

基于知识增强对比学习的长尾用户序列推荐算法

任永功, 周平磊, 张志鹏

(辽宁师范大学计算机与人工智能学院, 辽宁 大连 116029)

摘要: 序列推荐根据目标用户的历史交互序列, 预测其可能感兴趣的下一个物品。现有的序列推荐方法虽然可以有效捕获用户的历史交互序列中的长期依赖关系, 但是无法为交互序列较短且用户数量庞大的长尾用户提供精确推荐。为了解决此问题, 提出了一种基于知识增强对比学习的长尾用户序列推荐算法。首先, 基于知识图谱中的丰富实体关系信息, 构建一个基于语义的物品相似度度量, 分别提取原始序列中物品的协同关联物品。然后, 基于不同学习序列提出 2 种序列增强算子, 通过增强自监督信号解决长尾用户序列训练数据不足的问题。最后, 通过对比自监督任务和推荐主任务的网络参数共享的联合训练, 为长尾用户提供更精确的序列推荐结果。在实际数据集上的实验结果表明, 所提算法可以有效提高针对长尾用户的序列推荐精度。

关键词: 序列推荐; 长尾用户; 知识图谱; 对比学习

中图分类号: TP391

文献标志码: A

DOI: 10.11959/j.issn.1000-436x.2024107

Sequential recommendation algorithm for long-tail users based on knowledge-enhanced contrastive learning

REN Yonggong, ZHOU Pinglei, ZHANG Zhipeng

School of Computer Science and Artificial Intelligence, Liaoning Normal University, Dalian 116029, China

Abstract: Sequential recommendation predicts next items for users based on their historical interactions. Existing methods capture long-term dependencies but struggle to recommend precisely for users with short interaction sequences, especially for long-tail users. Therefore, a sequential recommendation algorithm for long-tail users based on knowledge-enhanced contrastive learning was proposed. Firstly, semantic item similarity was introduced by leveraging relationships between entities in the knowledge graph to extract correlated items from original sequences. Secondly, two sequence augmentation operators were proposed based on different contrastive learning views, addressing the problem of insufficient training for long-tail user sequences by augmenting self-supervised signals. Finally, precise sequence recommendations were provided for long-tail users by utilizing the joint training of shared network parameters between contrastive self-supervised tasks and the recommendation task. Experimental results on real-world datasets demonstrate the effectiveness of the proposed algorithm in improving performance for long-tail users.

Keywords: sequential recommendation, long-tail user, knowledge graph, contrastive learning

收稿日期: 2024-01-11; 修回日期: 2024-05-15

通信作者: 张志鹏, zhipengzhang@lnnu.edu.cn

基金项目: 国家自然科学基金资助项目(No.61976109); 辽宁省“兴辽英才计划”基金资助项目(No.XLYC2006005); 辽宁省高等学校科学研究基金资助项目(No.LJKZ0963); 辽宁省科技厅重点研发基金资助项目(No.2022JH2/101300271); 辽宁省教育厅校基本科研基金资助项目(No.LJKQZ20222431); 教育部产学研合作协同育人基金资助项目(No.202102550005); 辽宁省属本科高校基本科研基金资助项目(No.LS2024Q007)

Foundation Items: The National Natural Science Foundation of China (No.61976109), Liaoning Revitalization Talents Program (No.XLYC2006005), The Scientific Research Project of Liaoning Province (No.LJKZ0963), Key Research and Development Projects of Liaoning Provincial Department of Science and Technology (No.2022JH2/101300271), Liaoning Province Ministry of Education (No.LJKQZ20222431), Industry-University Collaborative Education Project of the Ministry of Education (No.202102550005), Basic Research Project of Liaoning Provincial Undergraduate Universities (No.LS2024Q007)

0 引言

推荐系统作为缓解信息过载问题的有效工具,能够预测并呈现用户可能感兴趣的内容或物品,同时降低用户的搜索成本。在电子商务、视频流媒体和新闻媒体等多个领域,推荐系统不仅提供了极大的便利,还显著改善了用户体验。对于企业而言,推荐系统通过提供个性化推荐机制不仅提高了用户满意度和留存率,还有助于推广效果和利润增长。

不同于一般的推荐任务,序列推荐作为推荐系统的一个分支,通过对用户与物品的交互序列按时间顺序进行建模,预测用户可能交互的下一个物品。该方法特别强调用户交互行为的时序性,能够反映用户长期或短期内的兴趣变化。由于用户的行为具有明显的动态变化特性,序列推荐能够捕捉用户的动态偏好,并基于用户在特定时间段内的行为来为用户推荐相关物品。为了获取用户的动态表示,用于处理序列任务的模型如循环神经网络(RNN, recurrent neural network)^[1]模型逐渐被应用于序列推荐任务中。近年来,基于注意力机制衍生出的Transformer模型^[2]为序列推荐提供了更好的建模能力,在揭示序列内部物品的关联关系上有了明显的效果提升。此外,预训练模型在序列推荐任务中也达到了较好的效果,如基于Transformer的双向编码器(BERT, bidirectional encoder representations from transformer)^[3]模型。其他模型通过微调Transformer模型或BERT模型也对用户行为序列进行了建模。尽管这些技术取得了一定进步,但是序列推荐仍面临一些挑战。图1展示了MovieLens数据集的公开子集ML-IM中用户交互数量的排名情况,横坐标表示用户交互数量的排名,依据用户交互数量由高至低排序,纵坐标则显示了各个排名的具体用户交互数量。从图1中可以观察到,大部分用户的交互数量较少,而交互数量较多的用户则相对较少。这表明高频交互的用户是一个较小的群体,这类用户被称为头部用户,而大多数用户的交互频率较低,这类用户被称为长尾用户。图1中用户交互数量本质上遵循长尾分布,序列推荐任务包含的数据集又相对稀疏^[4],其中每个用户的历史交互数量差异显著,长尾用户的交互数量相对较少,少数头部用户却贡献了大量交互,这种分布反映了用户交互行为中存在显著的不均衡问题。现有的序

列推荐模型在短序列上表现欠佳^[5],因此对长尾用户进行推荐目前仍然是一个比较严峻的挑战。长尾用户通常只对一小部分物体进行评价,或者只对特定的物品进行交互^[6]。大多数序列推荐的模型通常基于足够的监督信号进行有监督训练,且有监督模型对头部用户能够生成精确的推荐结果。然而针对长尾用户,由于其交互信息的稀疏性或单一性,在推荐准确性方面存在明显不足。

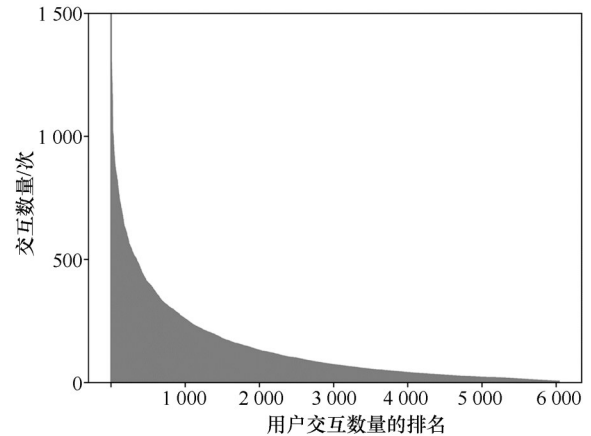


图1 MovieLens数据集的公开子集ML-IM中用户交互数量的排名

在常用的序列推荐数据集中,用户的交互行为一般符合二八定律,即整体用户中约80%为尾部用户,约20%为头部用户,这种分布导致了推荐偏差和用户活跃度不均匀的问题。为了解决这个问题,本文基于Transformer结构引入了知识图谱和对比学习^[7],提出了一种基于知识增强对比学习的长尾用户序列推荐(KECL-SR, knowledge-enhanced contrastive learning-based sequential recommendation)算法,旨在提供更优的用户表示。知识图谱作为一种辅助信息,能够有效缓解由于尾部用户数据稀疏带来的问题^[8]。尽管许多现有模型运用物品属性作为辅助的上下文信息,但是本文选择使用知识图谱的原因在于它能够建立物品间的相关性,从而确保对比学习过程中序列语义的不变性。本文算法受到了计算机视觉和自然语言理解领域在自监督表示学习方面研究的启发,采用对比学习来解决序列推荐中的数据稀疏和长尾用户推荐问题。这种算法面临的主要挑战在于确保在对比学习产生不同序列的变体后,仍能准确表示序列的语义不变。本文的核心思想是通过知识图谱为长尾用户的交互序列中随机选择的物品找到相似物品,并利用

这些相似物品与交互序列生成增强序列,进一步丰富对比学习的维度,以缓解长尾用户信息不足的问题。通过实验发现,本文引入知识图谱与对比学习后,能够显著提升推荐效果。

1 相关工作

序列推荐作为推荐系统的一个分支,重点考虑用户交互序列的时间顺序,以此捕捉用户在一定时间段内的动态偏好。早期的研究通过马尔可夫链为用户的历史兴趣进行建模。例如,Shani等^[9]将推荐问题转化为序列优化问题,并采用马尔可夫决策过程进行推荐,该方法假设用户的下一个行为基于之前的行为,但是在独立假设下,将过去的分量独立组合限制了推荐性能。随后,针对序列的推荐方法如RNN及其变体门控循环单元(GRU, gate recurrent unit)^[10]开始流行,它们对序列的建模方式比马尔可夫链更有效和准确。然而,随着多头注意力机制的提出,上述方法的使用逐渐减少,因为Transformer模型通过自注意力机制能够有效捕捉序列中的长距离依赖关系,并且展示出较好的灵活性和泛化能力。SASRec (self-attentive sequential recommendation)^[11]作为将Transformer模型应用到序列推荐中的先驱算法,对用户交互序列进行建模并成功捕捉到用户的复杂动态兴趣,显著提升了序列推荐效果。基于SASRec和BERT模型改进的BERT4Rec算法利用双向Transformer层和填空任务来增强序列推荐的性能。尽管如此,Transformer模型在解决长尾问题上仍存在不足。在序列推荐领域的研究中,针对长尾问题的专项研究相对匮乏,这仍然是一大挑战。在推荐系统中,用户与物品之间的交互是数据建模的重要基础。然而,在现实推荐场景中,用户与物品之间的互动频次呈现典型的长尾分布,符合帕累托定律。Liu等^[12]对基于分布式session的长尾推荐进行分析和优化,长尾分布意味着大量用户实际上处于冷启动阶段,这在文献^[13]中有详细讨论。受曝光偏差和流行度偏差的影响,长尾用户很难接触到长尾物品,因此其更倾向于点击流行物品。这种情况进一步受数据稀疏性的影响,导致序列推荐中的长尾用户推荐效果相对较差。Transformer在序列推荐中容易受长尾问题的影响,现有基于Transformer的模型在处理短序列时的表现仍然存在不足。从现有研究成果来看,部

分成果通过增加信息源来缓解数据稀疏问题,进而开展长尾推荐研究。Wang等^[14]通过实验分析验证了对比学习在缓解数据稀疏性和增强尾部用户推荐效果方面的有效性。本文在引入外部额外信息源的基础上,采用专门针对缓解数据稀疏问题的对比学习方法来解决长尾问题。

随着对比学习在计算机视觉和自然语言处理领域的广泛应用,其在推荐系统中的应用也日益增多。对比学习的核心优势在于通过对原始无标签数据集进行自监督学习,以获得良好的特征表示,从而有效应对序列推荐中的数据稀疏问题^[15]。对比学习通过生成原始数据的多个序列并使这些序列的互信息最大化来生成正样本对,这种方式能挖掘数据的增强序列,进而执行推荐系统的下游任务,以增强推荐系统的整体性能和鲁棒性。Yao等^[16]提出了一种自监督学习框架,该框架通过结合一组分类特征来学习物品表示。Zhou等^[17]利用物品、属性和子序列之间的关联关系,将带有最大互信息的自监督学习应用于序列推荐任务。文献^[18]将对比学习与序列推荐相结合,提出了对于序列的增强操作。文献^[16]利用物品之间的相关性,增强了对比框架内数据增强的稳健性。文献^[14]结合了数据增强、模型增强和变分增强3种增强策略进行对比学习。与上述工作不同,本文引入了知识图谱,并将知识图谱与生成多序列的对比学习方法相结合,旨在提升对比学习的学习能力,以更好地适应对比学习。

2 基于知识增强对比学习的长尾用户序列

2.1 推荐算法

为了解决序列推荐中长尾用户推荐精度不足的问题,本文提出了KECL-SR算法。该算法结合深度神经网络模型和对比学习框架进行长尾用户的序列推荐,模型示意如图2所示。首先,KECL-SR算法利用用户的交互行为和知识图谱信息提取单个用户的表示。其次,本文提出了一种基于语义的相似度度量方法,该方法通过寻找相似物品进行后续对比学习中的序列增强操作,从而获得更优的序列表示,以提升长尾用户的推荐性能。再次,本文提出了2种适用于对比学习的增强算子,在创建序列的对比学习序列时,这些算子通过随机选择序列中的位置并插入(I)或替换(S)相似物品,生成语义

相似的增强序列。最后, KECL-SR 算法采用主任务和对比学习任务联合训练的方式共享网络参数, 以进行训练, 为长尾用户生成推荐列表。

2.2 任务定义

在序列推荐任务中, 用户集合与项目集合分别记为 U 和 V , 其中单个用户和单个物品分别记为 u 和 v 。每个用户 $u \in U$ 拥有一组交互序列 $S_u = [v_1^u, v_2^u, \dots, v_T^u, \dots, v_{|S_u|}^u]$, 其中 $|S_u|$ 表示用户 u 的交互序列中包含的物品个数, $v_T^u \in V$ 表示用户 u 在位置 T 所交互的物品。在辅助信息方面, 本文利用知识图谱中的实体关系生成初始的物品向量表示。知识图谱提供了丰富的语义信息, 其结构通常可以形式化为一个有向图, 该图由一系列三元组构成, 可以描述为

$$G = \{(h, r, t) | h, t \in \mathcal{E}, r \in \mathcal{R}\} \quad (1)$$

其中, h 、 r 、 t 分别表示一个三元组中的头实体、关系和尾实体, \mathcal{E} 表示实体集合, \mathcal{R} 表示关系集合。序列推荐的任务是在给定序列的前提下, 预测下一个物品出现的概率, 如式(2)所示。

$$\operatorname{argmax}_{v_i \in V} v_i P(v_{|S_u|+1} = v_i | S_u, G) \quad (2)$$

本文通过给定一个用户的交互序列和一个知识图谱集, 计算出物品候选集中概率最大的物品并对其进行推荐。

2.3 基于知识图谱的预训练嵌入层

知识图谱在推荐系统中具有重要作用, 其显著提升了个性化推荐的准确性和用户体验。这是因为知识图谱包含了大量实体信息, 实体之间存在丰富的语义关联^[8]。推荐系统通过将这些实体信息与用户行为数据结合, 能够更全面地了解用户的兴趣和需求, 从而生成更精确的推荐结果。为了实现这一目标, 一种常用的方法是将知识图谱中的实体和关系参数化为向量表示, 本文选择采用 TransR 模型来构建知识图谱实体间的嵌入表示。对于知识图谱中的一个三元组 (h, r, t) , 根据优化平移原则, 通过 $e_h^r + e^r \approx e_t^r$ 来学习其嵌入表示, 其中, $e_h, e_t \in \mathbf{R}^d$ 和 $e^r \in \mathbf{R}^k$ 分别为 h, t 和 r 的嵌入表示, e_h^r 和 e_t^r 分别为 e_h 和 e_t 在关系 r 的向量空间中的投影表示。对于任意一个三元组, 其可信度评分计算式为

$$g(h, r, t) = \|W_r e_h + e_r - W_r e_t\|_2^2 \quad (3)$$

其中, $W_r \in \mathbf{R}^{k \times d}$ 是基于关系 r 的转换矩阵, 可将实体从 d 维向量空间投影到 k 维向量空间。TransR 模型训练旨在使知识图谱中已存在的三元组与其他三元组区分开, 损失函数可以描述为

$$\mathcal{L}_{KG} = \sum_{(h, r, t, t') \in \mathcal{T}} -\ln \sigma(g(h, r, t') - g(h, r, t)) \quad (4)$$

其中, \mathcal{T} 是由正负样本的三元组构成的集合, 负样

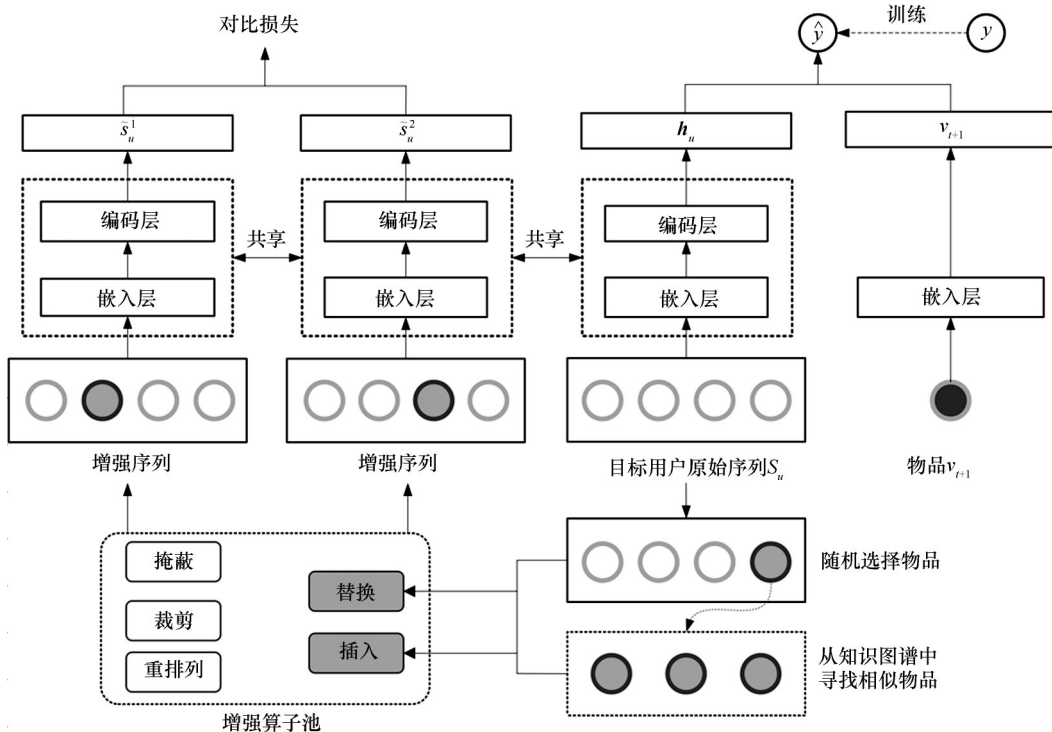


图2 KECL-SR 算法模型示意

本三元组 (h, r, t) 不在正样本集合 G 中, t' 是通过随机替换真实样本中的一个实体后得到的, 激活函数 σ 采用 Sigmoid 函数。

在获取知识图谱中的初始实体嵌入后, 将用户的交互序列 $S_u = [v_1^u, v_2^u, \dots, v_T^u, \dots, v_{|S_u|}^u]$ 转换成一个固定长度的序列 $s_u = (v_1, v_2, \dots, v_T, \dots, v_n)$, 其中 n 代表模型所能接受的最大序列长度。如果序列的长度超过 n , 则截取最近的 n 个用户的交互行为; 如果序列的长度小于 n , 则在交互序列前添加填充物品, 以达到序列长度为 n 的标准。在此过程中, 采用 0 号物品作为填充物品, 用于占位。接下来, 创建一个物品集合的嵌入矩阵 $M \in \mathbf{R}^{|I| \times d}$, 其中 d 代表单个物品嵌入的维度, 再加入文献[2]中使用的可学习位置嵌入 $P \in \mathbf{R}^{n \times d}$, 由此获得用于模型输入的嵌入矩阵 $E \in \mathbf{R}^{n \times d}$, 如式(5)所示。

$$E = \begin{bmatrix} M_{s_1} + P_1 \\ M_{s_2} + P_2 \\ \vdots \\ M_{s_n} + P_n \end{bmatrix} \quad (5)$$

其中, $E_i = M_{s_i} + P_i$ 。值得注意的是, 经过预训练层的嵌入表示只应用于相似物品的查找, 不在序列推荐任务中使用预训练嵌入。

2.4 基于行为序列的用户兴趣嵌入

为了通过用户的交互序列实现序列推荐任务, 本文使用 SASRec 方法^[11]中的 N 个 Transformer 编码器将用户的行为序列转化为用户的兴趣嵌入表示, 编码器结构如图 3(a) 所示。

Transformer 编码器能够将序列转化为嵌入表

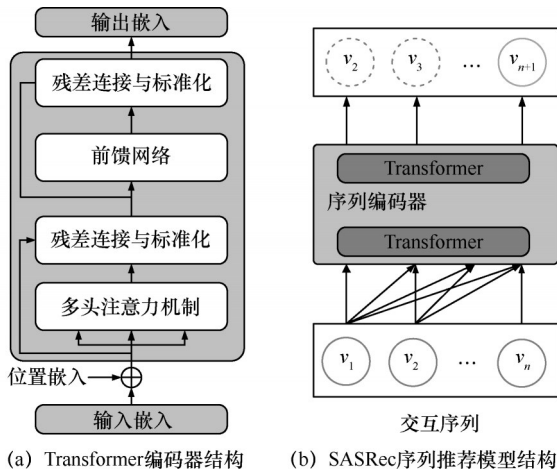


图 3 基于 Transformer 的序列推荐任务示意

示, 这个过程主要包含多头注意力机制和前馈网络。多头注意力机制模块由多个注意力头部组成, 每个注意力头部采用缩放点积注意力机制, 旨在学习序列中物品的重要性。对于输入的嵌入矩阵, 多头注意力机制的计算方法可以表示为

$$MH(H^l) = \text{concat} \{ \text{head}_1, \text{head}_2, \dots, \text{head}_h \} W^O \quad (6)$$

$$\text{head}_i = \text{Attention}(H^l W_i^Q, H^l W_i^K, H^l W_i^V) \quad (7)$$

其中, $W^O \in \mathbf{R}^{d \times d}$, $W_i^Q, W_i^K, W_i^V \in \mathbf{R}^{d \times \frac{d}{h}}$, H^l 表示第 $l+1$ 层的输入, 经过残差连接与标准化模块后, 再次经过一个非线性前馈网络, 前馈网络由 2 个线性层和一个非线性激活函数 ReLU 组成。经过 Transformer 模块处理后, 模型能够自适应地分层提取先前用户消费的物品信息。这个编码过程可以描述为

$$h_u = \text{SeqEnc}(s_u) \quad (8)$$

其中, h_u 表示序列 s_u 经过编码层后的输出嵌入表示。对于序列中的每个位置 T , h_u^T 为预测的下一个物品的表示。使用 Transformer 编码器进行序列推荐的模型结构如图 3(b) 所示, 最终从所有物品中选择与该序列相关性最高的物品进行推荐, 并采用对数似然损失函数来优化序列推荐任务, 如式(9)所示。

$$\mathcal{L}_{\text{main}} = -\log(\sigma(h_u^T e_{v_{T+1}})) - \sum_{v_j \notin s_u} \log(1 - \sigma(h_u^T e_{v_j})) \quad (9)$$

其中, $\mathcal{L}_{\text{main}}$ 表示在序列 s_u 位置 T 处的预测损失分数, σ 表示非线性激活函数 GELU, $e_{v_{T+1}}$ 表示用户 u 交互的物品 v_{T+1} 的嵌入, v_j 表示从随机采样的负例样本中选取的物品。该函数衡量了模型预测结果与实际用户行为之间的一致性。

2.5 基于语义的物品相似度度量

为提供对比学习增强算子所需的高度语义相关的物品集, 本节提出了一种基于语义的物品相似度度量方法, 其结合了 HNSW (hierarchical navigable small world)^[19] 算法和余弦相似度。HNSW 算法构建了一个分层图结构, 顶层图包含少量代表性节点, 实现在高维空间中快速检索出与特定物品语义最相关的物品。基于此索引, 采用余弦相似度公式计算物品间的相似度。具体地, 对于给定的两个物品 i 和物品 j , 根据其从知识图谱中获取的预训练向量 a_i 与 a_j , 计算它们之间的余弦相似度, 如式(10)所示。

$$\text{sim}_{ij} = \frac{\mathbf{a}_i \mathbf{a}_j}{\|\mathbf{a}_i\| \|\mathbf{a}_j\|} \quad (10)$$

余弦相似度反映了2个物品向量在高维空间中的方向接近程度,能够快速且精确地识别出与特定物品语义最接近的物品,并在整个物品集中高效检索出与特定物品最相似的其他物品。

2.6 基于语义增强的对比增强算子

借鉴Liu等^[20]所提的方式,KECL-SR在序列推荐任务中引入对比学习,从不同序列表示每个用户的交互序列,对主任务的学习进行补充。在对比学习的数据增强模块中,常见的操作符如裁剪(C)、遮蔽(M)和重排列(R)被广泛应用^[18],然而这些操作可能会破坏序列之间的相关性^[20]。特别是对于长尾用户来说,由于其交互序列较短,这类随机操作可能对最终的推荐结果产生负面影响。同时,由于长尾用户的交互信息较少,裁剪和遮蔽算子可能会使原始序列变得更短,导致对比监督信号过于稀疏。为了保持序列内部的物品相关性,本文在对比学习的辅助任务训练中提出两种基于语义增强的对比增强算子。这些算子利用知识图谱中的物品关系来填充原始用户序列,从而丰富用户序列的语义信息,以提供对比学习辅助任务中更好的特征表示,增强算子示意如图4所示。

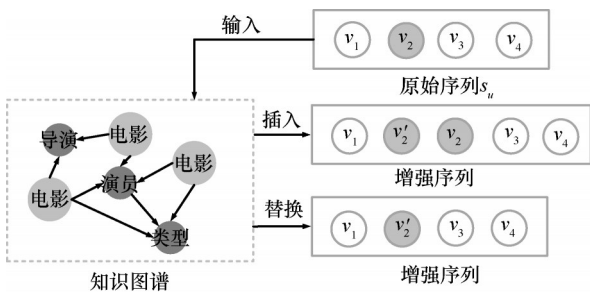


图4 基于语义增强的对比增强算子示意

图4中,输入序列为原始序列,通过2种增强算子注入与原始序列中物品 v_2 语义相关的物品 v_2' 。值得注意的是,仅在对比学习中采用此操作,而在主任务的深度模型中仍使用原始序列,以确保原始监督信号的准确性,具体介绍如下。

1) 插入:对于序列 s_u ,使用相似物品 \bar{v}_{idx_i} 对其进行填充,填充物品来源于知识图谱中提取的相似物品。填充后的序列记为

$$I(s_u) = [v_1, \dots, v_T, \bar{v}_{\text{idx}_i}, \dots, v_{|s_u|}] \quad (11)$$

选择 k 个物品随机插入序列 s_u 的物品之间,插入的物品均与其前或后的邻近物品在预训练过程中使用的知识图谱具有较高的语义相似度,其中, $k = \lfloor \beta n \rfloor$, $\beta \in [0, 1]$, $\text{idx}_i \in [1, 2, \dots, n]$,插入的物品数量受权重 β 控制。

2) 替换:与插入操作相对应,替换增强方法旨在提高推荐系统的性能,同时保持序列之间的相关性。替换后的序列记为

$$S(s_u) = [v_1, v_2, \dots, v_T, \bar{v}_{\text{idx}_i}, \dots, v_{|s_u|}] \quad (12)$$

替换算子同样从知识图谱中选取相关物品,替换原始序列中随机选择的 k 个物品,其中, $k = \lfloor \alpha n \rfloor$, $\alpha \in [0, 1]$, $\text{idx}_i \in [1, 2, \dots, n]$ 决定了被替换物品。

插入和替换算子相较于传统的裁剪、遮蔽和重排列3种算子,能更好地保持序列中的语义连贯性。传统算子如裁剪和遮蔽虽然提高了模型的鲁棒性,但是可能会在随机删除序列中的某些元素的同时移除了关键语义信息,从而影响序列的完整语义和序列内部物品的关联关系。相比之下,插入算子通过向序列中加入语义相关的物品,不仅保持了序列的完整性,还丰富了序列的表达,使得模型能够更好地理解用户的兴趣。替换算子通过将序列中的某些元素替换为具有高度相关性的其他元素,增强了序列的信息密度。这种方法在不改变序列长度的情况下,提升了序列的表达力,这对于长尾用户尤其重要,因为其交互数据比较少,传统算子的随机删除操作容易导致信息丢失。在知识图谱的帮助下,插入和替换操作在基于物品间的语义关系进行,这种基于关系的增强保证了物品间原有的关系不被破坏。通过这2种语义增强算子的应用,模型在训练过程中接触到更多样化的序列变体,有效提高了模型对未知数据的泛化能力。这些优势使得插入和替换算子在对比学习应用场景中处理长尾用户数据时比传统算子更有效。

2.7 对比损失函数

对比损失函数在自监督学习和表示学习领域中应用广泛^[21],其主要目的是学习能够区分不同实例或实体的表示。通过使相似实例之间的相似度最大化和不相似实例之间的相似度最小化,对比损失函数鼓励模型创建有意义且具有区分性的表示。为了让模型准确识别出生成的2种表示是否来自同一个用户的历史序列交互表示,本文通过使正样本对

之间的一致性最大化来优化编码器, 这些正样本对是由同一个序列产生的两种增强表示。在处理小批量数据时, 对于一组序列 $\{s_u\}_{u=1}^N$, 使用 2.6 节的增强算子生成 $2N$ 个增强序列 $\{\tilde{s}_1, \tilde{s}_2, \dots, \tilde{s}_{2u-1}, \tilde{s}_{2u}, \dots, \tilde{s}_{2N-1}, \tilde{s}_{2N}\}$ 。其中, $u \in \{1, 2, \dots, N\}$, 根据 Liu 等^[20]的方法, 将每个用户 u 产生的 $(\tilde{s}_{2u-1}, \tilde{s}_{2u})$ 视为一对正样本, 而其他 $2(N-1)$ 个增强序列作为这对样本的负样本。 $(\tilde{s}_{2u-1}, \tilde{s}_{2u})$ 会经过 Transformer 编码器, 并且与主任务网络共享参数, 表示为 $(\tilde{h}_{2u-1}, \tilde{h}_{2u})$ 。对于一对正样本, 使用 NT-Xent 损失函数可以描述为

$$\mathcal{L}_{\text{self}}(\tilde{h}_{2u-1}, \tilde{h}_{2u}) = -\log \frac{\exp(\text{sim}(\tilde{h}_{2u-1}, \tilde{h}_{2u}))}{\sum_{m=1}^{2N} \delta_{m \neq 2u-1} \exp(\text{sim}(\tilde{h}_{2u-1}, \tilde{h}_m))} \quad (13)$$

其中, sim 表示点积, 用于衡量 2 个放大序列表示的相似度, $\delta_{m \neq 2u-1} \in \{0, 1\}$ 是一个指示函数。最终将 Transformer 编码器输出的嵌入表示连接作为用户序列表示。

2.8 多任务学习

本文采用多任务策略, 充分利用了来自未标记原始数据的自监督信号, 以提升顺序推荐的性能。通过联合优化这些任务, 能够最大限度发挥两个任务的优势, 从而提高推荐系统的整体性能。为了有效地将未标记数据纳入学习过程, 并从序列数据中提取更有意义的表示, 以提高推荐的准确性, 在联合训练部分, Transformer 中多层感知器的参数是共享的, 以预测整个序列推荐的下一个物品。训练过程中的主任务和辅助任务进行共同训练, 如式(14)所示。

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \alpha \mathcal{L}_{\text{self}} \quad (14)$$

3 实验及结果分析

3.1 数据集选取

在本文的实验部分, 选择 MovieLens 数据集中 2 个数据规模不同的公开子集 ML-1M 和 ML-10M 进行研究。由于显式反馈不直接等同于用户的偏好, 本文实验将所有存在的评分转化为正样本, 对于缺失的评分则被视为用户没有兴趣的监督信号。用户对电影的评分行为附带时间戳信息, 表明了评分发生的具体时间, 本文根据时间戳对每个用户的评分行为进行排序, 确保序列中的事件反映了用户的真

实行为顺序, 对于每个用户, 构建了一个动态的交互序列, 为模型提供了学习用户行为随时间变化的能力。对于知识图谱连接数据集的部分, 本文参考文献[22]的方法, 将数据集中的物品与 freebase 中的实体相连接, 为了提高数据的稠密性, 连接过程删除了冗余节点与冷门关系。数据集信息如表 1 所示。

| 数据集 | 用户数/个 | 物品数/件 | 交互数/次 | 元组数/个 |
|--------|--------|--------|------------|-----------|
| ML-1M | 6 040 | 3 698 | 854 573 | 707 654 |
| ML-10M | 69 878 | 10 677 | 10 000 054 | 1 492 617 |

3.2 本文实验结果与相关方法对比

为了验证 KECL-SR 算法的有效性, 本文选取常用的序列推荐算法作为对比算法进行实验, 对比算法主要分为 3 类。

1) 非序列模型

BPR-MF (Bayesian personalized ranking matrix factorization)^[23]是一种具有代表性的非序列化推荐基线算法。该算法结合矩阵分解技术, 采用成对的贝叶斯个性化排序损失进行推荐。为了展示 KECL-SR 在处理序列数据时相较于传统非序列模型的优势, 通过对比表明 KECL-SR 在捕捉用户行为序列的动态特性和时间依赖性上的进步。

2) 基于 Transformer 的模型

SASRec^[11]是一种将 Transformer 应用于序列推荐任务的先驱模型, 其能够对用户的整个交互序列进行建模, 并实现精准个性化推荐。SASRec 作为对比算法, 有助于展示加入了对比学习的 KECL-SR 在处理用户行为序列上的差异。

3) 结合 Transformer 与对比学习的模型

CL4Rec^[18]是一种将对对比自监督学习与 Transformer 编码器结合的算法, 它在标记数据稀少的场景中效果显著, 且被认为是序列推荐中基于 Transformer 较优的算法之一。CL4Rec 在对比学习的结构中只使用了 3 种随机增强算子, 将 CL4Rec 作为对比算法可以证明 KECL-SR 中提出的 2 种算子的有效性。ContrastVAE^[14]同样结合了 Transformer 和对比学习, 加入了变分自编码器与序列化推荐, 并将对比学习引入 VAE 框架中, 是一种可以提升短序列即长尾用户的推荐性能的优秀模型。

3.3 评价指标

本文采用 leave-one-out 策略^[24]来评估 KECL-SR 在序列推荐上的表现。为了评估该方法对长尾用户的推荐效果,在构建测试集时,按用户的交互数量由少到多的顺序进行排序,从次数最少的用户开始,累加至覆盖全体用户交互数量的80%,从而界定长尾用户,这些用户的最后一次交互构成了测试集。对于每位用户,选择倒数第二次交互的物品作为验证集,其他交互物品则作为训练集。推荐列表可表示为 $R_u = \{r_u^1, r_u^2, \dots, r_u^K\}$, 其中 K 代表向用户 u 推荐的物品数量。设测试集中共有 M 个用户,本文选择以下2种度量评估方法。

1) HR: 反映了能够获得精确推荐的用户数量占全部用户数量的百分比^[25], 如式(15)所示。

$$HR@K = \frac{1}{M} \sum_u \mathbb{I}(|R_u^K \cap T_u|) \quad (15)$$

其中, $\mathbb{I}(x)$ 为指示函数, 当 $x > 0$ 时, $\mathbb{I}(x) = 1$, 否则 $\mathbb{I}(x) = 0$; R_u^K 表示推荐列表中前 K 个推荐物品。

2) NDCG: 标准化的累积折扣增益, 其考虑了正确推荐物品的位置^[26], 如式(16)所示。

$$NDCG@K = \frac{1}{Z} \sum_{j=1}^K \frac{2^{\mathbb{I}(|\{R_u^K\} \cap T_u|)}}{\text{lb}(j+1)} \quad (16)$$

其中, Z 为归一化常数。HR 和 NDCG 中的 K 值设定为5、10和20。

3.4 实验参数

本文实验在训练过程中遵循大部分相关研究内容的实验设置^[3,5,11], 删除了交互序列长度小于5个物品的用户。在序列增强部分, 对序列的扩充方法不会对验证集和测试集的选择产生影响。实验采用 PyTorch 框架, 并使用 Adam 优化器进行优化, 学习率设置为0.001, β_1 设置为0.9, β_2 设置为0.99。此外, 物品嵌入的维度大小设置为128, 批量大小设置为256, 注意力头的数量设置为2。以上设置对所有模型通用。

3.5 实验结果及分析

3.5.1 2种增强算子比较

为了验证 KECL-SR 提出的2种数据增强算子的有效性, 本文分别使用插入算子与替换算子进行训练, 实验结果如表2所示。通过实验数据的对比发现, 在2个 MovieLens 数据集上, 对于不同 K 的取值, 替换算子在 HR 和 NDCG 2项指标的长尾用户序列推荐上的结果表现都优于插入算子,

证明了替换操作对长尾用户推荐结果的提升明显优于插入操作。这是因为对序列进行替换增强, 可以在不破坏用户行为模式连贯性的同时丰富推荐系统的理解。特别是对于长尾用户, 这种方法有助于解决数据稀疏性的问题, 而过度的插入操作则可能引入过多噪声, 干扰了模型对用户偏好的学习。

表2 2种数据增强算子对比

| 数据集 | 度量方法 | 替换算子 | 插入算子 |
|--------|---------|---------|---------|
| ML-1M | HR@5 | 0.168 6 | 0.146 4 |
| | HR@10 | 0.259 9 | 0.231 3 |
| | HR@20 | 0.379 1 | 0.345 1 |
| | NDCG@5 | 0.112 2 | 0.095 3 |
| | NDCG@10 | 0.141 6 | 0.122 6 |
| | NDCG@20 | 0.171 6 | 0.151 1 |
| ML-10M | HR@5 | 0.172 4 | 0.142 2 |
| | HR@10 | 0.265 9 | 0.236 7 |
| | HR@20 | 0.383 8 | 0.355 1 |
| | NDCG@5 | 0.113 5 | 0.093 8 |
| | NDCG@10 | 0.143 6 | 0.130 9 |
| | NDCG@20 | 0.173 3 | 0.153 1 |

3.5.2 组合增强算子性能实验结果及分析

为了更深入地分析替换算子和插入算子的效果, 本文将5种增强算子放入增强算子池, 选择替换算子和插入算子中的一种与其他3种传统增强算子(遮蔽、裁剪和重排列)中的一种进行组合使用, 或者将两种新的增强算子组合在一起进行实验。组合方式采用随机调用策略, 在每次辅助任务调用时, 从预定义的一组数据增强方法中随机选择替换算子和插入算子中的一种使用。表3详细列出了 KFCL-SR 与基线算法在 HR 和 NDCG 指标上的对比。

实验结果显示, 非序列模型 BPR-MF 在所有实验中的表现均不如其他模型, 这主要是因为 BPR-MF 无法充分捕捉用户历史交互的时间序列特点。考虑用户的行为和兴趣随时间变化, 这对于推荐系统是至关重要的。BPR-MF 模型在捕捉长期依赖关系方面存在局限性, 可能导致推荐存在不准确性、缺乏时效性和个性化程度不足的问题。SASRec 模型通过引入多头自注意力机制, 有效地捕获了用户历史交互中的长期依赖关系, 比非序列化模型更精准地建模了用户偏好的演变,

从而提升了推荐的准确性和个性化水平。然而对于交互数量较少的长尾用户，SASRec 处理性能略显不足。

CL4Rec 通过在 SASRec 的基础上加入对比学习策略，缓解了长尾用户数据稀疏性问题。该策略通过构建正负样本对进行学习，从而挖掘用户行为中的隐性特征，提升了序列表示的质量。与传统的监督学习方法相比，CL4Rec 加入的对比学习方法不依赖于标注数据，大幅降低了数据收集和处理的成本。此外，通过优化正负样本对的相似度，CL4Rec 成功学习了用户和物品的嵌入表示。这种表示方式具有优秀的泛化能力，能够准确捕捉用户与物品之间的隐含关系，尤其在推荐长尾用户方面表现突出，但其缺少外部的辅助信息。KECL-SR 在 HR 上相较于 CL4Rec 提升了 1.70%~8.09%，在 NDCG 上提升幅度达 2.78%~7.77%。这一结果显现了 KECL-SR 在推荐系统性能提升方面的优势和有效性。相较于 ContrastVAE，KECL-SR 在 HR 上提升了 3.54%~7.57%，在 NGCG 上提升了 12.80%~24.62%。ContrastVAE 与 CL4Rec 在各项评价指标上的表现呈现优劣互补的态势。具体来说，在 MovieLens 的 2 个数据集上，CL4Rec 在大多数度量中展现了卓越的性能，能够提供更精确的推荐结果。这不仅彰显了 CL4Rec 算法的有效性，也突显了其在处理复杂推荐任务时的稳定性。KECL-SR 在处理长尾用户推荐任务时的表现优于上述基线算法，该算法在使用 SASRec 多头注意力编码器的基础上融合了知识

图谱和对比学习策略，改进了 CL4Rec 中的数据增强操作。这种混合使用策略使模型能够更好地适应不同类型数据的变化，提高了其在复杂推荐场景下的处理能力，有效捕获了用户短期和长期的兴趣演变，尤其在处理数据稀疏的长尾用户推荐问题上展现了模型的泛化能力。这是因为 KECL-SR 整合了传统算子和新型算子的优势，长尾用户通常在数据集中的交互频率较低，因此传统算子可能难以有效捕捉和模拟其行为模式。KECL-SR 模型的混合策略在保持原有行为模式连贯性的同时提供了丰富的语义信息，显著提升了对长尾用户的推荐质量。这表明针对长尾用户引入对比学习算子时应考虑引入更丰富的语义关系，以提高系统的整体性能和推荐的个性化程度。

3.6 消融实验

为了深入探究 KECL-SR 中各个组件的有效性与必要性，本文进行了一系列消融实验，旨在对比 KECL-SR 及不同算法的性能表现。

本节将 KECL-SR 与经典的 SASRec 进行对比实验，实验结果如图 5 所示，KECL-SR 相较于 SASRec 展现了良好的效果，这不仅证实了联合训练的优越性，更进一步确认了引入对比学习组件在为模型提供自监督信号方面的显著作用。KECL-SR 采用了与 SASRec 相同的序列编码器，但引入了受序列推荐约束的自监督学习信号，进一步增强了模型对用户兴趣演变的感知能力。对比实验结果表明，在序列推荐任务中，利用自监督信号具有必要性和有效性。

表 3 KECL-SR 与基线算法在 HR 和 NDCG 指标上的对比

| 数据集 | 度量方法 | BPR-MF | SASRec | CL4Rec | ContrastVAE | KECL-SR |
|--------|---------|---------|---------|---------|-------------|---------|
| ML-1M | HR@5 | 0.017 4 | 0.129 7 | 0.164 3 | 0.165 1 | 0.177 6 |
| | HR@10 | 0.042 2 | 0.198 9 | 0.257 7 | 0.255 6 | 0.266 7 |
| | HR@20 | 0.079 3 | 0.297 7 | 0.370 9 | 0.382 2 | 0.392 5 |
| | NDCG@5 | 0.010 9 | 0.083 7 | 0.111 8 | 0.108 3 | 0.115 4 |
| | NDCG@10 | 0.018 9 | 0.106 0 | 0.141 6 | 0.137 6 | 0.144 0 |
| | NDCG@20 | 0.028 2 | 0.130 9 | 0.170 1 | 0.169 6 | 0.175 6 |
| ML-10M | HR@5 | 0.023 9 | 0.129 0 | 0.168 7 | 0.144 5 | 0.173 4 |
| | HR@10 | 0.049 8 | 0.196 9 | 0.249 3 | 0.234 2 | 0.268 7 |
| | HR@20 | 0.086 1 | 0.284 4 | 0.371 6 | 0.345 9 | 0.390 2 |
| | NDCG@5 | 0.014 5 | 0.086 6 | 0.111 5 | 0.091 9 | 0.114 6 |
| | NDCG@10 | 0.022 8 | 0.108 4 | 0.137 5 | 0.120 6 | 0.145 2 |
| | NDCG@20 | 0.031 9 | 0.130 4 | 0.168 4 | 0.148 9 | 0.175 9 |

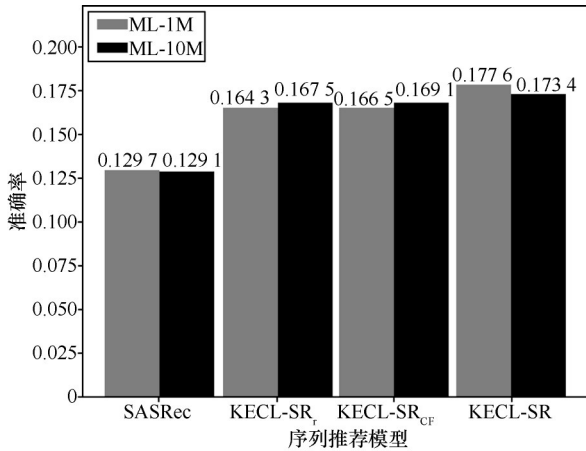


图5 ML-1M与ML-10M数据集上推荐算法的HR@5性能对比

在KECL-SR中使用结合知识图谱的对比学习方法,利用从知识图谱中获取的相似物品随机替换用户交互序列中的部分物品,为了验证利用知识图谱来选取相似物品的策略对推荐的增益,创建了一个变体。在这个变体中,从数据集内随机选取物品代替知识图谱中的相似物品,并采用替换算子进行实验,这种策略被命名为KECL-SR_r。由图5可以发现,KECL-SR的性能优于KECL-SR_r,采用相似物品进行替换的效果更显著,这突显了序列内部相关性的重要性。确保替换物品与被替换物品保持某种相似性可以避免破坏序列的内部联系,因此使用相似物品替换原交互序列中的物品可以获得质量更好的用户表示,这种表示在隐空间内更加接近。

为了验证知识图谱的有效性,本文使用其他方法替换序列中的物品。基于物品的协同过滤(ItemCF)^[27]方法的核心思想是根据用户的行为计算物品之间的相似度,对于一个用户,系统会根据其历史交互物品推荐最相似的其他物品。本文使用ItemCF方法找到与用户交互过的物品的相关物品,对原始序列进行替换,这种策略被命名为KECL-SR_{cf}。从图5中可以看出,KECL-SR的表现优于KECL-SR_{cf},这进一步验证了来自知识图谱的相似物品作为辅助信息,能够为推荐带来更精准的自监督信号,提高推荐的质量并确保引入的物品在隐空间中与原始物品保持高度的相似性,而不仅被视为噪声。

3.7 组合增强算子性能评估与讨论

为了研究不同增强算子对的组合在推荐系统中的效果,本文将不同的操作算子进行两两组合,包

括I、S、C、M和R,并在ML-1M数据集上对比这些组合的性能,以评估它们在改善推荐系统性能方面的有效性,结果如图6所示。

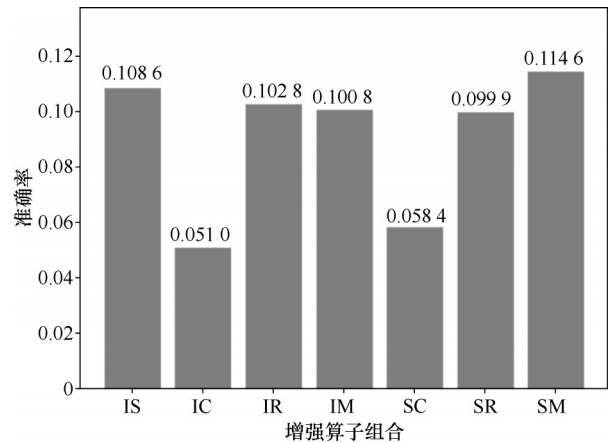


图6 ML-1M数据集上不同增强算子组合的NDCG@5性能对比

从图6可以看出,在所有测试的组合中,SM(替换和遮蔽)组合表现出最优性能,而IS(插入和替换)组合表现次优。相比之下,包含裁剪算子的组合性能较差。此外,在插入算子与裁剪、遮蔽、重排列这3种传统算子的组合中,IR(插入和重排列)表现最佳。总体而言,含有替换算子的组合性能普遍优于含有插入算子的组合,这一结果揭示了在长尾用户的序列推荐中,某些特定的算子组合可能更有效地捕捉长尾用户的潜在偏好。

SM组合通过改变原始用户序列的关键部分,提升了模型对不同用户行为模式的适应性和鲁棒性,特别是对于交互信息相对较少的长尾用户序列。同时,替换组合算子相较于插入组合算子表现更优,这表明替换算子能更有效地捕捉长尾用户的动态行为,因为替换算子在对比学习序列中的序列保持了交互物品之间的相关性和交互序列的长度。此外,实验结果显示,当裁剪算子与插入或替换算子结合时,其表现较差。这可能是由于裁剪操作减少了用户序列中的信息量,尤其在长尾用户的场景下,可能导致重要的交互信息缺失,从而影响推荐的准确性。尽管替换算子组合的整体性能略优于插入算子组合,但两者之间的性能差距并不明显。这表明在不同场景下,不同的增强算子组合可能各有其独特的优势。因此,未来的研究可以进一步探索如何根据具体的应用场景和数据集特性,选择最合适的增强算子组合。

3.8 超参数分析

本节将讨论实验选用的超参数对实验结果的影响。实验选用了性能最好的 SM 与本文提出的 2 种增强算子的组合方式 IS 进行参数分析。图 7 和图 8 分别给出了在 ML-1M 数据集上 SM 组合算子和 IS 组合算子的 NDCG@5 性能对比，其中， α 为 S 算子的替换比例， γ 为 M 算子的遮蔽比例， β 为 I 算子的插入比例，每种参数选用的范围为 [0.1, 0.5]。

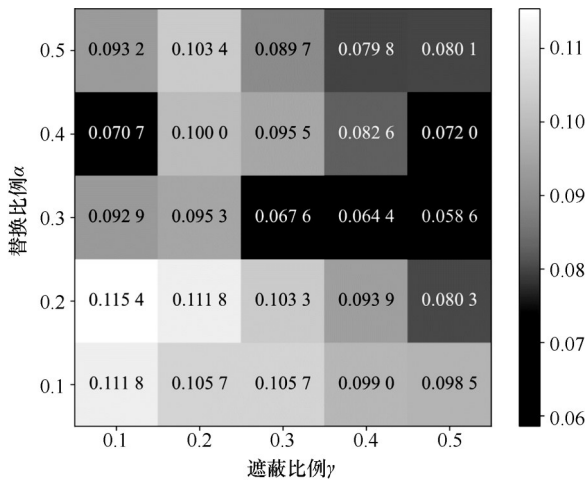


图 7 SM 组合算子的 NDCG@5 性能对比

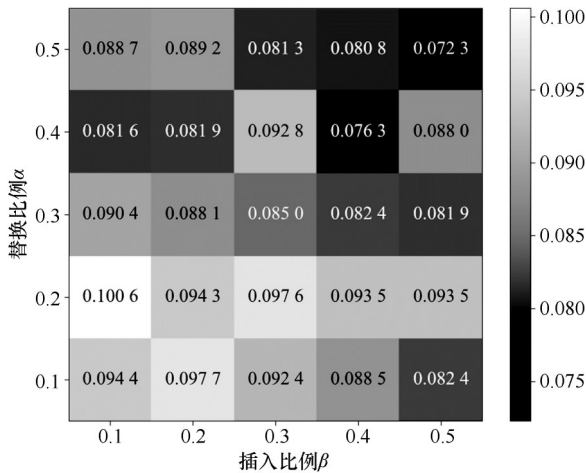


图 8 IS 组合算子的 NDCG@5 性能对比

对于 SM 组合算子来说，替换比例 α 对推荐系统性能的提升具有显著影响。当 $\alpha=0.2$ 时，性能表现相对较高，这表明适度的替换可以增强序列的信息密度，同时不会破坏原有的物品关联结构；当 $\alpha>0.2$ 时，性能有所下降；当 $\alpha=0.1$ 时，性能略低于 $\alpha=0.2$ 时的性能，这可能是由于较低的替换率未

能引入足够多的新信息，而较高的替换率则可能破坏了序列的内在结构。对于遮蔽比例 γ ，当其较低时 NDCG@5 的值较高，这表明轻度遮蔽在提升模型性能方面可能更有效；对于更高的遮蔽比例，性能则有所下降，说明过强的遮蔽可能对序列的语义完整性产生负面影响。

在 IS 组合算子的情况下，对于插入比例 β ，实验数据表明较低的插入比例对性能提升更有利。随着插入比例的提高，NDCG@5 的值普遍呈下降趋势。这意味着在插入过程中，较高的插入比例可能引入了过多的噪声或与原始用户交互的相关物品不够，从而稀释了用户兴趣的表示，影响了模型的预测准确性。

如果替换比例过高，可能会破坏用户的原始行为模式，引入不必要的随机性，从而降低了推荐的准确性，过度的遮蔽也可能导致模型无法准确捕捉用户的细粒度偏好。IS 组合算子在提高推荐系统的多样性和新颖性方面更突出，插入算子通过增加与用户历史行为相似的物品，能够丰富推荐列表，这对长尾用户来说可以提供更多的内容发现机会。插入操作可能会引入与用户实际偏好不完全相关的物品，从而影响推荐的连贯性和用户体验。在稳定性方面，随着参数的变化，IS 组合算子的性能下降幅度比 SM 组合算子小。

总体而言，SM 组合算子在保持推荐稳定性方面通常表现更优，其通过精细调控增强操作，能够保持用户行为的连贯性。IS 组合算子在提高推荐多样性和新颖性方面更具优势，但可能会牺牲一定的推荐稳定性。在追求高度个性化推荐的场景中，SM 组合算子更适用，而在探索新用户兴趣或增加推荐多样性的场景中，IS 组合算子可能更合适。

4 结束语

本文深入研究了面向长尾用户的推荐问题，为了突破现有序列推荐算法在这一问题上的局限性，利用知识图谱中丰富的实体信息提出了 KECL-SR 算法，从而有效增强对长尾用户的表示与推荐能力。通过结合知识图谱的对比学习策略，为序列推荐引入一种自监督学习方式，从而缓解数据稀疏和长尾用户推荐的问题，进一步提升了推荐质量。基于当前的研究结果，未来的工作集中在以下几个方面。

1) 进一步优化增强算子的选择和组合, 以及探索如何将这些技术应用于更广泛的数据集和推荐场景中。

2) 考虑引入更多样化的评估指标, 以全面评估不同增强算子组合在提升推荐系统性能方面的效果。

3) 计划进一步引入更丰富多样的知识图谱数据, 以持续优化模型性能。

4) 研究如何使模型更具普适性, 以适应更多类型的推荐系统场景。例如, 深入挖掘长尾用户的内部多样性, 并为其提供更精细的推荐服务。

参考文献:

- [1] HIDASI B, QUADRANA M, KARATZOGLOU A, et al. Parallel recurrent neural network architectures for feature-rich session-based recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems. New York: ACM Press, 2016: 241-248.
- [2] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. New York: ACM Press, 2017: 6000-6010.
- [3] SUN F, LIU J, WU J, et al. BERT4Rec: sequential recommendation with bidirectional encoder representations from transformer[C]//Proceedings of the Proceedings of the 28th ACM International Conference on Information and Knowledge Management. New York: ACM Press, 2019: 1441-1450.
- [4] MCAULEY J, TARGETT C, SHI Q F, et al. Image-based recommendations on styles and substitutes[C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2015: 43-52.
- [5] LIU Z W, FAN Z W, WANG Y, et al. Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2021: 1608-1612.
- [6] YIN J W, LIU C H, WANG W Q, et al. Learning transferrable parameters for long-tailed sequential user behavior modeling[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2020: 359-367.
- [7] JAISWAL A, BABU A R, ZADEH M Z, et al. A survey on contrastive self-supervised learning[J]. Technologies, 2020, 9(1): 2-22.
- [8] 张天成, 田雪, 孙相会, 等. 知识图谱嵌入技术研究综述[J]. 软件学报, 2023, 34(1): 277-311.
- [9] 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.
- [10] 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. New York: ACM Press, 2018: 843-852.
- [11] KANG W C, MCAULEY J. Self-attentive sequential recommendation[C]//Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM). Piscataway: IEEE Press, 2018: 197-206.
- [12] LIU S Y, ZHENG Y J. Long-tail session-based recommendation[C]//Proceedings of the 14th ACM Conference on Recommender Systems. New York: ACM Press, 2020: 509-514.
- [13] HU L, CAO L B, CAO J, et al. Improving the quality of recommendations for users and items in the tail of distribution[J]. ACM Transactions on Information Systems, 2017, 35(3): 1-37.
- [14] WANG Y, ZHANG H R, LIU Z W, et al. ContrastVAE: contrastive variational autoencoder for sequential recommendation[J]. arXiv Preprint, arXiv: 2209.00456, 2022.
- [15] WU J C, WANG X, FENG F L, et al. Self-supervised graph learning for recommendation[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2021: 726-735.
- [16] YAO T S, YI X Y, CHENG D Z, et al. Self-supervised learning for large-scale item recommendations[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. New York: ACM Press, 2021: 4321-4330.
- [17] ZHOU K, WANG H, ZHAO W X, et al. S³-rec: self-supervised learning for sequential recommendation with mutual information maximization[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York: ACM Press, 2020: 1893-1902.
- [18] XIE X, SUN F, LIU Z Y, et al. Contrastive learning for sequential recommendation[C]//Proceedings of the 2022 IEEE 38th International Conference on Data Engineering (ICDE). Piscataway: IEEE Press, 2022: 1259-1273.
- [19] MALKOV Y A, YASHUNIN D A. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(4): 824-836.
- [20] LIU Z W, CHEN Y J, LI J, et al. Contrastive self-supervised sequential recommendation with robust augmentation[J]. arXiv Preprint, arXiv: 2108.06479, 2021.
- [21] ZHANG T C, TIAN X, SUN X H, et al. Overview on knowledge graph embedding technology research[J]. Journal of Software, 2023, 34(1): 277-311.

- [21] BASTANLAR Y, ORHAN S. Self-supervised contrastive representation learning in computer vision[M]. London: IntechOpen, 2022.
- [22] ZHAO W X, HE G L, YANG K L, et al. KB4Rec: a data set for linking knowledge bases with recommender systems[J]. Data Intelligence, 2019, 1(2): 121-136.
- [23] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. New York: ACM Press, 2009: 452-461.
- [24] HEXN, LIAOLZ, ZHANGHW, et al. Neural collaborative filtering[C]// Proceedings of the 26th International Conference on World Wide Web. New York: ACM Press, 2017: 173-182.
- [25] KARYPIS G. Evaluation of item-based top-*N* recommendation algorithms[C]//Proceedings of the Tenth International Conference on Information and Knowledge Management. New York: ACM Press, 2001: 247-254.
- [26] JÄRVELIN K, KEKÄLÄINEN J. IR evaluation methods for retrieving highly relevant documents[J]. ACM SIGIR Forum, 2017, 51(2): 243-250.
- [27] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: item-to-item collaborative filtering[J]. IEEE Internet Computing, 2003, 7(1): 76-80.

[作者简介]



任永功 (1972-), 男, 辽宁兴城人, 博士, 辽宁师范大学教授, 主要研究方向为人工智能、数据挖掘、推荐系统。



周平磊 (1998-), 男, 辽宁大连人, 辽宁师范大学硕士生, 主要研究方向为数据挖掘、推荐系统。



张志鹏 (1988-), 男, 河南安阳人, 博士, 辽宁师范大学讲师, 主要研究方向为人工智能、数据挖掘、推荐系统。