

论文《Temporally and Distributionally Robust Optimization for Cold-Start Recommendation》 的复现报告

摘要

协同过滤 (CF) 推荐模型高度依赖于用户与项目之间的交互来学习 CF 表示, 因此在推荐冷启动项目方面存在不足。为了解决这个问题, 之前的研究主要引入项目特征 (如缩略图) 来推荐冷启动项目。他们在热启动条目上学习特征提取器, 使特征表征与交互相一致, 然后利用特征提取器提取冷启动条目的特征表征用于交互预测。遗憾的是, 冷启动项目的特征, 尤其是热门项目的特征, 往往会因时间特征偏移而与暖启动项目的特征相背离, 从而导致特征提取器无法准确学习冷启动项目的特征表示。

为了减轻时间特征偏移的影响, 我们考虑使用分布稳健优化 (DRO) 来增强特征提取器的生成能力。然而, 现有的 DRO 方法面临着一个不一致的问题: DRO 训练中强调的较差情况下的热启动项目可能与冷启动项目的分布不太一致。为了捕捉时间特征的变化并解决这一不一致性问题, 我们提出了一种具有新优化目标的新型时间 DRO, 即: 1) 整合最坏情况因子以提高最坏情况下的性能; 2) 设计一个变化因子以捕捉项目特征的变化趋势, 并加强对潜在人群的优化。

关键词: 冷启动推荐系统; 时序特征漂移; 分布鲁棒优化; 时序分布鲁棒优化

1 引言

在信息爆炸的时代，推荐系统在互联网中扮演着关键角色，其核心目的是帮助用户从海量信息中筛选出符合自身需求的内容，协同过滤（CF）作为其中最具代表性的方法 [1]，通过分析用户与物品的历史交互数据来学习 CF 表示，进而预测用户偏好。然而，随着内容创作能力的飞速提升，新物品（冷启动物品）如潮水般涌入推荐系统，每分钟就有大量视频上传至 YouTube 等平台，这使得传统 CF 方法在冷启动推荐场景中面临巨大挑战。

由于冷启动物品缺乏历史交互数据，传统 CF 方法无法有效为其提供推荐，导致推荐系统在物品侧的生态平衡被打破。为解决这一问题，先前研究尝试引入物品特征（如类别、缩略图等）来进行冷启动推荐 [2]，主要思路是在温暖物品上学习特征提取器，以使其特征表示与交互相匹配，进而用于冷启动物品的特征提取和交互预测。对于冷项目的推理，鉴于缺乏 CF 对应物，仅使用来自特征提取器的特征表示来估计用户偏好。这种范式的关键在于设计训练策略来调整特征表示和用户-项目交互，主要分为 2 种方法。1) 基于鲁棒训练的方法 [3]，同时使用特征表示和 CF 表示来预测相互作用。2) 基于辅助损失的方法 [4]，例如对比损失 [5] 和 GAN 损失 [6]。

但不幸的是，新的问题随之而来，如图 1 所示：冷启动物品的特征，尤其是热门冷启动物品的特征，会因各种环境因素（如疫情爆发）随时间发生变化，与温暖物品的特征逐渐产生差异。这使得基于经验风险最小化（ERM）训练的特征提取器 [7] 难以准确捕捉冷启动物品的特征表示，最终导致冷启动推荐性能下降。

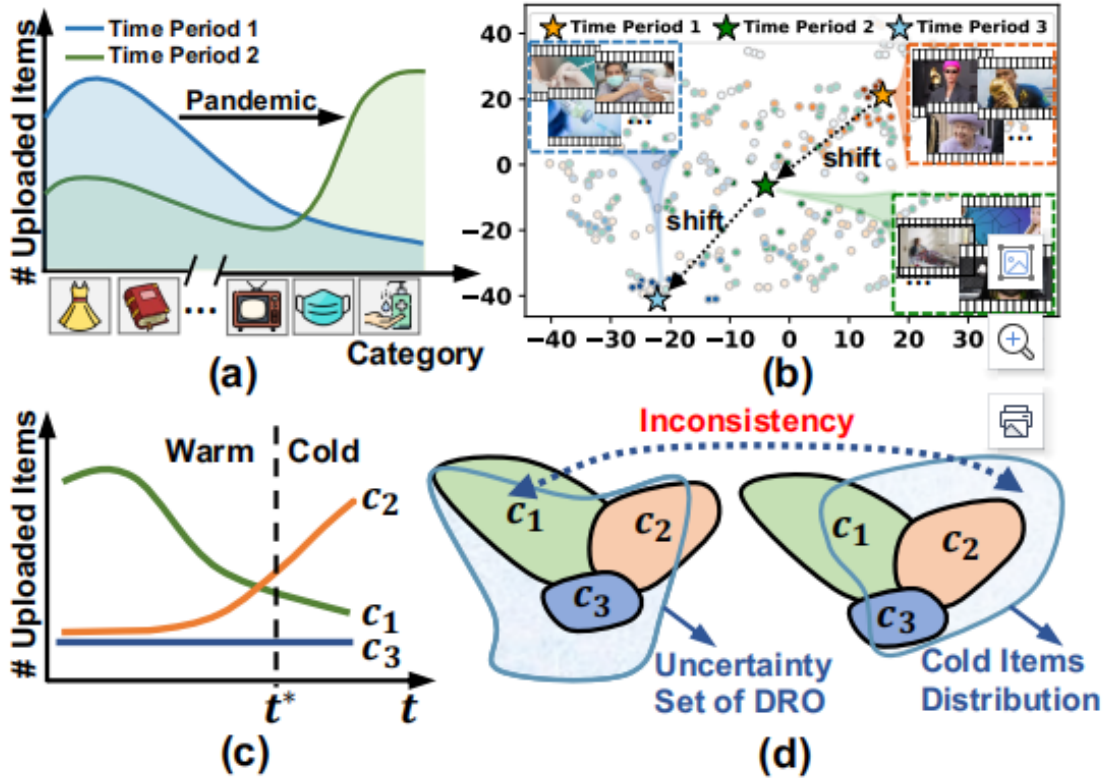


图 1. 冷启动推荐性能下降案例

为增强模型对时间特征转移的适应能力，分布鲁棒性优化（DRO）方法被考虑应用于冷

启动推荐。DRO 旨在通过优化最坏情况性能 [8] 来应对分布变化，但直接应用于冷启动推荐时，却出现了不一致问题。DRO 往往会过度关注温暖物品中的少数群体 [9]，而这些少数群体在后续冷启动物品中不一定能保持其受欢迎程度，这就可能削弱对冷启动物品中潜在受欢迎群体的优化，从而限制了模型在冷启动推荐中的性能提升。

在本文中，作者为解决冷启动推荐问题在特征提取与分布鲁棒性优化方面面临的严峻挑战。提出了一种名为 TDRO 的创新方法，考虑到项目特征的时间移动趋势以进行冷启动推荐。TDRO 主要优化点包括两个因素：1) 保证最坏情况性能的最坏情况因素，作者根据项目特征的相似性将温暖的项目分组，并优先考虑训练损失较大的项目组的改进；2) 一个捕捉项目特征变化趋势的转移因子，利用基于梯度的策略来强调对跨时间段逐渐流行的项目组的优化。最后作者在两种最先进的 (SOTA) 冷启动推荐方法上实例化了 TDRO，并在三个真实数据集上进行了广泛的实验。

2 相关工作

在冷启动推荐与分布鲁棒性优化相关的研究领域中，前人已经开展了诸多工作，这些成果既为后续研究提供了参考，也暴露出一些有待解决的问题。

针对冷启动推荐这一难题，传统的协同过滤 (CF) 方法由于依赖历史交互数据来学习 CF 表示 [10]，在冷启动物品缺乏此类数据时便难以发挥作用。为克服这一障碍，已有研究尝试从引入物品特征入手改进推荐效果 [11]。在这些研究中，基于鲁棒训练的方法试图利用特征和 CF 表示共同预测交互，并且通过随机改变 CF 表示来强化特征与交互之间的一致性。比如，DUIF [12] 直接抛弃 CF 表示，仅依赖特征表示与交互进行匹配；DropoutNet [3] 在训练交互预测器时，随机去除 CF 表示，以此增强特征与交互的关联性；M2TRec [2] 作为基于顺序的模型，整合了 Transformer 编码器来学习特征表示，且在训练和推理阶段都不使用 CF 表示；MTPR [13] 则采用多任务方式，用零向量替换部分暖项目 CF 表示以实现稳健学习。然而，这些方法在处理冷启动物品特征随时间变化的问题上存在明显不足，难以精准捕捉冷启动物品的特征变化，进而影响了推荐的准确性和效果。

辅助损失法致力于通过引入各种辅助损失，缩小特征与 CF 表示之间的距离，从而实现二者的匹配。例如，Heater [4] 运用专家混合模型提取个性化特征表示，并通过降低与预训练 CF 表示的距离来实现对齐；CB2CF [14] 引入通用特征提取器，利用 MSE 损失促使特征与 CF 表示相契合；CCFCRec [15] 通过最小化共现暖项目特征表示的距离，学习更为可靠的特征；CLCRec [5] 借助对比损失最大化特征与 CF 表示间的互信息，有效提升了二者的对齐程度；GAR [6] 采用对抗训练手段，有效减小了特征与 CF 表示之间的分布差异。即便如此，这些方法仍然无法完全消除时间特征转移带来的不利影响，冷启动推荐的性能仍有较大的提升空间。另外，不变学习方法（如 InvRL）尝试通过捕捉不变特征表示来实现稳健的交互预测，但却忽略了特征中的可变部分，而这部分对于预测温暖物品的交互可能非常关键。在分布鲁棒性优化 (DRO) 方面，其目的在于通过优化最坏情况性能，应对数据分布的变化。基于差异的 DRO 将不确定性集定义为围绕训练分布的区域，通过采用不同的差异度量（如 f-divergence、Wasserstein Distance [16] 等）来界定该区域，以此增强模型对分布变化的适应能力。不过，这种定义方式容易引发过度悲观的问题，即模型可能过度考虑不合理的分布情况，从而限制了其在实际应用中的实用性。Group - DRO [17] 将训练分布视为预定义 K 组的组

合，不确定性集定义为子群组合集，促使 DRO 关注有意义的分布变化。例如，在自然语言建模中，它可用于追踪主题间的分布变化；在推荐系统中，可优化不同用户组的最坏情况性能。但在直接应用于冷启动推荐时，仍存在与冷启动物品实际分布不相符的问题，无法有效适应冷启动场景下的分布变化。

除了上述主要方法外，还有其他一些旨在增强鲁棒性的方法被广泛探讨。例如不变学习方法，重加权策略等，具体建议感兴趣的读者参考优秀的调查以了解更多的细节。

3 本文方法

本文提出了 TDRO 应对冷启动推荐面临的挑战，TDRO 在 DRO 训练过程中引入两个关键目标：一是着重优化少数群体的最坏情况性能，确保模型在不同分布下具备较高的性能下限；二是精准捕捉物品特征的时间转移趋势，将优化重点置于可能在冷启动物品中流行的群体，从而提升模型在冷启动推荐中的整体性能和泛化能力。

3.1 分布鲁棒性优化 (DRO)

DRO 旨在应对数据分布变化，优化最坏情况性能。传统 DRO 在冷启动推荐中面临问题，本文采用 Group - DRO 避免过度悲观问题。Group - DRO 将训练分布视为 K 个预定义组的混合 $P = \sum_{i=1}^K w_i P(i)$ ，其不确定性集定义为子群混合集 ($Q := \left\{ \sum_{i=1}^K w'_i P'_i : w'_i \in \Delta_K \right\}$ ，其中 Δ_K 是 $K-1$ 维概率单纯形)。其目标是优化最坏情况性能，公式如下：

$$\theta_{DRO}^* = \arg \min_{\theta \in \Theta} \left\{ \max_{j \in [K]} \mathbb{E}_{(u_i, y_{ui}) \sim \tilde{P}_j} [\mathcal{L}(\theta; (u_i, y_{ui}, s_i))] \right\} \quad (1)$$

通过这种方式，DRO 在一定程度上能够提升模型对分布变化的适应性，但仍需进一步改进以适应冷启动推荐的特殊需求。

3.2 分组选择策略

3.2.1 基于特征聚类与时间划分

基于物品特征（如视觉特征），运用 K - means 聚类算法将温暖物品划分为 K 组（ K 值因数据集而异，例如在 Amazon 数据集上，对于 CLCRec 骨干模型，经实验验证 $K=5$ 最佳）。同时，按时间顺序把用户-物品交互数据划分为 E 个时间段（在 Amazon 数据集上，最佳 $E=5$ ），综合因子选择优化组。

在每次模型更新时，综合最坏情况因子和转移因子来选择优化组。具体公式如下：

$$j^* = \arg \min_{j \in \{1, \dots, K\}} - (1 - \lambda) \bar{\mathcal{L}}_j(\theta^t) + \lambda \sum_{e=1}^E \sum_{i=1}^K \beta_e \mathcal{L}_i^e(\theta^t - \eta \nabla_{\theta} \bar{\mathcal{L}}_j(\theta^t)) \quad (2)$$

其中 λ 平衡两个因子强度（在 Amazon 数据集上，对于 CLCRec 骨干模型，最佳 $\lambda=0.3$ ）。

最坏情况因子关注损失较大的少数群体，通过计算每个组的平均损失 $\bar{\mathcal{L}}_j(\theta^t)$ 进行组选择，损失越大的组越可能被选中（因为其 $-\bar{\mathcal{L}}_j(\theta^t)$ 越小）。

转移因子从两个角度考虑：一是强调冷启动物品中潜在受欢迎的群体，通过 $\beta_e = \exp(p \cdot e)$ 对后期时间段（更接近测试阶段）进行加权，其中 p 为控制陡峭度的超参数（在 Amazon 数据

集上，最佳 $p=0.2$)；二是通过计算转移趋势，即所有组在所有时间段内的梯度加权平均值，并计算组梯度 ($g_j = \eta \nabla_{\theta} \bar{\mathcal{L}}_j(\theta^t)$) 与转移趋势的内积，以此衡量组与潜在流行趋势的相似性，进而选择优化组。为提高计算效率，可通过一阶泰勒展开将公式 (2) 近似为基于梯度的形式：

$$j^* = \arg \max_{j \in \{1, \dots, K\}} (1 - \lambda) \bar{\mathcal{L}}_j(\theta^t) + \lambda \left\langle g_j, \sum_{e=1}^E \sum_{i=1}^K \beta_e g_i^e \right\rangle \quad (3)$$

3.2.2 梯度平滑技术

鉴于逐步优化策略在推荐系统中容易导致训练不稳定的问题，TDRO 从组重要性平滑和损失一致性增强两个方面对梯度进行优化。

组重要性平滑

通过为组分配权重向量 w 并动态调节其值来实现。具体公式如下：

$$w^{t+1} = \arg \max_{w_i \in [K]} \sum_i w_i \left[(1 - \lambda) \bar{\mathcal{L}}_i(\theta) + \lambda \left\langle g_i, \sum_{e=1}^E \sum_{j=1}^K \beta_e g_j^e \right\rangle \right] - \frac{1}{\eta_w} KL(w, w^t) \quad (4)$$

其中 w_i 是 w 的第 i 个元素， η 是学习率， $KL(p, q) = \sum_i p_i \log \frac{p_i}{q_i}$ 是 p 和 q 之间的 KL 散度。通过应用 KKT 条件，可以得到其封闭形式解：

$$w_i^{t+1} = \frac{w_i^t \exp \left(\eta_w \left[(1 - \lambda) \bar{\mathcal{L}}_i(\theta^t) + \lambda \left\langle g_i, \sum_{e=1}^E \sum_{j=1}^K \beta_e g_j^e \right\rangle \right] \right)}{\sum_s w_s^t \exp \left(\eta_w \left[(1 - \lambda) \bar{\mathcal{L}}_s(\theta^t) + \lambda \left\langle g_s, \sum_{e=1}^E \sum_{j=1}^K \beta_e g_j^e \right\rangle \right] \right)} \quad (5)$$

然后根据计算得到的权重更新模型参数：

$$\theta^{t+1} = \theta^t - \eta \sum_i w_i^{t+1} \nabla \bar{\mathcal{L}}_i(\theta^t) \quad (6)$$

损失一致性增强

为缓解数据稀疏性加剧导致的训练不稳定，采用保持经验损失的流式估计方法，其公式如下：

$$\bar{\mathcal{L}}_j^t \leftarrow (1 - \mu) \bar{\mathcal{L}}_j^{t-1} + \mu \bar{\mathcal{L}}_j^t \quad (7)$$

其中 μ 是控制流式步长的超参数，较小的 μ 会产生更保守的训练。通过这种方式，TDRO 能够在训练过程中更好地适应数据分布的变化，提高模型的稳定性和泛化能力。

4 复现细节

4.1 与已有开源代码对比

本报告实现的 TDRO 模型作者已公开源代码，因此报告的工作是建立在原代码基础的。

TDRO 源码：<https://github.com/Linxyhaha/TDRO>

在本文中实现的工作主要有三个：

对于数据集进行处理，原数据集作者没有给出来源，只有处理后的数据集，因此对.npy 格式的数据进行转化成.csv 格式，了解数据集的相关特征，并对数据集进行的一定的修改。

采用基于对比学习的冷启动推荐框架 CLCRec 作为基础模型，模型分成 2 部分，第一种是用户与交互物品的协同表示的对比学习对构造 (U-I)，让用户与有交互的物品协同表示更接近，使得模型在训练过程中能够通过对正样本中的协同信号与负样本中的差异，学习到用户对不同物品的偏好模式。第二种是物品之间的对比学习对构造，有交互的物品的协同表示和自身的特征表示更接近。其结构图如图2所示：

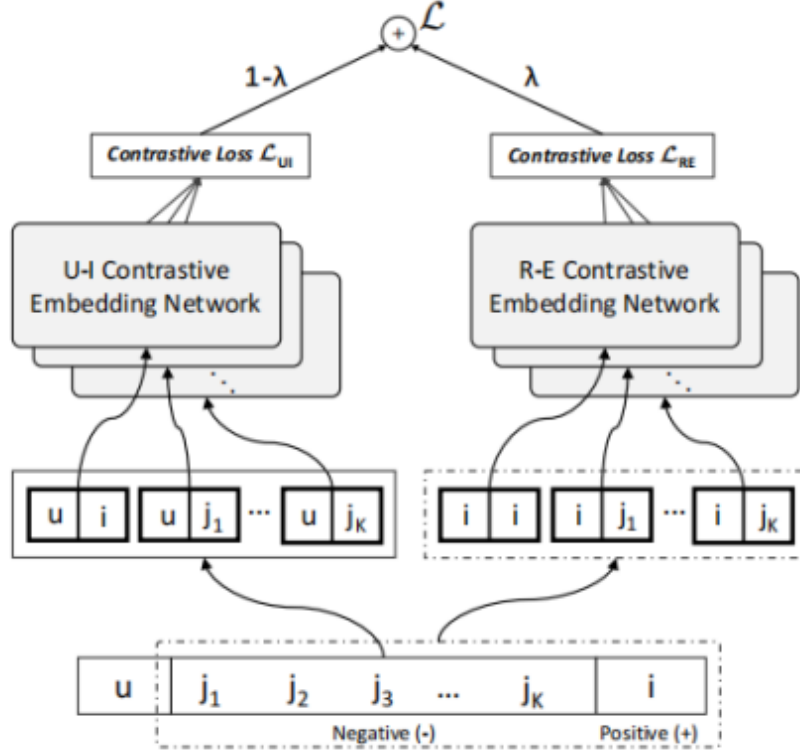


图 2. CLCRec 结构图

基于 CLCRec 获取对比损失，同时加入正则化损失得到分组损失。对最坏情况因子和转移因子进行调参，使其能选择最优组进行优化，同时调整超参数平衡强度。即改变对更少数群体或者潜在欢迎群体的重视程度。

4.2 数据集

在报告中，采用原论文使用的三个数据集。包括用户，评论，点赞量，物品标题等丰富的文本信息，其名称分别为 Amazon、Micro - video、Kwai。Amazon 源于服装领域，包含大量的图像视觉特征；Micro - video 取自微视频平台特定时段的交互信息，有丰富视听文特征；Kwai 为基准数据集，在推荐系统研究中具有一定的代表性，但是缺乏全局时间戳，作者采用随机划分。数据集详情可见图3：

数据集依时间序按 8:1:1 划分为训练、验证与测试集。运行时发现数据集部分缺失导致代码无法运行，因此尝试通过 github 联系了原作者反映了相关的情况，等待 2 天后很高兴收到了来自作者的回复，对数据集进行了相应的补充。由于 kwai 数据集缺乏时间戳，因此我尝试给 kwai 数据集的每一个项目生成随机时间戳，进行重新划分。

Dataset	#User	#Warm	#Cold	#Int	V	T	Density
Amazon	21,607	75,069	18,686	169,201	64	-	0.01%
Micro-video	21,608	56,712	7,725	276,629	64	768	0.02%
Kwai	7,010	74,470	12,013	298,492	64	-	0.05%

图 3. 数据集

4.3 实验环境搭建

由于本论文代码是基于 TDRO 源码的，所以只需要按 TDRO 的步骤加载环境即可。此外由于环境版本不同的原因，代码中的所有 `allow_pickle` 的值都改为 True 即可正常运行。

在模型参数设置方面，所有方法的 CF 和特征表示维度均设定为 128，并采用两层 MLP（隐藏层大小为 256）作为特征提取器。对于学习率和权重衰减，分别在 $1e-4, 1e-3, 1e-2$ 和范围 $1e-4, 1e-3, 1e-2, 1e-1$ 内进行调参。

对每个模型特定的超参数进行详细搜索，不同数据集和模型的最佳参数值本文已通过验证集上进行调优确定。主要是调整 4 个参数：聚类组数 K 、划分的时间段数量 E 、转移因子强度 λ 、周期重要性陡峭控制因子 p 。基于 CLCRec 骨干模型，对于 Amazon 数据集，最佳参数设定为 $K=5$, $E=5$, $\lambda=0.3$, $p=0.2$ ；对于 Micro-video 数据集，最佳参数设定为 $K=3$, $E=3$, $\lambda=0.1$, $p=0.2$ ；对于 Kwai 数据集，最佳参数设定为 $K=3$, $E=5$, $\lambda=0.3$, $p=0.2$ 。

对 Amazon 数据集的调参进行可视化展示，图4：

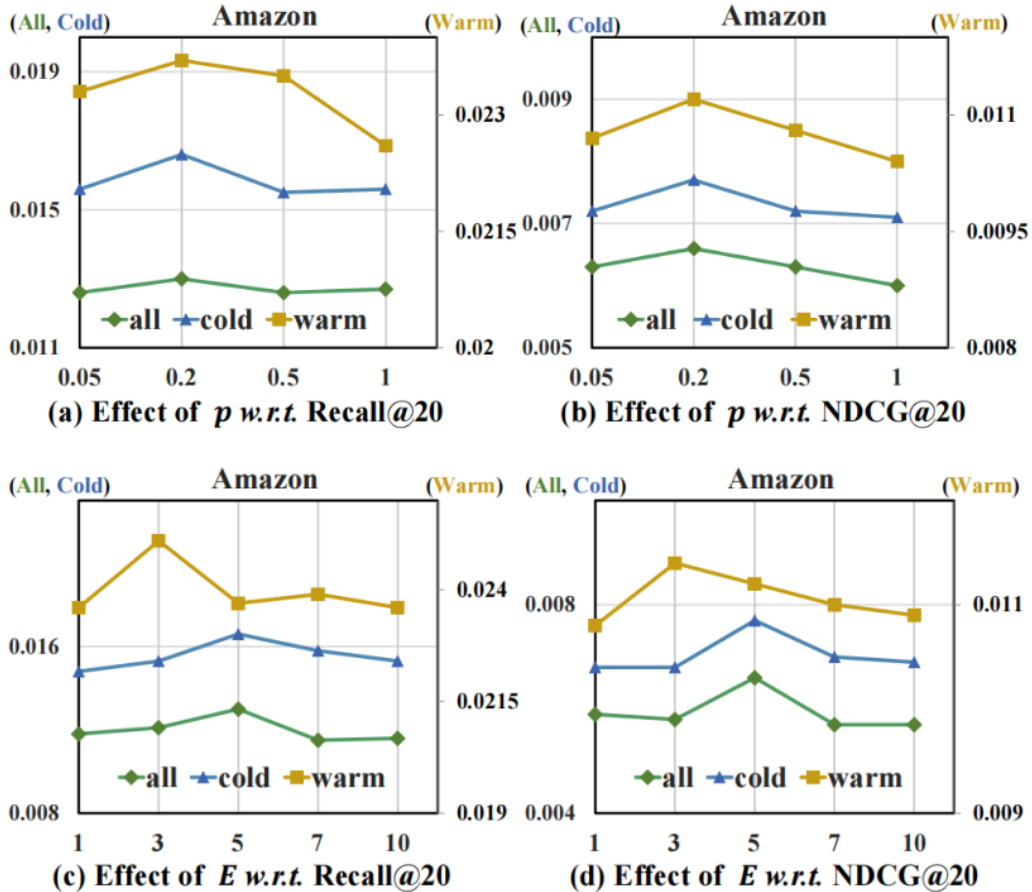


图 4. Amazon 数据集的 p, E 参数调整

4.4 创新点

对 Kwai 数据集引入随机时间戳

对 E, μ, λ, p 等重要参数进行调参

5 实验结果分析

本文采用全排序协议进行评估,考虑三种不同的设置:对所有物品进行全排序(记为“all”)、仅对温暖物品进行全排序(记为“warm”)、仅对冷物品进行全排序(记为“cold”)。同作者一样选择使用 Recall@20 和 NDCG@20 作为评估指标。

Recall@20 用于衡量在推荐列表的前 20 个物品中,真正相关物品出现的比例,反映了模型推荐出用户真正感兴趣物品的能力。计算公式:实际被推荐且与用户相关的物品数量/测试集中与用户相关的所有物品数量。

NDCG@20 则考虑了推荐列表中物品的排序位置,对排在前面的相关物品给予更高的权重。首先计算每个推荐物品的增益(Gain),若物品与用户相关则增益为 1,不相关则为 0。接着,对这些增益按照物品在推荐列表中的位置进行折损,越靠前的物品折损越小,越靠后的物品折损越大,以此体现靠前位置的物品对用户体验的重要性。

表1是复现的方法与本文的 TDRO 方法的评价指标对比,可以看到对于 Amazon 与 Micro-video 数据集,复现结果与论文结果误差不超过 10%,但是对于改动后的 Kwai 数据集,其 Recall 与 NDCG 在 all 与 Warm 上的表现有明显降低,其原因可能来自于对数据集的时间戳构造,可能随机生成的时间戳在设置上有差错,或者随机时间轴改变了数据的分布,模型在训练时可能无法很好地适应这种新的数据分布,破坏了原始数据中的潜在顺序关系,导致泛化能力下降。

Metric	Models	Amazon			Micro - video			Kwai		
		All	Warm	Cold	All	Warm	Cold	All	Warm	Cold
Recall@20	论文结果	0.013	0.0237	0.0166	0.0703	0.118	0.0761	0.0841	0.1016	0.0186
	复现结果	0.0132	0.0228	0.0144	0.0663	0.1054	0.0789	0.0583	0.071	0.0172
NDCG@20	论文结果	0.0066	0.0112	0.0077	0.0507	0.0794	0.0511	0.0597	0.0719	0.0081
	复现结果	0.0062	0.0107	0.0064	0.042	0.0652	0.0518	0.0404	0.0466	0.0078

表 1. 复现结果对比

6 总结与展望

通过对 TDRO 的复现,我了解了推荐系统中用到的基本模型 CLCRec 的结构,并且在对数据集的处理过程中,学习到数据集的查找途径以及数据格式如何转换,动手尝试对 Kwai 数据集进行处理,复现结果精度大幅度下降也让我对科研的态度更加严谨,在优化的过程中不是模块加的越多越好,要选择合适的,对 Kwai 数据集生成随机时间戳,没有考虑到用户的浏览顺序,反而打乱了用户数据的潜在顺序关系。同时在梳理代码时,对模型训练的步骤有了更加深度理解,并产生了一些想法。

在后续研究中，可以考虑在两个方向进行改进：（1）借鉴最新成果，尝试将大型语言模型与现有推荐系统架构结合，利用其捕捉物品细微语义信息，为冷启动物品提供更丰富特征表示，增强模型对用户偏好的理解和预测能力。（2）用优于基于特征聚类和时间划分方法的划分策略。例如结合深度学习中的先进算法，挖掘物品潜在关系，提升组划分质量，进而增强 TDRO 模型性能。

参考文献

- [1] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [2] Walid Shalaby, Sejoon Oh, Amir Afsharinejad, Srijan Kumar, and Xiquan Cui. M2trec: Metadata-aware multi-task transformer for large-scale and cold-start free session-based recommendations. In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 573–578, 2022.
- [3] Maksims Volkovs, Guangwei Yu, and Tomi Poutanen. Dropoutnet: Addressing cold start in recommender systems. *Advances in neural information processing systems*, 30, 2017.
- [4] Ziwei Zhu, Shahin Sefati, Parsa Saadatpanah, and James Caverlee. Recommendation for new users and new items via randomized training and mixture-of-experts transformation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1121–1130, 2020.
- [5] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. Contrastive learning for cold-start recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5382–5390, 2021.
- [6] Hao Chen, Zefan Wang, Feiran Huang, Xiao Huang, Yue Xu, Yishi Lin, Peng He, and Zhoujun Li. Generative adversarial framework for cold-start item recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2565–2571, 2022.
- [7] Vladimir Vapnik. Principles of risk minimization for learning theory. *Advances in neural information processing systems*, 4, 1991.
- [8] John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.
- [9] Yonatan Oren, Shiori Sagawa, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust language modeling. *arXiv preprint arXiv:1909.02060*, 2019.
- [10] Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. Warm up cold-start advertisements: Improving ctr predictions via learning to learn id embeddings. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 695–704, 2019.
- [11] Yitong Meng, Xiao Yan, Weiwen Liu, Huanhuan Wu, and James Cheng. Wasserstein collaborative filtering for item cold-start recommendation. In *Proceedings of the 28th ACM Conference on user modeling, adaptation and personalization*, pages 318–322, 2020.

- [12] Xue Geng, Hanwang Zhang, Jingwen Bian, and Tat-Seng Chua. Learning image and user features for recommendation in social networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4274–4282, 2015.
- [13] Xiaoyu Du, Xiang Wang, Xiangnan He, Zechao Li, Jinhui Tang, and Tat-Seng Chua. How to learn item representation for cold-start multimedia recommendation? In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3469–3477, 2020.
- [14] Oren Barkan, Noam Koenigstein, Eylon Yogev, and Ori Katz. Cb2cf: a neural multiview content-to-collaborative filtering model for completely cold item recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 228–236, 2019.
- [15] Zhihui Zhou, Lilin Zhang, and Ning Yang. Contrastive collaborative filtering for cold-start item recommendation. In *Proceedings of the ACM Web Conference 2023*, pages 928–937, 2023.
- [16] Jiashuo Liu, Jiayun Wu, Bo Li, and Peng Cui. Distributionally robust optimization with data geometry. *Advances in neural information processing systems*, 35:33689–33701, 2022.
- [17] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.