喜好要强于两者均未购买的物品。Zhao 等^[13]通过大量实验验证了上述假设的成立。Chen 等^[18]通过设计一种计算用户之间局部相似性的模式来改进 BPR 模型，提出局部相似度的贝叶斯个性化排序 (HLBPR, hybrid local Bayesian personal ranking)。Ding 等^[11]提出了一种改进的贝叶斯个性化排序 (BPR+view) 模型，该模型通过考虑用户的浏览信息减小了对负样本的搜索空间。这些模型虽然通过融入额外信息提升了原始 BPR 的推荐精度，但负样本的选取依然基于均匀采样。Rendle 等^[16]认为现实世界中物品的流行度大都符合长尾分布，这使负样本基于均匀采样的算法面临着收敛缓慢问题，鉴于此，提出了一种自适应的负

采样算法，并将其用于原始的 BPR 模型。

上述工作主要侧重于从用户角度优化物品之间的相对排名位置，忽略了物品之间内在的功能关系对用户决策的影响。本文同时从用户和物品 2 个角度对决策过程进行建模，即分别基于用户历史购买记录和物品的联合购买记录构建正负样本对。其次，在构建正负样本对时，上述模型均忽略了正样本的当前排名位置对于模型学习梯度的影响，鉴于此，本文设计了一种新颖的排序感知策略，该策略不仅能够自适应地选择负样本，而且能够根据正样本当前的排名位置动态控制模型学习梯度。

3 预备知识

3.1 符号定义

本文使用黑斜体的大写字母表示矩阵(如 \mathbf{P})，并使用黑斜体的小写字母表示向量(如 \mathbf{p})，集合用花体表示(如 \mathcal{U})。用字符上加 $\hat{\cdot}$ 表示预测值(如 \hat{r}_{ui})。本文的主要符号及其含义如表 1 所示。

表 1 主要符号及其含义

符号	含义
\mathcal{U}	用户集合($ \mathcal{U} = m$)
\mathcal{I}	物品集合($ \mathcal{I} = n$)
\mathcal{I}_u	用户 u 购买的物品集合
\mathcal{I}_k	与物品 k 联合购买的物品集合
\mathcal{T}	用户购买反馈信息集合
\mathcal{S}	联合购买反馈信息集合
\mathcal{D}_T	用户-物品成对样本集合
\mathcal{D}_S	物品-物品成对样本集合
d	潜在特征向量维度
$\mathbf{P} \in \mathbb{R}^{m \times d}$	用户潜在特征矩阵
$\mathbf{Q} \in \mathbb{R}^{n \times d}$	物品潜在特征矩阵
$\mathbf{R} \in \mathbb{R}^{m \times n}$	用户-物品交互矩阵
$\mathbf{C} \in \mathbb{R}^{n \times n}$	物品-物品互补矩阵

本文用矩阵 $\mathbf{R} = [r_{ui}]_{m \times n}$ 表示用户-物品交互矩阵，其中， u 表示用户， i 表示物品，用户 u 购买过物品 i 则 $r_{ui} = 1$ ，否则 $r_{ui} = 0$ 。矩阵 $\mathbf{C} = [c_{kv}]_{n \times n}$ 表示物品-物品互补矩阵， c_{kv} 表示物品 k 与物品 v 之间互补程度，用户经常联合购买物品 k 与物品 v 则 $c_{kv} = 1$ ，否则 $c_{kv} = 0$ 。用户-物品成对样本集合为 $\mathcal{D}_T = \{(u, i, j) | i \in \mathcal{I}_u, j \in \mathcal{I} \setminus \mathcal{I}_u\}$ ，其中 $\mathcal{I} \setminus \mathcal{I}_u$ 表示用户 u 未购买物品集合。物品-物品成对样本集合为

$\mathcal{D}_S = \{(k, v, w) | v \in \mathcal{I}_k, w \in \mathcal{I} \setminus \mathcal{I}_k\}$, 其中 $\mathcal{I} \setminus \mathcal{I}_k$ 表示未与物品 k 进行联合购买的物品集合。本文的目标是向每个用户生成一个个性化的物品排序列表, 表中的物品来自用户未购买过的物品 $\{j | (u, j) \notin \mathcal{T}\}$ 。

3.2 贝叶斯个性化排序

成对排序方法通常使用 $(u, i) \succ (u, j)$ 表示用户 u 对物品 i 的喜好强于物品 j 的。 $\delta(b)$ 是二元指示函数, b 为真, 则 $\delta(b) = 1$, 否则 $\delta(b) = 0$ 。 $\delta((u, i) \succ (u, j))$ 表示用户对物品 i 的喜好是否强于物品 j 。基于 $\delta((u, i) \succ (u, j))$ 的伯努利分布计算用户 u 与所有物品的成对喜好似然表示为

$$\begin{aligned} \text{LLP}(u) &= \prod_{i, j \in \mathcal{I}} \Pr(\hat{r}_{ui} > \hat{r}_{uj})^{\delta((u, i) \succ (u, j))} \\ &= \prod_{(u, i) \succ (u, j)} \Pr(\hat{r}_{ui} > \hat{r}_{uj}) \left[1 - \Pr(\hat{r}_{ui} > \hat{r}_{uj}) \right]^{1 - \delta((u, i) \succ (u, j))} \end{aligned} \quad (1)$$

基于式(1)的成对喜好似然表示, Rendle 等^[10]提出了一种贝叶斯个性化排序的推荐模型。从准确性和效率两方面来看, 该模型是解决推荐系统中物品排序任务的主流模型, 其满足以下假设条件。

假设 1 对于用户 u , 物品 $i \in \mathcal{I}_u$, 物品 $j \in \mathcal{I} \setminus \mathcal{I}_u$, 用户 u 对物品 i 的喜好程度强于物品 j , 其形式化可表示为 $(u, i) \succ (u, j)$ 或 $\hat{r}_{uij} > 0$ 。

基于该假设所有用户的成对喜好似然表示为

$$\begin{aligned} \text{LLP}_u &= \prod_{u \in \mathcal{U}} \text{LLP}(u) = \\ &= \prod_{u \in \mathcal{U}} \prod_{i \in \mathcal{I}_u} \prod_{j \in \mathcal{I} \setminus \mathcal{I}_u} \Pr(\hat{r}_{ui} > \hat{r}_{uj}) \left[1 - \Pr(\hat{r}_{ui} > \hat{r}_{uj}) \right] \end{aligned} \quad (2)$$

通常使用 Sigmoid 函数 $\sigma(x) = \frac{1}{1 + e^{-x}}$ 近似于 $\Pr(\hat{r}_{ui} > \hat{r}_{uj})$, 即 $\sigma(\hat{r}_{ui} - \hat{r}_{uj}) = \frac{1}{1 + e^{\hat{r}_{uj} - \hat{r}_{ui}}}$, 那么 $\Pr(\hat{r}_{ui} > \hat{r}_{uj}) \left[1 - \Pr(\hat{r}_{ui} > \hat{r}_{uj}) \right] = \sigma(\hat{r}_{ui} - \hat{r}_{uj})^2$ 。最大化式(2)等价于优化式(3)所示的目标函数。

$$\begin{aligned} \mathcal{L} &= \frac{1}{2} \ln \text{LLP}_u - \frac{\lambda_\theta}{2} \|\theta\|_2^2 = \\ &= \sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}_u} \sum_{j \in \mathcal{I} \setminus \mathcal{I}_u} \ln \sigma(\hat{r}_{ui} - \hat{r}_{uj}) - \frac{\lambda_\theta}{2} \|\theta\|_2^2 \end{aligned} \quad (3)$$

其中, λ_θ 为模型超参数, θ 为求解的模型参数。

4 联合成对排序的推荐方法

4.1 联合建模交互关系与互补关系

在现实生活中, 除用户自身喜好之外, 物品之间的功能关系往往也是用户做出购买决策的重要因素。在图 1 中, 用户经常同时购买手机、手机壳、耳机等物品, 文献[19-20]分析得出这些物品在功能上通常具有互补性。合理使用物品之间的互补关系对用户的决策过程进行建模, 不仅有助于提升推荐系统的精确度, 而且有助于增加推荐结果的多样性与用户的满意度。与文献[20]做法一致, 本文使用数据集联合购买记录来构建物品之间在功能上的互补关系。与建模用户和物品之间的交互关系相似, 对于物品之间的互补关系需要满足以下假设。

假设 2 对于物品 k , 物品 $v \in \mathcal{I}_k$, 物品 $w \in \mathcal{I} \setminus \mathcal{I}_k$, 存在物品 k 对物品 v 的互补程度强于对物品 w 的, 可形式化表示为 $(k, v) \succ (k, w)$ 或 $\hat{c}_{kvw} > 0$ 。

假定系统中存在 m 个用户和 n 个物品, 那么根据用户的历史购买记录可以构建用户-物品的交互矩阵 $\mathbf{R} = [r_{ui}]_{m \times n}$, 根据联合购买信息可以构建物品-物品的互补矩阵 $\mathbf{C} = [c_{kv}]_{n \times n}$ ^[19-20]。本文用 $\mathbf{P} \in \mathbb{R}^{m \times d}$ 表示用户潜在特征矩阵, $\mathbf{Q} \in \mathbb{R}^{n \times d}$ 表示物品潜在特征矩阵, \mathbf{p}_u 表示用户 u 的潜在特征向量, \mathbf{q}_i 表示物品 i 的潜在特征向量。在电子商务系统中, 物品之间的互补关系通常具有方向性, 譬如用户 u 最近购买了手机, 向其推荐手机壳是比较合适的; 但反向推荐不但不能增加商家的销售量, 反而会降低用户对系统的信任度。如果仅用矩阵 \mathbf{Q} 建模物品之间的互补关系, 将无法体现互补关系的有向性, 即 $\hat{c}_{kv} = \mathbf{q}_k^\top \mathbf{q}_v$, $\hat{c}_{vk} = \mathbf{q}_v^\top \mathbf{q}_k$, $\hat{c}_{kv} = \hat{c}_{vk}$ 。鉴于此, 本文引入互补潜在特征矩阵 $\mathbf{H} \in \mathbb{R}^{n \times d}$, 使学习的互补关系具有方向性, 即 $\hat{c}_{kv} = \mathbf{q}_k^\top \mathbf{h}_v$, $\hat{c}_{vk} = \mathbf{q}_v^\top \mathbf{h}_k$, $\hat{c}_{kv} \neq \hat{c}_{vk}$, 其中 \mathbf{h}_v 、 \mathbf{h}_k 分别表示建模物品 v 和 k 在功能互补方面的特征向量。联合假设 1 和假设 2, 从用户自身因素和物品之间的内在联系这 2 个角度建模用户的决策过程, 如式(4)所示。

$$\text{LLP} = \text{LLP}_u \times \text{LLP}_k, \text{LLP}_k = \prod_{k \in \mathcal{I}} \text{LLP}(k) \quad (4)$$

其中, 有

$$\begin{aligned} \text{LLP}(k) &= \prod_{v \in \mathcal{I}_k} \prod_{w \in \mathcal{I} \setminus \mathcal{I}_k} \Pr(\hat{c}_{kv} > \hat{c}_{kw})^\beta \\ &= \left[1 - \Pr(\hat{c}_{kv} > \hat{c}_{kw}) \right]^\beta \end{aligned}$$

其中, β 是超参数, 控制物品的互补关系在用户决策过程中所占的比重。与式(2)类似, 如使用 $\sigma(\hat{c}_{kv} - \hat{c}_{kw}) = \frac{1}{1 + e^{\hat{c}_{kv} - \hat{c}_{kw}}}$ 近似于 $\Pr(\hat{c}_{kv} > \hat{c}_{kw})$, 则有 $\Pr(\hat{c}_{kv} > \hat{c}_{kw})[1 - \Pr(\hat{c}_{kv} > \hat{c}_{kw})] = \sigma(\hat{c}_{kv} - \hat{c}_{kw})^2$ 。最大化式(4)等价于优化式(5)所示的目标函数。

$$\mathcal{L} = \frac{1}{2} \ln \text{LLP}_u + \frac{\beta}{2} \ln \text{LLP}_k - \frac{\lambda_\theta}{2} \|\theta\|_2^2 = \sum_{(u,i) \in \mathcal{T}} \sum_{j \in \mathcal{I} \setminus \mathcal{I}_u} \ln \sigma(\hat{r}_{ui} - \hat{r}_{uj}) + \beta \sum_{(k,v) \in \mathcal{S}} \sum_{w \in \mathcal{I} \setminus \mathcal{I}_k} \ln \sigma(\hat{c}_{kv} - \hat{c}_{kw}) - \frac{\lambda_\theta}{2} \|\theta\|_2^2 \quad (5)$$

其中, $\hat{r}_{ui} = \mathbf{p}_u^T \mathbf{q}_i$, $\theta = \{\mathbf{p}_u, \mathbf{q}_i, \mathbf{q}_j, \mathbf{q}_k, \mathbf{h}_v, \mathbf{h}_w\}$ 为需求解的模型参数, $\lambda_\theta \|\theta\|_2^2 = \lambda_p \|\mathbf{p}_u\|_2^2 + \lambda_q (\|\mathbf{q}_i\|_2^2 + \|\mathbf{q}_j\|_2^2 + \|\mathbf{q}_k\|_2^2) + \lambda_h (\|\mathbf{h}_v\|_2^2 + \|\mathbf{h}_w\|_2^2)$ 。

4.2 融合排序感知策略的联合成对排序模型

对于成对排序方法而言, 其最终优化目标是将用户喜欢的物品排在推荐列表比较靠前的位置。然而, 当前成对排序方法大都忽略了正样本的当前排名位置及负样本的选取对于推荐结果的影响。为了更加清晰地阐述, 本节给出一个直观的例子, 如图2所示。对比图2(a)和图2(b)可知, 图2(a)中正样本*i*的排名位置更接近于底部, 如果与图2(b)一样使用相同学习梯度来更新模型的参数, 这显然降低了模型的收敛速度。对比图2(a)和图2(c)可知, 图2(c)中正样本*i*与负样本*j*的相对排名位置趋向于理想的排名列表。如若选取类似于图2(c)中的正负样本对, 那么该负样本对于模型训练贡献甚微。为了解决当前多数方法面临的这个问题, 本文设计了一种新颖的排序感知策略, 该策略能够根据正样本的当前排名位置动态地定义权重函数, 并能够根据当前的正样本选取合适的负样本。

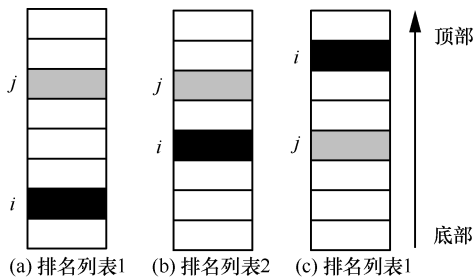


图2 用户*u*的3种排序列表

给定用户*u*, 本文用 γ_i 表示正样本*i*在用户推

荐列表中的排名位置。由图2可知, 正样本的排名位置越高(越接近底部), 对于模型的梯度更新更应该贡献越大。受此启发, 可定义权重函数为

$$\Gamma_u(\gamma_i) = \frac{1}{\Gamma(|\mathcal{I}|)} \sum_{f=1}^{\gamma_i} \frac{1}{f} \quad (6)$$

其中, $\Gamma(|\mathcal{I}|) = \sum_{f=1}^n \frac{1}{f}$ 为归一化项。在式(6)中最重要的是获取正样本*i*的当前排名位置。在训练模型时, 对于每个用户来说, 如果对所有物品均进行排序来获取正样本的*i*的排名位置, 通常需要较高的时间复杂度。为了避免对所有物品进行排序, 本文设计了一种近似的方式来快速获取 γ_i 。对于给定的样本对 $(u, i) \in \mathcal{T}$, 在 $|\mathcal{I} \setminus \mathcal{I}_u|$ 个用户*u*未交互的样本中随机选取负样本*j*, 直至满足条件 $\hat{r}_{ui} - \alpha_u \leq \hat{r}_{uj}$, 则停止取样并记录抽取的次数为 ζ_{ui} 。这里超参数 α_u 用于控制用户-物品正负样本的间隔, 其正样本*i*的排名位置 γ_i 可近似估计为

$$\gamma_i = \left\lfloor \frac{|\mathcal{I} \setminus \mathcal{I}_u| - 1}{\zeta_{ui}} \right\rfloor \quad (7)$$

其中, $\lfloor \cdot \rfloor$ 表示向下取整。同理, 对于给定的样本对 $(k, v) \in \mathcal{S}$, 在 $\mathcal{I} \setminus \mathcal{I}_k$ 个样本中随机选取负样本*w*, 直至满足条件 $\hat{c}_{kv} - \alpha_k \leq \hat{c}_{kw}$, 则停止取样并记录抽取的次数 ζ_{kv} 。这里超参数 α_k 用于控制物品-物品正负样本的间隔。为了降低模型的复杂度, 本文设置用户和物品2个角度的正负样本排名间隔相同, 即 $\alpha = \alpha_u = \alpha_k$ 。正样本*v*的排名位置 γ_v 和权重函数 $\Gamma_k(\gamma_v)$ 详细计算方式如式(8)和式(9)所示。

$$\Gamma_k(\gamma_v) = \frac{1}{\Gamma(|\mathcal{I}|)} \sum_{f=1}^{\gamma_v} \frac{1}{f} \quad (8)$$

$$\gamma_v = \left\lfloor \frac{|\mathcal{I} \setminus \mathcal{I}_k| - 1}{\zeta_{kv}} \right\rfloor \quad (9)$$

从式(6)和式(8)可看出, 权重函数是一个递增函数, γ_i 和 γ_v 越大表明正样本的排名位置越接近底部, 其权重值越大, 从而模型学习速度越快。通过将式(6)和式(8)代入式(5)中可得出最终联合成对排序模型的目标函数如式(10)所示。

$$\mathcal{L} = \sum_{(u,i) \in \mathcal{T}} \Gamma_u(\gamma_i) \sum_{j \in \mathcal{I} \setminus \mathcal{I}_u} \ln \sigma(\hat{r}_{ui} - \hat{r}_{uj}) + \beta \sum_{(k,v) \in \mathcal{S}} \Gamma_k(\gamma_v) \sum_{w \in \mathcal{I} \setminus \mathcal{I}_k} \ln \sigma(\hat{c}_{kv} - \hat{c}_{kw}) - \frac{\lambda_\theta}{2} \|\theta\|_2^2 \quad (10)$$

式(10)所示的目标函数通过排序感知策略不仅能选取比正样本排名更靠前的负样本,而且能根据正样本的当前排名位置动态地控制模型学习梯度。

4.3 模型求解

基于成对排序的推荐模型通常直接采用随机梯度下降优化算法^[9]求解模型的参数。然而从表2可以看出,用户的历史购买信息与物品的联合购买信息极度不平衡,即前者远比后者稠密,直接优化本文模型将会导致不能有效使用后者信息。鉴于此,本文采用交替优化方式^[21]对式(10)进行求解,即首先使用用户和物品之间的交互关系求解参数 \mathbf{p}_u 、 \mathbf{q}_i 、 \mathbf{q}_j , 然后基于物品之间的互补关系求解 \mathbf{q}_k 、 \mathbf{h}_v 、 \mathbf{h}_w 。对于 $\frac{\partial \mathcal{L}}{\partial \mathbf{p}_u}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_i}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_j}$ 的求解结果如式(11)所示。

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{p}_u} &= \Gamma_u(\gamma_i) \sigma(\hat{r}_{uj} - \hat{r}_{ui})(\mathbf{q}_i - \mathbf{q}_j) - \lambda_p \mathbf{p}_u \\ \frac{\partial \mathcal{L}}{\partial \mathbf{q}_i} &= \Gamma_u(\gamma_i) \sigma(\hat{r}_{uj} - \hat{r}_{ui}) \mathbf{p}_u - \lambda_q \mathbf{q}_i \\ \frac{\partial \mathcal{L}}{\partial \mathbf{q}_j} &= -\Gamma_u(\gamma_i) \sigma(\hat{r}_{uj} - \hat{r}_{ui}) \mathbf{p}_u - \lambda_q \mathbf{q}_j \end{aligned} \quad (11)$$

对于 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_k}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{h}_v}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{h}_w}$ 的求解结果如式(12)所示。

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \mathbf{q}_k} &= \beta \Gamma_k(\gamma_v) \sigma(\hat{c}_{kv} - \hat{c}_{kw})(\mathbf{h}_v - \mathbf{h}_w) - \lambda_q \mathbf{q}_k \\ \frac{\partial \mathcal{L}}{\partial \mathbf{h}_v} &= \beta \Gamma_k(\gamma_v) \sigma(\hat{c}_{kv} - \hat{c}_{kw}) \mathbf{q}_k - \lambda_h \mathbf{h}_v \\ \frac{\partial \mathcal{L}}{\partial \mathbf{h}_w} &= -\beta \Gamma_k(\gamma_v) \sigma(\hat{c}_{kv} - \hat{c}_{kw}) \mathbf{q}_k - \lambda_h \mathbf{h}_w \end{aligned} \quad (12)$$

本节在算法1给出了求解所提模型的详细步骤。

算法1 求解所提模型的详细步骤

输入 用户-物品交互矩阵 \mathbf{R} , 物品-物品互补矩阵 \mathbf{C} , 潜在特征向量维数 d , 学习速率 η , 正负样本排名间隔超参数 α , 权衡超参数 β , 最大迭代次数 L 。

输出 用户潜在特征矩阵 \mathbf{P} , 物品潜在特征矩阵 \mathbf{Q} , 互补潜在特征矩阵 \mathbf{H} 。

1) 随机初始化 \mathbf{P} 、 \mathbf{Q} 和 \mathbf{H} , 且基于正态分布(0, 0.01)

2) while iter \leftarrow 1 to L DO

- 3) for $rNum \leftarrow 1$ to $|\mathcal{T}|$ do
- 4) 随机采样 $(u, i) \in \mathcal{T}$
- 5) 根据 $\hat{r}_{ui} - \alpha \leq \hat{r}_{ij}$ 选择负样本 j , 并根据式(7) 计算正样本 i 排名
- 6) 根据式(6)计算 $\Gamma_u(\gamma_i)$
/*更新模型参数 \mathbf{p}_u 、 \mathbf{q}_i 、 \mathbf{q}_j */
- 7) $\mathbf{p}_u \leftarrow \mathbf{p}_u + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{p}_u}$
- 8) $\mathbf{q}_i \leftarrow \mathbf{q}_i + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{q}_i}$
- 9) $\mathbf{q}_j \leftarrow \mathbf{q}_j + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{q}_j}$
- 10) end for
- 11) for $sNum \leftarrow 1$ to $|\mathcal{S}|$ do
- 12) 随机采样 $(k, v) \in \mathcal{S}$
- 13) 根据 $\hat{c}_{kv} - \alpha \leq \hat{c}_{kw}$ 选择负样本 w , 并根据式(9)计算正样本 v 排名,
- 14) 根据式(8)计算 $\Gamma_k(\gamma_v)$
/*更新参数 \mathbf{q}_k 、 \mathbf{h}_v 、 \mathbf{h}_w */
- 15) $\mathbf{q}_k \leftarrow \mathbf{q}_k + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{q}_k}$
- 16) $\mathbf{h}_v \leftarrow \mathbf{h}_v + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{h}_v}$
- 17) $\mathbf{h}_w \leftarrow \mathbf{h}_w + \eta \frac{\partial \mathcal{L}}{\partial \mathbf{h}_w}$
- 18) end for
- 19) end while

算法1的时间复杂度主要包括负采样过程和各梯度变量的更新。获取用户-物品正负样本对和物品-物品正负样本对的时间复杂度分别为 $O(\bar{x})$ 和 $O(\bar{y})$, 其中, \bar{x} 表示建模交互关系时选取有效负样本平均需要的次数, 且 $1 \leq \bar{x} \leq \max\{\xi_{ui} | u \in \mathcal{U}, i \in \mathcal{I}_u\}$; \bar{y} 表示建模互补关系时选取有效负样本平均需要的次数, 且 $1 \leq \bar{y} \leq \max\{\zeta_{kv} | k \in \mathcal{I}, v \in \mathcal{I}_k\}$ 。计算各梯度 $\frac{\partial \mathcal{L}}{\partial \mathbf{p}_u}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_i}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_j}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{q}_k}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{h}_v}$ 、 $\frac{\partial \mathcal{L}}{\partial \mathbf{h}_w}$ 的时间复杂度均为 $O(d)$ 。因此, CPR 算法每次迭代的复杂度为 $O(|\mathcal{T}|(\bar{x} + d) + |\mathcal{S}|(\bar{y} + d))$, 其中, $|\mathcal{T}|$ 表示用户历史购买数量, $|\mathcal{S}|$ 表示联合购买数量。由此可看出, 算法1的整体复杂度与历史购买数量和联合购买数量呈线性相关。

5 实验结果及分析

5.1 数据集介绍

为了验证 CPR 算法在 Top- N 推荐上的有效性, 本文在 4 个不同规模的亚马逊数据集上进行实验。实验中所使用的数据集分别为 Health and Personal Care、Video Games、Pet Supplies 和 Cell Phones and Accessories, 这些数据集含有丰富的元数据信息, 如用户历史购买记录、物品图片信息、用户的评论、物品之间的联合购买等信息。在这些数据中, 本文使用用户历史购买记录构建用户和物品之间的交互关系, 使用联合购买信息^[19-20]构建物品和物品之间的互补关系。

实验中对 4 个数据集进行初步处理, 剔除历史购买记录少于 5 个的用户, 表 2 给出了最终所使用数据集的一些统计特性。

5.2 评价指标

1) 准确率和召回率

准确率 (Precision) 定义为正确预测的物品数与推荐列表长度 N 的比值。召回率 (Recall) 定义为正确预测的物品数与用户 u 在测试集中真正购买的物品数的比值。假定使用 I_{rec} 表示用户 u 的推荐列表, I_{tes} 表示用户 u 在测试集中真实购买的物品列表。准确率和召回率的定义如下所示。

$$\begin{aligned} \text{Precision}@N(u) &= \frac{|I_{rec} \cap I_{tes}|}{N} \\ \text{Recall}@N(u) &= \frac{|I_{rec} \cap I_{tes}|}{|I_{tes}|} \end{aligned} \quad (13)$$

2) 平均准确率

平均准确率 (MAP, mean average precision) 是衡量所有用户 Top- N 推荐列表的平均准确率, 用于评价推荐的平均质量。AP(average precision)为单个用户的平均准确率, 定义如式(14)所示。

$$\text{AP}@N(u) = \frac{\sum_{a=1}^N P(a)\delta(a)}{\min\{N, |I_{tes}|\}} \quad (14)$$

其中, $P(a)$ 表示在 Top- N 推荐列表中前 a 个位置的准确率; $\delta(a)$ 是一个指示函数, 若在测试集中用户购买了排序为 a 的物品, 则 $\delta(a)$ 指示函数值为 1, 否则为 0。MAP 为所有用户在 AP 上的平均值。

3) 平均折损累积增益

平均折损累积增益 (DCG, discounted cumulative gain) 是衡量推荐列表与真实购买列表相关度的指标。若用户 u 购买了推荐列表中第 j 个物品则 $rel_j = 1$, 否则 $rel_j = 0$ 。其定义如式(15)所示。

$$\text{DCG}(u) = \sum_{j=1}^N \frac{2^{rel_j} - 1}{\text{lb}(j+1)} \quad (15)$$

其中, $\text{DCG}^* @ (u)$ 是 $\text{DCG}(u)$ 的理想最大值, 平均折损累积增益 (NDCG, normalize discounted cumulative gain) 为所有用户 DCG 的平均值, 其定义如式(16)所示。

$$\text{NDCG} @ (u) = \frac{\text{DCG} @ (u)}{\text{DCG}^* @ (u)} \quad (16)$$

5.3 算法对比及实验设置

在实验中, CPR 算法及各对比算法均基于开源算法库 Librec 实现。为了验证物品之间在功能上的互补关系和排序感知策略对于推荐结果的影响, 本文选取以下 7 种主流推荐算法进行对比。

1) POP (popular): 该算法是一种非个性化推荐算法, 根据物品流行度向用户推荐当前最热门的物品。

2) BPR^[10]: 该推荐算法主要优化用户对不同物品的相对排序, 即用户对于已购买物品的喜好强于未购买物品, 从而得出每个用户对所有物品的偏序关系。

3) GBPR (group Bayesian personalized ranking)^[7]: 该算法是对 BPR 的一种扩展, 考虑了群组用户偏好对个体偏好的影响, 并弱化了原始模型中用户之间相互独立的假设。

4) AOBPR (adaptive over-sampling Bayesian personalized ranking)^[16]: 一种基于 BPR 的扩展推荐算法, 采用自适应采样策略优化采样。该算法每

表 2

4 个数据集的统计特性

数据集	用户数量	物品数量	历史购买记录	交互关系稠密度	联合购买记录	互补关系稠密度
Health and Personal Care	16 201	75 431	289 965	2.37×10^{-4}	48 603	8.5×10^{-6}
Video Games	8 057	26 729	157 494	7.31×10^{-4}	19 232	2.7×10^{-5}
Pet Supplies	7 417	33 798	117 385	4.68×10^{-4}	22 980	2.0×10^{-5}
Cell Phones and Accessories	9 534	53 479	139 141	2.73×10^{-4}	28 578	0.9×10^{-5}

隔固定的迭代次数更新一次物品排序列表,并根据当前的正样本排名选取负样本。

5) WARP (weighted approximate-rank pair wise ranking)^[22]: 由 Weston 等^[22]提出的一种成对排序学习算法,该算法同样采用自适应采样策略选取负样本,但与 AOBPR 不同,它基于间隔铰链损失 (hinge loss) 函数来求解目标函数。

6) UPR (uniform over-sampling pair wise ranking): 本文设计的对比算法之一,该算法同时考虑了用户-物品的交互关系和物品之间在功能上的互补关系,但采用均匀采样策略选取负样本,其目标函数如式(5)所示。

7) APR (adaptive over-sampling pair wise ranking): 本文设计的对比算法之一,该算法与 UPR 目标函数相同,但采用与 AOBPR 相同的自适应采样策略选取负样本。

在实验中,首先根据每个用户的历史购买记录的时间信息进行排序,然后选取前 70% 数据作为训练集,10% 用于验证集,剩余的 20% 作为测试集。所有推荐算法的学习速率选取范围为 {0.001, 0.005, 0.01, 0.05}, 正则化超参数的选取范围为 {0.001, 0.001, 0.1, 1}。CPR 算法中 α 选取范围为 {0.6, 0.8, 1, 1.2, 1.4, 1.6}, β 调参范围为 {0.1, 1, 2, 3, 4, 5, 6}。AOBPR 算法中 ρ 选取范围为 {0.01, 0.03, 0.05, 0.07, 0.1, 0.3, 0.5}。GBPR 算法中对于群组用户大小的设定与原文献保持一致,设置为 3。本文所有实验在 Windows 7 操作系统, 64 GB 内存, Intel(R) Xeon(R) CPU E5-26200 @ 2.00 GHz 服务器上进行。

5.4 实验结果与分析

实验 1 全局实验

本实验用于对比各推荐算法在 4 个不同规模和稀疏度的数据集上的性能。本实验设置潜在特征向量维度 d 分别为 50 和 100, 推荐列表长度 N 为 10。表 3 详细展示了各算法在 Precision、Recall、MAP 和 NDCG 这 4 个评价指标的实验结果(均已乘以 100), 从表 3 中可得出以下 4 个结论。

1) 在 4 个数据集上, BPR、GBPR 和 AOBPR 算法均优于非个性化推荐算法 POP, 这表明个性化推荐算法能够更好地发掘用户的喜好。

2) 由表 2 可知 4 个数据集的历史购买记录均极为稀疏。BPR 和 UPR 算法采用均匀采样策略, 但

UPR 同时考虑了用户的历史购买记录和物品的联合购买记录。从表 3 中不难看出, UPR 的推荐精度要优于 BPR, 这说明考虑辅助信息在一定程度上缓解了推荐系统面临的数据稀疏问题, 同时验证了本文假设 2 的成立。

3) 使用自适应负采样策略的 APR 推荐精度要明显优于使用均匀负采样策略的 UPR, 这说明负样本的选择策略对于模型的性能起着至关重要的作用。

4) 采用排序感知策略的 CPR 明显优于 APR。这是因为 CPR 每次迭代均考虑正样本的当前排名位置对于模型训练的影响, 动态控制了模型的学习梯度。

实验 2 不同推荐列表长度下的性能对比

为了验证各推荐算法在不同推荐列表长度 N 下的性能, 本实验在潜在特征向量维度为 100, 推荐列表长度 $N = \{10, 15, 20, 25\}$ 的条件下进行了实验。

由于空间限制, 本文仅展示了 Precision@ N 和 Recall@ N 指标上的实验结果, 如图 3 所示。从图 3 中可得出以下主要结论。

1) 在不同推荐列表长度下, 各算法具有相同的变化趋势; 随着 N 的增大, 在各算法的 Precision 逐渐下降, Recall 逐渐上升。

2) CPR 算法在 4 个数据集的不同推荐列表长度下均优于其他主流的推荐算法, 这进一步验证了本文算法的有效性及假设 2 的合理性。

实验 3 不同历史购买数量下的性能对比

为了分析在不同用户上各推荐算法的性能, 本实验依据训练集中用户的历史购买数量将用户划分为 1~10、11~15、16~20、>20 这 4 组, 图 4 显示了 4 个数据集中每组用户所占比例。从图 4 中可看出历史购买数量少于 10 个的用户占大部分, 即多数用户为不活跃用户。依据用户的历史购买数量进行分组之后, 使用训练集来学习各推荐算法, 然后在测试集上分别计算 4 组用户的推荐精度。图 5 展示了 CPR 与 3 种对比算法在 Precision@10 上的实验结果, 由此可得出以下结论。

1) 在第一组用户上, UPR 的精度均优于 BPR。这是因为该组用户历史购买数据比较稀疏, 融入物品之间的功能互补关系不仅有助于缓解数据稀疏问题, 同时更加合理地对用户和物品进行了建模。

表 3 在 4 个数据集上的实验结果

数据集	算法	$d=50$				$d=100$			
		Precision	Recall	MAP	NDCG	Precision	Recall	MAP	NDCG
Health and Personal Care	POP	0.499 3	1.070 1	0.552 9	0.989 5	0.499 3	1.070 1	0.552 9	0.989 5
	BPR	1.211 2	3.268 7	1.328 4	2.408 2	1.268 5	3.413 5	1.439 5	2.581 0
	GBPR	1.336 2	3.439 7	1.563 5	2.703 9	1.446 3	3.746 0	1.703 7	2.970 3
	AOBPR	1.428 8	3.736 1	1.629 0	2.888 2	1.533 7	3.993 4	1.818 9	3.157 1
	WARP	1.607 9	3.994 5	1.861 0	3.208 7	1.671 1	4.158 5	1.979 5	3.363 9
	UPR	1.282 8	3.512 7	1.427 2	2.583 0	1.366 2	3.836 7	1.609 6	2.877 8
	APR	1.591 0	4.407 3	1.948 2	3.399 4	1.723 2	4.798 7	2.148 7	3.730 2
	CPR	1.862 0	4.933 6	2.307 7	3.904 2	1.962 3	5.123 6	2.407 5	4.077 8
Video Games	POP	0.587 7	1.599 8	0.608 7	1.171 3	0.587 7	1.599 8	0.608 7	1.171 3
	BPR	2.445 6	6.649 6	2.594 6	4.859 8	2.479 5	6.626 0	2.575 9	4.879 4
	GBPR	3.337 2	9.106 0	3.881 0	6.898 2	3.522 4	9.614 0	4.164 6	7.357 5
	AOBPR	3.414 2	9.471 9	4.164 9	7.264 2	3.500 2	9.677 8	4.226 7	7.382 2
	WARP	3.532 8	9.743 5	4.202 7	7.399 3	3.581 0	10.02 6	4.355 2	7.622 2
	UPR	2.501 6	6.684 7	2.687 2	4.963 2	2.562 9	7.071 7	2.650 1	5.023 3
	APR	3.543 2	9.753 6	4.179 5	7.407 6	3.706 2	10.31 2	4.307 4	7.677 8
	CPR	3.737 5	10.44 0	4.579 5	7.994 6	3.909 5	11.01 3	4.795 3	8.330 6
Pet Supplies	POP	0.670 2	1.979 4	0.652 3	1.352 3	0.670 2	1.979 4	0.652 3	1.352 3
	BPR	1.647 1	4.862 7	1.836 2	3.485 7	1.678 6	4.954 6	1.862 1	3.531 7
	GBPR	1.732 9	5.069 4	2.040 9	3.756 5	1.925 7	5.648 3	2.298 8	4.214 0
	AOBPR	1.864 3	5.482 4	2.215 7	4.081 8	1.968 6	5.845 6	2.264 0	4.240 6
	WARP	1.971 4	5.847 8	2.388 0	4.345 3	1.992 9	5.822 2	2.453 5	4.437 9
	UPR	1.694 3	5.123 9	1.996 1	3.695 5	1.751 4	5.356 8	1.887 6	3.657 2
	APR	2.101 4	6.453 1	2.691 6	4.854 8	2.264 3	6.914 1	2.777 3	5.071 9
	CPR	2.232 9	6.778 9	2.856 6	5.111 9	2.360 0	7.101 6	2.972 2	5.325 7
Cell Phones and Accessories	POP	0.312 7	0.969 5	0.360 6	0.688 5	0.312 7	0.969 5	0.360 6	0.688 5
	BPR	1.194 9	4.093 2	1.600 0	2.845 2	1.296 9	4.406 8	1.771 8	3.111 4
	GBPR	1.211 7	4.070 2	1.780 2	3.010 7	1.376 5	4.576 3	2.001 2	3.392 9
	AOBPR	1.340 7	4.533 8	1.863 1	3.235 1	1.443 8	4.966 8	2.107 7	3.588 9
	WARP	1.462 8	4.942 2	2.186 1	3.653 2	1.498 7	5.042 7	2.154 5	3.658 5
	UPR	1.333 9	4.531 6	1.762 3	3.154 9	1.397 8	4.757 4	1.928 7	3.378 7
	APR	1.615 3	5.478 4	2.306 3	3.990 0	1.776 7	6.011 8	2.565 7	4.392 3
	CPR	1.780 1	5.968 9	2.495 9	4.307 3	1.811 5	6.091 7	2.714 6	4.543 0

2) 随着用户购买数量的增加，各推荐算法的 Precision@10 值也随之增加。这是因为用户的购买数量增多，训练模型可使用的数据变得更丰富，更易学习到用户的潜在特征向量。值得注意的是，CPR 算法在 4 组用户上均明显优于各对比算法，这是因为该算法不仅考虑了物品之

间在功能上的互补关系，同时采用排序感知的负采样策略，使每次迭代时模型学习梯度均趋向较优的方向。

实验 4 收敛性分析

为了分析 CPR 算法在收敛性上的优势，本实验统一设置各算法的潜在特征向量维度 d 为 50，并迭

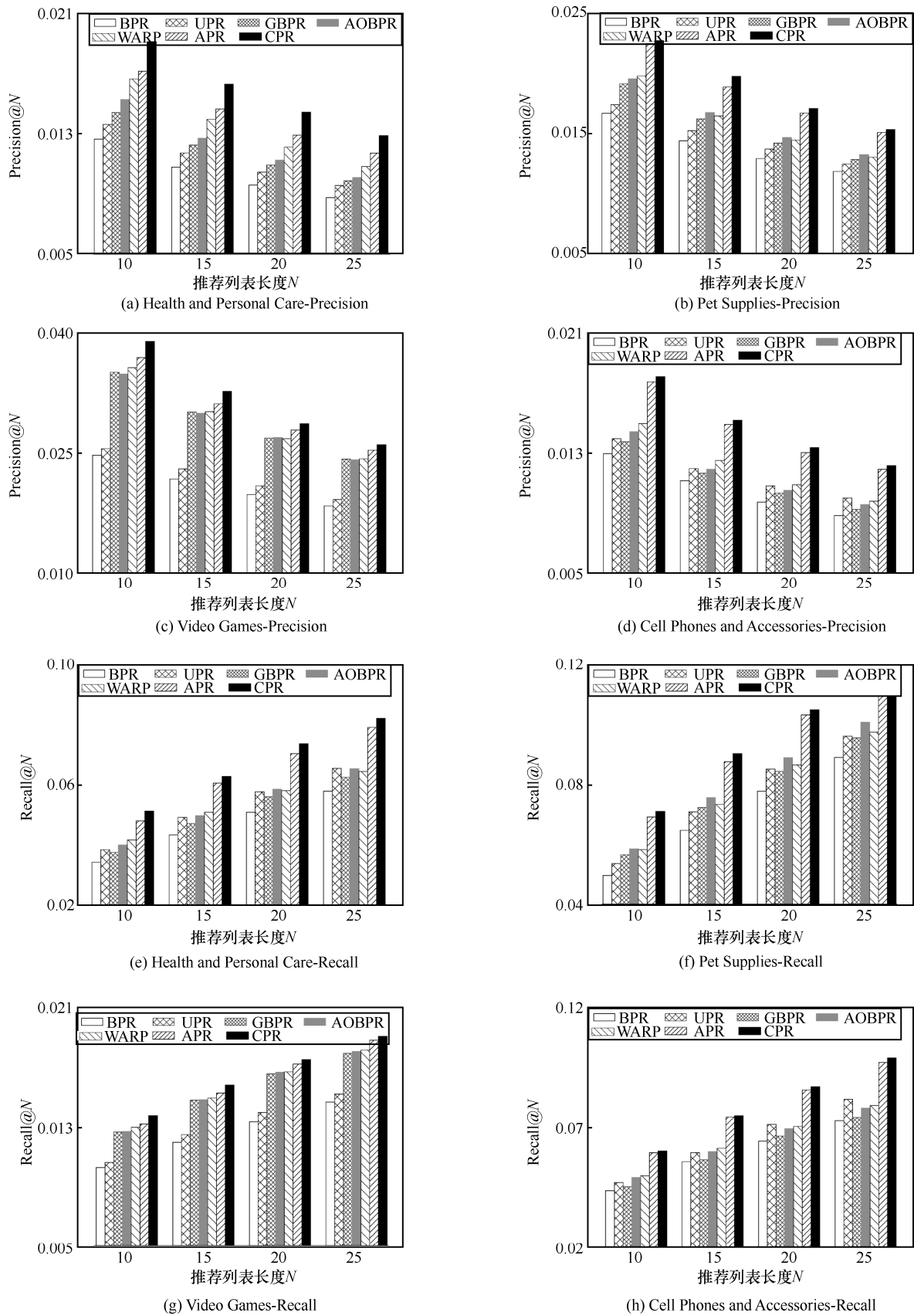


图 3 不同 N 值下的性能对比

代 1 000 次。图 6 详细展示了该实验结果，其中横轴表示迭代次数，纵轴表示评估指标(实验结果均已乘以 100)。由于各推荐算法在 4 个指标上具有相似的结论，本实验仅展示在 Precision@10 指标下各算法的收敛趋势。从图 6 中可得出以下 3 种结果。

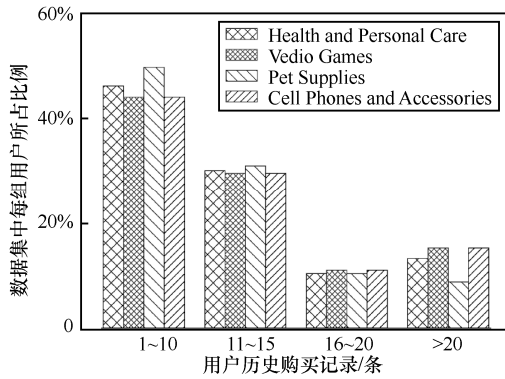


图 4 不同历史购买记录下用户的分布

1) 4 种推荐算法随着迭代次数的增加均趋于收敛，最终在很小的范围内波动。相比于采用均匀采样策略的算法 BPR，采用自适应采样策略的算法 AOBPR 和 APR 具有收敛快的特点，这与文献[16]结论一致。

2) APR 与 AOBPR 均采用自适应负采样策略，但具有不同的收敛速度。从图 6 可以看出，在 4 个数据集上融入物品之间功能互补关系的 APR 算法收敛速度更快。这说明辅助信息的融入有助于模型更易学习出用户和物品的潜在特征向量，从而加快了算法的收敛速度。

3) 除 Pet Supplies 以外，在其他数据集上迭代 30~50 次使用不同采样策略的 APR 和 CPR 收敛速度基本重合。当 APR 算法趋于收敛时，CPR 算法依然有推荐精度上的提升。这是因为 CPR 算法采用了排序感知采样策略，使得每次迭代不仅能够选取有效的负样本，并能根据正样本的当前排名位置动态控制模型的学习梯度。

实验 5 超参数对算法的影响

为了分析 CPR 算法中 2 个重要超参数对于推荐精度的影响，本文在 4 个数据集上详细讨论了不同取值下的实验结果，如图 7 所示。其中，超参数 α 取值范围为 {0.6,0.8,1.0,1.2,1.4,1.6}，超参数 β 取值范围为 {0,1.0,1.5,2,2.5,3}。从图 7 可得出以下 2 个结论。

1) α 控制着正样本与负样本之间的排名间隔，其值越大，表示在已知正样本排名位置的情况下，

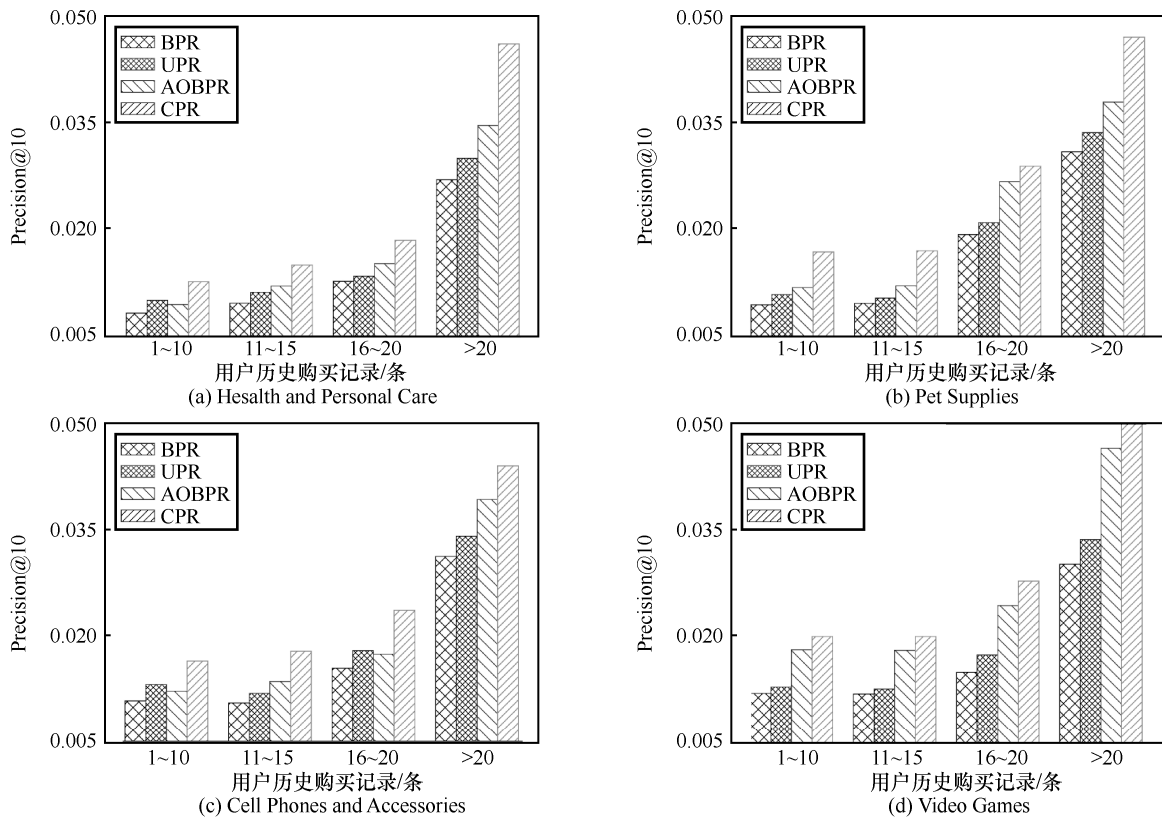


图 5 不同历史购买记录下各算法对比结果

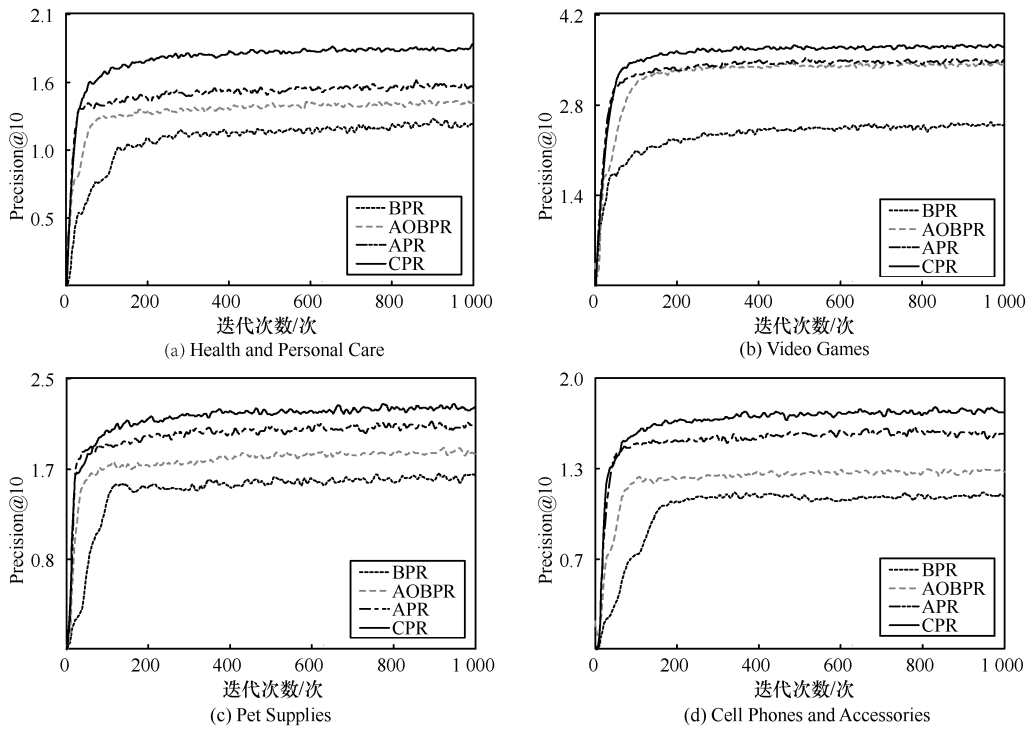


图 6 收敛性对比

选取负样本的排名越靠近顶部。在 4 个数据集上超参数 α 的值从 0.6 开始 NDCG@10 值先上升到达某个阈值后下降, 这说明正负样本的排名间隔直接影响着 CPR 算法的性能。 α 值较小时模型训练变慢, 在有限迭代次数下, 其推荐性能将会受到影响; 相反 α 值较大时, 不仅在选取负样本时可能需要更多时间开销, 而且当正样本排名比较靠近顶部时, 可能选择不到符合条件的负样本, 使该次迭代对于学习模型参数贡献甚微。

2) β 控制着物品之间的功能互补关系在用户决策过程中所占的比重。当 β 为 0 时, 表示仅考虑用户历史购买信息, 并采用排序感知策略选取负样本的实验结果。从图 7 可看出, 在超参数 β 为 0 时, 实验结果较差, 这说明物品之间在功能上的互补关系在用户决策过程中起着积极的作用。但当超参数 β 的值超过某个阈值时, 其推荐结果开始下降, 这因为过度依赖物品之间的互补关系将失去用户自身偏好的影响, 使推荐系统的个性化程度降低, 从

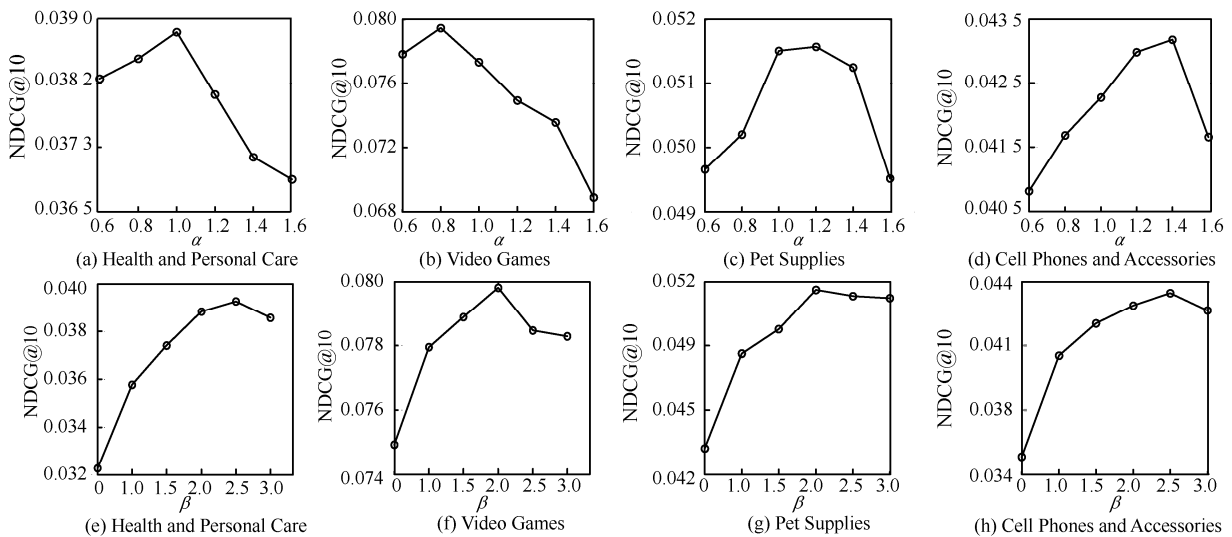


图 7 超参数的影响

表 4 在 4 个数据集上各算法运行时长(时:分:秒)

算法	Health and Personal Care	Video Games	Pet Supplies	Cell Phones and Accessories
BPR	00:05:21	00:02:55	00:02:08	00:02:37
AOBPR	04:06:06	00:47:30	00:48:36	01:32:36
WARP	04:50:55	01:26:49	01:38:23	02:08:46
GBPR	01:39:21	00:45:46	00:41:39	00:49:36
UPR	00:06:18	00:04:11	00:03:16	00:03:42
APR	05:18:46	00:56:17	00:58:19	01:46:41
CPR	04:37:01	00:48:44	00:51:42	01:27:53

而影响了推荐性能。

实验 6 各算法运行时间对比

为了检验本文算法的效率,实验 6 详细对比了不同推荐算法的训练时间。不同推荐算法在 4 个数据集上的训练时间如表 4 所示($d = 50$)。

从表 4 可观察到, AOBPR 和 WARP 在 4 个数据集上的训练时间均多于 BPR。这说明在每次迭代中选取高质量的负样本需要额外的时间开销。GBPR 和 UPR 在效率方面弱于 BPR, 这是因为 BPR 仅使用用户的历史购买记录, 而 GBPR 和 UPR 为了更准确地建模用户的喜好而引入了辅助信息。在 4 个数据集上, 本文算法 CPR 效率均优于 APR, 这说明本文所设计的排序感知采样策略要优于文献[16]所设计的自适应采样策略。综上所述, 从实验结果来看: 本文所提出的融合用户-物品交互关系和物品之间互补关系的 CPR 算法在 4 个数据集上其推荐准确性优于各对比算法; 在学习模型参数的效率方面, 本文算法与目前主流的推荐算法相当。

6 结束语

现实生活中, 用户的购买决策不仅出于自身的喜好, 物品之间的功能互补关系也起着重要作用。鉴于此, 本文从用户和物品 2 个角度对用户的决策过程进行建模, 提出一种联合成对排序的物品推荐模型。对于成对排序方法而言, 正样本的排名位置和负样本的选取策略将直接影响模型的精度及收敛速度。进一步, 本文设计了一种新颖的排序感知策略, 并基于该策略构建了一种高效的学习算法 CPR 用于求解所提模型的参数。大量实验结果表明, 本文的算法不仅缓解了数据稀疏问题, 同时加快了模型收敛, 并能够在不同历史购买数量的用户上表现出较优的推荐性能。

在未来研究中, 本文将探索如何结合其他辅助

信息, 如物品的视觉信息^[23-24]、社交关系^[25-26]、时间信息^[27]等来进一步改进所提出的模型。

参考文献:

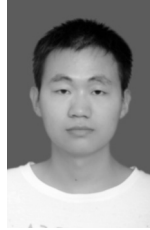
- [1] BOBADILLA J, ORTEGA F, HERNANDO A, et al. Recommender systems survey[J]. Knowledge-Based Systems, 2013(46): 109-132.
- [2] LIU Q, CHEN E, XIONG H, et al. Enhancing collaborative filtering by user interest expansion via personalized ranking[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2012, 42(1): 218-233.
- [3] 何明, 刘伟世, 张江. 支持推荐非空率的关联规则推荐算法[J]. 通信学报, 2017, 38(10): 18-25.
- [4] HE M, LIU S W, ZHANG J. Association rules recommendation algorithm supporting recommendation nonempty[J]. Journal on Communications, 2017, 38(10): 18-25.
- [5] SMITH B, LINDEN G. Two decades of recommender systems at amazon.com[J]. IEEE Internet Computing, 2017, 21(3): 12-18.
- [6] 吴宾, 姜铮铮, 叶阳东. 联合正则化的矩阵分解推荐算法[J]. 软件学报, 2018, 29(9): 2681-2696.
- [7] WU B, LOU Z Z, YE Y D. Co-regularized matrix factorization recommendation algorithm[J]. Journal of Software, 2018, 29(9): 2681-2696.
- [8] ADOMAVICIUS G, TUZHILIN A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(16): 734-749.
- [9] PAN W, CHEN L. GBPR: group preference based Bayesian personalized ranking for one-class collaborative filtering[C]//The International Joint Conference on Artificial Intelligence. AAAI Press, 2013: 2691-2697.
- [10] HU Y, KOREN Y, VOLINSKY C. Collaborative filtering for implicit feedback datasets[C]//The Eighth IEEE International Conference on Data Mining. IEEE, 2008: 263-272.
- [11] HE X, ZHANG H, KAN M Y, et al. Fast matrix factorization for online recommendation with implicit feedback[C]//The International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2016: 549-558.
- [12] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//The Conference on Uncertainty in Artificial Intelligence. AUAI Press, 2009: 452-461.
- [13] DING J, FENG F, HE X, et al. An improved sampler for Bayesian personalized ranking by leveraging view data[C]//Companion Proceedings of the Web Conference 2018. ACM, 2018: 13-14.
- [14] SUN Z, WU B, WU Y, et al. APL: adversarial pairwise learning for recommender systems[J]. Expert Systems with Applications,

- 2019(118): 573-584.
- [13] ZHAO T, MCAULEY J, KING I. Leveraging social connections to improve personalized ranking for collaborative filtering[C]//The ACM International Conference on Conference on Information and Knowledge Management. ACM, 2014: 261-270.
- [14] LERCHE L, JANNACH D. Using graded implicit feedback for Bayesian personalized ranking[C]//The ACM Conference on Recommender Systems. ACM, 2014: 353-356.
- [15] PARK Y J, TUZHILIN A. The long tail of recommender systems and how to leverage it[C]//The 2008 ACM Conference on Recommender Systems. ACM, 2008: 11-18.
- [16] RENDLE S, FREUDENTHALER C. Improving pairwise learning for item recommendation from implicit feedback[C]//The ACM International Conference on Web Search and Data Mining. ACM, 2014: 273-282.
- [17] HE X, LIAO L, ZHANG H, et al. Neural collaborative filtering[C]//The International Conference on World Wide Web. ACM, 2017: 173-182.
- [18] CHEN X, WANG P, QIN Z, et al. HLBPR: a hybrid local Bayesian personal ranking method[C]//The International Conference Companion on World Wide Web. ACM, 2016: 21-22.
- [19] MCAULEY J, PANDEY R, LESKOVEC J. Inferring networks of substitutable and complementary products[C]//The ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015: 785-794.
- [20] WANG Z, JIANG Z, REN Z, et al. A path-constrained framework for discriminating substitutable and complementary products in e-commerce[C]//The ACM International Conference on Web Search and Data Mining. ACM, 2018: 619-627.
- [21] OTUNBA R, RUFAY R A, LIN J. MPR: multi-objective pairwise ranking[C]//The ACM Conference on Recommender Systems. ACM, 2017: 170-178.
- [22] WESTON J, BENGIO S, USUNIER N. WSABIE: scaling up to large vocabulary image annotation[C]//The International Joint Conference on Artificial Intelligence. AAAI Press, 2011: 2764-2770.
- [23] HE R, MCAULEY J. VBPR: visual Bayesian personalized ranking from implicit feedback[C]//The AAAI Conference on Artificial Intelligence. AAAI Press, 2016: 144-150.
- [24] WU B, YE Y, CHEN Y. Visual appearance or functional complementarity: which aspect affects your decision making?[J]. Information Sciences, 2019(476): 19-37.
- [25] 彭舰,王屯屯,陈瑜,等.基于跨平台的在线社交网络用户推荐研究[J].通信学报, 2018, 39(3): 147-158.
PENG J, WANG T T, CHEN Y, et al. User recommendation based on cross-platform online social networks[J]. Journal on Communications, 2018, 39(3): 147-158.
- [26] 朱敬华,明骞.LBSN 中融合信任与不信任关系的兴趣点推荐[J].通信学报, 2018, 39(7): 157-165.

ZHU J H, MING Q. POI recommendation by incorporating trust-distrust relationship in LBSN[J]. Journal on Communications, 2018, 39(7): 157-165.

- [27] 孙光福,吴乐,刘淇,等.基于时序行为的协同过滤推荐算法[J].软件学报, 2013, 24(11):2721-2733.
SUN G F, WU L, LIU Q, et al. Recommendations based on collaborative filtering by exploiting sequential behaviors[J]. Journal of Software, 2013, 24(11): 2721-2733.

[作者简介]



吴宾 (1991-)，男，河南柘城人，郑州大学博士生，主要研究方向为推荐系统、社交网络及多媒体。



陈允 (1990-)，女，河南虞城人，郑州大学硕士生，主要研究方向为推荐系统和社交网络。



孙中川 (1992-)，男，河南原阳人，郑州大学硕士生，主要研究方向为推荐系统和对抗网络。



叶阳东 (1962-)，男，河南潢川人，博士，郑州大学教授、博士生导师，主要研究方向为机器学习、智能系统、数据库等。