



研究与开发

基于对比增强时间感知自注意力机制的序列推荐

于洋, 王瑞琴

(湖州师范学院信息工程学院, 浙江 湖州 313000)

摘要: 现有序列推荐模型在绝对交互时间的利用上存在不足, 导致用户偏好建模不准确。因此, 提出了基于对比增强时间感知自注意力机制的序列推荐模型 (sequential recommendation based on contrast enhanced time-aware self-attention mechanism, CTiSASRec)。首先, 注意力权重的计算过程整合了评分数据、绝对交互时间、位置信息和项目流行度; 其次, 将项目的绝对交互时间和位置顺序融合, 生成新的项目位置嵌入; 最后, 训练过程中利用对序列两次建模结果的对比学习来区分样本间的相似性和差异性, 进而提高模型的准确性和鲁棒性。在6个不同领域和规模的数据集上进行的实验表明, CTiSASRec的表现优于目前最先进的顺序推荐模型。

关键词: 推荐系统; 自注意力; 时间感知模型; 对比学习

中图分类号: TP391

文献标志码: A

doi: 10.11959/j.issn.1000-0801.2025003

Sequential recommendation based on contrast enhanced time-aware self-attention mechanism

YU Yang, WANG Ruiqin

School of Information Engineering, Huzhou University, Huzhou 313000, China

Abstract: The existing sequence recommendation models have shortcomings in utilizing absolute interaction time, resulting in inaccurate modeling of user preferences. Sequential recommendation based on contrast enhanced time-aware self-attention mechanism (CTiSASRec) was proposed. Firstly, the calculation process of attention weights integrated rating data, absolute interaction time, location information, and project popularity. Secondly, the absolute interaction time and location order of the project were integrated to generate a new project location embedding. Finally, during the training process, contrast learning based on the results of two modeling sequences was used to distinguish the similarities and differences between samples, thereby improving the accuracy and robustness of the model. Experimental studies conducted on six datasets of different fields and scales show that CTiSASRec outperforms state-of-the-art sequential recommendation models.

Key words: sequential recommendation, self-attention, time-aware model, contrast learning

收稿日期: 2024-10-13; 修回日期: 2024-11-29

通信作者: 王瑞琴, wrq@zjhu.edu.cn

基金项目: 国家自然科学基金面上项目 (No. 62277016)

Foundation Item: The National Natural Science Foundation of China (No.62277016)



0 引言

推荐系统的核心在于预测用户与特定项目互动的可能性，这些互动包括点击、评分、购买等多种行为。鉴于用户的互动历史是丰富的偏好数据宝库，对这些互动进行建模成为推荐系统领域的关键。在序列推荐的背景下，数据建模主要围绕两个维度展开：时间和顺序。基于时间的序列推荐强调对互动时间戳的建模，以捕捉用户偏好和项目属性中固有的时间细微差别。基于顺序的序列推荐根据互动时间戳对项目进行顺序排列，如 SASRec^[1]。这些方法均未利用与项目相关的特性信息，因此在个性化推荐方面存在一定的局限性。

现有的顺序推荐方法存在一定局限性，因为该类方法主要依赖项目在序列中的位置及项目标识符来进行后续项目的预测，如果两个用户拥有相同的访问序列，他们就会被推荐相同的下一个项目。为了弥补这一缺陷，研究者将时间戳引入顺序推荐框架中，如 TiSASRec^[2]。在实际应用中，如果两个用户表现出相同的交互序列，但是其中一个用户在短时间内产生了这些交互，而另一个用户在长时间内完成了这些交互，这两个用户会得到不同的预测结果。然而，TiSASRec 仅采用相对时间间隔，忽略了绝对时间的影响，这导致模型丧失了对用户兴趣顺序的准确捕捉能力。

本文在 TiSASRec 的基础上进行改进，提出了一种创新的位置编码方案，将绝对时间戳与位置信息无缝结合，促进了项目更全面地嵌入，捕捉到时间和顺序上的细微差别。从实验可以看出，本文所提的基于对比增强时间感知自注意力机制的序列推荐模型（sequential recommendation based on contrast enhanced time-aware self-attention mechanism, CTiSASRec）在密集和稀疏数据集上都优于目前最先进的顺序推荐算法。本文的主

要贡献总结如下。

(1) 引入新的位置嵌入编码方法，将绝对交互时间与项目序列位置融合，提供了更全面的建模视角。

(2) 提出基于评分信息、位置嵌入和项目流行度的自注意力机制，用于用户偏好建模。

(3) 在模型训练阶段，对同一用户多次序列建模的结果进行对比学习，从而优化模型。

1 相关工作

在序列推荐中，捕捉用户的长期和短期偏好至关重要，因为这些偏好通常驱动着不同但互补的决策过程。GRU4Rec^[3]利用门控循环单元（gated recurrent unit, GRU）捕捉用户会话中的复杂序列模式，从而实现高度个性化的推荐。CLF4SRec^[4]引入了一个多任务学习框架，结合对比学习和序列建模。LRURec^[5]专注于长序列推荐，通过递归将长输入序列划分为子序列来促进并行处理。以上模型仅考虑了序列的顺序关系，忽视了时间维度的信息。

近年来，注意力机制在推荐系统中得到了广泛的应用。DFAR^[6]在注意力机制的基础上引入对比学习进行建模。MOJITO^[7]是将物品的嵌入与物品文本嵌入带入注意力机制，再将其结果与原嵌入结合作为短期兴趣，将用户嵌入、物品序列嵌入与预测物品嵌入结合当作长期兴趣。Tri-CDR^[8]创新性地引入了一种三重跨域注意力机制，该机制能够精准地聚焦于与用户总体偏好及其特定目标域倾向紧密相关的信息，从而促进了跨域信息的无缝整合与增强。LSAB^[9]在自注意力模块中融入复杂的偏置策略，能够敏锐地识别和融合用户行为模式中的快速且细微的变化。以上模型虽然引入了注意力机制，却仍然缺乏对时间信息的关注。

时间感知的序列推荐在建模过程中关注用户交互的具体时间。TAMMF^[10]融合矩阵分解算法

和时间注意力网络，形成多任务学习模型。Time-LSTM^[11]基于长短期记忆（long short-term memory, LSTM）对用户的顺序动作进行建模。SLRC^[12]结合 Hawkes 过程和协同过滤，对重复消费的时间动态进行建模。HTP^[13]将物品时间戳细分为物品之间的相对时间间隔与物品的绝对时间，分别通过单独计算与融合计算进行建模。GDERec^[14]采用双组件架构，每个组件都精心设计了图神经网络进行参数优化。STASAN^[15]引入时空感知自注意力层，该层能够精确捕获和分析用户的个性化时空偏好。GCE-GNN^[16]利用图神经网络建模会话序列，建模过程引入全局上下文信息。与以上方法不同的是，本文模型创新性地引入了绝对时间信息，将绝对时间戳与位置信息进行融合，可以捕捉用户偏好建模过程中时间和顺序上的细微差别。

TiSASRec 模型是目前时间感知序列推荐模型中最先进的技术，它引入了相对时间戳的概念，认为序列中交互时间戳越接近的物品对下一个物品预测的影响越大。TiSASRec 在用户建模过程中考虑了 3 个关键因素：物品之间的时间间隔、物品的位置信息和物品嵌入信息。本文认为，TiSASRec 可以从以下几个方面进行改进。

(1) 表示项目位置的方法比较单一，可以考虑项目交互的绝对时间戳来丰富这种表示。

(2) 用户的个性化建模能力不强，融入个性化属性增强建模可以使推荐更贴合用户偏好。

(3) 用户表示缺乏差异化学习，可以在建模过程中融入对比学习来区分不同的用户表示。

2 模型设计

基于对比增强时间感知自注意力机制的序列推荐模型如图 1 所示，包括预处理层、嵌入层、自注意力层、预测层和对比层。预处理层从交互序列中精心提取关键特征，如绝对时间戳、项目流行度指标和用户评分数据。此外，它还将用户序列标准化为固定长度，以便于高效处理，最后将数据传入嵌入层。嵌入层将从预处理层得到的数据生成用户嵌入、项目嵌入、时间嵌入和位置嵌入的初始输入向量。这些嵌入是自注意力层后续计算的输入数据。自注意力层会利用多种因素（包括项目嵌入、位置信息、相对时间间隔、项目流行度和用户评分）来计算项目序列的注意力权重。这一复杂机制能够有效地捕捉和建模用户随时间变化的偏好。该层所得到的结果成为预测层和对比学习层的输入数据。预测层使用向量点

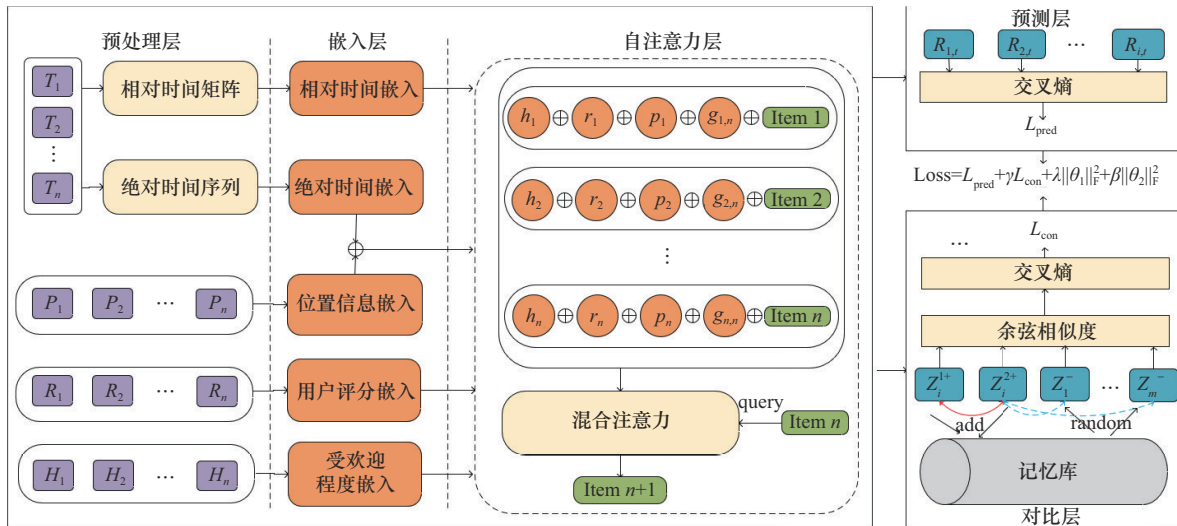


图 1 基于对比增强时间感知自注意力机制的序列推荐模型



积来对序列中的下一个项目进行评分和预测，它采用二元交叉熵和BPR损失（Bayesian personalized ranking loss）作为目标函数，以优化模型在训练过程中的性能。对比层通过最小化应用于同一用户序列的两个并行建模任务结果之间的差异来提升模型的能力。

2.1 预处理层

设 U 和 I 分别表示用户集和项目集，在时间感知序列推荐的背景下，每个用户的交互序列表示为 $S_u=(S_1^u, S_2^u, \dots, S_{|S_u|}^u)$ ，其中， $u \in U$ 且 $S_i^u \in I$ 。相应的交互时间序列表示为 $T_u=(T_1^u, T_2^u, \dots, T_{|T_u|}^u)$ ，用户评分序列表示为 $R_u=(R_1^u, R_2^u, \dots, R_{|R_u|}^u)$ ，项目流行度序列表示为 $H_u=(H_1^u, H_2^u, \dots, H_{|H_u|}^u)$ 。项目的流行度是通过计算其在数据集中出现的频率来获得的。

由于每个用户的交互序列长度不同，本文首先将所有用户的交互序列转换为固定长度的序列，记为 $S_u=(S_1^u, S_2^u, \dots, S_n^u)$ ，其中， n 表示最大固定长度。如果序列长度超过 n ，则仅考虑最近的 n 次交互。如果序列长度低于 n ，则在序列的左端添加“0”，直到序列长度达到 n 为止。同样，对于交互时间序列 T_u 、用户评分序列 R_u 和项目流行度序列 H_u 也经过相应的处理以获得固定长度的序列。其中，对于交互时间序列，本文采用用户交互序列中第一个项目的序列时间戳作为填充项。这种方法可以防止计算过程出现除以0的情况。

2.2 嵌入层

本文为项目创建一个矩阵 $M^I \in \mathbf{R}^{|I| \times d}$ ，其中， d 为隐藏维度。对于给定的用户交互序列，通过表查找操作获得前 n 个项目的嵌入，并将它们堆叠在一起以获得项目嵌入矩阵 $E^I \in \mathbf{R}^{n \times d}$ 。使用同样的方法，可以获得位置嵌入矩阵 $E^P \in \mathbf{R}^{n \times d}$ 、评分嵌入矩阵 $E^R \in \mathbf{R}^{n \times d}$ 、项目流行度嵌入矩阵 $E^H \in \mathbf{R}^{n \times d}$ 、相对时间嵌入矩阵

$E^G \in \mathbf{R}^{n \times d}$ 和绝对时间嵌入矩阵 $E^{Abs} \in \mathbf{R}^{n \times d}$ 。

接下来，将项目在序列中的位置与项目的绝对时间相结合，以创建新的位置信息。

$$E^P = W(E^{Abs} + E^P) \quad (1)$$

其中， W 是可学习参数。

为了便于后续的自注意力计算，针对每种嵌入向量分别生成键向量和值向量，且生成的键向量和值向量与原始向量相同。例如，位置嵌入向量 E^P 的键向量和值向量分别表示为 E_K^P 和 E_V^P 。 E_K^P 和 E_V^P 的值与位置嵌入向量 E^P 的值完全相同。

2.3 自注意力层

受基于相对位置的自注意力机制的启发，本文提出了混合感知自注意力机制。在计算注意力权重时，不仅考虑两个项目交互之间的时间间隔，还整合多种信息，包括项目的序列位置、交互的绝对时间、用户对项目的评分和项目的流行度，混合注意力框架如图2所示。

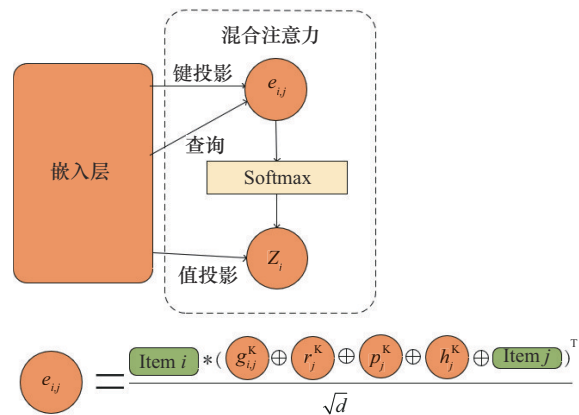


图2 混合注意力框架

对于输入矩阵 $E^I=(e_1, e_2, \dots, e_n)$ ，其中 $e_i \in \mathbf{R}^d$ ，通过自注意力计算生成一个输出序列 $Z=(z_1, z_2, \dots, z_n)$ ，计算式为：

$$z_i = \sum_{j=1}^n a_{ij}(e_j W^V + g_{i,j}^V + r_j^V + p_j^V + h_j^V) \quad (2)$$

其中， $W^V \in \mathbf{R}^{d \times d}$ 是值的输入投影权重， $g_{i,j}^V$ 是项目 i 与项目 j 的时间间隔， r_j^V 是项目 j 的评分， p_j^V

是项目 j 的位置信息, h_j^V 是项目 j 的受欢迎程度, 每一个系数 a_{ij} 都使用 Softmax 函数计算。

$$a_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^n \exp e_{ik}} \quad (3)$$

其中, e_{ij} 用一个基于输入数据、相对时间、项目位置、用户评分和项目流行度的函数进行计算, 具体为:

$$e_{ij} = \frac{m_{s_i} \mathbf{W}^Q (m_{s_j} \mathbf{W}^K + g_{i,j}^K + r_j^K + p_j^K + h_j^K)^T}{\sqrt{d}} \quad (4)$$

其中, $\mathbf{W}^Q \in \mathbf{R}^{d \times d}$ 和 $\mathbf{W}^K \in \mathbf{R}^{d \times d}$ 分别是查询和键的输入投影权重, 标度因子 \sqrt{d} 用于避免内积值过大, 特别是当向量维度较大时。

虽然上述混合感官注意力机制有效地融合了项目位置、评分、流行度、相对时间和绝对时间等信息来编码项目嵌入, 但本质上是通过线性变换来运作的, 这限制了其建模复杂关系的能力。为了解决这一问题, 本文在自注意力计算之后引入了一个前馈层, 由两个线性变换和一个非线性激活函数 ReLU 构成, 增强了模型的学习能力, 具体为:

$$\text{FFN}(z_i) = \max(0, z_i \mathbf{W}_1 + \mathbf{b}_1) \mathbf{W}_2 + \mathbf{b}_2 \quad (5)$$

其中, \mathbf{W}_1 和 $\mathbf{W}_2 \in \mathbf{R}^{d \times d}$ 是线性变换矩阵, \mathbf{b}_1 和 $\mathbf{b}_2 \in \mathbf{R}^d$ 是偏置向量。虽然不同项目之间的线性变换是相同的, 但它们在每一层使用不同的参数。

自注意力层和前馈层的堆叠会导致过拟合、训练过程不稳定、训练时间过长等问题, 本文使用 Dropout 正则化、残差连接和层归一化来解决这些问题, 如式 (6) 所示。

$$Z_i = z_i + \text{Dropout}(\text{FFN}(\text{LayerNorm}(z_i))) \quad (6)$$

2.4 预测层

通过对用户偏好向量和物品嵌入向量进行点积运算, 可以预测用户对候选物品的评分, 如式 (7) 所示。

$$R_i = \mathbf{Z} e_i \quad (7)$$

其中, $e_i \in \mathbf{R}^d$ 是项目 i 的嵌入。

由于用户交互是隐式数据, 所以不能直接优化偏好分数 $R_{i,t}$, 本文模型的目标是提供一个排序

的项目列表, 因此采用负抽样来优化项目的排名。对于每个期望的正输出 o_i , 抽取一个负项 $o'_i \notin S_u$, 生成一组成对的偏好顺序 $D = \{(S_u, T_u, o, o')\}$, 采用二元交叉熵作为损失函数, 如式 (8) 所示。

$$L_{\text{pred}} = - \sum_{S_u \in S} \sum_{t \in [1, 2, \dots, n]} [\text{lb}(\sigma(r_{o_t}, t)) + \text{lb}(1 - \sigma(r'_{o'_t}, t))] \quad (8)$$

其中, $\sigma(x)$ 表示 Sigmoid 函数。

2.5 对比层

为了从用户交互序列中获取个性化用户偏好向量, 并确保不同用户之间的偏好向量相互分离, 本文引入对比学习策略。该策略旨在使同一用户在不同编码下的建模结果尽可能相似, 同时使不同用户的建模结果尽可能不同。具体地, 对于同一个用户序列, 两次通过自注意力层得到的建模结果形成一个“正样本对”; 从其他用户序列通过自注意力层得到的建模结果中随机抽取一个, 与当前用户序列的建模结果形成一个“负样本对”。该策略旨在使同一用户在不同编码下的建模结果尽可能相似, 同时使不同用户的建模结果尽可能不同。定义对比损失函数为:

$$L_{\text{con}} = - \sum_{u=1}^{|D|} \ln \frac{e^{\text{sim}(Z_u^1, Z_u^2)/\tau}}{\sum_{v=1}^N (e^{\text{sim}(Z_u^1, Z_u^2)/\tau} + e^{\text{sim}(Z_u^1, Z_v)/\tau})} \quad (9)$$

其中, e 为自然常数, τ 为超参数, Z_u^1 和 Z_u^2 分别表示对当前用户序列进行两次建模所获得的结果, Z_v 表示另一个用户序列的建模结果。 D 是训练数据集, N 是负对数量, $\text{sim}(\cdot)$ 表示两个向量之间的相似度, 使用余弦相似度计算如式 (10) 所示。

$$\text{sim}(\mathbf{s}_1, \mathbf{s}_2) = \frac{\mathbf{s}_1 \mathbf{s}_2}{\|\mathbf{s}_1\| \|\mathbf{s}_2\|} \quad (10)$$

考虑单个训练批次内数据可能不完整, 仅依赖批次内比较可能影响模型的有效性。为了解决这一问题, 本文维护了一个全局数据记忆库, 用于存储每个用户的序列建模结果。当需要负样本



时, 可以从这个库中迅速高效地检索。

2.6 模型优化

将预测损失 L_{pred} 和对比损失 L_{con} 融合得到整体损失函数, 采用多任务联合优化的方式进行模型训练, 如式 (11) 所示。

$$\text{Loss} = L_{\text{pred}} + \gamma L_{\text{con}} + \lambda \|\theta_1\|_F^2 + \beta \|\theta_2\|_F^2 \quad (11)$$

其中, γ 是平衡对比模块的超参数, λ 和 β 是正则化超参数, $\theta_1 = \{E^I, E_K^P, E_V^P, E_K^G, E_V^G\}$ 和 $\theta_2 = \{E_K^R, E_V^R, E_K^H, E_V^H\}$ 是嵌入矩阵集合, $\|\cdot\|_F$ 是 Frobenius 范数。

2.7 复杂度分析

自注意力层的时间复杂度为 $O(n^2d)$, 而其余层的时间复杂度都没超过自注意力层, 所以模型的整体时间复杂度为 $O(n^2d)$ 。对于空间复杂度, 预处理层复杂度为 $O(n^2)$, 嵌入层复杂度为 $O(nd^2)$, 自注意力层复杂度为 $O(nd+d^2)$, 对比学习层复杂度为 $O(n^2d)$, 预测层复杂度为 $O(nd)$, 其中, $d \ll n$, 所以整体的空间复杂度为 $O(n^2d)$ 。

3 实验结果

3.1 数据集

本文选用来自亚马逊平台的类型、规模和稀疏程度各不相同的多个数据集进行实验。数据集统计信息见表 1。

表 1 数据集统计信息

| 数据集 | 用户数量 | 项目数量 | 交互次数/MB |
|---------------|---------|---------|---------|
| Gift | 458 | 157 | 0.002 |
| Digital_Music | 16 510 | 11 358 | 0.132 |
| Video_Games | 55 220 | 17 294 | 0.377 |
| CDs_and_Vinyl | 147 592 | 140 260 | 1.420 |
| Automotive | 193 621 | 79 039 | 1.291 |
| Movies_and_TV | 297 498 | 59 944 | 2.748 |

Gift 数据集特征明显、稀疏度高, 包含 458 名用户对 157 个商品的 2 052 条交互。Digital_Music 数据集比较稀疏, 包含了 16 510 名用户的音乐喜好, 涵盖了 11 358 个数字音乐产品, 记录了 135 244 次

用户与音乐的交互。Video_Games 数据集稀疏程度适中, 包含了 55 220 名玩家、17 294 个视频游戏标题以及 386 787 条交互记录。CDs_and_Vinyl 数据集密度较高, 整理了 147 592 名用户对 140 260 个 CD 和黑胶唱片商品的收藏偏好, 总计 1 454 745 次用户与商品的交互。Automotive 数据集密度高, 整理了 193 621 名汽车爱好者的全面偏好数据, 涵盖 79 039 款车型, 并详细记录了 1 322 876 次用户与汽车的交互。Movies_and_TV 数据集密度高, 整理了 297 498 名电影和电视爱好者的广泛观看记录, 涵盖 59 944 个媒体标题, 共计 2 814 153 次用户与内容的交互。

CTiSASRec 通过融入项目信息提升了模型的用户建模能力。这一优势在一般和密集数据集上都有积极的结果, 在稀疏数据集中遇到了轻微的挫折。这种差异可能源于稀疏环境中用户行为数据的缺乏, 此时融入项目信息并未显著增强模型对用户偏好的理解。

为了获得顺序推荐数据集, 对每个用户访问过的项目按照时间戳进行排序, 得到用户的交互序列。鉴于模型对时间编码的要求, 本文通过减去最小时间戳来调整每个序列中的交互时间, 确保每个用户序列都从时间“0”开始。提取每个序列的最后一个项目构成测试数据集, 提取序列中的倒数第 2 个项目构成验证数据集, 其余序列项目构成训练数据集。

3.2 评价指标

本文采用两种流行的 top- k 排名指标, 即 HR@10 和 NDCG@10, 评估推荐性能。HR@10 衡量推荐列表中排名前 10 位的项目命中的比率, 如式 (12) 所示。

$$\text{HR} = \frac{1}{S} \sum_{i=1}^S \text{hit}(i) \quad (12)$$

其中, S 为测试样本数, $\text{hit}(i)$ 用于表示第 i 个测试项是否包含在推荐列表中。若在, 则其值为 1, 否则为 0。

NDCG@10 不仅考虑命中率，还考虑推荐列表中不同位置的分数，如式 (13) 所示。

$$NDCG = \frac{1}{S} \sum_{i=1}^S \frac{1}{\text{lb}(p_i + 1)} \quad (13)$$

其中， p_i 为第 i 个测试项在模型推荐的项目列表中的位置。

对于每个用户，随机选择 100 个负项目，并与正项目组合在一起，由推荐模型对这些项目进行排名。然后，根据这 101 个项目的排名计算 HR@10 和 NDCG@10。

3.3 基准模型

(1) GRU4Rec^[3]: 采用门控循环单元来精心建模用户会话中固有的顺序依赖性。

(2) SASRec^[1]: 根据交互时间排列序列项目，采用自注意力机制进行序列建模。

(3) TiSASRec^[2]: 引入了时间自注意力机制，利用项目之间的时间间隔进行序列建模。

(4) HTP^[13]: 不仅考虑了绝对时间和时间间隔，还考虑了其他时间信息，如周期性。

3.4 对比结果

不同模型之间的性能比较结果见表 2，加粗的值表示所有模型中的最好性能，带下划线的值表示次好性能。从表 2 可知，与 GRU4Rec 等门控

循环网络模型相比，CTiSASRec 在捕捉用户行为模式中的长期依赖性和时间波动方面展现出卓越的能力。这一优势归功于 CTiSASRec 将自注意力机制与项目时间信息结合，使模型能够更有效地把握和建模用户行为的时间趋势。与 SASRec 等传统基于注意力的顺序推荐模型相比，CTiSASRec 通过融入项目信息提升了模型的用户建模能力。这一优势在一般密集和密集数据集上都有积极的作用，而在稀疏数据集中遇到了轻微的挫折。这种差异可能源于稀疏环境中用户行为数据的缺乏，此时融入项目信息并未显著增强模型对用户偏好的理解。值得注意的是，CTiSASRec 在不同数据集上以显著优势超越了 TiSASRec，这得益于其结合了对比学习和绝对时间信息，对比学习增强了模型从更广泛的用户行为数据中提取通用表征的能力，绝对时间信息的融入提高了模型辨别影响用户行为的细微时间差别的能力。虽然 CTiSASRec 和 HTP 都利用了绝对和相对时间信息，但 CTiSASRec 通过独特的绝对时间处理方式和对比学习策略来加强用户建模，这种独特的方法支撑了 CTiSASRec 在各种不同稀疏程度的数据集上都能取得显著的性能提升。

表 2 不同模型之间的性能比较结果

| 数据集 | 评价指标 | GRU4Rec | SASRec | TiSASRec | HTP | CTiSASRec |
|---------------|---------|---------|----------------|----------------|----------------|----------------|
| Gift | NDCG@10 | 0.052 3 | <u>0.245 1</u> | 0.233 9 | 0.241 5 | 0.287 3 |
| | HR@10 | 0.171 1 | <u>0.465 1</u> | 0.325 3 | 0.375 9 | 0.500 0 |
| Digital_Music | NDCG@10 | 0.265 1 | 0.514 5 | 0.491 9 | 0.510 5 | <u>0.513 0</u> |
| | HR@10 | 0.478 5 | 0.659 2 | 0.579 1 | 0.644 9 | <u>0.655 1</u> |
| Video_Games | NDCG@10 | 0.352 4 | 0.478 4 | 0.492 8 | 0.511 3 | <u>0.505 3</u> |
| | HR@10 | 0.476 8 | 0.691 9 | 0.700 5 | 0.731 7 | <u>0.730 7</u> |
| CDs_and_Vinyl | NDCG@10 | 0.186 3 | 0.354 2 | <u>0.410 8</u> | <u>0.315 8</u> | 0.424 5 |
| | HR@10 | 0.298 5 | 0.477 4 | <u>0.517 5</u> | <u>0.440 1</u> | 0.543 1 |
| Automotive | NDCG@10 | 0.321 4 | 0.401 2 | <u>0.414 3</u> | <u>0.447 4</u> | 0.455 9 |
| | HR@10 | 0.412 0 | 0.606 5 | <u>0.612 2</u> | <u>0.635 9</u> | 0.649 8 |
| Movies_and_TV | NDCG@10 | 0.449 5 | 0.621 3 | 0.610 2 | <u>0.633 2</u> | 0.643 2 |
| | HR@10 | 0.578 1 | 0.826 2 | 0.747 3 | <u>0.848 4</u> | 0.857 2 |



3.5 消融实验

本节通过消融实验深入探讨了模型各个组件的复杂工作机制，旨在阐明它们对模型整体性能的各自贡献，消融实验结果如图3所示，其中 **only_pos** 表示仅将位置信息纳入位置编码的实验，**only_abs** 表示仅将项目的绝对时间信息纳入位置编码的实验，

no_contrast 表示模型中不考虑对比学习层，**no_hot** 表示自注意力层不使用项目流行度信息。由图3可知，CTiSASRec的每个组件都在一定程度上对推荐性能有所贡献，移除任何组件都会导致不同程度的性能下降。相比之下，位置信息和绝对时间信息对模型性能的影响较为显著，而项目流行度信息的影

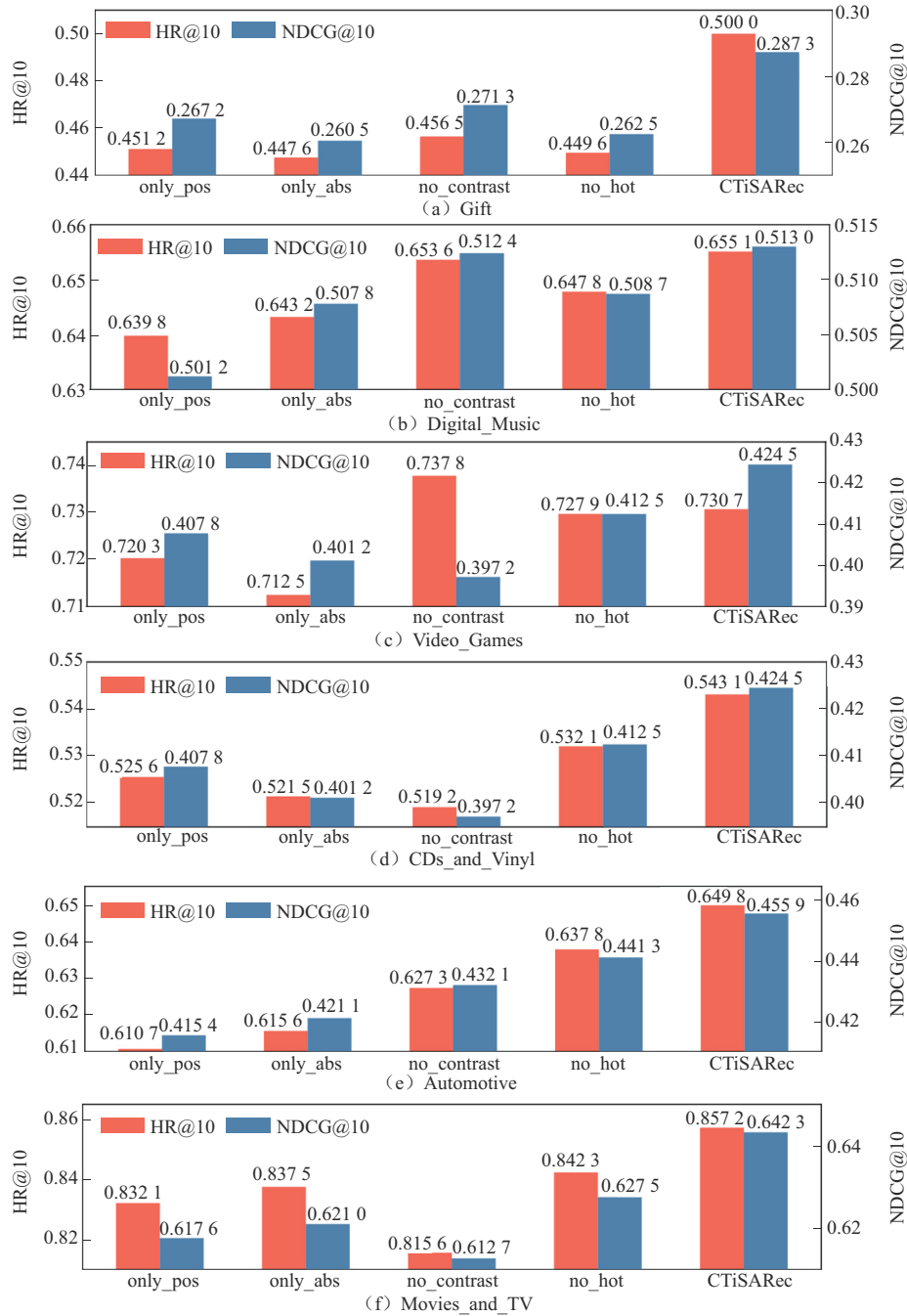


图3 消融实验结果

响相对较小。对比学习模块也对模型性能有较大影响，尤其是在 *Movies_and_TV* 这样的数据密集型数据集上。此外，不同组件的性能在不同数据集上存在差异。*only_abs* 在小规模、稀疏的数据集上对模型的推荐效果有着不错的提升，但在大规模和密集的数据集上没有很大的提升。*only_pos* 在大规模和密集的数据集上比 *only_abs* 效果更好，但小规模 and 稀疏的数据集上并没有 *only_abs* 效果好，这说明本文所提模型在不同规模的数据集上性能都有一定的提升。对比学习模块在大规模和密集的数据集上的提升十分显著，这一特性极大地促进了模型在海量数据中精确捕捉并学习数据之间关联的能力，项目流行度所有的数据集上对性能都有一定的提升，但没有特别突出。

3.6 参数敏感性实验

本文模型的主要参数包括学习率、对比学习中的“负样本对”数量以及正则化参数。本文逐一测试这些参数的敏感性，调参方法是挑选一个参数在一定范围内进行逐次递增，然后固定其余的参数。在 *Gift* 数据集上，学习率对性能的影响、正则化参数 λ 对性能的影响、对比学习负对数量对性能的影响和正则化参数 β 对性能的影响分别如图4、图5、图6和图7所示。可以看出，当学习率为0.0015、负样本对数量为2、正则化参数分别为0.00015和0.00010时，CTiSASRec模型达到了最佳的推荐性能。因此，本文模型训练过程中采用了这些参数设置。

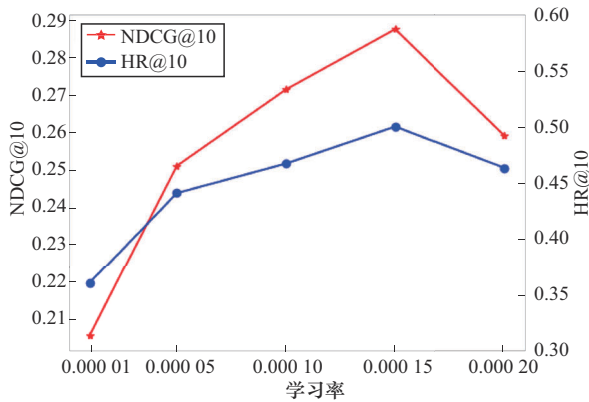


图4 学习率对性能的影响

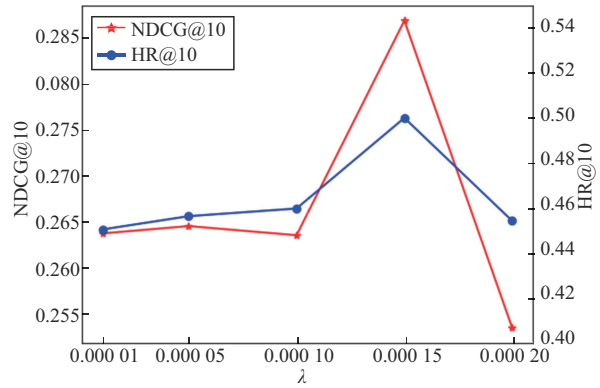


图5 正则化参数 λ 对性能的影响

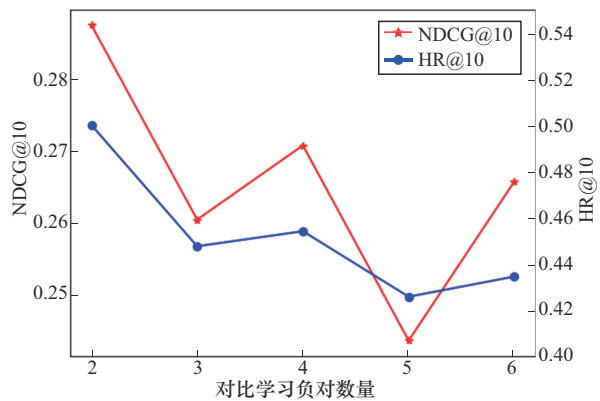


图6 对比学习负对数量对性能的影响

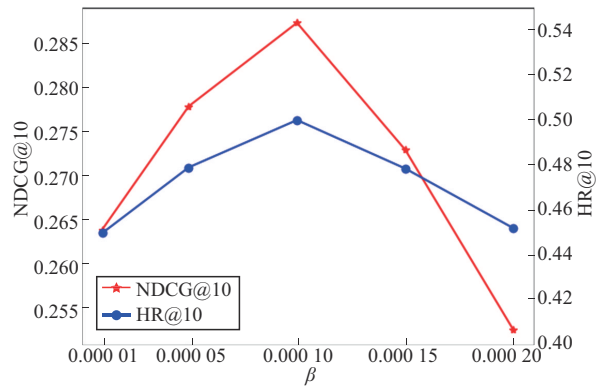


图7 正则化参数 β 对性能的影响

4 结束语

本文提出了一种新型的序列推荐模型 CTiSASRec，实现了时间感知自注意力机制与对比增强策略的融合。该模型能够灵活地融合多维度信息，包括项目间的时间间隔、绝对时间戳、项目序列位置、项目流行度和用户评分，从而精细



地建模用户偏好的动态演变。此外,对比学习模块的引入不仅改善了用户建模个性化程度,还通过多任务学习和联合优化显著提升了推荐性能。在6个类型、规模和稀疏度各异的数据集上进行的实验表明,CTiSASRec的性能明显优于目前最先进的基线模型。

CTiSASRec也存在一些局限性,首先,模型采用固定长度的序列以简化处理过程,对于有大量交互的活跃用户的长期行为可能会被掩盖,而且这种固定长度的方法缺乏灵活性,无法适应不同用户活动水平和交互频率的差异;其次,虽然模型融入了项目流行度和用户评分,但它们与其他特征(如对高分项目的小众偏好)的复杂相互作用仍有待深入探索;另外,模型没有区分用户的长期和短期兴趣,这可能会影响即时需求预测场景的准确性;最后,对比层的简单实现阻碍了多样对比任务的充分探索,从而限制了模型性能的优化,而且采用两次建模获取正样本对的模式,使模型的运行时间成倍增加。

未来设想对模型进行迭代增强,首先,引入针对用户实际交互数据的动态序列长度机制,在模型中融入长期和短期行为模式的刻画。例如:针对大量交互数据的活跃用户在模型中应该使用更长的交互序列,这样可以保证获取更精准的长期兴趣,对于非活跃用户应该使用较短的交互序列,这样可以减少空值对模型构建短期兴趣的影响。其次,利用先进的深度学习方法,如分解机和神经网络,来发现特征之间的复杂相互作用。最后,设计多样的对比学习任务,如预测未来多时间点的行为和对用户行为类型进行分类,以全面评估和增强模型的性能和泛化能力。

参考文献:

- [1] 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.
- [2] LI J C, WANG Y J, MCAULEY J. Time interval aware self-attention for sequential recommendation[C]//Proceedings of the 13th International Conference on Web Search and Data Mining. New York: ACM Press, 2020: 322-330.
- [3] HIDASIB, KARATZOGLOUA, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks[EB]. 2015.
- [4] ZHANG Y C, YIN G S, DONG Y X, et al. Contrastive learning with frequency domain for sequential recommendation[J]. Applied Soft Computing, 2023(144): 110481.
- [5] YUE Z R, WANG Y Q, HE Z K, et al. Linear recurrent units for sequential recommendation[C]//Proceedings of the 17th ACM International Conference on Web Search and Data Mining. New York: ACM Press, 2024: 930-938.
- [6] LIN G Y, GAO C, ZHENG Y, et al. Dual-interest factorization-heads attention for sequential recommendation[C]//Proceedings of the ACM Web Conference 2023. New York: ACM Press, 2023: 917-927.
- [7] TRAN V A, SALHA-GALVAN G, SGUERRA B, et al. Attention mixtures for time-aware sequential recommendation[C]//Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2023: 1821-1826.
- [8] MA H K, XIE R B, MENG L, et al. Triple sequence learning for cross-domain recommendation[J]. ACM Transactions on Information Systems, 2024, 42(4): 1-29.
- [9] HAN D, HUANG Y F, LIU J M, et al. LSAB: user behavioral pattern modeling in sequential recommendation by learning self-attention bias[J]. ACM Transactions on Knowledge Discovery from Data, 2024, 18(3): 1-20.
- [10] 金楠, 王瑞琴, 陆悦聪. 基于艾宾浩斯遗忘曲线和注意力机制的推荐算法[J]. 电信科学, 2022, 38(10): 89-97.
JIN N, WANG R Q, LU Y C. Ebbinghaus forgetting curve and attention mechanism based recommendation algorithm[J]. Telecommunications Science, 2022, 38(10): 89-97.
- [11] ZHU Y, LI H, LIAO Y, et al. What to do next: modeling user behaviors by TimeLSTM[C]//Proceedings of the Twentysixth International Joint Conference on Artificial Intelligence. Melbourne: AAAI Press, 2017:3602-3608.
- [12] WANG C Y, ZHANG M, MA W Z, et al. Modeling item-specific temporal dynamics of repeat consumption for recommender systems[C]//Proceedings of the World Wide Web Conference. New York: ACM Press, 2019: 1977-1987.
- [13] CHEN R, LIANG G T, MA C R, et al. HTP: exploiting holistic temporal patterns for sequential recommendation[C]//Proceedings of the 2023 International Joint Conference on Neural Net-

works (IJCNN). Piscataway: IEEE Press, 2023: 1-8.

- [14] QIN Y F, JU W, WU H J, et al. Learning graph ODE for continuous-time sequential recommendation[C]//Proceedings of the IEEE Transactions on Knowledge and Data Engineering. Piscataway: IEEE Press, 2024: 3224-3236.
- [15] SUN X X, DOU J F, WANG S D. Spatial-temporal aware self attention mechanism for next POI recommendation[C]//Proceedings of the 2024 IEEE 6th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). Piscataway: IEEE Press, 2024: 312-315.
- [16] WANG Z Y, 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. New York: ACM Press, 2020: 169-178.

[作者简介]



于洋 (1999-), 男, 湖州师范学院信息工程学院硕士生, 主要研究方向为数据挖掘、自然语言处理、个性化推荐。



王瑞琴 (1979-), 女, 博士, 湖州师范学院信息工程学院教授, 主要研究方向为自然语言处理、社交网络分析、个性化推荐。