



## 一种同伴知识互增强下的序列推荐方法

胡开喜 李琳 吴小华 解庆 袁景凌

### A Sequential Recommendation Method Enhanced by Peer Knowledge

HU Kai-Xi, LI Lin, WU Xiao-Hua, XIE Qing, YUAN Jing-Ling

在线阅读 View online: <https://doi.org/10.16383/j.aas.c220347>

---

## 您可能感兴趣的其他文章

### 应用知识图谱的推荐方法与系统

Recommendation Methods and Systems Using Knowledge Graph

自动化学报. 2021, 47(9): 2061–2077 <https://doi.org/10.16383/j.aas.c200128>

### 融合混合知识与MCTS的针灸排序方案设定方法

Acupuncture Sequential Scheming Method With Hybrid Knowledge and MCTS

自动化学报. 2020, 46(6): 1240–1254 <https://doi.org/10.16383/j.aas.c180120>

### 基于注意力机制的协同卷积动态推荐网络

Attention-based Collaborative Convolutional Dynamic Network for Recommendation

自动化学报. 2021, 47(10): 2438–2448 <https://doi.org/10.16383/j.aas.c190820>

### 复杂网络环境下基于信任传递的推荐模型研究

A Recommendation Model by Means of Trust Transition in Complex Network Environment

自动化学报. 2018, 44(2): 363–376 <https://doi.org/10.16383/j.aas.2018.c160395>

### 融合社交因素和评论文本卷积网络模型的汽车推荐研究

Social and Comment Text CNN Model Based Automobile Recommendation

自动化学报. 2019, 45(3): 518–529 <https://doi.org/10.16383/j.aas.2018.c170245>

### 基于知识的污水生化处理过程智能优化方法

Knowledge-based Intelligent Optimal Control for Wastewater Biochemical Treatment Process

自动化学报. 2017, 43(6): 1038–1046 <https://doi.org/10.16383/j.aas.2017.c170088>

# 一种同伴知识互增强下的序列推荐方法

胡开喜<sup>1</sup> 李琳<sup>1</sup> 吴小华<sup>1,2</sup> 解庆<sup>1</sup> 袁景凌<sup>1</sup>

**摘要** 序列推荐 (Sequential recommendation, SR) 旨在建模用户序列中的动态兴趣, 预测下一个行为. 现有基于知识蒸馏 (Knowledge distillation, KD) 的多模型集成方法通常将教师模型预测的概率分布作为学生模型样本学习的软标签, 不利于关注低置信度序列样本中的动态兴趣. 为此, 提出一种同伴知识互增强下的序列推荐方法 (Sequential recommendation enhanced by peer knowledge, PeerRec), 使多个具有差异的同伴网络按照人类由易到难的认知过程进行两阶段的互相学习. 在第 1 阶段知识蒸馏的基础上, 第 2 阶段的刻意训练通过动态最小组策略协调多个同伴从低置信度样本中挖掘出可被加强训练的潜在样本. 然后, 受训的网络利用同伴对潜在样本预测的概率分布调节自身对该样本学习的权重, 从解空间中探索更优的兴趣表示. 3 个公开数据集上的实验结果表明, 提出的 PeerRec 方法相比于最新的基线方法在基于 Top-*k* 的指标上不仅获得了更佳的推荐精度, 且具有良好的在线推荐效率.

**关键词** 序列推荐, 动态兴趣, 知识蒸馏, 刻意训练

**引用格式** 胡开喜, 李琳, 吴小华, 解庆, 袁景凌. 一种同伴知识互增强下的序列推荐方法. 自动化学报, 2023, 49(7): 1456–1470

**DOI** 10.16383/j.aas.c220347

## A Sequential Recommendation Method Enhanced by Peer Knowledge

HU Kai-Xi<sup>1</sup> LI Lin<sup>1</sup> WU Xiao-Hua<sup>1,2</sup> XIE Qing<sup>1</sup> YUAN Jing-Ling<sup>1</sup>

**Abstract** Sequential recommendation (SR) aims to model dynamic interests within users' interaction sequences and predict the next behaviour. Current knowledge distillation (KD)-based multi-model ensemble methods generally adopt teacher's probability distributions as soft labels to guide the sample learning of its students, posing a significant challenge to focus on the samples with less confidence. To this end, we propose a novel sequential recommendation method enhanced by peer knowledge (PeerRec) that makes multiple sequential prediction networks (peer for each other) with distinct knowledge conduct two-stage mutual learning by following the easy to hard human cognition. Based on the knowledge distillation in the first stage, the deliberate practice in the second stage mines the potential samples from the low confidence samples by combining multiple peers through a dynamic minimum group strategy. The training network can further employ the probability distributions from peers to adjust itself learning weight where better representations of dynamic interests can be derived. Extensive experiments on three public recommendation datasets show that the proposed PeerRec not only outperforms the state-of-the-art methods, but also achieves good efficiency in online recommendation.

**Key words** Sequential recommendation (SR), dynamic interest, knowledge distillation (KD), deliberate practice

**Citation** Hu Kai-Xi, Li Lin, Wu Xiao-Hua, Xie Qing, Yuan Jing-Ling. A sequential recommendation method enhanced by peer knowledge. *Acta Automatica Sinica*, 2023, 49(7): 1456–1470

收稿日期 2022-04-28 录用日期 2022-09-13

Manuscript received April 28, 2022; accepted September 13, 2022

国家自然科学基金 (62276196, 61602353), 湖北省重点研发设计项目 (2021BAA030), 国家留学基金委基金 (202106950041, 留金美 [2020] 1509), 安康市科学技术研究发展计划 (AK2020-GY-08) 资助

Supported by National Natural Science Foundation of China (62276196, 61602353), Key Research and Development Program of Hubei Province (2021BAA030), Foundation of China Scholarship Council (202106950041, LiuJinMei [2020] 1509), and Ankang Municipal Science and Technology Bureau (AK2020-GY-08)

本文责任编辑 张敏灵

Recommended by Associate Editor ZHANG Min-Ling

1. 武汉理工大学计算机与人工智能学院 武汉 430070 2. 安康学院电子与信息工程学院 安康 725000

1. School of Computer Science and Artificial Intelligence, Wuhan University of Technology, Wuhan 430070 2. School of Electronic and Information Engineering, Ankang University, Ankang 725000

随着互联网、5G 等信息技术的快速发展, 信息系统记录了大量的用户行为, 例如购物<sup>[1-2]</sup>、点击浏览<sup>[3-5]</sup>、移动位置<sup>[6-7]</sup>等. 人们将用户行为按发生的时间先后顺序排列, 可构成隐含用户兴趣的行为序列<sup>[8-10]</sup>. 序列推荐 (Sequential recommendation, SR) 旨在建模行为序列中动态兴趣的演变关系, 从候选集中预测下一个行为, 已广泛用于各类 Web 应用, 帮助人们解决信息过载的问题. 作为一项基础技术, 序列推荐备受学术界和工业界关注, 并逐渐推广到更多应用场景.

近年来, 深度学习通过层次化地建模序列数据中行为演变的特征表示, 在捕获用户兴趣方面表现出了优越性<sup>[1, 3, 5, 10-11]</sup>. 然而, 现实世界中用户与行为

的交互是在该用户当时兴趣的影响下产生的, 具有动态不确定的特点<sup>[12-15]</sup>. 以图 1 中的电影推荐为例, 假设用户按时间顺序观看了不同类型的电影 (喜剧片 m1, 恐怖片 m2, 爱情片 m3), 则历史的行为序列反映出了多种用户兴趣, 导致下一个行为的产生存在多种可能性 (恐怖片 m4 或喜剧片 m5). 在学习兴趣演变关系的过程中, 采用固定向量表示的单模型通常不足以建模行为背后隐含的动态兴趣, 使模型的精度提升有限. 此外, 用户观影行为较少时, 通过行为序列分析的用户兴趣也会更加不确定. 在行为序列提供信息有限的情况下, 如何通过建模用户动态兴趣缓解模型不确定性<sup>[16]</sup>, 是一个具有挑战性的问题.

目前, 大多数序列推荐方法<sup>[1, 9, 17]</sup> 通过建模行为序列在特征空间中的固定向量表示, 并引入额外的用户信息来帮助预测模型学习用户兴趣. 在不引入额外信息的情况下, 最近的研究工作<sup>[13-15]</sup> 利用分布函数表示用户兴趣, 使学习到的兴趣表示可以覆盖更大的行为序列空间. 如图 1(a) 所示, 每个观影行为隐含的动态兴趣都可以用一个分布函数来表示, 最后通过积分计算推断的兴趣分布表示与观影行为

向量表示 (三角形) 之间的距离预测下一个行为. 但是在模型参数中引入分布函数, 主要存在以下不足: 1) 通常需要预设一种先验的分布形式, 但只有少数几种分布可以求出积分的闭式解, 对更复杂的分布需要近似求解<sup>[15]</sup>; 2) 分布函数倾向于在有限的函数空间 (兴趣表示空间的实线上) 探索用户兴趣, 对模型精度的提升有限<sup>[18-19]</sup>.

针对上述分布函数的不足, 计算机视觉领域的工作<sup>[20-22]</sup> 通常采用多模型集成的方式建模图像中的不确定性, 并验证了随机初始化和扰乱数据样本的顺序就足以获得较好的性能<sup>[20]</sup>. 如图 1(b) 所示, 采用多模型集成建模动态兴趣时, 模型可以在不同的训练迭代中探索分布函数 (虚线) 以外的点, 并通过融合不同的模型推断更优的固定向量来表示用户兴趣. 同时, 为了提升模型在测试阶段的效率, 计算机视觉<sup>[22]</sup>、自然语言处理<sup>[23-24]</sup> 和推荐<sup>[25-26]</sup> 等领域的工作都表明知识蒸馏 (Knowledge distillation, KD)<sup>[27]</sup> 是一种有效的融合方式, 可以通过拉近两个模型兴趣表示的距离, 将多个教师模型的知识迁移到学生模型. 但是, 传统的知识蒸馏将教师模型预测的概率分布作为学生模型学习的软标签<sup>[28]</sup>, 导致学生模

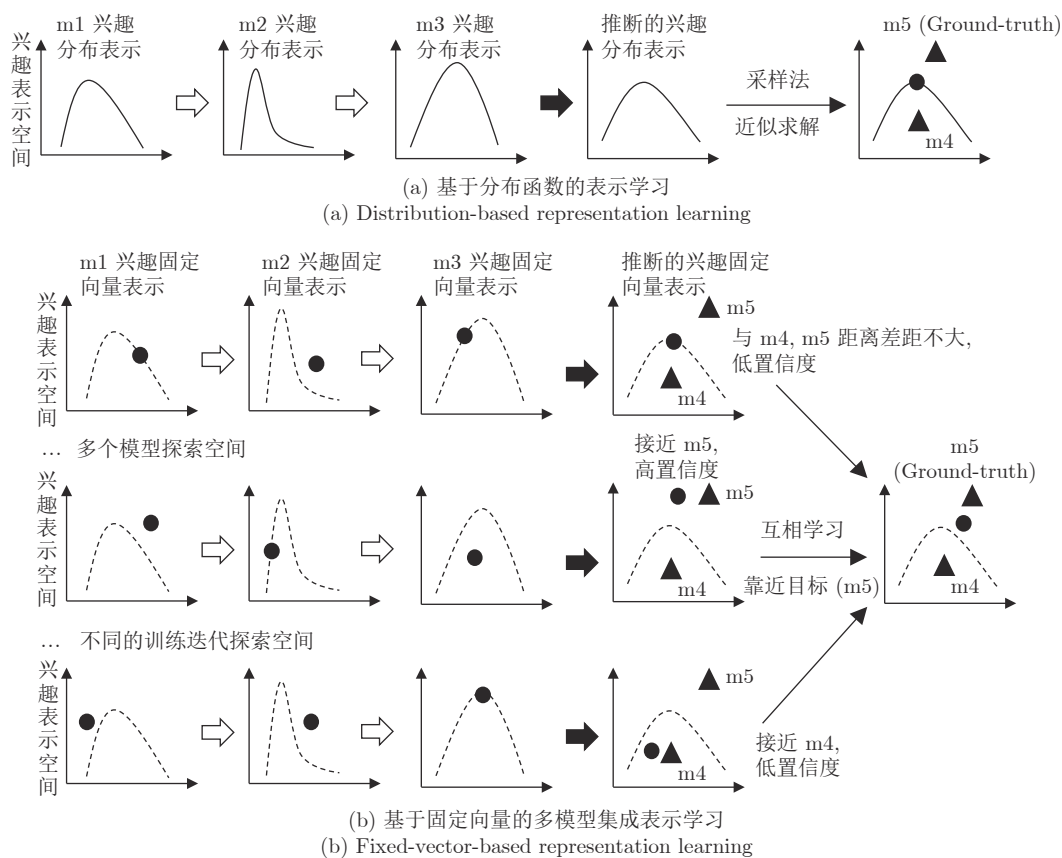


图 1 用户动态兴趣在潜在空间中的表示与推断

Fig. 1 The representation and inference of dynamic interests in latent representation spaces

型更倾向于从教师模型输出的高置信度样本中获取知识<sup>[29-30]</sup>. 由于用户的行为序列具有动态不确定的特点<sup>[16]</sup>, 在训练过程中如果教师模型学习的用户兴趣表示距离真实发生的下一个行为 (Ground-truth) 较远就会使模型产生低置信度, 而低置信度的样本中通常也包含可被加强训练的样本 (潜在样本), 可以提供更加丰富的信息, 帮助模型学习分类决策边界即类间信息<sup>[31]</sup>. 传统的知识蒸馏未能有效地从这些潜在样本中学习动态兴趣, 影响了推荐的精度.

为了缓解上述问题, 本文提出一种同伴知识互增强下的序列推荐方法 (Sequential recommendation enhanced by peer knowledge, PeerRec). 该方法首先设计了多个结构相同但随机初始化参数不同的同伴网络, 使每个同伴网络学习差异化的兴趣固定向量表示. 其次, 按照课程学习<sup>[32]</sup> 由易到难的认知过程, 本文通过第 1 阶段的知识蒸馏<sup>[27]</sup> 关注同伴输出的高置信度样本, 以模仿训练的方式拉近兴趣表示的距离. 然后, 在此基础上提出了第 2 阶段基于刻意训练<sup>[33]</sup> 的互相学习, 通过动态最小组策略将多个同伴组成学习共同体, 从低置信度样本中挖掘过滤噪音后的潜在样本, 推动模型在解空间中探索更优的兴趣表示. 相比于最好的基线方法, 本文提出的方法在推荐性能和在线推断效率上均具有优势. 总体而言, 本文的主要贡献如下:

1) 分析并总结了序列推荐中用户动态兴趣导致行为序列具有不确定的特点, 提出了一种同伴知识互增强下的序列推荐方法, 利用多个同伴建模差异化的用户兴趣固定向量表示, 并按照人类由易到难的认知过程互相学习, 共同提升序列推荐的精度;

2) 提出了一种基于刻意训练<sup>[33]</sup> 的互相学习方法, 在知识蒸馏的基础上进一步通过动态最小组策略增强模型对潜在样本中动态兴趣的学习;

3) 公开的 ML-1m、LastFM 和 Toys 数据集上的实验结果表明, 提出的 PeerRec 方法相对于最新的序列推荐方法在 NDCG@5 指标上分别提升了 9.02%、5.03% 和 1.31%, 且具有良好的推荐效率.

本文内容安排如下: 第 1 节为现有序列推荐方法以及以知识蒸馏为代表的知识迁移方法的概述; 第 2 节介绍问题定义、总体框架及 PeerRec 模型的网络结构; 第 3 节重点阐述同伴知识互增强的训练, 并进一步讨论模型在互相学习过程中的梯度变化; 第 4 节介绍数据集、实验设置、评价指标及分析结果; 第 5 节总结全文工作并展望未来的研究方向.

## 1 相关工作

本节首先介绍目前序列推荐中建模用户动态兴趣的方法, 然后阐述以知识蒸馏为代表的知识迁移

方法以及该类方法与提出的基于刻意训练<sup>[33]</sup> 的互相学习方式的异同.

### 1.1 序列推荐

在现有工作中, 动态兴趣的建模主要分为基于固定向量表示的方法和基于分布函数的方法.

基于固定向量表示的方法主要通过引入外部的信息来增强单模型学习兴趣的固定向量表示. 例如, Song 等<sup>[34]</sup> 对点击率预估中用户行为和额外的画像特征进行不同阶次的组合. Wang 等<sup>[9]</sup> 利用超图卷积建模不同时间段内用户与物品的动态关联, 并利用多头自注意力融合不同时刻的用户兴趣. STKG (Spatial-temporal knowledge graph) 模型<sup>[6]</sup> 通过构建一个带权重的兴趣点连接图和辅助的空间信息来优化节点的表示. BERT4Rec<sup>[17]</sup> 虽然没有引入额外信息, 但增强了对双向序列信息的建模. 此外, 模型去噪的方法<sup>[11, 35]</sup> 通过过滤动态性较高的噪音用户, 使模型捕获更加普遍的用户兴趣, 但也会丢失一部分有效的信息. 例如 Wang 等<sup>[11]</sup> 利用模型自身的响应来剔除损失较大的噪声样本. Qin 等<sup>[35]</sup> 通过对比用户偏好的估计, 设计了一种去噪生成器.

近年来, 基于分布函数的方法<sup>[12-15]</sup> 在不引入额外信息的情况下建模动态兴趣, 引起了研究人员的广泛关注. Jiang 等<sup>[15]</sup> 在模型输入的嵌入矩阵中引入高斯分布  $N(\mu, \Sigma)$ , 扩大了物品和用户的表示空间. Neupane 等<sup>[12]</sup> 结合元证据学习建模输出分布的参数. 最近, Fan 等<sup>[13-14]</sup> 在嵌入矩阵引入高斯分布的基础上, 进一步采用 Wasserstein 距离<sup>1</sup> 计算自注意力层和预测层中分布的距离.

上述基于单模型固定向量表示的方法假设动态兴趣是确定的, 并通过信息引入、去噪等方式优化向量表示. 基于分布函数的方法利用先验分布建模了动态兴趣, 但实践中对预设分布的近似求解复杂, 性能提升效果并不如模型集成的方法显著<sup>[18-19]</sup>. 受分布函数和知识蒸馏的启发, 本文采用多个同伴网络以互相学习的方式共同探索更佳的兴趣固定向量表示, 提升序列推荐的精度.

### 1.2 知识蒸馏

知识蒸馏起源于模型压缩<sup>[27, 36-37]</sup> 任务, 主要基于学生向教师模仿的策略, 在训练阶段将从多个模型提炼的知识迁移到单模型, 提升测试阶段的效率. 假设  $p_t$  和  $p_s$  为教师模型和学生模型在一个样本上预测的概率分布, 则教师-学生单向模仿的梯度为

<sup>1</sup> 一种基于最优传输理论衡量两个分布间距离的度量方式, 目前只在一维分布、高斯分布等少数几种分布上存在闭式解.



$$\nabla \mathcal{L} = (p_{s,*} - y_* p_{t,*}) + \sum_{i=1}^{C-1} (p_{s,i} - p_{t,i}) \quad (1)$$

其中,  $C$  为类别的数量,  $p_{t,*}$  和  $p_{s,*}$  分别代表教师和学生正例输出的概率, 即置信度,  $p_{t,i}$  和  $p_{s,i}$  分别代表教师和学生第  $i$  个负例输出的概率,  $y_*$  是正例样本的标签,  $*$  和  $i$  分别为概率分布中正例和负例的索引.

现有工作表明, 式 (1) 的第 1 项和第 2 项在模型训练中具有不同的作用<sup>[29, 38]</sup>. 从第 1 项中可以看出, 教师模型对正例输出的置信度  $p_{t,*}$  与学生模型对样本学习的重要性权重有较大的关联, 反映了样本学习的难度. 如果教师模型输出的置信度较大 ( $p_{t,*} \approx 1$ ), 则学生模型相当于从硬标签  $y_*$  中学习, 样本的梯度贡献较大. 反之, 教师模型输出的置信度较小, 则会在当前训练批次的全局训练信号上减少该样本的贡献. 第 2 项则更多地反映了样本内部负例与正例的相似性信息. 因此, 在模仿过程中, 学生模型会更倾向于从教师模型输出的高置信度样本中获取知识<sup>[29-30]</sup>.

后续的工作通过融合多个模型缓解建模中的不确定性, 并从学习方式和教师选择上优化了融合效果. 文献 [39-40] 优化了离线学习方式, 通过两个结构相同但参数不同的深度网络互相学习扩大参数的搜索空间, 减少模型因初始化导致的不确定性. 同一网络在不同的训练阶段上自蒸馏<sup>[29]</sup> 也是一种特殊的互相学习方式. 此外, 通过增加教师模型的数量来提升学生模型是一种直观的方式. 其中, 平均多个不同来源模型的输出, 是一种最直观的融合方式<sup>[39]</sup>, 但会损失一部分多样性. 随后, Chen 等<sup>[41]</sup> 引入了辅助和领导的概念来引导互相学习. Zhu 等<sup>[25]</sup> 引入了门控机制动态赋予教师不同的权重. Kang 等<sup>[26]</sup> 认为注意力加权的方式会引入弱关联信息, 因此, 每次只从多个教师中选择一个专家传递知识.

但知识蒸馏倾向于从教师模型输出的高置信度样本中获取知识, 未能有效发掘可进一步学习的低置信度样本. 不同于模仿策略, 刻意训练<sup>[33]</sup> 从另一个角度阐述了教师-学生传递知识的方式, 认为教师针对性地给学生训练有待提高 (低置信度) 的样本, 可以进一步提升学生模型. 因此, 按照人类课程学习<sup>[32]</sup> 由易到难的认知过程, 本文在知识蒸馏的基础上进一步提出了第 2 阶段基于刻意训练的互相学习, 加强模型从低置信度的潜在样本中学习动态兴趣.

## 2 PeerRec 模型

### 2.1 问题定义

序列推荐主要包含用户和按时间顺序排列的

行为序列. 本文定义如下概念: 给定一组用户集合  $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$  和行为集合  $\mathcal{B} = \{b_1, b_2, \dots, b_{|\mathcal{B}|}\}$ . 对于用户集合中的任一用户  $u \in \mathcal{U}$ , 其产生的行为序列可以定义为  $X_u = \{\mathbf{x}_1^{(u)}, \dots, \mathbf{x}_t^{(u)}, \dots, \mathbf{x}_{N-1}^{(u)}\}$ , 其中,  $\mathbf{x}_t^{(u)}$  是  $b_t^{(u)} \in \mathcal{B}$  的独热编码向量,  $b_t^{(u)}$  是用户  $u$  在  $t$  时刻的行为,  $N$  是行为序列的最大长度.

通过上述定义, 预测下一个用户行为的序列推荐任务定义如下: 给定历史的用户行为序列  $X_u$ , 通过学习一个预测模型  $\mathcal{M}: X_u \rightarrow \mathbf{p}_u$ , 输出用户  $u$  在  $N$  时刻与候选集中所有行为交互的概率分布  $\mathbf{p}_u$ . 通过将  $\mathbf{p}_u$  中的概率值从高到低排序, 可以推断出概率最高的行为是用户  $u$  最有可能交互的下一个行为  $\hat{y}$ . 值得注意的是, 下一个行为的预测是序列到序列 (Sequence to sequence, Seq2Seq) 预测的基础. 通过将历史序列和已预测的行为共同作为新的序列, 可迭代预测下一个行为, 实现 Seq2Seq 预测.

### 2.2 总体框架

本文的核心思想是集成多个具有差异的同伴网络建模动态兴趣的固定向量表示, 同时通过加强同伴在训练阶段的知识传递, 优化自身对行为序列的建模, 共同提升推荐精度. 因此, 本文提出了一种同伴知识互增强下的序列推荐方法 (PeerRec). 该方法主要以目前主流的多头自注意力网络为主体, 设计了包含多同伴网络的模型结构和同伴知识互增强的两阶段训练.

如图 2 所示, PeerRec 模型主要由行为序列嵌入层、用户兴趣表示层和行为预测层组成. 用户兴趣表示层需要学习输入序列中被掩盖行为的用户兴趣固定向量表示, 并通过行为预测层输出其与候选集中行为匹配的概率分布. 在训练阶段, 用户兴趣表示层中的多个同伴网络分别建模用户兴趣, 并按照人类由易到难的认知过程开展基于知识蒸馏和刻意训练的两阶段的互相学习 (如图中虚线箭头所指向的  $\mathbf{p}^{(1)}$  和  $\mathbf{p}^{(2)}$ ). 由于多个同伴网络知识互相传递, 在后续测试及工业部署阶段, 可以仅保留一个同伴网络, 提升推荐系统的在线推断效率.

值得注意的是, 互相学习通常需要双方有一定的共识和足够的差异性. 现有工作表明<sup>[20, 30]</sup>, 随机初始化就足以使模型获得多样性, 学习具有差异的用户兴趣固定向量表示. 因此, 本文在设计多个同伴网络时, 采用了共享的嵌入矩阵和相同的序列学习结构促使同伴网络对序列关系的捕获方式形成共识, 并通过不同的随机初始化方式确保同伴网络具备捕获差异化用户兴趣的能力.

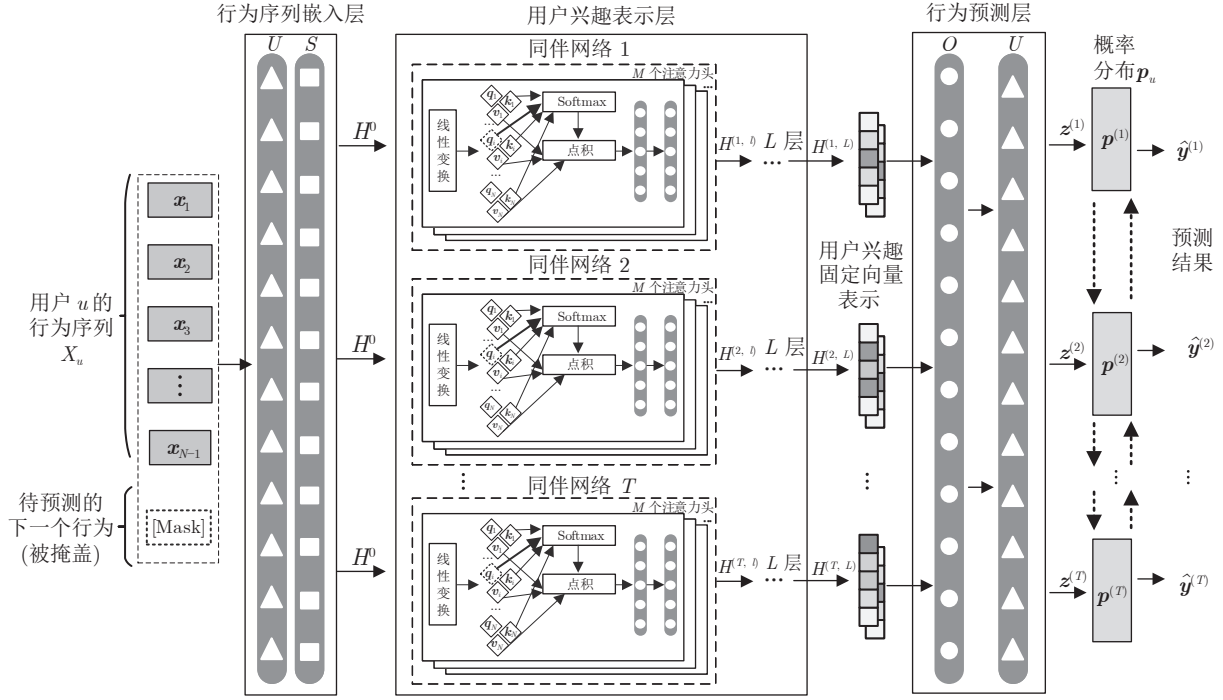


图2 PeerRec 模型的网络结构

Fig.2 The architecture of our proposed PeerRec

### 2.3 行为序列嵌入层

如图2左侧所示,行为序列嵌入层主要包含行为嵌入矩阵  $U$  和位置嵌入矩阵  $S$ . 首先,序列中的行为通过查询行为嵌入矩阵  $U \in \mathbf{R}^{|\mathcal{B}| \times d}$  转换为一个固定长度的表示向量,其中,  $|\mathcal{B}|$  是用户行为集合的大小,  $d$  是向量的维度. 行为嵌入矩阵  $U$  在输入和输出阶段共享,且没有引入任何预训练参数. 除了编码行为的特征信息,嵌入层还引入一个位置嵌入矩阵  $S \in \mathbf{R}^{N \times d}$ ,将用户行为在序列中的位置信息转换为一个固定长度的表示向量,其中,  $N$  是输入序列的最大长度. 最后,通过将行为特征表示向量和位置表示向量相加,可得到行为序列嵌入层的输出  $\mathbf{h}_i^0 = \mathbf{x}_i U + \mathbf{s}_i$ ,其中,  $\mathbf{x}_i$ ,  $\mathbf{s}_i$  和  $\mathbf{h}_i^0$  分别代表输入元素独热编码向量、第  $i$  个元素的位置表示向量和最终的输出向量. 嵌入层的输出矩阵  $H^0 = [\mathbf{h}_1^0, \dots, \mathbf{h}_N^0]$  将会分别输入到用户兴趣表示层的多个同伴网络中.

### 2.4 用户兴趣表示层

#### 2.4.1 用户兴趣表示学习

由于多头自注意力机制<sup>[42]</sup>可以捕获不同子空间的信息,提升模型的表达能力. 本文在现有工作<sup>[7]</sup>的基础上,设计  $T$  个结构相同但随机初始化参数不同的自注意力同伴网络,学习用户的兴趣表示. 如

图2所示,任意一个同伴网络都由  $L$  层编码器组成. 编码器包含一个多头自注意力子层和前馈网络子层,其中第  $l$  层编码器的输入为  $H^{l-1}$ ,输出为  $H^l$ .

多头自注意力子层将输入的编码矩阵  $H^{l-1}$  同时映射到  $M$  个子空间,得到  $M$  个注意力头的表示矩阵  $\{hd_1, \dots, hd_m, \dots, hd_M\}$ ,然后将这些矩阵拼接在一起,输出  $A_0^l$ . 该过程的定义为

$$A_0^l = \text{Concat}(hd_1, \dots, hd_m, \dots, hd_M) W_0^l + H^{l-1}$$

$$hd_m = \text{Softmax} \left( \frac{(H^{l-1} W_{Q_m}^l)(H^{l-1} W_{K_m}^l)^T}{\sqrt{d}} \right) \times (H^{l-1} W_{V_m}^l) \quad (2)$$

其中,  $W_{Q_m}^l, W_{K_m}^l, W_{V_m}^l \in \mathbf{R}^{d \times \frac{d}{M}}$  分别是第  $m$  个子空间的映射矩阵.  $W_0^l \in \mathbf{R}^{d \times d}$  是输出的映射矩阵,  $M$  是自注意力头的数量.  $A_0^l$  是经过残差连接后的输出.

为了引入更多的非线性因素,多头自注意力子层后面连接了两层前馈网络,具体为

$$\begin{cases} A_1^l = \text{GeLu}(A_0^l W_1^l + \mathbf{b}_1^l) \\ A_2^l = A_1^l W_2^l + \mathbf{b}_2^l \end{cases} \quad (3)$$

其中,  $W_1^l \in \mathbf{R}^{d \times d_h}$  和  $W_2^l \in \mathbf{R}^{d_h \times d}$  是两层前馈网络的权重矩阵,  $\mathbf{b}_1^l \in \mathbf{R}^{d_h}$  和  $\mathbf{b}_2^l \in \mathbf{R}^d$  是前馈网络的偏置向量.  $d_h$  是中间隐藏层的维度.  $\text{GeLu}(\cdot)$  是激活

函数. 经过残差连接后, 第  $l$  层编码器的输出为  $H^l = A_2^l + A_0^l$ . 第  $L$  层的输出  $H^L$  代表从行为序列中学习的用户兴趣表示.

#### 2.4.2 兴趣表示与同伴知识互增强的关系

为了引入多个同伴建模行为序列中的动态兴趣, 本文在用户兴趣表示层中设计了多个结构相同但参数不同的同伴网络, 每个同伴网络在前向传播的过程中独立地学习差异化的用户兴趣固定向量表示. 对于用户兴趣表示矩阵  $H^L$ , 本文进一步定义第  $j$  个同伴网络输出的用户兴趣表示  $H^{(j,L)} = [\mathbf{h}_1^{(j,L)}, \dots, \mathbf{h}_N^{(j,L)}]$ . 其中, 被掩盖行为的兴趣表示向量  $\mathbf{h}_N^{(j,L)}$  可通过共享的行为预测层计算与候选集中行为匹配后的概率分布  $\mathbf{p}^{(j)}$ .

由于不同的同伴网络可以学习到具有差异的用户兴趣表示, 通过行为预测层计算的概率分布  $\mathbf{p}^{(j)}$  也会反映出差异化的知识. 因此, 本文设计了一种同伴知识互增强训练方式, 通过同伴间互相交换概率分布  $\mathbf{p}^{(j)}$ , 优化兴趣表示的学习.

#### 2.5 行为预测层

如图 2 右侧所示, 经过  $L$  层同伴网络迭代的序列特征学习后, 被掩盖的下一个行为的用户兴趣表示向量  $\mathbf{h}_N^{(j,L)}$  输入到行为预测层. 该层包含一个在同伴网络间共享的前馈网络  $O$  和与行为序列嵌入层共享的行为嵌入矩阵  $U$ . 在第  $j$  个同伴网络中, 被掩盖行为的概率分布  $\mathbf{p}^{(j)}$  的计算过程为

$$\begin{cases} \mathbf{z}^{(j)} = \text{GeLu}(\mathbf{h}_N^{(j,L)} W_O + \mathbf{b}_O) U^T + \mathbf{b}_U \\ \mathbf{p}^{(j)} = \text{Softmax}(\mathbf{z}^{(j)}) \end{cases} \quad (4)$$

其中,  $W_O \in \mathbf{R}^{d \times d}$  是前馈网络的权重矩阵,  $\mathbf{b}_O \in \mathbf{R}^d$  和  $\mathbf{b}_U \in \mathbf{R}^{|\mathcal{B}|}$  分别是前馈网络和行为嵌入矩阵的偏置向量.  $\mathbf{z}^{(j)} = [z_1^{(j)}, \dots, z_{|\mathcal{B}|}^{(j)}]$  是模型最终的输出. 最后, 本文通过  $\text{Softmax}(\cdot)$  函数计算不同用户兴趣表示推导出的概率分布, 并利用该概率分布进行两阶段的同伴知识互增强的训练.

### 3 同伴知识互增强的训练

为了融合多个同伴网络对用户兴趣的建模, 提升测试阶段单模型的推荐性能, 本文结合课程学习<sup>[32]</sup>与知识迁移技术, 设计了由易到难的两阶段同伴知识互增强训练.

1) 第 1 阶段的知识蒸馏利用教师模型输出的概率分布作为软标签, 通过相对熵关注高置信度样本, 拉近同伴之间兴趣表示的距离. 借鉴文献 [39–40] 中的定义, 对于第  $i$  个被掩盖的待预测行为, 第  $j$  个同伴网络在第 1 阶段的损失函数定义为

$$\mathcal{L}_{S1}^{(j)} = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \left( -y_{i,*} \log(p_{i,*}^{(j)}) + \frac{1}{T-1} \sum_{l=1, l \neq j}^T \text{D}_{\text{KL}}(\mathbf{p}_i^{(l)} \parallel \mathbf{p}_i^{(j)}) \right) \quad (5)$$

其中,  $\mathcal{I}$  为训练批次中所有样本的集合,  $\mathbf{y}_i = [y_{i,1}, \dots, y_{i,*}, \dots, y_{i,|\mathcal{B}|}]$  为被掩盖行为的真实标签,  $\mathbf{p}_i^{(j)} = [p_{i,1}^{(j)}, \dots, p_{i,*}^{(j)}, \dots, p_{i,|\mathcal{B}|}^{(j)}]$  为输出的概率分布, 即所有候选行为下一次发生的可能性,  $*$  和  $i$  分别为概率分布中正例和负例的索引,  $T$  为同伴网络的数量,  $\mathbf{p}_i^{(l)}$  为第  $l$  个同伴网络输出的概率分布.  $\text{D}_{\text{KL}}(\cdot \parallel \cdot)$  为两个同伴网络输出概率分布的 KL (Kullback-Leibler) 散度.

当存在多个同伴网络时, 式 (5) 采用了损失平均的方式融合同伴的知识, 即同伴网络  $j$  依次向其他  $T-1$  个同伴学习, 再将彼此的 KL 散度均值作为损失. 相比于另一种概率平均的方式 ( $\mathbf{p}_i^{\text{avg}} = \frac{1}{T-1} \sum_{l=1, l \neq j}^T \mathbf{p}_i^{(l)}$ ), 损失平均可以缓解后验熵降低导致同伴网络间多样性丢失的问题. 已有文献 [26, 39] 对两种方式进行了深入分析, 进一步启发了本文在第 2 阶段的刻意训练中设计动态最小组策略, 并在第 4.7 节与损失平均进行实验对比.

2) 第 2 阶段基于刻意训练的互相学习以第 1 阶段知识蒸馏学习的模型参数作为新的初始化参数, 继续从低置信度样本中挖掘出可被加强训练的潜在样本, 增强模型对用户动态兴趣的建模. 刻意训练理论<sup>[33]</sup>认为教师应该找到一系列学生有待加强的知识, 并进行强化训练. 基于此, 本文设计了动态最小组策略 (第 3.1 节) 和同伴间刻意训练的损失函数 (第 3.2 节).

#### 3.1 动态最小组策略

本文认为低置信度的样本中通常包含有待加强的潜在样本, 而挖掘对受训学生模型有帮助的潜在样本的难点在于: 深度学习模型具有强大的数据拟合能力, 可以在没有捕获有效序列特征的情况下暴力拟合样本, 在后续训练过程中也会发生灾难性遗忘<sup>[43]</sup>, 仅凭受训学生模型本身输出的概率分布很难充分挖掘出需要关注的潜在样本. 因此, 本文首先设计了动态最小组策略  $\Omega(\cdot)$ , 利用多个同伴共同挖掘低置信度的潜在样本.

如图 3 所示, 动态最小组策略  $\Omega(\cdot)$  主要分为两步.

**步骤 1.** 为了尽可能多地挖掘出低置信度的样本, 本文收集  $T-1$  个同伴网络对正例预测概率最小的概率分布并重构为一个伪同伴. 对于第  $i$  个被掩盖的待预测行为, 第  $j$  个同伴网络的伪同伴定义为



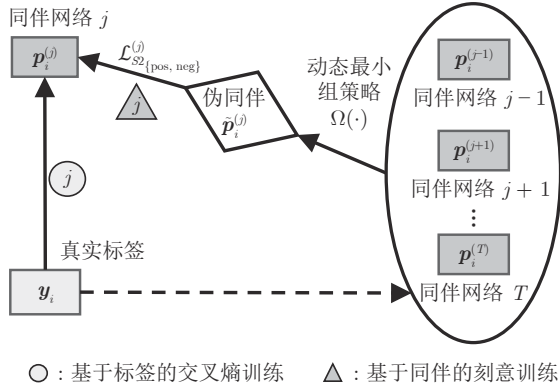


图 3 基于刻意训练的互相学习

Fig.3 An illustration of deliberate practice based mutual learning

$$\begin{aligned} \tilde{p}_i^{(j)} &= p_i^{(g)} \\ \text{s.t. } g &= \arg \min_k \Omega(j) = \{p_{i,*}^{(k)} | k = 1, \dots, T, \\ &\quad \text{且 } k \neq j\} \end{aligned} \quad (6)$$

其中,  $\tilde{p}_i^{(j)}$  是重构的伪同伴,  $p_i^{(g)}$  是同伴  $g$  对样本  $i$  预测的概率分布.

**步骤 2.** 由于低置信度样本通常也包括噪音样本, 对这些样本的过度拟合会影响模型的泛化性能<sup>[4]</sup>. 本文进一步假设如果一个序列不能被所有同伴网络预测正确, 那么它很可能是噪音且损失相对较大<sup>[4]</sup>. 因此, 本文在第 2 阶段的刻意训练中减少这些样本对模型学习的贡献, 进一步将伪同伴在样本  $i$  上的概率分布重新定义为

$$\tilde{p}_i^{(j)} = \begin{cases} y_i, & \text{rank}(p_{i,*}^{(\forall j)}) < \beta \times |\mathcal{I}| \\ \tilde{p}_i^{(j)}, & \text{其他} \end{cases} \quad (7)$$

其中,  $|\mathcal{I}|$  是训练批次中样本的总数量.  $p_{i,*}^{(\forall j)}$  代表任意一个同伴网络在样本  $i$  上输出正例的概率,  $\text{rank}(\cdot)$  是训练批次中样本  $i$  的交叉熵损失经降序排列后的名次. 考虑到过度地过滤不能被所有同伴网络预测正确的样本反而会丢失重要信息, 不利于建模动态兴趣, 本文还设置了一个超参数  $\beta$  来控制过滤噪音样本的比例.

### 3.2 同伴间刻意训练的损失函数

如图 3 中的三角形所示, 为了使受训学生模型注意到动态最小组策略筛选出的潜在样本, 本文利用伪同伴的概率分布计算受训学生模型训练的样本权重, 即刻意训练的强度. 对于第  $j$  个同伴网络, 其同伴间刻意训练的损失函数在正例上的定义为

$$\mathcal{L}_{S2_{\text{pos}}}^{(j)} = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} (1 - \tilde{p}_{i,*}^{(j)})^\gamma \log(p_{i,*}^{(j)}) \quad (8)$$

其中,  $p_{i,*}^{(j)}$  和  $\tilde{p}_{i,*}^{(j)}$  分别代表模型自身和伪同伴在正例上输出的概率值,  $(1 - \tilde{p}_{i,*}^{(j)})^\gamma$  为伪同伴概率影响下的刻意训练强度,  $\gamma$  为调节刻意训练强度的超参数.

式 (8) 中的刻意训练损失函数主要由伪同伴和模型自身输出的概率两部分构成, 可将多个同伴网络组成一个学习共同体, 从同一批次样本的全局训练信号上感知潜在的样本. 其包含如下性质:

**性质 1.** 模型自身和伪同伴响应一致, 都正确 (错误) 地预测出了下一个用户行为, 则输出概率  $p_{i,*}^{(j)}$  和  $\tilde{p}_{i,*}^{(j)}$  相对于同一个训练批次内的其他样本都较大 (小), 刻意训练强度  $(1 - \tilde{p}_{i,*}^{(j)})^\gamma$  相对较小 (大), 损失函数也相应较小 (大), 模型减少 (增加) 对该样本的学习.

**性质 2.** 模型自身和伪同伴响应不一致, 则损失函数的大小介于性质 1 的两种情况之间. 相对于同一个训练批次内的其他样本, 正确响应的模型为避免暴力拟合或灾难性遗忘<sup>[43]</sup> 会增加刻意训练强度; 错误响应的模型为关注损失更高的其他样本, 会降低刻意训练强度. 因此, 模型对样本的学习会同时考虑自身和同伴的学习状态, 从全局角度优化训练强度, 使模型更好地学习用户兴趣表示.

相似地, 为避免候选集中高度相似的行为对预测产生干扰, 本文对所有负例的损失函数定义为

$$\mathcal{L}_{S2_{\text{neg}}}^{(j)} = -\frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \sum_{k \neq *}^{|\mathcal{B}|} (\tilde{p}_{i,k}^{(j)})^\gamma \log(1 - p_{i,k}^{(j)}) \quad (9)$$

其中,  $p_{i,k}^{(j)}$  是概率分布  $p_i^{(j)}$  中的第  $k$  个负例元素的概率值,  $\tilde{p}_{i,k}^{(j)}$  是对应的伪同伴模型的概率值,  $(p_{i,k}^{(j)})^\gamma$  是伪同伴概率影响下的刻意训练强度,  $\gamma$  为调节刻意训练强度的超参数.

最后, 本文结合传统交叉熵损失 (图 3 中的圆形) 和同伴间刻意训练的损失函数, 计算最终的损失函数为

$$\begin{aligned} \mathcal{L}_{\text{total}}^{(j)} &= -\frac{\alpha}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} y_{i,*} \log(p_{i,*}^{(j)}) + \\ &\quad (1 - \alpha) \left( \mathcal{L}_{S2_{\text{pos}}}^{(j)} + \mathcal{L}_{S2_{\text{neg}}}^{(j)} \right) \end{aligned} \quad (10)$$

其中,  $\alpha$  为平衡传统交叉熵损失和同伴间刻意训练损失的超参数.

### 3.3 关于刻意训练对梯度的影响

为了更加直观地解释同伴网络是如何结合伪同伴和模型自身输出的概率建模动态的用户兴趣, 本节进一步探讨同伴间刻意训练的损失函数对第  $j$  个同伴网络梯度变化的影响. 对于同伴网络  $j$  中的正



例损失函数, 式 (8) 中关于模型最后一层输出  $z_*^{(j)}$  的梯度推导如下:

$$\begin{aligned} \frac{\partial \mathcal{L}_{S2_{\text{pos}}}^{(j)}}{\partial z_{i,*}^{(j)}} &= \frac{\partial \mathcal{L}_{S2_{\text{pos}}}^{(j)}}{\partial p_{i,*}^{(j)}} \frac{\partial p_{i,*}^{(j)}}{\partial z_{i,*}^{(j)}} = \\ &= -\frac{(1 - \tilde{p}_{i,*}^{(j)})^\gamma}{p_{i,*}^{(j)}} \times p_{i,*}^{(j)}(1 - p_{i,*}^{(j)}) = \\ &= (1 - \tilde{p}_{i,*}^{(j)})^\gamma (p_{i,*}^{(j)} - y_{i,*}) \end{aligned} \quad (11)$$

从式 (11) 中可以看出, 刻意训练中正例的梯度是标准交叉熵梯度的缩放, 而缩放因子是从伪同伴传递过来的刻意训练强度  $(1 - \tilde{p}_{i,*}^{(j)})^\gamma$ . 如果伪同伴对正例的预测发生了较大的偏离 (即  $\tilde{p}_{i,*}^{(j)} \approx 0$ ), 式 (11) 产生的梯度就接近于标准交叉熵梯度. 此时, 伪同伴会提示第  $j$  个同伴网络当前样本中的序列特征需要以较大梯度刻意训练, 如果第  $j$  个同伴网络也没有产生正确响应则产生较大的梯度; 如果第  $j$  个同伴网络产生正确响应则以中等强度的梯度学习. 只有所有同伴网络都产生正确响应, 才产生较小的梯度.

因此, 本文提出的同伴间刻意训练的损失函数可以在学习过程中动态地缩放梯度. 在训练阶段, 同伴网络会同时考虑其他网络在上一轮学习中输出的概率分布, 结合不同的用户兴趣表示来调整自身下一轮学习的方向, 直到彼此都预测正确.

## 4 实验与分析

为了评估提出的序列推荐模型 PeerRec 的有效性, 本文设置了以下 4 个研究问题 (Research question, RQ):

**RQ 1.** PeerRec 模型与目前最新的序列推荐模型相比, 推荐精度如何?

**RQ 2.** 本文提出的基于同伴知识互增强的训练方式, 对模型准确率的影响有多大?

**RQ 3.** 增加同伴网络的数量是否能进一步提升 PeerRec 模型的推荐精度?

**RQ 4.** 本文提出的 PeerRec 模型是否具有良好的在线推荐效率?

针对 RQ 1, 本文将对目前最新的序列推荐模型; 针对 RQ 2, 本文将通过抛弃实验设计不同的变体, 并观察模型的性能变化; 针对 RQ 3, 本文通过引入更多的同伴网络, 研究模型的性能变化; 针对 RQ 4, 本文在不同数据集上与最新的序列推荐模型对比, 观察模型的运行速率. 此外, 本文还对 PeerRec 模型的参数敏感性和不同初始化下的稳定性进行了讨论和分析.

### 4.1 数据集与预处理

通常, 用户的动态兴趣与序列中交互行为的数量存在较大的相关性. 一方面, 单个用户的交互行为较多, 更容易包含多样化的用户兴趣; 另一方面, 单个用户的交互行为较少, 可反映出有效信息不足导致的不确定性. 因此, 本文以平均序列长度为准则, 选取现有工作<sup>[1, 14, 17]</sup>中广泛使用的 3 个公开推荐数据集 (ML-1m 电影评分<sup>[17]</sup>、LastFM 音乐艺术家推荐<sup>[1]</sup>和 Toys 亚马逊评论子类别<sup>[1]</sup>) 进行了相关实验. 数据集的具体情况如表 1 所示.

表 1 实验集数据统计表  
Table 1 Statistics of dataset

	ML-1m	LastFM	Toys
用户数量	6 040	1 090	19 412
行为类别数量	3 416	3 646	11 924
最长序列的行为数量	2 275	897	548
最短序列的行为数量	16	3	3
序列的平均行为数量	163.50	46.21	6.63
序列行为数量的方差	192.53	77.69	8.50

借鉴文献 [1, 17] 中的数据预处理方式, 本文将用户行为按发生的时间先后顺序排列, 并剔除了序列长度小于 5 的不活跃用户, 将每个序列中的最末一个行为作为测试集, 倒数第二个行为作为验证集, 其余部分构成训练集. 最大序列长度  $N$  设置为 200. 为了保证序列的长度在最大范围内, 本文将较长的序列从右到左分割成多个子序列.

### 4.2 实验设置

本文使用 Adam 和线性衰减学习率 (初始值为  $1 \times 10^{-4}$ ) 训练模型, 训练的样本批量大小设置为 256. 在同伴网络中, 本文设置多头自注意力层数  $L$  为 2, 头数  $R$  为 2, 嵌入向量表示的维度  $d$  为 64, 中间隐藏层维度为 256. 本文采用目前主流的网格搜索方式<sup>[9, 14]</sup>, 通过验证集在  $\{0, 0.1, 0.2, 0.3, \dots, 0.9, 1.0\}$ ,  $\{0, 0.01, 0.02, 0.03\}$  和  $\{0, 0.5, 1.0, 1.5, 2.0\}$  范围内寻找同伴知识互增强训练超参数  $\alpha$ 、 $\beta$  和  $\gamma$  的最佳组合, 并报告其在测试集上的性能. 最后, 本文在 ML-1m、LastFM 和 Toys 数据集上得出的最佳参数组合  $(\alpha, \beta, \gamma)$  分别为  $(0.5, 0, 2.0)$ ,  $(0.5, 0.01, 1.0)$  和  $(0.5, 0.02, 1.0)$ .

本文的实验环境配置如下: Intel(R) Xeon(R) CPU E5-2650 v4 @ 2.20 GHz, NVIDIA TITAN Xp GPU (显存 12 GB), 11.0 版本的 CUDA. 实验代码采用 2.7 版本的 Python 和 1.14.0 版本的 Tens-

orflow 实现 PeerRec 模型。

### 4.3 评价指标

根据常用的评估方法<sup>[1, 17]</sup>, 序列推荐模型的性能可以通过 Top- $k$  命中率 (Hit ratio, HR@ $k$ )、Top- $k$  归一化折扣累计收益 (Normalized discounted cumulative gain, NDCG@ $k$ ) 和平均倒数排名 (Mean reciprocal rank, MRR) 评估, 这 3 个指标的取值越高, 模型的性能就越好。在本文的工作中, 指标的  $k$  值设为 {1, 5, 10}。当  $k$  取 1 时, HR@1 等于 NDCG@1。为了在大量的候选行为中实现高效计算, 本文随机选取 99 个负例行为与预测目标的行为进行排序。

### 4.4 基线模型

为了验证本文提出的 PeerRec 模型的有效性, 实验按照第 1.1 节中现有序列推荐方法对用户动态兴趣建模方式的分类, 设置了 6 个主流的序列推荐模型, 具体如下:

1) POP (Popularity). 最简单的基线模型, 仅通过用户与行为交互的频次 (流行度) 预测用户的下一个行为。

2) BERT4Rec (Sequential recommendation with bidirectional encoder representations from transformer)<sup>[17]</sup>. 一种基于 BERT 语言模型, 利用双向序列信息建模动态兴趣的推荐方法。

3) S<sup>3</sup>-Rec (Self-supervised learning for sequential recommendation)<sup>[1]</sup>. 一种利用互信息最大化建模序列和用户数据内部关联的序列推荐方法。为了公平对比, 本文去除了对额外用户信息建模的模块, 仅利用文中的 MIP (Masked item prediction) 和 SP (Segment prediction) 最大化行为序列的内部关联。

4) HyperRec (Next-item recommendation with sequential hypergraphs)<sup>[9]</sup>. 一种利用超图卷积建模不同时刻用户动态兴趣的推荐方法。

5) R-CE (Reweighted cross-entropy)<sup>[11]</sup>. 一种自适应的去噪策略, 本文使用该策略重新训练原始的 BERT4Rec 模型。

6) STOSA (Stochastic self-attention)<sup>[14]</sup>. 目前最新的建模序列动态兴趣和行为关系传递性的方法。

本文所使用的推荐模型 BERT4Rec、S<sup>3</sup>-Rec、HyperRec、STOSA 的源代码均由作者发布。所有模型都采用了原论文中推荐的参数设置。

### 4.5 基线模型的对比 (RQ 1)

表 2 列出了采用 2 个同伴网络时, 本文提出的 PeerRec 方法与基线方法的对比结果。为了便于比

较, 表 2 中加粗的数值表示最优的结果, 下划线标记的数值表示次优的结果。本文可得出如下结论:

1) 基于固定向量表示的方法在 ML-1m 电影数据集上表现较好, 但不同指标上的优势不稳定。从表 2 中可以看出, S<sup>3</sup>-Rec 在没有引入额外用户信息时表现不佳。HyperRec 虽然从行为序列的子图中提取了不同时刻的用户兴趣, 但序列子图所能提供的信息有限, 限制了模型的性能。BERT4Rec 的性能整体优于 S<sup>3</sup>-Rec 和 HyperRec, 说明 BERT4Rec 可以有效利用序列中的双向语义信息丰富兴趣表示的学习。基于去噪的 R-CE 方法在序列信息较丰富的 ML-1m 数据集上表现较好, 而在序列行为较少的 LastFM 和 Toys 数据集上表现较差, 说明用户交互行为较少时, 去噪方法更容易导致有效信息丢失, 使推荐性能欠佳。

2) 基于分布函数的 STOSA 方法在短序列上表现较好, 尤其在 Toys 数据集上取得了优越的性能, 原因在于概率分布增加了模型拟合的行为序列空间, 能够有效捕获用户行为之间的演变关系。在 Toys 数据集的 HR@1 指标上, PeerRec 性能稍弱于 STOSA, 原因在于 PeerRec 采用多同伴网络集成的方式, 可探索更大的解空间<sup>[18-19]</sup>, 在候选行为类别数量较大的 Toys 数据集上, 会更倾向于从整体优化推荐精度。

3) PeerRec 方法中的两个同伴网络在绝大多数指标上差异较小且优于基线方法。在 ML-1m、LastFM 和 Toys 数据集上, PeerRec 相比于最好的基线方法, 在 NDCG@5 指标上分别提升了 9.02%、5.03% 和 1.31%。原因在于 PeerRec 方法利用多个同伴网络建模行为序列中的动态兴趣, 可比基于分布函数的方法探索更大的表示空间, 且采用的两阶段同伴知识互增强训练方式优化了同伴网络对潜在样本中动态兴趣的建模能力。由于两个同伴性能差异较小并且得到共同提升, 在后续测试及部署阶段, 可以只保留一个同伴网络, 提升在线推荐效率。

### 4.6 同伴知识互增强训练的影响分析 (RQ 2)

如表 3 所示, 本文首先完全抛弃第 2 阶段的刻意训练, 分析其对模型性能的影响。然后, 如图 4 所示, 本文保留第 1 阶段的知识蒸馏并抛弃第 2 阶段刻意训练中的不同成分, 设置如下刻意训练的变体:

1) 变体 I. 抛弃式 (10) 中刻意训练对动态兴趣的建模 (即  $\alpha$  设置为 1), 仅保留第 1 阶段的知识蒸馏和第 2 阶段中的基于硬标签的交叉熵训练;

2) 变体 II. 抛弃式 (7) 中利用动态最小组策略过滤低置信度样本中的噪声。

通过对比知识蒸馏和 2 个变体在最严格的

表 2 与基线模型在精度指标上的对比  
Table 2 The comparison with baselines in terms of accuracy based metrics

数据集	模型	HR@1	HR@5	HR@10	NDCG@5	NDCG@10	MRR
ML-1m	POP	0.0407	0.1603	0.2775	0.1008	0.1383	0.1233
	BERT4Rec <sup>[17]</sup>	0.3695	<u>0.6851</u>	0.7823	<u>0.5375</u>	<u>0.5690</u>	0.5108
	S <sup>3</sup> -Rec <sup>[1]</sup>	0.2897	0.6575	<u>0.7911</u>	0.4557	0.5266	0.4535
	HyperRec <sup>[9]</sup>	0.3180	0.6631	0.7738	0.5014	0.5375	0.4731
	R-CE <sup>[11]</sup>	<u>0.3988</u>	0.6478	0.7404	0.5327	0.5627	<u>0.5179</u>
	STOSA <sup>[14]</sup>	0.3222	0.6546	0.7844	0.4967	0.5389	0.4716
	PeerRec (同伴 1)	0.4250	0.7197	<b>0.8141</b>	0.5843	0.6150	0.5600
	PeerRec (同伴 2)	<b>0.4252</b>	<b>0.7225</b>	<b>0.8141</b>	<b>0.5860</b>	<b>0.6157</b>	<b>0.5610</b>
LastFM	POP	0.0202	0.0908	0.1780	0.0544	0.0825	0.0771
	BERT4Rec <sup>[17]</sup>	0.1091	<u>0.3294</u>	0.4614	<u>0.2227</u>	<u>0.2648</u>	<u>0.2266</u>
	S <sup>3</sup> -Rec <sup>[1]</sup>	<u>0.1156</u>	0.2844	0.4229	0.2003	0.2452	0.2148
	HyperRec <sup>[9]</sup>	0.1146	0.3147	<u>0.4688</u>	0.2150	0.2646	0.2241
	R-CE <sup>[11]</sup>	0.0651	0.1835	0.2862	0.1243	0.1570	0.1397
	STOSA <sup>[14]</sup>	0.0752	0.2165	0.3412	0.1458	0.1860	0.1556
	PeerRec (同伴 1)	<b>0.1294</b>	<b>0.3495</b>	0.4789	<b>0.2339</b>	0.2755	0.2341
	PeerRec (同伴 2)	0.1248	0.3358	<b>0.4835</b>	0.2318	<b>0.2796</b>	<b>0.2378</b>
Toys	POP	0.0260	0.1046	0.1848	0.0652	0.0909	0.0861
	BERT4Rec <sup>[17]</sup>	0.1390	0.3379	<u>0.4596</u>	0.2409	0.2802	0.2444
	S <sup>3</sup> -Rec <sup>[1]</sup>	0.0990	0.3023	0.4393	0.2021	0.2463	0.2081
	HyperRec <sup>[9]</sup>	0.1147	0.2875	0.3909	0.2031	0.2365	0.2087
	R-CE <sup>[11]</sup>	0.1130	0.3189	0.4529	0.2179	0.2611	0.2233
	STOSA <sup>[14]</sup>	<b>0.1838</b>	<u>0.3587</u>	0.4550	<u>0.2749</u>	<u>0.3059</u>	<u>0.2732</u>
	PeerRec (同伴 1)	<u>0.1794</u>	0.3703	<b>0.4785</b>	<b>0.2785</b>	<b>0.3134</b>	<b>0.2810</b>
	PeerRec (同伴 2)	0.1782	<b>0.3706</b>	0.4778	0.2781	0.3127	0.2803

HR@1 指标上输出的平均值, 可得到如下结论.

1) 第 2 阶段的刻意训练可以在第 1 阶段知识蒸馏的基础上进一步提升性能. 本文提出的 PeerRec 模型是基线模型 BERT4Rec 的扩展, 即仅采用 1 个同伴网络时, PeerRec 模型可退化为 BERT4Rec 模型. 结合表 3 和图 4 可以看出, 实验结果与本文在第 1.2 节中的分析一致, 知识蒸馏的性能优于传统单模型的 BERT4Rec, 而 PeerRec 模型可以在知识蒸馏的基础上进一步增强对潜在样本中用户动态兴趣的学习, 表明了本文提出的两阶段同伴知识互增强训练的有效性.

2) 采用 2 个同伴网络进行两阶段的学习, 但在第 2 阶段只进行硬标签学习, 变体 I 的性能会出现不同程度下降. 如图 4 所示, 变体 I 与完整的 PeerRec 模型相比, 在 ML-1m、LastFM 和 Toys 数据集上的平均性能分别下降了 1.65%、7.43% 和 11.54%. 这表明通过刻意训练的互相学习可以有效地利用多个同伴建模序列中的动态兴趣, 提升模型的性能.

3) 抛弃动态最小组策略的噪音过滤后, 模型性

表 3 知识蒸馏与刻意训练对比

Table 3 The comparison between knowledge distillation and deliberate practice

	数据集	HR@1	NDCG@5	MRR
知识蒸馏 <sup>[30]</sup>	ML-1m	0.3952	0.5656	0.5386
	LastFM	0.1119	0.2301	0.2314
	Toys	0.1693	0.2761	0.2767
刻意训练 PeerRec	ML-1m	0.4251	0.5852	0.5605
	LastFM	0.1271	0.2329	0.2360
	Toys	0.1788	0.2783	0.2807

能出现下降. 通过对比完整的 PeerRec 模型和变体 II 可以发现, 在 LastFM 和 Toys 数据集上, 本文以比例  $\gamma$  动态地过滤在所有同伴网络中交叉熵损失都较高的样本, 减少它们对模型刻意训练的损失贡献, 有助于序列推荐精度的提升. 相比于纯粹去噪的 R-CE 方法, 完整的 PeerRec 模型在 ML-1m 数据集上没有过滤噪音 ( $\gamma$  设置为 0), 取得了更佳的性能. 这主要是因为该数据集用户序列较长且候选集较



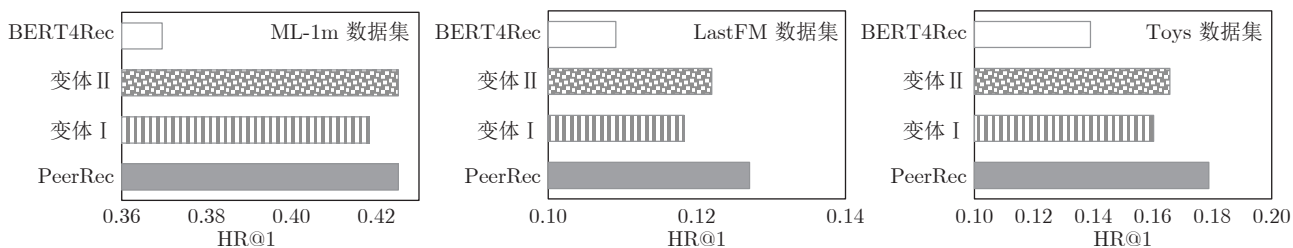


图 4 PeerRec 变体在 HR@1 指标上的对比

Fig. 4 The comparison between the variants of PeerRec in terms of HR@1

小, 可以通过动态兴趣建模利用信息, 避免纯粹去噪方法对信息的丢失。

#### 4.7 同伴网络数量的分析 (RQ 3)

为了研究 PeerRec 模型中同伴网络数量和动态最小组策略的影响, 本文在相同的硬件条件下, 通过采用行为类别数量较少的 ML-1m 数据集并减小样本批量的大小 (批量大小为 128), 使有限的内存可以容纳更多的同伴网络. 通过逐步增加同伴网络的数量, 对比分析动态最小组策略和损失平均<sup>[39]</sup>融合方式在 HR@1 指标上的均值 (图 5 中的阴影部分为动态最小组策略中同伴波动的范围), 可得出如下结论: 三个同伴网络可以进一步提升性能. 从图 5 中可以看出, 相对于 2 个同伴网络, 3 个同伴网络在 ML-1m 数据集上的性能提升了 1.36%. 实验结果表明, 刻意训练可以充分利用多个同伴网络之间的知识差异, 互相提升性能。

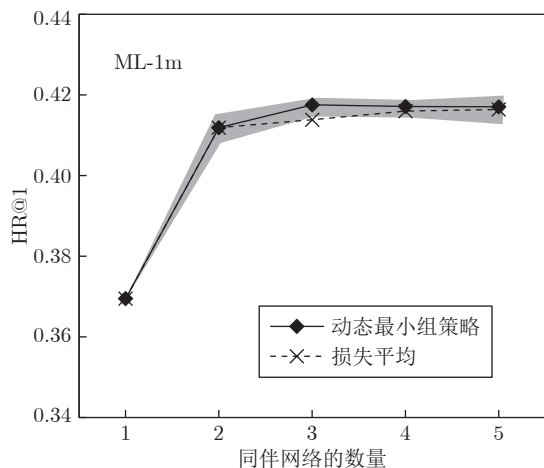


图 5 不同同伴网络数量的性能对比

Fig. 5 The evaluation on different number of peers

多个同伴网络 (3 个及以上) 提升效果有限. 从图 5 中可以看出, 在相同的参数设置下, 随着同伴网络数量增多, 模型性能逐渐上升, 而后平稳变化. 这说明采用 2 个或 3 个同伴网络时, 模型性能提升

的效果就足以饱和, 并不是同伴网络的数量越多, 模型的性能越好。

相比于损失平均的同伴融合方式, 第 2 阶段的刻意训练采用动态最小组策略, 可以取得更佳的性能. 这主要是因为通过同伴选择策略可以有效避免均值计算带来的同伴多样性丢失问题。

#### 4.8 在线推荐效率分析 (RQ 4)

不同于现有基线模型采用的单模型结构, PeerRec 模型利用多同伴网络互相学习的方式建模动态兴趣, 且在第 4.7 节中验证了 2 个或 3 个同伴网络就足以获得较好的性能. 因此, 本节进一步评估采用 2 个同伴网络的 PeerRec 模型和基线模型的运行效率. 如图 6 所示, 本文列出了样本批量为 256 时模型的迭代速率 (iter/s), 并标出了 PeerRec 模型在训练和测试阶段的速率差异. 可以发现, PeerRec 模型在 ML-1m 和 LastFM 数据集上超过目前最新的 STOSA 模型, 并在 Toys 数据集上保持相当的速率. 这主要是因为 PeerRec 模型在测试阶段只保留 1 个同伴网络, 模型的参数数量减少到与 BERT4Rec 模型一致, 可以显著提升在线推荐速率, 而 STOSA 模型概率分布的参数 (如均值和方差) 进一步增加了模型的参数量。

在训练阶段, 由于 PeerRec 模型采用了多个同伴网络, 迭代速率低于基于固定向量表示的基线模型, 但与基于分布函数的 STOSA 模型相当. 从图 6 可以看出, STOSA 模型迭代速率在 Toys 数据集上增大, 原因在于 STOSA 模型采用的贝叶斯个性化排序 (Bayesian personalized ranking, BPR) 损失在候选集较大时的计算代价更小. 值得注意的是, 在工业部署和实际应用中, 训练过程以较高的计算代价换取精度的提升通常是可以接受的, 而测试阶段的在线推荐通常对模型的效率有较高的要求. 因此, 本文提出的 PeerRec 模型的优势在于, 通过多个同伴间的互相学习不仅使模型精度获得提升, 而且可以保持良好的在线推荐效率。

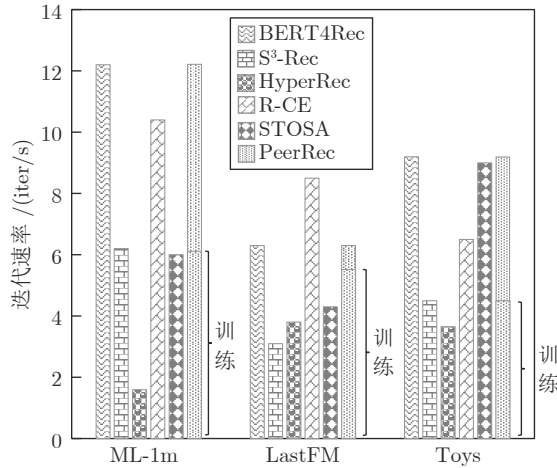


图6 Batch 大小设置为 256 时, 模型的迭代速率

Fig.6 The running speed of different models with batch size 256

#### 4.9 参数敏感性分析

本文在第 4.2 节给出了通过网格搜索<sup>[9, 14]</sup>获得的最佳参数组合. 为了单独研究不同参数对第 2 阶段刻意训练的影响, 本节分别调节最佳参数组合下超参数  $\alpha$ ,  $\beta$ ,  $\gamma$  的大小, 同时保持其他参数值不变, 并观察模型在测试集上的性能变化, 可以观察到如下现象: 1) 关于损失平衡因子  $\alpha$  的影响. 以图 7(a) 中阴影区域为界, 可以看出随着损失平衡因子  $\alpha$  由小变大, 模型会分别进入 3 种不同的学习状态, 即基于刻意训练的主导状态、刻意训练与交叉熵学习的平衡状态、基于交叉熵学习的主导状态. 在前两个状态中 (基于刻意训练的主导状态和平衡状态), 可以观察到两个同伴网络的性能差异较小, 而基于交叉熵学习的主导状态中两个模型的性能差异逐渐拉大, 表明本文提出的基于刻意训练的互相学习, 可以拉近两个同伴的距离. 此外, 相对于知识蒸馏, 交叉熵中对硬标签的训练, 同样可以关注到一部分潜在在样本, 只是增强的效果有限. 从图 7(a) 中可以

看出, 对于不同的学习状态, 曲线都会出现下降或先上升再下降的趋势, 这是因为在学习状态的切换过程中, 非主导作用的学习起到正则项的作用, 可以给模型提供关于参数学习的额外信息, 避免模型陷入欠拟合或过拟合. 2) 关于过滤噪音比例  $\beta$  的影响. 从图 7(b) 中可以看出, 在 ML-1m 数据集上, 随着过滤比例  $\beta$  增大, 模型性能下降. 这表明简单地将包含动态兴趣的序列当作噪音处理, 会损失一部分有用的信息. 3) 关于调节刻意训练强度的超参数  $\gamma$  的影响. 从图 7(c) 中可以看出, 随着  $\gamma$  增大, 模型性能整体呈现上升趋势, 说明较大的  $\gamma$  值会使模型对同伴传递的知识更加敏感, 一定程度上有利于模型对序列中的动态兴趣建模.

#### 4.10 初始化对模型稳定性的影响

本文主要采用截断正态分布 (Truncated normal distribution, TND) 随机初始化同伴网络, 使模型获得多样性<sup>[20, 39]</sup>. 为了分析不同初始化方式对模型稳定性的影响, 本文在两个同伴网络的基础上, 进一步对比另外两种主流的初始化方式, 即 Xavier<sup>[44]</sup> 和 Kaiming<sup>[45]</sup>, 并观察模型在各个指标上均值的差异.

从表 4 中可以看出, 本文提出的 PeerRec 模型受初始化方式的影响较小, 在 3 个数据集上都表现出良好的性能. 该实验结果与本文在第 3.2 节中的分析一致. 这主要是因为多个同伴网络从不同的初始点开始互相学习, 弱化了初始化对同伴网络的影响, 并且多个网络组成一个学习共同体, 可以从全局的角度探索更大的解空间, 寻找到比单个模型更佳的局部最优解, 获得较好的泛化性能.

## 5 总结和展望

本文针对序列推荐任务中, 由用户动态兴趣导致的行为序列建模不确定性问题进行研究, 提出了一种同伴知识互增强下的序列推荐方法 (PeerRec).

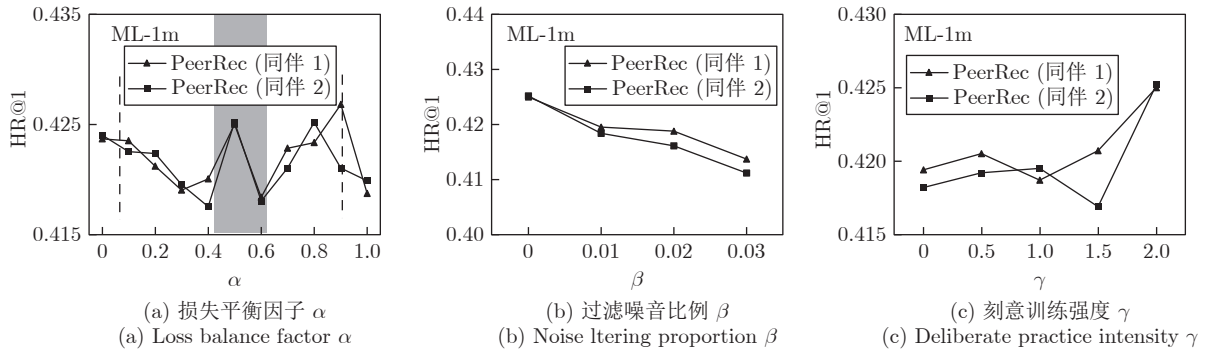


图7 超参数敏感性分析

Fig.7 Sensitivity analysis of hyper-parameters

表 4 PeerRec 模型采用不同初始化的性能对比  
Table 4 The performance comparison between different initializations of our PeerRec

数据集	初始化方式	HR@1	NDCG@5	MRR
ML-1m	TND	0.4251	0.5852	0.5605
	Xavier	0.4263	0.5852	0.5600
	Kaiming	0.4278	0.5911	0.5652
LastFM	TND	0.1271	0.2329	0.2360
	Xavier	0.1294	0.2397	0.2424
	Kaiming	0.1247	0.2257	0.2342
Toys	TND	0.1788	0.2783	0.2807
	Xavier	0.1775	0.2794	0.2811
	Kaiming	0.1806	0.2776	0.2804

该方法结合多个同伴网络和同伴知识互增强的训练, 允许多个同伴在同一框架下互相学习, 通过交换不同用户兴趣表示预测的概率分布, 优化自身对用户动态兴趣的建模. 通过同伴知识互增强的训练, 所有同伴网络的推荐性能可以得到共同提升, 且在测试阶段仅保留任意一个同伴网络, 可以获得良好的在线推荐效率. 3 个公开数据集上的实验结果表明了本文提出的 PeerRec 方法的优越性.

本文的后续工作将从两方面展开. 一方面, PeerRec 模型主要关注序列的下一个行为. 然而, 实际场景需要考虑更多的状况, 如多个用户之间的关联、连续行为的预测. 如何扩展 PeerRec 模型到多模态序列推荐、序列到序列 (Seq2Seq) 推荐等复杂的序列任务依然存在挑战. 另一方面, 第 2 阶段刻意训练虽然可以有效弥补第 1 阶段知识蒸馏的不足, 但这种分开训练的方式粒度较粗, 为进一步研究知识蒸馏内部机制留下了空间.

## References

- Zhou K, Wang H, Zhao W X, Zhu Y T, Wang S R, Zhang F Z, et al. S3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York, USA: ACM, 2020. 1893–1902
- Rao Zi-Yun, Zhang Yi, Liu Jun-Tao, Cao Wan-Hua. Recommendation methods and systems using knowledge graph. *Acta Automatica Sinica*, 2021, **47**(9): 2061–2077 (饶子云, 张毅, 刘俊涛, 曹万华. 应用知识图谱的推荐方法与系统. *自动化学报*, 2021, **47**(9): 2061–2077)
- Li X C, Liang J, Liu X L, Zhang Y. Adversarial filtering modeling on long-term user behavior sequences for click-through rate prediction. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. Madrid, Spain: ACM, 2022. 1969–1973
- Tang Wen-Bing, Ren Zheng-Yun, Han Fang. Attention-based collaborative convolutional dynamic network for recommendation. *Acta Automatica Sinica*, 2021, **47**(10): 2438–2448 (汤文兵, 任正云, 韩芳. 基于注意力机制的协同卷积动态推荐网络. *自动化学报*, 2021, **47**(10): 2438–2448)
- Guo Lei, Li Qiu-Ju, Liu Fang-Ai, Wang Xin-Hua. Shared-account cross-domain sequential recommendation with self-attention network. *Journal of Computer Research and Development*, 2021, **58**(11): 2524–2537 (郭磊, 李秋菊, 刘方爱, 王新华. 基于自注意力网络的共享账户跨域序列推荐. *计算机研究与发展*, 2021, **58**(11): 2524–2537)
- Rao X, Chen L S, Liu Y, Shang S, Yao B, Han P. Graph-flash-back network for next location recommendation. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Washington, USA: ACM, 2022. 1463–1471
- Meng Xiang-Wu, Liang Bi, Du Yu-Lu, Zhang Yu-Jie. A survey of evaluation for location-based mobile recommender systems. *Chinese Journal of Computers*, 2019, **42**(12): 2695–2721 (孟祥武, 梁弼, 杜雨露, 张玉洁. 基于位置的移动推荐系统效用评价研究. *计算机学报*, 2019, **42**(12): 2695–2721)
- Hu K X, Li L, Liu J Q, Sun D. DuroNet: A dual-robust enhanced spatial-temporal learning network for urban crime prediction. *ACM Transactions on Internet Technology*, 2021, **21**(1): Article No. 24
- Wang J L, Ding K Z, Hong L J, Liu H, Caverlee J. Next-item recommendation with sequential hypergraphs. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2020. 1101–1110
- Chen Cong, Zhang Wei, Wang Jun. Session-based sequential recommendation with auxiliary time prediction. *Chinese Journal of Computers*, 2021, **44**(9): 1841–1853 (陈聪, 张伟, 王骏. 带有时间预测辅助任务的会话式序列推荐. *计算机学报*, 2021, **44**(9): 1841–1853)
- Wang W J, Feng F L, He X N, Nie L Q, Chua T S. Denoising implicit feedback for recommendation. In: Proceedings of the 14th ACM International Conference on Web Search and Data Mining. New York, USA: ACM, 2021. 373–381
- Neupane K P, Zheng E, Yu Q. MetaEDL: Meta evidential learning for uncertainty-aware cold-start recommendations. In: Proceedings of the IEEE International Conference on Data Mining (ICDM). Auckland, New Zealand: IEEE, 2021. 1258–1263
- Fan Z W, Liu Z W, Wang S, Zheng L, Yu P S. Modeling sequences as distributions with uncertainty for sequential recommendation. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management. Queensland, Australia: ACM, 2021. 3019–3023
- Fan Z W, Liu Z W, Wang Y, Wang A, Nazari Z, Zheng L, et al. Sequential recommendation via stochastic self-attention. In: Proceedings of the ACM Web Conference. Lyon, France: ACM, 2022. 2036–2047
- Jiang J Y, Yang D Q, Xiao Y H, Shen C L. Convolutional Gaussian embeddings for personalized recommendation with uncertainty. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao, China: AAAI Press, 2019. 2642–2648
- Zhou X L, Liu H, Pourpanah F, Zeng T Y, Wang X Z. A survey on epistemic (model) uncertainty in supervised learning: Recent advances and applications. *Neurocomputing*, 2022, **489**: 449–465
- Sun F, Liu J, Wu J, Pei C H, Lin X, Ou W W, et al. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China: ACM, 2019. 1441–1450
- Ovadia Y, Fertig E, Ren J, Nado Z, Sculley D, Nowozin S, et al. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver, Canada: Curran Associates Inc., 2019. Article No. 1254



- 19 Fort S, Hu H Y, Lakshminarayanan B. Deep ensembles: A loss landscape perspective. arXiv preprint arXiv: 1912.02757, 2019.
- 20 Lakshminarayanan B, Pritzel A, Blundell C. Simple and scalable predictive uncertainty estimation using deep ensembles. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA: Curran Associates Inc., 2017. 6405–6416
- 21 Renda A, Barsacchi M, Bechini A, Marcelloni F. Comparing ensemble strategies for deep learning: An application to facial expression recognition. *Expert Systems With Applications*, 2019, **136**: 1–11
- 22 Deng D D, Wu L, Shi B E. Iterative distillation for better uncertainty estimates in multitask emotion recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). Montreal, Canada: IEEE, 2021. 3550–3559
- 23 Reich S, Mueller D, Andrews N. Ensemble distillation for structured prediction: Calibrated, accurate, fast — Choose three. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg, USA: ACL, 2020. 5583–5595
- 24 Jiao X Q, Yin Y C, Shang L F, Jiang X, Chen X, Li L L, et al. TinyBERT: Distilling BERT for natural language understanding. In: Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2020. Stroudsburg, USA: ACL, 2020. 4163–4174
- 25 Zhu J M, Liu J Y, Li W Q, Lai J C, He X Q, Chen L, et al. Ensembled CTR prediction via knowledge distillation. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management. New York, USA: ACM, 2020. 2941–2958
- 26 Kang S K, Hwang J, Kweon W, Yu H. DE-RRD: A knowledge distillation framework for recommender system. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management. New York, USA: ACM, 2020. 605–614
- 27 Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. In: Proceedings of the 28th Conference on Neural Information Processing Systems. Montreal, Canada: Curran Associates, 2014. 1–9
- 28 Shen Z Q, Liu Z C, Xu D J, Chen Z T, Cheng K T, Savvides M. Is label smoothing truly incompatible with knowledge distillation: An empirical study. In: Proceedings of the 9th International Conference on Learning Representations. Virtual Event, Austria: ICLR, 2020. 1–17
- 29 Furlanello T, Lipton Z C, Tschanen M, Itti L, Anandkumar A. Born-again neural networks. In: Proceedings of the 35th International Conference on Machine Learning. Stockholm, Sweden: PMLR, 2018. 1602–1611
- 30 Romero A, Ballas N, Kahou S E, Chassang A, Gatta C, Bengio Y. FitNets: Hints for thin deep nets. In: Proceedings of the 3rd International Conference on Learning Representations. San Diego, USA: ICLR, 2015. 1–13
- 31 Lin T Y, Goyal P, Girshick R, He K M, Dollár P. Focal loss for dense object detection. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). Venice, Italy: IEEE, 2017. 2999–3007
- 32 Bengio Y, Louradour J, Collobert R, Weston J. Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning. Montreal, Canada: ACM, 2009. 41–48
- 33 Ericsson K A. Deliberate practice and acquisition of expert performance: A general overview. *Academic Emergency Medicine*, 2008, **15**(11): 988–994
- 34 Song W P, Shi C C, Xiao Z P, Duan Z J, Xu Y W, Zhang M, et al. AutoInt: Automatic feature interaction learning via self-attentive neural networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China: ACM, 2019. 1161–1170
- 35 Qin Y Q, Wang P F, Li C L. The world is binary: Contrastive learning for denoising next basket recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2021. 859–868
- 36 Huang Zhen-Hua, Yang Shun-Zhi, Lin Wei, Ni Juan, Sun Sheng-Li, Chen Yun-Wen, et al. Knowledge distillation: A survey. *Chinese Journal of Computers*, 2022, **45**(3): 624–653 (黄震华, 杨顺志, 林威, 倪娟, 孙圣力, 陈运文, 等. 知识蒸馏研究综述. 计算学报, 2022, **45**(3): 624–653)
- 37 Pan Rui-Dong, Kong Wei-Jian, Qi Jie. Legal judgment prediction based on pre-training model and knowledge distillation. *Control and Decision*, 2022, **37**(1): 67–76 (潘瑞东, 孔维健, 齐洁. 基于预训练模型与知识蒸馏的法律判决预测算法. 控制与决策, 2022, **37**(1): 67–76)
- 38 Zhao B R, Cui Q, Song R J, Qiu Y Y, Liang J J. Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New Orleans, USA: IEEE, 2022. 11943–11952
- 39 Zhang Y, Xiang T, Hospedales T M, Lu H C. Deep mutual learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 4320–4328
- 40 Zhao H J, Yang G, Wang D, Lu H C. Deep mutual learning for visual object tracking. *Pattern Recognition*, 2021, **112**: Article No. 107796
- 41 Chen D F, Mei J P, Wang C, Feng Y, Chen C. Online knowledge distillation with diverse peers. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York, USA: AAAI, 2020. 3430–3437
- 42 Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, et al. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, USA: Curran Associates Inc., 2017. 6000–6010
- 43 Toneva M, Sordoni A, des Combes R T, Trischler A, Bengio Y, Gordon G J. An empirical study of example forgetting during deep neural network learning. In: Proceedings of the 7th International Conference on Learning Representations. New Orleans, USA: ICLR, 2019. 1–19
- 44 Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the 13th International Conference on Artificial Intelligence and Statistics. Sardinia, Italy: JMLR, 2010. 249–256
- 45 He K M, Zhang X Y, Ren S Q, Sun J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). Santiago, Chile: IEEE, 2015. 1026–1034



胡开喜 武汉理工大学计算机与人工智能学院博士研究生。2018 年获得重庆大学控制工程硕士学位。主要研究方向为序列预测。

E-mail: issac\_hkx@whut.edu.cn

(HU Kai-Xi Ph.D. candidate at the School of Computer Science and Artificial Intelligence, Wuhan University of Technology. He received his master degree in control engineering from Chongqing University in 2018. His main research interest is sequential prediction.)



**李 琳** 武汉理工大学计算机与人工智能学院教授. 2009 年获得日本东京大学博士学位. 主要研究方向为信息检索, 推荐系统. 本文通信作者.

E-mail: cathylin@whut.edu.cn

**(LI Lin** Professor at the School of Computer Science and Artificial Intelligence, Wuhan University of Technology. She received her Ph.D. degree from University of Tokyo, Japan, in 2009. Her research interest covers information retrieval and recommender systems. Corresponding author of this paper.)



**解 庆** 武汉理工大学计算机与人工智能学院副教授. 2013 年获得澳大利亚昆士兰大学博士学位. 主要研究方向为流数据挖掘与模式分析.

E-mail: felixxq@whut.edu.cn

**(XIE Qing** Associate professor at the School of Computer Science and Artificial Intelligence, Wuhan University of Technology. He received his Ph.D. degree from University of Queensland, Australia, in 2013. His research interest covers streaming data mining and pattern analysis.)



**吴小华** 武汉理工大学计算机与人工智能学院博士研究生. 2019 年获得西北大学计算机科学与技术硕士学位. 主要研究方向为可解释机器学习.

E-mail: xhwu@whut.edu.cn

**(WU Xiao-Hua** Ph.D. candidate at the School of Computer Science and Artificial Intelligence, Wuhan University of Technology. He received his master degree in computer science and technology from Northwest University in 2019. His main research interest is explainable machine learning.)



**袁景凌** 武汉理工大学计算机与人工智能学院教授. 2004 年获得武汉理工大学博士学位. 主要研究方向为分布式并行计算.

E-mail: yjl@whut.edu.cn

**(YUAN Jing-Ling** Professor at the School of Computer Science and Artificial Intelligence, Wuhan University of Technology. She received her Ph.D. degree from Wuhan University of Technology, in 2004. Her main research interest is parallel distributed computing.)