

# 基于隐式反馈数据的个性化游戏推荐

俞东进<sup>1</sup>, 陈聪<sup>1</sup>, 吴建华<sup>2</sup>, 陈耀旺<sup>1</sup>

(1. 杭州电子科技大学计算机学院, 浙江杭州 310018; 2. 杭州顺网科技股份有限公司, 浙江杭州 310012)

**摘要:** 现有推荐系统通常采用评分、评论等显式反馈数据实现个性化推荐. 然而, 显式反馈数据由于在实际中难以获取或因质量问题而往往变得不可用, 从而导致相关推荐算法的应用范围受到很大限制. 与此相反, 诸如点击行为、浏览记录等隐式反馈数据在现实中大量存在. 本文提出了一种面向游戏玩家的基于隐式反馈数据的游戏推荐方法. 该方法综合考虑了玩家操作次数、操作时长等隐式反馈数据及其时效性, 构建了基于伪评分的玩家对游戏的偏好模型, 而后通过改进了的SVD++ (Singular Value Decomposition++) 算法实现个性化游戏推荐. 在大规模真实数据集上的实验结果表明本文提出的方法具有更高的推荐精确率和召回率.

**关键词:** 推荐系统; 隐式反馈; 游戏推荐; 伪评分; SVD++

**中图分类号:** TP302 **文献标识码:** A **文章编号:** 0372-2112 (2018)11-2626-07

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2018.11.009

## Personalized Game Recommendation Based on Implicit Feedback

YU Dong-jin<sup>1</sup>, CHEN Cong<sup>1</sup>, WU Jian-hua<sup>2</sup>, CHEN Yao-wang<sup>1</sup>

(1. College of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China;

2. Hangzhou Shunwang Technology Co., Ltd., Hangzhou, Zhejiang 310012, China)

**Abstract:** The existing recommendation systems typically employ explicit feedbacks, such as ratings and reviews, to achieve personalized recommendations. However, since the explicit feedbacks are hard to obtain in practice or have poor quality, their applications in the recommendation field are limited. In contrast, implicit feedbacks such as click behaviors and browsing histories are widely available in reality. A game recommendation method based on implicit feedbacks is presented, to provide the personalized game recommendation for game players. This method integrates the time-based implicit feedbacks, such as the times and duration of user operations, and constructs the player's preference model based on the pseudo rating. Meanwhile, it gives the personalized game recommendation by the improved SVD++ (Singular Value Decomposition++) algorithm. The extensive results of the experiment based on the real data set show that the proposed method has the higher precision and recall.

**Key words:** recommendation system; implicit feedback; game recommendation; pseudo rating; singular value decomposition++ (SVD++)

## 1 引言

近年来, 推荐系统因其可以缓解信息过载问题和为用户提供个性化推荐服务而倍受工业界和学术界关注. 协同过滤推荐算法通过分析用户的历史行为数据预测用户对未知物品的喜好程度, 以此进行个性化推荐. 这里, 用户的历史行为数据可以分为能明确表达用户喜好倾向的显式反馈数据(如评分、评级、评论等), 以及不能直接表达用户喜好倾向的隐式反馈数据(如

浏览记录、购买历史、点击行为等)两大类.

现有推荐算法的研究主要集中在电影、音乐、电子商务等领域, 且大部分都是基于显式反馈数据. 而游戏与其他领域相比, 其具有更加独特的行为可重演性、操作灵活性和强交互性等特点. 比如大部分用户对同一部电影或同一篇新闻只会看一次. 虽然用户可能会循环播放同一首歌, 但每一首歌有固定的时长和内容, 因此游戏与音乐在行为重演性上也存在差异; 同时, 用户对游戏的操作更加灵活. 游戏中的情景、状态、行为都由

用户自己控制,而对于电影、新闻、歌曲等的内容,用户不管喜欢与否都只能被动地接受;此外,用户与游戏的交互性更强.与观看电影、新闻和欣赏歌曲过程中存在的单向弱交互不同,在用户操作游戏的过程中,用户行为与游戏反馈紧密联系又相互影响.因此一些应用在电影、新闻、音乐等领域的推荐算法并不能很好地被应用到游戏推荐中.另外,可用于游戏推荐的显式反馈数据在实际中往往难以获取,而诸如游戏时长等隐式反馈数据则在玩家日志中大量存在.因此,基于隐式反馈数据有望实现更有效的个性化游戏推荐.

本文提出了一种基于隐式反馈数据的网吧游戏推荐方法 DIIF-SVD++ (Discrete Integral Implicit Feedback SVD++). 首先,根据收集到的网吧玩家的游戏次数、游戏时长等历史日志数据,结合其数据时效性,构建玩家伪评分模型(即偏好模型).其次,在基于显式反馈数据的 SVD++ 推荐算法的基础上,通过去除其用户评分偏置项和隐式反馈数据部分、以及在损失函数中添加评分的权重系数,提出适用于基于隐式反馈数据游戏推荐的 IF-SVD++ (Implicit Feedback SVD++) 算法.最后,在真实数据集上对本文提出的伪评分模型以及 DIIF-SVD++ 算法进行有效性验证.实验结果表明本文提出的伪评分模型能更加准确地描述玩家的偏好,同时改进的推荐算法在精确率和推荐召回率上均高于对比方法.

## 2 相关工作

基于矩阵分解模型的推荐算法因其具有推荐准确度高、扩展性良好和灵活性较高等优点成为了目前推荐算法研究的最前沿领域之一.

最早的矩阵分解模型利用的是 SVD 分解<sup>[1]</sup>,但由于需要事先补全评分并且计算复杂度太高导致其在推荐系统领域的发展受到制约. Simon Funk 提出了 Funk-SVD<sup>①</sup> 算法,采用随机梯度下降法优化初始化的用户特征矩阵和物品特征矩阵,使其接近原始的评分矩阵,克服了以上两个缺陷. Koren 将 Funk-SVD 称为隐式特征模型(Latent Factor Model, LFM),在文献[2]中提出融合了用户评分偏差项、物品得分偏差项、全局评分平均项以及用户的隐式反馈数据的 SVD++ 模型. Koren 还考虑到时间因素的影响,在文献[3]提出了 TimeSVD 模型.除了在模型中引入一些偏置项和隐式反馈数据等基础数据之外,模型[4,5]中还考虑了用户的社交信息,文献[6]还将跨领域信息迁移到模型中.

然而以上推荐算法大多数都是基于显式评分数据,其并不能很好地适用于基于纯隐式反馈数据的推荐场景. Pan 等人<sup>[7]</sup>首次将基于隐式反馈数据的推荐问题称为单类协同过滤问题(One-Class Collaborative Filte-

ring, OCCF). 因为隐式反馈数据难以区分正负样本,为了防止模型发生偏斜,他们根据用户活跃度和物品流行度随机抽取物品作为负样本. Li 等人<sup>[8]</sup>对 OCCF 进行了扩展,通过引入更多的用户信息来提高算法的准确性. Wei Pan 和 Li Chen 在文献[9]中也对 OCCF 进行改进,通过引入群体偏好来减少对个人喜好过强的独立性假设.然而,单类协同过滤算法存在采样策略、采样结果集成等问题.文献[10]将时间信息作为第三维构成三维张量(Tensor),提出了快速张量分解的 LFM 方法. Delporte 等人<sup>[11]</sup>在 LFM 模型的基础上结合了用户的社交信息. Hu 等人<sup>[12]</sup>提出了样本置信度的概念,他们将用户操作过的物品设为 1、其他情况设为 0,以此构造用户对物品的 0-1 矩阵,并根据操作次数的不同赋予样本不同的权重,提出了带权重的最小二乘矩阵分解方法(Weighted Alternative Least Square, wALS).然而这种简单的 0-1 矩阵并不能很准确地表达用户的偏好. Levandoski 等人<sup>[13]</sup>将用户和物品之间的距离作为惩罚值构成伪评分,在算法中引入了地点信息.而 Yao 等人<sup>[14]</sup>采用随机游走的方法将用户到物品的排名分数作为伪评分,提出了基于图的上下文感知推荐模型.此外,还有一些基于学习排序的推荐算法,这些算法的关注点在于如何使推荐列表中用户越喜欢的物品排名越靠前,其主要可以分为逐点排序<sup>[15]</sup>、逐对排序<sup>[16]</sup>和逐表排序<sup>[17]</sup>.

## 3 模型设计

本节详细描述了本文提出的基于隐式反馈的个性化游戏推荐方法(DIIF-SVD++),其主要包含两部分:(1)玩家偏好模型的构建;(2)基于偏好模型对玩家进行游戏推荐.其总体流程如图 1 所示.

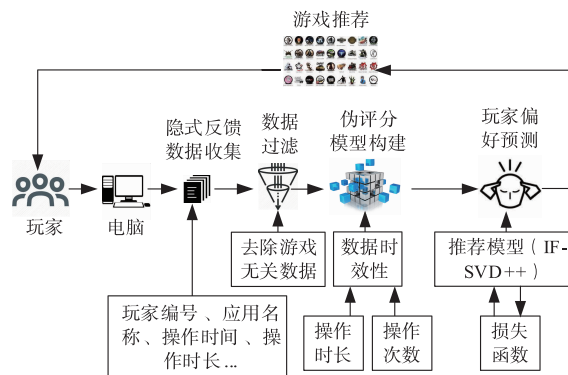


图1 基于隐式反馈的个性化游戏推荐方法(DIIF-SVD++)

### 3.1 伪评分模型构建

目前,大多数对推荐系统的研究都默认评分数据

① <http://sifter.org/~simon/journal/20061211.html>

是可靠的. 但是, 如果原始评分本身质量不高, 则在其基础上进行的推荐, 其有效性将受到很大影响. 由于游戏存在强交互性与可重演性等特点, 其隐式反馈数据相对于显式反馈数据, 内容更丰富, 且不受玩家主观偏见的影响, 因此更能体现玩家真实的偏好. 本文综合玩家历史操作次数和操作时长以及它们的时效性, 提出一种离散卷积型伪评分模型, 以有利于个性化精准推荐.

图 2 表示随时间  $T$  变迁的玩家历史行为记录, 每一小块矩形  $(t_{i,k}^u, d_{i,k}^u)$  代表玩家  $u$  对游戏  $i$  的一条操作记录, 其中  $t_{i,k}^u$  和  $d_{i,k}^u$  分别表示该玩家操作游戏  $i$  的第  $k$  条记录的开始时间和该次操作的持续时长.  $T_0^u$  表示在选取的  $K$  天集合内用户  $u$  的所有游戏操作记录中的最后一次操作开始时间.

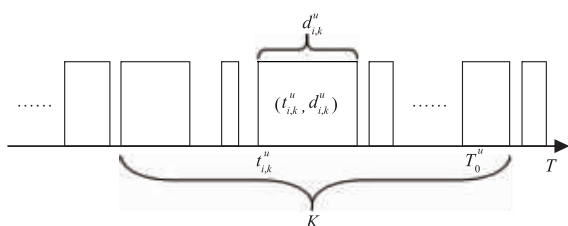


图2 某游戏玩家 $u$ 的历史行为示意图

根据遗忘规律公式  $\alpha \cdot t^\beta + \varphi$  (其中参数  $\alpha, \beta, \varphi$  可通过数据拟合得到), 本文提出的伪评分模型可通过公式(1)描述:

$$g_{u,i} = \sum_{k \in K} \{ d_{i,k}^u \cdot [32.76 \times (T_0^u - t_{i,k}^u)^{-0.0984}] \} \quad (1)$$

在公式(1)中,  $k \in K$  表示在  $K$  的时间范围内用户  $u$  对游戏  $i$  的所有操作记录.  $g_{u,i}$  表示玩家  $u$  对游戏  $i$  的伪评分, 其计算由两部分构成: 第一部分, 即  $d_{i,k}^u$ , 以第  $k$  次操作游戏  $i$  的时长作为本次操作的强度; 第二部分, 即  $32.76 \times (T_0^u - t_{i,k}^u)^{-0.0984}$ , 表示第  $k$  次操作游戏  $i$  对玩家  $u$  评价游戏  $i$  的影响大小, 具体参数根据艾宾浩斯遗忘数据<sup>①</sup>拟合得到. 若操作行为发生的日期越久远, 则该次行为对玩家偏好表达的影响越小.

伪评分通过隐式反馈数据的累加和得到, 其计算过程综合考虑了操作次数和操作时长的影响以及反馈数据的时效性, 这保证了评分的客观性与真实性, 同时因玩家偶尔的一两次误操作所计算得到的伪评分值很小, 故可以排除这类噪声数据对伪评分的影响. 此外, 因为可以根据伪评分的高低区分正、负反馈, 由此不但可以防止模型因缺乏负反馈而发生偏斜, 还可以避免由于引入负反馈而带来的噪声. 综上, 通过公式(1)计算得到的伪评分具有更高的质量, 有利于后期的推荐. 为了统一所有玩家的评价标准, 本文对  $g_{u,i}$  进行归一化处理, 使得最终的伪评分处于  $[0, 1]$  区间内, 如公式(2)所示, 其中  $m_u$  表示玩家  $u$  操作过的游戏款数.

$$\hat{g}_{u,i} = \frac{g_{u,i}}{\sum_{i=1}^{m_u} g_{u,i}} \quad (2)$$

### 3.2 玩家偏好预测

事实上, 很多基于显式评分的推荐算法并不能直接套用在伪评分上. 我们选用了基于显式评分的推荐系统中最具代表性 SVD++ 算法, 并在其基础上进行改进, 提出适用于基于伪评分的个性化游戏推荐算法 IF-SVD++.

首先, 通过公式(3)设置基线评分:

$$b_{u,i} = \mu + b_i \quad (3)$$

式中,  $b_{u,i}$  表示玩家  $u$  对游戏  $i$  的基线评分;  $b_i$  表示游戏与所有游戏平均评分的偏差;  $\mu$  表示所有游戏的平均评分, 为一固定值, 可以通过公式(4)计算得到:

$$\mu = \frac{\sum_{u=1}^n \sum_{i=1}^{m_u} g_{u,i}}{\sum_{u=1}^n m_u} \quad (4)$$

这里,  $n$  为所有玩家数. 相对于 SVD++ 中的基线评分, 本文舍去了不包含玩家主观评分所造成的评分偏差项.

基线评分因体现不出差异而不能很好地进行个性化推荐. 进一步添加玩家的个性化信息, 对玩家伪评分构成的矩阵进行分解, 得到相同潜在特征维度  $f$  下的游戏特征矩阵  $Q$  和玩家特征矩阵  $P$ , 将  $Q$  中游戏的特征向量  $q_i^T$  与  $P$  中玩家  $u$  的特征向量  $p_u$  两者相乘, 得到玩家  $u$  对游戏的伪评分预测值  $q_i^T \cdot p_u$ . 为了提高推荐结果的可解释性, 利用玩家操作过的游戏的特征向量  $(\frac{1}{\sqrt{m_u}} \sum_{i=1}^{m_u} q_i)$  来丰富玩家特征向量  $p_u$ . 因此, 添加了个性化因素之后玩家  $u$  对游戏  $i$  的预测伪评分  $\tilde{g}_{u,i}$  计算如公式(5)所示:

$$\begin{aligned} \tilde{g}_{u,i} &= b_{u,i} + q_i^T \cdot p_u \\ &= \mu + b_i + q_i^T \cdot \left( p_u + \frac{\sum_{i=1}^{m_u} q_i}{\sqrt{m_u}} \right) \end{aligned} \quad (5)$$

最后使用带正则项的  $F$  范数来量化重构误差, 得到公式(6)所示的损失函数:

$$\begin{aligned} L &= \sum_{u=1}^n \sum_{i=1}^{m_u} \left[ \frac{w_{u,i}}{2} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i})^2 \right] \\ &\quad + \frac{\lambda}{2} \cdot (b_i^2 + \|q_i\|^2 + \|p_u\|^2) \\ &= \sum_{u=1}^n \sum_{i=1}^{m_u} \left\{ \frac{w_{u,i}}{2} \cdot \left[ \hat{g}_{u,i} - \mu - b_i - q_i^T \right. \right. \end{aligned}$$

① <http://psychclassics.yorku.ca/Ebbinghaus/index.htm>

$$\cdot \left( \mathbf{p}_u + \frac{\sum_{i=1}^{m_u} \mathbf{q}_i}{\sqrt{m_u}} \right) \Bigg]^2 \Bigg\} + \frac{\lambda}{2} \cdot (b_i^2 + \|\mathbf{q}_i\|^2 + \|\mathbf{p}_u\|^2) \quad (6)$$

其中,  $w_{u,i}$  表示伪评分权重系数, 正则化参数  $\lambda$  用来防止模型过拟合. 在很多基于矩阵分解的推荐算法中, 很少有人考虑到评分的重要性差别. 本文认为不同的伪评分对损失函数的贡献度不一样, 对于伪评分越高的, 其预测值误差应更小, 因此为伪评分误差平方项增加了一个权重系数  $w_{u,i}$ . 在损失函数中, 伪评分越高其权重系数越大, 本文选取  $\hat{g}_{u,i}$  作为其预测误差的权重系数, 其也较符合游戏伪评分间权重的区分度.

为减少参数更新时的计算量, 本文采用随机梯度下降法来求在一定的迭代次数下该目标函数的最小值,  $D$  表示所随机选取的伪评分样本集, 将  $L$  分别对参数  $b_i, \mathbf{q}_i, \mathbf{p}_u$  求偏导:

$$\frac{\partial L}{\partial b_i} = - \sum_{(u,i) \in D} [\hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i})] + \lambda \cdot b_i \quad (7)$$

$$\frac{\partial L}{\partial \mathbf{q}_i} = - \sum_{(u,i) \in D} \left[ \hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i}) \cdot \left( \mathbf{p}_u + \frac{\sum_{i=1}^{m_u} \mathbf{q}_i}{\sqrt{m_u}} + \mathbf{q}_i \cdot \sqrt{m_u} \right) \right] + \lambda \cdot \mathbf{q}_i \quad (8)$$

$$\frac{\partial L}{\partial \mathbf{p}_u} = - \sum_{(u,i) \in D} [\hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i}) \cdot \mathbf{q}_i] + \lambda \cdot \mathbf{p}_u \quad (9)$$

然后用负梯度方向更新参数:

$$b_i \leftarrow b_i + \gamma \cdot [\hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i}) - \lambda \cdot b_i] \quad (10)$$

$$\mathbf{q}_i \leftarrow \mathbf{q}_i + \gamma \cdot \left[ \hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i}) \cdot \left( \mathbf{p}_u + \frac{\sum_{i=1}^{m_u} \mathbf{q}_i}{\sqrt{m_u}} + \mathbf{q}_i \cdot \sqrt{m_u} \right) - \lambda \cdot \mathbf{q}_i \right] \quad (11)$$

$$\mathbf{p}_u \leftarrow \mathbf{p}_u + \gamma \cdot [\hat{g}_{u,i} \cdot (\hat{g}_{u,i} - \tilde{g}_{u,i}) \cdot \mathbf{q}_i - \lambda \cdot \mathbf{p}_u] \quad (12)$$

其中参数更新步长  $\gamma$  和正则化参数  $\lambda$  通过交叉验证来确定. 根据参数更新方式, 本文提出的矩阵分解算法的时间复杂度为  $O(\psi \cdot n \cdot M_u \cdot f)$ , 其中  $\psi$  为模型更新的次数,  $M_u$  为玩游戏个数最多的那个玩家所玩的游戏个数.

## 4 实验评估

### 4.1 数据集与评估标准

实验采用某知名网吧平台运营商的游戏玩家行为数据集, 包括从 2016 年 6 月 1 日到 2016 年 10 月 1 日的 324,432,184 条玩家上网行为记录. 在清除与游戏无关的数据以及噪声数据后, 获得 31,787,488 条游戏操作记录, 涵盖 833,073 个玩家, 2,208 款游戏. 相关数据集

可从 <https://github.com/HDUCC/data> 下载. 为保护玩家隐私, 以上数据均经过匿名处理.

均方根误差 (Root Mean Square Error, RMSE) 被很多研究人员作为算法优劣的评估标准, 然而其并不适用于基于隐式反馈数据的推荐算法, 因为隐式反馈数据不包含显式评分, 无法计算 RMSE. 在实际应用中, 由于我们关注的是推荐结果是否符合玩家的口味, 而不是预测的评分与玩家评分的误差大小, 因此实验采用精确率 (precision) 和召回率 (recall) 作为算法的评估标准, 其计算方法如公式 (13)、(14) 所示. 其中,  $R_u$  是算法对于玩家  $u$  的推荐游戏集合,  $G_u$  是玩家  $u$  实际喜欢的游戏集合,  $T$  为测试集.

$$\text{precision} = \frac{1}{|T|} \sum_{u \in T} \frac{|R_u \cap G_u|}{|R_u|} \quad (13)$$

$$\text{recall} = \frac{1}{|T|} \sum_{u \in T} \frac{|R_u \cap G_u|}{|G_u|} \quad (14)$$

由于目前尚未有关于基于隐式反馈数据的游戏推荐算法, 同时考虑到数据集以及游戏本身的特殊性, 一些已有的基于隐式反馈数据的推荐算法也并不适用于游戏推荐, 故本文实验选择以下 5 种比较通用的推荐算法与本文方法 DIIF-SVD++ 进行对比:

(1) Popular: 即按照游戏的流行度进行推荐, 此处的流行度是根据游戏对应的玩家人数确定.

(2) wALS<sup>[12]</sup>: 将玩家操作过的物品设为 1、其他情况设为 0 构造 0-1 矩阵, 并根据游戏操作次数设置相对小的置信度权重, 将问题转化为 0-1 矩阵分解.

(3) TIF-SVD++ (Times-based IF-SVD++): 仅将玩家操作次数作为玩家对游戏的伪评分, 并用 IF-SVD++ 进行矩阵分解.

(4) DIF-SVD++ (Duration-based IF-SVD++): 仅将玩家操作时长作为玩家对游戏的伪评分, 并用 IF-SVD++ 进行矩阵分解.

(5) TDIF-SVD++ (Times-Duration-based IF-SVD++): 将玩家各次的操作时长除以时间差 ( $T_0^u, t_{i,k}^u$ ) 的累加和作为玩家对游戏的伪评分, 并用 IF-SVD++ 进行矩阵分解. 该方法未考虑遗忘规律对数据时效性的影响.

其中, Popular 算法和 wALS 算法作为与本文推荐方法的对比算法, TIF-SVD++、DIF-SVD++、TDIF-SVD++ 算法用于与本文伪评分模型的对比. 另外, 为便于标记, 本文采用 IF-SVD++ 统一表示基于隐式反馈的 SVD++ 推荐方法, 而不考虑其具体采用何种伪评分模型.

实验选取 70% 的伪评分作为训练集, 剩下的 30% 作为测试集, 并进行交叉实验, 取 3 次实验结果的平均值作为最终结果. 设置的潜在特征个数  $f=20$ , 正则化参数  $\lambda=0.05$ , 参数更新步长  $\gamma=0.01$ , 推荐的游戏数目  $h=2$ , 模型迭代次数  $\psi=20$ .

## 4.2 精确率和召回率

精确率随所采用的历史数据的时间跨度  $K$  的变化如图 3 所示.

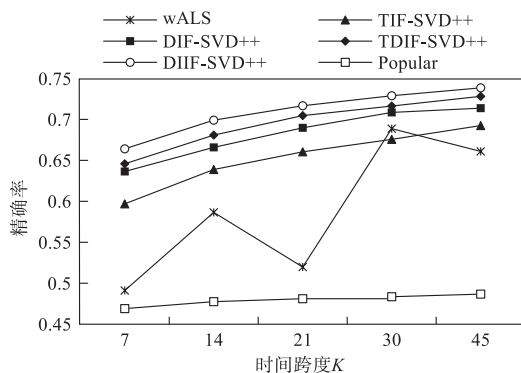


图3 不同时间跨度K下精确率对比

从图 3 中可以看出,本文提出的 DIIF-SVD++ 算法在  $K$  的不同取值范围内,其精确率均高于对比算法;同时,基于伪评分的各种算法(TIF-SVD++、DIF-SVD++、TDIF-SVD++、DIIF-SVD++)的精确率都比 Popular 算法和 wALS 算法要高,而 wALS 算法的精确率又比 Popular 算法要高,这说明了基于矩阵分解的算法的推荐精确率还是要高于基础的基于流行度的算法.除了 wALS 算法,其他基于伪评分的算法的精确率都随着  $K$  的增加而增加并且增速逐渐变缓,这说明采用的历史数据越多,玩家的伪评分模型变得越准确且趋于稳定,因此可以根据实际需求选取合适的  $K$  值而不必计算所有历史数据,从而大大减少计算量.另外,从图 3 可以看出,单纯以游戏时长作为伪评分的 DIF-SVD++ 的精确率要比单纯以游戏次数作为伪评分的 TIF-SVD++ 的精确率高,这也说明了以游戏时长评价玩家对游戏的偏好更加合理.最后,两种综合了游戏次数和游戏时长的伪评分(TDIF-SVD++、DIIF-SVD++)的精确率都比单纯使用游戏次数或游戏时长作为伪评分(TIF-SVD++、DIF-SVD++)的精确率要高,而基于本文提出的融合遗忘规律公式的伪评分的 DIIF-SVD++ 其预测效果又比未考虑遗忘规律的伪评分的 TDIF-SVD++ 的方法要好,这也证明了数据时效性对于预测精度具有重要意义,而且玩家对游戏的偏好程度并不是简单地反比于操作发生的时间差.

图 4 展示了实验召回率的结果,得出的结论与图 3 类似.

## 4.3 实验参数对精确率和召回率的影响分析

正则化参数  $\lambda$  用于约束拟优化的参数,以此来控制模型与真实场景的拟合程度.精确率和召回率随正则化参数  $\lambda$  的取值变化如图 5 所示. $\lambda$  取值太小容易造成模型过拟合,取值太大又会造成模型欠拟合.对于本

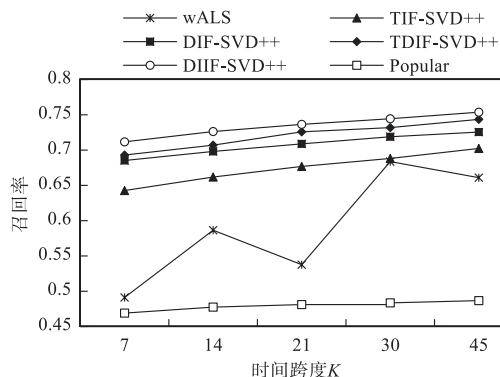


图4 不同时间跨度K下召回率对比

文实验, $\lambda$  取值在  $[0.05, 0.7]$  效果较好.

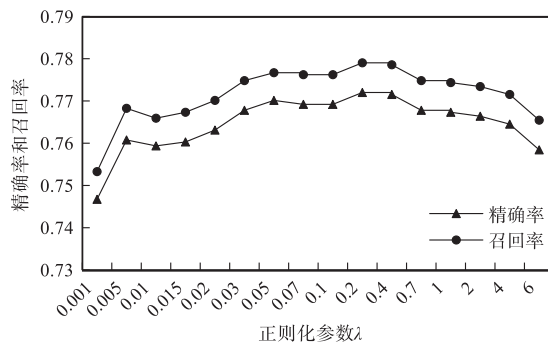


图5 不同 $\lambda$ 取值下精确率和召回率的变化

图 6 显示了精确率和召回率随参数更新步长  $\gamma$  取值的变化,在一定的迭代次数下,更新步长太小会导致模型未能收敛;而随着更新步长的增加,模型逐渐收敛,当模型收敛时,继续增加步长也就不再有效果.实际中,各个参数的最佳搭配需要根据大量的实验来确定.

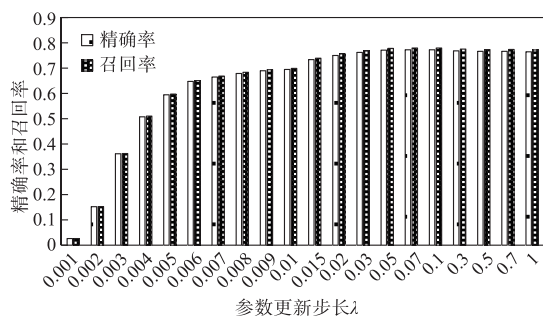


图6 不同 $\lambda$ 取值下精确率和召回率的变化

图 7 显示了精确率和召回率随潜在特征维度数  $f$  取值的变化.随着  $f$  的增加,精确率和召回率都缓慢下降.这说明对于游戏本身来说其潜在的特征较单一,主要靠关键的几个因素吸引玩家的兴趣.

图 8 显示了精确率和召回率如何随游戏推荐个数  $h$  的增加而变化.为了综合考虑精确率和召回率的变化,在这引入  $F$  值,其值为  $(2 \times \text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$ .从图 8 中可以看出随着游戏推荐个数的

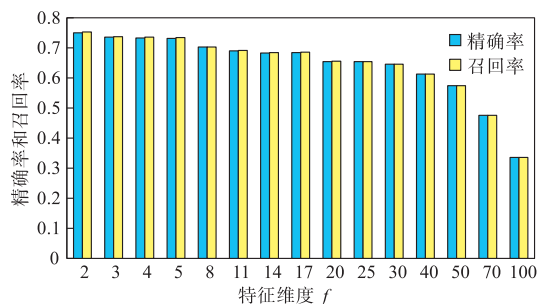


图7 不同特征维度f取值下精确率和召回率的变化

上升,召回率缓慢增长,但是精确率却急剧下降,从而F值也呈现下降趋势,这是由于大部分的玩家只是玩了游戏集中很少部分的游戏造成的.这也符合实际情况,即大部分的玩家通常只玩某几款游戏,从侧面也反映出给玩家推荐太多款游戏的意义不大.因此如果不是对游戏推荐个数有特殊要求,应尽量选取较少的游戏推荐个数.

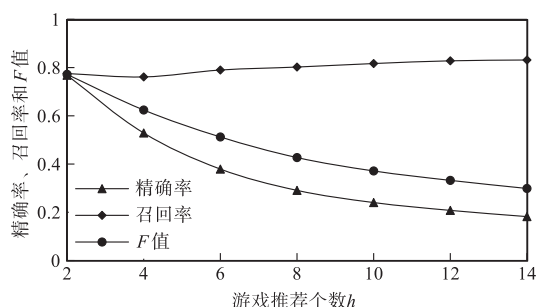


图8 游戏推荐个数h对精确率、召回率和F值的影响

#### 4.4 基于近邻的推荐算法对比分析

较少有人对采用隐式反馈数据的基于近邻的推荐算法进行实验,而伪评分可以近似地作为显式评分,故本文选取了较常用的基于近邻的协同过滤推荐算法进行实验:

(1)余弦相似度(Cosine Similarity, CS):其通过计算两个向量间的夹角大小来判断这两个向量的相似性,计算方法如公式(15)所示:

$$CS = \frac{\mathbf{U} \cdot \mathbf{V}}{\|\mathbf{U}\| \cdot \|\mathbf{V}\|} = \frac{\sum_{i=1}^m (U_i \cdot V_i)}{\sqrt{\sum_{i=1}^m U_i^2} \sqrt{\sum_{i=1}^m V_i^2}} \quad (15)$$

(2)皮尔逊相似度(Pearson Similarity, PS):其通过计算两个变量间的线性相关程度来表示相似性.计算方法如公式(16)所示:

$$PS = \frac{\sum_{i=1}^m [(U_i - \bar{U}) \cdot (V_i - \bar{V})]}{\sqrt{\sum_{i=1}^m (U_i - \bar{U})^2} \sqrt{\sum_{i=1}^m (V_i - \bar{V})^2}} \quad (16)$$

(3)斯皮尔曼相似度(Spearman Similarity, SS):其

通过计算排列后的变量之间的 Pearson 相关系数来表示相似性.

公式(15)、(16)中的  $\mathbf{U}$ 、 $\mathbf{V}$  代表玩家  $u$  和玩家  $v$  的各个伪评分  $U_i$ 、 $V_i$  构成的向量,  $\bar{U}$ 、 $\bar{V}$  分别是对应的平均伪评分,  $m$  表示所有游戏数目. 实验中选取 10 个最近邻, 推荐游戏个数为 2 个, 得到如图 9 和图 10 的精确率和召回率对比结果.

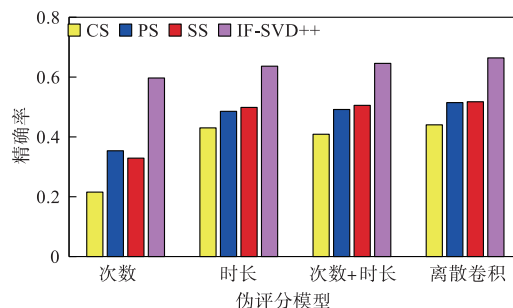


图9 不同近邻算法与IF-SVD++算法在不同伪评分下的精确率

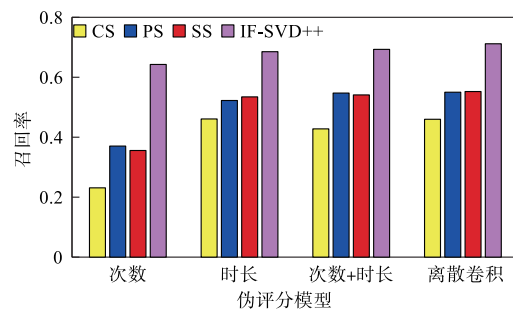


图10 不同近邻算法与IF-SVD++算法在不同伪评分下的召回率

从图 9 和图 10 可以看出,不管采用何种伪评分模型,本文提出的基于隐式反馈的推荐算法(IF-SVD++)的召回率和精确率均高于传统的基于近邻的推荐算法(CS、PS和SS).同时也可以看出,除了基于余弦相似度CS的近邻推荐算法之外,不管采用何种推荐算法(基于近邻、基于IF-SVD++),本文提出的离散卷积型伪评分模型与单独以次数、时长、次数+时长作为伪评分模型相比,其精确率和召回率都更高,这也说明了本文提出的伪评分模型更能准确地表达玩家的偏好.

综合以上的实验结果,不管是基于矩阵分解还是基于近邻的推荐算法,本文提出的伪评分模型相对于其他的伪评分模型更能准确地表达玩家的偏好,同时也可以看出本文提出的IF-SVD++算法推荐精确率和召回率更高.

## 5 总结

本文基于游戏领域具有的玩家行为可重演、人机交互性较强等特性,提出了一种表征玩家对游戏偏好的伪评分模型,并在此基础上通过改进了的SVD++算法实现个性化游戏推荐.在大量真实数据集上的实验

验证了本文方法的有效性。

未来我们将对以下问题作进一步研究:(1)伪评分模型的动态更新问题;(2)大数据量时的计算性能问题;(3)线上推荐的实际效果问题。

#### 参考文献

- [1] Billsus D, Pazzani M J. Learning collaborative information filters [A]. Proceedings of the 15th International Conference on Machine Learning [C]. Madison: ICML, 1998. 46 – 54.
- [2] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model [A]. Proceedings of the 14th International Conference on Knowledge Discovery and Data Mining [C]. Las Vegas: ACM, 2008. 426 – 434.
- [3] Koren Y. Collaborative filtering with temporal dynamics [J]. Communications of the ACM, 2010, 53(4): 89 – 97.
- [4] Yang B, Lei Y, Liu J, et al. Social collaborative filtering by trust [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(8): 1633 – 1647.
- [5] 俞春花, 刘学军, 等. 基于上下文相似度和社交网络的移动服务推荐方法 [J]. 电子学报, 2017, 45(6): 1530 – 1536.  
YU Chun-hua, LIU Xue-jun, et al. Mobile service recommendation based on context similarity and social network [J]. Acta Electronica Sinica, 2017, 45(6): 1530 – 1536. (in Chinese)
- [6] 王俊, 李石君, 等. 一种新的用于跨领域推荐的迁移学习模型 [J]. 计算机学报, 2017, 40(10): 2367 – 2380.  
WANG Jun, LI Shi-jun, et al. A new transfer learning model for cross-domain recommendation [J]. Chinese Journal of Computer, 2017, 40(10): 2367 – 2380. (in Chinese)
- [7] Pan R, Zhou Y, Cao B, et al. One-class collaborative filtering [A]. Proceedings of the 8th IEEE International Conference on Data Mining [C]. Pisa: IEEE, 2008. 502 – 511.
- [8] Li Y, Hu J, Zhai C X, et al. Improving one-class collaborative filtering by incorporating rich user information [A]. Proceedings of the 19th ACM International Conference on Information and Knowledge Management [C]. Toronto: ACM, 2010. 959 – 968.
- [9] Pan W, Chen L. GBPR: Group preference based bayesian personalized ranking for one-class collaborative filtering [A]. Proceedings of the 23rd International Joint Conference on Artificial Intelligence [C]. Beijing: IJCAI, 2013. 2691 – 2697.
- [10] Hidasi B, Tikk D. Fast ALS-based tensor factorization for context-aware recommendation from implicit feedback [J]. Machine Learning and Knowledge Discovery in Databases, 2012(2): 67 – 82.
- [11] Delporte J, Karatzoglou A, Matuszczyk T, et al. Socially enabled preference learning from implicit feedback data [A]. Joint European Conference on Machine Learning and Knowledge Discovery in Databases [C]. Berlin: EC-ML PKDD, 2013. 145 – 160.
- [12] Hu Y, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets [A]. Proceedings of the 8th IEEE International Conference on Data Mining [C]. Pisa: IEEE, 2008. 263 – 2721.
- [13] Levandoski J J, Sarwat M, Eldawy A, et al. Lars: A location-aware recommender system [A]. IEEE 28th International Conference on Data Engineering [C]. Washington: IEEE, 2012. 450 – 461.
- [14] Yao W, He J, Huang G, et al. A Graph-based model for context-aware recommendation using implicit feedback data [J]. World Wide Web, 2015, 18(5): 1351 – 1371.
- [15] Ostuni V C, Di Noia T, Di Sciascio E, et al. Top-n recommendations from implicit feedback leveraging linked open data [A]. Proceedings of the 7th ACM Conference on Recommender Systems [C]. Hong Kong: ACM, 2013. 85 – 92.
- [16] Li G, Ou W. Pairwise probabilistic matrix factorization for implicit feedback collaborative filtering [J]. Neurocomputing, 2016, 204: 17 – 25.
- [17] Wang S, Huang S, Liu T Y, et al. Ranking-oriented collaborative filtering: A Listwise Approach [J]. ACM Transactions on Information Systems, 2016, 35(2): 10.

#### 作者简介



俞东进 男,生于浙江平湖. 博士,教授,研究方向为大数据商务智能、软件仓库分析、服务计算等.  
E-mail: yudj@hdu.edu.cn



陈聪 男,生于浙江温州. 硕士研究生,研究方向为推荐系统.  
E-mail: chencong6837@foxmail.com