



基于对比学习和傅里叶变换的序列推荐算法

张少东, 杨兴耀*, 于 炯, 李梓杨, 刘岩松

(新疆大学软件学院 乌鲁木齐 830046)

【摘要】提出一种基于自注意力机制和傅里叶变换的序列推荐算法 CSFTRec。通过过滤原始数据中的噪声, 最大限度地提高自注意力机制对序列数据的特征捕捉能力。根据对比学习的特点, 在贝叶斯个性化排名的基础上引入一种新的对比损失, 用于联合训练, 可以缩短不同相似序列之间的距离。在 8 个公共数据集上的实验表明, CSFTRec 的收敛速度更快, 推荐精度有 3%~5% 的提高, 更适合处理序列数据。

关键词 对比学习; 推荐算法; 序列推荐; 自注意力机制

中图分类号 TP391 **文献标志码** A **doi**:10.12178/1001-0548.2022164

Sequence Recommendation Based on Contrast Learning and Fourier Transform

ZHANG Shaodong, YANG Xingyao*, YU Jiong, LI Ziyang, and LIU Yansong

(School of Software, Xinjiang University Urumqi 830046)

Abstract This paper proposes a sequence recommendation algorithm based on self-attention mechanism and Fourier transform, named CSFTRec. By filtering the noise in the original data, this algorithm maximizes the feature capturing ability of the self-attention mechanism on the sequence data. According to the characteristics of contrast learning, a new contrast loss is introduced on the basis of Bayesian personalized ranking for joint training, which can shorten the distance between different similar sequences. Experiments on eight public data sets show that CSFTRec converges faster and improves the recommendation accuracy by 3% to 5%, which indicates that CSFTRec is more suitable for processing sequence data.

Key words contrastive learning; recommender algorithm; sequential recommendation; self-attention mechanism

推荐算法在近些年发展非常迅速^[1], 受到各行各业的关注, 尤其是电商行业, 十分依赖于推荐算法的应用, 以此来提升目标用户的粘性和体验。随着信息量的增长, 推荐算法也成为应对信息过载的主要手段, 可以有效帮助用户从海量数据中筛选出目标项目, 并且预测接下来有可能交互的对象。

早期的推荐算法是根据用户或物品间的相似性达到推荐的目的^[2], 但算法效果并不理想。实际上, 用户的交互行为在大多数情况下是动态的, 且会随着时间的推移而不断改变。因此, 如何对用户的行为进行单独建模并挖掘其潜在兴趣成为推荐算法的研究热点。序列推荐是一种能够根据用户的历史行为来动态捕捉用户行为特征的重要推荐算法,

通过输入一个用户交互序列来预测用户下一步的行为。传统序列推荐算法在训练模型时会遵循监督学习范式, 通过引入一个或多个负样本, 使用贝叶斯个性化排名 (Bayesian personalized ranking, BPR) 损失函数^[3]来优化算法。但监督学习会将注意力重点放在样本标注与训练数据之间的关系对结果产生的影响上, 比较依赖数据标注的准确性^[4]。而且在对用户交互序列建模时也很容易受到噪声数据的影响, 使得模型非常脆弱, 无法做到精准推荐。

针对上述问题, 本文提出一种基于对比学习与傅里叶变换的自注意力序列推荐算法 CSFTRec, 通过训练方法的转变与噪声数据的消除提高算法性能。首先, 通过傅里叶变换在序列数据进入编码

收稿日期: 2022-05-31; 修回日期: 2022-09-21

基金项目: 国家自然科学基金 (61862060, 61966035, 61562086); 新疆维吾尔自治区教育厅项目 (XJEDU2016S035); 新疆维吾尔自治区自然科学基金 (2022D01C56)

作者简介: 张少东 (1999-), 男, 主要从事对比学习、推荐算法方面的研究。

*通信作者: 杨兴耀, E-mail: yangxy@xju.edu.cn

器 (Encoder) 之前过滤噪声。同时, 引入两种数据增强方式处理序列数据, 将原始序列与增强序列同时作为自注意力层的输入。最后, 通过改进的对比损失函数计算损失值。在 8 个公开数据集上的实验证明, 本文算法相比于目前的主流推荐算法, 能有效提高推荐结果的准确性。

1 序列推荐研究现状

现有的序列推荐算法可以通过其对用户偏好建模的类型分为 3 种。

1) 基于短期偏好的序列推荐算法, 以马尔科夫链推荐为代表, 低阶的马尔科夫链^[5]是根据当前交互项, 预测用户的下一交互项。高阶马尔科夫链^[6]在低阶的基础上增加能够对目标项产生影响的交互项个数, 但当用户交互序列较长时依然无法有效捕捉交互项之间的依赖关系, 所以无法体现用户的长期偏好。卷积神经网络也被应用于处理用户的交互序列^[7], 效果相较于马尔科夫链略有提升, 但核心思想还是对用户的短期偏好进行建模, 算法依然存在局限性。

2) 循环神经网络 (recurrent neural network, RNN) 是一类专门处理序列数据的神经网络, 通过 RNN 构建的序列推荐算法^[8]能够从全局的角度考虑整个交互序列, 以对用户的长期偏好进行建模。在 RNN 的基础上, 又拓展出基于长短期记忆 (long short-term memory, LSTM)^[9] 和基于门控循环单元 (gated recurrent unit, GRU) 的网络。这两种网络有更多的参数和更好的性能, 适合构建大型推荐网络, 也可以有效预防模型的过拟合。此外, 记忆网络 (memory networks, MemNN)^[10] 也可以用来对用户长期偏好建模, 通过引入一个外部存储器来保存序列与下一个交互项之间的依赖关系。MemNN 相较于 RNN 系列网络, 能有效降低算法的存储和计算压力。

3) 此外, 还有诸多模型综合考虑了长短期偏好对算法的影响。图神经网络 (graph neural networks, GNN)^[11] 在序列推荐中使用有向图表示用户的交互序列, 图中的一个节点就表示一个交互对象, 将每个序列映射为图中的一条路径。GNN 有着较强的可扩展性, 如融入上下文增强的 GCE-GNN^[12] 和融入多重加权图的 FGNN^[13], 分别从不同角度对 GNN 在推荐算法中的应用做出完善。但 GNN 存在信息丢失问题, 后续也提出了 LESSR^[14] 等一系列模型去处理信息丢失。注意力机制也是一种比较流行的

基于长短期偏好建模的序列处理算法, 早期应用于机器翻译任务中^[15]。NARM^[16] 则是使用普通注意力机制完成推荐任务, 其强调输入的各部分对输出的影响程度不同, NARM 的成功也使得注意力机制在推荐算法中广泛应用。本文选取自注意力机制作为序列推荐算法的核心, 本文所提方法主要基于对比学习和自注意力机制。

2 相关工作

2.1 对比学习

对比学习任务主要可以分为 3 个阶段。第一阶段通过一系列对原始数据的处理, 得出增强数据, 这种处理也被称为数据增强。第二阶段通过 Encoder 对训练数据进行学习, Encoder 主要用于提取数据特征。第三阶段对不同的数据特征进行对比学习, 比较不同特征之间的相似程度。

对比学习使用数据本身的特征信息来指导学习, 在计算机视觉 (computer vision, CV) 以及自然语言处理 (natural language processing, NLP) 领域得到广泛应用, 其更侧重于算法前后的逻辑设计, 而很少去对 Encoder 部分做直接优化, 甚至在部分对比学习任务中, 并没有对 Encoder 类型进行明确规定, 可以随意替换不同的 Encoder 完成不同领域的对比学习任务^[17]。如 ImageNet 等任务中, 对比学习的表现完全可以媲美传统的监督学习方法, 而且还有提升的潜力^[18]。

对比学习将原始数据和增强数据两两组合成为样本对, 增强后的数据相较于原始数据称为正样本 x^+ , 不同组合的增强数据可以相互成为负样本 x^- , 利用 Encoder $f(\cdot)$ 提取出数据的特征表示。对比学习目标则是通过不断修正正负样本之间的距离 $\text{score}(\cdot)$, 最终得出一个能够做到目标项和正样本之间距离尽可能近, 和负样本之间距离尽可能远的模型:

$$\text{score}(f(x), f(x^+)) \gg \text{score}(f(x), f(x^-))$$

对比学习通常有多个负样本, 若设负样本个数为 k , 则对比损失函数 InfoNCE^[19] 为:

$$\mathcal{L} = -\log \frac{\exp(g(x, x^+)/\tau)}{\exp(g(x, x^+)/\tau) + \sum_{k=1}^K \exp(g(x, x_k^-)/\tau)}$$

式中, $g(\cdot)$ 为相似度函数, 如余弦相似度; τ 是超参数温度系数, 温度越高, 样本的分布会相对平滑。通过优化 InfoNCE 可以提高相似样本之间的相似性, 降低两个不相似样本之间的相似性。

2.2 自注意力机制

对于序列模型来说, 处理数据时很难做到尽可能多地考虑数据前后的语义信息, 从而导致训练出的序列模型效果很不理想。如在词性标注任务当中, 当同一个词出现在一个句子的不同位置时, 传统的序列模型无法做到准确区分不同位置的词性区别, 会直接给予相同的词性标注^[20], 即使通过一些特殊方法, 能够将有限的上下文信息考虑进去, 也会在很大程度上增加模型的参数数量, 提升模型的训练成本。

而自注意力机制能够准确地揭示出句子中不同单词之间的语法和语义信息。并且从对比实验结果来看, 自注意力机制的模型效果要远超于传统的 CNN 或 RNN, 并且对于序列推荐问题效果也不错, 如基于自注意力机制的序列推荐算法 SASRec^[21]。注意力计算被定义为:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V}$$

式中, \mathbf{Q} 代表一次查询; \mathbf{K} 和 \mathbf{V} 是键值对; d 表示 \mathbf{K} 的维度大小。直观地说, \mathbf{Q} 、 \mathbf{K} 、 \mathbf{V} 表示 3 个矩阵, 由输入数据通过一个全连接网络或者核大小为 1 的卷积层得到。但当输入维度较大时, 点乘会导致数据剧烈增大, 且通过 softmax 函数后可得到的梯度又过小, 因此需要将 \mathbf{Q} 与 \mathbf{K} 的转置的乘积再乘以 d 的平方根的倒数, 最后通过 softmax 层与矩阵 \mathbf{V} 相乘, 得到注意力得分。

3 基于对比学习与傅里叶变换的自注意力推荐算法

由于自注意力机制 (SASRec) 对序列模型强大的表示能力和特征提取能力, 本文提出一种基于对比学习与傅里叶变换的自注意力推荐算法 CSFTRec, 该算法由 3 部分组成, 如图 1 所示。

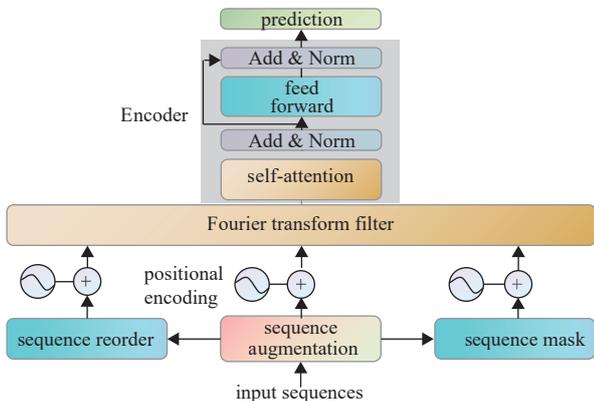


图 1 CSFTRec 整体架构

首先, 使用两种数据增强方式对原始序列进行处理。同时, 在序列数据进入 Encoder 前, 为其添加位置编码, 用以表示不同交互项之间的位置关系。

其次, 对原始数据和增强数据进行降噪处理, 傅里叶变换则是这一步的重要工具, 通过 FFT 将交互序列信号由时域转换到频域, 过滤噪音信号之后, 再通过 IFFT 将信号从频域转换到 Encoder 可处理的时域。

最后, 引入一种 Context-Context 语境对比损失函数^[22], 使其能够与 InfoNCE 损失联合完成对比学习任务, 综合考虑用户交互序列中各种有可能会影响到目标结果的特征因素, 以此来提高推荐算法的准确性。

3.1 针对序列推荐任务的数据增强

在 CV 领域, 不同的数据增强方式对不同的任务起到的作用也不尽相同, 如 SimCLR^[23], 通过对比现有的数据增强方法得出, 并不是每一种数据增强都能对学习任务起到促进作用^[24]。对比学习下的序列推荐依然遵循这一道理, 本文采用遮盖和置换^[25]两种增强方法。有实践表明, 这两种方法可以在做到数据增强的同时, 有效防止过拟合现象, 增强模型的鲁棒性。

遮盖增强是使用一个特殊值随机遮盖用户交互序列中的一个或部分交互项, 使其在与原始数据的对比学习中, 能够让 Encoder 将更多的注意力放在未被遮盖的部分, 其增强过程可以表示为:

$$\begin{aligned} p_1 &\sim \text{Beta}(\alpha = 3, \beta = 3), n_1 = \lfloor Np_1 \rfloor \\ \text{idx} &= \text{zeroslike}(S_t) \\ \text{idx}[n_1] &= 1, \text{Shuffle}(\text{idx}) \\ \tilde{S}_t &= \text{Copy}(S_t), \tilde{S}_t[\text{idx}] = [\text{mask}] \end{aligned}$$

对于给定长度为 N 的序列 S_t , 首先从 Beta 分布中随机抽取一个数值 p_1 , 用以计算掩盖位置 n_1 , 然后将特殊值 mask 覆盖在掩盖位置上完成数据增强。置换与遮盖相似, 区别在于置换是进行位置交换操作, 其增强过程可表示为:

$$\begin{aligned} p_2 &\sim \text{Beta}(\alpha = 3, \beta = 3), n_2 = \lfloor Np_2 \rfloor \\ s_{\text{start}} &\sim \{0, 1, 2, \dots, N - n_2\} \\ s_{\text{end}} &= s_{\text{start}} + n_2 \\ \tilde{S}_t &= \text{Copy}(S_t), \text{Shuffle}(\tilde{S}_t[s_{\text{start}} : s_{\text{end}}]) \end{aligned}$$

上述两种方法在使用中通过随机选择的方式对原始数据进行增强。

3.2 基于傅里叶变换的用户交互序列噪声过滤

研究发现不论是水文学^[26]、地震学^[27] 还是人类行为学^[28], 自然界和人类行为的低频信号通常是有

意义的周期性特征, 而高频信号却无法表现出这一特性。也有研究表明, 在序列推荐算法中, 模型往往更受益于低频信号^[29], 站在这一角度去看待用户交互序列, 可以将其理解为用户在某一特定场景下所释放出的信号表示, 而这种信号表示在一定程度上反映了用户行为的周期性特征, 这样就可以使用数字信号处理中的方法和工具对用户的信号表示进行处理, 达到降噪的目的。

现给定一个用户交互序列矩阵 $\mathbf{F}^l \in \mathbb{R}^{n \times d}$, 其中 l 表示训练批次中的一个样例, 将其通过快速傅里叶变换转换至频域:

$$\mathbf{X}^l = f(\mathbf{F}^l) \in \mathbb{C}^{n \times d}$$

式中, \mathbf{X}^l 是 \mathbf{F}^l 的复数形式张量表示; $f(\cdot)$ 是快速傅里叶变换。然后将其与滤波器 $\mathbf{W} \in \mathbb{C}^{n \times d}$ 相乘, 通过 \mathbf{W} 来调整张量中的信号表示:

$$\tilde{\mathbf{x}}^l = \mathbf{W} \odot \mathbf{X}^l$$

最后使用快速傅里叶逆变换对过滤后的序列进行处理, 将其由频域转换至时域:

$$\tilde{\mathbf{F}}^l = f^{-1}(\tilde{\mathbf{x}}^l) \in \mathbb{R}^{n \times d}$$

式中, $f^{-1}(\cdot)$ 为快速傅里叶逆变换, 同时为了缓解梯度消失和训练不稳定的问题, 在模型中再加一层 LayerNorm^[30] 和 Dropout^[31]:

$$\tilde{\mathbf{F}}^l = \text{LayerNorm}(\mathbf{F}^l + \text{Dropout}(\tilde{\mathbf{F}}^l))$$

3.3 基于 InfoNCE 与 Context-Context 损失的联合对比学习

传统的监督学习序列推荐算法大多会采用 BPR 作为训练目标, 将推荐问题看作是一个 Ranking 问题, 推荐效果的好坏, 表现为用户的期待项在 Ranking 中的位置是否靠前。而事实也证明 BPR 算法在推荐领域很有成效, 通过引入一个负样本, 基于大量的正负样本对, 根据贝叶斯后验来解决推荐问题:

$$\mathcal{L}_{\text{BPR}} = \sum_{(S_t, i_t)} -\log \sigma(\hat{y}(S_t, i_t) - \hat{y}(S_t, i_t^-))$$

该损失函数可以使目标项相比于负样本有更大概率获得更高的 Ranking 分数。实际上, BPR 算法也可以被推广为一种特殊的 InfoNCE 损失函数:

$$\mathcal{L} = \sum_{(S_t, i_t)} -\log \left(\frac{1}{1 + \exp(-(\hat{y}(S_t, i_t) - \hat{y}(S_t, i_t^-)))} \right)$$

这样推广的前提条件是公式中只允许存在一个负样本, 以及 InfoNCE 中的温度系数 τ 恒等于 1。这表明 BPR 与 InfoNCE 在一定程度上具有内在联系。将推广后的 InfoNCE 进一步优化得出:

$$\mathcal{L}_1 = \tau_1 \sum_{(S_t, i_t)} - \log \left(\frac{\exp(g(f(S_t), i_t) / \tau_1)}{\exp(g(f(S_t), i_t) / \tau_1) + \sum_{k=1}^K \exp(g(f(S_t), i_t) / \tau_1)} \right)$$

式中, k 是负样本个数, 相比于普通 BPR 算法, 该损失可以在完成对比任务的同时适应多个负样本的情况。此外, 如果先不考虑经过处理的增强序列, 两个具有相同目标的普通序列是否具有潜在的相似性, 以及这种相似性是否能对学习任务提供帮助, 即使他们从表面上看起来可能完全不相同, 这些都是在对比学习任务中需要着重考虑的问题, 但 BPR 和 InfoNCE 都只考虑到了相似序列。而 Context-Context 损失则可以解决这一问题, 可以将有着相似意图的序列表达尽可能拉近距离。为了优化这一目标, 将所有有着相同目标的序列以及其增强序列放在同一集合中, 集合中的每一个元素可以互相设置为正样本, 由此得到一个新的对比学习任务:

$$\mathcal{L}_2 = \tau_2 \sum_{\tilde{S}_t \in \mathcal{A}} \frac{1}{|T(\tilde{S}_t)|} \sum_{\tilde{S}'_t \in T(\tilde{S}_t)} l(f(\tilde{S}_t), f(\tilde{S}'_t))$$

式中, $l(\cdot)$ 是用来计算两个相似序列之间的对比损失, 为了完成上述的两种对比学习任务, 可以将其联合进行训练:

$$\min_{\theta} \mathcal{L} = \mathcal{L}_1 + \gamma \mathcal{L}_2 + \lambda \|\theta\|_2$$

式中, γ 用来控制 Context-Context 损失的学习权重; λ 是 L2 规范化系数。

4 实验分析

4.1 实验数据

在 Amazon、Yelp、MovieLens、Gowalla 上选取 8 个数据集进行实验, 其中 Amazon 选择 BaBy、Beauty、Movies、Toys、Food 这 5 个不同种类生成的子数据集, 均采用 5-core 版本。Yelp 由于数据量过大, 选取 2018 年之后的数据。MovieLens 选择 1M 版本。表 1 给出每个数据集交互序列、交互项、稀疏性的相关信息。

表 1 实验数据集

DataSet	Sequences	Items	Sparsity/%
Baby	19 445	7 011	99.91
Beauty	22 363	12 068	99.94
Movies and TV	2 088 620	186 349	99.99
Toys and Games	1 342 911	283 394	99.99
Grocery and Gourmet Food	14 684	8 687	99.90
Yelp-2018	104 072	54 034	99.92
ML-1M	6 040	3 704	95.58
Gowalla	76 894	304 440	99.77

4.2 评价指标及基准模型

由于 CSFTRec 的损失函数是在 BPR 的基础上进一步进行推导得出的, 所以采用 Top-K 推荐的召回命中率 (hit ratio, HR) 和归一化累积增益 (normalized discounted cummulative gain, NDCG) 作为模型评价指标^[32]:

$$HR@k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} I(g_u \leq k)$$

$$NDCG@k = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{I(g_u \leq k)}{\log_2(g_u + 1)}$$

式中, $I(\cdot)$ 是一个判别函数, 会根据是否满足条件返回 0 和 1。指标 HR 和 NDCG 越大, 代表推荐结果越准确, 算法效果越好。

4.3 实验方法

为了探明对比学习任务下的推荐算法和基于傅里叶变换的噪声过滤是否具有成效, 现将实验分为 3 组。第一组选取 SASRec、FMLP-Rec 以及没有过滤模块的 CSFTRec+ 作为基准模型, 主要用于探究过滤器模块对模型的具体影响, 其中 FMLP-Rec 是在 SASRec 上添加了自学习过滤器。第二组选择 BPRMF、POP、ComiRec^[33]、FPMC^[34]、GRU4Rec^[35] 作为基准模型。第三组选择 SASRec、BERT4Rec^[36]、TISASRec^[37] 为基准模型, 这 3 个模型是以自注意力机制为核心的推荐算法, BERT4Rec 和 TISASRec 是在 SASRec 的基础上进行了一定的改进, 其中 BERT4Rec 引入了双向 Transformer 编码器, TISASRec 引入了对于序列时间间隔的编码。上述实验全部基于 Python3.8 以及 Pytorch1.8.1 框架进行实现, 运行环境为 RTX 3090(24 GB)、15 核 AMD EPYC 7543 32-Core Processor 以及 80 GB 内存。模型均使用 Adam 优化器, 学习率统一设置为 1×10^{-4} , 每个模型均不进行预训练。

4.4 实验结果与分析

表 2 为过滤器模块对算法的影响, 可见 FMLP-Rec 和 CSFTRec 相较于基础模型在增加过滤器模

块后性能方面有所提升, 即使去除过滤器模块, 对比学习下的联合训练任务在大部分情况下也要表现的更好。此外, 表 3 为 4 个模型的时间性能对比, 在增加过滤器模块后, 必然会增加迭代时间, 但 CSFTRec 的迭代速度要快于普通 SASRec 和 FMLP-Rec, 这是由于新的对比学习损失加快了整体损失的计算, 而单纯的 BPR 损失要花费更多的时间。

表 2 过滤器模块对推荐算法的影响

DataSets	Methods	Metric			
		HR@5	HR@10	NDCG@5	NDCG@10
Yelp	SASRec	0.368 5	0.498 2	0.259 9	0.301 8
	FMLP-Rec	0.360 7	0.493 9	0.253 4	0.296 4
	CSFTRec+	0.378 0	0.515 0	0.263 2	0.307 5
	CSFTRec	0.382 4	0.521 1	0.268 0	0.312 8
	SASRec	0.702 0	0.805 0	0.550 2	0.583 8
MovieLens	FMLP-Rec	0.713 9	0.809 8	0.562 0	0.593 1
	CSFTRec+	0.703 8	0.798 3	0.553 5	0.584 4
	CSFTRec	0.715 4	0.812 9	0.566 8	0.598 6
Food	SASRec	0.385 5	0.474 2	0.287 6	0.316 4
	FMLP-Rec	0.396 3	0.483 7	0.300 4	0.328 7
	CSFTRec+	0.404 9	0.489 9	0.310 7	0.338 3
	CSFTRec	0.401 7	0.479 9	0.309 1	0.335 1
Gowalla	SASRec	0.617 4	0.738 5	0.477 1	0.516 3
	FMLP-Rec	0.618 2	0.737 2	0.478 4	0.517 0
	CSFTRec+	0.642 3	0.749 8	0.500 3	0.535 2
	CSFTRec	0.669 8	0.776 3	0.524 1	0.558 8
Movies	SASRec	0.694 9	0.781 6	0.572 9	0.601 2
	FMLP-Rec	0.691 0	0.775 9	0.571 3	0.598 9
	CSFTRec+	0.707 6	0.786 9	0.595 4	0.621 2
	CSFTRec	0.704 8	0.793 1	0.595 5	0.620 3

表 3 模型时间性能对比

算法	数据集			/s
	Gowalla	Yelp2018	ML-1M	
SASRec	188	41	40	
FMLP-Rec	240	53	52	
CSFTRec	120	34	26	
CSFTRec+	90	23	20	

基于表 4 和表 5 的结果, 可以得出非序列推荐算法性能明显要更差一些, 这表明序列推荐算法在对比学习任务的 Encoder 选择中应该有比较高的优先级。其次, SASRec、TISASRec、BERT4Rec 这 3 个模型在每个数据集上的表现均强于传统的 RNN 架构算法和基于马尔科夫链的推荐算法, 这是由于自注意力机制的编码器具有更多的参数, 可以更好的捕获序列特征。CSFTRec 在大部分情况下要表现出更强的性能, 且训练过程中投入的数据越多, CSFTRec 的性能提升越高, 这是由于对比学习过程中使用了数据增强, 虽然所有模型在同样的数据集上进行训练, 但 CSFTRec 实际上要比监督学习任务的训练数据更多。

表 4 CSFTRec 与非序列推荐算法和序列推荐算法的性能比较

DataSets	Methods	Metric			
		HR@5	HR@10	NDCG@5	NDCG@10
Baby	POP	0.204 6	0.314 4	0.131 9	0.167 1
	BPRMF	0.229 1	0.339 8	0.151 2	0.186 8
	FPMC	0.260 6	0.371 4	0.181 2	0.216 8
	ComiRec	0.267 9	0.388 5	0.180 0	0.218 9
	GRU4Rec	<u>0.291 9</u>	<u>0.415 8</u>	<u>0.197 9</u>	<u>0.237 9</u>
	CSFTRec	0.300 8	0.422 2	0.208 7	0.244 3
Beauty	POP	0.173 5	0.290 0	0.114 1	0.151 6
	BPRMF	0.363 6	0.464 9	0.264 3	0.297 1
	FPMC	0.344 8	0.428 6	<u>0.264 8</u>	0.291 6
	ComiRec	0.367 5	0.479 0	0.264 6	<u>0.300 8</u>
	GRU4Rec	0.326 8	0.438 3	0.234 2	0.270 1
	CSFTRec	0.376 2	<u>0.469 7</u>	0.290 1	0.320 4
Food	POP	0.206 5	0.334 7	0.130 1	0.171 3
	BPRMF	0.354 7	0.460 9	0.248 4	0.282 9
	FPMC	0.359 1	0.438 1	<u>0.279 6</u>	<u>0.305 1</u>
	ComiRec	<u>0.366 3</u>	<u>0.474 4</u>	0.259 6	0.294 7
	GRU4Rec	0.362 6	0.472 4	0.257 8	0.293 3
	CSFTRec	0.401 7	0.479 9	0.309 1	0.335 1
Toys	POP	0.338 5	0.408 3	0.276 6	0.299 1
	BPRMF	0.376 0	0.460 1	0.281 9	0.309 2
	FPMC	0.405 1	0.491 0	0.315 1	0.342 9
	ComiRec	<u>0.437 6</u>	<u>0.527 7</u>	0.338 0	<u>0.367 2</u>
	GRU4Rec	0.395 8	0.513 3	0.284 9	0.323 0
	CSFTRec	0.445 9	0.575 1	<u>0.325 2</u>	0.367 0
Movies	POP	0.635 3	0.757 1	0.500 7	0.540 2
	BPRMF	0.456 0	0.563 1	0.373 9	0.452 3
	FPMC	0.630 0	0.746 1	0.498 3	0.536 0
	ComiRec	<u>0.661 3</u>	<u>0.770 4</u>	<u>0.524 1</u>	<u>0.559 5</u>
	GRU4Rec	0.601 3	0.734 5	0.451 5	0.494 7
	CSFTRec	0.704 8	0.793 1	0.595 5	0.620 3

表 5 CSFTRec 与以自注意力机制为核心的 Transformer 系序列推荐算法的性能比较

DataSets	Methods	Metric			
		HR@5	HR@10	NDCG@5	NDCG@10
Baby	SASRec	0.290 8	0.409 9	0.202 9	0.241 3
	FMLP-Rec	0.282 8	0.393 1	0.196 6	0.232 1
	CSFTRec+	<u>0.297 1</u>	<u>0.417 9</u>	<u>0.205 3</u>	0.247 7
	CSFTRec	0.300 8	0.422 2	0.208 7	<u>0.244 3</u>
Beauty	SASRec	0.367 3	0.460 1	0.277 6	0.307 5
	FMLP-Rec	<u>0.371 1</u>	<u>0.464 4</u>	<u>0.281 0</u>	<u>0.311 0</u>
	CSFTRec+	0.368 3	0.459 2	0.280 1	0.309 8
	CSFTRec	0.376 2	0.469 7	0.290 1	0.320 4
Food	SASRec	0.385 5	0.474 2	0.287 6	0.316 4
	FMLP-Rec	<u>0.390 2</u>	0.482 0	0.293 6	<u>0.322 5</u>
	CSFTRec+	0.389 2	0.477 4	<u>0.293 8</u>	0.322 3
	CSFTRec	0.401 7	<u>0.479 9</u>	0.309 1	0.335 1
Toys	SASRec	0.368 3	0.450 5	0.290 4	0.316 9
	FMLP-Rec	0.406 2	0.488 8	0.320 6	0.347 3
	CSFTRec+	<u>0.419 3</u>	<u>0.510 2</u>	<u>0.323 3</u>	<u>0.352 8</u>
	CSFTRec	0.445 9	0.575 1	0.325 2	0.367 0
Movies	SASRec	0.694 9	0.781 6	0.572 9	0.601 2
	FMLP-Rec	0.708 4	0.785 6	<u>0.580 6</u>	<u>0.609 4</u>
	CSFTRec+	0.701 1	<u>0.790 8</u>	0.578 1	0.607 2
	CSFTRec	<u>0.704 8</u>	0.793 1	0.595 5	<u>0.620 3</u>

此外, 图 2 为 CSFTRec 与 InfoNCE 的训练结果比较, 明显可见 InfoNCE 的效果不如引入了

Context-Context 损失的联合训练任务, 且 CSFTRec 有着更快的收敛速度, 这要归功于序列进入 Encoder 之前的噪声过滤, 使得自注意力机制能够更快的学习到序列特征, 而不会过多的关注那些无意义的干扰。

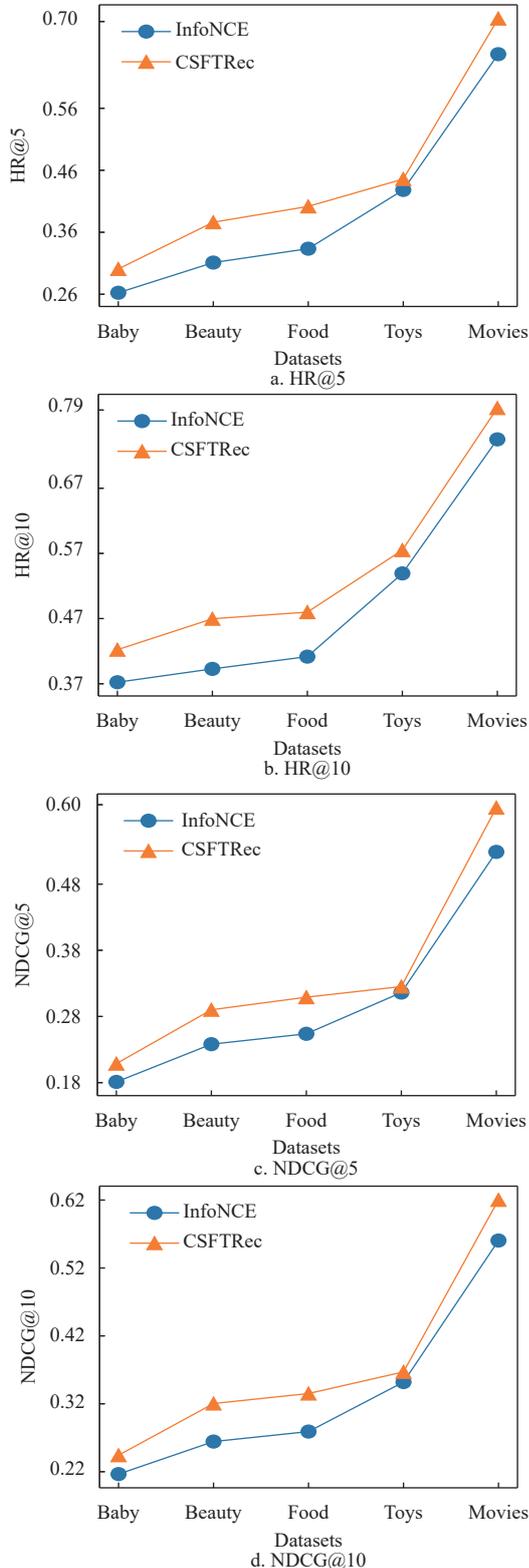
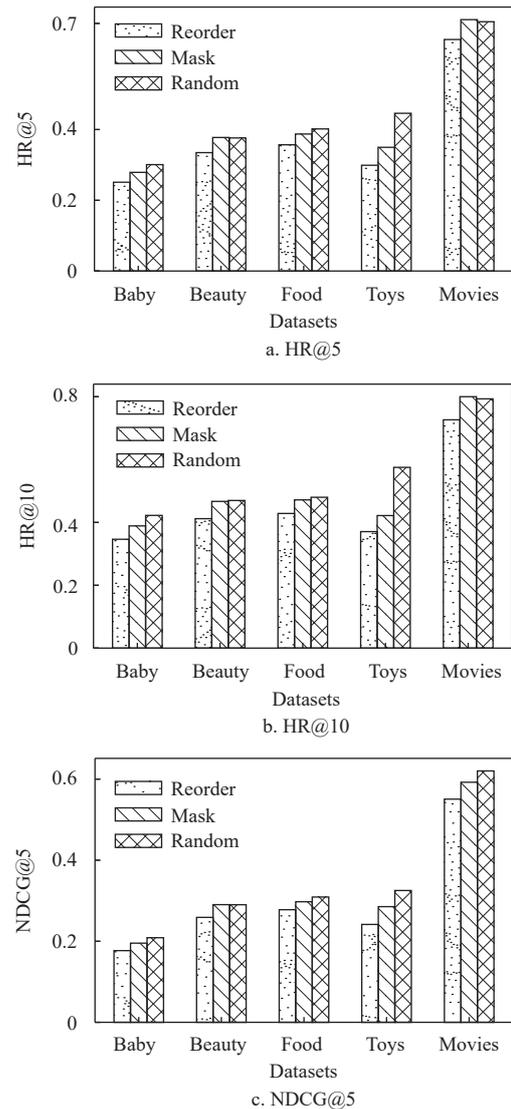


图 2 InfoNCE 与 CSFTRec 在不同数据集上的表现

4.5 参数影响分析

为了使 CSFTRec 的性能进一步提升, 探究了不同的超参数对 CSFTRec 产生的影响。在无监督学习任务中, 数据增强的方式是否合理会直接影响到模型的最终效果, 因此将本文所提到的两种数据增强方式单独应用到 CSFTRec 中, 将其在 5 个数据集上的表现与使用随机数据增强的 CSFTRec 进行对比。如图 3 所示, 可以明显看出掩盖方式在每个数据集上的表现都要强于置换, 这是由于掩盖会使自注意力机制去重点学习序列数据当中未被掩盖部分的序列特征, 多次的随机掩盖, 会使序列的每一个子序列特征可以被充分学习到, 所以会有一个较好的性能表现。但从另一个方面来看, 使用了混合数据增强方式的 CSFTRec 表现却又不输于掩盖, 也就是说置换并没有拉低 CSFTRec 的性能, 反而使其在大多数数据集上的表现要比随机掩盖更强, 这也证明了使用两种数据增强方式混合处理数据的可行性。



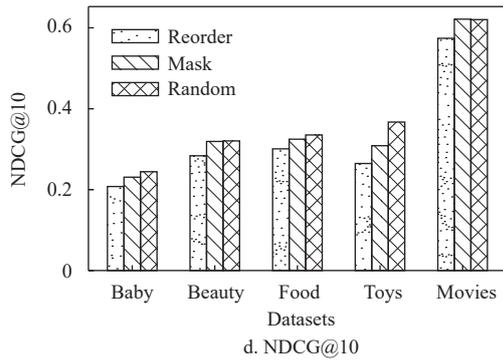


图 3 不同数据增强方式对 CSFTRes 的影响

此外, 在 Food 和 Toys 两个数据集上研究了 BatchSize 和负样本个数对模型的影响。图 4 为不同的 BatchSize 对 CSFTRec 的影响, 可以看到 CSFTRec 更受益于较大的 BatchSize, 这是由于在 Context-Context 损失中, 较大的训练 Batch 会提高多个序列共享一个目标的概率, 从而使得 CSFTRec 有更好的表现。在图 5 中, 随着负样本个数的增加, 两个数据集上的表现大不相同。在小数据集上, 只保留一个负样本可以最大化模型性能, 但在大数据集上, 模型却又受益于更多的负样本, 这是由于联合训练任务在数据较少时更依赖于 Context-Context 损失, 在数据较多时, 才会逐渐从对比损失中获得训练增益。

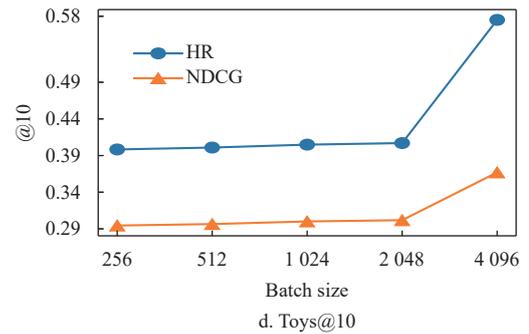
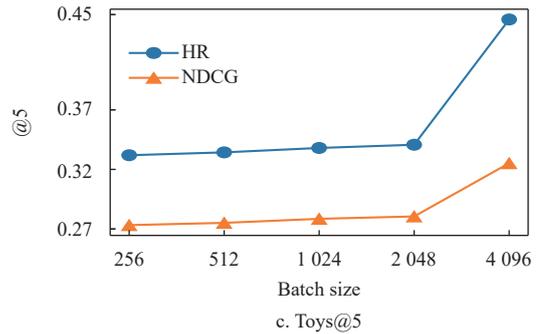
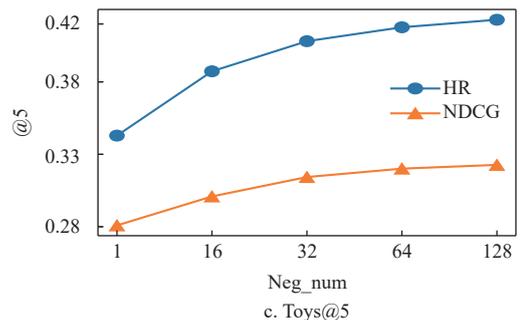
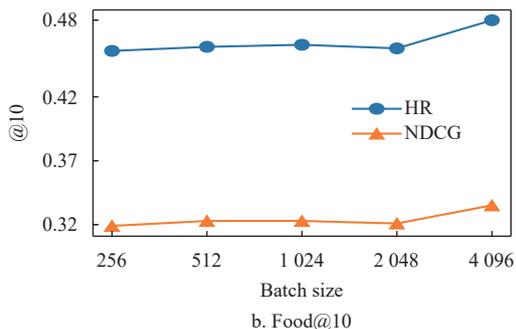
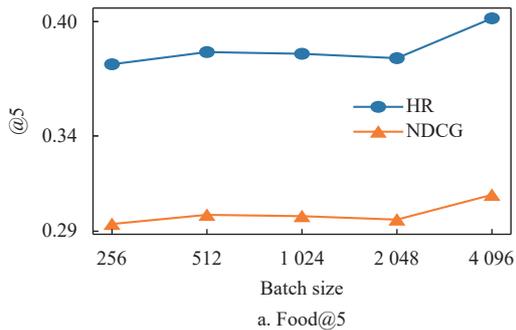
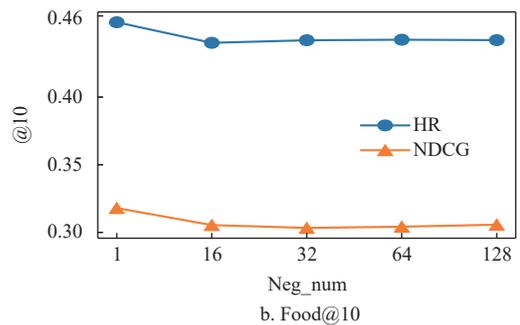
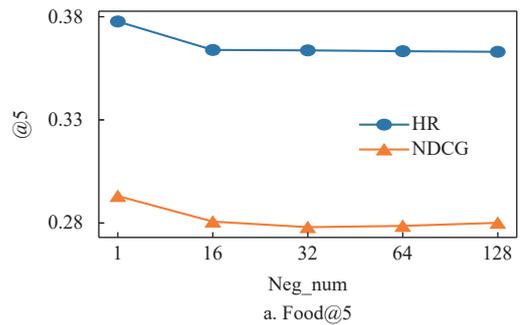


图 4 Food 和 Toys 上 BatchSize 对 CSFTRec 的影响



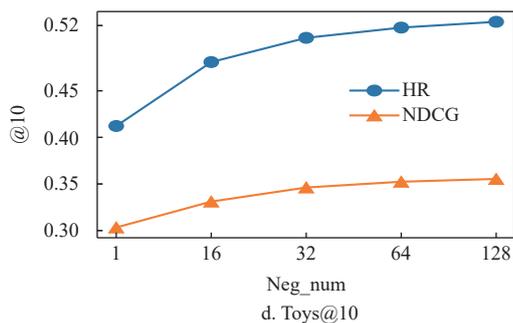


图 5 不同负样本数量对 CSFTRec 的影响

总体而言,在对比学习任务中,不论是数据增强方式还是 BatchSize 和负样本个数均会对模型有较明显的影响,如果选择了合理的参数配置,就可以保证模型逐渐收敛并获得一个较好的模型表现。

5 结束语

序列推荐算法的研究有着广泛的应用背景和实用性。针对现有序列推荐算法在对序列数据进行特征建模时容易受到噪声数据的影响,且传统的对比损失在推荐任务中表现不佳的问题,本文基于自注意力机制的编码器,使用傅里叶变换对序列数据进行噪声过滤,最大化编码器对序列数据的特征捕获能力,通过进一步引入 Context-Context 损失与由 BPR 推广的对比损失联合训练,提高推荐算法的性能。最后在 5 个亚马逊工业数据集上进行实验并与当下流行的算法模型进行对比,验证了本文所提出的序列推荐算法拥有较好的性能,对于对比学习在推荐算法领域中的应用有着重要意义。

参 考 文 献

- [1] ZHANG S, YAO L, SUN A, et al. Deep learning based recommender system: A survey and new perspectives[J]. ACM Computing Surveys (CSUR), 2019, 52(1): 1-38.
- [2] FAN H, WU K, PARVIN H, et al. A hybrid recommender system using knn and clustering[J]. International Journal of Information Technology & Decision Making, 2021, 20(2): 553-596.
- [3] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[EB/OL]. [2022-5-29]. <https://arxiv.org/abs/1205.2618>.
- [4] SINGH A, THAKUR N, SHARMA A. A review of supervised machine learning algorithms[C]//2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom). [S.l.]: IEEE, 2016: 1310-1315.
- [5] SHANI G, HECKERMAN D, BRAFMAN R I, et al. An MDP-based recommender system[J]. Journal of Machine Learning Research, 2005, 6(9): 1265-1295.
- [6] POLYZOU A, NIKOLAKOPOULOS A N, KARYPIS G. Scholars walk: A Markov chain framework for course recommendation[J]. International Educational Data Mining Society, 2019: 396-401.
- [7] TANG J, WANG K. Personalized top-*n* sequential recommendation via convolutional sequence embedding[C]//Proceedings of the 11th ACM International Conference on Web Search and Data Mining. [S.l.]: ACM, 2018: 565-573.
- [8] DONKERS T, LOEPP B, ZIEGLER J. Sequential user-based recurrent neural network recommendations [C]//Proceedings of the 11th ACM Conference on Recommender Systems. [S.l.]: ACM, 2017: 152-160.
- [9] BHARADHWAJ H, JOSHI S. Explanations for temporal recommendations[J]. KI-Künstliche Intelligenz, 2018, 32(4): 267-272.
- [10] CHEN X, XU H, ZHANG Y, et al. Sequential recommendation with user memory networks [C]//Proceedings of the 11th ACM International Conference On Web Search and Data Mining. [S.l.]: ACM, 2018: 108-116.
- [11] WU S, TANG Y, ZHU Y, et al. Session-Based recommendation with graph neural networks [C]//Proceedings of the AAAI Conference on Artificial Intelligence. [S.l.]: AAAI, 2019: 346-353.
- [12] WANG Z, WEI W, CONG G, et al. Global context enhanced graph neural networks for session-based recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. [S.l.]: ACM, 2020: 169-178.
- [13] QIU R, LI J, HUANG Z, et al. Rethinking the item order in session-based recommendation with graph neural networks[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. [S.l.]: ACM, 2019: 579-588.
- [14] CHEN T, WONG R C W. Handling information loss of graph neural networks for session-based recommendation [C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. [S.l.]: ACM, 2020: 1172-1180.
- [15] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30.
- [16] LI J, REN P, CHEN Z, et al. Neural attentive session-based recommendation[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. [S.l.]: ACM, 2017: 1419-1428.
- [17] VAN DEN O A, LI Y, VINYALS O. Representation learning with contrastive predictive coding[EB/OL]. [2022-7-10]. <https://arxiv.org/abs/1807.03748>.
- [18] CHEN X, FAN H, GIRSHICK R, et al. Improved baselines with momentum contrastive learning[EB/OL]. [2022-7-28]. <https://arxiv.org/abs/2003.04297>.
- [19] HE K, FAN H, WU Y, et al. Momentum contrast for unsupervised visual representation learning [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. [S.l.]: IEEE, 2020: 9729-

- 9738.
- [20] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space [EB/OL]. [2022-8-10]. <https://arxiv.org/abs/1301.3781>.
- [21] KANG W C, MCAULEY J. Self-Attentive sequential recommendation[C]//2018 IEEE International Conference on Data Mining (ICDM). [S.l.]: IEEE, 2018: 197-206.
- [22] WANG C, MA W, CHEN C. Sequential recommendation with multiple contrast signals[J]. *ACM Transactions on Information Systems (TOIS)*, 2022, 41(1): 1-27.
- [23] CHEN T, KORNBLITH S, NOROUZI M, et al. A simple framework for contrastive learning of visual representations[C]//International Conference on Machine Learning. [S.l.]: PMLR, 2020: 1597-1607.
- [24] JAIN P, JAIN A, ZHANG T, et al. Contrastive code representation learning[EB/OL]. [2022-8-28]. <https://arxiv.org/abs/2007.04973>.
- [25] WU Z, WANG S, GU J, et al. Clear: Contrastive learning for sentence representation[EB/OL]. [2022-6-19]. <https://arxiv.org/abs/2012.15466>.
- [26] SANG Y F, WANG D, WU J C, et al. The relation between periods' identification and noises in hydrologic series data[J]. *Journal of Hydrology*, 2009, 368(1-4): 165-177.
- [27] STAMMLER K. SeismicHandler-programmable multi-channel data handler for interactive and automatic processing of seismological analyses[J]. *Computers & Geosciences*, 1993, 19(2): 135-140.
- [28] UNUMA M, ANJYO K, TAKEUCHI R. Fourier principles for emotion-based human figure animation [C]//Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques. [S.l.]: ACM, 1995: 91-96.
- [29] ZHOU K, YU H, ZHAO W X, et al. Filter-Enhanced MLP is all you need for sequential recommendation [C]//Proceedings of the ACM Web Conference 2022. [S.l.]: ACM, 2022: 2388-2399.
- [30] BA J L, KIROS J R, HINTON G E. Layer normalization[EB/OL]. [2022-6-29]. <https://arxiv.org/abs/1607.06450>.
- [31] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. *The Journal of Machine Learning Research*, 2014, 15(1): 1929-1958.
- [32] ZHAO W X, CHEN J, WANG P, et al. Revisiting alternative experimental settings for evaluating top-*N* item recommendation algorithms[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. [S.l.]: ACM, 2020: 2329-2332.
- [33] CEN Y, ZHANG J, ZOU X, et al. Controllable multi-interest framework for recommendation[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. [S.l.]: ACM, 2020: 2942-2951.
- [34] 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. [S.l.]: ACM, 2010: 811-820.
- [35] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-Based recommendations with recurrent neural networks[EB/OL]. [2022-9-11]. <https://arxiv.org/abs/1511.06939>.
- [36] SUN F, LIU J, WU J, et al. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. [S.l.]: ACM, 2019: 1441-1450.
- [37] LI J, WANG Y, MCAULEY J. Time interval aware self-attention for sequential recommendation[C]//Proceedings of the 13th International Conference on Web Search and Data mining. [S.l.]: ACM, 2020: 322-330.

编辑 叶芳