

Time Interval Aware Self-Attention for Sequential Recommendation

用于顺序推荐的时间间隔感知自注意力

报告作者: 刘予桁 22330074, 马俊 22330079

论文原作者: Jiacheng Li, Yujie Wang, Julian McAuley

论文出处: WSDM '20, February 3–7, 2020, Houston, TX, USA

1 引言

在数字化时代,推荐系统对各领域至关重要。顺序推荐系统虽重要,但传统模型如马尔可夫链、循环神经网络、部分自注意力机制模型存在局限,无法有效处理用户行为时间间隔信息。本研究提出 TiSASRec 模型,融合项目位置与时间间隔,构建用户行为模型。在多数数据集实验中,全面验证其有效性与优越性,剖析组件及参数影响,为顺序推荐研究提供新思路,提升推荐准确性,满足用户需求,增强平台竞争力,推动技术发展。

2 背景

2.1 顺序推荐系统的重要性及传统模型局限

互联网飞速发展使信息爆炸,推荐系统成为筛选海量数据关键工具,顺序推荐系统在电商、新闻、社交等场景核心地位显著。如电商中用户购买有顺序,把握此有助于精准推荐后续商品;新闻推荐依阅读顺序可提升相关性;社交网络里好友或内容推荐顺序能助力发现新关系与话题。

传统顺序推荐模型虽有推动作用,但存在局限。马尔可夫链基于前 L 个操作推荐,高稀疏性下能捕捉短期转换,可复杂场景中因简单建模,难以捕捉用户兴趣动态及长期依赖,如电商购买决策受多因素影响,它可能忽略重要关联。循环神经网络虽理论能处理长期偏好,但实际面对稀疏数据集时,需大量数据训练,不然难学行为模式,且有训练时长、梯度问题,限制其应用。

2.2 时间推荐与顺序推荐的差异

时间与顺序推荐是挖掘用户交互历史的重要路径。时间推荐侧重绝对时间戳建模,捕捉用户和项目时间动态,如电商平台部分商品随时间有周期性销量波动,其能分析不同时段行为数据考虑时间对偏好影响。

而顺序推荐主要挖掘项目顺序模式,以往常将交互历史视为简单有序序列,忽视时间间隔关键因素。像音乐播放应用中,短时间内连续播放歌曲关联强,长时间间隔播放则弱。现有研究此不足致推荐结果准确性和个性化受限,本研究提出 TiSASRec 模型填补时间间隔建模空白,提升性能与体验。

3 模型架构

在时间感知顺序推荐的设定下,我们定义用户集合为 U ,项目集合为 I 。对于任意用户 $u \in U$,其行为序列表示为 $S^u = (S_1^u, S_2^u, \dots, S_{|S^u|}^u)$,其中 $S_t^u \in I$,同时对应有时间序列 $T^u = (T_1^u, T_2^u, \dots, T_{|T^u|}^u)$ 。在训练过程中,模型在时间步 t 时,依据前 t 个项目以及项目 i 和 j 之间的时间间隔 r_{ij}^u 来预测下一个项目。

模型的输入为 $(S_1^u, S_2^u, \dots, S_{|S^u|-1}^u)$ 和序列中任意两个项目间的时间间隔 $R^u \in \mathbb{N}^{(|S^u|-1) \times (|S^u|-1)}$ ，期望输出则是序列中的后续项目，即 $(S_2^u, S_3^u, \dots, S_{|S^u|}^u)$ 。通过这样的问题定义，为模型的构建和训练明确了方向，旨在从用户的历史行为和时间信息中挖掘规律，实现准确的项目推荐。

3.1 个性化时间间隔处理

为了适应模型训练的需求，我们对用户的交互序列和时间序列进行标准化处理。将训练序列 $(S_1^u, S_2^u, \dots, S_{|S^u|-1}^u)$ 转换为固定长度 n 的序列 $s = (s_1, s_2, \dots, s_n)$ 。若原始序列长度大于 n ，则仅保留最近的 n 个动作；若小于 n ，则在左侧添加填充项直至长度达到 n 。对于时间序列 $T^u = (T_1^u, T_2^u, \dots, T_{|T^u|-1}^u)$ ，同样处理为固定长度 $t = (t_1, t_2, \dots, t_n)$ ，短序列用第一个非填充项时间戳 $T_{s_1}^u$ 填充，长序列则取前 n 个时间戳。

进一步地，我们将时间间隔视为用户序列中项目间的重要关系进行建模。考虑到不同用户交互频率的差异，我们通过将每个时间间隔除以用户序列中最短非零时间间隔 r_{min}^u ($r_{min}^u = \min(R^u)$) 来获取个性化时间间隔 $r_{ij}^u = \left\lfloor \frac{|t_i - t_j|}{r_{min}^u} \right\rfloor$ ，从而构建用户 u 的关系矩阵 $M^u \in \mathbb{N}^{n \times n}$ ，其元素为：

$$M^u = \begin{bmatrix} r_{11}^u & r_{12}^u & \dots & r_{1n}^u \\ r_{21}^u & r_{22}^u & \dots & r_{2n}^u \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1}^u & r_{n2}^u & \dots & r_{nn}^u \end{bmatrix}$$

同时，为避免稀疏关系编码以及增强模型对未见过时间间隔的泛化能力，我们将考虑的两个项目间最大相对时间间隔限制为 k ，对关系矩阵进行裁剪操作 $M_{clipped}^u = clip(M^u)$ ，确保 $r_{ij}^u = \min(k, r_{ij}^u)$ 。

3.2 嵌入层

在模型的嵌入层，我们首先为项目创建嵌入矩阵 $M^I \in \mathbb{R}^{|I| \times d}$ ，其中 d 为潜在维度，用常量零向量表示填充项的嵌入。通过嵌入查找操作，获取前 n 个项目的嵌入并堆叠成矩阵 $E^I \in \mathbb{R}^{n \times d}$ ，即：

$$E^I = \begin{bmatrix} m_{s_1} \\ m_{s_2} \\ \vdots \\ m_{s_n} \end{bmatrix}$$

对于自注意力机制中的键和值，我们分别采用两个不同的可学习位置嵌入矩阵 $M_K^P \in \mathbb{R}^{n \times d}$ 和 $M_V^P \in \mathbb{R}^{n \times d}$ ，这种设计在无需额外线性变换的情况下，能更好地适应自注意力机制的计算需求。经检索后得到相应的嵌入 E_K^P 和 E_V^P ，分别为：

$$E_K^P = \begin{bmatrix} p_1^k \\ p_2^k \\ \vdots \\ p_n^k \end{bmatrix}, \quad E_V^P = \begin{bmatrix} p_1^v \\ p_2^v \\ \vdots \\ p_n^v \end{bmatrix}$$

类似地，我们构建相对时间间隔嵌入矩阵 $M_K^R \in \mathbb{R}^{k \times d}$ 和 $M_V^R \in \mathbb{R}^{k \times d}$ ，在检索裁剪后的关系矩阵 $M_{clipped}^u$ 后，得到嵌入矩阵 $E_K^R \in \mathbb{R}^{n \times n \times d}$ 和 $E_V^R \in \mathbb{R}^{n \times n \times d}$ ，且这两个相对时间间隔嵌入矩阵为对称矩阵，主对角线元素均为零，例如 E_K^R 可表示为：

$$E_K^R = \begin{bmatrix} r_{11}^k & r_{12}^k & \dots & r_{1n}^k \\ r_{21}^k & r_{22}^k & \dots & r_{2n}^k \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1}^k & r_{n2}^k & \dots & r_{nn}^k \end{bmatrix}$$

3.3 时间间隔感知自注意力层

受相对位置自注意力机制的启发，本模型在传统自注意力机制基础上进行创新，同时考虑项目的时间间隔和位置信息。对于输入序列 $E^I = (m_{s_1}, m_{s_2}, \dots, m_{s_n})$ ($m_{s_i} \in \mathbb{R}^d$)，计算新序列 $Z = (z_1, z_2, \dots, z_n)$ ($z_i \in \mathbb{R}^d$)，其中每个输出元素 z_i 由线性变换后的输入元素与关系/位置嵌入的加权和计算得出，即

$$z_i = \sum_{j=1}^n \alpha_{ij} (m_{s_j} W^V + r_{ij}^V + p_j^V)$$

这里 $W^V \in \mathbb{R}^{d \times d}$ 是值的输入投影。

权重系数 α_{ij} 通过 softmax 函数计算：

$$\alpha_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^n \exp e_{ik}}$$

而 e_{ij} 则由考虑输入、关系和位置的兼容性函数计算：

$$e_{ij} = \frac{m_{s_i} W^Q (m_{s_j} W^K + r_{ij}^K + p_j^K)^T}{\sqrt{d}}$$

其中 $W^Q \in \mathbb{R}^{d \times d}$ 和 $W^K \in \mathbb{R}^{d \times d}$ 分别为查询和键的输入投影， \sqrt{d} 用于避免高维情况下内积过大。

考虑到序列预测的因果关系，由于模型在预测第 $(t+1)$ 个项目时应仅依赖前 t 个项目信息，而原始时间感知自注意力层的第 t 个输出包含所有输入信息，因此我们禁止 Q_i 和 $K_j (j > i)$ 之间的链接 ($Q_i = m_{s_i} W^Q$, $K_j = m_{s_j} W^K + r_{ij}^K + p_j^K$)。

此外，为赋予模型非线性能力，在每个时间感知注意力层之后，我们应用两个线性变换，并在中间加入 ReLU 激活函数，构建逐点前馈网络 $FFN(z_i) = \max(0, z_i W_1 + b_1) W_2 + b_2$ ($w_1, w_2 \in \mathbb{R}^{d \times d}$, $b_1, b_2 \in \mathbb{R}^d$)，且不同层间线性变换参数不同。同时，为解决模型训练过程中的过拟合、不稳定（如梯度消失）及训练时间长等问题，我们采用层规范化（LayerNorm）、残差连接和 dropout 正则化技术，公式为

$$Z_i = z_i + \text{Dropout}(FFN(\text{LayerNorm}(z_i)))$$

其中层规范化通过

$$\text{LayerNorm}(x) = \alpha \odot \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta$$

实现（ \odot 为元素乘积， μ 和 σ 为 x 的均值和方差， α 和 β 为学习到的缩放因子和偏差项）。

3.4 预测层

在经过多层时间间隔感知自注意力模块的处理后，模型获得了项目、位置和时间间隔的综合表示。为了预测下一个项目，我们采用潜在因子模型，根据用户前 t 个项目及其时间间隔的表示 Z_t 和项目 i 的嵌入 $M_i^I \in \mathbb{R}^d$ ，计算用户对项目 i 的偏好分数

$$R_{i,t} = Z_t M_i^I$$

通过比较不同项目的偏好分数，实现对下一个可能感兴趣项目的预测，从而为用户提供个性化的推荐服务。

3.5 模型创新点

TiSASRec 模型在时间感知顺序推荐领域具有显著创新。首先，在个性化时间间隔处理方面，通过对用户交互序列和时间序列的标准化及独特的时间间隔建模，有效适应了不同用户交互频率差异，并增强了对未见过时间间隔的泛化能力。其将时间间隔视为关键关系，经特定计算构建关系矩阵，并通过合理裁剪避免稀疏编码问题。

在嵌入层，为项目、自注意力机制中的键和值以及相对时间间隔分别构建了专门的嵌入矩阵，且键和值的位置嵌入矩阵设计独特，无需额外线性变换即可满足计算需求，这种多维度的嵌入方式为模型准确捕捉信息奠定了基础。

时间间隔感知自注意力层是模型的核心创新之一，它在传统自注意力机制上融合了项目时间间隔和位置信息，通过创新的加权和计算方式及兼容性函数确定权重系数，同时考虑因果关系对注意力链接进行限制。此外，采用逐点前馈网络并结合层规范化、残差连接和 dropout 正则化技术，有效解决了过拟合、不稳定及训练时长等问题，提升了模型性能。

最后，预测层采用潜在因子模型，基于前序项目及其时间间隔表示与项目嵌入计算偏好分数，实现精准的项目推荐，充分利用了模型前期处理所获取的综合信息，为用户提供高度个性化的推荐服务。

4 实验复现

4.1 ReChorus 框架

在推荐系统领域，深度学习催生众多推荐算法，可复现性问题因多方面差异而凸显。ReChorus 框架基于 PyTorch 构建，含 Preprocessing、Reader、Model、Runner 和 Utils 层。Preprocessing 层将公开数据转成固定格式输入文件，Reader 层为 Model 层提供处理后数据，Model 层实现关键功能，Runner 层规范流程并优化性能，Utils 层提供工具。在 Amazon Grocery 和 MovieLens 等数据集上，用留一法划分数数据集，以 HR 和 NDCG 为指标实验。通过 ReChorus 框架进行复现实验，可以缓解可复现性问题，支撑推荐算法研究和模型对比，

4.2 对比模型

本次研究采用多个典型推荐模型实验分析。这些模型在各自擅长场景和不同数据特性下特点鲜明、性能各异，形成丰富对比关系，为研究提供全面深入参考，助于探究推荐系统性能差异与优化方向。POP 模型依项目在全体用户训练集的交互频次排序生成推荐结果，操作简单，但难兼顾用户个性化需求，挖掘用户偏好不足。FPMC 模型融合矩阵分解与一阶马尔可夫链技术，能捕捉用户长期偏好和项目动态转换规律，在顺序推荐任务中有优势，可适应数据序列关系提供贴合行为序列的推荐。GRU4Rec 模型针对会话推荐场景建模用户动作序列，GRU4Rec + 模型采用特定策略提升性能，可洞察会话中行为模式变化与兴趣动态转移，提供精准适时推荐。Caser 模型将项目序列嵌入时间与潜在空间“图像”，捕获高阶马尔可夫链信息，挖掘用户行为潜在模式规律，为推荐提供可靠依据，在挖掘潜在关联方面优势显著。SLRCPlus 模型在推荐中作用重要，对特定数据和用户行为有良好适应性与有效性。BUIR 模型在推荐任务中不可或缺，提升整体推荐效果。

4.3 数据集

在本次实验中，选用 Groceries、MIND - Large 和 MovieLens 三个特色数据集。MovieLens 数据丰富且密集，用户行为多样频繁，有力检验模型在充足数据下的性能。Groceries 聚焦食品杂货，数据稀疏，用户购买低频，能考验模型应对稀疏数据能力。MIND - Large 有独特特征，在数据分布和用户行为上与其他两者有别，提供不同评估视角。数据预处理时，对这三个数据集采取有效措施。先筛选掉交互少于 5 的用户和项目，降低数据噪声。再将时间戳转为时间间隔并离散化，确保数据可嵌入模型分析，保障数据质量与模型有效性，为实验奠定基础。

4.4 任务与评价指标

在本次复现实验里，核心是 Top - k 推荐，旨在从众多候选项目中选出最契合用户兴趣的前 k 个。因用户常关注推荐列表前端少数项目，精准确定这 k 个项目对提升用户体验和系统实用性极为关键。

具体操作时，模型先全面分析评估所有候选项目，学习用户历史行为、项目特征及相互关系等信息，算出每个项目相对用户的满意程度得分，再按得分降序选前 k 个作为推荐结果。如电影推荐中 k 设为 10 时，模型综合考量用户观影记录、电影类型等因素计算得分，挑出得分最高的 10 部电影推荐给用户，以在有限展示空间提供有价值信息，助用户发现兴趣项目，提升系统效率与准确性，促进平台发展。

在推荐系统评估方面，采用 $HR@k$ 和 $NDCG@k$ 两个关键指标。 $HR@k$ 衡量推荐列表前 k 个位置是否命中用户感兴趣项目，公式为 $HR@k = \frac{\text{命中的项目数}}{\text{测试集的项目数}}$ ，数值越高成功率越高。 $NDCG@k$ 更全面考虑推荐项目位置因素及相关性得分，先算 DCG ，再算 $IDCG$ ，最后得 $NDCG@k = \frac{DCG@k}{IDCG}$ ，常见二分类相关性场景且 $IDCG = 1$ 时可简化计算，它能精细评估推荐列表质量，重视命中项目位置权重，位置越靠前且相关性越高贡献越大。

与原文仅做 @10 实验不同，本次复现开展了 @5、@10、@20、@50 实验，可深入剖析模型在不同推荐规模下表现，为后续研究提供更丰富视角和详实数据依据。

4.5 参数设置

本次实验为确保算法对比公平有效并结合现有算力精心设置参数。模型层面， emb_size 设为 64 平衡特征捕捉与复杂度， num_heads 和 num_layers 均设为 1 把控结构复杂程度， $dropout$ 取 0 维持模型完整， $l2$ 正则化参数设为 1×10^{-6} 防过拟合。数据处理上， $batch_size$ 与 $eval_batch_size$ 都定为 256 保障效率与稳定，选用 *Grocery_and_Gourm...* 数据集契合应用场景。训练进程中， $epoch$ 设为 1， lr 设为 0.0001，用 *Adam* 优化器优化参数， $early_stop$ 设为 10 防过度训练与过拟合，节省资源时间。其他参数方面， num_neg 设为 1， $num_workers$ 设为 5， $random_seed$ 取 0 确保可重复性， $save_final_results$ 设为 1 便于结果分析评估， $test_all$ 设为 0， $topk$ 设为 5, 10, 20, 50 提供多维度评估视角， gpu 设为 0 在指定 *GPU* 上训练加速进程。

4.6 结果与分析

表 1 - 4 呈现了除 Caser 外各模型在 Groceries、MIND - Large、MovieLens 数据集的推荐性能，TiSASRec 在多指标表现突出。在 Groceries 集， $NDCG@5$ 上 TiSASRec 达 0.2945，相比 BPRMF (0.1364)、Caser (0.1338) 提升 79.68%。MIND - Large 集， $NDCG@5$ 指标 TiSASRec 为 0.109，比 BPRMF (0.0269)、Caser (0.046) 提升 27.49%。MovieLens 集， $NDCG@5$ 上 TiSASRec 为 0.3791，相较于 BPRMF (0.2311)、Caser (0.2594) 提升 42.79%。

总体而言，TiSASRec 在这三个数据集的 $NDCG$ 和 HR 指标上均优于其他基线方法。因其利用注意力机制依项目、位置和时间间隔调整权重，且能自适应关注项目。虽实验计算资源有限致结果与原文有差异，但模型优势仍明显，提升趋势显著，证明其有效且有潜力。原论文仅 @10 指标，本研究多维度实验，@5 凸显早期准确性，@20 反映中等规模稳定性，@50 考验广泛有效性与泛化能力，为评估和后续研究提供依据，未来将改善条件深入探究。

数据集/模型	BPRMF	FPMC	GRU4Rec	POP	SLRCPlus	BUIR	TiSASRec	提升比
Groceries	$NDCG@5:0.1364$	0.1594	0.1314	0.1281	<u>0.1639</u>	0.1334	0.2945	79.68%
	$HR@5:0.2211$	0.246	0.211	0.2031	<u>0.2598</u>	0.2075	0.3902	50.19%
MIND-Large	$NDCG@5:0.0269$	<u>0.0855</u>	0.0782	0.0431	0.0423	0.042	0.109	27.49%
	$HR@5:0.0441$	<u>0.1353</u>	0.1324	<u>0.0755</u>	0.0735	0.0627	0.1765	30.45%
MovieLens	$NDCG@5:0.2311$	<u>0.2655</u>	0.2623	0.2264	0.2293	0.2182	0.3791	42.79%
	$HR@5:0.3392$	0.3942	<u>0.3949</u>	0.3333	0.3417	0.325	0.5139	30.13%

表 1: $NDCG@5$ & $HR@5$ 性能对比

数据集/模型	BPRMF	FPMC	GRU4Rec	POP	SLRCPlus	BUIR	TiSASRec	提升比
Groceries	NDCG@10:0.1757	0.1996	0.1765	0.1689	<u>0.2103</u>	0.1776	0.3241	54.11%
	HR@10:0.3417	0.3697	0.3515	0.33	<u>0.403</u>	0.3444	0.4816	19.50%
MIND-Large	NDCG@10:0.0427	<u>0.1052</u>	0.1049	0.0698	0.057	0.0518	0.1383	31.46%
	HR@10:0.0941	0.1971	<u>0.2147</u>	0.1598	0.1186	0.0941	0.2676	24.64%
MovieLens	NDCG@10:0.2835	<u>0.3194</u>	0.3169	0.28	0.2807	0.2716	0.4278	33.94%
	HR@10: 0.5017	0.5605	<u>0.5647</u>	0.4997	0.501	0.4903	0.6639	17.57%

表 2: NDCG@10 & HR@10 性能对比

数据集/模型	BPRMF	FPMC	GRU4Rec	POP	SLRCPlus	BUIR	TiSASRec	提升比
Groceries	NDCG@20:0.2026	0.2274	0.2144	0.2064	<u>0.2434</u>	0.2107	0.35	43.80%
	HR@20:0.4481	0.4799	0.5005	0.4784	<u>0.5337</u>	0.4754	0.5846	9.54%
MIND-Large	NDCG@20:0.0699	<u>0.1475</u>	0.1334	0.0911	0.0688	0.0782	0.1807	22.51%
	HR@20:0.2029	<u>0.3667</u>	0.3284	<u>0.2441</u>	0.1657	0.201	0.4363	18.98%
MovieLens	NDCG@20:0.3322	0.3645	<u>0.366</u>	0.3281	0.3294	0.3211	0.4651	27.08%
	HR@20:0.6945	0.7397	<u>0.7585</u>	0.6907	0.6942	0.6862	0.8107	6.88%

表 3: NDCG@20 & HR@20 性能对比

数据集/模型	BPRMF	FPMC	GRU4Rec	POP	SLRCPlus	BUIR	TiSASRec	提升比
Groceries	NDCG@50:0.2441	0.2699	0.2533	0.2442	<u>0.2801</u>	0.2483	0.3883	38.63%
	HR@50:0.6598	0.6968	0.6967	0.669	<u>0.7196</u>	0.6658	0.7788	8.23%
MIND-Large	NDCG@50:0.1260	0.1818	<u>0.186</u>	0.1377	0.1184	0.1427	0.2355	26.61%
	HR@50:0.4902	<u>0.5382</u>	<u>0.5912</u>	0.4843	0.4196	0.5373	0.7186	21.55%
MovieLens	NDCG@50: 0.3789	<u>0.4055</u>	0.4043	0.3749	0.3765	0.3694	0.4952	22.12%
	HR@50:0.9273	0.944	<u>0.9489</u>	0.9238	0.9287	0.9266	0.96	1.17%

表 4: NDCG@50 & HR@50 性能对比

5 总结

本研究致力于对 TiSASRec 模型的复现工作，旨在深入探究其在推荐系统中的性能表现与应用潜力。在复现过程中，我们精心构建了实验体系，涵盖数据集的审慎筛选、参数的细致设置、评价指标的合理选定以及模型训练与评估流程的严格把控。选用 Groceries、MIND - Large 和 MovieLens 数据集，充分考虑其数据特性差异，为模型提供多样化测试场景，并实施有效预处理操作以提升数据质量。在参数设置方面，兼顾模型复杂度与性能平衡、计算资源限制及训练效率优化等因素，确定了如嵌入维度、batch 大小、学习率等一系列关键参数。采用 Hit Rate@k 和 Normalized Discounted Cumulative Gain@k 作为核心评价指标，从不同维度精准衡量模型推荐效果。实验结果显示，TiSASRec 模型在与多个基线模型对比中表现卓越。在不同数据集及推荐列表长度下，其 NDCG 和 HR 指标均有显著提升，充分彰显模型在捕捉用户偏好、融合多维度信息及提升推荐准确性方面的优势。尽管受计算资源限制与原文结果存在差异，但模型提升趋势明显，验证其有效性与潜力。同时，相比原文仅关注 @10 指标，本复现开展的 @5、@20、@50 等多维度实验，深入剖析模型在不同推荐规模下表现，为后续研究提供丰富视角与详实数据，有力推动推荐系统领域发展，未来将持续优化计算条件深化研究。

项目仓库

https://github.com/liuyh357/TiSASRec_ReChorus.git

References

- Cheng, Heng-Tze et al. (2016). Wide Deep Learning for Recommender Systems. arXiv: 1606.07792 [cs.LG]. URL: <https://arxiv.org/abs/1606.07792>.
- Harper, F. Maxwell and Joseph A. Konstan (Dec. 2015). “The MovieLens Datasets: History and Context”. In: ACM Trans. Interact. Intell. Syst. 5.4. ISSN: 2160-6455. DOI: 10.1145/2827872. URL: <https://doi.org/10.1145/2827872>.
- Li, Jiacheng, Yujie Wang, and Julian McAuley (2020). “Time Interval Aware Self-Attention for Sequential Recommendation”. In: Proceedings of the 13th International Conference on Web Search and Data Mining. WSDM '20. Houston, TX, USA: Association for Computing Machinery, pp. 322–330. ISBN: 9781450368223. DOI: 10.1145/3336191.3371786. URL: <https://doi.org/10.1145/3336191.3371786>.
- Li, Jiayu et al. (2024). ReChorus2.0: A Modular and Task-Flexible Recommendation Library. arXiv: 2405.18058 [cs.IR]. URL: <https://arxiv.org/abs/2405.18058>.
- McAuley, Julian et al. (2015). Image-based Recommendations on Styles and Substitutes. arXiv: 1506.04757 [cs.CV]. URL: <https://arxiv.org/abs/1506.04757>.
- Semmelrock, Harald et al. (2023). Reproducibility in Machine Learning-Driven Research. arXiv: 2307.10320 [cs.LG]. URL: <https://arxiv.org/abs/2307.10320>.