

文章编号:1672-3961(2024)02-0069-11 DOI:10.6040/j.issn.1672-3961.0.2023.016

# 面向冷启动用户的元学习与图转移学习序列推荐

李璐, 张志军, 范钰敏, 王星, 袁卫华\*

(山东建筑大学计算机科学与技术学院, 山东 济南 250101)

**摘要:**为解决推荐系统用户冷启动问题,提出面向冷启动用户的元学习与图转移学习序列推荐(sequential recommendation for cold-start users with meta graph transitional learning, MetaGTL)。MetaGTL在不使用其他辅助信息的前提下,采用图神经网络(graph neural network, GNN)建模序列间物品高阶关系生成用户物品嵌入;将交互序列构造为物品对集合,使用序列编码模块捕捉物品间的转移关系,动态建模用户兴趣;采用注意力机制,生成准确的用户特征;采用基于梯度的元学习方法训练模型,生成初始化模型;对模型的工作性能和结果进行详细分析,结合基线模型进行对比评价。试验结果表明,基于元学习与图转移学习的MetaGTL在缺少辅助信息用户冷启动任务中具有更高的预测精度。

**关键词:**推荐系统;序列推荐;用户冷启动;图神经网络;元学习;深度学习

**中图分类号:**TP391.3 **文献标志码:**A

**引用格式:**李璐, 张志军, 范钰敏, 等. 面向冷启动用户的元学习与图转移学习序列推荐[J]. 山东大学学报(工学版), 2024, 54(2):69-79.

LI Lu, ZHANG Zhijun, FAN Yumin, et al. Sequential recommendation for cold-start users with meta graph transitional learning[J]. Journal of Shandong University (Engineering Science), 2024, 54(2):69-79.

## Sequential recommendation for cold-start users with meta graph transitional learning

LI Lu, ZHANG Zhijun, FAN Yumin, WANG Xing, YUAN Weihua\*

(School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, Shandong, China)

**Abstract:** To solve the cold-start problem for users of the recommendation systems, a sequential recommendation for cold-start users with meta graph transitional learning (MetaGTL) was proposed. MetaGTL used a graph neural network (GNN) to model higher order relationship between sequences of items to generate user embeddings and item embeddings without the use of auxiliary information. The sequence encoder constructed interacted sequences as sets of item pairs and captured transferred relationship among items. The user interest representation module used the attention mechanism to generate accurate user profile. The gradient-based meta-learning method was used to train the model to obtain an initialization model. The performance and result of the proposed approach were analyzed in detail through comparing with the baseline models. The experimental results showed that compared with other model methods, the proposed MetaGTL-based on meta-learning and graph transfer learning had higher prediction accuracy in user cold-start tasks without auxiliary information.

**Keywords:** recommendation system; sequence recommendation; user cold-start; graph neural network; meta-learning; deep learning

收稿日期:2023-01-11

基金项目:国家自然科学基金资助项目(61902221, 62177031);山东省自然科学基金资助项目(ZR2021MF099, ZR2022MF334);山东省教学改革研究项目(M2021130, M2022245, Z2022202);山东省优质专业学位教学案例库建设项目(SDYAL2022155);山东省重点研发计划(软科学项目)资助项目(2021RKY03056)

第一作者简介:李璐(1998—),男,山东潍坊人,硕士研究生,主要研究方向为推荐系统。E-mail:2238177039@qq.com

\*通信作者简介:袁卫华(1977—),女,山东青岛人,副教授,硕士生导师,博士,主要研究方向为推荐系统及机器学习。

E-mail:huahua\_qingdao@126.com

## 0 引言

随着在线信息不断增长,推荐系统已成为解决信息过载问题的重要工具。推荐算法主要分为基于协同过滤的推荐算法<sup>[1]</sup>、基于内容的推荐算法<sup>[2]</sup>和混合推荐算法<sup>[3]</sup>。在有大量用户物品交互数据参与训练的条件下,这些算法具有良好的性能。

序列推荐通过对用户行为序列建模、学习用户兴趣偏好对用户即将发生的行为进行预测<sup>[4]</sup>。随着深度学习在推荐领域的不断发展,序列推荐算法的性能也在逐步提高。当有冷启动用户加入时,因为缺乏足够的交互信息,序列推荐算法无法准确建模用户偏好,即序列推荐的用户冷启动问题<sup>[5]</sup>。冷启动问题是推荐领域的经典问题,严重影响推荐系统的性能<sup>[6]</sup>。解决用户冷启动问题不仅可以提高用户对系统的忠诚度和满意度,还可以使用户及时发现新商品,以提高产品销量<sup>[7]</sup>。

序列推荐中的用户冷启动问题与物品冷启动问题是推荐系统的重要研究方向<sup>[5]</sup>。目前,用户冷启动问题已被广泛研究<sup>[8]</sup>。研究者利用人口统计学信息<sup>[9]</sup>,计算用户的偏好或特征实现推荐。对于新物品,通过特征信息(如电影的场景、形式等)计算相似度,利用物品协同过滤算法进行推荐<sup>[10]</sup>。但在现实情况下,这些信息难以获得。用户可能会因保护隐私拒绝提供属性信息,新加入的物品可能难以给出准确有效的特征信息。近期,研究者利用元学习在少样本学习中的优势,将冷启动用户推荐视为一个新任务,利用训练任务中得到的先验经验,基于模型快速适应新任务,产生符合用户偏好的推荐列表<sup>[8]</sup>。目前,利用元学习解决推荐的冷启动问题已成为主流方法,基于梯度的元学习方法(model-agnostic meta-learning, MAML)是一种经典的模型无关元学习方法<sup>[11]</sup>,在各任务梯度矢量和方向上对参数进行优化,使模型沿梯度方向能够快速得到新任务的最优参数解。文献[6]提出基于元学习的冷启动序列推荐模型 Mecos 缓解序列推荐中的物品冷启动问题;文献[9]基于 MAML 提出解决用户冷启动问题的模型 MeLU;文献[12]为解决不同场景下的推荐问题,提出一个端到端的模型  $s^2$  meta。实践证明,

与基于深度学习的推荐模型相比,基于元学习的推荐模型在冷启动场景中性能表现更好<sup>[13]</sup>。由于许多基于元学习的方法在训练期间需要辅助信息或侧重于挖掘序列内物品的相关性并以静态方式建模用户兴趣,忽略了序列间物品相关性的影响。当辅助信息缺失或用户偏好短期内快速变化时,这些方法的性能会受到影响。因此,如何在不使用辅助信息的情况下充分挖掘用户/物品间的关系,动态建模用户兴趣,并能够快速适应交互项有限的冷启动用户推荐任务,是推荐系统迫切需要解决的问题之一。

基于上述分析,本研究提出面向冷启动用户的元学习与图转移学习序列推荐(sequential recommendation for cold-start users with meta graph transitional learning, MetaGTL),以解决上述问题。首先,基于用户交互序列构建用户-物品二部图,利用图神经网络建模序列间物品的高阶关系;其次,为了动态建模用户兴趣,设计序列编码模块捕获序列内物品间的转移关系,为每个用户生成特定的偏好表示。为了适应冷启动推荐任务,本研究使用元学习方法对模型进行训练,从交互数据丰富的用户中学习常见的行为模式,优化初始模型参数,使模型通过一次或几次梯度下降就可以准确建模冷启动用户的兴趣偏好,为其产生个性化推荐列表。

## 1 相关工作

### 1.1 序列推荐

序列推荐通过对用户行为序列建模,学习用户兴趣偏好,对用户行为进行预测。基于马尔可夫链的方法<sup>[14-15]</sup>,根据转移概率预测下一个可能与用户产生交互的物品。马尔可夫假设当前交互项仅取决于最近一个或几个交互,所以无法捕获序列间的长期依赖关系。随着深度学习的不断发展,研究者使用循环神经网络(recurrent neural network, RNN)<sup>[16-18]</sup>、卷积神经网络(convolutional neural networks, CNN)<sup>[19-21]</sup>、图神经网络(graph neural network, GNN)<sup>[22-27]</sup>等建模用户的动态偏好:GRU4Rec 模型不仅关注用户的最近行为,还关注用户的先前行为对目标行为的影响,以有效建模序列的长期依赖关系<sup>[18]</sup>;Caser 模型使用卷积滤波器预测用户可能交互的  $N$  个物品<sup>[19]</sup>;SR-GNN 模型将交互序列构造为图结构数据,利用 GNN 学习物品间

高阶关系<sup>[23]</sup>;DA-GCN模型将每个域中的用户和物品链接为图,设计了域感知的图卷积网络学习特定于用户的节点表示<sup>[24]</sup>;ISSR模型通过建模序列间物品相关性与序列内物品相关性生成序列嵌入表示<sup>[25]</sup>;LightGCN模型通过在用户-物品交互矩阵上进行线性传播学习用户和物品的嵌入,将学习到的所有层嵌入的加权和作为最终嵌入,在简化模型结构的同时保证预测的精度<sup>[26]</sup>;RL-ISN模型通过跨域推荐器和基于强化学习的域过滤器解决共享账户跨域序列推荐问题<sup>[27]</sup>;为了进一步提高推荐精度,SASRec模型<sup>[28]</sup>和NARM模型<sup>[29]</sup>将注意力机制应用到序列推荐中,提取序列中有价值的细粒度信息。上述方法虽然在一般场景中具有较好的推荐性能,但在冷启动场景中均无法准确建模冷启动用户偏好和物品特征表达。

### 1.2 冷启动推荐

早期的冷启动推荐解决方案利用辅助信息或领域知识建模用户偏好或物品特征表达。在基于内容的方法中,一些研究者通过提取用户或物品的属性特征进行推荐:文献[7]利用人口统计学信息对用户进行分组,使模型基于同组用户的评分对冷启动用户进行推荐;文献[30]结合基于潜在因子模型的协同过滤和概率主题进行建模,有效缓解了新物品的冷启动问题。在利用社交网络数据解决推荐系统冷启动问题的研究中,文献[31]利用社交网络中朋友的偏好数据为冷启动用户产生推荐列表;文献[32]利用Node2vec模型计算用户的网络邻居,利用网络邻居与物品的交互缓解冷启动问题。上述研究通过引入物品属性、人口统计学、社交网络等辅助信息缓解冷启动问题,忽略了挖掘用户-物品、物品-物品间深层次的交互关系,当这些辅助信息缺失时,其性能会大幅度降低。因此,对这些模型而言,数据稀疏性、辅助信息是否可用是解决冷启动问题的主要障碍。

### 1.3 元学习

一些工作将推荐系统的用户、物品推荐视为任务,把冷启动问题转换为元学习中的新任务自适应问题:文献[33]提出MetaCF模型,在交互数据有限的情况下学习协同过滤模型,更加准确建模用户兴趣,但缺乏捕获用户动态偏好的能力;文献[5]提出元转移学习模型MetaTL,仅使用有限交互的序列动态建模冷启动用户偏好。这些方法均未使用辅助信息,通过捕获序列内物品的相关性缓解冷启动问

题,但是忽略了不同序列间物品的相关性。由于序列间物品的相关性可以反映出用户的一般偏好并缓解数据稀疏问题<sup>[13]</sup>,基于此,本研究提出基于元学习训练框架的图转移学习序列推荐模型MetaGTL。MetaGTL不使用辅助信息,采用基于梯度的元学习方法训练模型,通过建模序列间的物品关系以及序列内物品的转移关系,结合注意力机制生成准确的用户特征,实现面向冷启动用户的个性化推荐。

## 2 问题定义

对于用户,推荐任务是一个二分类问题<sup>[31]</sup>。对于元学习,需要在冷启动问题的背景下制定每一项任务的定义。本研究将数据集中用户交互数量大于 $M$ 的用户视为一般用户 $u \in U_{reg}$ ,将交互数量小于 $M$ 且大于 $N$ 的用户视为冷启动用户 $u \in U_{new}$ 。

由于元学习策略的目标是学习一个对所有用户足够通用的推荐模型,并通过微调能够适应特定用户,因此,本研究将训练任务 $T_{train}$ 定义为 $U_{train} = \{u_1, u_2, \dots, u_n\}$ , $u_i \in U_{reg}$ ,从每个用户 $u$ 的交互序列中选择 $B_1$ 个连续物品集作为支持集 $S_{support}^u = \{v_1, v_2, \dots, v_{B_1}\}$ , $B_2$ 个物品作为查询集 $S_{query}^u = \{v_1, v_2, \dots, v_{B_2}\}$ 。

在训练任务中,使用支持集 $S_{support}$ 对模型进行训练,使用查询集 $S_{query}$ 对模型进行测试调整。在测试任务中,将冷启动用户的最后一个交互项作为冷启动测试集,利用其他交互项对模型进行训练,通过测试集微调。

## 3 方法

MetaGTL模型框架如图1所示,主要由以下4部分组成:图嵌入模块、序列编码模块、用户兴趣表示模块、元学习训练模块。图嵌入模块通过用户-物品二部图建模序列间物品的高阶关系<sup>[25]</sup>,其详细过程如图2所示;序列编码模块学习序列内物品的转移关系;用户兴趣表示模块利用注意力机制将序列间物品关系与序列内物品关系相融合,生成用户的动态兴趣表示;元学习训练模块对模型进行训练,利用在训练任务中学习到的先验知识,使模型快速适用冷启动推荐任务。

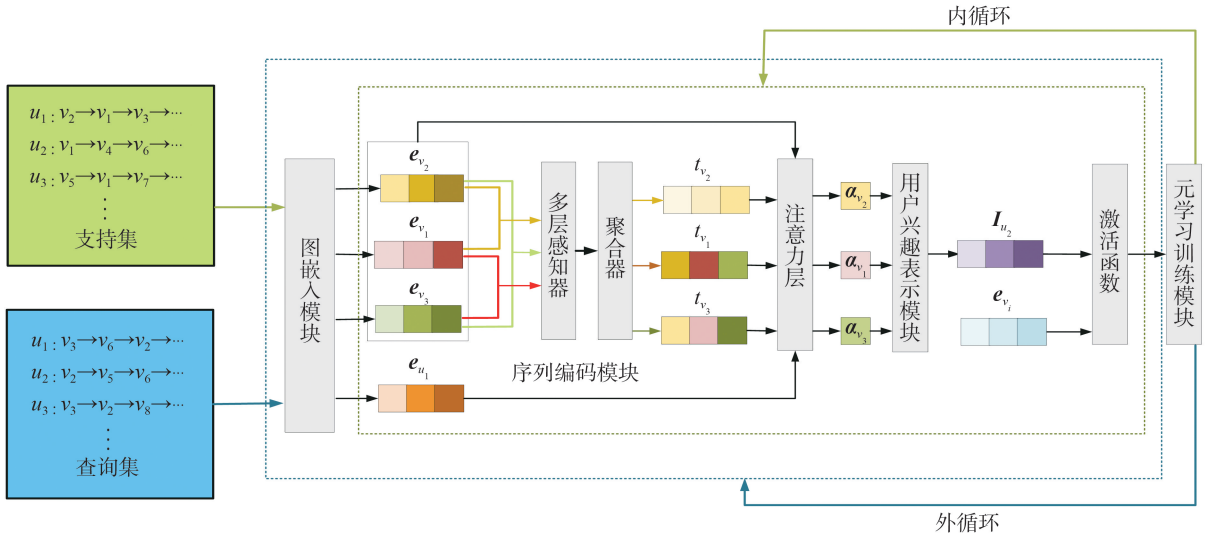


图1 MetaGTL模型框架

Fig.1 MetaGTLmodel framework

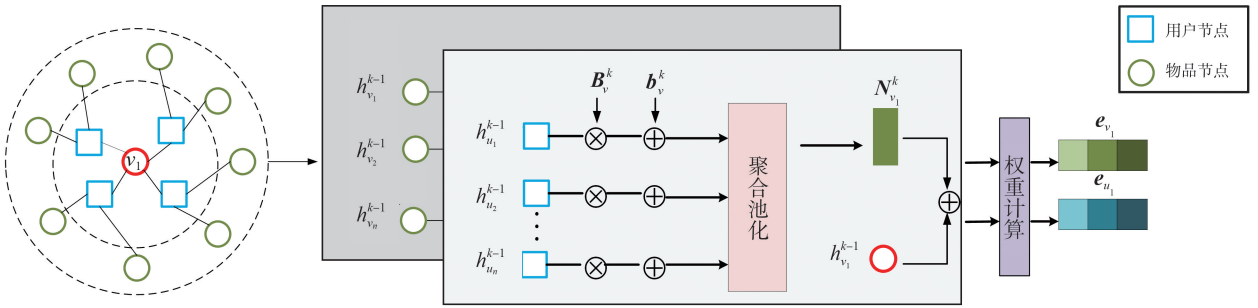


图2 图嵌入模块框架

Fig.2 Graph embedding module framework

### 3.1 通过图嵌入模块获取序列间物品相关性

随着图神经网络的不断发展,将交互序列建模为图结构数据,使用图神经网络捕获物品间的复杂关系,能够生成更加准确的用户及物品嵌入表达。为了解决冷启动推荐中模型只建模序列内物品的相关性而忽略了不同序列之间的物品相关性这一突出问题<sup>[5-6]</sup>,本研究将图神经网络与元学习结合,设计了图嵌入模块。利用序列间物品的关系丰富用户/物品表达,为冷启动用户生成准确的嵌入表达。本研究参照文献[25]将用户交互序列构造为用户-物品二部图,利用图神经网络对物品间高阶关系的优势建模,从不同的序列中收集信息,利用查询集  $\mathbf{S}_{\text{query}}$  在全局更新过程中为冷启动用户生成良好的初始嵌入,以提高模型对冷启动用户的推荐能力。

如图2所示,用户-物品二部图包含2种类型的节点,即用户节点和物品节点。如果用户与物品交互,则在用户和物品之间存在边。在图2中,以目标节点物品  $v_i \in \mathbf{V}$  为例,与  $v_i$  相邻的节点定义为  $n_{v_i}$ ,初始嵌入表示为  $\mathbf{x}_{v_i} \in \mathbf{R}^d$  (用户节点  $u_i$  的初始化嵌

入为  $\mathbf{x}_{u_i} \in \mathbf{R}^d$ ),对应的第  $k$  层隐表示为  $h_{v_i}^k \in \mathbf{R}^d$  (用户节点  $u_i$  的第  $k$  层的隐表示为  $h_{u_i}^k \in \mathbf{R}^d$ )。在图卷积网络,节点的特征表达既依赖于节点自身的信息,也需要聚合邻居的特征:

$$N_{v_i}^k = R_{\text{eLU}} \left( \frac{1}{|n_{v_i}|} \sum_{u \in n(v_i)} \mathbf{B}_{v_i}^k \cdot h_u^{k-1} + \mathbf{b}_{v_i}^k \right), \quad (1)$$

$$N_{u_i}^k = R_{\text{eLU}} \left( \frac{1}{|n_{u_i}|} \sum_{v \in n(u_i)} \mathbf{B}_{u_i}^k \cdot h_v^{k-1} + \mathbf{b}_{u_i}^k \right), \quad (2)$$

式中:  $N_{v_i}^k$  为在第  $k-1$  层的所有邻居上应用聚合函数得到第  $k$  层物品节点  $v_i$  的邻域表示;  $N_{u_i}^k$  为在第  $k-1$  层的所有邻居上应用聚合函数得到第  $k$  层用户节点  $u_i$  的邻域表示;  $\mathbf{B}_{v_i}^k, \mathbf{B}_{u_i}^k$  分别为第  $k$  层物品节点  $v_i$  和用户节点  $u_i$  的权重矩阵,  $\mathbf{B}_{v_i}^k, \mathbf{B}_{u_i}^k \in \mathbf{R}^d$ ;  $\mathbf{b}_{v_i}^k, \mathbf{b}_{u_i}^k$  分别为第  $k$  层物品节点  $v_i$  和用户节点  $u_i$  的偏置向量,  $\mathbf{b}_{v_i}^k, \mathbf{b}_{u_i}^k \in \mathbf{R}^d$ ;  $R_{\text{eLU}}$  为激活函数。

将式(1)(2)中的邻域信息与目标节点表示相结合作为该层的用户、物品嵌入表示:

$$h_{v_i}^k = \sigma \left( \mathbf{D}_{v_i}^k \cdot [h_{v_i}^{k-1}; N_{v_i}^{k-1}] + \mathbf{d}_{v_i}^k \right), \quad (3)$$

$$h_{u_i}^k = \sigma \left( D_u^k \cdot [h_{u_i}^{k-1}; N_{u_i}^{k-1}] + d_u^k \right), \quad (4)$$

式中:  $D_v^k, D_u^k$  为第  $k$  层物品节点和用户节点的变换权重矩阵,  $D_v^k, D_u^k \in \mathbf{R}^d$ ;  $d_v^k, d_u^k$  为第  $k$  层物品节点和用户节点的变换偏置向量,  $d_v^k, d_u^k \in \mathbf{R}^d$ ;  $\sigma$  为激活函数;  $[\cdot]$  为串联。

由于不同层的嵌入具有不同的语义,包含了序列间物品的高阶关系,为了使嵌入表达更加全面、丰富,本研究将不同层的嵌入表示经加权结合得到最终的用户、物品嵌入表示:

$$e_{v_i} = \sum_{k=1}^K \alpha_k \cdot h_{v_i}^{k-1}, \quad (5)$$

$$e_{u_i} = \sum_{k=1}^K \alpha_k \cdot h_{u_i}^{k-1}, \quad (6)$$

式中,  $\alpha_k$  为第  $k$  层的权重矩阵,用来表示在第  $k$  层嵌入的重要性。为了降低模型的训练复杂度,本研究参照文献[26]将  $\alpha_k$  中的元素统一设置为  $1/(K+1)$ ,其中  $K$  为图卷积网络的层数。为便于元学习模块描述,用  $\theta_g$  表示图嵌入模块的所有可训练参数(包括用户、物品的初始嵌入矩阵)。

### 3.2 通过序列编码模块获取序列内物品转移关系

对用户行为序列进行建模,捕获序列内物品的依赖关系,有利于学习用户的动态兴趣。在冷启动场景中,由于用户的交互物品过少,现有的序列推荐方法无法准确建模用户偏好。经图嵌入模块后,为了建模每个用户的动态偏好,提高模型在用户冷启动场景下的泛化能力,本研究设计了序列编码模块,利用节点之间的邻居关系得到更准确的用户、物品嵌入表示。在序列编码模块中利用 Transition 方法提取序列内物品的转移信息<sup>[5]</sup>,动态建模用户偏好。

在训练任务中,用户  $u$  的交互序列表示为  $S_{\text{support}}^u = \{e_{v_1}, e_{v_2}, \dots, e_{v_i}, \dots, e_{B_1}\}$ 。以  $S_{\text{support}}^u$  作为输入,将其中每个物品  $v_i$  与其之前的物品按交互顺序生成交互物品对集  $c_{v_i} = \{v_1 \rightarrow v_i, v_2 \rightarrow v_i, \dots, v_{i-1} \rightarrow v_i\}$ ,由于交互序列的第1个物品前面没有其他交互项,将其交互物品对集定义为  $c_{v_1} = \{v_1 \rightarrow v_1\}$ 。每个物品对通过  $L$  层的多层感知网络获得2个物品间的映射关系:

$$r^0 = e_{v_{i-1}} \oplus e_{v_i}, \quad (7)$$

$$r^l = R_{\text{cLU}} \left( W^l r^{l-1} + b^l \right), \quad (8)$$

$$R_{(v_{i-1}, v_i)} = W^l r^{l-1} + b^l, \quad (9)$$

式中:  $r^0$  表示将2个物品向量进行连接;  $r^l$  为第  $l$  层多层感知网络的输出结果;  $v_{i-1}$  与  $v_i$  分别为头实体和尾实体;  $l$  为多层感知网络的层数,  $l \in \{1, \dots, L\}$ ;  $W^l, b^l$  分别为第  $l$  层的权重矩阵和偏置向量,

$W^l \in \mathbf{R}^{d \times 2d}$ ,  $b^l \in \mathbf{R}^d$ ;  $R_{(v_{i-1}, v_i)}$  为物品对  $v_{i-1}$  与  $v_i$  的转移信息。

由于每个物品对集中存在多个物品对,对其进行聚合生成当前物品的物品转移信息

$$t_{v_i} = \frac{1}{|c_{v_i}|} \sum_{(v_{i-1} \rightarrow v_i) \in c_{v_i}} R_{(v_{i-1}, v_i)}. \quad (10)$$

### 3.3 基于注意力机制的用户兴趣表示模块

在基于注意力机制的用户兴趣表示模块中,本研究将序列间物品转移信息与序列内物品转移信息相结合,生成最终的用户兴趣表示。为了捕捉每个用户动态兴趣,本研究使用注意力网络为每个输入生成一个权重,用该权重表示用户当前行为对最终偏好的影响。通常情况下,用户交互序列中最后一项往往对预测交互项起重要作用<sup>[6]</sup>,本研究在生成注意力权重时考虑了最后交互项  $v_{k_1}$  的影响,物品  $v_i$  在用户  $u$  最终决策上的权重矩阵

$$\alpha_{uv_i} = \frac{\exp(\alpha'_{uv_i})}{\sum_{1 \leq i \leq B_1} \exp(\alpha'_{uv_i})}, \quad (11)$$

$$\alpha'_{uv_i} = \sigma W_1 (W_2 [e_u; e_{v_i} + W_3 t_{v_i}] + W_4 e_{v_{k_1}} + b_2) + b_1, \quad (12)$$

式中,  $W_1, W_2, W_3, W_4, b_1, b_2$  为多层感知网络的参数矩阵,  $W_1, W_2 \in \mathbf{R}^{d \times 2d}$ ,  $W_3, W_4 \in \mathbf{R}^{d \times d}$ ,  $b_1, b_2 \in \mathbf{R}^d$ 。

考虑序列内物品顺序相关性和时间动态的用户兴趣

$$I_u = \sum_{1 \leq i \leq B_1} \alpha_{uv_i} \cdot e_{v_i}, \quad (13)$$

式中  $B_1$  为支持集用户序列长度。

### 3.4 预测与训练

将用户  $u$  对物品  $v_i$  的预测得分  $p_{u,v}$  表示为用户兴趣  $I_u$  和物品嵌入  $e_{v_i}$  的内积,即:

$$p_{u,v} = S_{\text{oftmax}} (I_u^T, e_{v_i}), \quad (14)$$

式中  $S_{\text{oftmax}}$  为激活函数。

本研究利用贝叶斯个性化排序学习参数<sup>[34]</sup>,使用户观察到的交互项概率估计高于未观察到的交互项。损失函数

$$L_T = \sum_{(u, v, v^-) \in T_i} -\ln(S_{\text{igmoid}}(p_{u,v} - p_{u,v^-})) + \lambda \|\theta\|^2, \quad (15)$$

式中:  $S_{\text{igmoid}}$  为激活函数;  $T_i$  为第  $i$  个任务的训练样本,  $T_i = \{(u, v, v^-) \mid (u, v) \in \mathbf{R}^+, (u, v^-) \in \mathbf{R}^-\}$ ,其中  $\mathbf{R}^+$  为与用户产生交互的物品集合,  $\mathbf{R}^-$  为与用户未产生交互的物品集合;  $\lambda$  为正则化系数;  $\theta$  为模型所有可训练参数的统称。

### 3.5 元学习器

为了使模型能够快速适应冷启动推荐任务,本

研究使用基于梯度的元学习方法优化  $\theta_g$  和序列编码模块参数  $\theta_t$ 。

模型训练任务由内循环和外循环 2 个阶段组成,内循环用于优化用户兴趣建模,外循环通过元学习更新整个模型的参数。由于用户、物品表示是全局共享的,在内循环中不会更新物品的嵌入表示。内循环使用支持集  $S_{\text{support}}$  训练模型,通过梯度下降更新序列编码模块,参数更新公式为:

$$\theta'_t = \theta_t - \alpha \nabla_{\theta_t} L_{i \in D_s}(f_{\theta_g, \theta_t}), \quad (16)$$

式中,  $\alpha$  为内循环学习率,  $L_i$  为不同训练任务的损失,  $f$  为本研究提出的方法 MetaGTL。外循环优化过程使用查询集  $S_{\text{query}}$  优化模型的全部参数:

$$\theta_g \leftarrow \theta_g - \beta \nabla_{\theta_g} \sum_{i \in D_q} L_i(f_{\theta_g, \theta_t}), \quad (17)$$

$$\theta_t \leftarrow \theta_t - \beta \nabla_{\theta_t} \sum_{i \in D_q} L_i(f_{\theta_g, \theta_t}), \quad (18)$$

式中  $\beta$  为外循环学习率。经过元训练任务,可以得到用户、物品的良好初始嵌入和整个模型所需要的参数。元测试任务使用少量的冷启动用户数据对模型进行微调,可以快速适用冷启动用户的推荐任务。算法 1 列举了 MetaGTL 算法的步骤。

#### 算法 1 MetaGTL 算法

输入: 图嵌入初始化参数  $\theta_g$ 、序列编码模块初始化参数  $\theta_t$ 、任务集  $z(T)$ 。

输出:  $\theta_g$ 、 $\theta_t$ 、 $e_u$ 、 $e_v$ 。

(1) 划分任务  $T_i \sim z(T)$ ;

(2) for all  $T_i$  do

(3) 将  $T_i$  中用户连续交互的  $B_1$  个连续物品划分为支持集  $S_{\text{support}}$ ;

(4) 根据式(1)~(6)获得  $e_u$ 、 $e_v$ ;

(5) 将支持集物品序列划分为物品对集  $c_v$ ;

(6) 根据式(7)~(10)计算物品转移信息  $t_v$ ;

(7) 根据式(11)~(13)得到用户兴趣  $I_u$ ;

(8) 根据式(14)(15)计算损失;

(9) 根据式(16)元学习器使用支持集更新参数  $\theta_t$ ;

(10) end for

(11) 将  $T_i$  中用户连续交互的  $B_2$  个连续物品划分为查询集  $S_{\text{query}}$ ;

(12) 根据式(17)(18)使用查询集更新  $\theta_g$ 、 $\theta_t$ 、 $e_u$ 、 $e_v$ ;

(13) 返回  $\theta_g$ 、 $\theta_t$ 、 $e_u$ 、 $e_v$ 。

## 4 试验

为了验证 MetaGTL 的算法性能,本研究在 3 个

公开数据集上进行了一系列试验,并回答以下问题。

问题 1: 模型在冷启动场景下性能表现是否良好?

问题 2: 模型在不同数据量下的性能表现如何?

问题 3: 模型在不同图嵌入模块层数下的性能表现如何?

问题 4: 模型中的每个模块在冷启动场景中发挥了多大的作用?

### 4.1 数据集

本研究使用的 3 个数据集为 Amazon-Electronics、MovieLens-1M 和 Tmall。亚马逊产品数据集 Amazon 记录了用户对商品的评价和时间戳信息,具有高度稀疏性,是推荐系统常用的数据集,本研究使用其子集 Amazon-Electronics; MovieLens-1M 提供了电影领域的用户-物品交互数据,与文献[8]的工作相同,本研究将用户-物品评分高于阈值的设置为 1,其他的设置为 0;天猫数据集 Tmall 来自 IJCAI-15 比赛,由天猫平台中的用户行为日志构成。各数据集统计信息如表 1 所示。

表 1 数据集的统计信息  
Table 1 The statistics of the data set

数据集	物品数/个	用户数/个	交互数/条	密度/%
Amazon-Electronics	63 002	192 403	443 589	0.01
MovieLens-1M	6 040	3 952	575 281	2.41
Tmall	57 677	25 958	623 124	0.04

### 4.2 评价指标

为评估所提方法的性能,本研究在试验中采用 3 个常用的评估指标  $HR@N$ 、 $MRR@N$  和  $NDCG@N$ ,其中  $N$  取 10。 $HR@10$  表示推荐系统在前 10 个推荐位置中命中用户感兴趣的物品的概率, $MRR@10$  用于衡量推荐系统将用户最感兴趣的物品在前 10 个推荐位置的平均精确度, $NDCG@10$  用于衡量推荐列表中前 10 个物品的推荐相关性。

### 4.3 基线方法

(1) GRU4Rec:通过门控单元与循环神经网络捕捉序列中的长期依赖关系,记录早期行为对后期行为的影响。

(2) Caser:卷积序列嵌入模型,通过卷积滤波器学习用户的序列模式捕获用户偏好。

(3) SR-GNN:通过将会话序列建模为图形结构数据,通过注意力网络将物品的向量表征融合,得到用户的向量表达。

(4) SASRec:通过注意力网络计算用户历史行

为的权重,突出“相关”的历史行为,使预测结果更加准确。

(5) NARM:通过使用注意力机制的混合编码器,提取顺序行为信息和用户在当前会话中的主要意图。

(6) MetaTL:通过模拟冷启动场景,将 Transition 结构与元学习结合提取物品间的转移关系。

(7) MeLU:利用模型无关元学习方法,在各任务梯度矢量的方向上对参数进行优化,沿梯度方向快速得到新任务的最优参数解。

(8) MetaCF:以学习一个准确的协同过滤模型为目标,使模型能够在有限交互的情况下快速适应冷启动用户。

#### 4.4 参数设置

参照 MetaCF,在 MovieLens-1M 数据集中将交互数量大于 5 的用户视为普通用户,将交互数量为 3~5 的用户视为冷启动用户。为了模拟冷启动场景,对冷启动用户最多保留 5 个最早行为。在 Amazon-Electronics 和 Tmall 数据集中,将交互数量为 2~5 的用户视为冷启动用户,交互数量大

于 5 的视为普通用户。在元训练中,本研究将  $U_{train}$  的用户数设置为 15,即 15 个用户组成一个元任务。每个用户的支持集  $S_{support}^u$  用户数为 5,查询集  $S_{query}^u$  用户数为 15。

本研究使用 Python 3 中的 Pytorch 库运行模型<sup>[35]</sup>。通过对学习率进行网格搜索,设置内循环学习率  $\alpha = 10^{-4}$ ,外循环学习率  $\beta = 10^{-2}$ 。批量设置为 16,所有嵌入维度设置为 128。图卷积网络层数  $K$  设置为 2,MLP 网络层数  $L$  设置为 3<sup>[5]</sup>。对于所有的基线方法,模型参数按原文献进行设置。

#### 4.5 试验结果分析

##### 4.5.1 模型在冷启动场景下的性能表现

本节主要回答问题 1。解决用户冷启动场景下的推荐问题是本研究的主要任务。在此场景下,冷启动用户不参与模型训练,仅在测试阶段使用少量的冷启动用户行为对模型进行微调。表 2 展示了所有模型在 3 个数据集上的表现。通过与基线模型进行比较,本研究提出的 MetaGTL 在所有数据集、不同评价指标下都取得了最好的性能,证明了本研究提出模型能够有效缓解用户冷启动问题。

表 2 MetaGTL 与其他基线模型的性能比较

Table 2 Performance comparison between MetaGTL and the baseline methods

模型	Amazon-Electronics			Movielens-1M			Tmall		
	MRR@ 10	NDCG@ 10	HR@ 10	MRR@ 10	NDCG@ 10	HR@ 10	MRR@ 10	NDCG@ 10	HR@ 10
Caser	0.042	0.038	0.151	0.135	0.169	0.281	0.071	0.083	0.171
SASRec	0.052	0.035	0.076	0.175	0.205	0.341	0.062	0.071	0.131
GRU4Rec	0.069	0.046	0.136	0.201	0.238	0.406	0.252	0.221	0.341
NARM	0.071	0.075	0.156	0.232	0.268	0.426	0.281	0.241	0.376
SR-GNN	0.093	0.064	0.189	0.248	0.279	0.459	0.291	0.264	0.401
MeLU	0.251	0.202	0.285	0.288	0.318	0.485	0.329	0.302	0.426
MetaCF	0.303	0.214	0.347	0.368	0.335	0.511	0.358	0.324	0.455
MetaTL	0.341	0.278	0.363	0.389	0.377	0.564	0.371	0.365	0.471
MetaGTL	<b>0.391</b>	<b>0.305</b>	<b>0.381</b>	<b>0.401</b>	<b>0.435</b>	<b>0.602</b>	<b>0.394</b>	<b>0.389</b>	<b>0.493</b>

从表 2 可以看出,在各数据集上,不同基线模型的性能存在较明显的差异。在 Amazon-Electronics 数据集上,未使用元学习的模型(如 Caser、GRU4Rec 和 SASRec 等)性能表现都比较差,这说明在数据稀疏的情况下,冷启动问题会严重影响模型性能。其中,Case 在所有模型中性能最差,SASRec 和 NARM 模型虽然使用了注意力机制捕获用户偏好,但在冷启动场景下仍无法达到令人满意的性能。

在 Movielens-1M 数据集中,基线模型表现与 Amazon-Electronics 数据集上的表现一致,但 MRR@ 10、NDCG@ 10 和 HR@ 10 较其在 Amazon-Elec-

tronics 数据集略有提升,说明冷启动问题对模型的影响相对较小。

在 Tmall 数据集中,Caser 和 SASRec 与其他模型相比性能较差,因为这 2 个模型对用户的整个行为序列建模,不重视用户短期兴趣,在冷启动场景下难以达到较好的性能。GRU4Rec、NARM 侧重于对用户的短期偏好进行建模,其性能明显优于前 2 个模型。SR-GNN 模型通过使用图神经网络捕获物品间的高阶关系以生成更加准确的嵌入表达,在冷启动场景中也能有较好的表现。

MeLU、MetaCF、MetaTL 等基于元学习的模型性能明显优于其他的基线模型。MeLU 侧重于使用

辅助信息建模解决冷启动问题,因此在没有辅助信息的场景下表现性能较差。*MetaCF* 和 *MetaTL* 未使用辅助信息,在冷启动场景下利用局部和全局信息快速适应冷启动用户的特性,取得了较好的推荐效果。与 *MetaCF* 相比,*MetaGTL* 的性能又有较大的提升,这是因为该模型通过 GNN 建模序列间的物品关系,通过序列编码模块进一步挖掘序列内物品的转移关系,动态捕获用户兴趣变化,利用元学习方法对 *MetaGTL* 进行训练,使模型能够快速适应冷启动推荐任务,从而为冷启动用户提供高质量的推荐内容。

#### 4.5.2 训练数据量对模型性能的影响

本节主要回答问题 2。冷启动问题产生的主要原因是缺乏交互数据,*MetaGTL* 能否在减少训练数据使用量的场景下达到更好的性能呢?为了验证 *MetaGTL* 的有效性,将数据集划分为不同大小的训练数据,从原训练数据量的 10% 开始训练模型,对冷启动用户进行推荐测试,每次增加 20%,直至达到 100%。这里选取的模型包括未使用元学习框架的 *GRU4Rec*、*NARM* 和基于元学习的基线模型 *MetaTL*、*MetaGTL*。

表 3 为在 Tmall 数据集上 *MRR@10* 随训练数据量占比增加的变化趋势。由表 3 可以看出,当数据量占比过少时,所有方法的性能都不理想,而随着训练数据量占比增加,所有模型的性能都在提升。与其他 3 种算法相比,*MetaGTL* 在所有的训练子集上性能最优。当训练数据量占比为 10% 时,所有模型的性能不佳。当训练数据量占比增加至 30% 时,使用元学习方法的模型性能提升较大。随着训练数据量占比的不断增加,*GRU4Rec* 与 *NARM* 模型性能逐步提高,说明未使用元学习方法的模型需要充足训练数据才能有好的性能。

表 4 图嵌入模块层数对 *MetaGTL* 性能影响对比

Table 4 Comparison of the effect of graph embedding module layer

模型	Amazon-Electronics		MovieLens-1M		Tmall	
	NDCG@10	HR@10	NDCG@10	HR@10	NDCG@10	HR@10
<i>MetaGTL</i> -1	0.299	0.378	0.429	0.593	0.379	0.488
<i>MetaGTL</i> -2	0.305	0.381	0.435	0.602	0.389	0.493
<i>MetaGTL</i> -3	0.301	0.378	0.431	0.598	0.387	0.491
<i>MetaGTL</i> -4	0.281	0.352	0.402	0.569	0.354	0.469

#### 4.5.4 消融试验分析

本节主要回答问题 4。为了验证 *MetaGTL* 中的各个模块在冷启动场景中的有效性,本研究在 Electronics 数据集上基于 *MRR@10* 指标进行消融试验,试验结果如表 5 所示。

*MetaGTL*-G 为移除模型中的图嵌入模块。为

*MetaTL* 与 *MetaGTL* 在训练数据量占比达到 50%~70% 时性能提升显著,在训练数据量占比为 70%~100% 时性能提升幅度较小,这证明了使用元学习方法的 *MetaGTL* 可以在减少训练数据量的场景下更快地适应冷启动推荐任务。

表 3 Tmall 数据集中不同训练数据量场景下 *MRR@10* 的对比  
Table 3 *MRR@10* under different amount of training data in Tmall dataset

训练数据量占比/%	<i>GRU4Rec</i>	<i>NARM</i>	<i>MetaTL</i>	<i>MetaGTL</i>
10	0.076	0.081	0.092	0.102
30	0.112	0.123	0.165	0.185
50	0.144	0.162	0.207	0.309
70	0.173	0.198	0.302	0.339
90	0.195	0.235	0.357	0.381
100	0.211	0.241	0.365	0.389

#### 4.5.3 图嵌入模块层数对模型的影响

本节主要回答问题 3。为了研究图嵌入模块层数  $K$  对 *MetaGTL* 性能的影响, $K$  设置为  $\{1, 2, 3, 4\}$ ,相应的模型表示为 *MetaGTL*-1、*MetaGTL*-2、*MetaGTL*-3、*MetaGTL*-4,在 3 个数据集上的试验结果如表 4 所示。从表 4 中可以看出,增加图嵌入模块层数能够提高模型 *MetaGTL* 的性能,因为在消息传递的过程中通过聚合邻居信息更新节点表示,可以得到更好的用户偏好或物品特征表达。当图嵌入模块层数为 2 时,模型达到最佳性能;继续增加图嵌入模块层数,模型性能开始下降;当图嵌入模块层数为 4 时,模型性能下降较大,本研究将性能下降归因为 GNN 的过平滑问题<sup>[36]</sup>。当图嵌入模块层数不断增加时,多层图卷积网络会使兴趣不同的用户有相似的嵌入并引入更多噪声。试验结果表明,本研究将图嵌入模块层数设置为 2,以准确建模序列间物品相关性。

为了验证建模序列间物品相关性的作用,本研究使用通用的低维稠密嵌入代替模型中的图嵌入方法<sup>[37]</sup>。由表 5 可以看出,由于不使用图嵌入模块,*MetaGTL*-G 在 Electronics 数据集上的 *MRR@10* 有所下降,说明在冷启动场景下传统的嵌入方法受到数据稀疏问题的影响,无法生成准确的嵌入表达,

导致模型的预测能力下降,证明使用图神经网络挖掘序列间物品的高阶关系能够在冷启动场景下生

成准确的用户、物品嵌入表示,提高模型的预测精度。

表5 Electronics数据集上的消融研究  
Table 5 Ablation studies on Electronics dataset

模型	图嵌入模块	序列编码模块	元学习训练模块	MRR@10
MetaGTL-G	移除	使用	使用	0.354
MetaGTL-S	使用	移除	使用	0.336
MetaGTL-M	使用	使用	移除	0.308
MetaGTL	使用	使用	使用	0.391

MetaGTL-S为移除模型中的序列编码模块。为了验证提出的序列编码模块能否有效地建模用户的动态偏好,将移除序列编码后的公式表示为:

$$\alpha'_{uv_i} = \mathbf{W}_1 \sigma(\mathbf{W}_2 [\mathbf{e}_u; \mathbf{e}_v] + \mathbf{b}_2) + \mathbf{b}_1. \quad (19)$$

由表5可以看出,与MetaGTL相比,MetaGTL-S的性能有较大幅度下降,说明在冷启动场景下,由于用户交互序列过短,模型无法为用户生成准确的特征表示。而MetaGTL中序列编码器利用序列内物品的转移关系能够捕获用户的偏好变化情况,结合注意力机制,计算交互序列中物品的影响权重,为用户生成准确的偏好表示。即使在冷启动场景下,仍能动态捕获用户的兴趣变化并生成准确的特征表示。

MetaGTL-M为移除模型中的元学习训练模块。移除元学习训练模块后,本研究使用ADAM优化器<sup>[38]</sup>对模型进行训练。从表5中可以看到,去除元学习训练模块后,模型在冷启动推荐任务中的性能急剧下降,说明传统的模型训练方法无法使推荐模型适应冷启动推荐任务。使用元学习方法训练模型,能够提高模型对冷启动推荐任务的适应能力,为冷启动用户提供符合兴趣偏好的内容,解决传统训练方法无法快速适应冷启动推荐任务的问题。

## 5 结论

本研究提出了模型MetaGTL解决序列推荐中的冷启动问题。该方法能够建模序列间物品关系和序列内物品转移关系,结合注意力机制为冷启动用户生成兴趣表示。经过训练,MetaGTL能够快速为冷启动用户生成偏好表示,为推荐系统用户提供符合用户偏好的内容。在3个广泛使用的数据集上的大量试验证明,在交互数据稀疏的用户冷启动场景下MetaGTL推荐性能显著提高,并且能够减少训练模型时的数据量。本研究提出的MetaGTL能够适用常规推荐任务,克服交互数据稀疏造成的用户

冷启动问题。未来,将进一步挖掘用户交互序列中的信息,并考虑如何为新用户生成更加准确的推荐内容。

### 参考文献:

- [1] SCHAFFER J B, FRANKOWSKI D, HERLOCKER J, et al. Collaborative filtering recommender systems[M]. Berlin, Heidelberg: Springer, 2007: 291-324.
- [2] LIU Q, WU S, WANG D Y, et al. Context-aware sequential recommendation[C]//2016 IEEE 16th International Conference on Data Mining (ICDM). Barcelona, Spain: IEEE, 2016: 1053-1058.
- [3] CHEN W, NIU Z D, ZHAO X Y, et al. A hybrid recommendation algorithm adapted in e-learning environments[J]. World Wide Web, 2014, 17: 271-284.
- [4] XIE Y Q, ZHOU P L, KIM S. Decoupled side information fusion for sequential recommendation[C]//Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. Madrid, Spain: SIGIR, 2022: 1611-1621.
- [5] WANG J L, DING K Z, JAMES C. Sequential recommendation for cold-start users with meta transitional learning[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2021: 1783-1787.
- [6] ZHENG Y J, LIU S Y, LI Z K, et al. Cold-start sequential recommendation via meta learner[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, California, USA: AAAI, 2021, 35(5): 4706-4713.
- [7] LIKA B, KOLOMVATSOS K, HADJIEFTHYMIANES S. Facing the cold start problem in recommender systems[J]. Expert Systems with Applications, 2014, 41(4): 2065-2073.
- [8] HUANG X W, SANG J T, YU J. Learning to learn a cold-start sequential recommender[J]. ACM Transactions on Information Systems (TOIS), 2022, 40(2): 1-25.
- [9] LEE H, IM J, JANG S, et al. MeLU: meta-learned user preference estimator for cold-start recommendation[C]//

- Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: ACM, 2019: 1073-1082.
- [10] BOBADILLA J, ORTEGA F, HERNANDO A, et al. A collaborative filtering approach to mitigate the new user cold start problem[J]. Knowledge-Based Systems, 2012, 26: 225-238.
- [11] FINN C, ABBEEL P, LEVINE S. Model-agnostic meta-learning for fast adaptation of deep networks[C]//Proceedings of the 34th International Conference on Machine Learning. Sydney, Australia: PMLR, 2017: 1126-1135.
- [12] DU Z X, WANG X W, YANG H X, et al. Sequential scenario-specific meta learner for online recommendation [C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: ACM, 2019: 2895-2904.
- [13] BHARADHWAJ H. Meta-learning for user cold-start recommendation[C]//2019 International Joint Conference on Neural Networks (IJCNN). Budapest, Hungary: IEEE, 2019: 19028320.
- [14] CHENG C, YANG H Q, LYU M R, et al. Where you like to go next: successive point-of-interest recommendation[C]//Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence. Beijing, China: AAAI, 2013: 2605-2611.
- [15] RENDLE S, FREUDENTHALER C, SCHMIDT-THIEME L. Factorizing personalized Markov chains for next-basket recommendation[C]// Proceedings of the 19th International Conference on World Wide Web. New York, USA: ACM, 2010: 811-820.
- [16] MIKOLOV T, KARAFIÁT M, BURGET L, et al. Recurrent neural network based language model[C]//Proceedings of the 16th Annual Conference of the International Speech Communication Association, INTERSPEECH. Chiba, Japan: ISCA, 2010: 1045-1048.
- [17] ZHU Y, LI H, LIAO Y K, et al. What to do next: modeling user behaviors by time-LSTM[C]//Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne, Australia: AAAI, 2017: 3602-3608.
- [18] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks[EB/OL]. (2016-03-29) [2023-01-10]. <https://arxiv.org/abs/1511.06939>.
- [19] TANG J X, WANG K. Personalized top-N sequential recommendation via convolutional sequence embedding [C]//Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. New York, USA: ACM, 2018: 565-573.
- [20] YUAN F J, KARATZOGLOU A, ARAPAKIS I, et al. A simple convolutional generative network for next item recommendation[C]//Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. New York, USA: ACM, 2019: 582-590.
- [21] YUAN F J, HE X N, JIANG H C, et al. Future data helps training: modeling future contexts for session-based recommendation[C]//Proceedings of The Web Conference 2020 (WWW'20). New York, USA: ACM, 2020: 303-313.
- [22] WANG X, HE X N, WANG M, et al. Neural graph collaborative filtering[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2019: 165-174.
- [23] WU S, TANG Y Y, ZHU Y Q, et al. Session-based recommendation with graph neural networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto, California, USA: AAAI, 2019: 346-353.
- [24] GUO L, TANG L, CHEN T, et al. DA-GCN: a domain-aware attentive graph convolution network for shared-account cross-domain sequential recommendation [EB/OL]. (2021-05-07) [2023-01-10]. <https://arxiv.org/pdf/2105.03300>.
- [25] LIU F, LIU W W, LI X T, et al. Inter-sequence enhanced framework for personalized sequential recommendation[EB/OL]. (2020-04-29) [2023-01-10]. <https://arxiv.org/pdf/2004.12118>.
- [26] HE X N, DENG K, WANG X, et al. LightGCN: simplifying and powering graph convolution network for recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, USA: ACM, 2020: 639-648.
- [27] GUO L, ZHANG J Y, CHEN T, et al. Reinforcement learning-enhanced shared-account cross-domain sequential recommendation[J/OL]. IEEE Transactions on Knowledge and Data Engineering. (2022-06-21) [2023-01-10]. <https://ieeexplore.ieee.org/document/9802739>.
- [28] WANG C K, MCAULEY J. Self-attentive sequential recommendation[C]//2018 IEEE International Conference on Data Mining (ICDM). Singapore: IEEE, 2018: 197-206.
- [29] LI J, REN P J, CHEN Z M, et al. Neural attentive session-based recommendation[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. New York, USA: ACM, 2017: 1419-1428.

- [30] WANG C, BLEI D M. Collaborative topic modeling for recommending scientific articles[C]//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM, 2011: 448-456.
- [31] LIU F K, LEE H J. Use of social network information to enhance collaborative filtering performance[J]. Expert Systems with Applications, 2010, 37(7): 4772-4778.
- [32] ZHANG Y J, SHI Z K, ZUO W L, et al. Joint personalized Markov chains with social network embedding for cold-start recommendation[J]. Neurocomputing, 2020, 386: 208-220.
- [33] WEI T X, WU Z W, LI R R, et al. Fast adaptation for cold-start collaborative filtering with meta-learning[C]//2020 IEEE International Conference on Data Mining (ICDM). Sorrento, Italy: IEEE, 2020: 661-670.
- [34] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: bayesian personalized ranking from implicit feedback [EB/OL]. (2012-05-09) [2023-01-10]. <https://arxiv.org/pdf/1205.2618>.
- [35] PASZKE A, GROSS S, CHINTALA S, et al. Automatic differentiation in pytorch[EB/OL]. (2017-10-29) [2023-01-10]. <https://pytorch.org/docs/stable/notes/autograd.html>.
- [36] LIU F, CHENG Z Y, ZHU L, et al. Interest-aware message-passing gcn for recommendation[C]//Proceedings of the Web Conference 2021. New York, USA: ACM, 2021: 1296-1305.
- [37] HE M K, PAN W K, MING Z. BAR: behavior-aware recommendation for sequential heterogeneous one-class collaborative filtering[J]. Information Sciences, 2022, 608: 881-899.
- [38] KINGMA D P, BA J L. ADAM: a method for stochastic optimization [EB/OL]. (2017-01-30) [2023-01-10]. <https://arxiv.org/pdf/1412.6980>.

(编辑:孙亚彤)

(上接第68页)

- [20] WALKER R T. Geography, VonThünen, and Tobler's first law: tracing the evolution of a concept[J]. Geographical Review, 2022, 112(4): 591-607.
- [21] AN D, TONG X, LIU K, et al. Understanding the impact of built environment on metro ridership using open source in Shanghai[J]. Cities, 2019, 93: 177-187.
- [22] CHAI D, WANG L, YANG Q. Bike flow prediction with multi-graph convolutional networks[C]//Proceedings of the 26th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Seattle, USA: ACM, 2018: 397-400.
- [23] 黄皓,周丽华,黄亚群,等. 基于混合深度模型的虚假信息早期检测[J]. 山东大学学报(工学版), 2022, 52(4): 89-98.  
HUANG Hao, ZHOU Lihua, HUANG Yaqun, et al. Early detection of fake news based on hybrid deep model [J]. Journal of Shandong University (Engineering Science), 2022, 52(4): 89-98.
- [24] 孙鸿昌,周凤余,单明珠,等. 基于模式划分的空调能耗混合填补方法[J]. 山东大学学报(工学版), 2022, 52(1): 9-18.  
SUN Hongchang, ZHOU Fengyu, SHAN Mingzhu, et al. Mode division based hybrid filling method of air conditioning energy consumption[J]. Journal of Shandong University (Engineering Science), 2022, 52(1): 9-18.
- [25] 蒋阳升,俞高赏,胡路,等. 基于聚类站点客流公共特征的轨道交通车站精细分类[J]. 交通运输系统工程与信息, 2022, 22(4): 106-112.  
JIANG Yangsheng, YU Gaoshang, HU Lu, et al. Refined classification of urban rail transit stations based on clustered station's passenger traffic flow features[J]. Journal of Transportation Systems Engineering and Information Technology, 2022, 22(4): 106-112.

(编辑:孙亚彤)