

对跨域序列推荐模型 TJAPL 复现

摘要

在推荐系统领域,如何精准洞察用户的多领域交织的兴趣偏好,以及缓解单域数据稀疏问题,是关键研究问题,故跨域序列推荐应运而生。本报告聚焦跨域序列推荐模型 TJAPL 的复现研究。原论文通过目标域注意力偏好学习、跨域用户注意力偏好学习和跨域局部注意力偏好学习构建 TJAPL 模型以提升推荐效果。本复现采用新电商数据集 Rees46,严格按照时间划分数据,全候选集评估,采用两种评估方式(仅评估下一个和评估所有物品),并对比 SASRec 和 CD-SASRec 进行实验。结果显示 TJAPL 方法仅在个别领域稳定提升,其他领域 SASRec 表现较好。复现完整实现模型架构,包括目标域、跨域用户及局部注意力偏好学习等,验证原论文方法的同时,表明现有跨域模型需进一步创新,且改进后的实验设置使研究更严谨科学,为跨域序列推荐研究提供了更可靠的参考。

关键词: 跨域序列推荐; 注意力学习; 迁移学习; 数据划分; 评估策略

1 引言

在推荐应用逐渐广泛的同时,我们对可利用信息的认识也越来越深入,我们逐渐认识到更加广阔的信息相关。如用户在影视、书籍、音乐等不同领域的偏好存在着千丝万缕的联系,举例来说,喜爱观看科幻电影的用户,极有可能热衷于阅读科幻小说,同时也偏爱科幻主题的音乐。反观传统推荐系统,多数情况下仅依据单一领域的数据进行搭建,难以精准且全面地洞察用户在多个相关领域交织的兴趣偏好,跨域序列推荐这一概念便在这样的情境下顺势而生,力求深度融合多元信息,从而为用户呈上更为贴合心意、个性化十足的推荐内容。

从数据特性角度来看,单域数据存在严重的稀疏问题,以影视领域为例,小众类型影片或新上架作品的用户交互数据十分有限,基于这样稀疏的数据构建推荐模型,极易陷入过拟合困境,导致推荐的精准度大打折扣。而跨域序列推荐能够巧妙地整合影视、书籍、音乐等多领域数据,有效扩充数据量,缓解单域数据稀疏状况。

现越来越多的工作致力于跨域序列推荐算法的研究,本报告选取 2024 年在 Information Sciences 上最新发表的 TJAPL [11] 跨域序列算法进行复现。在本次复现中,实验应用在新数据集上采取更严谨的实验设置与评估方法探究跨域序列模型 TJAPL [11] 实际推荐效果。

2 相关工作

对于跨域序列推荐 [3],早期 π -Net [8] 及其改进版本 PSJNet [9] 在共享账户场景中采用 RNNs 建模用户行为序列并进行跨域知识转移。MIFN [7] 利用 GRUs 编码用户行为序列并结

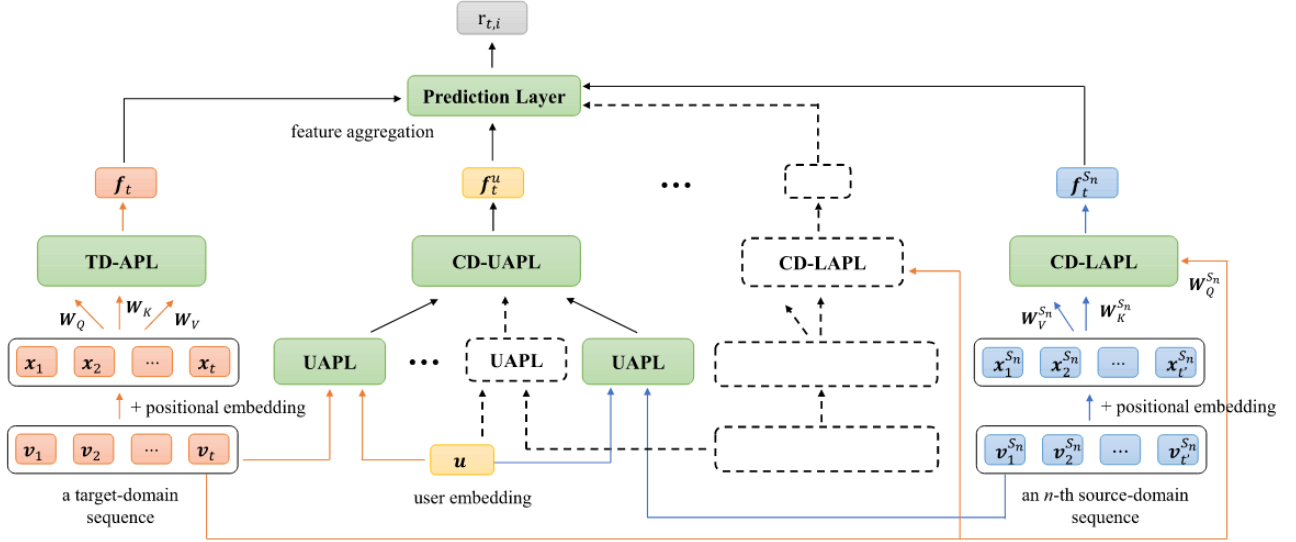


图 1. TJAPL 模型框架图 [11]

合知识图谱提升跨域知识转移效果。DA-GCN [4] 采用图神经网络（GNN）建模复杂交互关系和结构信息。CD-SASRec [1] 基于 SASRec 将源域聚合向量融入目标域项目嵌入以实现信息跨域转移，RecGURU [6] 通过对抗学习统一不同域用户嵌入。C²DSR [2] 将由 GNN 编码好的表征输入 Attention 编码器进行更深层次的建模。MGCL [10] 从用户角度和物品角度分别学习不同领域本地物品表征和全局物品表征。DREAM [12] 也提出用对比学习最小化不同领域序列间的用户偏好。然而，这些模型存在一些问题，如 RNNs 模型表达能力有限，部分模型忽略跨域项目转换模式，且多数仅考虑单源域，难以应用于多域场景。

3 本文方法

3.1 本文方法概述

本节根据论文介绍 TJAPL 模型架构，主要分为目标域注意力偏好学习（TD-APL），跨域用户注意力偏好学习（CD-UAPL）和跨域局部注意力偏好学习（CD - LAPL）。图 1 为该模型的框架图。

3.2 目标域注意力偏好学习（TD-APL）

本论文首先引入相关向量表示，将用户 u 在目标域序列 V 中的嵌入表示为 $\mathbf{u} \in \mathbb{R}^d$ ，其中 d 为潜在向量维度，这是一个可学习的向量。对于目标域序列 V_t ，其嵌入表示为 $\mathbf{V} = \{v_1, v_2, \dots, v_t\}$ ，源域序列 V^{S_n} 的嵌入表示为 $\mathbf{V}^{S_n} = \{v_1^{S_n}, v_2^{S_n}, \dots, v_{t'}^{S_n}\}$ 。由于自注意力模型本身无法直接考虑先前项目的位置信息，因此该方法添加一个可学习的位置嵌入 $\mathbf{P} = \{p_1, p_2, \dots, p_L\} \in \mathbb{R}^{L \times d}$ 到序列嵌入中。通过这种方式，该方法得到位置感知输入嵌入 $\mathbf{X} = \{x_1, x_2, \dots, x_t\}$ 和 $\mathbf{X}^{S_n} = \{x_1^{S_n}, x_2^{S_n}, \dots, x_{t'}^{S_n}\}$ ，其计算方式如下：

$$\begin{aligned} x_i &= v_i + p_i \\ x_i^{S_n} &= v_i^{S_n} + p_i \end{aligned}$$

接下来，该方法将序列 \mathbf{X} 输入到一系列堆叠的自注意力块 (SABs) 中。每个 SAB 可以看作是由一个自注意力层 (SAL) 和一个前馈网络 (FFN) 组成。

$$\alpha_i = \text{softmax}(x_t W_Q (x_i W_K)^T), \forall i \in \{1, 2, \dots, t\}$$

$$h_t = \sum_{i=1}^t \alpha_i (x_i W_V)$$

其中， $x_t W_Q$ 、 $x_i W_K$ 和 $x_i W_V$ 分别代表查询 (Query)、键 (Key) 和值 (Value)， W_Q 、 W_K 、 $W_V \in \mathbb{R}^{d \times d}$ 是可学习的参数，它们的引入增强了模型的灵活性。

在获得自注意力层的输出 h_t 后，该方法将其输入到一个两层的前馈网络 $FFN(h_t)$ 中，以使模型能够探索数据中的非线性特征。

$$f_t = \text{ReLU}(h_t W^{(1)} + b^{(1)}) W^{(2)} + b^{(2)}$$

其中， $W^{(1)}$ 、 $W^{(2)} \in \mathbb{R}^{d \times d}$ 和 $b^{(1)}$ 、 $b^{(2)} \in \mathbb{R}^d$ 是两层前馈网络的可学习参数。

堆叠多个自注意力块有助于模型提取更为复杂的顺序模式。该方法将第 b 个 ($b > 1$) 自注意力块表示为：

$$h_t^{(b)} = \text{SAL}(f_t^{(b-1)})$$

$$f_t^{(b)} = \text{FFN}(h_t^{(b)})$$

最终，该方法取顶层自注意力块的输出向量 $f_t^{(b)} \in \mathbb{R}^d$ 作为目标域注意力偏好 f_t 。

3.3 跨域用户注意力偏好学习 (CD-UAPL)

受到现有单域序列推荐研究中致力于识别用户长期偏好以生成用户一般兴趣的工作启发，该方法针对跨域序列推荐提出了一种全新的模块——跨域用户注意力偏好学习 (CD-UAPL)。

根据在目标域注意力偏好学习中引入的注意力机制，一种有效的方法是将可学习的用户向量 $\mathbf{u} \in \mathbb{R}^d$ (即用户 u 的嵌入) 作为注意力层中的查询。目标域用户注意力偏好可以通过以下方式进行形式化计算：

$$\beta_i = \text{softmax}(\mathbf{u} W_{Q_u} (v_i W_{K_u})^T), \forall i \in \{1, 2, \dots, t\}$$

$$z_t = \sum_{i=1}^t \beta_i (v_i W_{V_u})$$

其中， v_i 是项目 i 的初始嵌入， W_{Q_u} 、 W_{K_u} 、 $W_{V_u} \in \mathbb{R}^{d \times d}$ 是与公式 (3) 中类似的可学习参数， $\mathbf{u} W_{Q_u}$ 表示查询。 z_t 表示目标域用户注意力偏好，它代表了用户 u 在目标域截至时间步 t 的整体偏好。

考虑到同一用户在不同域中的行为往往反映出相似的偏好。此外，当用户在目标域中的行为数据高度稀疏时，仅依赖目标域内的交互可能无法准确捕捉用户的整体偏好。基于此，该方法将第 n 个源域中的用户注意力偏好形式化表示如下：

$$\beta_i^{S_n} = \text{softmax}(\mathbf{u} W_{Q_u}^{S_n} (v_i^{S_n} W_{K_u}^{S_n})^T), \forall i \in \{1, 2, \dots, t'\}$$

$$z_t^{S_n} = \sum_{i=1}^{t'} \beta_i^{S_n} (v_i^{S_n} W_{V_u}^{S_n})$$

其中, $v_i^{S_n}$ 是第 n 个源域序列的初始输入嵌入, $W_{Q_u}^{S_n}$ 、 $W_{K_u}^{S_n}$ 、 $W_{V_u}^{S_n} \in \mathbb{R}^{d \times d}$ 是可学习参数, $z_t^{S_n}$ 表示用户在第 n 个源域中的注意力偏好。

然后, 该方法将拼接后的向量 \mathbf{z} 输入到一个多层感知机 (MLP) 中, 通过 MLP 的学习能力, 得到最终的跨域用户注意力偏好表示 f_t^u , 计算过程如下:

$$f_t^u = \mathbf{z}W^{(u)} + b^{(u)}$$

其中, $W^{(u)} \in \mathbb{R}^{(1+N)d \times d}$ 和 $b^{(u)} \in \mathbb{R}^d$ 是 MLP 的可学习参数。该方法将最终输出的向量 $f_t^u \in \mathbb{R}^d$ 作为跨域用户注意力偏好, 它综合了用户在多个域中的偏好信息, 为跨域推荐提供了更全面的用户偏好表示。

3.4 跨域局部注意力偏好学习 (CD - LAPL)

考虑到用户在目标域中的下一次交互可能与他们在某个源域中最近交互的项目相关, 该方法提出了跨域局部注意力偏好学习 (CD - LAPL) 模块。

该方法将目标域中最后一个时间步的项目嵌入 v_t 作为查询, 将第 n 个源域序列的位置感知输入嵌入 \mathbf{X}^{S_n} 作为键和值, 从而构建跨域注意力层。该跨域注意力层的计算过程可以形式化表示为:

$$\alpha_i^{S_n} = \text{softmax}(v_t W_Q^{S_n} (x_i^{S_n} W_K^{S_n})^T), \forall i \in \{1, 2, \dots, t'\}$$

$$h_t^{S_n} = \sum_{i=1}^{t'} \alpha_i^{S_n} (x_i^{S_n} W_V^{S_n})$$

其中, $W_Q^{S_n}$ 、 $W_K^{S_n}$ 、 $W_V^{S_n} \in \mathbb{R}^{d \times d}$ 是可学习参数。与目标域注意力偏好学习类似, 该方法在跨域局部注意力偏好学习中也采用了一个两层的前馈网络 (FFN) 来进一步提升模型的性能, 其计算过程如下:

$$f_t^{S_n} = \text{ReLU}(h_t^{S_n} W^{S_n(1)} + b^{S_n(1)}) W^{S_n(2)} + b^{S_n(2)}$$

其中, $W^{S_n(1)}$ 、 $W^{S_n(2)} \in \mathbb{R}^{d \times d}$ 和 $b^{S_n(1)}$ 、 $b^{S_n(2)} \in \mathbb{R}^d$ 分别是两层前馈网络的权重和偏差。每个跨域注意力块可以看作是由一个跨域注意力层和一个后续的前馈网络组成。

3.5 预测

最后, 该方法将 TD - APL 的输出 f_t 、CD - UAPL 的输出 f_t^u 以及各个源域对应的 CD - LAPL 的输出 $f_t^{S_n}$ ($1 \leq n \leq N$) 进行拼接操作, 得到一个综合的特征向量 \mathbf{o} , 计算过程如下:

$$\mathbf{o} = \text{concat}[f_t, f_t^u, \dots, f_t^{S_N}]$$

其中, $\mathbf{o} \in \mathbb{R}^{(2+N)d}$ 表示拼接后的向量。然后, 该方法将拼接向量 \mathbf{o} 输入到一个多层感知机 (MLP) 中, 通过 MLP 的学习能力, 得到用户偏好的最终表示 o_t , 计算过程如下:

$$o_t = \mathbf{o}W^{(o)} + b^{(o)}$$

其中, $W^{(o)} \in \mathbb{R}^{(2+N)d \times d}$ 和 $b^{(o)} \in \mathbb{R}^d$ 是 MLP 的可学习参数, $o_t \in \mathbb{R}^d$ 表示用户偏好的最终表示。最后, 该方法可以通过计算用户偏好最终表示 o_t 与项目 i 的嵌入向量 v_i 的内积, 得到项目 i 在时间步 t 的预测得分 $r_{t,i}$, 计算过程如下:

$$r_{t,i} = o_t(v_i)^T$$

表 1. Rees46 数据集相关信息

	User_num	Item_num	Interaction_num	Aug_len	Density
Electronics		12716	458631	28.27	0.22%
Appliances	16223	11041	297288	18.33	0.17%
Computers		7473	314785	19.40	0.26%

通过上述预测层的设计，该方法能够综合考虑来自不同模块的信息，从而生成更准确的推荐预测结果，提高跨域序列推荐系统的性能。

4 复现细节

4.1 与已有开源代码对比

原文章代码基于 Amazon 留一法数据序列，并只部分候选集（100 个物品）中进行评估。为了使实验更加严谨，结果更加具有理论参考性，本次复现主要做出下列几个改进：

- **新电商数据集 Rees46¹**。其相较于 Amazon 评论数据集体现更多的行为偏好的及时性。
- **按时间划分数数据集**。相比于以往的留一法，存在时间泄露的问题，也就是在预测实际时间点之后的数据也会在前面的训练中被模型输入训练。所以在本次实验中，本论文采用更加严谨的按时间划分方法，本论文将所有交互记录按时间顺序排序，并按比例 8:1:1 划分数据，分别形成训练集、验证集和测试集。
- **全候选集评估**。在进行推荐时，将所有可能的项目都纳入候选范围，不进行预先筛选。这样的评估方式更具全面性和理论最优性。

5 数据集介绍

5.1 数据集信息

在这里，我没有采用原论文中的 Amazon²数据集，因为该数据集是由评论组成的交互记录，在预测用户偏好时交互记录并没有很好的及时性。所以，在此我采用一个新处理的数据集 Rees46³，其为电商数据集，其多为时间商品的浏览和购买记录序列。该数据集的相关信息如表 1。

5.2 数据集处理

按时间划分

因为以往的留一法，存在时间泄露的问题，也就是在预测实际时间点之后的数据也会在前面的训练中被模型输入训练。所以在本次实验中，本论文采用更加严谨的按时间划分方法，

¹<https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store>

²<http://jmcauley.ucsd.edu/data/amazon/>

³<https://www.kaggle.com/datasets/mkechinov/ecommerce-behavior-data-from-multi-category-store>

本论文将所有交互记录按时间顺序排序，并按比例 8:1:1 划分数据，分别形成训练集、验证集和测试集。

处理步骤

下面给出，本实验对于这个新数据集的处理步骤：

- (1) 读取文件，消除重复记录，仅保留最后一条记录。不区分行为类型，并将所有行为均视为隐式反馈。
- (2) 在单个域，删除出现次数少于 5 次的物品，以及交互物品少于 5 次的用户。
- (3) 删除非重叠用户。
- (4) 按时间 8: 1: 1 划分训练集、验证集、测试集。
- (5) 在验证集和测试集中删除冷启动物品和用户（即前面训练集中没有出现过的物品和用户）。
- (6) 得到最终的训练集、验证集、测试集数据文件。

5.3 评估方式改进

评估候选集

除此之外，现有的跨域序列方法均采用**部分候选集**（从整个物品集合中依特定规则选取部分项目作为候选，如基于用户历史行为、热门程度、相似度等筛选 100 个物品）去评估数据，这样虽然可以提高计算效率，但是评估范围过小，并不具备全面性。

所以在本次实验中采用**全候选集**，即在进行推荐时，将所有可能的项目都纳入候选范围，不进行预先筛选。这样的评估方式更具全面性和理论最优性。不会遗漏任何项目，能最大程度保证推荐结果的多样性，满足用户潜在的多样化需求。例如，用户兴趣发生突变或探索新领域时，全候选集更有可能发现并推荐相关项目。从理论上讲，在所有项目中进行选择能够找到全局最优的推荐结果，不局限于部分项目范围内的局部最优。

评估数量

在评估数量的方面，我们采用两种评估方式，其中一个为**只评估下一个物品**。根据前面本论文采用了按时间划分的数据处理方式，会形成验证集和测试集中不止一个物品的情况，如图 2。为了更加全面、严谨的评估，在本实验中也采用**评估验证集和测试集中所有物品**。

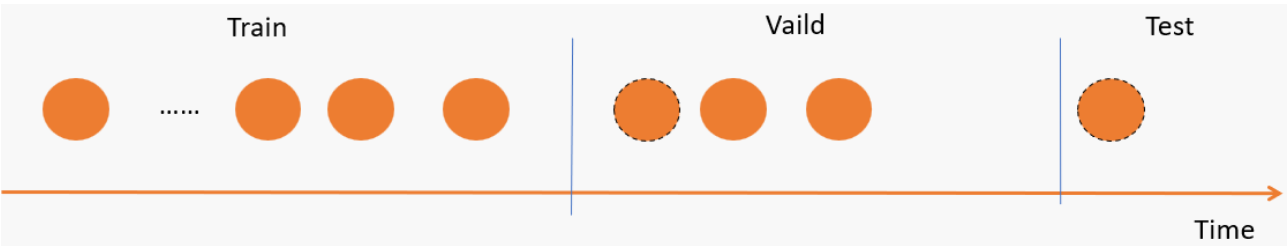


图 2. 按时间划分序列示意图

表 2. 实验结果表（只评估下一个物品）

Target domain	Models	Source domain	Recall@10	Recall@20	Recall@50	NDCG@10	NDCG@20	NDCG@50
Computers	SASRec		<u>0.131552</u>	0.190396	<u>0.291173</u>	<u>0.066068</u>	<u>0.080990</u>	<u>0.100919</u>
	CDSASRec	Appliances	0.130538	0.188367	0.289821	0.064941	0.079416	0.099553
		electronics	<u>0.13134</u>	<u>0.194827</u>	0.290561	0.064922	<u>0.080909</u>	0.099741
	TJAPL	Appliances	0.129185	0.189043	0.287792	0.065260	0.080401	0.099885
		Electronics	0.130668	<u>0.192140</u>	<u>0.291905</u>	<u>0.065270</u>	0.080830	<u>0.100521</u>
Appliances	SASRec		<u>0.108449</u>	<u>0.151009</u>	0.214061	<u>0.055043</u>	<u>0.065777</u>	<u>0.078237</u>
	CDSASRec	Computers	0.100883	0.145019	<u>0.214376</u>	0.052947	0.064094	<u>0.077934</u>
		Electronics	0.101721	0.147418	<u>0.222535</u>	0.046767	0.058379	0.073217
	TJAPL	Computers	<u>0.104035</u>	<u>0.149433</u>	0.213115	<u>0.053630</u>	<u>0.064973</u>	0.077614
		Electronics	0.101095	0.144914	0.205634	0.052795	0.063742	0.075854
Electronics	SASRec		<u>0.156634</u>	<u>0.226669</u>	<u>0.345908</u>	0.083649	0.101322	0.124981
	CDSASRec	Computers	0.151386	0.217648	0.332131	0.081521	0.098110	0.120714
		Appliances	0.148088	0.213142	0.328375	0.081227	0.097620	0.120496
	TJAPL	Computers	0.155650	0.226341	0.345744	<u>0.085679</u>	<u>0.103567</u>	<u>0.127244</u>
		Appliances	<u>0.165087</u>	<u>0.238150</u>	<u>0.357960</u>	<u>0.089998</u>	<u>0.108386</u>	<u>0.132157</u>

表 3. 实验结果表（评估测试集中所有物品）

Target domain	Models	Source domain	Recall@10	Recall@20	Recall@50	NDCG@10	NDCG@20	NDCG@50
Computers	SASRec		<u>0.118524</u>	<u>0.179257</u>	0.274081	<u>0.084271</u>	<u>0.102943</u>	0.129798
	CDSASRec	Appliances	0.115908	0.171197	0.269818	<u>0.083235</u>	0.100298	0.127653
		Electronics	0.115629	0.175794	<u>0.276594</u>	0.083196	<u>0.101563</u>	<u>0.129873</u>
	TJAPL	Appliances	0.116157	0.175642	0.273056	0.082080	0.100623	0.128336
		Electronics	<u>0.119009</u>	<u>0.177889</u>	<u>0.278899</u>	0.082910	0.101087	0.129491
Appliances	SASRec		0.090601	<u>0.134329</u>	0.191532	<u>0.066538</u>	<u>0.080038</u>	<u>0.096436</u>
	CDSASRec	Computers	<u>0.091521</u>	0.130562	<u>0.202347</u>	0.061016	0.073157	0.092964
		Electronics	0.088014	0.127571	0.189775	0.061459	0.073965	0.091555
	TJAPL	Computers	0.090264	0.131059	<u>0.195614</u>	0.064590	<u>0.077275</u>	<u>0.095362</u>
		Electronics	<u>0.094079</u>	<u>0.133247</u>	0.189019	<u>0.064815</u>	0.077010	0.093152
Electronics	SASRec		<u>0.128153</u>	<u>0.187290</u>	<u>0.296811</u>	<u>0.102216</u>	<u>0.119896</u>	<u>0.150729</u>
	CDSASRec	Computers	0.117547	0.172474	0.278803	0.093766	0.110228	0.140480
		Appliances	0.122378	0.179173	0.281867	0.097007	0.114011	0.143099
	TJAPL	Computers	<u>0.135093</u>	<u>0.194803</u>	<u>0.307672</u>	<u>0.106736</u>	<u>0.124590</u>	<u>0.156293</u>
		Appliances	0.126756	0.182431	0.292591	0.101335	0.118061	0.149187

5.4 实验细节

本次复现除了 TJAPL [11], 为了更有对比性, 也比较了两个基线方法, 经典序列推荐模型 SASRec [5] 和跨域序列模型 CD-SASRec [1]。

实验细节中, 本次复现将对结果影响比价大的注意力块数在 $\{1, 2, 3\}$ 内取值, 隐藏层数在 $\{64, 128\}$ 内取值, 进行调参。并在验证集中选定最好的参数 (值得注意的是, 超参数包括最好的迭代次数 epoch), 重新在测试集中训练得到测试集上的最终效果。随机种子为 2020, 学习率为 0.0001, 以及注意力头数为 1, Dropout 率为 0.5。其他的, 本次复现采用早停策略, 当 NDCG@20 在 20 次评估中均未提升, 则停止训练。

6 实验结果分析

表 2 和表 3 为复现的结果, 其中我们将指标的最好的前两个结果用下划线标注, 最好的结果用粗体表示。

根据实验结果发现, 在更加严谨的实验环境中, 跨域序列的方法仅在 Electronics 领域中有稳定的提升效果。在其他两个领域内 SASRec 的表现较好。猜测原因可能是, 现有跨域模型的效果并没有很好的表现出来, 需要更进一步的研究探索和创新。

7 总结与展望

在本次对跨域序列推荐模型 TJAPL 的复现研究中, 我们深入探讨了跨域序列推荐领域的相关工作, 并详细介绍了 TJAPL 模型的架构及其实验复现过程。采用新数据集 Rees46, 其行为偏好及时性更强, 数据处理工作繁杂, 包括多步清理与按时间划分等操作, 确保数据质量。数据集划分摒弃留一法, 采用更严谨的按时间排序 8:1:1 比例划分法。评估方式采用全候选集并评估所有物品, 全面且具理论最优性。实验对比发现跨域方法仅在 Electronics 领域提升明显, 其他领域 SASRec 表现佳, 表明现有跨域模型需改进。未来, 我们将优化模型, 分析跨域模型效果不佳原因, 改进结构算法, 如优化注意力机制以提升不同领域推荐性能, 推动跨域序列推荐技术发展, 为用户提供更精准个性化推荐服务。

参考文献

- [1] Nawaf Alharbi and Doina Caragea. Cross-domain self-attentive sequential recommendations. In *ICDSA*, 2022.
- [2] Jiangxia Cao, Xin Cong, Jiawei Sheng, Tingwen Liu, and Bin Wang. Contrastive cross-domain sequential recommendation. In *CIKM*, 2022.
- [3] Shu Chen, Zitao Xu, Weike Pan, Qiang Yang, and Zhong Ming. A survey on cross-domain sequential recommendation. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence*, pages 7989–7998, 2024.

- [4] Lei Guo, Hao Liu, Lei Zhu, Weili Guan, and Zhiyong Cheng. DA-DAN: A dual adversarial domain adaption network for unsupervised non-overlapping cross-domain recommendation. *TOIS*, 2023.
- [5] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *ICDM*, 2018.
- [6] Chenglin Li, Mingjun Zhao, Huanming Zhang, Chenyun Yu, Lei Cheng, Guoqiang Shu, Beibei Kong, and Di Niu. RecGURU: Adversarial learning of generalized user representations for cross-domain recommendation. In *WSDM*, 2022.
- [7] Muyang Ma, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Lifan Zhao, Peiyu Liu, Jun Ma, and Maarten de Rijke. Mixed information flow for cross-domain sequential recommendations. *TKDD*, 2022.
- [8] Muyang Ma, Pengjie Ren, Yujie Lin, Zhumin Chen, Jun Ma, and Maarten de Rijke. π -net: A parallel information-sharing network for shared-account cross-domain sequential recommendations. In *SIGIR*, 2019.
- [9] Wenchao Sun, Muyang Ma, Pengjie Ren, Yujie Lin, Zhumin Chen, Zhaochun Ren, Jun Ma, and Maarten de Rijke. Parallel split-join networks for shared account cross-domain sequential recommendations. *TKDE*, 2023.
- [10] Zitao Xu, Weike Pan, and Zhong Ming. A multi-view graph contrastive learning framework for cross-domain sequential recommendation. In *RecSys*, 2023.
- [11] Zitao Xu, Weike Pan, and Zhong Ming. Transfer learning in cross-domain sequential recommendation. *Information Sciences*, 2024.
- [12] Xiaoxin Ye, Yun Li, and Lina Yao. DREAM: Decoupled representation via extraction attention module and supervised contrastive learning for cross-domain sequential recommender. In *RecSys*, 2023.