

基于对比学习和元优化学习的序列推荐方法

谢林泽¹, 陈平华^{1*}, 邓柏城²

(1. 广东工业大学 计算机学院, 广东 广州 510006;

2. 广东科学技术职业学院 经济管理学院, 广东 珠海 519090)

摘要: 序列推荐是根据用户和项目的历史交互记录对用户兴趣建模, 进行下一项目推荐。对比学习 (CL) 作为一种辅助信息能够有效地提高推荐模型质量, 但现有基于对比学习的序列推荐方法采取随机数据增强方式存在的效果不稳定及难以泛化的问题, 为此, 提出了一种基于对比学习和元优化学习的序列推荐方法。首先, 在数据增强环节, 根据序列中项目之间的时间间隔为序列生成数据分布更加均匀的数据增强视图; 其次, 构建可学习的模型增强模块, 用于捕获数据增强视图中潜在的语义信息, 增强模型的泛化能力; 最后, 为解决数据增强模块和模型增强模块之间不同优化目标问题, 使用元优化学习方法优化更新两个模块之间的参数, 进而完成推荐。在 Beauty、Sports 和 Yelp 等三个公开数据集上的实验结果显示, 在召回率和归一化折损累计增益指标上, 相较于其它基线模型, CLMLRec 均有显著提升, 表明该模型具有良好的推荐性能。

关键词: 序列推荐; 对比学习; 元优化学习; 数据增强; 模型增强

中图分类号: TP391.3

文献标识码: A

文章编号: 1673-629X(2024)10-0148-08

doi:10.20165/j.cnki.ISSN1673-629X.2024.0192

Sequential Recommendation Method Based on Contrastive Learning and Meta-optimized Learning

XIE Lin-ze¹, CHEN Ping-hua^{1*}, DENG Bai-cheng²

(1. School of Computer Science, Guangdong University of Technology, Guangzhou 510006, China;

2. School of Economics and Management, Guangdong Institute of Science and Technology, Zhuhai 519090, China)

Abstract: Sequential recommendation is to model user interest based on historical interaction records between users and projects, and recommend the next project. As a kind of side information, contrast learning (CL) can effectively improve the quality of recommendation models, but the existing sequential recommendation methods based on contrast learning are unstable and difficult to generalize by using random data enhancement. To address the above problem, a sequence recommendation method based on contrastive learning and meta-optimized learning is proposed. Firstly, in the data augmentation step, the data augmentation view with more uniform data distribution is generated for the sequence according to the time interval between the items in the sequence. Secondly, a learnable model augmentation module is constructed to capture the potential semantic information in the data augmentation view and enhance the generalization ability of the model. Finally, in order to solve the problem of different optimization objectives between the data augmentation module and the model augmentation module, the meta-optimized learning is used to optimize and update the parameters between the two modules to complete the recommendation. Experimental results on three publicly available datasets, including Beauty, Sports and Yelp, showed that CLMLRec has significantly improved in terms of recall and NDCG compared with other baseline models, indicating that the model has good recommendation performance.

Key words: sequential recommendation; contrastive learning; meta-optimized learning; data augmentation; model augmentation

0 引言

序列推荐是一类考虑用户购买行为时序性特征的推荐算法, 其中建模商品的序列交互特征对提高推荐

结果准确性具有非常重要的意义^[1-2]。为缓解推荐系统的数据稀疏和交互噪声问题, 各种辅助信息被引入到推荐系统中使用。最近, 各种对比自监督学习方法

收稿日期: 2024-02-23

修回日期: 2024-06-26

基金项目: 广东省重点领域研发计划项目(2023B1111050010, 2020B0101100001)

作者简介: 谢林泽(2000-), 男, 硕士研究生, 研究方向为推荐系统; 通讯作者: 陈平华(1967-), 男, 教授, 硕导, CCF 杰出会员(47269D)、CCF 广州委员, 研究方向为大数据, 人工智能。

已经展示出它们在这些中高效学习模型表征的卓越能力。

现有的基于对比学习的推荐模型^[3-7]虽然取得了不错的效果,但大多数的对比模型在数据增强中采用随机数据增强方式,没有考虑到序列中项目之间的分布特性,如时间间隔。此外,针对现有数据增强方法可能会导致样本语义信息偏差的问题,为了保留语义信息,各种模型增强方式先后被提出。但大多数模型都是在数据增强或模型增强上手动执行随机增强操作,仅仅依靠这些不可学习的增强模块,无法在不同的数据集自适应寻找到最合适的增强操作^[8]。

受元学习^[9-14]启发,该文提出一种基于对比学习和元优化学习的序列推荐方法(sequential recommendation method based on contrastive learning and meta-optimized learning, CLMLRec)。贡献总结如下:

(1)在对比学习的数据增强模块中,使用了基于时间间隔的数据增强方式,针对不同用户序列,构建分布更加平衡的数据增强视图。

(2)构建可学习的模型增强模块,自适应捕获隐藏在数据增强视图中的语义信息特征,并提出应用元优化学习方法,解决数据增强与模型增强模块不同优化目标的问题。

(3)在 Beauty, Sports 和 Yelp 三个公开数据集上进行实验,验证了该方法的有效性。

1 相关工作

1.1 序列推荐

序列推荐的目的是给用户的历史交互行为建模,并根据用户过去的行为序列来预测将会发生的下一次交互^[15]。在早期,基于马尔可夫链的方法将当前用户行为序列映射到马尔可夫链中,根据该用户上一次交互来预测下一时刻交互过程中用户的行为^[16]。深度神经网络兴起后,鉴于其优秀的表现能力,RNN、GRU 等神经网络模型先后应用于序列推荐中^[17-18]。鉴于注意力机制对序列中各项目重要程度出色的建模能力,Kang 等人在 SASRec 模型^[19]中提出了一种基于自注意力的序列推荐方法,自适应地为序列中的每个项目分配权重。Sun 等人提出完全基于自注意力机制的深度神经网络模型 BERT4Rec^[20],可以使用位置嵌入模块来捕获用户交互序列中不同项目的重要性,成功解决了 RNN 在建模长序列时出现的梯度消失问题^[21]。

1.2 对比学习

对比学习是一种自监督学习方法,通过最大化同一实例的两个不同增强视图之间的一致性来训练编码

器,从而提高嵌入表达的鲁棒性^[22]。Xu 等人提出的 CL4SRec^[3]采用三种数据增强方式(裁剪/遮掩/重排)来构建自监督信息,并使用具有对比自监督目标的多任务训练框架来增强用户表征;Liu 等人提出 CoSeRec^[4]将数据增强(插入和替换)的项目相似度信息与对比学习目标相结合,以最大化增强序列的一致性。

只依赖数据增强方式的对比学习并不能保留足够的语义信息,最近出现了通过修改或增强模型本身来生成不同的增强视图以构建对比损失。如 DuoRec^[5]提出了一种采样策略来制定正样本,并使用 dropout 进行模型级增强;SRMA^[6]引入了三种模型增强方法(即“神经遮掩”“层丢弃”和“编码器补充”),并将它们与数据增强相结合以构建视图对;ECGAN-Rec^[7]提出了序列增强模块和对比 GAN 模块来实现数据增强和模型增强的结合。

1.3 元学习

元学习^[23]利用以往的知识经验来指导新任务的学习,具有学会学习的能力。大多数现有的元学习推荐任务都采用基于优化的策略,即初始化元优化器的全局共享参数,然后通过多个任务学习该元优化器的初始参数,使新任务能快速适应模型的最佳点。例如, Lee 等人提出的 MeLU 模型^[10],目的是学习神经网络的全局初始化参数作为先验知识。Lu 等人^[11]通过局部和全局更新整个异构信息网络来学习泛化良好的初始参数。为了解决新用户的数据稀疏问题,CBML^[12]将基于优化的元学习应用于基于自注意力机制的序列推荐模型,设计了一个基于集群的元学习框架,用于在相似的序列任务之间传递共享的元知识。同样,Huang 等人提出了一种基于 MAML^[13]框架的 metaCSR 模型^[14],用于全局学习不同任务的序列推荐之间的参数。

2 问题描述

该文使用 U 和 I 表示用户和项目的集合。用 $S_u = (v_1^u, \dots, v_j^u, \dots, v_n^u)$ 表示用户 u 的历史交互序列,其中 v_j^u 表示用户 u 在时间步 j 交互的项目, n 为序列长度。每个交互序列 S_u 都有对应的项目时间间隔 $T_u = (t_1, \dots, t_j, \dots, t_{n-1})$,其中 t_j 为项目 v_j 和 v_{j+1} 时间戳之间的时间间隔。模型的目标是根据用户的交互序列 S_u 和 T_u ,预测该用户在 $n+1$ 时可能交互的项目。

3 CLMLRec 模型

CLMLRec 模型框架如图 1 所示。框架由四部分构成:数据增强、用户表征、模型增强、元优化学习策略。

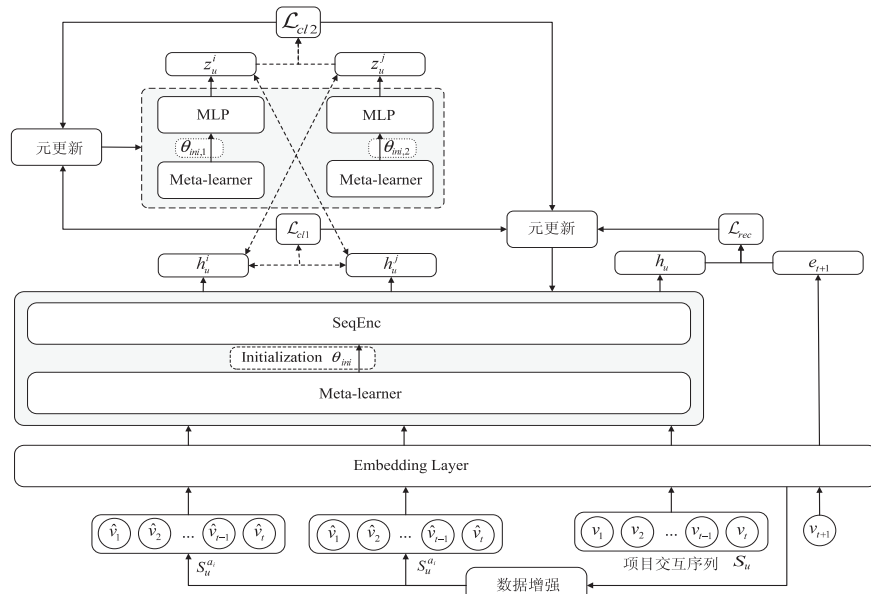


图 1 CLMLRec 的框架

3.1 基于时间间隔的数据增强

CL4SRec^[3] 和 CoSeRec^[4] 提出的五种数据增强方式需要很多的经验知识,不同数据集很难泛化,同时,用户经验不同时,数据增强的效果很难保证。该文提

出五种基于时间间隔的增强方式,可选择不同增强方式生成目标项生成更健壮的增强序列。如图 2,可以构建同一序列的不同视图,但仍然保持隐藏在用户历史行为的兴趣偏好。

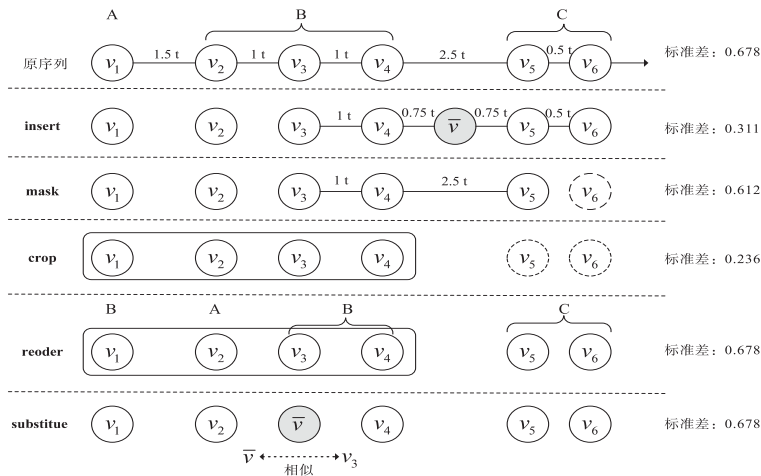


图 2 基于时间间隔的数据增强方式

(1) 基于时间间隔的裁剪:对于每个用户的历史交互序列 S_u ,选取长度为 $L_c = \eta N (\eta \in [0, 1])$ 的连续子序列,然后分别计算所有子序列的时间间隔的标准差,保留标准差值最小的连续子序列。可表示为:

$$S_u^{crop} = \operatorname{argmin} \operatorname{std}(v_i, v_{i+1}, \dots, v_{i+L_c-1}) \quad (1)$$

(2) 基于时间间隔的遮掩:对于每个用户历史交互序列 S_u ,按时间间隔标准差从小到大进行排序,选择 $\operatorname{top} - \lambda k (k = \lambda N (\lambda \in [0, 1]))$ 个时间间隔的周围项目进行遮掩。遮掩会增大项目之间的时间间隔,从而增大总序列的标准差,所以仅对较小的时间间隔的周围项目进行遮盖,使序列在整体上更为均匀。可表示为:

$$S_u^{Mask} = [\hat{v}_1, \hat{v}_2, \dots, \hat{v}_n]$$

$$\hat{v}_i = \begin{cases} v_i, & t \notin \operatorname{argmin} - k - \operatorname{indexes}(\operatorname{ASC}(T_u)) \\ [\operatorname{mask}], & t \in \operatorname{argmin} - \\ & k - \operatorname{indexes}(\operatorname{ASC}(T_u)) \end{cases}$$

(2)

(3) 基于时间间隔的重排:和基于时间间隔的裁剪方法一样,选取标准差值最小的连续子序列,其长度为 $L_r = \mu N (\mu \in [0, 1])$,然后进行洗牌得到新的子序列 $[\hat{v}_i, \hat{v}_{i+1}, \dots, \hat{v}_{i+L_r-1}] = \operatorname{argmin} \operatorname{std}(v_i, v_{i+1}, \dots, v_{i+L_r-1})$,从而生成新的序列。可表示为:

$$S_u^{reorder} = [v_1, v_2, \dots, \hat{v}_i, \dots, \hat{v}_{i+L_r-1}, \dots, v_n] \quad (3)$$

(4) 基于时间间隔的替换:选取目标项的方法与基于时间间隔的遮掩一样,都是选取时间间隔小的周围项目,不同的是使用与目标项最相似的项目替换选

取的目标项,更容易保持与原始序列相似的偏好模式。可表示为:

$$S_u^{\text{substitute}} = [v_1, v_2, \dots, \bar{v}_{\text{idx}_i}, \dots, v_n] \quad (4)$$

$$\text{idx}_i \in \text{argmin} - n - \text{indexes}(\text{ASC}(T_u))$$

其中, \bar{v}_{idx_i} 是和 v_{idx_i} 最相似的项目。

(5) 基于时间间隔的插入: 计算所有序列的时间间隔的标准差, 然后从大到小进行排序, 选取 top- m 个时间间隔位置 $\{\text{idx}_1, \text{idx}_2, \dots, \text{idx}_m\}$, 其中 $h \in \gamma N$ ($\gamma \in [0, 1]$), 插入用户未交互过且与上下文项目相似的项目。插入会缩小项目之间的时间间隔, 从而减小总序列的标准差, 所以要在较大的时间间隔进行插入, 使序列在整体上更为均匀。可表示为:

$$S_u^{\text{insert}} = [v_1, v_2, \dots, \bar{v}_{\text{idx}_i}, v_{\text{idx}_i}, \dots, v_n] \quad (5)$$

$$\text{idx}_i \in \text{argmin} - m - \text{indexes}(\text{DESC}(T_s))$$

其中, $i \in \{1, 2, \dots, h\}$, 并且 \bar{v}_{idx_i} 是和 v_{idx_i} 最相似的项目, 增强后的序列 S_u^{insert} 长度为 $|S_u| + h$ 。

(6) 根据序列长度选择增强方式: 用户的历史交互序列是趋向于长尾分布的, 考虑到项目扰动对短序列的影响较大, 所以要对短序列选取特定的增强方式^[20]。使用超参数 K 来决定一个序列是长序列或短序列, 然后应用如下的数据增强方式:

$$S_u^a = \begin{cases} a(S_u), a \in \{S, I, M\}, |S_u| \leq K \\ a(S_u), a \in \{S, I, M, C, R\}, |S_u| > K \end{cases} \quad (6)$$

其中, a 是从增强方式集合中随机选取的一种增强操作。

3.2 用户表征

用户表征主要由嵌入层和表征学习层组成。

(1) 嵌入层: 首先, 构造一个 embedding 矩阵 $V \in R^{n \times d}$, 对于每个用户序列 $s = [v_1, v_2, \dots, v_i]$ 表示为 $s = [v_1, v_2, \dots, v_i]$, 其中 v_i 是嵌入向量。为了保持序列的时间顺序, 构造一个位置编码矩阵 $P \in R^{T \times d}$, 其中 T 的大小限制用户历史序列的最大长度。然后将项目 embedding 和位置编码进行相加得到 Transformer 每个时间步 t 上交互的输入向量。总的序列 embedding 计算公式如下:

$$e^u = \begin{bmatrix} v_1 + p_1 \\ v_2 + p_2 \\ \dots \\ v_n + p_n \end{bmatrix} \quad (7)$$

其中: $v_i + p_i$ 是在时间步 t 时的交互的输入向量; p_i 是时间步 t 的位置编码。

(2) 表征学习层: 对于给定的序列嵌入向量 e^u , 通过堆叠多层 Encoder 模块学习序列的向量表征 h 。总体公式如下:

$$h = \text{Encoder}_{\theta_e}(e^u) \quad (8)$$

其中, θ_e 为 Encoder 的参数, 初始化为 $\theta_{\text{ini}, e}$; S_u^a 为根据公式 6 得到的增强序列。

3.3 可学习的模型增强

无论是数据增强还是模型增强, 由于其灵活性不够以及强随机性, 需要很多的先验知识和较大的训练批次, 都无法取得较好的效果和泛化能力。该文同时利用数据增强和模型增强的优势, 构建两个自适应和可学习的增强模块来捕获在数据增强视图中潜在的语义信息。

$$z_u^i = \mathcal{G}_{\theta_1}(h_u^i), z_u^j = \mathcal{G}_{\theta_2}(h_u^j) \quad (9)$$

其中, $\mathcal{G}_{\theta_1}(\cdot)$ 和 $\mathcal{G}_{\theta_2}(\cdot)$ 为增强模块, 是使用 GELU (Gaussian Error Linear Unit, 高斯误差线性单元) 作为激活函数的 3 层 MLP (Multi-Layer Perceptron, 多层感知机); θ_1 和 θ_2 是增强模块的参数, 分别初始化为 $\theta_{\text{ini}, 1}$ 和 $\theta_{\text{ini}, 2}$; h_u^i 和 h_u^j 是根据公式 8 生成的两个不同的数据增强视图; z_u^i 和 z_u^j 为计算后得到的两个不同的模型增强视图。可学习的模型增强模块, 一方面缓解了随机数据增强难以泛化的问题, 另一方面通过提取更丰富的信息用于对比学习辅助任务, 缓解了数据稀疏和噪声的问题。

3.4 元优化学习策略

为了利用未标记原始数据的自监督信息来提高推荐的性能, 一般是利用推荐监督任务 and 对比学习辅助任务进行联合优化^[3]。但在本模型中, encoder 的主要任务是学习更好的用户和项目表征用于推荐任务模型, 模型增强模块的主要任务是从数据增强视图中抽取出更加丰富的信息特征用于对比学习, 直接使用联合学习并不能学习得到两个模块最优的参数^[24], 因此使用元优化学习的训练方式交替更新两个模块之间的参数。整个元优化学习的过程可分为三个步骤:

(1) 初始化 encoder 的参数 $\theta_{\text{ini}, e}$ 和模型增强模块的参数 $\theta_{\text{ini}, 1}$ 和 $\theta_{\text{ini}, 2}$, 然后将原序列 S_u 和增强后的序列 S_u^a 、 S_u^a 输入到 encoder 中, 由公式 8 计算生成 h_u 、 h_u^i 和 h_u^j 三个不同的视图。接着计算目标项 $e_{t+1} \in V$ 的推荐概率, 以预测用户 u 在 $t+1$ 时刻点击该目标项的可能性, 具体相应的预测概率 \hat{y} 可以通过以下方式生成:

$$\hat{y}_{u, e_{t+1}} = \text{softmax}(h_u \cdot e_{t+1}^T) \quad (10)$$

其中, $\hat{y}_{u, e_{t+1}} \in R^n$ 。期望用户 u 真正选择的项目 v 的分数 \hat{y} 会更高, 因此采用交叉熵损失函数来优化模型的参数, 具体的目标函数可表示为:

$$\mathcal{L}_{\text{Rec}} = - \sum_{i=1}^{|S^u|} y_{u, e_{t+1}} \log(\hat{y}_{u, e_{t+1}}) + (1 - y_{u, e_{t+1}}) \log(1 - \hat{y}_{u, e_{t+1}}) \quad (11)$$

然后将数据增强视图 h_u^i 和 h_u^j 经过增强模块生成得到 z_u^i 和 z_u^j 两个不同的模型增强视图, 最后联合推荐

损失和对比损失函数反向传播来更新 encoder 的参数 θ_e 直到收敛,具体联合优化的目标函数如下:

$$\mathcal{L}_1 = \mathcal{L}_{\text{Rec}} + \lambda_1 \mathcal{L}_{\text{ssl1}} + \lambda_2 \mathcal{L}_{\text{ssl2}} \quad (12)$$

其中, λ_1 和 λ_2 是用来平衡各损失任务的学习权重的超参数。 $\mathcal{L}_{\text{ssl1}}$ 和 $\mathcal{L}_{\text{ssl2}}$ 代表两种不同的损失函数,它们都是基于带有 softmax 采样的负对数似然损失函数,用来区分两种表示是否来自相同的用户历史序列。具体来说,对于一小批次的序列 $\{S_u\}_{u=1}^N$,对每一个序列 S_u 采用两种不同的增强操作,所以能获得 $2N$ 个增强序列 $[s_{u_1}^{a_1}, s_{u_1}^{a_2}, s_{u_2}^{a_1}, s_{u_2}^{a_2}, \dots, s_{u_N}^{a_1}, s_{u_N}^{a_2}]$ 。将 $(s_{u_1}^{a_1}, s_{u_1}^{a_2})$ 看作正样本,剩下的 $2(N-1)$ 增强样本视为负样本 neg。然后对每一个正样本由公式 8 编码成 $(\tilde{h}_u^{a_1}, \tilde{h}_u^{a_2})$ 。所以具体的损失函数公式如下:

$$\mathcal{L}_{\text{ssl}}(\tilde{h}_u^{a_1}, \tilde{h}_u^{a_2}) = -\log \frac{\exp(\text{sim}(\tilde{h}_u^{a_1}, \tilde{h}_u^{a_2}))}{\exp(\text{sim}(\tilde{h}_u^{a_1}, \tilde{h}_u^{a_2})) + \sum_{\tilde{h} \in \text{neg}} \exp(\text{sim}(\tilde{h}_u^{a_1}, \tilde{h}))} \quad (13)$$

其中, $\text{sim}(\cdot)$ 为内积,用来衡量两个增强视图之间的相似性。因此,第一种只取决于数据增强视图 h_u^i 和 h_u^j 的损失函数 $\mathcal{L}_{\text{ssl1}}$ 可表示为:

$$\mathcal{L}_{\text{ssl1}} = \mathcal{L}_{\text{ssl}}(h_u^i, h_u^j) \quad (14)$$

第二种损失函数 $\mathcal{L}_{\text{ssl2}}$ 由数据增强视图 h_u^i 、 h_u^j 和模型增强视图 z_u^i 、 z_u^j 共同生成,可表示为:

$$\mathcal{L}_{\text{ssl2}} = \mathcal{L}_{\text{ssl}}(\tilde{h}_1, \tilde{z}_2) + \mathcal{L}_{\text{ssl}}(\tilde{h}_2, \tilde{z}_1) + \mathcal{L}_{\text{ssl}}(\tilde{z}_1, \tilde{z}_2) \quad (15)$$

(2)通过第一步学习得到了 encoder 的最优参数,然后利用学习后得到的 encoder 重新编码用户的增强序列,重新生成四个不同的增强视图,并且重新计算损失 $\mathcal{L}_{\text{ssl2}}$,通过反向传播来更新两个增强模块 $\mathcal{G}_{\theta_1}(\cdot)$ 和 $\mathcal{G}_{\theta_2}(\cdot)$ 的参数 θ_1 和 θ_2 直到收敛,具体的损失计算公式如下:

$$\mathcal{L}_2 = \mathcal{L}_{\text{ssl2}} \quad (16)$$

(3)通过第二步学习得到了模型增强模块的最优参数,重新生成两个不同的模型增强视图,然后联合推荐损失和对比损失重新计算 \mathcal{L}_1 ,反向传播来更新 encoder 的参数 θ_e 直到收敛,具体损失计算公式如下:

$$\mathcal{L}_3 = \mathcal{L}_1 \quad (17)$$

利用元优化学习的方法,根据 encoder 的性能表现更新模型增强的参数,通过这样的训练方法,模型增强可以基于相对有限的交互来学习判别增强视图。

4 实验

4.1 数据集

为了评估文中模型的推荐性能,实验采用三个真实世界中具有代表性的数据集。Amazon 数据集收集

来自 Amazon 的用户评论数据,Amazon 是世界上最大的电子商务网站之一,在实验中使用了 Sports 和 Beauty 两个子类别。Yelp 是一个著名的商业推荐数据集。在上述数据集中,丢弃交互次数少于 5 次的非活跃用户序列数据^[2,4-5]。处理后的数据集的具体信息如表 1 所示。

表 1 数据集统计

数据集	用户数	物品数	交互数	平均长度	稀疏率/%
Beauty	22 363	12 101	198 502	8.87	99.92
Sports	35 598	18 357	296 337	8.32	99.95
Yelp	30 431	20 033	316 354	10.42	99.95

4.2 评价指标

实验采用 TopN 推荐评价指标来评估推荐算法的性能,包括召回率 (Recall) 和归一化折损累计增益 (Normalized Discounted Cumulative Gain, NDCG)。

Recall@K 用来推荐算法预测准确性。 $T(u)$ 表示正样本数量, $R(u)$ 表示测试的总样本数。

$$\text{Recall@K} = \frac{\sum_{u \in U} R(u) \cup T(u)}{\sum_{u \in U} T(u)} \quad (18)$$

NDCG@K 不仅可以判断系统有没有推荐出正确项目,还可以通过对序列中的靠前项目进行增强影响,靠后项目进行减弱影响的方式来判断推荐所得序列的合理性, IDCG_u 是理想情况下的最大 DCG 值。

$$\text{NDCG@K} = \frac{\text{DCG}_u @ K}{\text{IDCG}_u} \quad (19)$$

该文分别取值 $K=10$ 和 $K=20$ 与基线方法进行比较。以上两个评价指标值越高,表示预测结果表现越好。

4.3 基线模型

为了验证 CLMLRec 模型的有效性,选择对比的基线模型如下:

(1)BPR^[25]:一种具有代表性的基于协同过滤推荐方法,其具有两两排序损失的传统矩阵分解模型,能针对每一个用户的商品喜好分别进行排序。

(2)GRU4Rec^[26]:第一个在序列推荐中使用门控循环单元来捕获序列信息的 RNN 方法,并利用会话并行小批量训练和基于排名的损失函数进行改进。

(3)Caser^[18]:将 CNN 应用到序列推荐系统中,利用卷积核提取局部信息的能力成功地捕获到了用户的局部动态交互特征。

(4)SASRec^[19]:基于自注意力机制,在每个时间步中,模型从用户的历史交互记录中找出相关项目,并自适应地将权重分配给先前项目,使用它们预测下一个项目。

(5)BERT4Rec^[20]:一种使用深度双向自注意力机

制进行序列推荐的模型,在提取序列信息方面取得了相当不错的效果。

(6)CL4SRec^[3]:首次在序列推荐中使用数据增强和对比学习。

(7)CoSeRec^[4]:进一步提出了两种更具信息性的数据增强方式,来提高对比学习的性能。

(8)DuoRec^[5]:提出了一种抽样策略来制定正样本,并使用 dropout 进行模型增强。

4.4 实验结果与分析

4.4.1 对比实验

该文提出的 CLMLRec 模型与 7 个基线模型应用于 3 个不同数据集上的整体效果如表 2 所示。

表 2 结果对比

数据集	指标	BPR	GRU4Rec	Caser	SASRec	BERT4Rec	CL4SRec	CoSeRec	DuoRec	CLMLRec	Improve/%
Beauty	R@ 10	0.029 6	0.028 4	0.034 2	0.056 2	0.052 6	0.056 9	0.067 5	<u>0.068 6</u>	0.079 7	16.18
	R@ 20	0.047 4	0.042 7	0.064 3	0.081 7	0.082 5	0.088 5	0.101 5	<u>0.102 2</u>	0.112 2	9.78
	N@ 10	0.014 7	0.015 0	0.022 6	0.030 5	0.026 3	0.027 7	<u>0.038 1</u>	0.035 9	0.040 9	7.35
	N@ 20	0.019 2	0.018 6	0.029 8	0.034 9	0.033 8	0.035 6	0.046 7	<u>0.047 0</u>	0.049 1	4.47
Sports	R@ 10	0.021 5	0.025 8	0.026 1	0.033 3	0.035 9	0.041 4	0.042 1	<u>0.043 7</u>	0.046 9	7.32
	R@ 20	0.036 9	0.042 1	0.039 9	0.050 0	0.060 4	0.062 9	0.063 7	<u>0.066 8</u>	0.070 7	5.84
	N@ 10	0.010 5	0.014 2	0.013 5	0.017 2	0.015 6	0.019 1	<u>0.021 5</u>	0.019 5	0.023 5	9.30
	N@ 20	0.014 4	0.018 6	0.017 8	0.022 4	0.021 6	0.026 2	0.028 2	<u>0.029 2</u>	0.029 5	2.57
Yelp	R@ 10	0.047 8	0.048 6	0.049 3	0.057 2	0.050 8	0.058 1	0.061 3	<u>0.062 7</u>	0.068 6	9.40
	R@ 20	0.068 3	0.084 8	0.078 4	0.090 8	0.078 6	0.089 6	0.096 5	<u>0.097 6</u>	0.105 6	8.20
	N@ 10	0.030 2	0.023 9	0.024 5	0.031 1	0.027 9	0.031 2	0.032 2	<u>0.034 5</u>	0.037	7.14
	N@ 20	0.035 4	0.033	0.032 3	0.039 6	0.035 2	0.039 0	0.042 5	<u>0.043 2</u>	0.046 3	7.18

实验结果表明:与其它模型相比,CLMLRec 在三个推荐场景下始终达到最佳性能。CLMLRec 相对于次优对比模型在 Beauty、Sports 和 Yelp 三个数据集上分别取得了 9.45%、6.26% 和 7.98% 左右的提升。此外,在 Beauty 上呈现出比其它两个数据集更明显的改进,这体现了该模型可以基于有限的交互来学习更多的增强视图,对交互量少的数据更有帮助。

与非序列推荐模型(如 BPR)相比,序列推荐模型的表现更好,这表明挖掘用户的序列模式对于下一项预测的重要性。基于 Transformer 的方法比非增强的序列推荐模型(如 GRU4Rec、Caser)获得了更好的性能,这表明自注意力机制在捕获序列模式方面比一般的 CNN 和 RNN 模型更有效。基于自监督的模型相对于经典模型(如 GRU4Rec、Caser 和 SASRec)更有效。其中与使用 MIP 任务训练的模型 BERT4Rec 不同,使用对比学习训练的 CL4SRec、CoSeRec 和 DuoRec 的总体效果更有效,这表明使用对比学习可以通过最大化用户和项目的共同信息来生成更多的表征。与使用数据增强的 CL4SRec、CoSeRec 相比,DuoRec 的整体表现较优,说明引入模型增强可以进一步提高性能。

4.4.2 消融实验

(1)模型的有效性。

为了进一步验证提出的基于时间间隔的数据增强模块和可学习的模型增强模块,在三个数据集上进行消融实验。首先构造 CLMLRec 模型的三个变种模型 CLMLRec-1、CLMLRec-2 和 CLMLRec-3,其中

CLMLRec-1 剔除了数据增强模块,只保留模型增强模块;CLMLRec-2 剔除了模型增强模块,只保留数据增强模块;CLMLRec-3 将基于时间间隔的数据增强模块换为以往的随机数据增强方式;然后采用与 CLMLRec 相同的训练方式进行实验,将三个变种模型与 CLMLRec 用同样的参数指标,在三个数据集上进行推荐效果的比较,结果如图 3 所示。

从图 3 的实验结果可以看出:与其他变体相比,CLMLRec 始终保持最佳推荐性能,这表明所有组件都是产生最佳结果所必需的。此外,可以观察到 CLMLRec-3 与 CLMLRec-2 相比推荐性能大幅下降,证明考虑项目之间的时间间隔可以为数据增强生成更加平衡的增强序列视图。CLMLRec-1 和 CLMLRec-2 比 CLMLRec 推荐性能差,说明了基于时间间隔的数据增强方式和可学习的模型增强模块相结合能够有效提高推荐系统的表现性能,并且可以看出,CLMLRec-2 比 CLMLRec-1 的性能略低,说明了可学习的模型增强模块比基于时间间隔的数据增强方式的性能提升效果更高。

(2)元优化学习的有效性。

为验证本模型 CLMLRec 的基于元优化学习训练的有效性,在同等使用条件下,将 CLMLRec 与使用联合学习训练的 CLMLRec-J 进行实验对比,结果如表 3 所示。可以看出,基于元优化学习训练的模型在所有数据集上都优于使用联合学习训练的模型,证明了基于元优化学习训练方式可以基于相对有限的交互来学习更多的判别增强视图。

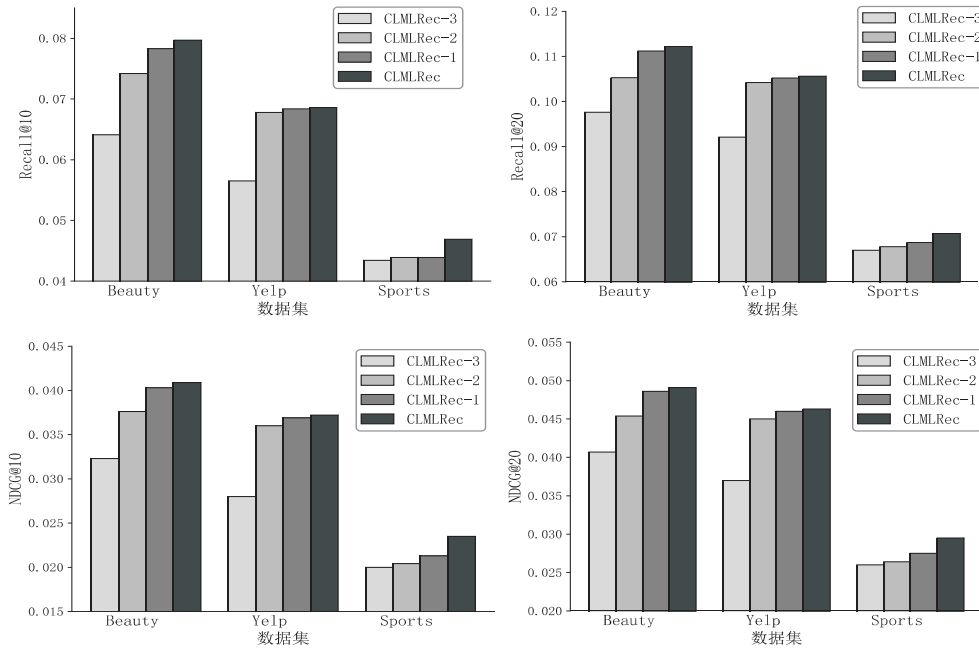


图 3 CLMLRec 模块的有效性

表 3 元优化学习的有效性

模型	Beauty		Sports		Yelp	
	Recall@ 20	NDCG@ 20	Recall@ 20	NDCG@ 20	Recall@ 20	NDCG@ 20
CLMLRec	0.112 2	0.049 1	0.070 7	0.029 5	0.105 6	0.046 3
CLMLRec-J	0.107 5	0.046 8	0.067 8	0.028 0	0.104 8	0.045 6

4.4.3 参数敏感度分析

本节分别对 λ_1 和 λ_2 两个超参数进行调参实验, 来分析两个参数变化对对比学习损失函数的影响, 结果如图 4 所示。

从图 4 中可以看出, 在相同 NDCG@ 10 的评估指标下, CLMLRec 的性能在分配给 λ_1 和 λ_2 不同的权重下都有峰值, 证明了所提框架的有效性, 也说明了选择

合适的权重可以提高推荐性能。在数据集 Beauty 中 λ_1 和 λ_2 分别取 0.03 和 0.03 时推荐效果普遍最好, 在数据集 Sports 中 λ_1 和 λ_2 分别取 0.07 和 0.09 时推荐效果普遍最好, 在数据集 Yelp 中 λ_1 和 λ_2 分别取 0.01 和 0.05 时推荐效果普遍最好, 并且可观察到 λ_2 比 λ_1 的值普遍要大, 说明可学习的模型增强比基于时间间隔的数据增强对于模型更重要、敏感度更高。

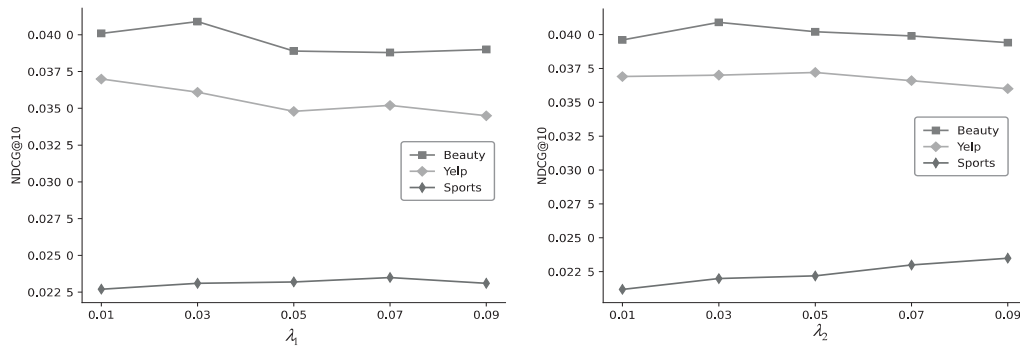


图 4 参数 λ_1 和 λ_2 对模型的影响

5 结束语

受对比学习和元学习的启发, 该文提出一种基于对比学习和元优化学习的序列推荐方法 CLMLRec。该方法根据序列项目之间的时间间隔来为序列生成更加平衡的数据增强视图; 然后, 构建可学习的模型增强模块作为数据增强的补充方法, 捕获在数据增强视图

中潜在的语义信息; 最后, 使用元优化学习训练模型增强模块, 优化更新模块之间的参数, 完成个性化推荐任务。在三个真实数据集上的实验结果表明, 该方法优于对比方法。在未来的工作中, 将考虑项目冷启动问题对于模型推荐性能的影响, 并进一步研究如何在保证不伤害用户长期兴趣的前提下充分满足用户的短期兴趣等问题。

参考文献:

- [1] 邓凯,黄佳进,秦进.基于物品的统一推荐模型[J].计算机应用,2020,40(2):530-534.
- [2] 张文龙,钱付兰,陈洁,等.基于双重最相关注意力网络的协同过滤推荐算法[J].计算机应用,2020,40(12):3445-3450.
- [3] XIE X, SUN F, LIU Z, et al. Contrastive learning for sequential recommendation[C]//2022 IEEE 38th international conference on data engineering (ICDE). Kuala Lumpur: IEEE, 2022:1259-1273.
- [4] LIU Z, CHEN Y, LI J, et al. Contrastive self-supervised sequential recommendation with robust augmentation[J]. arXiv:2108.06479, 2021.
- [5] QIU R, HUANG Z, YIN H, et al. Contrastive learning for representation degeneration problem in sequential recommendation[C]//Proceedings of the fifteenth ACM international conference on web search and data mining. Virtual Event, AZ: ACM, 2022:813-823.
- [6] LIU Z, CHEN Y, LI J, et al. Improving contrastive learning with model augmentation[J]. arXiv:2203.15508, 2022.
- [7] NI Shuang, ZHOU Wei, WEN Junhao, et al. Enhancing sequential recommendation with contrastive generative adversarial network[J]. Information Processing & Management, 2023, 60(3):257290878.
- [8] YANG Yuhao, HUANG Chao, XIA Lianghao, et al. Debiased contrastive learning for sequential recommendation[C]//Proceedings of the ACM web conference 2023. [s.l.]: ACM, 2023:1063-1073.
- [9] WEI T, WU Z, LI R, et al. Fast adaptation for cold-start collaborative filtering with meta-learning[C]//2020 IEEE international conference on data mining (ICDM). Sorrento: IEEE, 2020:661-670.
- [10] 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. Anchorage: ACM, 2019:1073-1082.
- [11] LU Y, FANG Y, SHI C. Meta-learning on heterogeneous information networks for cold-start recommendation[C]//Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. CA: ACM, 2020:1563-1573.
- [12] SONG J, XU J, ZHOU R, et al. CBML: a cluster-based meta-learning model for session-based recommendation[C]//Proceedings of the 30th ACM international conference on information & knowledge management. Queensland: ACM, 2021:1713-1722.
- [13] FINN C, ABBEEL P, LEVINE S. Model-agnostic meta-learning for fast adaptation of deep networks[C]//International conference on machine learning. Sydney: PMLR, 2017:1126-1135.
- [14] HUANG X, SANG J, YU J, et al. Learning to learn a cold-start sequential recommender[J]. ACM Transactions on Information Systems, 2022, 40(2):1-25.
- [15] HAMID R A, ALBAHRI A S, ALWAN J K, et al. How smart is e-tourism? A systematic review of smart tourism recommendation system applying data management[J]. Computer Science Review, 2021, 39:100337.
- [16] RENDLE S, FREUDENTHALER C, SCHMIDT-THIEME L. Factorizing personalized Markov chains for next-basket recommendation[C]//19th international world wide web conference 2010. Raleigh North Carolina: ACM, 2010:811-820.
- [17] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks[J]. arXiv:1511.06939, 2015.
- [18] TANG Jiaxi, WANG Ke. Personalized Top-N sequential recommendation via convolutional sequence embedding[C]//Proceedings of the eleventh ACM international conference on web search and data mining. Marina Del Rey: ACM, 2018:565-573.
- [19] KANG W C, MCAULEY J. Self-attentive sequential recommendation[C]//2018 IEEE international conference on data mining (ICDM). Singapore: IEEE, 2018:197-206.
- [20] 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. Beijing: ACM, 2019:1441-1450.
- [21] GEHRING J, AULI M, GRANGIER D, et al. Convolutional sequence to sequence learning[C]//International conference on machine learning. Sydney: PMLR, 2017:1243-1252.
- [22] YU J, YIN H, XIA X, et al. Self-supervised learning for recommender systems: a survey[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 32(6):335-355.
- [23] HOSPEDALES T, ANTONIOU A, MICAELLI P, et al. Meta-learning in neural networks: a survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 44(9):5149-5169.
- [24] NICHOL A, ACHIAM J, SCHULMAN J. On first-order meta-learning algorithms[J]. arXiv:1803.02999, 2018.
- [25] RENDLE S, FREUDENHLER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//Proceedings of the 2018 world wide web conference. Lyon: ACM, 2009:452-461.
- [26] HIDASI B, KARATZOGLOU A. Recurrent neural networks with top-k gains for session-based recommendations[C]//Proceedings of the 27th ACM international conference on information and knowledge management. Torino: ACM, 2018:843-852.