结合序列依赖与全局信息的会话推荐方法

曹家伟,段汶君,孙 倩,袁卫华

(山东建筑大学 计算机科学与技术学院,山东 济南 250101)

摘 要: 会话推荐旨在根据当前匿名行为序列预测下一个最可能的交互项目,研究的关键问题之一是如何利用项目的序列信息对匿名用户进行有效推荐。针对现有会话推荐方法未充分考虑序列依赖信息与来自其他会话的全局信息等问题,提出一种结合序列依赖与全局信息的会话推荐方法SDGI。该方法通过卷积时间感知的门控循环单元网络学习项目间的序列依赖关系,借助图神经网络构建局部与全局图以获取全局项目转移信息。为解决偏差与过拟合问题,引入一种结合门控机制的轻量级图卷积网络层获得全局级项目嵌入,并应用焦点损失函数处理正负样本不平衡问题。在Diginetica、Tmall、Yoochoose 3个公共数据集上与12种基线方法进行比较,实验结果表明,SDGI的性能相较基线方法有较大提升,结合序列依赖与全局信息能有效提高会话推荐性能。

关键词: 推荐系统;会话推荐;图神经网络;序列依赖

DOI: 10. 11907/rjdk. 241070 **开放科学(资源服务)标识码(OSID):**

中图分类号: TP391 **文献标识码:** A **文章编号:** 1672-7800(2025)001-0001-08

Session Recommendation Method Combining Sequence Dependency and Global Information

CAO Jiawei, DUAN Wenjun, SUN Qian, YUAN Weihua

(*School of Computer Science and Technology*, *Shandong Jianzhu University*, *Ji'nan* 250001, *China*)

Abstract: Session recommendation aims to predict the next most likely interaction item based on the current anonymous behavior sequence, and one of the key research questions is how to effectively recommend anonymous users using the sequence information of the item. Aiming at the problem that existing session recommendation methods do not fully consider sequence dependency information and global information from other sessions, a session recommendation method SDGI that combines sequence dependency and global information is proposed. This method learns the sequence dependency relationships between items through convolutional time aware gated recurrent unit networks, and constructs local and global graphs using graph neural networks to obtain global item transition information. To address the issues of bias and overfitting, a lightweight graph convolutional network layer combined with gating mechanism is introduced to obtain global level item embeddings, and a focus loss function is applied to handle the problem of imbalanced positive and negative samples. Comparing with 12 baseline methods on three public datasets, Diginetica, Tmall, and Yoochoose, the experimental results show that the performance of SDGI is significantly improved compared to baseline methods, indicating that combining sequence dependencies with global information can effectively improve session recommendation performance.

Key Words: recommendation system; session-based recommendation; graph neural network; sequence dependence

收稿日期: 2024-01-23

扫描二维码阅读全文:

基金项目: 国家自然科学基金项目(62176142,62177031);山东省自然科学基金项目(ZR2021MF099,ZR2022MF334);山东省本科教学改革研究项目(M2021130,M2022245);山东省研究生优质教育教学资源项目(SDYAL2022155);济南市市校融合发展战略工程项目(JNSX2023064)

作者简介: 曹家伟(1999-),女,山东建筑大学计算机科学与技术学院硕士研究生,研究方向为推荐系统;段汶君(1991-),女,博士,山东建筑大学计算机科学与技术学院讲师,研究方向为行为识别、图像分割、推荐系统;孙倩(1983-),女,硕士,山东建筑大学计算机科学与技术学院讲师,研究方向为软件工程与推荐系统;袁卫华(1977-),女,博士,山东建筑大学计算机科学与技术学院副教授、硕士生导师,研究方向为机器学习与推荐系统。本文通讯作者:袁卫华。

0 引言

目前,推荐系统被广泛应用于各种互联网平台以缓解信息过载问题。传统推荐方法,如协同过滤推荐利用用户的身份信息与长期历史交互来推断他们感兴趣的内容,但无法为未登录的用户或那些具有短期交互历史的用户进行推荐[1-3]。基于会话的推荐系统为当前正在进行的会话中的匿名目标用户预测下一次交互行为,如点击、浏览或购买等。传统的会话推荐通过马尔可夫链对用户与项目的交互进行建模,但其时序假设为下一个项目完全基于前一个项目,因此不能捕获长期的序列依赖性[4-5]。为克服传统推荐方法的局限性,基于循环神经网络(Recurrent Neural Network,RNN)的方法被应用于会话推荐中并取得了成功,其主要通过对交互项目的序列关系进行建模来实现推荐,但是忽略了更深层次的项目与项目之间的交互行为[6-7]。近年来,基于图神经网络(Graph Neural Networks,GNN)的方法被应用于基于会话的推荐任务,其可以捕获项目之间更复杂的关系,克服以往方法的局限性,逐渐成为会话推荐的主流方法[8-9]。

1 相关研究

会话推荐早期使用马尔可夫链对用户与项目的交互进行建模。例如,Rendle 等[4]提出 FPMC(Factorized Personalized Markov Chain)模型,通过矩阵分解与一阶马尔可夫链组合的方法来捕获序列模式与长期用户偏好;Wang 等[10]提出的 HRM(Hierarchical Representation Model)对 FPMC 进行了改进,本质上是在 FPMC 中加入了非线性转换。该类方法不能捕获长期序列依赖性。

近年来,深度学习技术蓬勃发展。基于 RNN 的会话推荐通过对给定交互的顺序依赖关系建模来预测下一个可能的交互。例如,Hidasi 等[6]提出的 GRU4Rec 模型首先将 RNN 引入会话推荐中;Li 等[7]提出的 NARM(Neural Attentive Session-based Recommendation)模型在基于 RNN 的会话推荐工作中加入注意力机制来改进 GRU4Rec;Tan 等[11]通过数据扩充技术与处理会话数据的时间偏移来增强 RNN 性能;Liu 等[12]提出一个短期注意力优先模型,通过简单的多层感知器网络与注意力机制来捕捉局部和全局的用户兴趣;Wu 等[13]将上下文信息映射到低维真实向量特征中,然后融合到基于 RNN 的会话推荐模型中;Zhang 等[14]将 RNN 与卷积神经网络相结合,采用一种具有项目级注意力机制的门控循环单元(Gated Recurrent Unit,GRU)学习用户的一般兴趣,利用卷积运算学习用户的动态兴趣。基于 RNN 的会话推荐通常只能捕获点式依赖而忽略整体依赖,即几个交互共同影响下一个交互。

基于 GNN 的方法根据会话序列构建会话图,从而进行物品特征的表示学习。与传统方法和基于 RNN 的方法相比,基于 GNN 的推荐方法性能有了很大提高。例如,Wu 等[8]提出的 SR-GNN(Session-based Recommendation with Graph Neural Networks)使用 GNN 学习复杂的顺序转换相互作用,以模拟局部与全局偏好,但其只关注了当前会话,性能容易受到用户行为稀疏性与噪声数据的影响;Wang 等[15]提出的 GCE-GNN(Global Context Enhanced Graph Neural Networks)通过对所有会话的成对项目转换建模来学习全局级项目嵌入,并采用反向位置编码与软注意机制的聚合方法来学习会话序列中每个项目的贡献;Xu 等[16]提出的 GC-SAN(Graph Contextualized Self-attention Network)模型将自注意机制与 GNN 相结合,通过图信息聚合来捕获项目的依赖关系;Xia 等[17]提出的 COTREC 模型通过自监督学习图协同训练对稀疏序列进行增强,从而提升推荐模型的性能;Peng 等[18]提出的 GC-HGNN(Global-Context supported Hypergraph Neural Network)模型采用超图卷积神经网络与图注意网络来获取全局上下文信息与局部信息,并采用注意力机制学习会话序列的最终表示;Wang 等[19]构建了具有时间间隔的会话图学习项目间复杂的交互信息,以提高会话推荐的准确性;Dong 等[20]使用感知器分别对无向图与有向图进行建模,以获得会话中的高阶与低阶项目表示,并设计了一个位置层来计算位置信息。基于 GNN 的会话推荐方法通过挖掘被推荐项目与相应序列上下文之间的复杂关系来提供较为准确的推荐结果,但该类方法通常只考虑会话中当前节点与其邻居节点的信息传递,而忽略了节点在会话中的序列依赖关系。

虽然现有方法能够在一定程度上提升会话推荐的性能,但仍存在以下问题:①在大多数情况下,会话中的项目具有很强的序列依赖性,可能包含重要的用户偏好信息,如果不考虑其序列相关性则可能会导致重要信息丢失,从而影响后续推荐性能;②在对用户偏好进行建模时,大多数基于 GNN 的方法只考虑当前会话并堆叠多层生成嵌入表达,容易导致偏差与过拟合等问题;③在模型训练中存在正负样本不平衡的问题,可能会导致推荐性能下降。为解决上述问题,本文提出一种结合序列依赖与全局信息的会话推荐方法 SDGI(Sequence Dependency and Global Information),主要贡献为:①针对现有推荐方法容易忽略序列依赖关系的问题,采用卷积时间感知的 GRU 网络学习相邻项目的序列依赖关系;②为学习全局级用户偏好信息,提出一种新型轻量级图卷积网络层学习所有会话中的全局级项目嵌入,并使用一种门控网络有效缓解模型堆叠层数过多导致的偏差与过拟合问题,使用图注意网络学习当前会话中的局部级项目嵌入;③为解决模型训练过程中正负样本不平衡的问题,应用焦点损失函数学习不同项目的点击概率,以生成精准推荐结果;④在 3 个真实数据集上验证了 SDGI 的有效性。

2　问题描述

2.1　问题定义

假设$\mathcal{V}=\{v_1,v_2,\cdots,v_N\}$为项目集合,$N$为项目总数,$s=[v_{s,1},v_{s,2},\cdots,v_{s,n}]$为匿名会话,$n$为当前会话中的项目数,$v_{s,k}\in\mathcal{V}(1\leqslant k\leqslant n)$为会话$s$内用户交互的项目。对于点击序列$s_k=\left[v_{s,1},v_{s,2},\cdots,v_{s,k}\right](1\leqslant k<N)$,基于会话推荐的目标为根据当前会话推荐用户最有可能点击的前m个项目。

2.2　图模型构造

2.2.1　全局图构造

会话图只能提取当前会话中的项目转换信息。由于利用其他会话的项目转换帮助当前会话建模用户偏好非常重要,本文构造全局图来提取所有会话中项目的全局信息,并将该信息用于学习所有会话中的项目嵌入[15]。

设$\mathcal{G}_g=(\mathcal{V}_g,\mathcal{E}_g)$为全局图,其中$\mathcal{V}_g$为节点集;$\mathcal{E}_g$为所有会话序列中相邻项构成的边,$\mathcal{E}_g=\{e_{ij}^g|(v_i,v_j)|v_{s,i}\in V,v_j\in N_{v_i}^{\varepsilon}\}$;邻域为集合$N_{v_i}^{\varepsilon}$,其中$\varepsilon$为项目的邻域范围。当邻域集内相邻项目的距离$d\leqslant\varepsilon$时,设邻域集是无向的,那么图$\mathcal{G}_g$即为一个无向的加权图。

2.2.2　局部图构造

对于每个会话序列$s=[v_{s,1},v_{s,2},\cdots,v_{s,n}]$,构造一个加权有向图$\mathcal{G}_l=(\mathcal{V}_l,\mathcal{E}_l)$来模拟当前会话中相邻项的模式,其中$\mathcal{V}_s$与$\mathcal{E}_s$分别为节点集与边集;$(v_i,v_j)\in\mathcal{E}_l$表示节点$v_i$与$v_j$之间存在相邻边。边集中包含4种类型的边,分别为$e_{in}$、$e_{out}$、$e_{in-out}$与$e_{self}$,其中$e_{in}$为输入边,存在从$v_i$到$v_j$的过渡;$e_{out}$为输出边;$e_{in-out}$为输入—输出边,表示存在从$v_i$到$v_j$或从$v_j$到$v_i$的边;$e_{self}$表示在项目本身中有一个循环转换。这些边可以帮助模型更容易地在会话级别捕获项目之间的关系。在嵌入层创建一个初始嵌入矩阵$W_0\in\mathbb{R}^{|V|*d}$,将每个节点v映射到向量$\boldsymbol{h}_{v_i}$中,其中d为节点的嵌入维数。

3　SDGI方法

SDGI方法整体架构如图1所示。会话序列采用卷积时间感知的GRU网络学习相邻项目的序列依赖关系,使用注意力机制增强模型对局部序列依赖关系的表示能力。构建全局图与局部图,以提取所有对话中项目的全局与局部信息。对于全局级项目嵌入,采用轻量级图卷积方法,并结合门控网络,以有效整合与利用信息。对于当前会话中的局部级项目嵌入,使用图注意网络进行学习。通过Sum-pooling方法,将项目的全局与局部上下文信息表示融合作为项目的全局信息关系表示,并通过注意力机制处理融合后的特征。在模型的预测部分使用可调节的权重对预测结果进行合并,应用焦点损失函数学习不同项目的点击概率,将其与候选项目进行匹配,为下一次点击生成推荐列表。

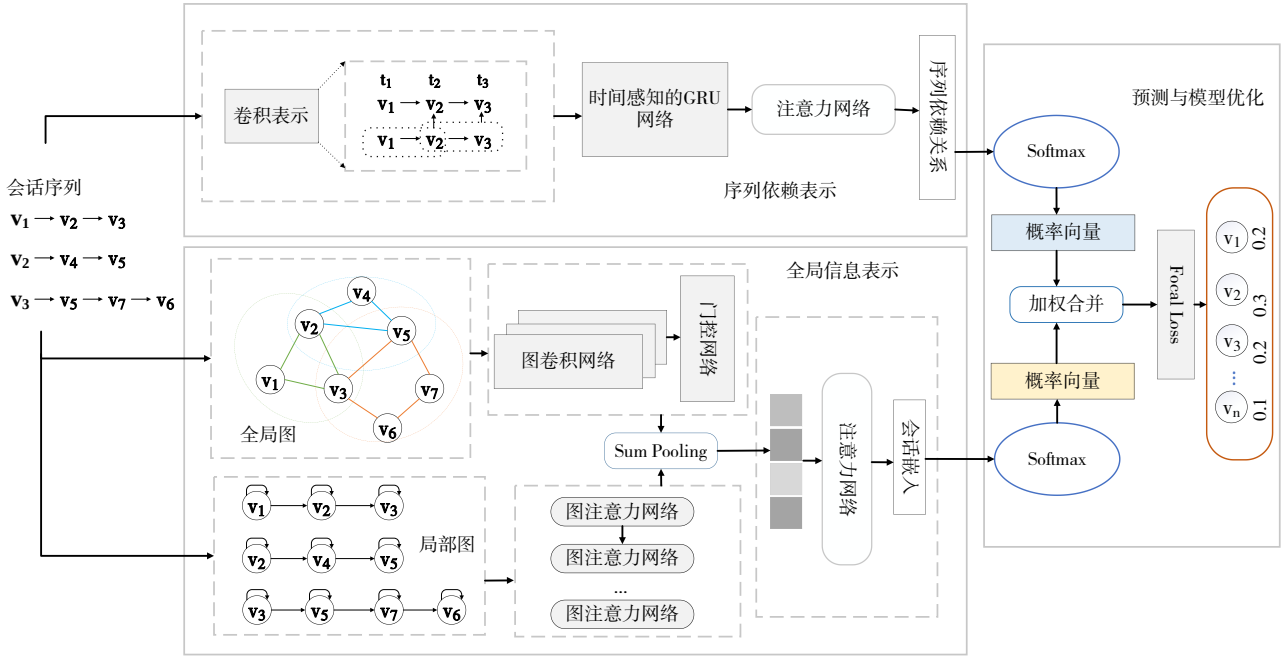

Fig. 1　Framework of SDGI method

图1　SDGI方法整体架构

3.1　序列依赖表示

3.1.1　具有序列模式的卷积表示

同一会话中用户的购物行为通常具有时序性。本文利用卷积时间感知的GRU网络捕获会话序列中项目的序列依赖关系。利用不同大小的卷积核学习局部特征,卷积核的大小$k\in\{k_1,k_2,\cdots,k_m\}$。对于一个在时间$t_i$的项目$h_{v_i}^{t_i}$,与前面$k-1$个项目一起生成一个新的卷积表示$u_k^{t_i}$。最初的$k-1$项使用零向量表示为:

$$u_k^{t_i} = f(h_{v_i}^{t_i}, h_{v_{i-1}}^{t_{i-1}}, \cdots, h_{v_{i-k+2}}^{t_{i-k+2}}, 0, \cdots, 0) \tag{1}$$

式中：f表示卷积操作。

对多滤波器卷积表示进行平均池化，得到项目$h_{v_i}^{t_i}$的最终卷积向量为$f_{v_i}^{t_i}$。定义为：

$$f_{v_i}^{t_i} = \text{avg}\{u_{k_1}^{t_i}, u_{k_2}^{t_i}, \cdots, u_{k_m}^{t_i}\} \tag{2}$$

所有项目的卷积向量可以表示为$f = \{f_{v_1}^{t_1}, f_{v_2}^{t_2}, \cdots, f_{v_n}^{t_n}\}$，将项目的卷积表示输入到时间感知的GRU网络中，以捕捉局部的序列依赖关系。

3.1.2 时间感知的GRU网络

为了更好地捕捉局部的序列依赖关系，进一步考虑连续的时间，本文设计了一种时间感知的GRU网络[21]。表示为：

$$z_{v_i}^{t_i} = \sigma(W_z^1 f_{v_i}^{t_i} + W_z^2 c_{v_i}^{t_i - 1} + b_z) \tag{3}$$

$$r_{v_i}^{t_i} = \sigma(W_r^1 f_{v_i}^{t_i} + W_r^2 c_{v_i}^{t_i - 1} + b_r) \tag{4}$$

式中：σ为sigmoid函数；W_z^1、W_z^2、W_r^1、$W_r^2 \in \mathbb{R}^{d \times d}$与$b_z$、$b_r \in \mathbb{R}^d$为可训练的参数；$\boldsymbol{f}_{v_i}^{t_i} \in \mathbb{R}^d$为项目$v_i$的卷积向量表示；$z_{v_i}^{t_i}$与$r_{v_i}^{t_i} \in \mathbb{R}^d$分别为更新门与重置门。

$$\tilde{c}_{v_i}^{t_i} = \tanh(W_c^1(r_{v_i}^{t_i} \odot c_{v_i}^{t_i - 1}) + W_c^2 f_{v_i}^{t_i} + b_c) \tag{5}$$

$$c_{v_i}^{t_i} = \left(1 - z_{v_i}^{t_i}\right) \odot c_{v_i}^{t_i - 1} + z_{v_i}^{t_i} \odot \tilde{c}_{v_i}^{t_i} \tag{6}$$

式中：W_c^1、$W_c^2 \in \mathbb{R}^{d \times d}$与$b_c \in \mathbb{R}^d$为可训练的参数；$\odot$表示元素乘积；$\tilde{c}_{v_i}^{t_i}$表示临时隐藏状态；$c_{v_i}^{t_i} \in \mathbb{R}^d$表示当前时间步的最终隐藏状态。

用户在会话中发生的顺序行为之间的序列依赖关系被很好地编码为隐藏状态。本文进一步使用注意力机制增强模型对序列依赖关系的表示能力。注意力权重向量的计算公式为：

$$\alpha_i = \frac{\exp(e_i)}{\sum_{k=1}^{n} \exp(e_k)} \tag{7}$$

$$e_i = q_1^T \tanh(W_1 c_{v_i}^{t_i} + b_1) \tag{8}$$

式中：$W_1 \in \mathbb{R}^{d \times d}$、$q_1 \in \mathbb{R}^d$ 与$b_1 \in \mathbb{R}^d$为可学习的参数；$c_{v_i}^{t_i}$为项目v_i在时间t_i的隐藏状态。

利用注意力权重对隐藏状态进行加权求和，得到加权表示序列依赖关系的S_p。计算公式为：

$$S_p = \sum_{i=1}^{n} \alpha_i c_{v_i}^{t_i} \tag{9}$$

3.2 全局信息表示

3.2.1 全局级项目嵌入学习

一个项目可能涉及多个会话，从中可以获得有用的项目转换信息，以有效帮助当前预测。为了区分不同邻居对当前项目的重要性以及同一类型邻居的重要性，本文选择轻量级图卷积网络，采用简单的加权与聚合器[22]。计算公式为：

$$h_{v_i}^{g,(k)} = \sum_{v_j \in \mathcal{N}_{v_i}^g} \frac{1}{\sqrt{|\mathcal{N}_{v_i}^g|}\sqrt{|\mathcal{N}_{v_j}^g|}} h_{\mathcal{N}_{v_i}^g}^{(k-1)} \tag{10}$$

式中：$h_{v_i}^{g,(k)}$为项目v_i在k层传播之后的嵌入；$\mathcal{N}_{v_i}^g$为项目$v_{s,i}$的邻居集合；$\mathcal{N}_{v_j}^g$为项目v_j的邻居集合；$\frac{1}{\sqrt{|\mathcal{N}_{v_i}^g|}\sqrt{|\mathcal{N}_{v_j}^g|}}$为对称归一化项，可以避免嵌入规模随着图卷积运算而增大[23]。

为了缓解模型学习全局级项目嵌入时堆叠层数过多导致的偏差与过拟合问题，采用门控网络有效整合与利用信息，以获取最终的全局级项目嵌入$h_{v_i}^g \in \mathbb{R}^d$。计算公式为：

$$g_{v_i} = \sigma(W_2 h_{v_i} + W_3 h_{v_i}^{g,(k)}) \tag{11}$$

$$h_{v_i}^g = (1 - g_{v_i}) \odot h_{v_i} + g_{v_i} \odot h_{v_i}^{g,(k)} \tag{12}$$

式中：$g_{v_i} \in \mathbb{R}^d$为门控融合向量；$h_{v_i} \in \mathbb{R}^d$为初始项目嵌入；$W_2, W_3 \in \mathbb{R}^{d \times d}$为可训练的参数；$\sigma$为sigmoid激活函数。

3.2.2 局部级项目嵌入学习

不同邻居对当前项目的重要性不同。为学习项目的成对表示，采用图注意力网络计算不同邻居节点对当前节点的影响，为其分配不同的注意权重以区分重要性，并通过线性组合获得每个节点的输出特征。计算公式为：

$$h_{v_i}^s = \sum_{v_j \in \mathcal{N}_{v_i}^s} \varphi_{ij} h_{v_j} \tag{13}$$

式中：$\varphi_{i,j}$为控制相邻项目重要性权重的系数。由于会话图中项目的邻居对自身的重要性不同，利用注意机制来学习不同节点之间的权重。注意系数可以通过元素乘积与非线性变换来计算。公式为：

$$\varphi_{i,j} = \text{LeakyReLU}(a_{r_{ij}}^\top (h_{v_i} \odot h_{v_j})) \tag{14}$$

式中：$\varphi_{i,j}$表示节点v_j的特征对节点v_i的重要性；LeakyReLU为激活函数；r_{ij}表示v_i与v_j之间的关系。会话图中有4种类型的边关系，针对不同关系训练4个权重向量，分别为a_{in}、a_{out}、a_{in-out}与a_{self}，$a_* \in \mathbb{R}^d$为权重向量。由于图中不是每两个节点均连接，仅计算节点$j \in \mathcal{N}_{v_i}^S$的$\varphi_{i,j}$，其中$\mathcal{N}_{v_i}^S$为$v_i$的一阶邻居。为使不同节点之间的系数具有可比性，通过Softmax函数对注意力权重进行归一化。公式为：

$$\varphi_{i,j} = \frac{\exp(\varphi_{i,j})}{\sum_{v_k \in \mathcal{N}_{v_i}^s} \exp(\varphi_{i,k})} \tag{15}$$

如此便得到局部级项目嵌入$h_{v_i}^s \in \mathbb{R}^d$。该项目由当前会话中邻居与项目本身的特征聚合而成。

3.2.3 信息融合

在获得物品的全局表示与局部表示后，利用Sum-pooling操作融合两个层次的信息，表示为$h_{v_i}^*$：

$$h_{v_i}^* = \text{SumPooling}(h_{v_i}^g, h_{v_i}^s) \tag{16}$$

基于式(16)可以得到会话中所涉及项目的嵌入，即$H = \{h_{v_1}^*, h_{v_2}^*, .., h_{v_l}^*\}$。会话表示与项目信息密切相关，为了得到会话表示，对会话序列中项目的信息进行平均：

$$s^* = \frac{1}{l}\sum_{i=1}^{l} h_{v_i}^* \tag{17}$$

采用注意力机制更好地捕捉会话中不同项目之间的重要性和相关性。公式为：

$$\beta_i = q_2^T \sigma(W_4 s^* + b_2) \tag{18}$$

式中：$W_4 \in \mathbb{R}^{d\times d}$、$q_2 \in \mathbb{R}^d$ 与 $b_2 \in \mathbb{R}^d$ 为可学习的参数。

$$S_g = \sum_{i=1}^{l} \beta_i h_{v_i}^* \tag{19}$$

如此以来便得到全局信息表示 S_g，其融合了全局级与局部级信息。

3.3　预测层与模型优化

将序列依赖关系表示 S_p 与全局信息表示 S_g 分别作为用户偏好计算每个候选项目的分数，对其进行排序获得推荐列表。通过预测层可以得到两个概率向量 $\hat{y}_g$ 与 $\hat{y}_p$。计算公式分别为：

$$\hat{y}_g = softmax(S_g{}^T h_{v_i}) \tag{20}$$

$$\hat{y}_p = softmax(S_p{}^T h_{v_i}) \tag{21}$$

使用可调权重对预测结果进行合并，计算最终预测结果 $\hat{y}$。公式为：

$$\hat{y} = (1-\omega)^* \hat{y}_p + \omega^* \hat{y}_g \tag{22}$$

式中：ω 为全局信息的权重，选择概率最高的 K 个候选项目作为推荐结果。

为解决正负样本不平衡的问题，受到 Lin 等[24]研究启发，应用焦点损失代替传统的交叉熵损失对模型进行优化。实际值 y 与预测结果 $\hat{y}$ 之间的损失函数定义为：

$$\begin{aligned}\mathcal{L}(\hat{y}) = \{&-\alpha\sum_{i=1}^{n}(1-\hat{y}_i)^{\gamma}\log\hat{y}_i, \quad y_i = 1\\ &-(1-\alpha)\hat{y}_i^{\gamma}\log(1-\hat{y}_i), \quad y_i = 0\end{aligned} \tag{23}$$

式中：$\hat{y}_i \in \hat{y}$ 为项目 v_i 在当前会话中作为下一次点击出现的概率；α 为平衡正负样本比例的因子；γ 为解决可区分样本与不可区分样本不平衡问题的因子。

4　实验方法与结果分析

4.1　实验环境

采用 PyTorch 作为模型训练任务的主要框架。使用 Python3.8 进行编译，以确保代码的兼容性和稳定性。处理器为 RTX 3080 Ti(12 GB)，Intel(R) Xeon(R) Silver 4214R CPU。

4.2　数据集及其预处理

使用 Diginetica(https://competitions.codalab.org/competitions/11161)、Tmall(https://tianchi.aliyun.com/dataset/dataDetail? dataId=42)、Yoochoose(http://2015.recsyschallenge.com/challege.html)3 个公共数据集验证模型性能。其中，Diginetica 数据集发布于 2016 年的 CIKM 杯，由典型的交易数据组成，经常被用于基于会话的推荐任务中。遵循之前的工作，取最近一周的数据作为测试集，使用剩余数据进行训练[8,15]；Tmall 数据集来自于 IJCAI-15 竞赛，包含 Tmall 购物平台上的匿名用户购物数据；Yoochoose 数据集由 Recsys Challenge 中提取，包含用户点击的零售商网站列表。由于 Yoochoose 数据集相当庞大，训练时间成本非常高，遵循之前的工作，使用最近 1/64 的数据进行训练[8,12,25]。

对 3 个数据集进行预处理，过滤掉长度为 1 的所有会话和出现次数少于 5 次的项目[8,16]。利用分裂的方法生成序列及相应的标签来扩充数据，对会话序列 $S=[v_{s,1}, v_{s,2}, \cdots, v_{s,n}]$ 生成一系列序列和标签 $\left(\left[v_{s,1}\right], v_{s,2}\right), ([v_{s,1}, v_{s,2}], v_{s,3}), \cdots, ([v_{s,1}, \cdots, v_{s,n-1}], v_{s,m})$。预处理后的数据集统计信息见表 1。

Table 1　Preprocessed dataset statistical information

表 1　预处理后的数据集统计信息

统计数据	Diginetica	Tmall	Yoochoose1/64
点击	982 961	818 479	557 249
训练会话	719 470	351 268	369 859
测试会话	60 858	25 898	55 898
项目种类	43 097	40 728	16 766
平均长度	5.12	6.69	6.16

4.3　评价指标

为便于与基准模型进行比较，本文采用两个广泛使用的基于排名的度量：P@K 与 MRR@K，其中 K 表示推荐项目的数量。

P@K 被广泛用作预测准确性的衡量标准，表示正确推荐的项目在前 K 个项目中的比例。计算公式为：

$$P@K = \frac{n_{\text{hit}}}{N} \tag{24}$$

式中：N 为测试数据中的项目总数；n_{hit} 为排名列表前 K 个项目中被正确推荐的项目数。

MRR@K 表示在前 K 个推荐项目中正确推荐项目的平均倒数排名。如果排名超过 K，那么倒数排名会被设为 0。MRR 指标考虑了推荐列表的排名顺序，较大的 MRR 值表示正确推荐通常出现在排序列表的前面。其计算公式为：

$$MRR@K = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\text{rank}_i} \tag{25}$$

式中：rank_i 为项目 i 在推荐排名前 K 列表中的排名，如果 i 不在其中，则 $\frac{1}{\text{rank}_i}$ 为 0。

4.4　基线方法

为评估 SDGI 方法的总体推荐性能，选取 12 个基线方法与其进行性能比较。基线方法包括 3 种传统推荐方法、4 种基于 RNN 的推荐方法与 5 种基于 GNN 的方法。具体为：①POP。以项目出现的频繁次数进行推荐；②FPMC[4]。结合矩阵分解和一阶马尔可夫链来捕捉顺序效应和用户偏好，在计算推荐分数时不考虑用户的潜在表示；③Item-KNN[26]。根据当前会话项目与其他项目之间的相似性推荐项目；④GRU4Rec[6]。一个基于 RNN 的模型，使用 GRU 模拟用户序列，利用会话并行小批量训练过程，并采用基于排名的损失函数模拟用户序列；⑤NARM[7]。改进了 GRU4Rec，将注意力机制应用到基于 RNN 的会话推荐方

法中；⑥STAMP[12]。采用基于注意力的多层感知机模型，能够捕捉用户依赖于最后一项的当前兴趣，并将其与长期兴趣相结合以提高性能；⑦CSRM[27]。利用记忆网络研究最近的几个会话，以更准确地预测当前会话的意图；⑧SR-GNN[8]。采用门控GNN层获得项目嵌入，通过捕获该会话的全局偏好和当前兴趣的注意力网来生成用于推荐的会话表示；⑨GC-SAN[16]。结合GNN与多层自注意力网络，通过对局部邻域项目转换进行建模提取长程依赖关系，提高推荐性能；⑩GCE-GNN[15]。使用两级图模型从局部与全局上下文中捕获项目转换关系，并考虑反向位置信息生成基于会话推荐的会话表示；⑪DHCN[28]。基于超图神经网络捕获高阶不配对关系，并集成自监督学习作为改进推荐任务的辅助方法；⑫AGNN-GC[29]。提出一种新型注意力机制，在融合全局级与局部级项目嵌入表示后增强当前会话中项目的特征表示。

4.5 参数设置

为进行公平比较，本文采用各文献中提供的数据预处理方法和参数设置，以获得各基线方法的最佳性能。本文方法使用平均值为0、标准偏差为0.1的高斯分布初始化所有参数，选择初始学习率为0.001的Adam，每3个epoch后衰减0.1以优化参数[15]。在焦点损失函数中，α的取值为0.25，γ的取值为2[24]。

4.6 实验结果与分析

4.6.1 SDGI方法与基线方法性能比较

表2为SDGI方法与基线方法性能比较结果。每列中的最佳结果以粗体突出显示，而基线模型的最佳结果以下划线突出显示。可以看出，SDGI在3个数据集中取得了最佳结果。在传统推荐方法中，POP性能最差，Item-KNN性能最佳，但均不如基于RNN和GNN的方法。在基于RNN的方法中，CSRM拥有两个并行的记忆模块，性能最佳；FPMC表现最差。与基于RNN的方法相比，基于GNN的方法性能进一步提升。SR-GNN将GNN应用于基于会话的推荐，并在最后一项上使用自注意力机制生成会话嵌入；GC-SAN使用多层自注意力网络，总体上可以实现比SR-GNN更好的性能；GCE-GNN整合来自全局上下文和当前兴趣的信息学习项目嵌入，其性能优于GC-SAN，这说明在推荐任务中利用来自其他会话信息的重要性；DHCN将会话序列建模为超图，学习项目之间复杂的高阶关系，取得了优异性能，但是不及GCE-GNN，因其仅侧重于全局特征；AGNN-GC通过引入新型注意力机制对GCE-GNN方法进行改进，其性能超越了以上基于GNN的推荐方法，并在所有基线方法中取得了最佳性能，但是忽略了会话的序列依赖关系对于推荐性能的影响，与本文的研究动机相符。

与基线方法相比，SDGI方法考虑了全局信息的影响以及序列依赖关系在推荐任务中的作用，同时改进了获取全局信息的方法，对损失函数进行了优化。这些综合改进措施共同促使SDGI方法性能得到有效提升。

Table 2 Performance comparison result of SDGI method and baseline methods

表2 SDGI方法与基线方法性能比较结果

方法	Diginetica		Tmall		Yoochoose1/64	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
POP	1.18	0.28	2.00	0.90	6.71	1.65
ItemKNN	35.75	11.57	9.15	3.31	51.60	21.81
FPMC	22.14	6.66	16.06	7.32	45.62	15.01
GRU4Rec	30.79	8.22	10.93	5.89	60.64	22.89
NARM	48.32	16.00	23.30	10.70	68.32	28.63
STAMP	46.62	15.13	26.47	13.36	68.74	29.67
CSRM	50.55	16.38	29.46	13.96	69.85	29.71
SR-GNN	51.26	17.78	27.57	13.72	70.57	30.94
GC-SAN	49.11	16.73	21.80	10.17	70.59	30.25
GCE-GNN	54.22	19.03	33.42	15.42	70.91	30.63
DHCN	53.66	18.51	31.42	15.05	69.87	29.88
AGNN-GC	54.35	19.00	**33.68**	15.54	70.99	30.90
SDGI	**54.45**	**19.05**	33.50	**15.89**	**71.43**	**31.01**

4.6.2 SDGI与模型变体性能比较

为进一步分析各个组件对本文模型性能的影响，设计模型的3种变体：①SDGI-w/o-SP。表示没有使用卷积时间感知的GRU网络学习序列依赖的版本，即没有考虑到序列依赖信息对推荐性能的影响；②SDGI-w/o-G。表示没有使用全局级项目嵌入学习的变体，即没有考虑来自其他会话信息的影响；③SDGI-w/o-L。表示没有使用局部级项目嵌入学习的变体。将这3种变体与SDGI在3个数据集上进行性能比较，结果见表3。可以看出，在这3个数据集上，每个组件对模型性能的贡献是不同的，每列的最佳结果以粗体突出显示。在3个数据集上，SDGI性能最佳。SDGI-w/o-SP由于没有考虑序列依赖关系，其性能与SDGI相比有较大下降，证实了考虑序列依赖关系对于提升模型性能的重要性。SDGI-w/o-SP的性能优于SDGI-w/o-G和SDGI-w/o-L，SDGI-w/o-SP虽然没有考虑序列依赖关系，但其基于GNN建立了全局图与局部图，基于全局信息实现推荐，而SDGI-w/o-G与SDGI-w/o-L没有同时建立全局图与局部图，证实了基于全局信息实现推荐的重要性。

Table 3 Performance comparison between SDGI and different variants

表3 SDGI与模型变体性能比较

方法	Diginetica		Tmall		Yoochoose1/64	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
SDGI-w/o-SP	54.24	19.05	33.26	15.65	71.21	30.18
SDGI-w/o-G	53.75	18.96	32.60	15.19	71.30	30.25
SDGI-w/o-L	53.30	18.51	30.73	14.18	71.22	30.52
SDGI	**54.45**	**19.05**	**33.50**	**15.89**	**71.43**	**31.01**

4.6.3 不同损失函数对模型性能的影响

比较交叉熵损失函数与焦点损失函数对本文模型性能的影响，结果见表4。可以看出，在3个数据集上，使用焦点损失函数的模型性能大多优于使用传统交叉熵损失函数的模型。这是由于传统的交叉熵损失函数在面对不

平衡的类别分布时通常会受到影响,导致模型训练困难。而焦点损失函数能使模型更加关注那些难以分类的样本,从而有效应对正负样本数据不平衡的问题。在会话推荐任务中,用户的兴趣偏好可能使某些物品的点击次数远多于其他物品,导致数据可能会出现不平衡的问题,因此焦点损失函数能够取得更好的性能,但在实际应用中需要根据具体数据集来分析。

Table 4　Impact of different loss functions on model performance

表 4　不同损失函数对模型性能的影响

损失函数	Diginetica		Tmall		Yoochoose1/64	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
Cross Entropy Loss	54.39	**19.06**	33.49	15.54	71.42	30.99
Focal loss	**54.45**	19.05	**33.50**	**15.89**	**71.43**	**31.01**

4.6.4　不同权重 ω 对模型性能的影响

超参数 ω 为公式(22)中用于控制序列依赖表示和全局信息表示在预测层得到的两个概率向量所形成的最终预测分数的比例,是影响模型性能的一个重要超参数。图 2 与图 3 分别为不同 ω 值对 Diginetica 和 Tmall 数据集上模型性能的影响,ω 的值在{0.2,0.4,0.6,0.8,1}中调整。可以看出,在 Diginetica 数据集上,模型性能随着 ω 的增加而逐渐提高,ω 为 0.9 时性能相对较佳。在 Tmall 数据集上,模型性能随着 ω 的增加呈现出与 Diginetica 数据集上类似的趋势,ω 为{0.8,0.9}时性能相对较佳。全局信息表示的预测得分普遍高于序列依赖关系表示的得分。当 ω 为 1 时,模型性能开始下降,因此设置合理的 ω 值平衡两个部分的得分可以帮助模型更好地提升性能。在实际应用中,需要根据具体数据集和任务进行调优,以找到最佳 ω 值。

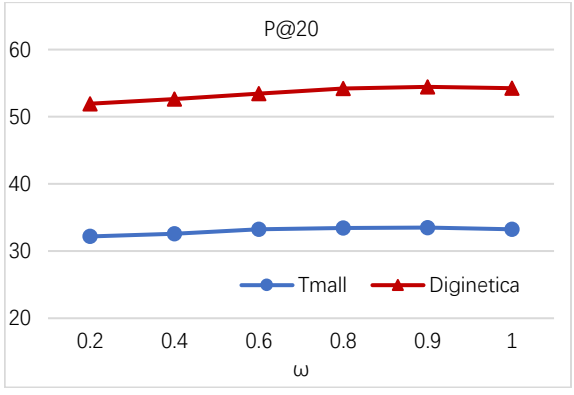

Fig. 2　P@20 for different ω

图 2　不同 ω 值的 P@20

5　结语

本文提出一种结合序列依赖与全局信息的会话推荐方法 SDGI,用于更准确有效地对匿名用户进行推荐。针对当前会话序列,采用卷积时间感知的 GRU 网络学习相邻项目的序列依赖关系,并采用 GNN 构建局部图和全局图学习全局的项目转移信息。所有会话中的全局级项目

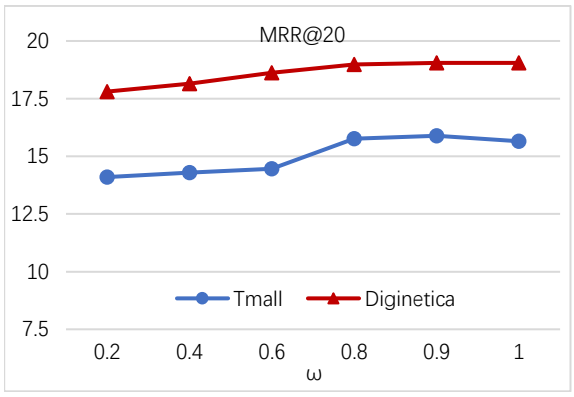

Fig. 3　MRR@20 for different ω

图 3　不同 ω 值的 MRR@20

嵌入采用轻量级图卷积方法,结合门控网络有效整合和利用信息;当前会话中的局部级项目嵌入使用图注意网络模块来学习。在模型的预测部分使用可调节的权重对预测结果进行合并,应用焦点损失函数学习不同项目的点击概率,将其与候选项目进行匹配,为下一次点击生成推荐列表。在 3 个真实数据集上进行了大量实验,结果表明,SDGI 模型的性能优于 12 种基线模型。然而,当交互序列数据非常有限时,SDGI 可能会面临数据稀疏性的问题,从而难以学习到有效的用户偏好及项目特征。未来研究将针对数据稀疏性问题进一步优化模型性能,利用对比学习等方法进行数据增强,以适应更多领域的会话数据。

参考文献:

[1] KOREN Y, BELL R, VOLINSKY C. Matrix factorization techniques for recommender systems[J]. Computer, 2009, 42(8): 30-37.

[2] YANG B, LEI Y, LIU J, et al. Social collaborative filtering by trust[J]. Transactions on Pattern Analysis and Machine Intelligence, 2016, 39(8): 1633-1647.

[3] WANG X, HE X, WANG M, et al. Neural graph collaborative filtering[C]// Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019: 165-174.

[4] RENDLE S, FREUDENTHALER C, SCHMIDT T L. Factorizing personalized markov chains for next-basket recommendation[C]//Proceedings of the 19th International Conference on World Wide Web, 2010: 811-820.

[5] WU X, LIU Q, CHEN E, et al. Personalized next-song recommendation in online karaokes[C]//Proceedings of the 7th ACM Conference on Recommender Systems, 2013: 137-140.

[6] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks. [DB/OL]. https://arxiv. org/pdf/1511. 06939.

[7] LI J, REN P, CHEN Z, et al. Neural attentive session-based recommendation[C]//Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017: 1419-1428.

[8] WU S, TANG Y, ZHU Y, et al. Session-based recommendation with graph neural networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence, 2019: 346-353.

[9] YU F, ZHU Y, LIU Q, et al. TAGNN: target attentive graph neural networks for session-based recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Infor-

mation Retrieval,2020:1921-1924.

[10] WANG P, GUO J, LAN Y, et al. Learning hierarchical representation model for nextbasket recommendation[C]//Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval,2015:403-412.

[11] TAN Y K,XU X, LIU Y. Improved recurrent neural networks for session-based recommendations[C]//Proceedings of the 1st Workshop on Deep Learning for Recommender Systems,2016:17-22.

[12] LIU Q, ZENG Y,MOKHOSI R,et al. STAMP: short-term attention/memory priority model for session-based recommendation[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,2018:1831-1839.

[13] WU T, SUN F, DONG J, et al. Context-aware session recommendation based on recurrent neural networks[J]. Computers and Electrical Engineering,2022,100:107916.

[14] ZHANG J, MA C, MU X, et al. Recurrent convolutional neural network for session-based recommendation [J]. Neurocomputing, 2021, 437: 157-167.

[15] WANG Z, WEI W, CONG G, et al. Global context enhanced graph neural networks for session-based recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval,2020:169-178.

[16] XU C,ZHAO P,LIU Y,et al. Graph contextualized self-attention network for session-based recommendation [C]//Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 2019: 3940-3946.

[17] XIA X,YIN H,YU J,et al. Self-supervised graph co-training for session-based recommendation[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management,2021:2180-2190.

[18] PENG D, ZHANG S. GC-HGNN: a global-context supported hypergraph neural network for enhancing session-based recommendation[J]. Electronic Commerce Research and Applications,2022,52:101129.

[19] WANG H,ZENG Y, CHEN J,et al. Interval-enhanced graph transformer solution for session-based recommendation[J]. Expert Systems with Applications,2023,213:118970.

[20] DONG L,ZHU G,WANG Y,et al. A graph positional attention network for session-based recommendation[J]. Access,2023,11:7564-7573.

[21] BAI T, ZOU L, ZHAO W X,et al. CTRec: a long-short demands evolution model for continuous-time recommendation[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval,2019:675-684.

[22] HE X, DENG K, WANG X, et al. Lightgcn: simplifying and powering graph convolution network for recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval,2020:639-648.

[23] THOMAS N, KIPF, MAX W. Semi-supervised classification with graph convolutional networks[DB/OL]. https://arxiv. org/abs/1609. 02907.

[24] LIN T Y,GOYAL P,GIRSHICK R,et al. Focal loss for dense object detection[C]//Proceedings of the International Conference on Computer Vision,2017:2980-2988.

[25] KOREN Y. Collaborative filtering with temporal dynamics[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,2009:447-456.

[26] SARWAR B,KARYPIS G,KONSTAN J, et al. Item-based collaborative filtering recommendation algorithms[C]//Proceedings of the 10th International Conference on World Wide Web,2001:285-295.

[27] WANG M, REN P, MEI L, et al. A collaborative session-based recommendation approach with parallel memory modules[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval,2019:345-354.

[28] XIA X,YIN H,YU J,et al. Self-supervised hypergraph convolutional networks for session-based recommendation[C]//Proceedings of the AAAI Conference on Artificial Intelligence,2021:4503-4511.

[29] CHEN Y, TANG Y, YUAN Y. Attention-enhanced graph neural networks with global context for session-based recommendation[J]. Access, 2023,11:26237-26246.

(责任编辑:尹晨茹,毛宛婷)