

### 基于全局特征增强的会话推荐算法

靳博文, 王庆梅, 胡承佐, 魏嘉呈

#### 引用本文

靳博文, 王庆梅, 胡承佐, 魏嘉呈. [基于全局特征增强的会话推荐算法](#)[J]. 计算机科学, 2023, 50(11A): 220800205-8.

JIN Bowen, WANG Qingmei, HU Chengzuo, WEI Jiacheng. [Global Feature Enhanced for Session-based Recommendation](#) [J]. Computer Science, 2023, 50(11A): 220800205-8.

---

### 相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

#### Similar articles recommended (Please use Firefox or IE to view the article)

#### [一种融合CNN和Swin Transformer的医学显微图像分割模型](#)

Medical Microscopic Image Segmentation Model Based on CNN Structure and Swin Transformer  
计算机科学, 2023, 50(11A): 230200119-8. <https://doi.org/10.11896/jsjcx.230200119>

#### [基于注意力机制和ConvLSTM的船舶交通流量预测算法](#)

Ship Traffic Flow Prediction Algorithm Based on Attention Mechanism and ConvLSTM  
计算机科学, 2023, 50(11A): 230800067-7. <https://doi.org/10.11896/jsjcx.230800067>

#### [一种基于Meta-learning改进的特征交互算法](#)

Improved Feature Interaction Algorithm Based on Meta-learning  
计算机科学, 2023, 50(11A): 230100087-8. <https://doi.org/10.11896/jsjcx.230100087>

#### [基于图卷积网络和注意力机制的诊断预测](#)

Diagnosis Prediction Based on Graph Convolutional Network and Attention Mechanism  
计算机科学, 2023, 50(11A): 221100232-6. <https://doi.org/10.11896/jsjcx.221100232>

#### [基于知识蒸馏和高效通道注意力的异常检测](#)

Novelty Detection Method Based on Knowledge Distillation and Efficient Channel Attention  
计算机科学, 2023, 50(11A): 220900034-10. <https://doi.org/10.11896/jsjcx.220900034>

# 基于全局特征增强的会话推荐算法

靳博文<sup>1,2</sup> 王庆梅<sup>1,2</sup> 胡承佐<sup>1</sup> 魏嘉呈<sup>1</sup>

1 北京科技大学国家材料服役安全科学中心 北京 100083

2 南方海洋科学与工程广东省实验室(珠海)海洋工程材料与腐蚀控制创新团队 广东 珠海 519080

(m202121149@xs.ustb.edu.cn)

**摘要** 基于会话的推荐系统的研究通常侧重于在使用浅层神经网络的同时通过聚集节点的  $K$  跳邻域来对当前会话用户偏好建模,但是此类方法面临着过平滑的问题。为此,提出了一种基于全局特征增强的会话推荐算法(GFE-SR)。首先,该方法在会话图中利用图神经网络和注意力机制获得会话级项目表示。其次,在全局图的特征传播阶段给每个节点的最近邻域按比例赋予权重来限制过平滑问题,通过全局图进行特征表征融合,获得特征增强的全局级项目表示。然后,通过注意力机制聚合两种项目表示对当前会话的用户偏好进行建模,最终输出候选项目的预测概率。在3个基准数据集上的实验表明,该方法的性能优于现有的最佳方法如 GCE-GNN 等,最高可提升 5.2%,证明了该方法的有效性。

**关键词**: 会话推荐;推荐系统;图卷积网络;注意力机制;过平滑

**中图分类号** TP391

## Global Feature Enhanced for Session-based Recommendation

JIN Bowen<sup>1,2</sup>, WANG Qingmei<sup>1,2</sup>, HU Chengzuo<sup>1</sup> and WEI Jiacheng<sup>1</sup>

1 National Center for Materials Service Safety, University of Science and Technology Beijing, Beijing 100083, China

2 Innovation Group of Marine Engineering Materials and Corrosion Control, Southern Marine Science and Engineering Guangdong Laboratory, Zhuhai, Guangdong 519080, China

**Abstract** Most session-based recommendation system researches commonly model current session user preferences by aggregating  $K$ -hop neighborhoods of nodes while using shallow neural networks. However, such methods face the problem of over-smoothing. This paper proposes global feature enhanced for session-based recommendation (GFE-SR). Firstly, this method utilizes graph neural network and attention mechanism to obtain session-level item representations. Then, in feature propagation stage of the global graph, the nearest neighbors of each node are proportionally weighted to limit over-smoothing to obtain the feature-enhanced global-level item representations. These two item representations are aggregated through an attention mechanism to model current session user preferences. The final output is the probability of the candidate item. Experiments on three public datasets show that this method outperforms the state-of-the-art methods such as GCE-GNN, with a maximum improvement up to 5.2%, which proves the effectiveness of the proposed method.

**Keywords** Session-based recommendation, Recommendation system, Graph convolutional network, Attention mechanism, Over-smoothing

## 1 引言

推荐系统通过向用户推荐有用的内容,成功地解决了信息过载的问题,因此在各种在线平台上发挥着至关重要的作用。传统的推荐算法(如协同过滤<sup>[1]</sup>)通常依赖于用户档案以及用户的长期历史交互信息,但当这些信息不可用(例如,未登录的用户)或者信息有限(例如,短期的历史交互)的情况下,传统推荐系统在最近的现实场景中可能表现不佳,诸如 YouTube 和 Tiktok 等移动流媒体面临数据稀疏和冷启动的问题。近年来,基于会话的推荐引起了广泛关注。基于会话的推荐通过给定的按时间排序的匿名行为序列来预测下一个

可能交互(例如,点击或者购买)的项目。

早期的基于会话的推荐主要分为两类:基于相似度的推荐和基于链的推荐。前者偏重于当前会话中项目的共现信息,而忽略了顺序行为模式;后者主要推断用户在所有项目上选择的所有可能的序列,这对于现实世界中项目数量巨大的场景来说可能会遇到棘手的计算问题。最近许多基于深度学习的推荐算法被提出,它们通过成对的项目转换信息来对给定会话的用户偏好建模<sup>[2-7]</sup>。目前这些方法都取得了令人鼓舞的成果,但是仍然面临以下问题。首先,其中的一些方法通过使用循环神经网络(如 GRU4REC<sup>[2]</sup>, NAMR<sup>[4]</sup>)和记忆力网络(如 STAMP<sup>[8]</sup>)的方法,按时间顺序提取会话中成对

基金项目:南方海洋科学与工程广东省实验室(珠海)创新团队建设项目(311020012)

This work was supported by the Innovation Group Project of Southern Marine Science and Engineering Guangdong Laboratory (Zhuhai) (311020012).

通信作者:王庆梅(qmwang@ustb.edu.cn)

项目的转换信息来判断匿名用户的偏好,但是一个会话中可能包含多个用户的选择,甚至会出现噪声,因而这些方法无法对嵌入过程中项目转换的内在复杂顺序进行建模,从而无法生成所有项目之间的正确依赖关系。其次,其他基于神经网络的方法<sup>[9-11]</sup>加入注意力机制,如 SR-GNN<sup>[10]</sup>,它们通过基于会话中每一项和最后一项之间的成对项目转换,计算相对重要性来学习整个会话的表示,但是它们的性能在很大程度上都依赖于最后一项与当前会话用户偏好的相关性。

除此之外,以往的对基于会话推荐的研究几乎都只是基于当前会话来对用户偏好建模,而忽略了其他会话中有用的项目转换模式。如 CRM<sup>[6]</sup>将会话视为最小粒度,并度量当前会话和最新会话之间的相似性来提取协同关系。但是,这会将其其他会话的相关信息和不相关信息都编码到当前会话的表示当中,从而降低模型的性能<sup>[7]</sup>。GCE-GNN<sup>[12]</sup>在构建全局图中通过图卷积网络聚合邻居信息获得项目表示,这样模型不可避免地会出现过平滑(over-smoothing)的问题<sup>[13]</sup>,从而在一定程度上降低了模型性能。

为了解决上述问题,本文提出了一种新的方法,即基于全局特征增强的会话推荐算法(Global Feature Enhanced for Session-based Recommendation, GFE-SR)。总体来说,本文主要贡献如下:

(1)提出了一种基于全局特征增强的会话推荐算法,通过有效地利用来自两个层次的图模型(即会话图和全局图)的成对项目转换信息来提高对当前会话的推荐性能。

(2)在全局图中融合更深层的邻居特征以获得增强的全局特征,并且缓解了过平滑问题。

(3)在3个真实数据集上进行了广泛的实验,实验结果表明本文方法的性能超过了包括最先进方法在内的10种基准方法。

## 2 国内外研究现状

### 2.1 传统会话推荐算法

早期的基于马尔可夫链的方法将当前会话映射到马尔可夫链,根据用户的前一个动作推断下一个可能的动作。Rendle等<sup>[4]</sup>提出一种基于矩阵分解和一阶马尔可夫链相结合的混合推荐方法,其既能捕捉会话模式,又能捕捉用户长期偏好。但是它通过忽略用户的潜在表征来适应匿名会话推荐场景。然而,基于马尔可夫链的方法通常侧重于对相邻两个项目之间的顺序转变进行建模。相反,本文提出的模型将顺序的项目转换为图结构数据,以捕捉会话推荐中项目转移模式的内在顺序。

除了基于马尔可夫链的方法,基于协同过滤(Collaborative Filtering, CF)的方法也具有较强竞争力。Item-KNN<sup>[1]</sup>可以通过推荐与当前会话的最后一项最相似的项目来对会话推荐进行扩展。KNN-RNN<sup>[15]</sup>利用基于共现的KNN模型来提取会话表示。Wang等<sup>[6]</sup>提出了一种端到端的神经模型CSRMM,该模型利用融合选通机制来学习组合不同来源的特征。然而,在为当前会话整合其他会话的隐含向量时,此类模型会受到噪声的影响。相反,本文利用其他会话中的项目表示来增强当前会话的项目表示,然后将它们集成到会话表示中,可以缓解噪声的影响。

### 2.2 基于深度学习的会话推荐算法

随着神经网络的快速发展,越来越多的研究开始对序列

数据在神经网络中建模,并应用于会话推荐场景。Hidasi等<sup>[2]</sup>提出的GRU4REC,是第一个将循环神经网络(RNN)应用于会话推荐,它采用多层门控循环单元(Gated Recurrent Unit, GRU)对项目交互会话进行建模。之后, Tan等<sup>[16]</sup>通过引入数据增强扩展了GRU4REC的方法<sup>[2]</sup>。Li等<sup>[4]</sup>提出了将注意力机制结合到堆栈GRU编码器中的NARM,以捕获更具代表性的会话推荐项目转换信息。Liu等<sup>[8]</sup>提出了一种基于注意力的短期记忆网络(STAMP)。NARM<sup>[4]</sup>和STAMP都通过使用注意力机制来强调最后一次点击的重要性。受Transformer<sup>[17]</sup>的启发, SASRec<sup>[3]</sup>通过堆叠多个层捕获项目之间的相关性。然而,基于RNN的方法侧重于对相邻项目<sup>[18]</sup>的顺序转换进行建模,以通过给定会话的时间顺序推断用户偏好,因此无法对复杂的项目转换模式(例如,非相邻项目之间的转换)。

最近,一些研究在当前会话构建的图上使用基于图神经网络(Graph Neural Networks, GNN)的模型来学习会话推荐的项目嵌入<sup>[19-23]</sup>。Wu等<sup>[10]</sup>提出的SR-GNN利用门控神经网络来学习会话图上的项目表示,然后通过将每个学习到的项目表示与注意力机制相结合来获得一个有代表性的会话表示,根据每个项目的相似度计算到最后一个会话。基于GNN的会话推荐流程如图1所示。随着SR-GNN的成功,会话推荐也产生了许多变体,例如CSI-GNN<sup>[11]</sup>和FGNN<sup>[24]</sup>通过将目标节点邻居的嵌入与多头注意力聚合来学习每个项目的表示,并通过将每个学习的嵌入与每次与会话的相关性重复组合来生成最终的会话表示。然而,所有这些方法都对当前会话上的项目转换信息进行建模。Wang等<sup>[12]</sup>提出的GCE-GNN考虑到了所有会话中项目的转换信息,但是不可避免地承受着过平滑的问题。相比之下,本文提出的模型在继承了之前工作优点的基础上以更加细致的方式对当前会话的用户建模,并且缓解了过平滑的问题,提升了模型性能。

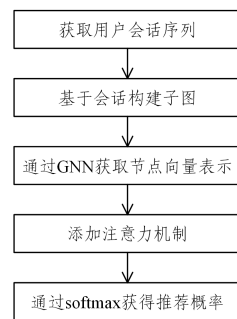


图1 基于GNN的会话推荐流程

Fig. 1 Workflow of the session-based recommendation based on GNN

## 3 问题描述

### 3.1 符号表示

在基于会话的推荐中,用  $V = \{v_1, v_2, \dots, v_m\}$  表示所有的项目,用  $S = \{v_i^1, v_i^2, \dots, v_i^l\}$  表示每个匿名会话,  $S$  为按时间顺序排列的一系列交互(例如用户点击项目)。其中  $v_i^l$  表示项目  $v_i$  在会话  $S$  中产生交互,  $l$  表示会话  $S$  的长度。对于一个会话  $S$ , 基于会话的推荐问题的就是从  $V$  中推荐最有可能被当前会话  $S$  的用户点击的  $\text{top-K}$  ( $1 \leq K \leq |V|$ ) 项目。

此外,本文把  $V$  中的每个项目在  $t$  时间步上编码到一个统一的嵌入空间,即  $\mathbf{h}_i^t \in \mathbb{R}^d$  ( $d$  表示项目的向量表示的维度),那么初始化嵌入的表示就是  $\mathbf{h}_i^0 \in \mathbb{R}^{|V|}$ ,本文使用基于独热

编码(one-hot)的嵌入方法,通过可训练的矩阵  $\mathbf{W}_0 \in \mathbb{R}^{d \times |V|}$  将每个项目转换为  $d$  维的隐含向量  $\mathbf{h}_{v_i}$ 。表 1 定义了文中常使用的符号。

表 1 符号定义表

Table 1 Notations

符号	含义	符号	含义
$V$	项目集合	$T$	全局图中加入自环的标准化邻接矩阵
$S$	某个匿名会话	$A$	全局图中每个节点之间的关系组成的邻接矩阵
$l$	会话 $S$ 的长度	$\tilde{A}$	加入自环的邻接矩阵矩阵
$\mathcal{G}_s$	会话图	$D$	对角矩阵
$\mathcal{G}_g$	全局图	$\tilde{D}$	$\tilde{A}$ 的度矩阵
$e_{ij}^g$	全局图中节点 $i$ 和节点 $j$ 的边	$\beta$	控制传播矩阵信息和单位阵比例超参数
$\mathcal{N}_\epsilon(v_i)$	节点 $i$ 的 $\epsilon$ -邻域	$\tilde{S}$	更新后的传播矩阵
$\mathbf{h}_{v_i}$	节点 $i$ 的向量表示	$K$	卷积层数
$\mathbf{a}_{r_{ij}}$	节点 $i$ 和节点 $j$ 之间关系的权重向量	$\gamma$	控制传播矩阵和项目自身之间比例的超参数
$\theta_{ij}$	会话图节点 $j$ 对节点 $i$ 的注意力系数	$\mathbf{h}_{v_i}^s$	最终的项目隐含向量表示
$\alpha_{ij}$	归一化后的注意力系数	$\mathbf{p}_i$	第 $i$ 个项目的位向量
$\mathbf{N}_{v_i}^1$	节点 $i$ 的一阶邻居	$z_i$	位向量与会话中的项目表示整合后的位置信息
$\mathbf{h}_i^s$	会话级项目表示	$s'$	整合会话中所有项目表示后的会话信息
$x_i$	节点 $i$ 在 GCN 中的特征表示	$\omega_i$	会话表示中的注意力系数
$X$	每个节点特征叠加的矩阵	$\mathbf{h}_s$	分配注意力系数的会话表示
$\mathbf{h}_v^{g,(k)}$	第 $k$ 层的全局级项目表示	$\hat{\Delta}_{v_i}$	项目 $v_i$ 在当前会话的下一时刻被点击的概率
$\tau_{ij}$	节点 $i$ 与节点 $j$ 之间的边的权重	$\mathbf{y}_i$	下一时刻真实点击项目的向量表示

### 3.2 会话图模型

基于会话的图旨在通过对当前会话中成对相邻项的顺序模式进行建模来学习会话级项目的表示。受 SR-GNN<sup>[10]</sup> 的启发,把每个会话序列转换为通过 GNN 图神经网络学习当前会话中项目的表示的会话图。其定义如下:对于给定的会话  $S = \{v_1^s, v_2^s, \dots, v_l^s\}$ ,令  $\mathcal{G}_s = (\mathcal{V}_s, \mathcal{E}_s)$  为其对应的会话图,其中  $\mathcal{V}_s \subseteq V$  是  $S$  中交互项目的集合,  $\mathcal{E}_s = \{e_{ij}^s\}$  表示会话图的边集,其中  $e_{ij}^s$  表示会话  $S$  中两个相邻的项目  $(v_i^s, v_j^s)$  之间的边。

与 SR-GNN<sup>[10]</sup> 工作不同的是,本方法的会话图有 4 种类型的边,分别是  $r_{in}$ ,  $r_{out}$ ,  $r_{in-out}$  和  $r_{self}$ ,这些边取决于项目  $i$  和项目  $j$  之间的关系。对于边  $(v_i^s, v_j^s)$ ,  $r_{in}$  表示仅从  $v_j^s$  到  $v_i^s$  有边,  $r_{out}$  表示仅从  $v_i^s$  到  $v_j^s$  有边,  $r_{in-out}$  表示从  $v_j^s$  到  $v_i^s$  和从  $v_i^s$  到  $v_j^s$  都存在边,  $r_{self}$  表示项目与其自身连接的边。

### 3.3 全局图模型

基于全局的图旨在捕获全局级别的用于学习所有会话中的项目表示的转换信息,也就是说全局图是基于所有会话中

的  $\epsilon$ -邻域的项目集而构建的。全局图的定义如下:设  $\mathcal{G}_g = (\mathcal{V}_g, \mathcal{E}_g)$  为全局图,其中  $\mathcal{V}_g$  表示  $V$  中所有项目的节点集,  $\mathcal{E}_g = \{e_{ij}^g \mid (v_i, v_j) \mid v_i \in V, v_j \in \mathcal{N}_\epsilon(v_i)\}$  表示边集,每一项对应于所有会话中的两项,其中  $\mathcal{N}_\epsilon(v_i)$  表示节点  $v_i$  的  $\epsilon$ -邻域。此外,本文通过生成每个节点  $v_i$  与其相邻边的权重的方法来区分节点邻居的重要性。即对于每个边  $(v_i, v_j)$  ( $v_j \in \mathcal{N}_\epsilon(v_i), v_i \in V$ ),使用它在所有会话中的频率作为对应边的权重。因此,图  $\mathcal{G}_g$  是一个无向加权图。出于对效率的考虑,只保留了图  $\mathcal{G}_g$  上每个项目  $v_i$  权重最高的前  $N$  条边,并且不会动态更新全新图的拓扑结构。

## 4 算法描述

本文提出了一种基于全局特征增强的会话推荐算法。该算法的目标是同时利用会话级和全局级的项目之间的隐含向量转换,对当前会话的用户偏好进行建模,以此来推荐。该算法的总体框架如图 2 所示,它主要包括 4 个部分。

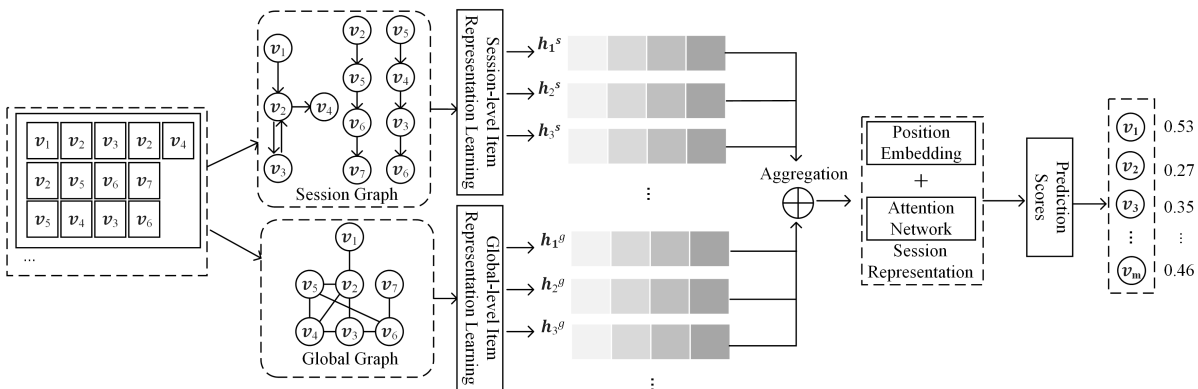


图 2 算法总体框架图

Fig. 2 Workflow of the proposed method

(1)会话级项目表示学习层。本层次在会话图( $\mathcal{G}_s$ )上通过图神经网络和注意力机制来学习当前会话中的会话级项目隐含向量。

(2)全局级项目表示学习层。本层通过叠加更深层的邻居特征获得增强的全局特征,并且缓解了过平滑的问题。

(3)会话表示层。本层次通过聚合在全局学习层和会话

学习层得到的隐含向量,对当前会话的用户偏好进行建模。

(4)生成推荐层。本层次输出候选项目的预测推荐概率。

#### 4.1 会话级项目表示学习层

由于会话图中项目的邻居对自身的重要性不同,利用注意力机制来学习不同节点之间的权重。注意力系数可以通过以下方式来计算:

$$\theta_{ij} = \text{LeakyReLU}(\mathbf{a}_{r_{ij}}^T (\mathbf{h}_{v_i} \odot \mathbf{h}_{v_j})) \quad (1)$$

其中,  $\mathbf{h}_{v_i}$  与  $\mathbf{h}_{v_j}$  分别为节点  $v_i$  和其邻居节点  $v_j$  的向量表示;  $\theta_{ij}$  表示节点  $v_j$  对节点  $v_i$  的权重; 本文选择使用 *LeakyReLU* 作为激活函数;  $\mathbf{a} \in \mathbb{R}^d$  是权重向量,  $r_{ij}$  代表  $v_j$  和  $v_i$  之间的关系, 对于节点之间的不同关系, 本文训练了 4 种权重向量, 即  $\mathbf{a}_{r_{in}}$ ,  $\mathbf{a}_{r_{out}}$ ,  $\mathbf{a}_{r_{in-out}}$  和  $\mathbf{a}_{r_{self}}$ ;  $\odot$  表示元素乘积。

由于并不是图中所有的节点都两两相连, 所以只计算了节点  $j \in \mathbf{N}_{v_i}^s$  以将图结构注入模型, 其中  $\mathbf{N}_{v_i}^s$  是节点  $v_i$  的一阶邻居。为了使不同节点之间的系数具有可比性, 本文通过 *softmax* 函数对注意力系数进行归一化。

$$\alpha_{ij} = \text{softmax}(\theta_{ij}) \quad (2)$$

其中,  $\theta_{ij}$  为邻居节点间的初始注意力系数。因为每个节点的邻居不同, 所以节点对彼此的贡献是不相等的, 因此在式(2)中归一化后的注意力系数  $\alpha_{ij}$  是非对称的。接下来, 通过将注意力系数与其对应的特征进行线性组合来获得每个节点的输出特征向量表示。

$$\mathbf{h}_{v_i}^s = \sum_{v_j \in \mathbf{N}_{v_i}^s} \alpha_{ij} \mathbf{h}_{v_j} \quad (3)$$

会话图中的项目表示由项目本身及其在当前会话中的邻居的特征聚合而成。注意力机制, 降低了噪声对会话级项目表示学习的影响。

#### 4.2 全局级项目表示学习层

学习层使用 Kipf 和 Welling<sup>[25]</sup> 中提到的方法构建图卷积网络 GCN。GCN 学习多层结构中每个节点的特征  $x_i$  的新特征表示, 然后将其输入到线性分类器中。对于第  $k$  图卷积层, 将所有节点的输入节点表示为  $\mathbf{h}_{v_i}^{g,(k-1)}$ , 输出节点表示为  $\mathbf{h}_{v_i}^{g,(k)}$ , 那么初始节点的隐含向量表示就是原始的输入特征:

$$\mathbf{h}_{v_i}^{g,(0)} = \mathbf{X} \quad (4)$$

其中,  $\mathbf{h}_{v_i}^{g,(0)}$  是初始节点在会话图的隐含向量;  $\mathbf{X} \in \mathbb{R}^{d \times d}$  是图中所有特征向量的叠加, 即  $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T$ 。本方法通过每个图卷积层中的特征传播阶段来学习全局级项目的隐含向量, 并且参考 SGC<sup>[26]</sup> 的方法来移除非线性变换。首先在特征传播阶段, 在每层网络的最开始, 需要对每个节点  $v_i$  的特征向量与其邻居的特征向量进行均值化:

$$\bar{\mathbf{h}}_{v_i}^{(k)} \leftarrow \frac{1}{d_i + 1} \mathbf{h}_{v_i}^{(k)} + \sum_{j=1}^n \frac{\tau_{ij}}{\sqrt{(d_i + 1)(d_j + 1)}} \mathbf{h}_{v_j}^{(k-1)} \quad (5)$$

其中,  $\mathbf{h}_{v_i}^{(k)}$  表示节点  $v_i$  在第  $k$  层节点的特征向量,  $\bar{\mathbf{h}}_{v_i}^{(k)}$  表示均值化后得到的特征向量,  $\tau_{ij}$  代表全局图中节点  $v_i$  与节点  $v_j$  之间的边的权重,  $d_i$  和  $d_j$  表示节点  $v_i$  和节点  $v_j$  在对角矩阵中对应的元素。

本方法通过以下矩阵运算的方式对式(5)进行了简化, 首先用  $\mathbf{T}$  表示加入自环的标准化邻接矩阵:

$$\mathbf{T} = \tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \quad (6)$$

其中,  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ ,  $\mathbf{I}$  是单位阵,  $\mathbf{A} \in \mathbb{R}^{d \times d}$  是一个与全局图  $\mathcal{G}_g$  对应的对称的邻接矩阵,  $\tilde{\mathbf{D}} = \mathbf{D} + \mathbf{I}$  是  $\tilde{\mathbf{A}}$  的度矩阵, 对角矩阵  $\mathbf{D} = \text{diag}(d_1, \dots, d_n)$ , 其对角线上的值为邻接矩阵  $\mathbf{A}$  中的行元素

的和, 表示为  $d_i = \sum_j \tau_{ij}$ 。

考虑到项目在一个会话中有重复出现的情况, 并且会话中重复出现的项目可能携带更重要的信息, 因此本方法使用更新后的传播矩阵  $\tilde{\mathbf{T}}$  来减少其他项目可能带来的噪声干扰, 同时提高矩阵中的自注意力信息。

$$\tilde{\mathbf{T}} = \beta(\mathbf{T} + \mathbf{I} - \mathbf{T} \odot \mathbf{I}) + (1 - \beta)\mathbf{I} \quad (7)$$

其中,  $\beta$  是用来控制传播矩阵信息和单位阵信息比例的超参数, 依次来控制传播过程中重复出现的节点在信息吸收中所占比例。

受 Zhu 和 Wang 等<sup>[26-27]</sup> 工作的启发, 本方法对每个节点的最近邻域按比例赋予更大的权重来限制过平滑, 以此来叠加高阶邻居特征, 则最终的特征向量的更新过程表示如下:

$$\mathbf{h}_{v_i}^{g,(k)} = \frac{1}{K} \sum_{k=1}^K ((1 - \gamma) \tilde{\mathbf{T}}^k \mathbf{h}_{v_i}^{g,(k-1)} + \gamma \mathbf{h}_{v_i}^{g,(k-1)}) \mathbf{W}_1 \quad (8)$$

其中,  $k \in K$  代表第  $k$  层卷积,  $\gamma$  为控制传播矩阵和项目自身表示之间比例的超参数,  $\mathbf{W}_1 \in \mathbb{R}^{d \times 2d}$  是权重矩阵。

最后, 对全局级隐含向量表示使用 dropout<sup>[29]</sup> 函数来避免其出现过拟合的问题:

$$\mathbf{h}_{v_i}^{g,(k)} = \text{dropout}(\mathbf{h}_{v_i}^{g,(k)}) \quad (9)$$

在得到每个项目的全局级表示和会话级表示后, 本方法通过求和池化操作 (Sum Pooling) 计算项目的最终表示:

$$\mathbf{h}_{v_i}' = \mathbf{h}_{v_i}^s + \mathbf{h}_{v_i}^{g,(k)} \quad (10)$$

#### 4.3 会话表示层

基于前面学习到的项目表示, 接下来将介绍如何获得会话表示。在本文的方法中, 会话表示是基于会话中涉及到的所有项目构造的。值得注意的是, 不同项目对下一次预测的贡献是不相等的。更直观地说, 在会话中越靠后被点击的项目就越能代表用户当前的偏好, 这表明靠后的项目对推荐的重要性更高。此外, 找到用户的主要目的和过滤当前会话中的噪声也同样重要。因此, 本方法结合了位置信息和会话信息来进行更好的预测。

将会话序列输入到图神经网络中, 可以得到会话序列中涉及的所有项目的表示, 即  $\mathbf{H}^s = [\mathbf{h}_{v_1}', \mathbf{h}_{v_2}', \dots, \mathbf{h}_{v_l}']$ , 本方法还使用了一个可学习的位置嵌入矩阵  $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_l]$ , 其中  $\mathbf{p}_i \in \mathbb{R}^d$  是特定位置  $i$  的位置向量,  $l$  是当前会话序列的长度。位置信息  $z_i$  通过级联和非线性变换的方式进行整合:

$$z_i = \tanh(\mathbf{W}_2 [\mathbf{h}_{v_i}' \parallel \mathbf{p}_{l-i+1}] + \mathbf{b}_1) \quad (11)$$

其中,  $\mathbf{W}_2 \in \mathbb{R}^{d \times 2d}$  和  $\mathbf{b}_1 \in \mathbb{R}^d$  是可学习的参数,  $\parallel$  表示连接操作,  $\tanh$  是激活函数。

由于会话序列的长度不是固定的, 本方法使用了反转的位置嵌入表示。与前向位置信息相比, 当前项目与预测项目之间的距离包含了更多更有效的信息。比如在会话  $\{v_2 \rightarrow v_3 \rightarrow ?\}$  中, 项目  $v_3$  在第二个位置并且对下一步的预测产生很大的影响; 而在会话  $\{v_2 \rightarrow v_3 \rightarrow v_5 \rightarrow v_6 \rightarrow v_8 \rightarrow ?\}$  中, 项目  $v_3$  同样出现在第二个位置, 但是对推荐的重要性就会相对减少。因此, 用相反的位置信息可以更准确地提示每个项目的重要性。

本方法通过计算会话中的所有项目隐含向量表示的平均值获得会话信息  $s'$ :

$$s' = \frac{1}{l} \sum_{i=1}^l \mathbf{h}_{v_i}' \quad (12)$$

接下来, 本方法通过软注意力机制学习相应的权重:

$$\omega_i = \mathbf{q}^T \sigma(\mathbf{W}_3 z_i + \mathbf{W}_4 s' + \mathbf{b}_2) \quad (13)$$

其中,  $\mathbf{W}_3 \in \mathbb{R}^{d \times d}$ ,  $\mathbf{W}_4 \in \mathbb{R}^{d \times d}$ ,  $\mathbf{q} \in \mathbb{R}^d$ ,  $\mathbf{b}_2 \in \mathbb{R}^d$  都是可学习的参数,  $\sigma(\cdot)$  代表 sigmoid 函数。

最后,将项目的隐含向量表示进行线性组合就可以得到会话表示:

$$\mathbf{h}_s = \sum_{i=1}^l \omega_i \mathbf{h}'_{v_i} \quad (14)$$

会话表示  $\mathbf{h}_s$  由当前会话中涉及到的所有项目构成,其中每个项目的贡献不仅取决于会话图中的信息,还取决于会话中的时间顺序。

#### 4.4 生成推荐层

基于得到的会话表示  $\mathbf{h}_s$ , 根据每个候选项目的向量表示  $\mathbf{h}_{v_i}$  和当前会话表示  $\mathbf{h}_s$ , 得到最终的推荐概率。首先使用点积, 然后通过 softmax 函数得到输出  $\hat{y}_i$ :

$$\hat{y}_i = \text{softmax}(\mathbf{h}_s^\top \mathbf{h}_{v_i}) \quad (15)$$

其中,  $\hat{y}_i \in \hat{\mathbf{y}}$  表示项目  $v_i$  在当前会话的下一时刻被点击的概率, 概率越高,  $v_i$  就越可能被推荐。

本方法把损失函数定义为预测结果  $\hat{\mathbf{y}}$  的交叉熵:

$$\mathcal{L}(\hat{\mathbf{y}}) = - \sum_{i=1}^m y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (16)$$

其中,  $y_i$  代表了会话序列下一时刻真实点击项目的独热编码向量 (one-hot embedding)。

在模型的最后, 使用基于时间的反向传播算法来训练。由于在基于会话序列的推荐场景中大多数会话都是相对较短的序列, 为了防止过拟合的出现, 本方法采用了较小的训练次数。

#### 4.5 复杂度分析

本文提出的 GFE-SR 算法的时空复杂度主要来源于全局级项目表示学习层, 该过程的空间复杂度为  $O(|\epsilon| + nd)$ , 其中  $|\epsilon|$  为边数,  $d$  为特征维度,  $n$  为节点个数; 时间复杂度为  $O(K|\epsilon|d + Knd)$ , 其中  $K$  是卷积层数, 对于每个矩阵之间的乘法  $\tilde{\mathbf{T}}^k \mathbf{h}_v^{g_s(k-1)}$  在每层卷积上都需要花费  $|\epsilon|d$  的时间, 因此需要进行  $K$  次这样的计算。  $nd$  是添加特征  $\mathbf{h}_v^{g_s(k-1)}$  的计算成本。

## 5 实验结果与分析

### 5.1 数据集与数据预处理

本文使用了 3 个基准数据集, 分别是 Diginetica<sup>1)</sup>, Tmall<sup>2)</sup> 和 Nowplaying<sup>3)</sup>。Diginetica 数据集是由 2016 年 CIKM Cup 发布的由典型的交易数据组成的数据。Tmall 数据集来自 IJCAI-15 竞赛, 其中包含匿名用户在天猫网购平台上的购物日志。Nowplaying 数据集由 Zangerle 等人整理标注<sup>[30]</sup>, 它记录了用户的音乐收听行为。

遵循 SR-GNN 等工作<sup>[10]</sup>的预处理方法, 在 3 个数据集中过滤了长度为 1 的会话和出现次数少于 5 次的项目。与 STAMP 的工作<sup>[8]</sup>类似, 将最后一周的会话设置为测试数据, 剩余的历史数据用于训练。此外, 对于一个会话  $S = [v_1, v_2, \dots, v_n]$ , 通过拆分输入序列生成对应序列和相应的标签来对上述 3 个数据集进行训练和测试, 即  $([v_1], v_2)$ ,  $([v_1, v_2], v_3)$ ,  $\dots$ ,  $([v_1, v_2, \dots, v_{n-1}], v_n)$ , 其中  $[v_1, v_2, \dots, v_{n-1}]$  是生成的序列,  $v_n$  代表下一时刻点击的项目标签<sup>[12]</sup>。最终经过预处理后的数据集统计数据如表 2 所列。

表 2 实验数据集统计表

Dataset	Diginetica	Tmall	Nowplaying
# click	982961	818479	1367963
# train	719470	351268	825304
# test	60858	25898	89824
# items	43097	40728	60417
avg. len.	5.12	6.69	7.42

### 5.2 评价指标

遵循之前的工作<sup>[8,10,12]</sup>, 选择使用 P@20 和 MRR@20 这两种在会话序列推荐中广泛使用的度量标准作为算法的评价指标。

### 5.3 参数设置

遵循之前的工作<sup>[4,8,10,12]</sup>, 把隐含向量的维度固定为 100, 将所有模型的小批量处理 (mini-batch) 大小设置为 100。为了公平比较, 将保持每个模型的超参数一致。对于 CSRMM<sup>[6]</sup>, 将内存大小设置为与批处理大小一致的 100。

对于本文的方法, 将所有超参数都使用均值为 0、标准差为 0.1 的高斯分布函数进行初始化。本文使用初始学习率为 0.001 的 Adam 优化器对参数进行优化, 并且学习率将在每 3 次迭代后衰减 0.1。另外设置 L2 为  $10^{-5}$ , 并且在 dropout 中设置比率为  $\{0.1, 0.2, \dots, 0.9\}$ , 选择训练集的 10% 的随机子集作为验证集。此外, 还设置了相邻项的数目和相邻项的最大距离  $\epsilon$  分别为 12 和 3。

### 5.4 与相关方法比较

本文将 GFE-SR 与包括传统方法和最先进模型在内的 10 个基准模型进行了比较。实验结果如表 3 所列。

(1) POP 策略是在训练集中推荐 top-K 出现频率最高的项目。

(2) Item-KNN<sup>[1]</sup> 根据当前会话的项目与其他会话的项目之间的相似性来推荐。

(3) FPMC<sup>[14]</sup> 通过结合矩阵分解和马尔可夫链来捕捉序列表效应和用户偏好。

(4) GRU4REC<sup>[2]</sup> 是一种基于 RNN 循环神经网络的模型, 它利用 GRU 对用户序列进行建模。

(5) NARM<sup>[4]</sup> 通过在基于会话的推荐中加入注意力机制改进了 GRU4REC<sup>[2]</sup>。

(6) STAMP<sup>[8]</sup> 通过完全依赖当前会话中最后一项的自注意力来捕捉用户的短期兴趣, 从而使用关注层来取代之前的所有 RNN 编码器。

(7) SR-GNN<sup>[10]</sup> 通过使用门控神经网络来获得项目的表示, 然后像 STAMP<sup>[8]</sup> 一样对最后一项加入自注意力来计算会话级的表示。

(8) CSMR<sup>[6]</sup> 通过使用记忆力网络研究最后的  $m$  个会话, 以更好地预测当前会话用户的意图。

(9) FGNN<sup>[19]</sup> 通过设计一个加权的注意力图层来学习

<sup>1)</sup> <https://competitions.codalab.org/competitions/11161>

<sup>2)</sup> <https://tianchi.aliyun.com/dataset/dataDetail?dataId=42>

<sup>3)</sup> <http://dbis-nowplaying.uibk.ac.at/#nowplaying>

项目表示,并通过图级特征提取器学习下一个项目推荐的会话。

(10)GCE-GNN<sup>[12]</sup>通过会话图和全局图学习项目的会话级和全局级的项目嵌入,并通过软注意力机制聚合两种学习表示以用于推荐。

表 3 实验结果对比

Table 3 Experimental results comparison of different methods

Dataset Methods	Diginetica		Tmall		Nowplaying	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
POP	1.18	0.28	2.00	0.90	2.28	0.86
Item-KNN	35.75	11.57	9.15	3.31	15.94	4.91
FPMC	22.14	6.66	16.06	7.32	7.36	2.82
GRU4Rec	30.79	8.22	10.93	5.89	7.92	4.48
NARM	48.32	16.00	23.30	10.70	18.59	6.93
STAMP	46.62	15.13	26.47	13.36	17.66	6.88
CSR	50.55	16.38	29.46	13.96	18.14	6.42
SR-GNN	51.26	17.78	27.57	13.72	18.87	7.47
FGNN	51.36	18.47	25.24	10.39	18.78	7.15
GCE-GNN	<u>54.22</u>	<u>19.04</u>	<u>33.42</u>	<u>15.42</u>	<u>22.37</u>	<u>8.40</u>
GFE-SR	<b>54.54</b>	<b>19.10</b>	<b>33.61</b>	<b>16.22</b>	<b>22.60</b>	<b>8.63</b>

通过表 3 可以看出,本文提出的方法在 P@20 和 MRR@20 这两个度量方面都取得了最好的效果,证实了本文方法的有效性。另外,在传统方法中,POP 策略因为只推荐 top-K 项目,所以展现的性能最差。Item-KNN 和 FPMC 在传统方法中取得了较好的效果,证明项目之间的相似性对推荐具有一定的有效性,但是它们没有考虑时间序列对推荐性能的影响。

与传统的方法相比,基于神经网络的方法在基于会话推荐的场景中可以展现出更好的性能。但是基于会话的推荐不仅仅是一个序列建模问题,因为用户的偏好可能会在会话中发生改变。

随后的 NARM 将循环神经网络和注意机制相结合,并将序列中的最后一个隐藏状态作为用户的主要偏好,进而提升了推荐的性能。我们还观察到,完全基于注意力的方法 STAMP 在 Tmall 数据集上的性能优于 NARM,因为该方法在会话的最后一个条目上加入了自注意力机制来模拟用户短期兴趣,这一结果证明了在不同的条目上分配不同的注意权重对会话编码有提升作用,但是 STAMP 忽略了项目在会话中的时间顺序。

在所有的基准方法中,基于图神经网络的方法在 Diginetica 和 Nowplaying 数据集的表现更好,SR-GNN 和 FGNN 通过将每个会话序列建模为一个子图,并使用 GNN 对条目进行编码,验证了 GNN 在基于会话的推荐中的有效性,这表明图建模比序列建模(RNN)或注意力建模更适合于基于会话的推荐。GCE-GNN 的性能在 3 个数据集上都优于 SR-GNN 和 FGNN,这表明了全局上下文和局部上下文对基于会话的推荐有更好的增强作用。本文提出的方法在 3 个数据集上优于所有基准,相比于目前最好的 GCE-GNN,本文方法在 Diginetica 上最多高出 0.6%,在 Tmall 上最多高出 5.2%,在 Nowplaying 上最多高出 2.8%。本文在构建全局图中缓解了过平滑问题,模型可以更好地聚合高阶邻居信息,从而获得更好的性能。本文认为 GFE-SR 在 Diginetica 性能上的提升不如在另外两个数据集上的原因是 Diginetica 数据集中的会话平均长度比其他两个数据集短,而 GFE-SR 使用了反转的位置嵌入表示更适合较长的会话。

## 5.5 消融实验

为了证明本文提出的模型中各组成部分是有效的,进行了以下 4 组消融实验,并对实验结果进行分析。

- (1)验证全局级特征和会话级特征的有效性。
- (2)验证对全局特征增强的有效性。
- (3)验证模型具有缓解过平滑的能力。
- (4)验证超参数对模型性能的有效性。

为了验证全局级特征和会话级特征是有效的,本文设计了以下 3 种变体模型:(1)w/o global,没有全局级特征,只有会话级特征;(2)w/o local,没有会话级特征,只有全局级特征;(3)global+local,同时具有会话级特征和全局级特征。实验结果如表 4 所列。

表 4 不同特征模型的性能对比

Table 4 Performance comparison of different feature models

Dataset Methods	Diginetica		Tmall		Nowplaying	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
-w/o global	39.57	13.97	33.50	14.99	21.39	6.89
-w/o local	38.75	13.66	31.13	15.18	21.95	7.15
-global+local	<b>54.54</b>	<b>19.10</b>	<b>33.62</b>	<b>16.22</b>	<b>22.60</b>	<b>8.63</b>

实验结果表明,删除每一种特征都会导致性能下降,这表明全局级特征和会话级特征都有利于提升基于会话的推荐的性能。而且,这两种特征是相辅相成的,可以在不同的方面提升模型性能。

为了验证使用 SSGC 模型增强全局特征是有效的,本文设计了以下 4 种变体模型:(1)global-w/o SSGC+local-w/o SSGC,对全局级特征和会话级特征都不增强;(2)local-SSGC,只增强会话级特征;(3)global-SSGC,只增强全局级特征;(4)global-SSGC+local-SSGC,同时对会话级特征和全局级特征增强。实验对比结果如表 5 所列。

表 5 不同特征增强模型的性能对比

Table 5 Performance comparison of different feature enhancement models

Dataset Methods	Diginetica		Tmall		Nowplaying	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
global-w/o SSGC+local- w/o SSGC	54.22	19.04	33.42	15.42	22.37	8.40
local-SSGC	52.69	18.51	29.84	14.59	21.51	7.22
global-SSGC	<b>54.54</b>	<b>19.10</b>	<b>33.62</b>	<b>16.22</b>	<b>22.60</b>	<b>8.63</b>
global-SSGC+ local-SSGC	50.60	18.14	31.11	15.09	22.38	7.52

结果表明,当仅对全局级特征增强时,模型取得了最好的性能;当仅对会话级特征增强或者同时增强局部级和会话级特征时,模型的性能可能会下降。本文认为由于全局图聚合了所有会话中的项目,所以可以传播更多的信息,随着 SSGC 的特征增强,GFE-SR 可以进一步地聚合高阶邻居信息,从而提升模型的性能。为了验证本文的方法可以缓解过平滑问题,将 GFE-SR 与 GCE-GNN<sup>[12]</sup>在不同层数下展现的性能作对比,并将 MRR@20 作为评价指标,具体结果如表 6 所列。

通过表 6 的结果可观察到,在 Diginetica, Tmall 和 Nowplaying 数据集上,本文的方法在 K=8 时始终获得最佳性能。总体结果表明,GFE-SR 可以比 GCE-GNN 更好地聚合更大的邻域,同时减小过平滑的影响。此外,与 GFE-SR 相比,由于过平滑,GCE-GNN 的性能随着层数超过 32 层而迅速下降。

表6 层数  $K$  对模型的影响Table 6 Effect of layers  $K$  on models

Dataset	Methods	Layers( $K$ )					
		2	4	8	16	32	64
Diginetica	GCE-GNN	18.18	18.49	18.72	18.51	11.14	10.55
	GFE-SR	19.02	19.04	<b>19.10</b>	19.07	19.05	19.01
Tmall	GCE-GNN	15.31	15.36	15.29	15.46	14.61	13.29
	GFE-SR	16.15	16.19	<b>16.22</b>	16.05	15.95	15.80
Nowplaying	GCE-GNN	8.43	8.16	7.86	7.43	6.54	4.41
	GFE-SR	8.09	8.31	<b>8.63</b>	8.41	8.15	8.02

式(7)中的系数  $\beta$  可以控制传播矩阵信息和单位阵信息比例。为了分析  $\beta$  的影响,在 0 到 1 的范围改变  $\beta$  的值,并在图 3(a) — 图 3(c) 分别表示  $\beta$  对 Diginetica, Tmall 和 Nowplaying 3 个数据集的影响。结果表明,适合的  $\beta$  值可以有效提升 GFE-SR 的性能,具体来说,对于 Tmall 数据集和 Now-

playing 数据集,  $\beta$  设置为 0.5 左右时性能最优;而对于 Diginetica 数据集,  $\beta$  设置为 0.7 左右可以获得最优的性能;此外,当  $\beta$  越接近 1 时,就越可能导致性能不佳,本文认为保留一定的自身特征是必要的。

图 3(d) — 图 3(f) 给出了式(8)中  $\gamma$  的变化对模型性能的影响。本文在 0 到 1 的范围改变  $\gamma$  的值。结果表明,适合的  $\gamma$  值可以有效提升 GFE-SR 的性能,具体来说,对于 Tmall 数据集和 Nowplaying 数据集,模型的整体性能会随着  $\gamma$  的增加而增加;对于 Diginetica 数据集,模型的性能会在开始加入  $\gamma$  时,随着  $\gamma$  值的提升而提升,当  $\gamma$  超过 0.2 左右时, GFE-SR 的性能开始下降。这说明不同数据集对项目自身表示的依赖程度不同,适当增加项目自身表示信息对模型性能的提升具有积极作用。

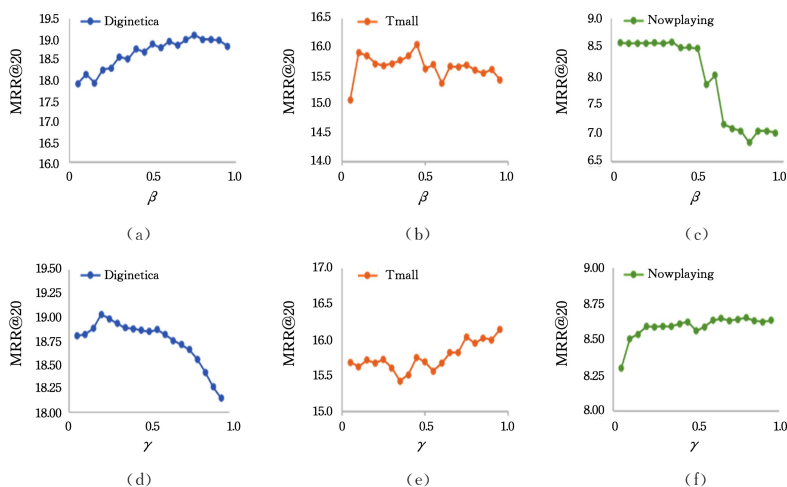


图3 超参数对模型性能的影响

Fig. 3 Effect of hyperparameters on model performance

**结束语** 本文提出了一种基于全局特征增强的会话推荐算法 GFE-SR。该方法一方面在会话图中利用神经网络和注意力机制获得会话级项目表示;另一方面,通过全局图的特征表征融合进行特征增强,获得全局级项目表示,并且通过对节点最近邻按比分配更大的权重缓解过平滑。最后,通过软注意力机制聚合两种项目表示。综合实验表明,该方法在多个基准数据集上显著优于 10 个基准方法。后续将在研究全局特征增强的基础上,寻找更好的项目表示方式,进一步提高基于会话推荐的整体性能。

## 参考文献

- [1] 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.
- [2] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks [J]. arXiv:1511.06939, 2015.
- [3] KANG W C, MCAULEY J. Self-attentive sequential recommendation [C] // 2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018:197-206.
- [4] 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.
- [5] WANG H, LIU G, LIU A, et al. DMRAN: A Hierarchical Fine-Grained Attention-Based Network for Recommendation [C] // IJ-CAI. 2019:3698-3704.
- [6] 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.
- [7] WANG S, HU L, WANG Y, et al. Modeling multi-purpose sessions for next-item recommendations via mixture-channel purpose routing networks [C] // International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence. 2019.
- [8] 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.
- [9] LI Y, TARLOW D, BROCKSCHMIDT M, et al. Gated graph sequence neural networks [J]. arXiv:1511.05493, 2015.
- [10] 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.
- [11] HU C, WANG Q, et al. Complex Structural Information for Ses-

- sion-based Recommendation with Graph Neural Networks[J]. *Computer Engineering*, 2022, 48(5): 82-90.
- [12] 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.
- [13] DING H. *Research on Optimization Algorithm of Graph Convolutional Neural Network*[D]. Beijing: University of Science and Technology, 2022
- [14] RENDLE S, FREUDENTHALER C, SCHMIDT-THIEME L. Factorizing personalized markov chains for next-basket recommendation[C]// *Proceedings of the 19th International Conference on World Wide Web*. 2010: 811-820.
- [15] JANNACH D, LUDEWIG M. When recurrent neural networks meet the neighborhood for session-based recommendation[C]// *Proceedings of the Eleventh ACM Conference on Recommender Systems*. 2017: 306-310.
- [16] 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.
- [17] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. *arXiv:1706.03762*, 2017.
- [18] WANG S, HU L, WANG Y, et al. Sequential recommender systems: challenges, progress and prospects[J]. *arXiv:2001.04830*, 2019.
- [19] WANG C, WANG C, WANG Z, et al. DeepDirect: Learning directions of social ties with edge-based network embedding[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2018, 31(12): 2277-2291.
- [20] WANG Z, YE X, WANG C, et al. Network embedding with completely-imbalanced labels[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2020, 33(11): 3634-3647.
- [21] WANG Z, YE X, WANG C, et al. RSDNE: Exploring relaxed similarity and dissimilarity from completely-imbalanced labels for network embedding[C]// *Proceedings of the AAAI Conference on Artificial Intelligence*. 2018.
- [22] WANG C, WANG C, WANG Z, et al. Edge2vec: Edge-based social network embedding[J]. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2020, 14(4): 1-24.
- [23] WANG Z, WANG J, GUO Y, et al. Zero-shot node classification with decomposed graph prototype network[C]// *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2021: 1769-1779.
- [24] QIU R, LI J, HUANG Z, et al. Rethinking the item order in session-based recommendation with graph neural networks[C]// *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2019: 579-588.
- [25] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks[J]. *arXiv:1609.02907*, 2016.
- [26] WU F, SOUZA A, ZHANG T, et al. Simplifying graph convolutional networks [C] // *International Conference on Machine Learning*. PMLR, 2019: 6861-6871.
- [27] ZHU H, KONIUSZ P. Simple spectral graph convolution[C]// *International Conference on Learning Representations*. 2020.
- [28] WANG Z, WANG C, PEI J, et al. Multiple source detection without knowing the underlying propagation model[C]// *Proceedings of the AAAI Conference on Artificial Intelligence*. 2017.
- [29] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: a simple way to prevent neural networks from overfitting [J]. *The Journal of Machine Learning Research*, 2014, 15(1): 1929-1958.
- [30] ZANGERLE E, PICHL M, GASSLER W, et al. # nowplaying music dataset: Extracting listening behavior from twitter[C]// *Proceedings of the First International Workshop on Internet-scale Multimedia Management*. 2014: 21-26.



**JIN Bowen**, born in 1999, postgraduate. His main research interests include data mining and recommendation.



**WANG Qingmei**, born in 1975, Ph.D, associate researcher. Her main research interests include deep learning and computer vision.