融合序列模式评分的策略梯度推荐算法

官 蕊 丁家满 贾连印 游进国 姜 瑛
(昆明理工大学信息工程与自动化学院 云南 昆明 650500)
(云南省人工智能重点实验室 云南 昆明 650500)

摘 要 推荐算法在一定程度上解决了信息过载问题,但传统推荐模型在挖掘数据特性方面有待改进。为此,结合强化学习方法提出一种融合序列模式评分的策略梯度推荐算法。将推荐过程建模为马尔可夫决策过程;分析推荐基础数据特性模式,设计以序列模式评分为奖励的反馈函数,在算法的每一次迭代过程中学习;通过对累积奖励设计标准化操作来降低策略梯度的方差。将该方法应用到电影推荐中进行验证,结果表明所提方法具有较好的推荐准确性。

关键词 强化学习 马尔可夫决策过程 策略梯度 序列模式

中图分类号 TP301.6 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2022.03.036

POLICY GRADIENT RECOMMENDATION ALGORITHM COMBINING SEQUENCE PATTERN RATING

Guan Rui Ding Jiaman Jia Lianyin You Jinguo Jiang Ying
(*Faculty of Information Engineering and Automation*, *Kunming University of Science and Technology*, *Kunming 650500*, *Yunnan*, *China*)
(*Artificial Intelligence Key Laboratory of Yunnan Province*, *Kunming 650500*, *Yunnan*, *China*)

Abstract The recommendation algorithm solves the problem of information overload to a certain extent, but the traditional recommendation model needs to be improved in mining the characteristics of the data. For these problems, we propose a policy gradient recommendation algorithm combining sequence pattern rating based on the reinforcement learning method. The recommendation process was modeled as Markov decision process; the characteristic pattern of recommended basic data was analyzed and a feedback function was designed with sequential pattern rating as reward to learn in each iteration of the algorithm; the variance of the policy gradient was reduced by designing a standardized operation on the cumulative reward; The method was applied to movie recommendation for verification, the experimental results showed that the proposed method has a good recommendation accuracy.

Keywords Reinforcement learning Markov decision process Policy gradient Sequence pattern

0 引 言

交互式推荐系统(IRS)在大多数个性化服务中起着关键作用[1]。不同于传统推荐方法将推荐过程定义为静态过程[2],IRS 连续向用户推荐商品并获得他们的反馈,从这种交互过程中学习推荐策略。IRS 方法分为两类:基于多臂赌博机(MAB)推荐和基于强化学习(RL)推荐。其中,基于 MAB 推荐尝试将交互推荐建模为 MAB 问题。Koren 等[3]和 Zeng 等[4]采用线性模型来估计各臂的置信上限(UCB)。此外,一些研究者尝试将 MAB 与矩阵分解技术相结合进行推荐[5]。这一类推荐方法无法实时适应用户偏好的改变,虽然实现最大化用户的当前收益,但其忽略了用户的长期收益。

强化学习在需要动态交互和长期规划(例如 Atari

收稿日期:2020-01-10。国家自然科学基金项目(61562054)。**官蕊**,硕士生,主研领域:数据挖掘,强化学习。**丁家满**,副教授。**贾连印**,副教授。**游进国**,副教授。**姜瑛**,教授。

游戏、棋类博弈和自动驾驶等)的多种场景中应用取得了显著成功[6]。近几年,众多学者应用强化学习解决推荐问题,显示出其处理 IRS 交互性的潜力。基于 RL 的推荐方法将推荐过程定义为马尔可夫决策过程(MDP),对用户状态进行建模,以最大化长期推荐奖励[7],这一类方法包括:基于值(Value-based)的方法(例如,Q-Learning)和基于策略(Policy-based)的方法(例如,策略梯度),构成了解决 RL 问题的经典方法[8]。Taghipour 等[9]提出将网页信息与 Q-Learning 算法结合解决网页推荐问题。有研究者在 Q-Learning 中引入值函数估计和记忆库机制,提出 Deep Q Network(DQN)方法[10]。Zhao 等[11]将正面和负面反馈均融入 DQN 框架,提出了将目标项目与竞争者项目之间 Q 值差异最大化,以引导推荐过程的方向和目标的统一。由于 Q-Learning 算法和 DQN 算法都是 Value-based 的学习方法,通过对 Bellman 方程进行迭代最终收敛到最优价值函数,这种方法计算量大,而且在一些特殊的场景下 Q 值难以计算[12]。作为一种 Policy-based 学习方法,策略梯度(policy gradient)则不存在这一问题,这种方法可以直接对策略进行学习。Chen 等[13]提出了一个基于层次聚类树的策略梯度推荐框架,通过寻找从树根到叶子的路径选取推荐项目。Chen 等[14]提出将离线策略梯度修正的方法用于动作空间数以百万计的 Youtube 在线推荐系统,解决了因只能从之前记录反馈而产生推荐的数据偏差问题。

上述推荐算法均获得了良好的推荐效果,但在挖掘数据特性方面有待改进。为此,本文提出一种融合序列模式评分的策略梯度推荐算法(Sequence Pattern Rating Recommendation,SPRR),首先分析评分数据的序列模式,设计融合序列模式评分的奖励作为交互式推荐的反馈信息;其次针对策略梯度方差大的问题,通过对累计奖励回报设计标准化操作来降低策略梯度的方差,学习更优的推荐策略,解决电影推荐问题。

1 问题定义

利用强化学习解决推荐问题,通常基于马尔可夫决策过程原理建立推荐过程模型。马尔可夫决策过程由状态集 S、动作集 A、奖励函数 R、状态转移函数 T 和策略函数 π 组成,利用五元组 $<S,A,R,T,\pi>$ 表示。其中,状态集 S 定义为用户和推荐系统的历史交互记录,其包含推荐项目、奖励回报和统计信息。在时间步 t,将状态 s_t 定义为二元组[item, reward],其中 item 和 reward 为推荐项目和相应用户反馈的奖励回报信息。为了对用户历史交互记录进行编码,受 Lei 等[15]提出的快速训练的简单递归单元 SRU 的启发,本文利用 RNN 模型来学习状态的隐藏表示。动作集 A 是智能体(Agent)可选的所有离散动作集合。所有可选的动作集合取决于当前的状态 s_t,表示为 $A(s_t)$。在时间步 t,选择一个推荐项目 a 推荐给用户。

奖励函数 $R(s,a)$ 也称为强化函数,是一种即时奖励或惩罚。推荐系统根据当前历史交互记录 s 向用户推荐项目 a 之后,用户反馈给推荐系统一个奖励,表示用户对该推荐项目的评价。状态转移函数 $T(s,a)$ 是一个描述环境状态转移的函数。由于状态是用户的历史交互记录,一旦推荐一个新的项目给用户,并受到用户的反馈,用户的状态也就发生了相应的变化。作为对时间步 t 选择执行动作 a_t 的结果,该函数将环境状态 s_t 转移到 s_{t+1}。策略函数 $\pi(a|s)$ 描述了 Agent 的行为,它是从环境状态到动作的一种映射。策略函数定义为所有可选的候选动作项目的概率分布。策略函数 $\pi(a|s)$ 表示为:

$$\pi(a_t|s_t;\theta) = \frac{\exp\{\theta^{\mathrm{T}}a_t\}}{\sum_{a\in A(s_t)}\exp\{\theta^{\mathrm{T}}a\}} \tag{1}$$

式中:θ 是策略参数。

包含上述 MDP 元素的序列(episode)为一次推荐过程,包含用户状态、推荐动作和用户反馈的序列(s_1, $a_1,r_1,s_2,a_2,r_2,\cdots,s_n,a_n,r_n,s_{n+1}$)。在此序列中,推荐算法根据用户状态 s_1 向用户推荐项目 a_1,用户反馈给推荐系统此次推荐的奖励回报 r_1,用户状态 s_1 相应的转变为 s_2,当序列到达满足预定义条件的状态 s_{n+1} 时结束。

2 SPRR 算法

2.1 奖励函数设计

奖励回报作为用户对推荐 Agent 行为的反馈,是指导强化学习方向的关键。奖励函数设置的优劣将直接影响算法的收敛速度和学习效果,因此在强化学习中至关重要。目前大多数基于 RL 推荐方法的奖励函数设置较单一,一般定义用户评分、用户点击量和用户购买量等标量信息作为反馈,未挖掘推荐过程中用户反馈行为的评分序列模式。用户评分历史记录中,除了反映用户对推荐项目的满意度外,在一定程度上也反映了用户的评分偏好。若用户的连续正面评分记录

越多,表示用户对已推荐项目的满意度越高,用户对于后续推荐项目的评分为正面的概率越大;若用户的连续负面评分记录越多,表示用户对已推荐项目的满意度较低,用户对于后续推荐项目的评分为负面的概率越大。

受文献[14]启发,本文在状态 s_t 下推荐动作 a_t 后,将用户反馈的奖励定义为两部分。一部分是经验奖励回报,定义为用户对推荐项目 a_t 的评分;另一部分是序列模式奖励回报,定义为在状态 s_t 下推荐项目 a_t 之前,用户的评分序列模式奖励。评分序列模式奖励定义为用户的连续正面平均评分和连续负面平均评分之差,利用计算正负面平均评分的方式学习用户的评分序列模式。奖励函数 $R(s,a)$ 计算式为:

$$R(s,a) = r_{ij} + \alpha(r_p - r_n) \tag{2}$$

式中:r_{ij} 是用户 i 对推荐项目 j 的评分,若用户 i 未对推荐项目 j 评分则为 0;r_p 是用户的连续正面平均评分,计算式为式(3);r_n 是用户连续负面平均评分计算式为式(4);α 为平衡参数用以权衡经验奖励回报和序列模式奖励回报的比重。

$$r_p = \frac{r_{ip_1} + r_{ip_2} + \cdots + r_{ip_{t-1}}}{t-1} \tag{3}$$

$$r_n = \frac{r_{in_1} + r_{in_2} + \cdots + r_{in_{t-1}}}{t-1} \tag{4}$$

式中:p_i 为连续正面评分计数;n_i 为连续负面评分计数。

2.2　改进的策略参数学习

本文通过强化学习中策略梯度方法 REINFORCE 来学习策略参数 θ。策略梯度方法是强化学习中的另一大分支。与 Value-based 的方法(Q-learning,Sarsa 算法)类似,REINFORCE 也需要同环境进行交互,不同的是它输出的不是动作的价值,而是所有可选动作的概率分布。SPRR 算法的目标是最大化期望累积折扣奖励 $J(\pi_\theta)$,表示为:

$$J(\pi_\theta) = E_{L\sim\pi_\theta}[G(L)] \tag{5}$$

式中:$L = \{x_{a_0}, x_{a_1}, \cdots, x_{a_N}\}$ 是推荐列表。$G(L)$ 定义为推荐列表的累积折扣奖励,其计算式为:

$$G(L) = \sum_{i=1}^{n} \gamma^{i-1} r_i \tag{6}$$

假定一个完整推荐序列的状态、动作和奖励回报的轨迹为 $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \cdots, s_n, a_n, r_n, s_{n+1})$,则期望累积折扣奖励 $J(\pi_\theta)$ 的梯度 $\nabla_\theta J(\pi_\theta)$ 的计算式为:

$$\nabla_\theta J(\pi_\theta) = G_t \nabla_\theta \log \pi_\theta(a_t | s_t; \theta) \tag{7}$$

利用该梯度调整策略参数,得到策略参数更新式为:

$$\theta = \theta + \eta \nabla_\theta J(\pi_\theta) \tag{8}$$

式中:η 为学习率,用来控制策略参数 θ 更新的速率。式(7)中的 $\nabla_\theta \log \pi_\theta(a_t | s_t; \theta)$ 梯度项表示能够提高推荐轨迹 τ 出现概率的方向,乘以累积折扣奖励之后,可以使得单个序列内累积奖励回报最高的轨迹 τ 越"用力拉拢"概率密度。即若收集了很多累积奖励回报不同的推荐轨迹,通过上述训练过程会使得概率密度向累积奖励回报更高的方向移动,最大化高奖励回报推荐轨迹 τ 出现的概率。

在某些序列中,由于每个序列的累积奖励回报都不为负,那么所有梯度 $\nabla_\theta J(\pi_\theta)$ 的值也均为大于等于 0 的。此时,在训练过程中收集的每个推荐轨迹,都会使概率密度向正的方向"拉拢",很大程度减缓了学习速率,使得梯度 $\nabla_\theta J(\pi_\theta)$ 的方差很大。因此,本文对累计奖励回报使用标准化操作来降低梯度 $\nabla_\theta J(\pi_\theta)$ 的方差。

本文通过设计基准来构造累积奖励回报。基准定义为 $n-1$ 条推荐序列累积奖励回报的均值。将当前推荐序列的累积折扣奖励减去基准,以保证累积奖励回报不总为正数。可以看出,一个序列内获得的奖励总和 $\sum_{i=t}^{n} \gamma^{i-t} r_i$ 超过 b 越多,对应的推荐轨迹被选中的概率越大。因此,累积奖励回报 G_t 计算式为:

$$G_t = \sum_{i=t}^{n} \gamma^{i-t} r_i - b \tag{9}$$

通过改进的累积奖励回报使得算法能提高总奖励回报较大的推荐轨迹的出现概率,同时降低总奖励回报较小的推荐轨迹的出现概率,保证策略参数 θ 沿着有利于产生最高总奖励回报的动作的方向移动,使好的动作得到更高的奖励。

本文提出的 SPRR 推荐算法的具体描述如下算法 1。算法 2 取样序列算法描述了算法 1 中获取取样序列的过程。

算法 1　SPRR 算法

输入:序列长度 N,候选推荐集合 D,学习率 η,折扣因子 γ,奖励函数 R。

输出:策略参数 θ。

1. for $j = 1$ to n do
2. 随机初始化策略参数 θ_j;
3. end for
4. $\theta = (\theta_1, \theta_2, \theta_3, \cdots, \theta_n)$;

5. repeat
6. $\Delta\theta=0$;
7. $(s_0,a_0,r_1,\cdots,s_{M-1},a_{M-1},r_M)\leftarrow$ SampleAnEpisode(θ,N,D,R); //算法 2
8. for $t=0$ to N do
9. $G_t=\sum_{i=t}^{n}\gamma^{i-t}r_i-b$; //式(9)
10. $\nabla_\theta J(\pi_\theta)=G_t\nabla_\theta\log\pi_\theta(a_t|s_t;\theta)$; //式(7)
11. end for
12. end for
13. $\theta=\theta+\eta\nabla_\theta J(\pi_\theta)$; //式(8)
14. until converge;
15. 返回 θ;

算法 2 SampleAnEpisode Algorithm 算法

输入:策略参数 θ,序列长度 N,奖励函数 R。

输出:推荐序列 E。

1. 初始化 $s_0=[0]$;
2. for $t=0$ to N do
3. 取样动作 $a_t\in A(s_t)\sim\pi(a_t|s_t;\theta)$ //式(1)
4. $r_{t+1}=R(s_t,a_t)$; //式(2)
5. if $t<N$ then
6. 转移 s_t 至 s_{t+1};
7. 添加(s_t,a_t,r_{t+1})至 E 的末端;
8. end for
9. 返回 E;

3 实验与结果分析

3.1 评分序列模式验证

实验采用 MovieLens(10M)和 Netflix 数据集。MovieLens(10M)数据集包含从 MovieLens 网站收集的 1 000 万条用户对电影的评分。Netflix 数据集包含从 Netflix 比赛中收集的 1 亿条评分。数据集的统计信息如表 1 所示。

表 1 数据集的统计信息

数据集	用户数	项目数	评分数	用户平均评分数	项目平均评分数
MovieLens	69 878	10 677	10 000 054	143	936
Netflix	48 189	17 770	100 498 277	209	5 655

本文对电影推荐中的数据集 MovieLens(10M)和 Netflix 进行实证分析,来验证推荐过程中评分序列模式的存在。两个数据集中均包含许多用户会话,每个会话根据时间戳包含用户对不同项目的评分且两个数据集均为五级评分。假定 3 分及以上的评分为正面评分,其他评分为负面评分。假设用户 u 对推荐项目 i 评分之前有 b 个连续正(负)评分的评级,将连续正(负)计数定义为 b。因此,每一推荐动作可以与特定的连续正(负)计数关联,设置 b 为 1 ~ 5,计算具有相同计数 b 的平均评分。图 1 - 图 2 给出了在两个数据集下不同连续正(负)计数下相应的平均评分,可以观察到用户评分行为存在的序列模式,即对连续正计数较大的项目用户,倾向于给出线性较高的评分;反之,连续负计数较大的项目用户倾向于给出线性较低的评分。原因可能是用户之前观看的电影越让其感兴趣,用户满意度越高。因此,用户倾向于对当前推荐的电影给予较高的评价。若用户之前观看的电影越让其不感兴趣,用户满意度越低。因此,用户倾向于对当前推荐的电影给予较低的评价。

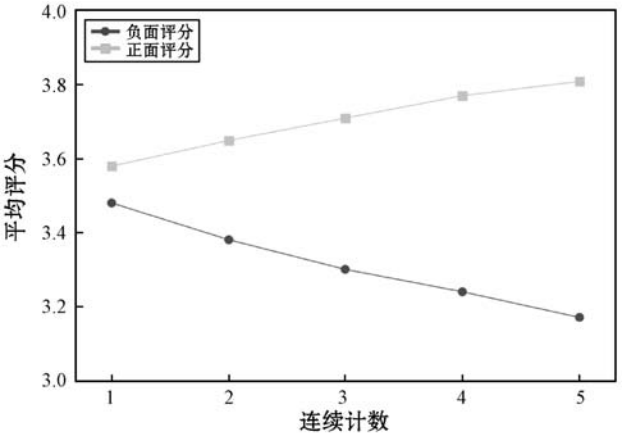

图 1 MovieLens 在不同连续计数下的平均评分

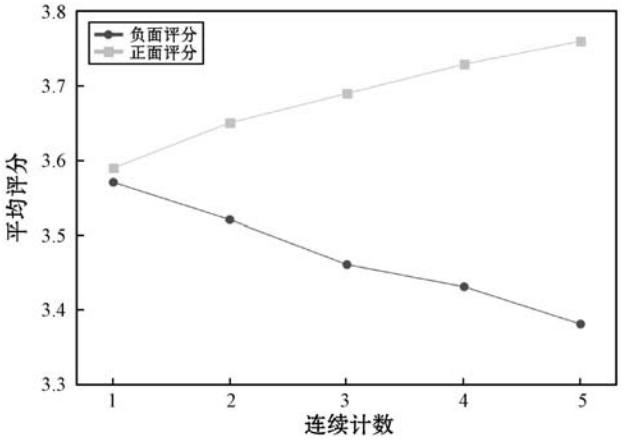

图 2 Netflix 在不同连续计数下的平均评分

3.2 结果分析

所有实验中,由于基于 RL 的方法的目标是获得最大的累积奖励回报,因此本文使用测试集中用户对推荐项目的平均奖励(Reward)作为一个评价指标。平均奖励指的是测试集中的用户对于算法推荐项目的评价。若平均奖励越大,说明用户对于推荐项目越感兴趣,整体满意度越大;反之则说明用户对于推荐项目

越不感兴趣,整体满意度越小。此外,采用了准确性(Precision@k)和归一化折扣累积增益(NDCG@k)作为评价指标。Precision 评测推荐的准确性,NDCG 评测推荐列表的优劣。算法目标是将用户感兴趣的项目尽量靠前推荐。k 分别取 10、20 和 30 验证算法效果。对于每个用户,所有评分大于等于 3.0 的项目都被视为用户感兴趣的相关项目,而小于 3.0 评分的项目则被视为用户不感兴趣的项目。

将两个数据集分别分为训练集和测试集,其中训练集占 80%,测试集占 20%。在所有实验中,根据训练经验分别设置学习率 η 和折扣因子 γ 为 0.02 和 0.9,奖励函数中的平衡参数 α 为 0、0.1 和 0.2 进行实验。当 $\alpha=0$ 时,此时奖励函数只考虑用户评分进行推荐。在每个推荐序列中,一旦一个推荐项被推荐给用户后,该项目将会从候选推荐集中删除,避免在一个推荐序列中重复推荐。将提出的 SPRR 算法同基于 MAB 的方法 HLinearUCB 和基于 RL 的方法 DQN-R 对比。表 2 列出了平衡参数取 0.2 时,SPRR 推荐算法和对比算法在平均奖励上的对比结果。

表 2　不同数据集的 Reward 值

算法	MovieLens Reward	Netflix Reward
HLinearUCB	0.122 5	0.124 7
DQN-R	0.254 0	0.253 3
SPRR	0.315 5	0.325 4

由表 2 可知,因基于 RL 的方法具有动态交互和长期规划的能力,与基于 MAB 的方法比较时,基于 RL 的方法 DQN-R 和 SPRR 均获得了较高的平均奖励。在基于 RL 的方法中,本文提出的 SPRR 取得了最高的平均奖励,在 MovieLens 和 Netflix 数据集上较 DQN-R 分别提高了 24% 和 28%。分析可知,SPRR 取得了较高的平均奖励主要有两个原因:(1) 融合序列模式评分的奖励给 SPRR 方法加入了额外的用户偏好信息;(2) 与传统的基于 RL 决策方法不同,本文提出的改进的策略参数更新方法,通过设计的基准可以实现让策略参数沿着有利于产生最高奖励回报的动作的方向移动,可以使用户可能感兴趣的动作得到更高的奖励,学习到更好的推荐策略。

表 3 列出了随着平衡参数的增大即在算法中不断增加序列模式奖励回报的比重,SPRR 推荐算法在两个数据集上的平均奖励的变化,用以验证评分序列模式对算法的影响。由表 3 可知,随着平衡参数 α 的递增,SPRR 算法在两个数据集上平均奖励均逐渐提高,在 Netflix 数据集上提高了近 71%。表明融合序列模式评分的奖励具有提高 SPRR 推荐平均奖励的能力。平均奖励的提升,也说明用户对于推荐项目满意度越大。

表 3　两个数据集下不同平衡参数的 Reward 值

SPRR	Reward(MovieLens)	Reward(Netflix)
$\alpha=0.0$	0.188 8	0.190 5
$\alpha=0.1$	0.249 6	0.257 7
$\alpha=0.2$	0.315 5	0.325 4

图 3 和图 4 是随着平衡参数 α 的递增在两个数据集上 SPRR 算法准确性。由图 3 和图 4 可知,随着 α 递增,一方面,k 取 10、20 和 30 时的算法准确性逐渐提高,当 $k=30$ 时,算法准确性最高;另一方面,在 α 取 0.2 时,算法在 k 取 10、20 和 30 时的整体准确性都达到最好的性能。

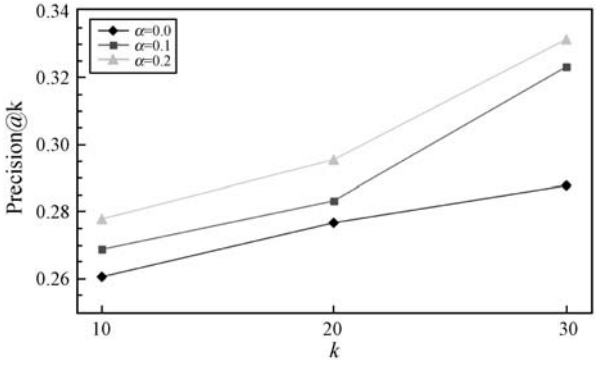

图 3　MovieLens 上的 Precision@k 值

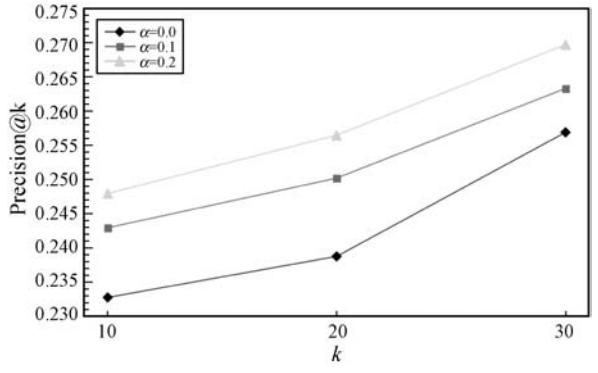

图 4　Netflix 上的 Precision@k 值

图 5 和图 6 是随着平衡参数 α 的递增在两个数据集上的 NDCG 值。由图 5 和图 6 可知,随着 α 递增,本文算法在 Netflix 数据集上 NDCG 有较明显的提升,推荐列表的质量较高。在 $\alpha=0.2$ 和 $k=30$ 时,算法的 NDCG 取得最高值,实现尽可能将用户感兴趣的项目靠前推荐,提升用户体验。分析可知,首先增加评分序列模式奖励的比重给算法增加了额外的用户评分偏好信息,使得算法的推荐准确性逐渐提高;其次本文通过奖励反馈信息不断调整推荐策略,k 值越大,使用户兴趣被策略参数更好地学习,算法准确性越高。

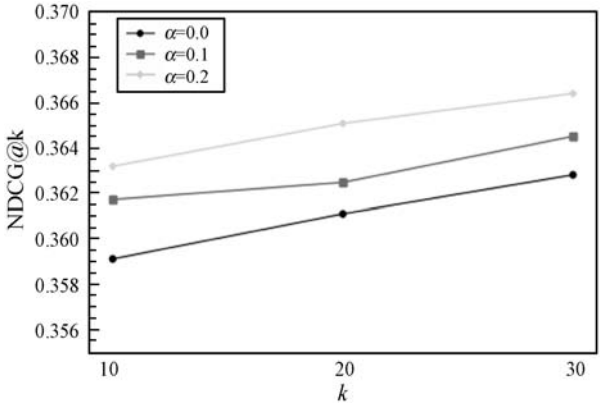

图 5 MovieLens 上的 NDCG@ k 值

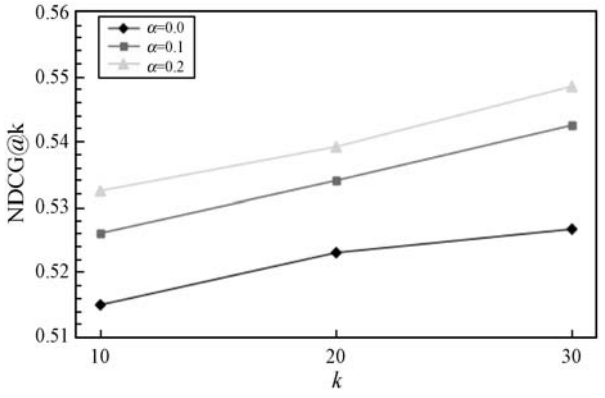

图 6 Netflix 上的 NDCG@ k 值

4 结 语

本文提出一种融合序列模式评分的策略梯度推荐算法。一方面,将推荐过程建模为马尔可夫决策过程,设计融合序列模式评分的奖励作为交互式推荐的反馈信息,帮助推荐;另一方面,通过对累计奖励回报设计标准化操作来降低策略梯度的方差,实现提高累积奖励较大的推荐轨迹的出现概率,同时降低累积奖励较小的推荐轨迹的出现概率,学习更优的推荐策略,解决推荐问题。实验结果表明,SPRR 推荐算法不仅有效,而且提高了推荐准确性。在以后的工作中,将继续挖掘影响推荐性能的因素,得到性能更优的推荐模型。

参 考 文 献

[1] Zhao X X,Zhang W N,Wang J. Interactive collaborative filtering[C]//Proceedings of the 22nd ACM International Conference on Information & Knowledge Management. ACM, 2013:1411 – 1420.

[2] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8):30 – 37.

[3] Zeng C Q, Wang Q, Molhtari S,et al. Online context-aware recommendation with time varying multi-armed bandit[C]// Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM,2016: 2025 – 2034.

[4] Wang H Z, Wu Q Y, Wang H N. Learning hidden features for contextual bandits[C]//Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. ACM, 2016: 1633 – 1642.

[5] Kawale J, Bui H, Kveton B, et al. Efficient Thompson sampling for online matrix-factorization recommendation[C]// Proceedings of the 28th International Conference on Neural Information Processing Systems. ACM,2015:1297 – 1305.

[6] Tan H H, Lu Z Y, Li W J. Neural network based reinforcement learning for real-time pushing on text stream[C]//Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2017:913 – 916.

[7] Wang Z, Freita N D, Lanctot M. Dueling network architectures for deep reinforcement learning[C]//Proceedings of the International Conference on Machine Learning. ACM, 2016: 1995 – 2003.

[8] Hasselt H V, Guez A, Silver D. Deep reinforcement learning with double Q-learning[EB]. arXiv:1509.06461,2016.

[9] Taghipour N,Kardan A. A hybrid web recommender system based on Q-learning[C]//Proceedings of the 2008 ACM symposium on Applied computing. ACM,2008:1164 – 1168.

[10] Mnih V,Kavukcuglu K,Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature,2015,518(7540):529 – 533.

[11] Zhao X Y,Zhang L,Ding Z, et al. Recommendations with negative feedback via pairwise deep reinforcement learning[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery& Data Mining. ACM, 2018:1040 – 1048.

[12] 刘文竹,黄勃,高永彬,等. Item2vec 与改进 DDPG 相融合的推荐算法[J]. 武汉大学学报(理学版),2019,65(3): 297 – 302.

[13] Chen H, Dai X, Cai H, et al. Large-scale interactive recommendation with tree-structured policy gradient[EB]. arXiv:1811.05869, 2018.

[14] Chen M, Beutel A, Covington P, et al. Top-K off-policy correction for a REINFORCE recommender system[EB]. arXiv:1812.02353,2018.

[15] Lei T, Zhang Y,Wang S, et al. Training RNNs as fast as CNNs[EB]. arXiv:1709.02755,2017.