http://bhxb.buaa.edu.cn jbuaa@buaa.edu.cn
DOI: 10.13700/j.bh.1001-5965.2024.0795

融合双通道注意力和时间编码的图神经网络推荐算法

白鹤，李钰*，郝明，徐婷婷
(辽宁理工学院 信息工程学院，锦州 121000)

摘　　要：针对多数图神经网络推荐算法未结合时间信息导致信息浪费和推荐结果不准确的问题，提出融合双通道注意力和时间编码的图神经网络推荐算法GUITRec。该模型充分利用数据集中信息，将用户自身特征、物品信息及用户与物品交互行为建为图结构，通过图神经网络进行数据聚合以形成用户和物品表征行为，从而更全面地捕捉两者特征及关系。引入双通道注意力机制卷积块注意力模块(CBAM)为用户和物品表征行为赋予不同权重，使模型依据权重进行深度学习，更加关注重要特征信息。引入时间编码器模型对用户与物品交互时间及用户位置编码，新增日、周、月周期编码和位置编码，充分挖掘时间和位置信息，为推荐提供更丰富的上下文。在Ciao数据集上进行实验，在平均绝对误差(MAE)、均方误差(MSE)、均方根误差(RMSE)这3个评价指标上，GUITRec均优于基线模型。

关 键 词：双通道注意力机制；图神经网络；时间编码；位置编码；推荐算法
中图分类号：TN911.1; TP301.6
文献标志码：A　　**文章编号**：1001-5965(2026)11-3864-15

随着5G时代的到来，社交媒体平台的迅速崛起进一步推动了信息和数据的爆炸性增长，频繁的互联网使用使人们面临着海量信息的冲击。其中夹杂着大量并不能直接满足用户需求或兴趣的冗余内容，增加了用户处理信息的负担。面对复杂多变的互联网信息，用户常常难以快速筛选并提取出真正需要的内容。同时，信息的更新速度远远超过了用户的处理能力，导致用户难以跟上信息发展的节奏。这种信息处理能力与信息更新速度之间的错位，直接造成了信息使用效率的下降，这种现象被称为“信息过载”。Goldberg等[1]于1992年提出推荐系统的概念，该概念的出现有效弥补了搜索引擎的缺陷，有效利用用户和物品的信息为用户带来个性化的推荐，从而过滤冗余信息，提升了用户的体验感。

整体上看，推荐系统主要经过3个发展阶段：浅层模型、神经网络模型、基于图神经网络的模型。传统的推荐模型[2]在关于协同过滤效果的分析方面，主要基于用户与物品相似性而实现，但在获取用户与物品特征信息方面存在不足。基于此，有学者基于浅层模型[3]提出协同过滤方法，即以用户和物品的表示学习呈现推荐任务，如矩阵分解[4]与因子分解机[5]，但在用户行为信息较为复杂的情况下，浅层方法很难充分发挥预期作用，无法从用户行为数据中获取大量信息[6]。在此背景下，有学者基于神经网络提出推荐模型，通过深度学习来增强推荐系统的表达能力，例如，将多层感知机与矩阵分解相结合，或采用深度因子分解机来提升推荐效果，如有机结合矩阵分解与多层感知，或借助深度因子分解机等模型。尽管如此，该类模型未将用

收稿日期：2024-11-06；**录用日期**：2025-01-03；**网络出版时间**：2025-01-08 15:39
网络出版地址：link.cnki.net/urlid/11.2625.V.20250108.1414.001
基金项目：2023年度辽宁省教育厅基本科研项目
*通信作者. E-mail：1531281593@qq.com

引用格式：白鹤，李钰，郝明，等. 融合双通道注意力和时间编码的图神经网络推荐算法[J]. 北京航空航天大学学报，2026，52(11)：3864-3878. Bai H，Li Y，Hao M，et al. Graph neural network recommendation algorithm fused with dual-channel attention and time encoding[J]. Journal of Beijing University of Aeronautics and Astronautics，2026，52(11)：3864-3878 (in Chinese).

户与物品的交互数据中的图结构信息、高阶结构信息用于预测与训练中。图神经网络是近几年兴起的一种推荐算法[7], 带来了全新的解决方案, 其消息传递机制与图结构有聚合高阶邻域特征信息的作用, 从而更精准地捕捉节点之间的关系和特征相互作用。这使图神经网络逐渐成为推荐系统中最具潜力的方法之一[7]。闫阳等[8]提出融合语义信息与注意力的图神经网络推荐算法; 潘润超等[9]提出一种基于深度图神经网络的协同过滤推荐算法, 通过在图神经网络中加入初始残差连接和恒等映射, 显著提高了推荐系统的召回率和归一化折损累计增益。冯兴杰等[10]针对传统推荐算法未能充分利用用户和商品评论数据的缺陷, 提出了通过图神经网络提取关联关系及深度学习提取评论内容的优势, 结合用户和商品偏好进行特征融合, 从而提高推荐效果。Zhong 等[11]在图卷积网络(graph convolutional network, GCN)中融合了多头注意力机制。此外, Scarselli 等[12]对图神经网络模型进行了详细分析, 而 Li 等[13]通过使用门控递归单元(gated recurrent unit, GRU)提升了图神经网络的表达能力。Zhao 等[14]进一步将 GRU 与 GCN 相结合, 提出了时间图卷积网络(temporal graph convolutional network, T-GCN)模型。Ma 等[15]基于概率矩阵分解(probabilistic matrix factorization, PMF)模型, 引入了用户社交关系矩阵, 提出了社交推荐(social recommendation, SoRec)模型, 从而有效缓解了冷启动问题; 邬硕等[16]利用图神经网络进行短期偏好学习, 利用双向长短时记忆(long short-term memory, LSTM)进行长期偏好兴趣学习, 获得用户长期偏好表示, 提高推荐算法的准确率; 于健等[17]提出一种基于高阶和时序特征的图神经网络社会化推荐算法模型, 利用 GRU 对用户的近期动态偏好, 采用注意力机制获取用户表征。然而, 大多数现有的图神经网络仅侧重于用户与项目之间的图结构关系, 忽视了时间上的特性, 即周期性。因此, 本文提出一种结合双通道注意力机制和时间编码的图神经网络推荐模型, 构建了用户-物品图和物品-物品图 2 个图结构, 分别通过图神经网络学习用户表示和物品表示, 并引入时间编码器建模用户与物品交互的时序信息, 同时, 利用双通道注意力机制为不同特征赋予权重, 最终通过残差网络融合各特征以生成最终的预测结果。本文创新如下:

1) 充分利用数据集中的信息, 将用户的自身特征、物品的信息及用户与物品的交互行为建立为图结构, 并通过图神经网络对上述图结构进行聚合, 形成用户表征行为和物品表征行为, 同时, 引入双通道注意力机制卷积块注意力模块(convolutional block attention module, CBAM)对用户和物品的表征行为赋予不同的权重, 以便后续模型根据权重进行深度学习。在该过程中, CBAM 的引入对于提升推荐效果具有重要作用。CBAM 结合了通道注意力和空间注意力, 分别对特征图的全局语义信息和局部空间信息进行动态加权, 有效突出关键特征。通过通道注意力机制, CBAM 捕捉用户全局兴趣特征和物品全局属性的重要性; 通过空间注意力机制, 聚焦用户与物品交互行为中重要模式。这种动态加权过程不仅增强了用户和物品表征的精准度, 而且使模型能够更精细地建模用户与物品之间的交互相关性。

2) 本文模型中引入了时间编码器模型, 对数据集中用户与物品交互的时间及用户所处的位置(即邮编)进行编码, 新增了日周期编码、周周期编码、月周期编码和位置编码, 充分对数据中的时间和位置等信息进行挖掘。时间编码模型通过捕捉用户在特定时间段的偏好和周期性行为规律, 可以更准确地预测用户在未来时间的偏好; 结合位置编码, 模型能够识别地域性的行为特征差异, 如不同地区用户的消费习惯, 从而为推荐提供更个性化的支持; 此外, 日、周、月等多粒度时间编码的引入, 使模型能够在不同时间尺度上建模用户的动态偏好变化, 提升推荐的时效性和精准度。

3) 在 Ciao 公共数据集上进行了充分实验, 以验证上述创新点的有效性, 并选择平均绝对误差(mean absolute error, MAE)、均方根误差(root mean square error, RMSE)和均方误差(mean squared error, MSE)作为评价指标, 选取经典的推荐算法作为基准模型。实验结果显示, 本文模型在 MAE、MSE 和 RMSE 上的表现均优于基准模型。此外, 本文还进行了消融实验和分析, 深入探讨了各部分对模型性能的影响。

1　相关工作

1.1　图神经网络推荐算法

图结构是一种常见且重要的数据结构, 能够有效对节点和边进行建模。其在推荐系统中具有广泛应用, 例如, 用户和项目的交互信息可以用节点来表示, 边则体现了用户与项目之间的关联关系[18]。但传统的深度学习模型很难充分利用这些多元化的数据结构, 从而导致部分信息的丢失或浪费, 进而影响推荐系统的整体性能。伴随着社交网络和知识图谱等众多图形数据的出现, 如何有效处理和挖掘这些复杂结构中的信息成为提升推荐系统性

能的关键挑战。随着社交网络、知识图谱等大量图形数据的涌现，如何有效处理这些数据成为关键问题。因此，图神经网络应运而生，以更好地应对这些挑战。图神经网络是一种基于递归神经网络(recurrent neural network, RNN)和卷积神经网络(convolutional neural network, CNN)发展而来的新型深度学习模型，特别适用于处理非欧几里得空间的数据。在推荐系统中，图神经网络的基本框架如图1所示。

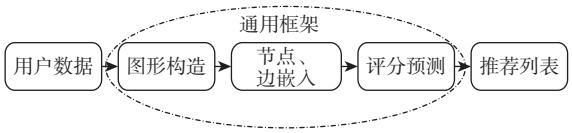

图1 图神经网络推荐算法框架

Fig. 1 Graph neural network recommendation algorithm framework

1.2 融合时间的推荐算法

在信息爆炸的当下，推荐算法至关重要，而时间因素对其准确性意义重大。时间影响着用户兴趣和行为，用户偏好随时间变化，新兴趣不断出现，旧喜好可能淡化。推荐算法若能捕捉时间信息，分析不同时间段用户行为模式和兴趣动态，就能更精准推荐符合当下需求的内容[19]。如在购物领域，不同季节和节日用户消费需求不同，推荐算法考虑时间因素可提高推荐准确性和用户购物体验；在影视娱乐方面，时间变化会使热门题材和风格演变，结合时间维度能推送当下流行作品满足用户娱乐需求，因此，研究人员开始将时间信息引入到推荐算法的研究中。Azri等[20]提出Auto TimeSVD++模型，引入了时间维度作为评分预测的关键因素，通过对用户偏好和物品流行度随时间的变化进行建模，实现了对时间敏感的个性化推荐；He等[21]提出动态可扩展图卷积(dynamically expandable graph convolution, DEGC)算法，将用户时间偏好模型作为用户嵌入初始化，以更好地捕捉个体层面的偏好变化。

1.3 通道卷积注意力机制 CBAM

通道卷积注意力机制[22]CBAM的核心思想是对输入特征图进行两阶段的提炼：①通道注意力模块，即借助通道注意力模块确定重要通道；②空间注意力模块，即通过空间注意力模块明确富含信息的部分。这样的双重注意力机制使CBAM可以全方位地捕捉特征中的关键信息。图2展示了CBAM注意力机制的构造。

1.3.1 通道注意力模块

通道注意力模块的原理主要是关注输入的特征图中具有意义的内容。分别采用平均池化层与

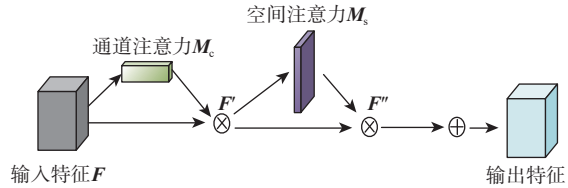

图2 CBAM框架

Fig. 2 CBAM framework

最大池化层对输入特征图的空间维度进行压缩处理，将其前向传递至一个共享的多层感知机网络。将2个输入进行相加，并经过一个激活函数，从而产生通道注意力，如式(1)和式(2)所示。图3为通道注意力的结构。

$$\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{F})=\sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{F}))+\mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{F})))=\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\boldsymbol{F}^{\mathrm{c}}_{\mathrm{avg}}))+\boldsymbol{W}_1(\boldsymbol{W}_0(\boldsymbol{F}^{\mathrm{c}}_{\mathrm{max}}))) \tag{1}$$

$$\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{F})=\sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{F}))+\mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{F})))=\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\boldsymbol{F}^{\mathrm{c}}_{\mathrm{avg}}))+\boldsymbol{W}_1(\boldsymbol{W}_0(\boldsymbol{F}^{\mathrm{c}}_{\mathrm{max}}))) \tag{2}$$

式中：σ为sigmod激活函数；MLP()为全连接层；AvgPool()为平均池化层；$\boldsymbol{W}_0$、$\boldsymbol{W}_1$为各级权重；MaxPool()为最大池化层；$\boldsymbol{F}$为输入特征图；$\boldsymbol{F}^{\mathrm{c}}_{\mathrm{avg}}$和$\boldsymbol{F}^{\mathrm{c}}_{\mathrm{max}}$分别为通道平均池化和最大池化后的特征图。

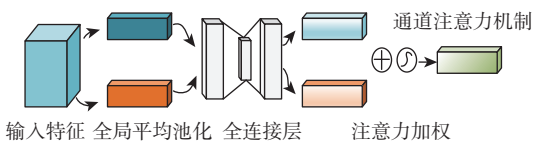

图3 通道注意力结构

Fig. 3 Channel attention structure

1.3.2 空间注意力模块

空间注意力模块与通道注意力有所不同，其聚焦于具有信息性的部分，与通道注意力形成互补。具体步骤为：首先，沿着通道轴分别进行平均池化和最大池化操作，提取出不同的全局特征表示；然后，将这2种池化操作的结果连接起来，产生2个图；最后，利用一个卷积操作后再串联一个激活函数，从而得到空间注意力$\boldsymbol{M}_{\mathrm{s}}$，如式(3)所示。图4为空间注意力机制的结构。

$$\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{F})=\sigma(f_{\mathrm{c}}^{k\times k}([\mathrm{AvgPool}(\boldsymbol{F});\mathrm{MaxPool}(\boldsymbol{F})]))=\sigma(f_{\mathrm{c}}^{k\times k}([\boldsymbol{F}^{\mathrm{s}}_{\mathrm{avg}};\boldsymbol{F}^{\mathrm{s}}_{\mathrm{max}}])) \tag{3}$$

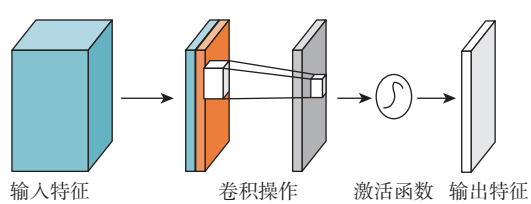

图4 空间注意力机制结构

Fig. 4 Structure of spatial attention mechanisms

式中: f_c为卷积操作; k为卷积核大小; $\boldsymbol{F}^{\mathrm{s}}_{\mathrm{avg}}$和$\boldsymbol{F}^{\mathrm{s}}_{\mathrm{max}}$分别为空间平均池化和最大池化后的特征图。

1.4　SENet 注意力机制

压缩与激励网络 (squeeze-and-excitation networks, SENet) 是一种专注于通道注意力的机制, 通过 Squeeze-and-Excitation 操作提升网络对重要通道特征的关注。其核心思想是通过全局平均池化 (global average pooling, GAP)来获取通道全局特征, 通过权重重新分配, 强化重要特征, 图 5 展示了 SENet 的主要结构。SENet 的详细操作步骤如下。

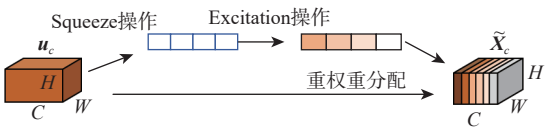

图 5　SENet 网络结构

Fig. 5　SENet network structure

1) Squeeze 阶段。该阶段主要通过 GAP 将输入的特征图在空间维度上压缩, 提取每个通道的全局特征。具体来说, 对于每个通道, 计算所有位置的平均值, 生成一个包含所有通道全局统计信息的特征向量z_c, 用以表示每个通道的整体重要性:

$$z_c = F_{\mathrm{sq}}(\boldsymbol{u}_c) = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\boldsymbol{u}_c(i,j) \tag{4}$$

式中: H和W分别为特征图的高和宽; $\boldsymbol{u}_c(i,j)$为通道c上特定位置的特征值; F_{sq}为 Squeeze 函数。

在该阶段会输出一个大小为$1\times1\times C$的特征向量$\boldsymbol{z}=[z_1,z_2,\cdots,z_c]$, 其包含了每个通道的全局语义信息。

2) Excitation 阶段。在该阶段, SENet 利用一个轻量级的 2 层全连接网络对 Squeeze 阶段提取的通道特征向量进行建模。共分为 2 个阶段: 第 1 层全连接将通道数降维到一个较小的维度C_r, 通过压缩比r减少计算复杂度:

$$\boldsymbol{f}_1 = \mathrm{ReLU}(\boldsymbol{W}_1\boldsymbol{z}+\boldsymbol{b}_1) \tag{5}$$

式中: $\boldsymbol{f}_1$为降维后的通道特征向量; $\boldsymbol{W}_1$为权重矩阵; $\boldsymbol{b}_1$为偏置项。

第 2 层全连接还原到原始通道维度C, 通过 sigmoid 激活函数生成每个通道的权重(注意力分数)。这些权重表示每个通道在特征图中的重要性:

$$\boldsymbol{s} = \sigma(\boldsymbol{W}_2\boldsymbol{f}_1+\boldsymbol{b}_2) \tag{6}$$

式中: $\boldsymbol{s}$为每个通道的权重, 元素范围在 0～1 之间; $\boldsymbol{W}_2$为权重矩阵; $\boldsymbol{b}_2$为偏置项。

该阶段会输出大小为$1\times1\times C$的注意力向量$\boldsymbol{s}$, 表示每个通道的权重。

3) 重权重分配。SENet 将 Excitation 阶段生成的通道权重应用到原始特征图上, 通过逐通道的加权操作重新调整每个通道的特征强度。权重高的通道被强化, 而权重低的通道被抑制, 从而提升对关键特征的关注, 优化模型对输入特征的表达能力:

$$\tilde{\boldsymbol{X}}_c = \boldsymbol{s}_c\boldsymbol{X}_c \tag{7}$$

式中: $\boldsymbol{s}_c$为通道c的权重; $\boldsymbol{X}_c$为通道c的特征图。

在该阶段会输出大小为$H\times W\times C$的特征图$\tilde{\boldsymbol{X}}_c$, 其中, 每个通道已经按照重要性重新加权。

1.5　Transformers 注意力机制

Transformers 的核心是其强大的注意力机制, 尤其是自注意力机制, 其使 Transformers 能够捕捉序列中任意位置的全局依赖关系。

自注意力机制能够更有效地学习结构化和非结构化数据的特征。实质是通过 key-value 的键值对注意力权重进行优化, 从而自动学习注意力权重。模型会从大量特征中学习到有用的特征, 并从特征之间构建注意力关系, 以便将关键特征更准确地抽取出来。在推荐算法中通过使用 query、key、value 来建立语义关系, 从而帮助推荐系统对用户偏好进行识别和学习。具体步骤如下:

将输入的信息$\boldsymbol{X}$线性投影到 3 个不同的空间, 得到 3 个不同的向量: 查询向量$\boldsymbol{Q}$、键向量$\boldsymbol{K}$、值向量$\boldsymbol{V}$:

$$\boldsymbol{Q} = \boldsymbol{W}_{\boldsymbol{Q}}\boldsymbol{X} \tag{8}$$

$$\boldsymbol{K} = \boldsymbol{W}_{\boldsymbol{K}}\boldsymbol{X} \tag{9}$$

$$\boldsymbol{V} = \boldsymbol{W}_{\boldsymbol{V}}\boldsymbol{X} \tag{10}$$

式中: $\boldsymbol{W}_{\boldsymbol{Q}}$、$\boldsymbol{W}_{\boldsymbol{K}}$、$\boldsymbol{W}_{\boldsymbol{V}}$为可学习参数。

根据键值对注意力机制来计算最后的输出:

$$\mathrm{attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\cdot\boldsymbol{V} \tag{11}$$

式中: attention() 为注意力机制的封装函数; softmax() 为归一化函数; d_k为 key 中的向量维度, 能够减小 key 维度变化对结果的影响, 使梯度更加稳定。

表 1 总结了 CBAM、SENet 和 Transfromers 的优势与不足。

综上所述, CBAM 相较于 SENet 和 Transformers 的主要优势在于轻量化设计和通道+空间注意力的结合, 既能高效捕获局部与全局特征, 又具备较强的多模态适应性。此外, CBAM 计算成本低, 适合大规模推荐系统, 且模块化设计便于集成, 比 SENet 更全面, 比 Transformers 更高效。

表1 CBAM、SENet和Transformers的优势与不足

Table 1 Advantages and disadvantages of CBAM, SENet, and Transformers

特征	主要关注点	结构	复杂度	处理能力	对稀疏特征的处理能力	计算资源	优点
CBAM	通道和空间注意力	2个子模块:通道注意力+空间注意力	轻量级,计算成本低	捕获局部与全局信息(通道和空间层次)	较强,可以捕捉到物品或用户行为特征的空间依赖	较低,适合大规模应用	轻量级、灵活、多模态融合能力强
SENet	通道注意力	压缩与激励	较轻量,计算成本较低	主要捕获通道级别的全局信息	较强,但主要针对通道特征	较低,适合资源有限的应用	高效、轻量、适合处理高维稀疏特征
Transformers	自注意力	自注意力层	计算复杂度高	捕获长序列中的全局依赖,适合长序列推荐	对长序列的稀疏特征建模效果好,但计算开销较大	高,需要较强的硬件支持(尤其是内存和计算力)	强大的全局依赖建模能力,适合复杂数据处理

2 模型框架

2.1 模型概述

针对传统的图神经网络使用简单的注意力机制提取图结构中的信息,没有更加细致准确地挖掘图结构中深层次的信息,而且忽略了数据集中用户与物品交互的时间及地点等信息,不仅造成了信息的浪费,而且未准确地对用户进行个性化推荐的问题。本文提出融合双通道注意力和时间编码的图神经网络推荐算法GUITRec,其具体结构如图6所示,模型一共分为3部分,分别是物品建模、用户建模和时间建模。

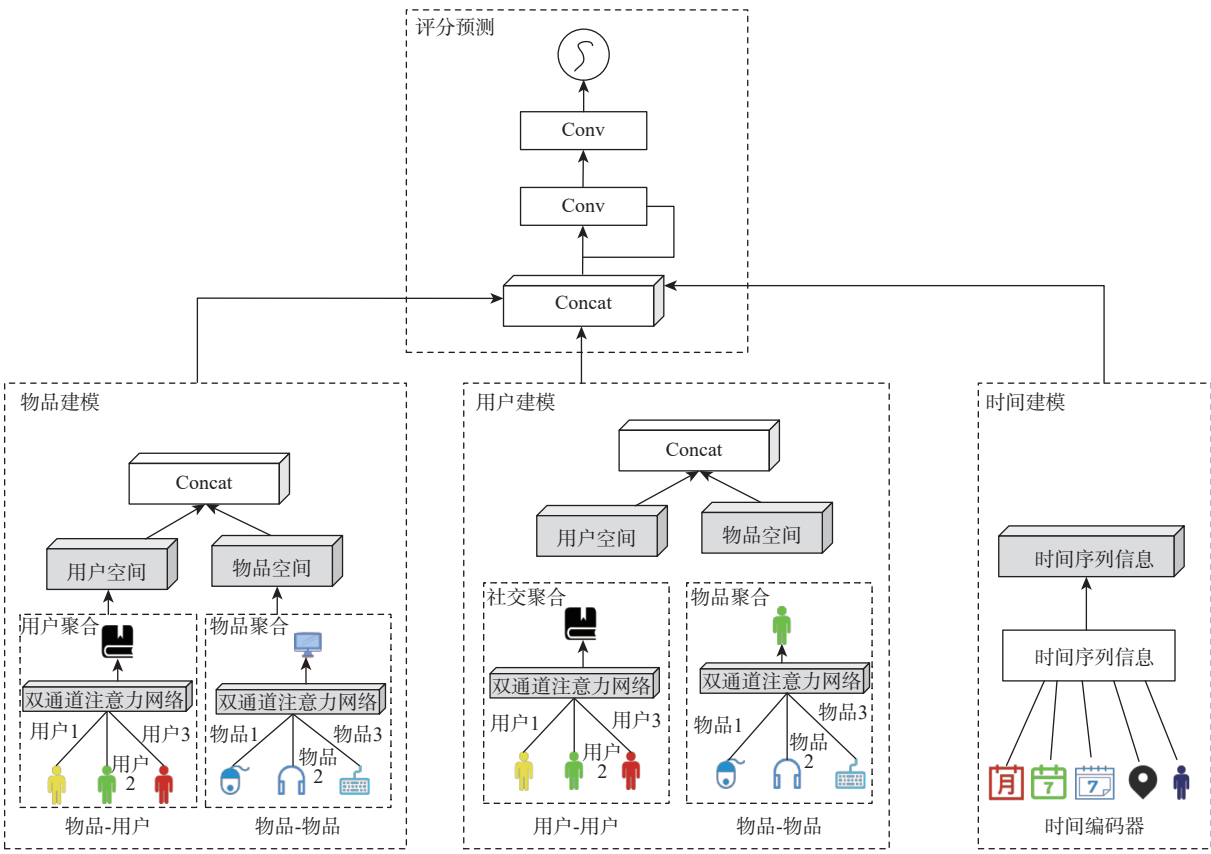

图6 GUITRec模型结构

Fig. 6 GUITRec model structure

2.2 物品建模

在物品建模过程中,物品的嵌入表示学习扮演着核心角色,其不仅捕捉了物品与用户交互行为的直接反馈,还深入挖掘了物品间复杂而微妙的相似关系,这为学习物品的特征关系提供了不同的视角。本文提出一种综合性的物品建模策略,融合了2种关键的聚合机制——用户聚合与物品聚合,旨在从数据集的丰富信息中高效且精确地学习物品的特征表示。

2.2.1 物品建模-用户聚合

用户-物品图作为一种关键数据结构,不仅包含了用户与物品之间的交互行为,还包含用户与物品

的评分。利用用户-物品图中丰富的交互数据与评分, 通过聚合函数, 能够挖掘并学习每个物品 v_j 在用户空间中的潜在特征表示 $\boldsymbol{z}_j^{\mathrm{U}}$, 从而更有效地捕捉物品与用户之间的关系和交互信息, 并收集、整合与其相关的用户交互记录与评分数据, 以此构建出既全面又精确的物品信息模型:

$$\boldsymbol{z}_j^{\mathrm{U}} = \sigma(\boldsymbol{W}' \cdot A_{\mathrm{users}}(\boldsymbol{f}_{jt}) + \boldsymbol{b}) \qquad \forall t \in B_{(j)} \tag{12}$$

式中: A_{users} 为用户聚合函数, 主要用于将用户的交互和评分信息进行聚合; $B_{(j)}$ 为与物品 v_j 交互过的用户集合; $\boldsymbol{f}_{jt}$ 为用户意见交互系数; $\boldsymbol{W}'$ 为权重矩阵; $\boldsymbol{b}$ 为偏置项。

即使针对同一个物品, 用户在与该物品的交互过程中, 受自身状态或周围环境等因素的影响, 可能会对该物品给出不同的评分。这种差异化的评分信息能够有效反映出用户兴趣的动态变化, 同时, 也能帮助捕捉物品的不同特征, 从而对物品的潜在表征进行更深入的学习和建模。因此, 在用户与物品 v_j 的交互过程中, 需要引入用户意见交互系数来表示用户对物品的动态评分变化。该系数可以量化用户对同一物品的不同意见, 有助于模型更准确地理解用户偏好随时间的变化及其对推荐结果的影响。

$$\boldsymbol{f}_{jt} = g_{\mathrm{u}}([\boldsymbol{p}_t \oplus \boldsymbol{e}_{\mathrm{r}}]) \tag{13}$$

式中: $\boldsymbol{p}_t$ 为原始的用户嵌入信息; $\boldsymbol{e}_{\mathrm{r}}$ 为用户的评分信息; "$\oplus$" 表示连接操作; g_{u} 为用户聚合信息的函数, 主要用于将评分信息和用户的原始信息进行融合。

对于聚合函数 A, 常对 $\boldsymbol{f}_{jt}$ 进行加权平均。但该方法将与物品 v_j 监护过的全部用户所含有的信息均认为是同等作用, 往往会忽视重要信息, 同时, 也会带来无用信息干扰最后的预测结果。针对上述问题, 本文将线性加权平均的聚合方式替换为通道卷积注意力机制即 CBAM。CBAM 注意力机制的好处是可以自适应地调整图结构中节点特征及节点之间关系的重要性, 使模型可以更好地捕捉图结构中的关键信息, 同时, 也可以更好地关注对推荐结果有重要影响的特征, 从而提升推荐的准确性。

引入 CBAM 注意力机制后, 物品 j 在用户空间中的潜在特征表示重新计算, 如式 (14) 所示, 不再对原始用户特征 $\boldsymbol{f}'_{jt}$ 直接求和, 而是对经过注意力加权后的用户特征进行聚合:

$$\boldsymbol{z}_j^{\mathrm{U}} = \sigma\left(\boldsymbol{W}' \sum_{\forall t \in B_{(j)}} \boldsymbol{f}''_{jt} + \boldsymbol{b}\right) \tag{14}$$

其中, 加权后的用户特征 $\boldsymbol{f}'_{jt}$ 由空间注意力模块对原始特征逐元素加权得到。具体地, 将原始用户特征 $\boldsymbol{f}_{jt}$ 输入空间注意力函数 $\boldsymbol{M}_{\mathrm{s}}(\cdot)$ 得到与 $\boldsymbol{f}_{jt}$ 同维度的空间注意力权重图, 再与原特征逐元素相乘, 表达式为

$$\boldsymbol{f}'_{jt} = \boldsymbol{M}_{\mathrm{s}}(\boldsymbol{f}_{jt}) \otimes \boldsymbol{f}_{jt} \tag{15}$$

为进一步描述特征, 将经过第一次加权后的特征 $\boldsymbol{f}'_{jt}$ 再次送入空间注意力模块进行二次加权, 得到更深层的精修特征, 计算如下:

$$\boldsymbol{f}''_{jt} = \boldsymbol{M}_{\mathrm{s}}(\boldsymbol{f}'_{jt}) \otimes \boldsymbol{f}'_{jt} \tag{16}$$

通道注意力模块的计算如下:

$$\begin{aligned}\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{f}_{jt}) &= \sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{f}_{jt})) + \mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{f}_{jt}))) = \\ &\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\mathrm{avg}(\boldsymbol{f}_{jt}^{\mathrm{c}}))) + \boldsymbol{W}_1(\boldsymbol{W}_0(\max(\boldsymbol{f}_{jt}^{\mathrm{c}}))))\end{aligned} \tag{17}$$

空间注意力模块的计算如式 (18) 所示: 对输入特征分别沿通道维度执行平均池化与最大池化, 得到 2 个二维空间描述图; 将两者在通道维度上拼接后, 经过一个 $k \times k$ 卷积层 $c^{k \times k}$ 进行融合, 再经 Sigmoid 激活函数 σ 生成空间注意力权重。

$$\begin{aligned}\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{f}_{jt}) &= \sigma(c^{k \times k}([\mathrm{AvgPool}(\boldsymbol{f}_{jt}); \mathrm{MaxPool}(\boldsymbol{f}_{jt})])) = \\ &\sigma(c^{k \times k}[\mathrm{avg}(\boldsymbol{f}_{jt}^{\mathrm{s}}); \max(\boldsymbol{f}_{jt}^{\mathrm{s}})])\end{aligned} \tag{18}$$

式中: "$\otimes$" 表示逐元素相乘; $\boldsymbol{f}_{jt}^{\mathrm{c}}$、$\boldsymbol{f}_{jt}^{\mathrm{s}}$ 分别为通道注意力、空间注意力机制中用户 i 与物品 j 的初始交互特征表示。

2.2.2　物品建模-物品聚合

物品和物品之间具有某些属性的相似, 并不是独立存在的。以鼠标与键盘为例, 这 2 类物品虽功能各异, 却共享了 "电子器材" 这一核心属性, 并紧密关联于电脑、平板等更广泛的电子设备生态系统之中, 因此, 在构建物品特征模型时, 需超越单一物品的界限, 深入探索其与其他物品之间的相似性与关联性。

为此, 本文提出在物品空间中引入 "物品学习系数" 概念, 旨在通过聚合具有相似关系或共享属性的物品群体, 来增强单个物品的特征表示能力。该方法不仅考虑了物品自身的固有属性, 还融入了其在整个物品网络中的位置与角色, 从而实现了对物品特征的更全面、更深入的刻画。具体聚合方式为

$$\boldsymbol{h}_j^{\mathrm{I}} = \sigma(\boldsymbol{W}' \cdot A_{\mathrm{items}}(\boldsymbol{y}_{jk}) + \boldsymbol{b}) \qquad \forall k \in I_{(j)} \tag{19}$$

式中: $\boldsymbol{h}_j^{\mathrm{I}}$ 为物品 j 经过相似物品聚合后得到的隐藏表示向量; A_{items} 为物品聚合函数; $\boldsymbol{y}_{(j)}$ 为与物品 v_j 相似的物品集合; $\boldsymbol{y}_{jk}$ 为物品相似系数。

当用户和物品发生交互行为时, 可以通过物品的评分或与目标物品相似度的物品评分等信息作为参考。例如, 当《热辣滚烫》上映时, 想去观看这部影片的人不仅会去关注这部影片的评分, 而且还会去看贾玲导演或参与拍摄的其他影片的评分情

况以选择是否观看该部影片。因此，对于物品v_j和物品v_k中具有相似的属性，引入物品相似性感知系数$\boldsymbol{y}_{jk}$来表示物品之间的相似性：

$$\boldsymbol{y}_{jk}=g_{\mathrm{v}}([\boldsymbol{q}_k\oplus\boldsymbol{k}_{\mathrm{s}}]) \tag{20}$$

式中：$\boldsymbol{q}_k$为与物品具有相似属性的嵌入信息；g_{v}为融合物品相似信息和相似度系数的函数；$\boldsymbol{k}_{\mathrm{s}}$为物品之间的相似度系数，本文采用皮尔逊相似度系数。

同理，物品之间的相似信息也同样不是同等重要的。例如，键盘和鼠标这2个物品之间重要的相似信息是“电子设备”“电脑”，不重要的相似性信息则是“青年人”。因此，应引入注意力机制对不同的相似信息赋予不同的权重。因此，本文聚合函数引入了通道卷积注意力机制CBAM对物品之间的相似信息进行监控，给重要的相似性信息赋予较大的权重，给不重要的信息或无用的信息赋予较小的权重：

$$\boldsymbol{h}_j^{\mathrm{I}}=\sigma\left(\boldsymbol{W}'\sum_{\forall k\in I_{(j)}}\boldsymbol{y}_{jk}^{''}+\boldsymbol{b}\right) \tag{21}$$

$$\boldsymbol{y}'_{jk}=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{y}_{jk})\otimes\boldsymbol{y}_{jk} \tag{22}$$

$$\boldsymbol{y}_{jk}^{''}=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{y}'_{jk})\otimes\boldsymbol{y}'_{jk} \tag{23}$$

$$\begin{aligned}\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{y}_{jk})&=\sigma(\mathrm{MLP}(\mathrm{avg}(\boldsymbol{y}_{jk}))+\mathrm{MLP}(\max(\boldsymbol{y}_{jk})))=\\&\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\mathrm{avg}(\boldsymbol{y}_{jk}^{\mathrm{c}})))+\boldsymbol{W}_1(\boldsymbol{W}_0(\max(\boldsymbol{y}_{jk}^{\mathrm{c}}))))\end{aligned} \tag{24}$$

$$\begin{aligned}\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{y}_{jk})&=\sigma(c^{k\times k}([\mathrm{avg}(\boldsymbol{y}_{jk});\max(\boldsymbol{y}_{jk})]))=\\&\sigma(c^{k\times k}[\mathrm{avg}(\boldsymbol{y}_{jk}^{\mathrm{s}});\max(\boldsymbol{y}_{jk}^{\mathrm{s}})])\end{aligned} \tag{25}$$

式中：$\boldsymbol{y}_{jk}^{\mathrm{c}}$、$\boldsymbol{y}_{jk}^{\mathrm{s}}$分别为通道注意力、空间注意力物品相似性感知系数。

2.2.3 物品表示

为充分地对物品的特征进行学习，需要将用户聚合和物品聚合的结果向量进行拼接，形成最终的物品特征向量：

$$\boldsymbol{I}=[\boldsymbol{z}_j^{\mathrm{U}}\oplus\boldsymbol{h}_j^{\mathrm{I}}] \tag{26}$$

2.3 用户建模

用户空间的建模与物品空间的建模相似，均采用2种聚合方式，即用户聚合和物品聚合。这是因为用户不仅与物品存在交互关系，还可能与其他用户存在好友关系。因此，使用这2种建模方式，不仅能够深入挖掘用户与物品之间的互动，还能够通过用户之间的相似性捕捉到更多潜在信息。这种双重聚合方法能够更全面地刻画用户行为，提升模型对用户偏好和物品特征的理解能力。

2.3.1 用户建模-物品聚合

在物品聚合函数中，将多个用户交互过的相同物品信息进行聚合，得到聚合后的隐藏表示向量：

$$\boldsymbol{h}_i^{\mathrm{I}}=\sigma(\boldsymbol{W}'\cdot A_{\mathrm{items}}(\boldsymbol{X}_{ia}+\boldsymbol{b}))\qquad\forall a\in C_{(i)} \tag{27}$$

式中：$C_{(i)}$为与用户u_i交互过的物品集合；$\boldsymbol{X}_{ia}$为用户u_i与项目v_a的交互评分信息。

当用户和物品进行交互时，可以通过不同的方式来展现自己对该物品的喜恶，例如，人们在看电影时可以通过对电影的评分、是否向周围的人推荐作为自己的评价方式；人们在看抖音、快手等小视频时，可以通过停留时间、是否感兴趣等来展示自己的喜好。上述信息对用户建模是非常重要的，可以更好地建立用户画像，更加精确地捕捉用户偏好。因此，引入用户嵌入向量$\boldsymbol{e}_{\mathrm{r}}\in\mathbf{R}_d$，$d$为嵌入向量维度，将用户的评分等信息融入向量中，同时，引入$\boldsymbol{X}_{ia}$作为物品表征向量和用户向量融合的向量：

$$\boldsymbol{X}_{ia}=[\boldsymbol{q}_{\mathrm{u}}\oplus\boldsymbol{e}_{\mathrm{r}}] \tag{28}$$

式中：$\boldsymbol{q}_{\mathrm{u}}$为用户$u_i$与物品$v_a$交互的物品向量。

同理，用户建模的聚合函数仍采用通道卷积注意力机制：

$$\boldsymbol{h}_i^{\mathrm{I}}=\sigma\left(\boldsymbol{W}'\sum_{\forall a\in C_{(i)}}\boldsymbol{X}_{ia}^{''}+\boldsymbol{b}\right) \tag{29}$$

$$\boldsymbol{X}'_{ia}=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{X}_{ia})\otimes\boldsymbol{X}_{ia} \tag{30}$$

$$\boldsymbol{X}_{ia}^{''}=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{X}'_{ia})\otimes\boldsymbol{X}'_{ia} \tag{31}$$

$$\begin{aligned}\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{X}_{ia})&=\sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{X}_{ia}))+\\&\mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{X}_{ia})))=\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\mathrm{avg}(\boldsymbol{X}_{ia}^{\mathrm{c}})))+\\&\boldsymbol{W}_1(\boldsymbol{W}_0(\max(\boldsymbol{X}_{ia}^{\mathrm{c}}))))\end{aligned} \tag{32}$$

$$\begin{aligned}\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{X}_{ia})&=\sigma(c^{k\times k}([\mathrm{AvgPool}(\boldsymbol{X}_{ia});\mathrm{MaxPool}(\boldsymbol{X}_{ia})]))=\\&\sigma(c^{k\times k}[\mathrm{avg}(\boldsymbol{X}_{ia}^{\mathrm{s}});\max(\boldsymbol{X}_{ia}^{\mathrm{s}})])\end{aligned} \tag{33}$$

式中：$\boldsymbol{X}_{ia}^{\mathrm{c}}$、$\boldsymbol{X}_{ia}^{\mathrm{s}}$分别为通道注意力、空间注意力物品表征向量和用户向量融合的向量。

2.3.2 用户建模-用户聚合

用户偏好会受到周围人或偏好相似用户的影响，因此，对用户的偏好进行聚合十分重要。但不同的用户受到社交的影响强度不同，因此，采用CBAM注意力机制对目标用户不同的社交赋予不同的权重。需要先获取用户-用户的信息：

$$\boldsymbol{h}_i^{\mathrm{s}}=\sigma(\boldsymbol{W}'\cdot A_{\mathrm{friends}}(\boldsymbol{h}_f+\boldsymbol{b}))\qquad\forall f\in F_{(i)} \tag{34}$$

将用户-用户的相似向量$\boldsymbol{h}_f^{\mathrm{I}}$送入CBAM注意力机制中，即用户聚合函数$A_{\mathrm{friends}}$为CBAM注意力机制：

$$\boldsymbol{h}_i^{\mathrm{s}}=\sigma\left(\boldsymbol{W}'\sum_{\forall f\in f_{(i)}}\boldsymbol{h}_f^{''}+\boldsymbol{b}\right) \tag{35}$$

$$\boldsymbol{h}'_f=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{h}_f)\otimes\boldsymbol{h}_f \tag{36}$$

$$\boldsymbol{h}_f^{''}=\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{h}'_f)\otimes\boldsymbol{h}'_f \tag{37}$$

$$\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{h}_f)=\sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{h}_f))+\mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{h}_f)))=\sigma(\boldsymbol{W}_1(\boldsymbol{W}_0(\mathrm{avg}(\boldsymbol{h}_f^{\mathrm{c}})))+\boldsymbol{W}_1(\boldsymbol{W}_0(\max(\boldsymbol{h}_f^{\mathrm{c}}))))\tag{38}$$

$$\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{h}_f)=\sigma(c^{k\times k}([\mathrm{AvgPool}(\boldsymbol{h}_f);\mathrm{MaxPool}(\boldsymbol{h}_f)]))=\sigma(c^{k\times k}[\mathrm{avg}(\boldsymbol{h}_f^{\mathrm{s}});\max(\boldsymbol{h}_f^{\mathrm{s}})])\tag{39}$$

式中: $\boldsymbol{h}_f^{\mathrm{c}}$、$\boldsymbol{h}_f^{\mathrm{s}}$分别为通道注意力、空间注意力用户-用户的相似向量。

2.3.3　用户表示

为充分地对用户的特征进行学习, 需要将用户聚合和物品聚合的结果向量进行拼接, 形成最终的用户特征向量:

$$\boldsymbol{U}=[\boldsymbol{h}_i^{\mathrm{I}}\oplus\boldsymbol{h}_i^{\mathrm{s}}]\tag{40}$$

2.4　时间建模

该模型中的时间编码采用了多维度的方式进行建模。通过局部时间戳、全局时间戳、周编码、月编码等不同层次的时间建模方式全面地记录和反映时间信息。这种时间建模方式可以捕捉不同时间尺度下的数据变化和特征。局部时间戳可以更精细地定位特定事件或行为发生的具体时间点; 全局时间戳则可以提供一个整体的时间参照, 便于在更大的时间范围内进行数据分析和比较。周编码和月编码有助于从周期性的角度观察数据的趋势和规律, 如某些行为在每周或每月的特定时间段会呈现出一定的模式。通过这些多维度的时间编码, 可以更深入地理解数据与时间的关系, 为后续的数据分析和算法应用提供更丰富的时间信息基础。

相比于现有的时间建模方法(如 LSTM 等时间序列模型), 本文提出的时间编码方法具有显著的优势和特点。传统的时间序列模型(如 LSTM、GRU)主要通过递归方式逐步捕捉时间序列的动态变化, 适合处理连续的时间序列数据。然而, 这些方法通常只关注时间信息的顺序依赖性, 对时间中的全局、局部和周期性特征的捕捉能力有限, 且当序列较长时容易产生梯度消失等问题。

本文提出的时间编码方法则采用多维度建模, 对局部时间、全局时间和周期性时间特征进行编码。该方法能够捕捉时间的局部和全局规律, 使模型既可以精准定位特定时间点的行为特征, 又可以从更大的时间范围捕获行为趋势; 周期性时间(周编码、月编码)的加入, 能够自然地捕捉用户行为的周期性模式, 无需额外设计复杂的递归结构。时间编码的结构如图 7 所示。

2.4.1　全局时间编码

在全局时间编码中, 本文先将时间戳转化为数值型的秒级时间, 再提起年份、月、日和周等信息。在推荐系统中, 时间的周期性十分明显。如音乐推荐场景中, 周一早上人们常需充满活力的音乐开启工作, 周五晚上则倾向轻松音乐迎接周末; 电商推荐中, 换季时人们对不同季节的服装有周期性需求; 影视推荐方面, 节假日人们倾向大片或喜剧片, 工作日则可能选择短时长剧集或纪录片。推荐系统考虑时间周期性, 能够有效地捕捉时间的周期性。无论是一天、一周还是一年, 时间都具有明显的周期性。通过正弦和余弦函数, 可以将时间戳映射到一个周期为相应时间长度的曲线上。因此, 在全局时间编码阶段, 本文将对时间信息进行周期化表示, 详细信息如下:

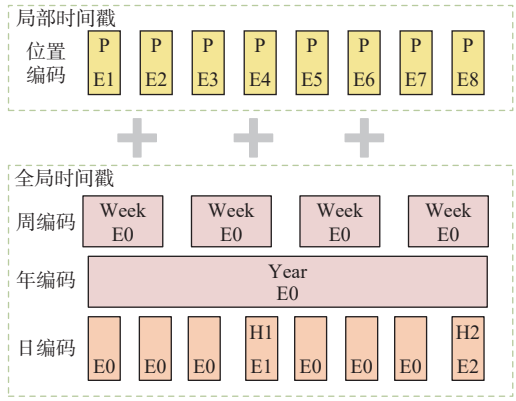

图 7　时间建模结构

Fig. 7　Time modeling structure

1) 日编码。其可反映短期用户行为变化, 适应短期趋势并评估短期推荐效果, 如根据一天不同时段调整推荐内容:

$$\boldsymbol{n}_{\mathrm{day}}=\left[\sin\left(\frac{2\pi T}{24\times60\times60}\right),\cos\left(\frac{2\pi T}{24\times60\times60}\right)\right]\tag{41}$$

式中: $\boldsymbol{n}_{\mathrm{day}}$为日编码; T为时间戳, s; 分母 24×60×60 表示一天的秒数。

将时间戳映射到一个周期为一天的正弦和余弦曲线上, 这两者的结合使时间信息可以嵌入一个二维圆形轨迹中, 能够同时保留周期性特征和连续性特征。例如, 当时间戳 T=43 200 s 时, 即为一天中的中午 12 点, 将 T 代入式(41)中可得到当前时间戳对应的日编码为[0, −1], 表示中午 12 点的时间特征。

2) 周编码。工作日和周末差异, 适应周期性活动, 还能评估推荐系统的稳定性, 如根据不同日子提供不同类型推荐, 以满足用户多样化需求:

$$\boldsymbol{n}_{\mathrm{week}}=\left[\sin\left(\frac{2\pi T}{24\times60\times60\times7}\right),\cos\left(\frac{2\pi T}{24\times60\times60\times7}\right)\right]\tag{42}$$

式中: $\boldsymbol{n}_{\mathrm{week}}$为周编码; 分母$24\times60\times60\times7$为一周的秒数。

周编码能够捕捉一周内的行为变化, 如周一到

周五的工作日和周六、周日的休息日模式。例如，T=304 200 s，即周三中午12点，将T代入式(42)中即可得到周编码为[0,−1]，表示一周中周三的中午时间特征。

3) 年编码。其能捕捉季节性和周期性需求，用于长期用户行为分析和规划长期推荐策略：

$$\boldsymbol{n}_{\text{year}}=\left[\sin\left(\frac{2\pi T}{24\times60\times60\times365}\right),\cos\left(\frac{2\pi T}{24\times60\times60\times365}\right)\right] \tag{43}$$

式中：$\boldsymbol{n}_{\text{year}}$为年编码；分母$24\times60\times60\times365$为一年的秒数。

年编码能够捕捉一年中长周期的时间变化，如季节性消费习惯或节假日购物模式。例如，T=15 768 000 s，即一年中的第182天，即6月底，将T代入式(43)中即可得到年编码为[0,−1]，表示一年中的时间特征。

2.4.2 局部时间编码——位置编码

在Transformers中，位置编码起着至关重要的作用，其为输入序列中的每个位置赋予独特的表示，使模型能够捕捉序列中的顺序信息。将邮编作为位置信息的输入，例如，邮编“12345”可以被看作一个长度为5的序列，每个数字占据一个位置。对于邮编序列，设向量维度为d_{model}，邮编位置索引为p_{os}，取值范围为0～4，分别对应邮编中的5个位置。同时，l表示向量维度的索引。位置编码的计算式为

$$P_{(p_{\text{os}},2l)}=\sin\left(p_{\text{os}}/1\,000^{\frac{2l}{d_{\text{model}}}}\right) \tag{44}$$

式(44)表示在偶数维度上，位置编码由正弦函数计算得出。其中，p_{os}决定了正弦函数的相位，而随着l和d_{model}的变化，正弦函数的频率也会相应改变。这样可以为不同的位置和维度生成独特的编码值。

$$P_{(p_{\text{os}},2l+1)}=\cos\left(p_{\text{os}}/1\,000^{\frac{2l}{d_{\text{model}}}}\right) \tag{45}$$

在奇数维度上，位置编码由余弦函数计算得出。与正弦函数类似，通过调整p_{os}、l和d_{model}，可以得到不同位置和维度的独特编码值。

最后得到的位置编码是正余弦值相互交替的：

$$\boldsymbol{P}_{p_{\text{os}}}=\begin{bmatrix}\sin(\omega_1 p_{\text{os}})\\ \cos(\omega_1 p_{\text{os}})\\ \sin(\omega_2 p_{\text{os}})\\ \cos(\omega_2 p_{\text{os}})\\ \vdots\\ \sin(\omega_{d/2} p_{\text{os}})\\ \cos(\omega_{d/2} p_{\text{os}})\end{bmatrix} \tag{46}$$

式中：$\omega_{d/2}=1/1\,000^{\frac{2l}{d_{\text{model}}}}$。

将局部时间编码和位置时间编码进行拼接，形成最终的时间信息编码：

$$\boldsymbol{T}_{\text{s}}=[\boldsymbol{n}_{\text{day}},\boldsymbol{n}_{\text{week}},\boldsymbol{n}_{\text{year}},\boldsymbol{P}_{p_{\text{os}}}] \tag{47}$$

2.5 评分预测及模型训练

在评分预测部分，将用户建模、物品建模和时间建模部分所输出的向量进行拼接融合，再将融合向量送入前馈网络中进行评分预测：

$$\begin{cases}\boldsymbol{g}_1=[\boldsymbol{U}\oplus\boldsymbol{I}\oplus\boldsymbol{T}_{\text{s}}]\\ \boldsymbol{g}_3=\sigma(\boldsymbol{W}_3\boldsymbol{g}_2+\boldsymbol{b}_3)\\ \boldsymbol{g}_2=\sigma(\boldsymbol{W}_2\boldsymbol{g}_1+\boldsymbol{b}_2)\\ \boldsymbol{g}_2=\sigma(\boldsymbol{W}_2\boldsymbol{g}_1+\boldsymbol{b}_2)\\ \quad\vdots\\ \boldsymbol{g}_{n-1}=\sigma(\boldsymbol{W}_n\boldsymbol{g}_{n-1}+\boldsymbol{b}_n)\end{cases} \tag{48}$$

$$R'_{ij}=\boldsymbol{W}'^{\text{T}}\boldsymbol{g}_{n-1} \tag{49}$$

式中：$\boldsymbol{g}_n$为第n层的特征表示；R'_{ij}为用户u_i对物品v_j的预测评分。

由于本文采用预测评分，在训练过程中采用损失函数L_{s}：

$$L_{\text{s}}=\frac{1}{2|N|}\sum_{i,j\in N}(R'_{ij}-R_{ij}) \tag{50}$$

式中：N为用户-物品的交互数量；R_{ij}为用户u_i对物品v_j的真实评分。

3 实验结果及分析

3.1 实验数据集

本文采用Ciao数据集，其来源于Ciao.co.uk，是一个用户生成内容的平台，用户可以在该平台对各种产品和服务进行评分和评论。Ciao数据集包含了大量的用户对DVD等产品的评分和评论信息，这些评分通常以数字形式(1～5)反映用户对产品的满意度或喜好程度。同时，数据集还包含了产品的类别信息及时间戳等信息。Ciao数据集的详细信息如表2所示。

表2 Ciao数据集的详细信息

Table 2 Details of the Ciao dataset

用户数量	物品数量	评分	评分密度	社交连接	社交密度
7 317	104 975	283 320	0.036 9	111 781	0.208 7

其中，评分密度是衡量用户评分数据稀疏程度的一个指标：

$$p_{\text{r}}=N_{\text{r}}/(N_{\text{u}}N_{\text{i}}) \tag{51}$$

式中：p_{r}为评分密度；N_{r}为评分数量；N_{u}为用户数量；N_{i}为物品数量。

$$p_s = N_c/[N_u(N_u-1)] \tag{52}$$

式中: p_s为社交密度; N_c为社交连接数。

3.2　评价指标

本文采用的评价指标为 MAE、MSE、RMSE。

1) MAE 是推荐算法中常用的评价指标, 其是预测值与实际值之差绝对值的平均数:

$$E_{\mathrm{MAE}} = \frac{1}{M}\sum_{m=1}^{M}|y_m-\hat{y}_m| \tag{53}$$

式中: y_m为第m个样本的真实输出值; $\hat{y}_m$为第m个样本的预测输出值; M为样本总数。

MAE 的值越小, 模型的预测越准确。

2) RMSE 衡量的是预测值与实际值之间的偏差程度, 具体计算为预测值与实际值之差平方和的平方根:

$$E_{\mathrm{RMSE}} = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(y_m-\hat{y}_m)^2} \tag{54}$$

RMSE 的取值范围为 0 到正无穷大, 数值越小表示模型的预测误差越小, 表明模型的预测能力越强。

3) MSE 用于衡量预测值与实际值之间平均误差的平方值, MSE 越小, 说明预测效果越好:

$$E_{\mathrm{MSE}} = \frac{1}{M}\sum_{m=1}^{M}(y_m-\hat{y}_m)^2 \tag{55}$$

3.3　实验设置

3.3.1　参数设置

将数据集随机划分 70% 为训练集, 20% 为验证集, 10% 为训练集。在所有的实验中, 参数均先使用均值为 0、标准差为 0.1 的标准正态分布进行初始化。超参数则利用 optuna 优化参数进行寻优, 其中, 用户嵌入大小的搜索范围为 [8,16,32,64,128,256], 批次大小的搜索范围为 [32,64,128,256,512], 学习率的搜索范围为 [0.000 5,0.001 0,0.005 0,0.010 0,0.050 0,0.100 0], Dropout 率的搜索范围为 [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9]。激活函数为 ReLU, 为防止训练过程中产生过拟合现象, 在训练过程中引入早停策略, 设置耐心值为 5 , 如果验证集上的 MSE 连续 5 次没有提高, 则中止实验, 显示评价指标。经过实验, 最优参数如表 3 所示。

表 3　参数最佳值

Table 3　Optimal parameter values

嵌入维度	学习率	批次大小	Dropout率
64	0.000 5	64	0.8

图 8 展示了不同嵌入维度对实验结果的影响。可知, 嵌入维度对推荐算法的性能有显著影响。在嵌入维度从 8 增加到 64 的过程中, 模型的性能逐步提升, 表现为 MAE、MSE 和 RMSE 指标值的显著下降, 分别在嵌入维度为 64 时达到最低值(约为 0.720、0.817 和 0.904), 表明此时模型的特征表示能力和预测效果达到最佳。嵌入维度为 64 提供了一个理想的平衡点, 能够充分捕获用户和物品之间的复杂关系, 同时, 避免因维度不足导致的信息丢失问题。然而, 当嵌入维度进一步增加到 128 和 256 时, 这些指标值均开始上升, 说明过高的嵌入维度可能引入噪声或导致模型过拟合, 降低泛化性能, 同时增加了计算成本。因此, 选择嵌入维度为 64 作为最佳值, 既能够捕获丰富的用户和物品特征, 又能够避免高维度带来的过拟合和计算资源浪费问题。

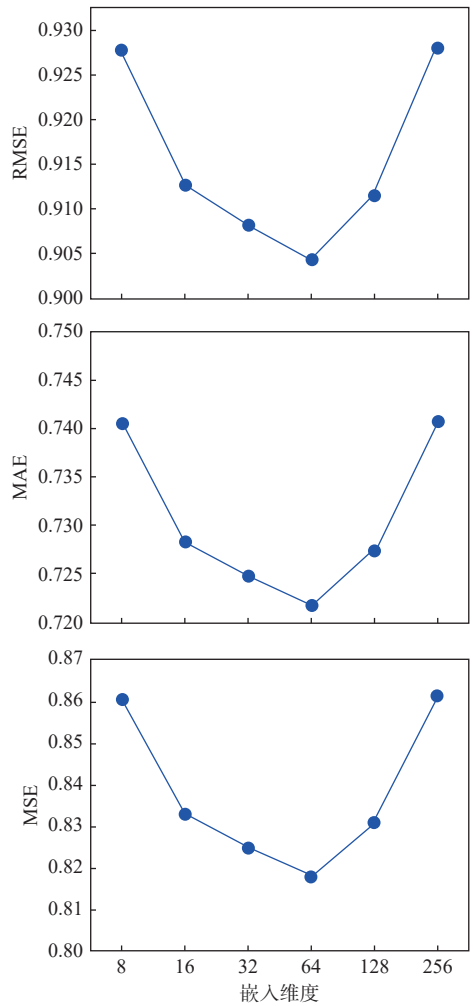

图 8　不同的嵌入维度实验结果

Fig. 8　Experimental results for different embedding dimensions

实验 1　学习率对实验结果的影响。由图 9 可以看出, 不同学习率下 RMSE、MSE 和 MAE 均表现出相似的变化趋势: 随着学习率的增加, 这 3 项指标逐步升高, 表明模型性能逐渐下降。当学习率为 0.000 5 时, RMSE、MSE 和 MAE 分别达到最低

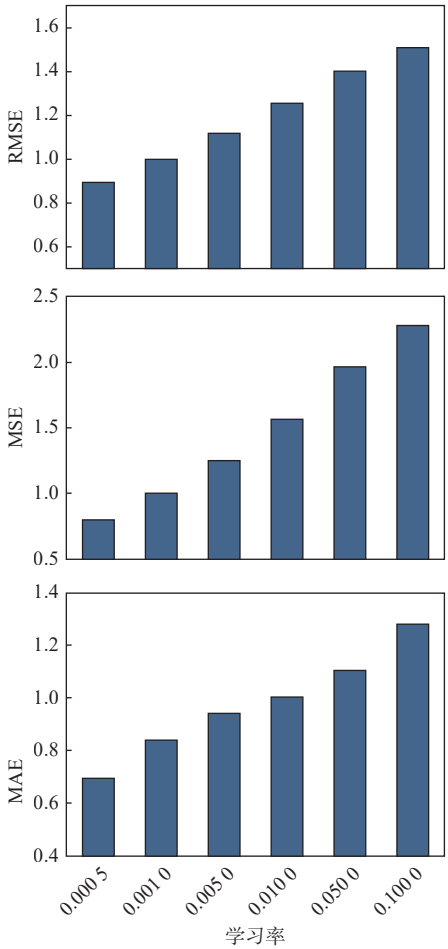

图 9 不同学习率对应的实验结果

Fig. 9 Experimental results corresponding to different learning rates

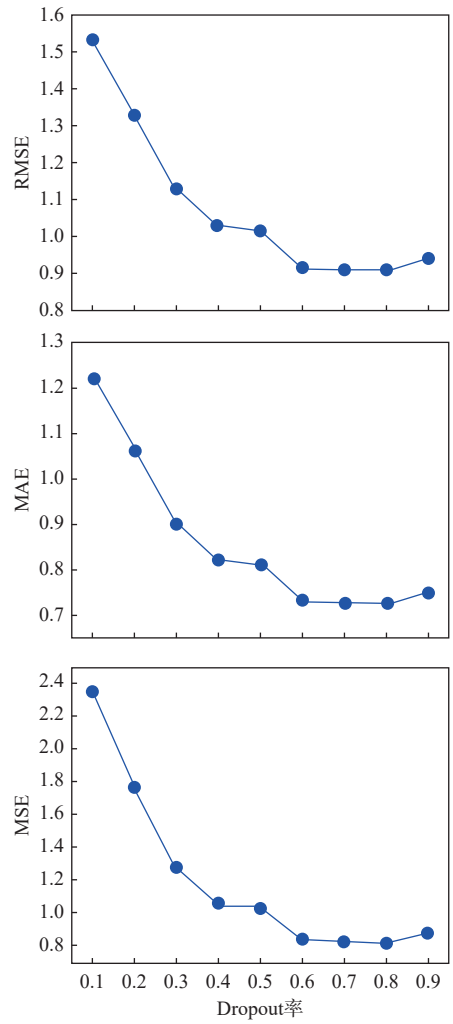

图 10 不同 Dropout 率对应的实验结果

Fig. 10 Experimental results corresponding to different Dropout rates

值，说明此学习率能够有效控制模型误差，实现最佳性能；而随着学习率增加至 0.010 0 及以上，RMSE、MSE、MAE 这 3 个指标不降反升，模型的优化能力显著下降。特别是学习率为 0.100 0 时，3 项误差指标达到最大值，表明过高的学习率导致模型优化过程失控。综上，在本实验中选择学习率的值为 0.000 5。

实验 2 Dropout 率对实验结果的影响。不同的 Dropout 率对模型有着重要的影响。通过对图 10 分析可以发现，随着 Dropout 率的逐步增加，模型的主要性能指标均呈现出逐渐下降的趋势，表明适当的 Dropout 率可以有效地抑制模型过拟合并提升泛化能力。当 Dropout 率达到 0.8 时，3 个性能指标均达到其最优值，此时模型实现了最优。然而，当 Dropout 率进一步增大至 0.9 时，指标不降反升，这表明过高的 Dropout 率会对模型训练带来不利影响。因此，选择 Dropout 率为 0.8 作为本次实验的最优值。

3.3.2 对比模型

1）PMF[23] 是一种基于概率模型的矩阵分解模型，通过将用户-项目评分矩阵分解为 2 个潜在因子矩阵，以处理稀疏数据并进行推荐系统中的预测。

2）SoRec 是一种结合社交网络信息进行推荐的模型，通过利用用户之间的社交关系来增强推荐系统的效果，认为用户的社交关系可以帮助更好地捕捉用户偏好，从而提高推荐的准确性和个性化。

3）社交矩阵分解（social matrix factorization，SocialMF）[15] 是一种推荐系统模型，其结合了矩阵分解技术和社交网络信息，通过考虑用户间的信任关系来提高推荐的准确性。

4）神经矩阵分解（neural matrix factorization，NeuMF）[24] 是一种将矩阵分解与神经网络结合的推荐模型，旨在利用神经网络的非线性建模能力来增强传统矩阵分解的表现。其通过同时学习线性（矩阵分解）和非线性（多层神经网络）特征的交互，能够捕捉到用户和项目之间更加复杂的关系，从而提高推荐精度。

5）基于图的推荐（graph-based recommendation，

GraphRec)[25] 是一种利用图神经网络技术的 SoRec 系统模型。其通过分析社交网络中的用户-物品交互关系图, 捕捉用户行为的深层社会影响, 为用户提供更加个性化和准确的推荐。

3.3.3　与基线模型对比结果

如表 4 所示, 将本文提出的 GUITRec 模型与多种基线模型进行对比, 以验证其有效性。对比的基线模型包括 PMF 模型、SoRec 模型、SocialMF 模型、NeuMF 模型和 GraphRec 模型, 实验数据的结果范围均在 0～1 之间, 以便统一衡量各模型的性能表现。

表 4　与基线模型对比数据

Table 4　Data compared to the baseline model

模型	MSE	MAE	RMSE	F_1
PMF[23]	0.896 4	0.755 6	0.946 8	0.744 7
SoRec	0.835 7	0.729 6	0.914 2	0.755 6
SocialMF[15]	0.819 2	0.722 3	0.905 1	0.757 7
NeuMF[24]	0.816 7	0.721 2	0.903 7	0.756 9
GraphRec[25]	0.813 6	0.727 4	0.902 0	0.758 6
GUITRec	0.813 1	0.720 0	0.902 0	0.762 1

在各项误差指标上, GUITRec 模型均表现出了优势。与 PMF 模型相比, GUITRec 模型的 MSE、MAE 和 RMSE 都有显著降低, 这表明 GUITRec 模型在预测用户对未知物品的评分方面更加准确。与其他结合社交网络信息的模型(如 SoRec 模型、SocialMF 模型和 GraphRec 模型)相比, GUITRec 模型也在误差指标上有更好的表现, 说明其在利用社交信息进行推荐方面更加有效。与 NeuMF 模型相比, 虽然两者在某些指标上较为接近, 但 GUITRec 模型的 MAE 更低, 说明 GUITRec 模型在 MAE 方面表现更好。

从实验结果显著性上来看, 本文提出的 GUITRec 模型在所有指标上均表现出色, 特别是在 F_1 分数上达到 0.762 1, 优于其他所有模型。相比传统 PMF 模型, GUITRec 模型的 F_1 分数提升了 2.34%, 相较于表现较好的 GraphRec 模型也有小幅提升。这表明, GUITRec 模型通过用户建模、物品建模和时间建模的方式, 显著优化了推荐的精确性和覆盖率, 能够更好地平衡推荐质量。

综上所述, 通过与多种基线模型的对比表明, GUITRec 模型在推荐准确性和有效性方面具有明显优势。

3.3.4　消融实验

为验证本文模型各部分的有效性, 设置了 3 组消融实验。

1) 为验证本文模型中用户和物品建模的有效性, 分别进行用户建模单独实验(GUITRec_user)、物品建模单独实验(GUITRec_item)及组合实验(GUITRec), 并对比不同实验的 MSE、MAE 和 RMSE 指标, 如图 11 所示。

从图 11 可以看出, 组合实验中, 加入用户建模和物品建模后, 推荐结果在各项误差指标上均有明显的改善。具体来说, MSE、MAE 和 RMSE 的降低意味着预测值与实际值之间的误差减小, 推荐结果更加准确。单独的用户建模和物品建模已经能够在一定程度上提升推荐效果, 而将两者结合起来进行组合实验, 效果进一步提升。这充分说明用户建模和物品建模在推荐系统中是有效的, 其能够为推荐结果提供更丰富的信息和更准确的预测, 从而提高推荐系统的性能和质量。

2) 设置 3 组对比实验来测试 CBAM 注意力机制在 GUITRec 模型中的有效性。对比实验分别为去掉 CBAM 注意力、换为 attention 注意力的 GUITRec_attention、将 CBAM 注意力换为 SENet 注意力机制的 GUITRec_SENet 及保留 CBAM 注意力机制的 GUITRec, 如图 12 所示。

通过对比可以发现, 在各项误差指标上, 保留

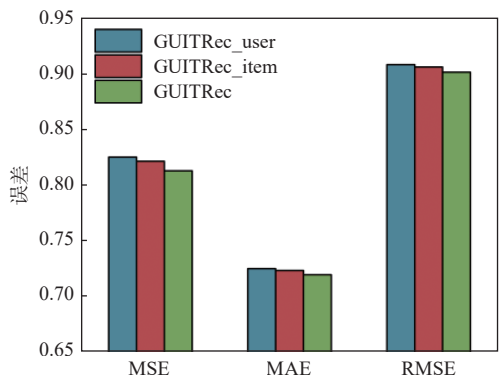

图 11　验证组合建模实验的有效性

Fig. 11　Validating effectiveness of combinatorial modeling experiments

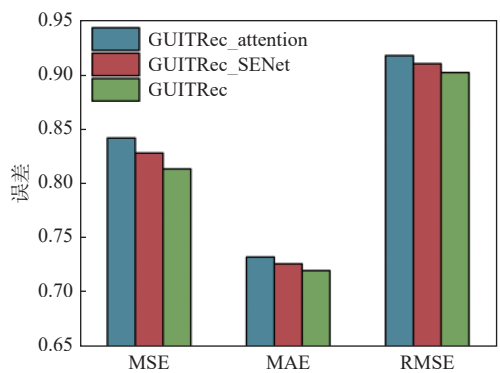

图 12　CBAM 注意力机制有效性实验数据

Fig. 12　Experimental data on effectiveness of CBAM attention mechanism

CBAM 注意力机制的 GUITRec 模型表现最佳。相比之下，换为 attention 注意力机制后，误差指标明显增大；换为 SENet 注意力机制后，虽然比 attention 注意力机制效果好，但仍不如 CBAM 注意力机制。这充分说明在 GUITRec 模型中，CBAM 注意力机制能够有效提升推荐结果的准确性。其能够更好地关注重要的特征信息，从而降低预测值与实际值之间的误差，提高推荐系统的性能。

3）为验证时间建模在推荐模型中的重要性，通过对比去掉时间建模的 GUITRec_notime 模型和包含时间建模的原 GUITRec 模型的实验结果，观察时间建模是否有效，如图 13 所示。

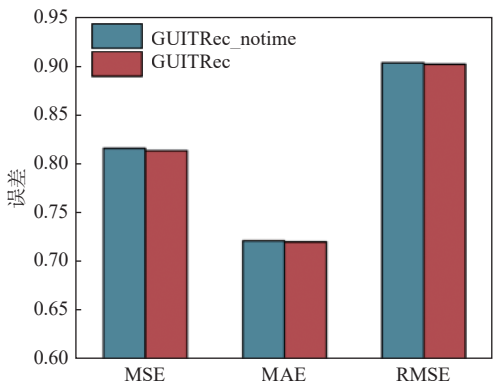

图 13 时间建模有效性实验数据

Fig. 13 Experimental data of time modeling effectiveness

从图 13 可以发现，GUITRec_notime 模型的各项指标均高于 GUITRec 模型的各项指标。在各项误差指标上，包含时间建模的原模型都表现更优。

具体来说，MSE、MAE 和 RMSE 的值越小，表示预测值与实际值之间的误差越小，推荐结果越准确。通过对比可以看出，加入时间建模后，模型的误差进一步降低，表明时间建模能够为推荐模型提供有价值的信息，帮助模型更好地理解用户行为和物品特征随时间的变化，从而提高推荐的准确性和有效性。

综上所述，实验结果充分证明了时间建模在推荐模型中的重要性，加入时间建模能够显著提升推荐结果。

4 结 论

本文针对当前多数图神经网络推荐算法在处理用户与物品交互数据时未充分考虑时间信息的问题，提出一种创新的融合双通道注意力和时间编码的图神经网络推荐算法 GUITRec，旨在通过更全面地捕捉和利用数据集中的各类信息，提升推荐系统的准确性和有效性。

1）GUITRec 将用户自身特征、物品信息及用户与物品之间的交互行为构建为图结构，并利用图神经网络进行数据聚合，以形成用户和物品的表征行为。这不仅充分利用了数据集中的信息，还有助于更深入理解用户和物品的特征及其相互关系。

2）为进一步提升模型的性能，引入了双通道注意力机制 CBAM。通过为用户和物品的表征行为赋予不同的权重，CBAM 使模型能够在进行深度学习时更加关注重要的特征信息，从而提高了推荐的准确性。

3）针对时间信息在推荐系统中的重要性，引入时间编码器模型。该模型不仅对用户与物品的交互时间进行编码，还考虑了用户位置信息，并新增了日、周、年编码和位置编码。这些额外的编码信息为推荐系统提供了更丰富的上下文，有助于模型更好地理解用户的行为模式和偏好。

4）为验证 GUITRec 算法的有效性，在 Ciao 数据集上进行了实验。实验结果表明，在 MAE、MSE 和 RMSE 这 3 个评价指标上，GUITRec 算法均优于基线模型。这一结果充分证明了 GUITRec 算法在推荐系统领域的优越性和实用性。

5）除上述实验结果外，还进行了充分的消融实验，以进一步验证模型中各个组件的有效性。实验结果表明，无论是双通道注意力机制还是时间编码器模型，都对提升模型性能起到了重要作用。这一发现不仅进一步证实了本文的设计思路，也为后续的研究提供了有益的参考。

综上所述，本文融合双通道注意力和时间编码的图神经网络推荐算法 GUITRec 有效提升了推荐效果。

参考文献（References）

[1] Goldberg D, Nichols D, Oki B M, et al. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12): 61-70.

[2] 张磊, 陈俊亮, 孟祥武, 等. 基于 BP 神经网络的协作过滤推荐算法[J]. 北京邮电大学学报, 2009, 32(6): 42-46.
Zhang L, Chen J L, Meng X W, et al. BP neural networks-based collaborative filtering recommendation algorithm[J]. Journal of Beijing University of Posts and Telecommunications, 2009, 32(6): 42-46(in Chinese).

[3] 舒新峰, 曹望美, 王曙燕. 多特征融合的 MOOC 推荐模型[J]. 计算机工程与应用, 2023, 59(10): 123-133.
Shu X F, Cao W M, Wang S Y. Multi-feature fusion based model for MOOC recommendation[J]. Computer Engineering and Applications, 2023, 59(10): 123-133(in Chinese).

[4] Rendle S. Factorization machines[C]//Proceedings of the 2010 IEEE International Conference on Data Mining. Piscataway: IEEE Press, 2010: 995-1000.

[5] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for

recommender systems[J]. Computer, 2009, 42(8): 30-37.

[6] 唐宇, 吴贞东. 基于残差网络的轻量级图卷积推荐方法[J]. 计算机工程与应用, 2024, 60(3): 205-212.
Tang Y, Wu Z D. Light graph convolution recommendation method based on residual network[J]. Computer Engineering and Applications, 2024, 60(3): 205-212(in Chinese).

[7] 吴静, 谢辉, 姜火文. 图神经网络推荐系统综述[J]. 计算机科学与探索, 2022, 16(10): 2249-2263.
Wu J, Xie H, Jiang H W. Survey of graph neural network in recommendation system[J]. Journal of Frontiers of Computer Science and Technology, 2022, 16(10): 2249-2263(in Chinese).

[8] 闫阳, 王雷全, 李家瑞. 融合语义信息与注意力的图神经网络推荐算法[J]. 计算机系统应用, 2023, 32(4): 214-222.
Yan Y, Wang L Q, Li J R. Graph neural network recommendation algorithm fused with semantic information and attention[J]. Computer Systems and Applications, 2023, 32(4): 214-222(in Chinese).

[9] 潘润超, 虞启山, 熊泓霏, 等. 基于深度图神经网络的协同推荐算法[J]. 计算机应用, 2023, 43(9): 2741-2746.
Pan R C, Yu Q S, Xiong H F, et al. Collaborative recommendation algorithm based on deep graph neural network[J]. Journal of Computer Applications, 2023, 43(9): 2741-2746(in Chinese).

[10] 冯兴杰, 生晓宇. 基于图神经网络与深度学习的商品推荐算法[J]. 计算机应用研究, 2021, 38(12): 3617-3622.
Feng X J, Sheng X Y. Item recommendation algorithm based on GNN and deep learning[J]. Application Research of Computers, 2021, 38(12): 3617-3622(in Chinese).

[11] Zhong T, Zhang S M, Zhou F, et al. Hybrid graph convolutional networks with multi-head attention for location recommendation[J]. World Wide Web, 2020, 23(6): 3125-3151.

[12] Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2009, 20(1): 61-80.

[13] Li Y J, Tarlow D, Brockschmidt M, et al. Gated graph sequence neural networks[C]//Proceedings of the 4th International Conference on Learning Representations. San Juan: ICLR, 2016.

[14] Zhao L, Song Y J, Zhang C, et al. T-GCN: a temporal graph convolutional network for traffic prediction[J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 21(9): 3848-3858.

[15] Ma H, Yang H X, Lyu M R, et al. SoRec: social recommendation using probabilistic matrix factorization[C]//Proceedings of the 17th ACM Conference on Information and Knowledge Management. New York: ACM, 2008: 931-940.

[16] 邬硕, 汪海涛, 姜瑛, 等. 融合图神经网络与长短期偏好的序列推荐算法[J]. 信息技术, 2024(2): 66-72.
Wu S, Wang H T, Jiang Y, et al. Sequence recommendation algorithm based on Graph Neural Network and long short term preference[J]. Information Technology, 2024(2): 66-72(in Chinese).

[17] 于健, 赵满坤, 高洁, 等. 基于高阶和时序特征的图神经网络社会化推荐算法研究[J]. 计算机科学, 2023, 50(3): 49-64.
Yu J, Zhao M K, Gao J, et al. Study on graph neural networks social recommendation based on high-order and temporal features[J]. Computer Science, 2023, 50(3): 49-64(in Chinese).

[18] 曾碧卿, 丁明浩, 宋逸云. 基于图卷积网络的知识感知对话生成[J]. 计算机系统应用, 2023, 32(7): 47-56.
Zeng B Q, Ding M H, Song Y Y. Knowledge-aware conversation generation based on graph convolutional network[J]. Computer Systems and Applications, 2023, 32(7): 47-56(in Chinese).

[19] 袁满, 褚润夫, 袁靖舒, 等. 融合上下文信息的图神经网络推荐模型研究[J]. 吉林大学学报 (信息科学版), 2023, 41(4): 693-700.
Yuan M, Chu R F, Yuan J S, et al. Research on graph neural network recommendation model of integrating context information[J]. Journal of Jilin University (Information Science Edition), 2023, 41(4): 693-700(in Chinese).

[20] Azri A, Haddi A, Allali H. AutoTimeSVD++: a temporal hybrid recommender system based on Contractive autoencoder and Matrix factorization[C]//Proceedings of the Smart Applications and Data Analysis. Berlin: Springer International Publishing, 2022: 93-103.

[21] He B W, He X, Zhang Y X, et al. Dynamically expandable graph convolution for streaming recommendation[C]//Proceedings of the ACM Web Conference 2023. New York: ACM, 2023: 1457-1467.

[22] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module[C]//Proceedings of the Computer Vision-ECCV 2018. Berlin: Springer International Publishing, 2018: 3-19.

[23] Yoo J, Choi S. Probabilistic matrix tri-factorization[C]//Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. Piscataway: IEEE Press, 2009: 1553-1556.

[24] Lee M, Choi P, Woo Y. A hybrid recommender system combining collaborative filtering with neural network[C]//Proceedings of the Adaptive Hypermedia and Adaptive Web-Based Systems. Berlin: Springer, 2002: 531-534.

[25] Bai T, Zhang Y J, Wu B, et al. Temporal graph neural networks for social recommendation[C]//Proceedings of the 2020 IEEE International Conference on Big Data (Big Data). Piscataway: IEEE Press, 2020: 898-903.

Graph neural network recommendation algorithm fused with dual-channel attention and time encoding

BAI He，LI Yu*，HAO Ming，XU Tingting

(College of Information Engineering，Liaoning Institute of Science and Engineering，Jinzhou 121000，China)

Abstract：In response to the problem that most graph neural network recommendation algorithms do not incorporate temporal information, leading to wasted information and inaccurate recommendations, this paper proposes a graph neural network recommendation algorithm, GUITRec, that fuses dual-channel attention with time encoding. Utilizing all of the data in the dataset, this model creates a graph structure based on the user's attributes, the item's details, and the user-item interaction behavior. Data aggregation is carried out using graph neural networks to create user and item representation behaviors, more thoroughly capturing the traits and connections of both. Secondly, the dual-channel attention mechanism convolutional block attention module (CBAM) is introduced to assign different weights to the user and item representation behaviors, enabling the model to learn according to those weights and pay more attention to important features. Finally, a time encoder model is introduced to encode the interaction time between the user and the item and the user's location, and new daily, weekly, and monthly cycle encodings and location encodings are added to fully mine time and location information and provide a richer context for recommendations. Experiments on the Ciao dataset show that GUITRec is superior to the baseline model in terms of the three evaluation indicators of mean absolute error (MAE), mean squared error (MSE), and root mean square error (RMSE).

Keywords：dual-channel attention mechanism；graph neural network；time encoding；location encoding；recommendation algorithm

Received：2024-11-06；**Accepted**：2025-01-03；**Published Online**：2025-01-08 15：39

URL：link.cnki.net/urlid/11.2625.V.20250108.1414.001

Foundation item：Basic Scientific Research Project of Liaoning Provincial Department of Education in 2023

* **Corresponding author**. E-mail：1531281593@qq.com