知识增强策略引导的交互式强化推荐系统

张宇奇[1,2]，黄晓雯[1,2]，桑基韬[1,2]
1. 北京交通大学计算机与信息技术学院，北京 100044；
2. 交通数据分析与挖掘北京市重点实验室，北京 100044

摘要
推荐系统是解决社会媒体信息过载问题的重要手段。为了解决传统推荐系统无法优化用户长期体验的问题，研究人员提出了交互式推荐系统，并尝试使用深度强化学习优化推荐策略。但是，强化推荐算法面临反馈稀疏、从零学习影响用户体验、物品空间大等问题。为了解决上述问题，提出一种改进的知识增强策略引导的交互式强化推荐模型KGP-DQN。该模型构建行为知识图谱表示模块，将用户历史行为和知识图谱结合，解决反馈稀疏问题；构建策略初始化模块，根据用户历史行为为强化推荐系统提供初始化策略，解决从零学习影响用户体验的问题；构建候选集筛选模块，根据行为知识图谱上的物品表示进行动态聚类，从而减少物品空间，解决动作空间大的问题。在3个真实数据集上进行了实验，实验结果表明，KGP-DQN可以快速有效地对强化推荐系统进行训练，其在3个数据集上的推荐准确率均超过80%。

关键词
交互式推荐系统；深度强化学习；知识图谱；策略初始化；候选集筛选

中图分类号：TP391 文献标志码：A doi: 10.11959/j.issn.2096-0271.2022033

Knowledge-enhanced policy-guided interactive reinforcement recommendation system

ZHANG Yuqi[1,2], HUANG Xiaowen[1,2], SANG Jitao[1,2]
1. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
2. Beijing Key Lab of Traffic Data Analysis and Mining, Beijing 100044, China

Abstract
The recommendation system is an important means to solve the problem of information overload in social media. To solve the problem that traditional recommendation systems cannot optimize the long-term user experience, researchers have proposed the interactive recommendation system and tried to use deep reinforcement learning to optimize the strategy of recommendation. However, the reinforcement recommendation algorithm faces problems such as sparse feedback, learning from zero which damages the user experience, and large item space. To solve the above problems, an improved interactive reinforcement recommendation model KGP-DQN was proposed. The model constructed a behavioral

knowledge graph representation module, which combines user historical behavior and knowledge graph to solve the problem of sparse feedback. The model constructed a strategy initialization module to provide an initialization strategy for the reinforcement recommendation system based on user historical behaviors to solve the problem of learning from zero. The model constructed the candidate select module which creates candidates by dynamic clustering based on the item representation on the behavioral knowledge graph to solve the problem of large action space. The experiments were conducted on three real-world datasets. The experimental results show that the KGP-DQN method can quickly and effectively train the reinforcement recommendation system and its recommendation accuracy on three datasets is more than 80%.

Key words
interactive recommendation system, deep reinforcement learning, knowledge graph, policy initialization, candidate select

0 引言

随着网络的快速发展，信息过载问题越来越严重，人们难以及时有效地从海量数据中找到感兴趣的物品和信息。为了缓解信息过载问题，推荐系统应运而生。传统的推荐系统往往采用单步推荐的方式，导致推荐系统无法在推荐过程中动态学习用户的偏好。为了解决该问题，交互式推荐系统[1]被提出，并在近几年吸引了越来越多研究人员的关注。交互式推荐系统采用多步推荐的方式，在一次会话内进行多次推荐，并依据用户的反馈动态调整自身的推荐策略，从而为用户提供更准确的推荐结果。

由于深度强化学习在决策时关注动作的长期奖励，在动态环境中体现了较强的决策能力，因此，研究人员开始使用深度强化学习模型建模交互式推荐系统。Mahmood T等人[2]构建的基于model-based强化学习的交互式推荐系统和近些年提出的基于深度Q网络（deep Q network，DQN）的交互式推荐系统[3]都取得了不错的效果。基于深度强化学习的交互推荐系统能在推荐过程中灵活地调整推荐策略，提升推荐系统的准确率，并使用户长期获得良好的推荐体验。

尽管将深度强化学习技术应用到交互式推荐系统中取得了不错的进展，但基于深度强化学习的交互式推荐系统在实际应用中仍然面临巨大的挑战。深度强化学习的引入要求交互式推荐系统在与在线用户交互的过程中进行学习，从而避免离线学习的估计偏差问题。然而，在实际场景中，用户的反馈是非常稀疏的，而深度强化模型需要从零开始学习和试错的特性使得基于深度强化学习的交互式推荐系统需要大量的数据训练才可学习到最优策略，这一特性会影响用户的体验和推荐系统的收益。为了解决上述问题，研究人员在利用强化学习建模交互式推荐系统的同时，尝试引入知识图谱[4-5]。知识图谱将具有相同属性的物品节点通过边进行连接，使得用户对一个物品的反馈可以通过知识图谱中的边间接传播到其他物品。如**图1**所示，用户喜欢《盗梦空间》，其原因可能是他喜欢导演诺兰，那么他可能也会喜欢诺兰的其他电影，而在知识图谱中诺兰的其他电影与《盗梦空间》之间由于“诺兰”这一节点的存在而相互连接。因此，通过知识图谱，用户喜欢《盗梦空间》这条记录可以揭露出用户对其他相关电影（如《致命魔术》）的喜好。这使得对一个物品的反馈可以传播到其他在图谱中与

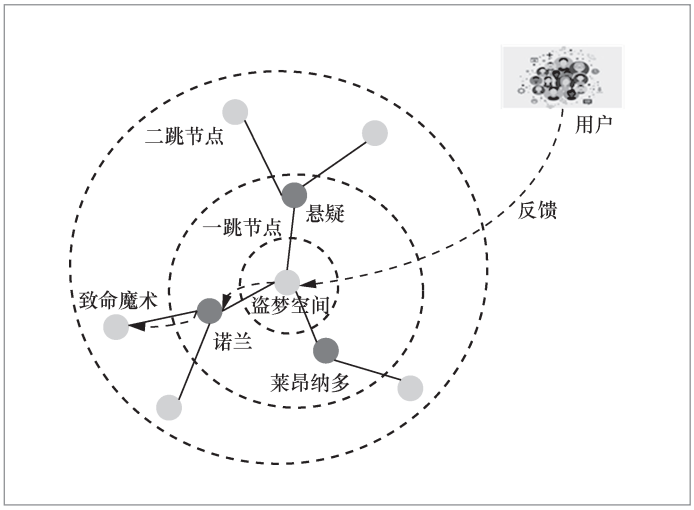


图1 知识图谱解决反馈稀疏问题示例

该物品相连的物品，通过挖掘图谱中符合用户喜好的物品，知识图谱可以有效地缓解反馈稀疏的问题。

然而，知识图谱虽然可以缓解强化推荐系统的反馈稀疏问题，但是仍无法解决在推荐初期需要从零学习导致的用户体验差的问题。对此，一个简单的解决思路是直接利用离线的历史数据训练推荐系统，再将其进行在线训练。这一思路存在的问题是离线训练的推荐效果与线上推荐效果存在偏差，导致最终的推荐效果无法达到全局最优。但离线收集的交互数据仍然具有一定的价值，依然可以用于指导交互式推荐系统进行训练。此外，强化推荐还面临动作空间大的问题，这严重影响推荐系统的效率和准确率。利用离线数据可为交互式推荐系统提供合适的候选集，减少动作空间，进而有效缓解上述问题。综上所述，离线数据可以为决策初始化一个合适的方向并制定合适的候选集，从而保证强化推荐系统的启动效果，最终获得一个优秀的策略。

基于以上思路，本文提出一种知识增强策略引导的交互式强化推荐系统，尝试将包含用户交互行为的知识图谱和深度强化学习算法结合，以解决上述问题。具体来说，本文基于深度策略网络提出了一种改进的知识增强策略引导的交互式强化推荐模型KGP-DQN（knowledge graph-based policy-guided DQN）。KGP-DQN包含3个模块：一是行为知识图谱表示模块，该模块构建行为知识图谱，利用图卷积网络（graph convolutional network，GCN）和循环神经网络对用户状态进行表示；二是策略初始化模块，该模块利用图卷积网络拟合用户的历史行为，为强化推荐生成初始化价值；三是候选集筛选模块，该模块利用物品的图表示进行动态聚类，生成候选集，缩小动作空间。

首先，KGP-DQN为了缓解反馈的稀疏性，将用户反馈和知识图谱结合，构建行为知识图谱，使用户的偏好在知识图谱中的相关项之间传递，使一个交互记录可以影响多个与之相连的物品，从而有效解决反馈稀疏性问题。其次，KGP-DQN通过行为知识图谱获得用户和物品的表示，并利用相关表示离线训练策略初始化网络，之后将策略初始化网络与强化推荐网络结合，使得推荐系统在在线训练时不需要从零学习，从而使整个强化推荐系统在在线训练的初期也能有良好的表现。最后，本文构建了候选集筛选模块来处理动作空间大的问题，根据初始化网络得到的物品表示对物品进行动态聚类，并在每一次推荐时依据初始化网络的结果对候选集进行筛选，使强化推荐Q网络可以更好地对结构表示相近的物品进行辨别，从而更好地提升推荐性能。

本文的贡献总结如下。

- 构建行为知识图谱，考虑图信息传播后物品之间的相关项，结合图卷积网络和门控循环单元（gated recurrent unit，GRU）学习物品与用户的先验表示，解决

反馈稀疏的问题。

● 构建策略初始化模块对推荐系统的强化训练进行初始化，降低从零学习在前期对用户体验的影响。

● 构建候选集筛选模块，根据物品的图表示进行动态聚类，生成推荐的候选集，有效地解决强化推荐中动作空间大的问题。

● 在3个真实数据集上设计了多组实验，实验结果表明，KGP-DQN可以在进行较少的交互轮次后达到很好的推荐性能，并且在训练初期也能有不错的表现。

1 相关工作

1.1 经典推荐算法

协同过滤算法[6]利用用户之间、物品之间的相似性来推荐物品；因子分解机算法[7]考虑通过特征之间的高阶组合来学习推荐策略，这些方法虽然取得了一定的成功，但是它们忽视了用户历史物品之间的时序关系。基于GRU的推荐方法[8]对用户的历史交互物品的序列关系进行建模并取得了不错的成就，深度兴趣网络（deep interest network，DIN）[9]在序列建模的基础上加入了注意力机制，从而进一步提升了推荐效果。然而，上述方法都是单步式的推荐，它们的优化目标均为最大化及时反馈，没有考虑用户的长期体验，为了进一步优化用户的长期体验，研究人员将深度强化学习引入推荐系统。

1.2 基于强化学习的推荐算法

Mahmood T等人[2]利用策略迭代的方法寻找最优策略。近几年，人们越来越多地使用基于模型的强化学习算法解决推荐问题。基于模型的强化推荐算法可以分为3类：基于策略梯度的推荐算法、基于DQN的推荐算法和基于深度确定性策略梯度（deep deterministic policy gradient，DDPG）的推荐算法。基于策略梯度的推荐算法直接对整个物品空间学习一个分布，并根据这个分布采用动作，从而完成推荐[10]。基于DQN的推荐算法为每一个物品计算一个Q值，之后选择Q值最大的物品作为最后的推荐物品。近几年，越来越多的研究人员喜欢基于DQN的强化推荐算法。Zheng G J等人[11]将DQN算法和DBGD（dueling bandit gradient descent）结合，用来解决新闻推荐的问题。Zou L X等人[12]在建模用户行为时将用户行为分为意图行为（点击、购买）和长期行为（停留时长、重新访问），再利用DQN算法解决推荐问题。Zhao X Y等人[3]将推荐中的负反馈也考虑进用户状态标志中，之后再利用DQN解决推荐问题。基于DDPG的推荐算法[13]将物品用一个连续向量表示。其中，动作家直接输出物品的向量表示，评论家则对这个物品进行打分。然而上述推荐方法忽视了用户之间、物品之间的相关性，并且推荐系统的反馈稀疏问题导致推荐策略效果不理想。为了充分考虑用户之间、物品之间的关联，在一定程度上解决反馈稀疏的问题，研究人员将知识图谱引入强化推荐系统。

1.3 基于知识图谱的强化推荐算法

基于知识图谱的强化推荐算法可以分为两类：基于知识图谱推理的算法和基于知识图谱表示的算法。基于知识图谱推理的强化推荐算法通过在知识图谱

上根据用户的历史行为进行图上推理，为用户推荐最可能喜欢的物品。Xian Y K等人[14]提出策略引导路径推理（policy-guided path reasoning，PGPR）方法，利用知识图谱进行明确的推理，从而进行推荐。基于知识图谱表示的强化推荐算法通过图卷积网络等技术，对知识图谱上的物品节点进行表示，并利用这个表示进行推荐。Zhou S J 等人[4]提出KGQN（knowledge graph enhanced Q-learning framework for interactive recommendation）方法，利用图卷积网络学习知识图谱中的物品表示，进而对用户进行表示，然后利用这些表示进行推荐。Wang P F 等人[5]提出KERL（knowledge-guided reinforcement learning）框架，该框架在深度强化推荐中利用知识图谱增强状态表示，从而达到更好的推荐效果。

虽然现有的基于知识图谱的强化推荐算法在一定程度上解决了强化推荐中的反馈稀疏问题，但是，它们仍面临动作空间大、从零学习影响用户体验等问题。为了解决这些问题，Dulac-Arnold G等人[13]提出利用k近邻（k-nearest neighbor，KNN）算法解决动作空间大的问题。Zhou S J 等人[4]提出在利用图网络解决反馈稀疏问题的同时，采用二跳节点来约束候选集，并在推荐初期采取基于流行度的策略进行推荐。但是，目前的强化推荐算法没有有效地利用用户的历史交互信息，仍无法在推荐前期为用户提供符合用户个性化偏好的策略，会严重影响用户体验。本文通过将知识图谱与用户历史行为结合，构建行为知识图谱表示模块、策略初始化模块、候选集筛选模块，从而解决现有强化推荐系统反馈稀疏、从零学习影响用户体验以及动作空间大的问题。

2 问题定义和建模

在交互式推荐系统中，推荐系统根据用户的历史行为为用户推荐可能喜欢的物品，用户接收推荐后做出反馈（购买、点击等行为），这个交互过程会一直持续到用户离开推荐系统。

传统的推荐算法在处理交互式推荐时只关注优化及时奖励，这导致在交互式推荐场景下不利于用户的长期体验。本文将交互式推荐系统建模为马尔可夫决策过程，利用与深度强化学习相关的算法解决上述问题。如**图2**所示，在强化交互式推荐场景下，对应的各个组件如下。

- 环境：所有用户和物品的信息。
- 智能体：推荐系统。
- 状态：访问推荐系统的用户的特征以及该用户访问过的物品的特征。
- 奖励：用户的反馈，如点击次数、浏览时间、回归时间等。
- 动作：推荐物品。
- 目标：最大化累积奖励。

本文将按照上述设定建模交互式推荐系统，并提出KGP-DQN模型来解决强化交互式推荐中存在的反馈稀疏、从零学习影响用户体验、动作空间大的问题。

3 KGP-DQN模型

下面将详细介绍本文提出的KGP-DQN模型。

KGP-DQN模型整体流程如**图3**所示，当用户访问推荐系统时，将用户的离线数据用图谱的形式进行表示，并将该行为图谱与知识图谱结合生成行为知识图谱，利用图卷积网络和循环神经网络生成用户的序列表示和物品的节点表示。根据物品的

节点表示对整个物品空间进行动态聚类，从而生成候选集。根据用户的序列表示对候选集中的物品进行聚类，从而生成初始策略。针对候选集中的物品，根据初始策略生成最终的推荐方案。

KGP-DQN模型示意图如**图4**所示。KGP-DQN模型包含3个关键模块：一是行为知识图谱表示模块，该模块利用行为知识图谱对用户的状态进行表示，以解决反馈稀疏问题；二是策略初始化模块，该模块生成初始化策略用于引导强化网络训练，以解决从零学习影响用户体验的问题；三是候选集筛选模块，该模块生成物品数量较小的候选集，以解决动作空间大的问题。

为了便于阐述和理解KGP-DQN模型，本文使用一些符号来进行简化，相关符号及其说明见**表1**。

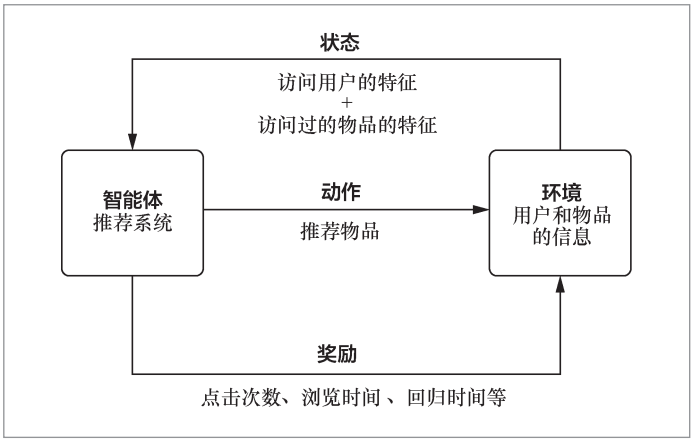


图 2 强化推荐框架

3.1 行为知识图谱表示模块

为了解决强化推荐系统中的反馈稀疏问题，KGP-DQN构建了行为知识图谱表示模块。首先，该模块将用户历史行为与知识图谱 $G=(E,R)$ 结合，将用户构建成节点并加入图谱中，之后将用户交互历史中的物品节点与用户节点相连，从而得到边类型为用户-物品-属性的、包含行为信息的知识图谱。

本文将该包含行为信息的知识图谱

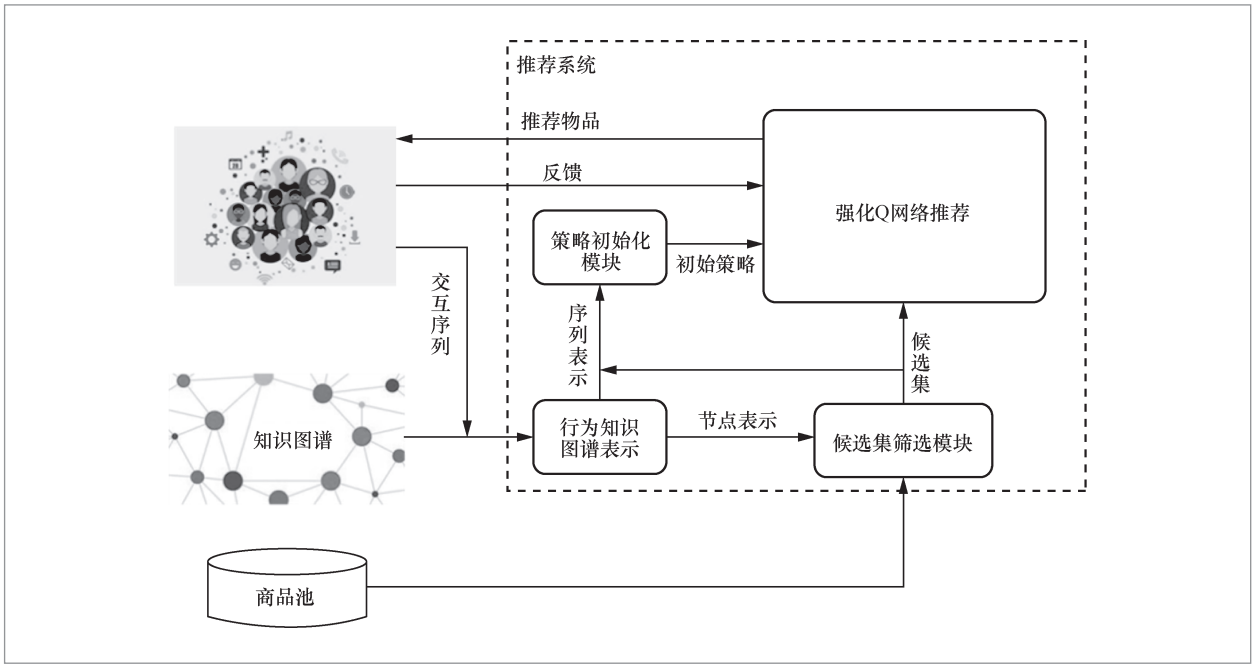


图 3 KGP-DQN 模型整体流程

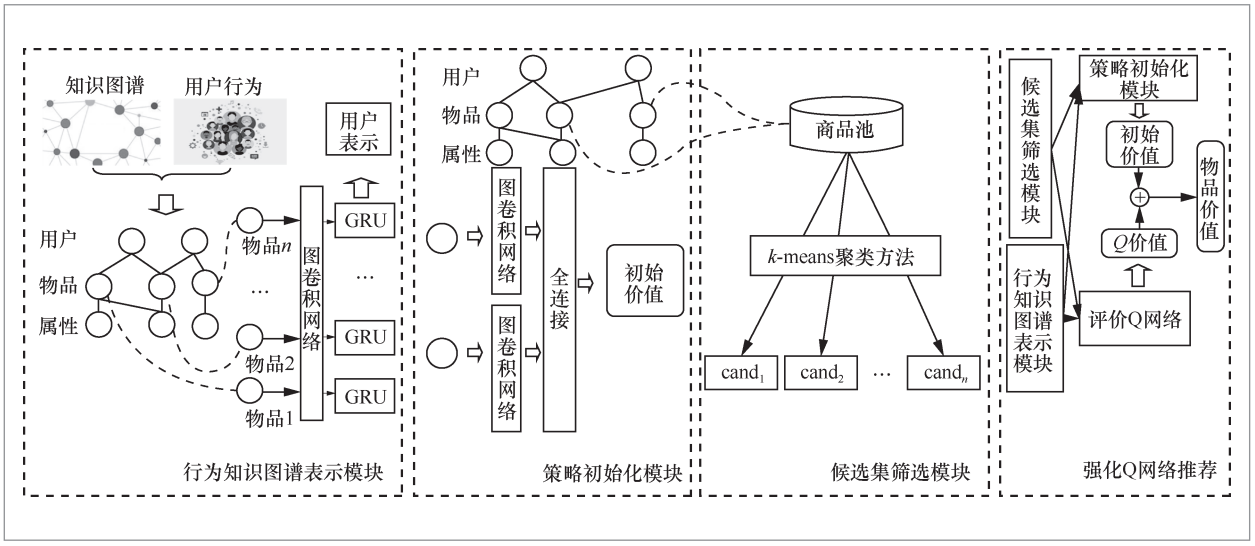


图 4 KGP–DQN 模型示意图

表 1 符号及其描述

符号	描述
$\boldsymbol{U}$	交互式推荐系统中的物品表示
$\boldsymbol{I}$	交互式推荐系统中的物品表示
$G=(E,R)$	G 为知识图谱，E 为点信息，R 为边信息
$G'=(E',R')$	G' 为行为知识图谱，E' 为点信息，R' 为边信息
$O_t=(i_1,i_2,i_3,\cdots,i_n)$	O_t 为用户的历史交互行为，i为用户访问过的物品
$\boldsymbol{S}_t$	t 时刻用户的表示
r_t	t 时刻用户给推荐系统的奖励
I_t	t 时刻推荐系统的候选集
θ_s	状态表示网络的参数
θ_q	推荐评价Q网络的参数
θ'_q	目标Q网络的参数
D	记忆库

命名为行为知识图谱 $G'=(E',R')$，并利用该图谱学习用户和物品的表示。在交互式推荐系统中，用户的交互行为代表了用户的偏好，因此该模块利用用户的历史交互行为得到用户的表示。对于每一个候选物品，先获取其向量表示 $\boldsymbol{i}_t\in R^d$，其中 d 表示向量表示的维度。之后，利用图卷积网络对图谱中的节点信息进行传播，从而获得更好的物品表示。

行为知识图谱表示模块利用多层图卷积网络学习图中的节点表示，每一层的计算步骤如下。

对于每一个节点，行为知识图谱表示模块计算其邻居表示：

$$\boldsymbol{e}_{N(h)}^{k-1}=\frac{1}{|N(h)|}\sum_{i\in N(h)}\boldsymbol{e}_i^{k-1} \tag{1}$$

其中，$N(h)$表示节点的邻居。之后，行为知识图谱表示模块利用邻居表示对节点本身的表示进行更新：

$$\boldsymbol{e}_h^k=\sigma\left(\boldsymbol{W}_k\boldsymbol{e}_{N(h)}^{k-1}+\boldsymbol{B}_k\boldsymbol{e}_h^{k-1}\right) \tag{2}$$

其中，$\boldsymbol{W}_k$ 和 $\boldsymbol{B}_k$ 是可学习的网络参数，σ 是激活函数，$\boldsymbol{e}_h^k$ 为节点 h 第 k 次传播后的表示。这里选择ReLU函数，计算式为 ReLU=max(0,x)。通过多层的图卷积网络计算后，行为知识图谱表示模块得到知识图谱上物品的节点表示 $\boldsymbol{i}_t$ 和用户的节点表示 $\boldsymbol{u}_t$。用户历史信息代表了用户偏好且历

史信息是一个序列化的信息，因此行为知识图谱表示模块通过循环神经网络利用用户的近期历史信息对用户进行表示。行为知识图谱表示模块引入GRU神经网络来建模用户的表示。每个GRU的细胞单元更新如下：

$$\boldsymbol{z}_t = \sigma_g\left(\boldsymbol{W}_z\boldsymbol{i}_t + \boldsymbol{U}_z\boldsymbol{h}_{t-1} + \boldsymbol{b}_z\right) \quad (3)$$

$$\boldsymbol{r}_t = \sigma_g\left(\boldsymbol{W}_z\boldsymbol{i}_t + \boldsymbol{U}_z\boldsymbol{h}_{t-1} + \boldsymbol{b}_r\right) \quad (4)$$

$$\boldsymbol{h}'_t = \sigma_h(\boldsymbol{W}_h\boldsymbol{i}_t + \boldsymbol{U}_h(\boldsymbol{r}_t \circ \boldsymbol{h}_{t-1}) + \boldsymbol{b}_h) \quad (5)$$

$$\boldsymbol{h}_t = (1-\boldsymbol{z}_t)\circ\boldsymbol{h}_{t-1} + \boldsymbol{z}_t \circ \boldsymbol{h}'_t \quad (6)$$

其中，$\boldsymbol{z}_t$ 和 $\boldsymbol{r}_t$ 分别表示更新门和重置门的输出向量，$\boldsymbol{W}$ 和 $\boldsymbol{U}$ 表示网络的权重矩阵，$\boldsymbol{b}$ 表示网络的偏置矩阵，σ 表示激活函数，$\circ$ 表示点乘操作。隐藏层表示 $\boldsymbol{h}_t$ 的更新采用线性结合方式，将前一次计算得到的 $\boldsymbol{h}_t$ 和本次更新的新隐藏层候选 $\boldsymbol{h}_{t-1}$ 结合。隐藏层表示 $\boldsymbol{h}_t$ 被看作 t 时刻用户的向量表示，这种表示是一种带有序列特征的表示。之后，隐藏层表示向量 $\boldsymbol{h}_t$ 会被输入强化Q网络中。为了简化表达式，该部分的所有网络参数被表示成 θ_s，包括卷积神经网络部分和GRU部分。

在**图4**中，行为知识图谱表示模块将用户历史交互物品的节点表示 $\boldsymbol{i}_t$ 输入GRU，得到GRU的隐藏层表示 $\boldsymbol{h}_t$ 并将其作为用户的序列表示；其中，关于这部分的网络参数 θ_s 如何进行训练的问题，会在第3.4节进行介绍。通过行为知识图谱表示模块，KGP-DQN得到的物品表示是包含知识图谱邻居信息的，从而有效缓解了反馈稀疏的问题。

3.2 策略初始化模块

为了解决从零开始训练导致的前期对用户体验带来严重影响的问题，本文构建了策略初始化模块。该模块利用用户的历史交互数据，生成初始化策略，为推荐系统提供先验策略，从而指导推荐系统进行训练，并保证推荐系统在推荐初期也会有不错的效果。策略初始化模块利用行为知识图谱，通过对用户的历史行为进行学习，得到可以代表用户对候选物品的喜欢程度的物品初始价值，并依据这个初始价值对强化推荐提供指导。策略初始化模块利用图卷积网络对行为知识图谱上的用户和物品节点进行表示，图卷积网络具有节点传播的特点，因此该表示可以体现出用户对物品及其相关属性的偏好，并且该表示包含了一定的结构信息。得到该表示后，策略初始化模块搭建全连接网络，根据用户的历史行为学习用户对物品的兴趣：

$$\boldsymbol{L} = \sigma\left(\boldsymbol{W}_1\mathrm{concat}(\boldsymbol{i},\boldsymbol{u}) + \boldsymbol{b}_1\right) \quad (7)$$

$$V = \sigma\left(\boldsymbol{W}_2 L + \boldsymbol{b}_2\right) \quad (8)$$

其中，$\boldsymbol{W}$ 表示全连接网络的权重矩阵，$\boldsymbol{b}$ 表示神经网络的偏置矩阵，$\boldsymbol{L}$ 表示全连接层得到的隐层表示，$\mathrm{concat}(\boldsymbol{i},\boldsymbol{u})$ 表示将用户表示和物品表示以拼接的方式结合，V 表示最终得到的物品初始价值。为了便于进一步说明，将兴趣表示为 $V_{u,i} = P(\boldsymbol{u},\boldsymbol{i})$。

在**图4**中，策略初始化模块将行为知识图谱表示模块构建的行为知识图谱作为输入，通过图卷积网络学习用户的表示，不同于行为知识图谱表示模块得到的用户序列表示，该表示被认为包含更多结构特征。之后，该模块根据用户的节点表示$\boldsymbol{i}$和物品的节点$\boldsymbol{u}$得到用户对某一物品的兴趣价值 $V_{u,i}$，并根据该兴趣价值为推荐系统初始化策略。策略初始化模块与强化推荐系统的结合方式以及策略初始化模块进一步的训练方式将在第3.4节介绍。通过策略初始化模块，KGP-DQN得到了候选物品的初始化价值，该价值在推荐前期作为推荐的指导，从而缓解强化推荐从零学习导致的用户体验降低问题。

3.3 候选集筛选模块

为了解决动作空间大的问题，本文构建了候选集筛选模块。该模块选择使用在策略初始化模块得到的物品的节点表示 i 进行动态聚类，即在整个推荐流程中每推荐 τ 次，对整个物品空间进行一次重新聚类，本次聚类的初始化类中心为 $\tau-1$ 次聚类的聚类中心。采用动态聚类的原因是在推荐过程中，物品在图谱中的表示会随着用户交互行为的增加而变化，定时的重新聚类可以保证候选集筛选模块适应这些变化，更好地根据物品本身的特征以及用户对物品的行为来生成最优的候选集。

该模块选择k-means聚类作为候选集筛选模块中的聚类方式，具体流程如算法1所示。完成聚类后，将类中心的表示以及用户的节点表示输入策略初始化模块中，得到用户对每一个簇的兴趣价值，选择兴趣价值最高的簇作为用户的基本候选集。同时，从该簇之外的其他簇中随机采样一定数量的物品，并将其加入基本候选集中得到最终的候选集。本文认为通过候选集筛选模块得到的候选集中包含的物品是用户最感兴趣的一批物品，同时，由于这批物品的表示比较相似，模型在该候选集中学习的时候，也可以学习到物品之间的一些更具有代表性的特征，从而进一步提升推荐的效果。本文将候选集筛选模块表示为 $I=I'(G',\boldsymbol{u})$。

算法1：候选集筛选

输入：物品的表示$\boldsymbol{e}$

输出：候选集I_t

随机化n个聚类中心

重复以下步骤直到收敛：

对于每一个物品，计算它属于的类

$$c^i=\operatorname{argmin}_j(\| \boldsymbol{e}_i-\boldsymbol{u}_j\|)$$

对于每一个类，计算类中心

$$\boldsymbol{u}_j=\frac{\sum_{i=1}^{m}\{c^i=j\}\boldsymbol{e}_i}{\sum_{i=1}^{m}\{c^i=j\}}$$

在**图4**中，候选集筛选模块利用策略初始化模块生成的物品表示进行动态聚类，将类中心的表示与用户序列表示输入策略初始化模块中，得到初始兴趣价值并以此选择候选集。最后，每推荐 τ 次，候选集筛选模块就会对整个物品空间进行重新聚类，以得到新的候选集。通过候选集筛选模块，KGP-DQN将整个候选集空间分割成n个子候选集，其中每个子候选集中的物品数量远小于整个候选集中的物品数量，从而有效解决了强化推荐动作空间大的问题。

3.4 强化Q网络推荐

获得用户表示、候选集以及用户的初始兴趣后，本文构建了强化推荐模块。该模块设计了一个Q网络来结合这些信息以及训练前文提到的模块，从而更好地提升推荐系统的性能。该模块使用double-DQN[15]训练整个推荐系统。

该模块通过试错的方式训练模型的参数。在交互式推荐系统的过程中，推荐系统在接收到用户的表示后，将策略初始化输出的初始价值与Q网络输出的Q值结合，得到最终的价值 Q'：

$$Q'=\alpha Q+(1-\alpha)V \tag{9}$$

其中，α 为策略初始化模块和强化推荐模块的结合参数。然后通过 ε 贪心算法推荐物品（以概率 ε 随机推荐，以$(1-\varepsilon)$的概率按照最大的 Q' 价值推荐候选集中的物品）。完成推荐后，推荐系统获得用户的反馈，并将该反馈作为奖励。之后，这次推荐会以经验的方式存储在记忆库中。每次更新时，Q网络会从记忆库中采样一部分经验，使用均方误差损失函数对Q网络进

行更新，同时也会对策略初始化表示进行更新。

$$L(\theta_q)=E_{(o_t,i_t,r_t,o_{t+1})\sim D}\left|y_t-Q(\boldsymbol{s}_t,\boldsymbol{i}_t;\theta_q)\right|^2 \quad (10)$$

$$L(\theta_p)=E_{(o_t,i_t,r_t,o_{t+1})D}\left|y_t-P(\boldsymbol{u},\boldsymbol{i})\right|^2 \quad (11)$$

其中，θ_q、θ_p分别表示Q网络和策略初始化网络的参数，y_t是根据最优Q值计算出的目标值。

$$y_t=r_t+\gamma\max_{i_{t+1}\in I_{t+1}}Q(\boldsymbol{s}_t,\boldsymbol{i}_t;\theta_q) \quad (12)$$

其中，γ表示强化学习中奖励的折扣因子。

为了解决传统DQN过估计的问题，该模块在使用评价Q网络的同时使用了一个目标Q网络。评价Q网络进行正常的反向传播并在每个训练步更新自己的参数；目标Q网络是评价Q网络的一个复制，每隔一定的训练步长会根据评价Q网络的参数进行更新。

$$y_t=r_t+\gamma\,Q'\left(s_{t+1},\arg\max_{i_{t+1}\in I_{t+1}}Q(\boldsymbol{s}_t,\boldsymbol{i}_t;\theta_q);\theta'_q\right) \quad (13)$$

其中，θ'_q表示目标Q网络的参数，该模块采用软更新的方式对这个参数进行更新：

$$\theta'_q=\tau\theta_q+(1-\tau)\theta'_q \quad (14)$$

在**图4**中，强化Q网络推荐模块将行为知识图谱表示模块得到的用户序列表示和物品表示作为输入，利用策略初始化模块对候选集筛选模块得到的候选集中的每个物品计算初始价值，再将初始价值与Q网络的输出结合，从而得到最终的价值。

KGP-DQN的整体训练流程如算法2所示。本文提出的KGP-DQN主要关注如何利用知识提高交互式推荐系统的采样效率，并在推荐前期为用户提供一个较好的策略。KGP-DQN模型中的强化学习算法可以被其他强化学习算法代替。

算法2：训练KGP-DQN

输入：D、α、τ、ε

输出：θ_q、θ_p、θ_s

初始化θ_q、θ_p、θ_s、$\theta_q\rightarrow\theta'_q$

重复以下步骤直到收敛：

对于一个用户$\boldsymbol{u}$，推荐T次

根据用户交互历史o_t，得到用户序列表示$\boldsymbol{s}_t$

- 根据行为图谱，得到$\boldsymbol{u}_t$、$\boldsymbol{i}_t$
- 得到候选集$I=I'(G',\boldsymbol{u})$

得到初始价值$V_{u,i}=P(\boldsymbol{u}_t,\boldsymbol{i}_t;\theta_p)$

得到Q价值$Q=Q\left(\boldsymbol{s}_t,\boldsymbol{i}_t;\theta_q\right)$

得到最终价值$Q'=\alpha Q+(1-\alpha)V$

根据Q'按照ε贪心策略进行推荐

- 获得用户的反馈r_t

如果$r_t>0$，将$\boldsymbol{i}$加入用户交互历史

将$\left(\boldsymbol{i}_t,o_t,r_t,o_{t+1},I_t\right)$存入$D$中

从D中随机采样一部分数据

根据o_t、o_{t+1}得到用户表示$\boldsymbol{s}_t$、$\boldsymbol{s}_{t+1}$

- 得到y_t

通过SGD方法更新θ_q、θ_p、θ_s

- 更新θ'_q

4 实验结果及分析

4.1 数据集

本文使用3个真实数据集来测试KGP-DQN模型是否有效。

- MovieLens-1m数据集：包含986 495条数据，这些数据的评分从1到5。本文根据数据中的电影从TMDB网站进行爬虫，爬取电影的类型、导演、职员等15种属性，从而得到电影的知识图谱。对于MovieLens-1m数据集，本文将评分高于3.5分的数据作为正样本，标签为1；将评分低于3.5分的数据作为负样本，标签为0。

● Last.fm数据集：包含76 693条数据，这些数据没有评分信息。本文利用数据集中的物品属性信息构建知识图谱，共有33个属性。对于Last.fm数据集，本文将所有存在交互行为的样本作为正样本，标签为1；将随机采样无交互行为的样本作为负样本，标签为0。

● Yelp数据集：包含1 368 606条交互数据，这些数据没有评分信息。本文利用数据集中的物品属性信息构建知识图谱，共有590个属性。对于Yelp数据集，本文将所有存在交互行为的样本作为正样本，标签为1；将随机采样无交互行为的样本作为负样本，标签为0。

数据集统计信息见**表2**。

4.2 在线环境模拟

由于交互式推荐问题的交互性，笔者希望推荐系统可以进行在线学习，因此构建了一个模拟器用来模拟在线环境。

本文利用矩阵分解[16]算法训练用户和物品的表示。对于MovieLens-1m数据集，本文将评分归一化到[-1,1]区间内，然后将这个评分作为推荐系统的奖励；对于Last.fm和Yelp数据集，本文将正样本评分置为1、负样本评分置为-1进行训练，然后将这个评分作为推荐系统的奖励。最后，环境模拟器的输出为[-1,1]区间的一个评分。

表 2 数据集统计

对比项		MovieLens-1m	Last.fm	Yelp
交互数据	用户数/个	5 417	1 801	27 675
	物品数/个	3 650	7 432	70 311
	属性数/个	15	33	590
	数据数/条	986 495	76 693	1 368 606
知识图谱	实体数/个	99 060	9 266	98 576
	三元组数/个	207 939	138 217	2 533 827

对于每一个数据集，本文将每个用户的行为取前80%的数据作为训练集，后20%作为测试集。

4.3 评价指标

由于交互式推荐的目标是最大化奖励，本文直接将奖励作为一个评价指标。奖励可以表示用户对推荐结果的满意度。

$$\text{Reward}=\frac{1}{\#\text{users}\times T}\sum_{\text{users}}\sum_{t=1}^{T}\gamma^{t}R(\boldsymbol{s}_t,\boldsymbol{i}_t) \quad (15)$$

其中，T为一次会话内的推荐次数，γ为强化学习中奖励的折扣因子，$R(\boldsymbol{s}_t,\boldsymbol{i}_t)$为当前状态下推荐$\boldsymbol{i}_t$的即时奖励，#users为用户的数量。

同时，本文也将准确率和召回率作为评价指标。

准确率为：

$$\text{Precision@}T=\frac{1}{\#\text{users}\times T}\sum_{\text{users}}\sum_{t=1}^{T}\theta_{\text{hit}} \quad (16)$$

召回率为：

$$\text{Recall@}T=\frac{1}{\#\text{users}}\sum_{\text{users}}\sum_{t=1}^{T}\frac{\theta_{\text{hit}}}{\#\text{preferences}} \quad (17)$$

其中，#preferences表示用户交互数据中所有环境反馈为正向的物品总数。T取值35，后文中的准确率为Precision@35，召回率为Recall@35。

对于推荐物品，如果用户的评分大于一定阈值，则$\theta_{\text{hit}}=1$；反之，$\theta_{\text{hit}}=0$。对于MovieLens-1m数据集、Last.fm数据集和Yelp数据集，本文将这个阈值设置为0。

4.4 对比方法

本文将KGP-DQN模型与4种代表性算法进行对比，其中协同过滤是非常有代表性的传统推荐算法，GRU4REC（gated

recurrent unit for recommendations)是经典的序列推荐算法，DQNR(deep Q-net recommendations)是非常有代表性的基于强化学习的推荐算法，KGQN是非常有代表性的将知识图谱和强化学习结合的推荐算法。

- 协同过滤[6]算法是一种经典的基于用户相似度的推荐算法。在交互式推荐场景中，本文利用用户交互过的物品进行用户相似度对比，选择相似度最高用户的评分最高物品作为被推荐的物品。
- GRU4REC[8]是一种经典的基于循环神经网络的推荐算法，常被用于解决序列推荐问题。该算法利用用户的交互历史，考虑历史中的顺序关系，通过GRU神经网络预测用户可能喜欢的下一个物品。
- DQNR[11]是一种基于DQN的推荐算法。它利用神经网络构建Q函数，在给定状态下通过该函数预测每个物品的价值。最后选择价值最高的物品作为推荐结果。
- KGQN[4]是一种将知识图谱与DQN推荐结合的算法。该算法利用知识图谱解决交互式推荐系统中的反馈稀疏问题。它利用知识图谱学习物品的表示，通过GRU进一步学习用户序列特征，最后设计Q值函数，从而推荐Q值最高的物品。
- KGP-DQN模型是本文提出的知识增强策略引导的交互式强化推荐模型。

4.5 参数设置

在KGP-DQN中，本文将GCN的层数设置为2，所有模型输出的表示维度都设置为64。将GCN之后的全连接网络设置为两层，采用ReLU激活函数。将初始化策略与强化策略结合的参数α设置为0.5，动态聚类的类别数n设为15，本文将在第4.9节对这两个参数的选择进行具体的说明。其他参数采取随机初始化的方法。所有参数在训练时都采用Adam优化器。本文使用PyTorch实现模型，并在一块NVIDIA GTX 1080Ti GPU上训练网络。

4.6 总体性能对比

所有模型的性能对比结果见**表3**。本文得出了如下结论。

- 对比方法具有明显提升。这证明了利用历史交互数据为强化推荐进行学习指导的有效性。
- DQNR、KGQN、KGP-DQN等强化方法相对其他方法在奖励上有明显提高，表明强化推荐方法更能提升用户的体验。KGP-DQN获得的奖励最高，表明本文方法对提升用户体验具有明显的效果。
- 与协同过滤算法和GRU4REC算法相比，KGP-DQN模型在奖励上有显著提高，这说明采用深度强化学习建模交互式推荐系统对提升用户体验具有显著效果。
- 与DQNR算法相比，KGP-DQN模型在奖励和准确率上都获得了提升，这说明将知识图谱引入强化推荐系统能有效提升推荐系统的性能。
- 与KGQN算法相比，KGP-DQN模型在奖励和准确率上都获得了提升，这说明在引入知识图谱的基础上充分利用用户的历史数据可以显著提升推荐系统的性能。
- 对比MovieLens-1m和Last.fm数据集，在MovieLens-1m数据集下，KGP-DQN模型的性能提升较高，这是因为MovieLens-1m数据集的交互矩阵更稠密，使得策略初始化模块可以更好地根据历史信息学习用户的偏好。

4.7 模型效率对比

本文的一个出发点是解决强化推荐

表 3 所有模型的性能对比

对比项	MovieLens-1m			Last.fm			Yelp		
	奖励	准确率	召回率	奖励	准确率	召回率	奖励	准确率	召回率
协同过滤	-0.07	51%	1.31%	0.05	62%	0.9%	-0.15	49%	0.2%
GRU4REC	0.15	68%	1.74%	0.20	72%	1.1%	0.22	61%	0.3%
DQNR	0.26	75%	1.92%	0.36	78%	1.2%	0.26	63%	0.3%
KGQN	0.37	77%	1.98%	0.45	84%	1.2%	0.40	82%	0.4%
KGP-DQN	0.47	82%	2.11%	0.53	87%	1.3%	0.47	91%	0.4%

从零开始学习带来的用户体验降低问题，并提升强化推荐的采样效率。KGP-DQN模型、DQNR、KGQN算法在MovieLens-1m数据集、Last.fm数据集和Yelp数据集上的奖励随迭代次数的变化曲线如**图5**所示，其中横轴表示训练的步数，纵轴表示当前训练步下模型得到的奖励值。从**图5**可以看出：知识图谱的引入有助于提升模型训练的速度。由于加入了策略初始化模块，KGP-DQN模型在强化训练初期奖励值较高，且会在较短的训练步长下达到最优。

4.8 不同模块的影响

本节分析了模型的不同模块对模型总体性能的影响。在KGP-DQN模型中，本文提出了3个重要的模块：行为知识图谱表示模块、策略初始化模块、候选集筛选模块。消融实验结果见**表4**，其中，KGP-DQN-kg表示去掉行为知识图谱表示模块，KGP-DQN-pg表示去掉行为策略初始化模块，KGP-DQN-cs表示去掉候选集筛选模块。(-%)表示各方法相对KGP-DQN的性能下降率。

由**表4**可以得出以下结论。

- 行为知识图谱表示模块对整体结果影响较大。这是因为该模块引入了行为知识图谱并利用GCN和GRU神经网络对用户表示进行建模。GCN考虑了物品之间的结构关系，将用户对物品的反馈传播到相邻的物品上，有效解决了反馈稀疏问题；GCN则充分考虑了物品之间的序列关系，二者结合得到了更好的状态表示。
- 策略初始化模块对整体性能也有重要的影响。这是因为该模块一方面为推荐系统的学习提供先验方向，解决了强化推荐从零学习的问题，并指导推荐系统更好地学习；另一方面利用了用户的结构信息，

表 4 消融实验结果

对比项	MovieLens-1m		Last.fm		Yelp	
	奖励	准确率	奖励	准确率	奖励	准确率
KGP-DQN-kg	0.26 (-45%)	75% (-9%)	0.36 (-32%)	78% (-10%)	0.26 (-44%)	63% (-31%)
KGP-DQN-pg	0.37 (-21%)	78% (-5%)	0.45 (-15%)	84% (-3%)	0.39 (-17%)	85% (-7%)
KGP-DQN-cs	0.46 (-2%)	81% (-1%)	0.50 (-6%)	85% (-2%)	0.47 (-0%)	89% (-2%)
KGP-DQN	0.47	82%	0.53	87%	0.47	91%

使得推荐系统可以更全面地对用户特征进行建模。

● 候选集筛选模块对推荐性能的提升也有一定的帮助。通过缩小候选集解决动作空间大的问题，一方面使候选物品中的噪声有所减少，从而帮助模型更好地学习；另一方面，对相似样本进行区分，更有利于模型学习。

4.9 参数分析

本节分析KGP-DQN模型的不同超参数对模型的影响，包括将初始化策略与强化策略结合的参数α对推荐准确率的影响，以及候选集筛选模块中的动态聚类类别数n对推荐准确率的影响。最终的结果如**图6**和**图7**所示。

从**图6**和**图7**可以看出，对于结合参数α，当$\alpha=0.5$时，效果最好；对于动态聚类类别数n，当$n=15$时，效果最好。因此，本文选择$\alpha=0.5$和$n=15$作为KGP-DQN模型的超参数，并对这两个超参数的影响进行进一步分析。

● 对动态聚类类别数的影响：类别少时，候选集中噪声较多，不利于模型进行学习；类别多时，候选集中无法包含用户的喜好物品的概率更大。

● 对结合参数的影响：当参数更偏向强化输出Q值时，策略初始化模块无法为强化推荐网络提供较强的初始化策略，导致性能受到影响；当参数更偏向策略初始化模块输出的价值时，由于缺少强化推荐网络可以建模长期奖励的优点，模型的性能会受到影响。

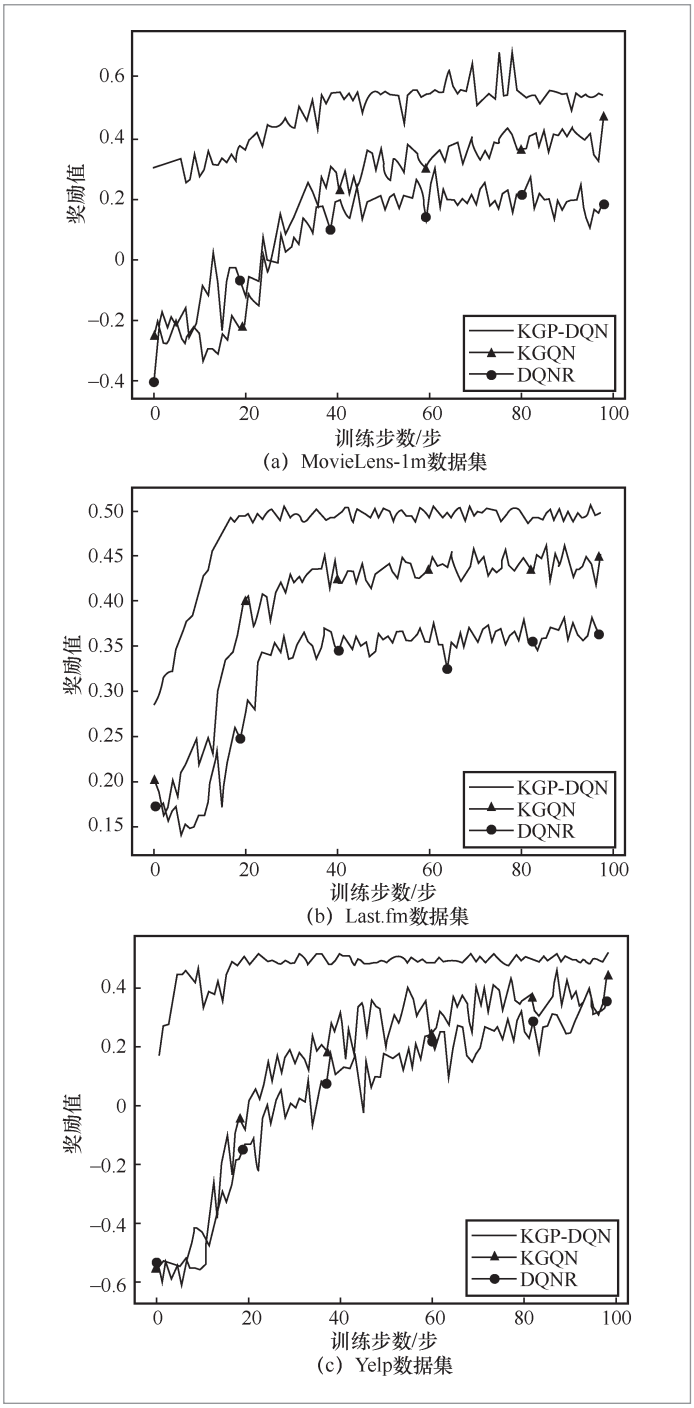


图 5 模型效率对比

4.10 动态聚类分析

本节分析KGP-DQN模型每推荐τ次更新聚类结果对模型的影响。下面分别从定量和定性的角度进行分析。

本实验利用DBI（Davies-Bouldin）指数[17]评价动态聚类效果。由**表5**可以看出，随着推荐过程的进行，候选集筛选模块动

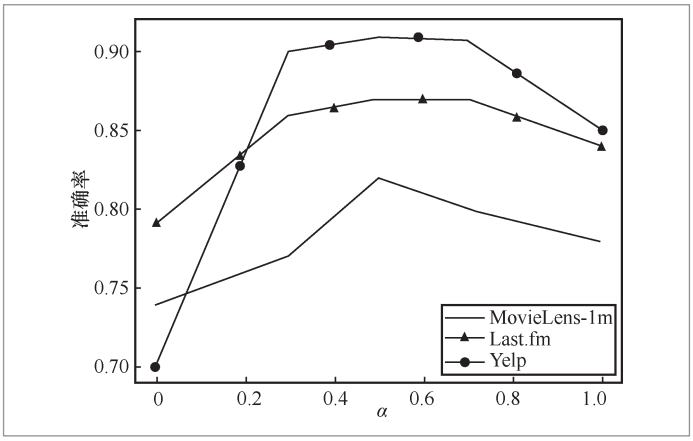


图 6 结合参数的影响

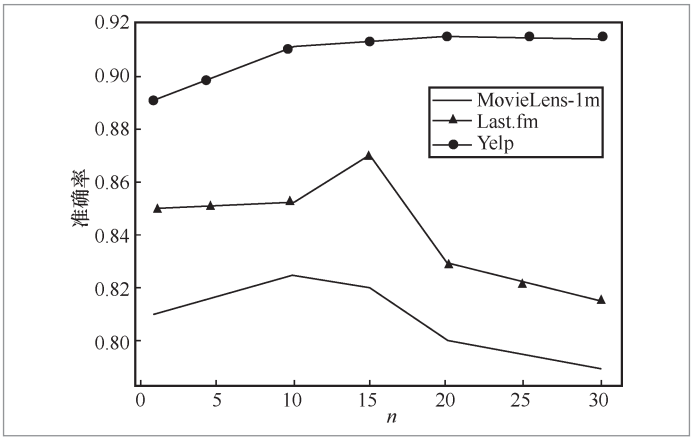


图 7 聚类类别数的影响

表 5 动态聚类的 DBI 指数

推荐次数/次	MovieLens-1m	Last.fm	Yelp
0	2.71	3.74	4.97
100	2.43	3.62	4.91
200	2.26	3.56	4.60
300	2.15	3.54	4.05

态聚类的效果会逐步提升。

聚类后的类别分布可视化情况如图8所示。从图8可以看出，随着训练的进行，原来分布边界不明确的类别的分布边界会变得更明确，如MovieLens-1m中的红色类、Last.fm的橙色类以及Yelp中的紫色类。

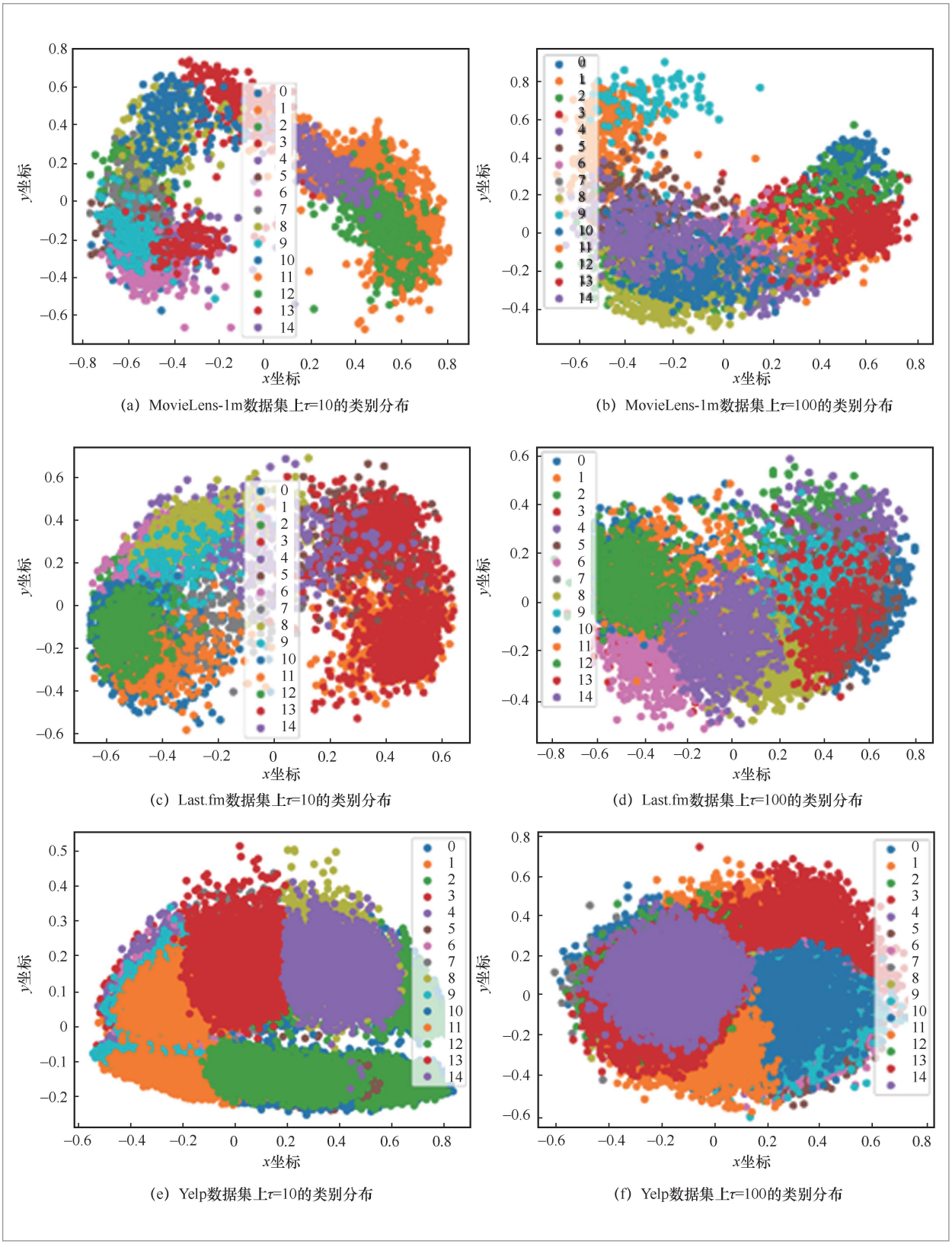
5 结束语

本文提出一种知识增强策略引导的交互式强化推荐模型KGP-DQN。该模型构建行为图谱表示模块，该模块构建行为知识图谱，利用GCN和GRU对用户表示建模，从而解决稀疏反馈的问题；构建策略初始化模块，该模块利用行为知识图谱上的节点表示，从用户行为历史中学习初始化价值，从而解决从零学习影响用户体验的问题；构建候选集筛选模块，该模块利用物品的节点表示进行动态聚类以生成物品数量较少的候选集，从而解决动作空间大的问题。实验结果验证了本文提出的模型的有效性，即KGP-DQN在提升训练速度的同时也保证了推荐准确率。相比现有工作在推荐前期采取基于流行度的推荐方法，本文提出的KGP-DQN模型在推荐前期仍可以个性化地为用户进行推荐，从而提升用户的体验。在后续的工作中，本文考虑利用迁移强化学习中的策略重使用等方法结合策略初始化模块和Q网络推荐模块，并尝试使用更先进的聚类方法构建候选集筛选模块。

参考文献:

[1] WANG H Z, WU Q Y, WANG H N. Factorization bandits for interactive recommendation[C]//Proceedings of the 31st AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2017: 2695-2702.

[2] MAHMOOD T, RICCI F. Learning and adaptivity in interactive recommender systems[C]//Proceedings of the 9th



（a）MovieLens-1m数据集上τ=10的类别分布

（b）MovieLens-1m数据集上τ=100的类别分布

（c）Last.fm数据集上τ=10的类别分布

（d）Last.fm数据集上τ=100的类别分布

（e）Yelp数据集上τ=10的类别分布

（f）Yelp数据集上τ=100的类别分布

图 8　聚类结果变化

International Conference on Electronic Commerce. New York: ACM Press, 2007: 75–84.

[3] ZHAO X Y, ZHANG L, DING Z Y, 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. New York: ACM Press, 2018: 1040–1048.

[4] ZHOU S J, DAI X Y, CHEN H K, et al. Interactive recommender system via knowledge graph–enhanced reinforcement learning[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2020: 179–188.

[5] WANG P F, FAN Y, XIA L, et al. KERL: a knowledge–guided reinforcement learning model for sequential recommendation[C]// Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2020: 209–218.

[6] GOLDBERG D, NICHOLS D, OKI B M, et al. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12): 61–70.

[7] RENDLE S. Factorization machines[C]// Proceedings of 2010 IEEE International Conference on Data Mining. Piscataway: IEEE Press, 2010: 995–1000.

[8] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session–based recommendations with recurrent neural networks[J]. arXiv preprint, 2015, arXiv:1511.06939.

[9] ZHOU G R, ZHU X Q, SONG C R, et al. Deep interest network for click–through rate prediction[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2018: 1059–1068.

[10] CHEN H K, DAI X Y, CAI H, et al. Large–scale interactive recommendation with tree–structured policy gradient[C]// Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2019: 3312–3320.

[11] ZHENG G J, ZHANG F Z, ZHENG Z H, et al. DRN: a deep reinforcement learning framework for news recommendation[C]// Proceedings of the 2018 World Wide Web Conference. [S.l.:s.n.], 2018: 167–176.

[12] ZOU L X, XIA L, DING Z Y, et al. Reinforcement learning to optimize long–term user engagement in recommender systems[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2019: 2810–2818.

[13] DULAC–ARNOLD G, EVANS R, SUNEHAG P, et al. Reinforcement learning in large discrete action spaces[J]. arXiv preprint, 2015, arXiv:1512.07679.

[14] XIAN Y K, FU Z H, MUTHUKRISHNAN S, et al. Reinforcement knowledge graph reasoning for explainable recommendation[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2019: 285–294.

[15] HASSELT H V, GUEZ A, SILVER D. Deep reinforcement learning with double Q–learning[C]//Proceedings of the 30th AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2016.

[16] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30–37.

[17] DAVIES D L, BOULDIN D W. A cluster separation measure[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1979, PAMI–1(2): 224–227.

作者简介

张宇奇（1997– ），男，北京交通大学计算机与信息技术学院硕士生，主要研究方向为强化学习、推荐系统等。

黄晓雯（1993– ），女，博士，北京交通大学计算机与信息技术学院讲师，主要研究方向为多媒体计算、数据挖掘、用户建模、推荐系统等，在国内外学术会议/期刊上发表学术论文10余篇。

桑基韬（1985– ），男，博士，北京交通大学计算机与信息技术学院教授。2017年入选北京交通大学“卓越百人”计划。曾获中国电子学会科学技术奖自然科学一等奖、北京市科学技术奖、中国科学院院长特别奖、ACM中国新星奖等。主要研究方向为社会多媒体计算、多源数据挖掘、可信赖机器学习等。作为负责人先后主持国家自然科学基金重点项目、国家重点研发计划课题、北京市杰出青年科学基金等多个项目。

收稿日期： 2021-11-03
通信作者： 黄晓雯，xwhuang@bjtu.edu.cn
基金项目： 中央高校基本科研专项资金资助项目（No.2021RC217）
Foundation Item: The Fundamental Research Funds for the Central Universities (No.2021RC217)