基于兴趣注意力网络的会话推荐算法

崔少国[1*]，独 潇[1]，张宜浩[2]

（1. 重庆师范大学 计算机与信息科学学院，重庆 401331；2. 重庆理工大学 两江人工智能学院，重庆 400054）

摘要 针对现有基于图神经网络的会话推荐算法对用户主要兴趣偏好提取不充分的问题，提出了一种基于兴趣注意力网络的会话推荐算法（Session-Based Recommender Method Based on Interest Attention Network，SR-IAN）。首先，使用图神经网络捕获物品之间的上下文转换关系，得到物品的图嵌入向量；其次，将图嵌入向量输入兴趣注意力网络中，提取用户的主要兴趣偏好；然后通过注意力层对物品的图嵌入向量进行加权区分；最后，通过预测层得到候选物品的点击概率值并对其进行排序。算法模型在 3 个公开数据集 Diginetica、Retailrocket 和 Tmall 上进行了实验验证，相比基准模型在 MRR@20 指标上分别有 0.942%、1.183% 和 2.977% 的提升，同时降低了模型时间复杂度，验证了该方法的有效性和高效性。

关键词 注意力机制；图神经网络；推荐算法；自注意力网络；会话推荐

中图分类号 TP391 **文献标志码** A DOI 10.12178/1001-0548.2022307

Session-Based Recommender Algorithm Based on Interest Attention Network

CUI Shaoguo[1*], DU Xiao[1], and ZHANG Yihao[2]

(1. School of Computer and Information Sciences, Chongqing Normal University, Chongqing 401331, China;
2. School of Liangjiang Artificial Intelligence, Chongqing University of Technology, Chongqing 400054, China)

Abstract Aiming at the problems of insufficient extraction of users' main interest preferences in session-based recommender algorithms based on graph neural networks, a Session-Based Recommender Method Based on Interest Attention Network (SR-IAN) is proposed. First, the graph neural network is used to obtain the context transformation relationships between the items, and the graph embedding vectors of the items are obtained; Secondly, the graph embedding vector input into the interest attention network to extract the user's main interest preferences; Then the graph embedding vectors of the items are weighted by the attention layer; Finally, the click probability values of the candidate items are obtained through the prediction layer and sorted. The proposed algorithm model was verified by experiments on three public datasets Diginetica, Retailrocket and Tmall, which showed an improvement of 0.942%, 1.183% and 2.977% compared with the baseline model on MRR@20. Besides, the time complexity of the model is reduced, which verifies the effectiveness and high efficiency of the proposed method.

Key words attention mechanism; graph neural network; recommender algorithm; self-attention network; session-based recommendation

随着大数据和数字经济的发展，互联网信息呈爆炸式增长，同时也带来了“信息过载”的问题[1]。如何高效地从复杂多样的数据中获取用户需要的数据信息已成为当前重要的研究课题[2]。推荐系统可以根据用户需求为用户推荐其感兴趣的信息，现已成为解决信息过载问题的关键技术之一。目前，推荐系统已广泛应用于电子商务、音乐电影、新闻热点等领域[3]。

传统的推荐算法倾向于利用所有用户和物品的历史交互数据进行建模，然后根据用户对物品的兴趣偏好进行推荐[4]。然而在许多应用场景中，用户身份可能是未知的，并且在当前会话中只有用户历

收稿日期：2022−09−12；修回日期：2022−12−28
基金项目：重庆市自然科学基金面上项目（CSTB2022NSCQ-MSX1206）；重庆市教委重点项目（KJZD-K202200510）；重庆市科技局技术预见与制度创新项目（CSTB2022TFII-OFX0042）
作者简介：崔少国，博士，教授，主要从事大数据与人工智能等方面的研究。
*通信作者 E-mail: csg@cqnu.edu.cn

史的行为序列[5]。为了解决上述问题，基于会话的推荐系统近年来受到广泛关注，其根据匿名用户的历史行为序列来预测用户下一个感兴趣的物品并产生推荐[6]。其中，会话指用户在一段时间间隔内点击的物品行为序列。早期方法主要是基于马尔可夫链（Markov Chains, MC）[7] 和循环神经网络（Recurrent Neural Network, RNN）[8]。但这两种方法只能对物品之间的连续单向转换关系进行建模，忽略了其他物品之间的转换关系。为解决上述问题，基于图的会话推荐算法模型被提出[9]，此类算法模型首先将会话中所有物品之间的转换构建为图，再使用图神经网络学习物品的上下文信息。

近年来，自注意力网络（Self-Attention Network, SAN）[10] 因其可以捕捉序列中远距离物品之间的转换关系，在序列推荐中被广泛使用[11-12]。文献 [13] 通过使用图神经网络捕获物品的上下文信息，同时使用自注意力网络捕获物品的全局依赖关系，通过融合全局和当前依赖关系，提出了 GC-SAN 模型。但在用户交互过的所有历史物品中，用户的主要兴趣偏好往往只占一小部分，在使用自注意力网络进行长期兴趣偏好提取时，需要计算所有物品之间的交互，使得用户主要兴趣偏好的注意力权重在计算时可能不准确，最终导致推荐结果产生偏差，并且这种计算方法会使得时间成本成倍增加。

针对此问题，本文提出了一种基于兴趣注意力网络的会话推荐算法模型（SR-IAN）。本文的主要贡献有以下 3 点。

1）针对现有基于图神经网络的会话推荐算法对用户的主要兴趣提取不充分的问题，通过使用兴趣注意力网络提取用户的主要兴趣偏好，从而增强用户主要兴趣偏好对最终推荐结果的影响，提高推荐结果的准确性。

2）通过使用兴趣注意力网络，将用户的所有历史物品映射到其主要的兴趣偏好上，从而使得自注意力机制的计算复杂度降低，加快模型的训练速度。

3）使用注意力层对物品的图嵌入向量进行加权，以区分其对推荐结果的不同影响，减少无用特征向量对推荐结果的影响。

1 相关工作

1.1 基于会话的传统推荐算法

早期的基于会话的推荐算法主要分为基于相似性和基于链的方法。文献 [14] 提出基于物品邻域的方法，通过计算物品在同一会话中的共现频率来衡量物品之间的相似性，将用户最后点击物品的 k 近邻物品作为推荐结果，但此方法未考虑到物品之间的顺序关系；文献 [15] 提出了一种基于马尔可夫链的方法，其通过分解用户的个性化概率转移矩阵来产生推荐；文献 [16] 通过融合马尔可夫决策过程和隐因子模型，提出了一种马尔可夫链模型，该模型不仅可以捕捉到用户的长期兴趣偏好，还可以捕捉到物品交互的顺序关系；文献 [7] 提出了一种矩阵分解结合马尔可夫链的序列推荐模型 FPMC，从而获取相邻交互物品之间的隐藏顺序关系。基于马尔可夫链的方法大多都只考虑到了用户点击序列中连续物品之间的单向转换关系，无法捕获远距离物品之间的转换关系。

1.2 基于会话的深度学习推荐算法

随着深度学习的不断发展，循环神经网络在序列数据建模方面取得显著成果。文献 [8] 通过使用门控循环单元（Gate Recurrent Unit, GRU）来捕捉序列中长期的用户兴趣偏好，提出了 GRU4Rec 推荐模型；文献 [17] 在 GRU4Rec 模型基础上通过加入注意力机制提出了 NARM 模型，该模型考虑到会话中每个物品的重要程度，以捕获会话中具有代表性的物品信息；文献 [18] 提出了一种短期注意力记忆模型 STAMP，其能够从会话上下文的长期记忆中获取用户的长期兴趣偏好，同时将用户最后一次点击行为作为用户的当前短期兴趣偏好；文献 [11] 使用自注意力网络来捕获序列中用户的长期兴趣偏好，提出了 SASRec 模型；文献 [19] 使用低阶分解的自注意力网络和解耦位置编码方法来建模用户的长期兴趣偏好，从而更精确更高效地建模用户的长期依赖关系。

近年来，图神经网络（Graph Neural Network, GNN）在深度学习中兴起并且在各个领域得到了广泛的应用[20]。文献 [9] 使用图神经网络在有边相连的节点之间进行信息的传播，提出 SR-GNN 模型；文献 [21] 在 SR-GNN 的基础上加入目标注意力网络提出了 TA-GNN 模型，能准确地捕捉不同候选物品的优先级。文献 [13] 使用图神经网络和自注意力网络分别捕获物品的局部依赖和全局依赖关系，提出了 GC-SAN 模型，通过两者之间的优势互补来提升模型性能。但是直接使用自注意力网络对用户主要兴趣偏好提取往往是不准确的，并且

由于自注意力网络需要计算所有物品之间的交互，使得模型的计算时间复杂度较高。针对此问题，本文提出了使用兴趣注意力网络来提取用户主要兴趣偏好，使得自注意力机制在计算时只需关注用户的主要兴趣偏好，从而降低了自注意力机制计算的时间复杂度，并且通过注意力层对会话嵌入向量进行加权区分，以提高推荐结果的准确性。

2　模型设计

本文所提 SR-IAN 算法模型主要是根据用户历史点击物品序列来进行下一个物品的预测。首先使用图神经网络捕获物品之间的上下文转换关系，从而得到物品的图嵌入向量；其次使用兴趣注意力网络对历史物品序列中用户的主要兴趣偏好进行提取，然后通过注意力层对物品的图嵌入向量进行加权，得到用户的长期兴趣偏好，将其和用户的当前兴趣偏好进行融合，得到最终用户的会话表示，最后通过预测层得到每一个物品的点击概率值并按照概率值大小进行排序，将概率最大的前 K 个物品推荐给用户。算法模型的结构如图 1 所示。

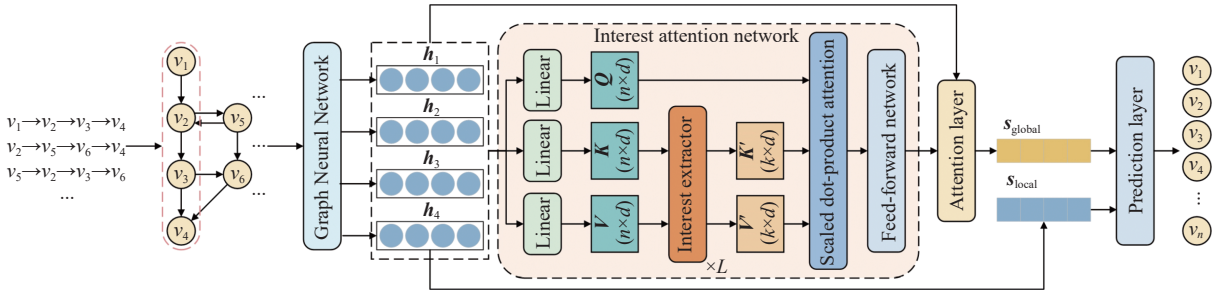


图 1　模型结构图

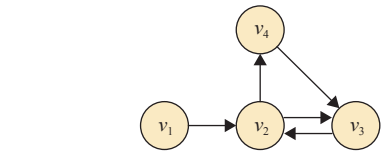
2.1　相关定义

基于可读性和后续公式表述，本文首先对文中所使用的符号进行定义及描述。$V=\{v_1,v_2,\cdots,v_n\}$表示会话中所有点击物品的集合，$S=\{s_1,s_2,\cdots,s_m\}$表示所有会话的集合，每一个会话对应一个物品点击序列$s_i=(v_1,v_2,\cdots,v_l)$，其中l表示会话长度，$v_i\in V$表示在会话s中第i次点击的物品。

算法模型的目的是根据当前会话中的物品点击序列来预测下一时刻用户可能点击的物品v_{t+1}，在输入会话序列数据后，算法模型根据输入信息得到每一个候选物品的点击概率值$\hat{y}=\{\hat{y}_1,\hat{y}_2,\cdots,\hat{y}_n\}$，其中$\hat{y}_i$表示候选物品$v_i$的点击概率值。将所有物品的概率值经过排序后，选择概率值最大的前 K 个物品构成推荐列表，推荐给用户。

2.2　会话图构建

首先将一个会话序列建模为一个有向图$G_s=\{V_s,E_s,\boldsymbol{A}_s\}$，其中$V_s$表示会话中出现的物品节点集合，$E_s$表示物品之间有向边的集合，$\boldsymbol{A}_s$表示当前会话的邻接矩阵。目的是将原会话序列转换为图结构，以便图神经网络对特征进行提取。在会话s中，每个节点表示一个物品$v_i\in V_s$，每一条边$(v_i,v_{i+1})\in E_s$表示用户在当前会话序列中依次点击了v_i和v_{i+1}两个物品节点，对于在当前会话中出现多次的物品，边的权重通过除以节点的度来进行归一化。具体而言，假设$\boldsymbol{A}_s^{(\mathrm{I})},\boldsymbol{A}_s^{(\mathrm{O})}\in\mathbb{R}^{n\times n}$分别为有向图$G_s$的入度矩阵和出度矩阵，其中矩阵的每一行分别表示该节点与其他节点之间的入边关系和出边关系，其值大小即为边的权重。当会话序列为$s=[v_1,v_2,v_3,v_2,v_4,v_3]$时，其归一化后的入度矩阵和出度矩阵如图 2 所示。



a. 会话图结构

	1	2	3	4
1	0	1	0	0
2	0	0	1/2	1/2
3	0	1	0	0
4	0	0	1	0

b. 出度矩阵

	1	2	3	4
1	0	0	0	0
2	1/2	0	1/2	0
3	0	1/2	0	1/2
4	0	1	0	0

c. 入度矩阵

图 2　会话图结构及出度、入度矩阵构建示意图

2.3　图神经网络模块

本文所使用的是由文献 [22] 提出的门控图神经网络（Gate Graph Nerual Network, GGNN），通过此方法可以在节点连接较多的情况下自动提取会

话图的特征。首先将会话图中的节点v_i通过嵌入层映射到d维嵌入空间中，得到节点的隐向量$\boldsymbol{u}_i \in \mathbb{R}^{\mathrm{d}}$，其次，对于会话图中$t$时刻的每个物品的节点向量$\boldsymbol{u}_i$，不同节点之间的信息传递定义为：

$$\begin{aligned}\boldsymbol{a}_t = \mathrm{Concat}(\boldsymbol{A}_t^{(\mathrm{I})}([\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_n]\boldsymbol{W}_a^{(\mathrm{I})}+\boldsymbol{b}^{(\mathrm{I})}),\\ \boldsymbol{A}_t^{(\mathrm{O})}([\boldsymbol{u}_1,\boldsymbol{u}_2,\cdots,\boldsymbol{u}_n]\boldsymbol{W}_a^{(\mathrm{O})}+\boldsymbol{b}^{(\mathrm{O})}))\end{aligned} \tag{1}$$

式中，$\boldsymbol{W}_a^{(\mathrm{I})},\boldsymbol{W}_a^{(\mathrm{O})} \in \mathbb{R}^{d\times d}$表示可学习的参数矩阵；$\boldsymbol{b}^{(\mathrm{I})},\boldsymbol{b}^{(\mathrm{O})} \in \mathbb{R}^{d}$表示偏置向量；$\boldsymbol{A}_t^{(\mathrm{I})},\boldsymbol{A}_t^{(\mathrm{O})} \in \mathbb{R}^{1\times n}$分别表示物品的节点向量$\boldsymbol{u}_t$的入度和出度矩阵的第$t$行向量。从而，$\boldsymbol{a}_t$中包含了物品的节点向量$\boldsymbol{u}_i$邻域的上下文信息。然后，将$\boldsymbol{a}_t$和会话图中前一时刻的物品的节点向量$\boldsymbol{u}_{t-1}$输入到图神经网络中，得到图神经网络的输出$\boldsymbol{h}_t$，其计算过程如下：

$$\boldsymbol{z}_t = \sigma(\boldsymbol{W}_z\boldsymbol{a}_t + \boldsymbol{U}_z\boldsymbol{u}_{t-1}) \tag{2}$$

$$\boldsymbol{r}_t = \sigma(\boldsymbol{W}_r\boldsymbol{a}_t + \boldsymbol{U}_r\boldsymbol{u}_{t-1}) \tag{3}$$

$$\tilde{\boldsymbol{h}}_t = \tanh(\boldsymbol{W}_h\boldsymbol{a}_t + \boldsymbol{U}_h(\boldsymbol{r}_t \odot \boldsymbol{u}_{t-1})) \tag{4}$$

$$\boldsymbol{h}_t = (1-\boldsymbol{z}_t)\odot\boldsymbol{u}_{t-1} + \boldsymbol{z}_t \odot \tilde{\boldsymbol{h}}_t \tag{5}$$

式中，$\boldsymbol{W}_z,\boldsymbol{W}_r,\boldsymbol{W}_h \in \mathbb{R}^{2d\times d},\boldsymbol{U}_z,\boldsymbol{U}_r,\boldsymbol{U}_h \in \mathbb{R}^{d\times d}$均为可学习的参数矩阵；$\sigma(\cdot)$表示 sigmoid 函数；$\odot$表示矩阵对应元素位置相乘；$\boldsymbol{z}_t,\boldsymbol{r}_t$分别表示 GRU 的更新门和重置门。其中，更新门决定要保留的信息，而重置门决定要丢弃的信息。

2.4 兴趣注意力网络模块

将会话序列输入图神经网络后，得到会话图中所有节点的隐向量$\boldsymbol{h}_i$，然后将其输入兴趣注意力网络中。兴趣注意力网络主要是通过一个映射函数$f:\mathbb{R}^{n\times d}\to\mathbb{R}^{k\times d}$将所有历史物品映射为用户所偏好的几类，从而在后续注意力矩阵计算时减少计算量。具体来说，假设用户的历史兴趣偏好可以分为k类（k是一个远远小于n的常数），给定会话图中物品节点的嵌入矩阵$\boldsymbol{H}=[\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_n]\in\mathbb{R}^{n\times d}$作为输入，首先计算物品到兴趣映射的分布矩阵$\boldsymbol{D}\in\mathbb{R}^{n\times k}$，其计算公式为：

$$\boldsymbol{D} = \mathrm{softmax}\left(\boldsymbol{H}\cdot\boldsymbol{\delta}^{\mathrm{T}}\right) \tag{6}$$

式中，$\boldsymbol{\delta}\in\mathbb{R}^{k\times d}$是一个可学习的参数矩阵。然后，使用$\boldsymbol{D}$矩阵对输入的物品嵌入矩阵进行相乘，得到用户的低维兴趣表示：

$$\boldsymbol{H}^{(\mathrm{I})} = f(\boldsymbol{H}) = \boldsymbol{D}^{\mathrm{T}}\cdot\boldsymbol{H} \tag{7}$$

经过f函数后，物品节点的嵌入矩阵$\boldsymbol{H}$被映射为低维兴趣表示$\boldsymbol{H}^{(\mathrm{I})}\in\mathbb{R}^{k\times d}$，从而有效减少了注意力矩阵的大小，使得网络的前向传播效率更快。而且用户的潜在兴趣捕获了物品序列中所体现的用户总体偏好，因此，模型对用户主要兴趣偏好的关注会更准确。

然后，将输入的嵌入矩阵通过$\boldsymbol{W}_Q,\boldsymbol{W}_K,\boldsymbol{W}_V$映射转换为$\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\in\mathbb{R}^{n\times d}$矩阵，其次将多头自注意力机制中的$\boldsymbol{K},\boldsymbol{V}\in\mathbb{R}^{n\times d}$通过物品到兴趣的函数$f$映射为$\boldsymbol{K}',\boldsymbol{V}'\in\mathbb{R}^{k\times d}$矩阵，最终得到所有物品节点的表示：

$$\begin{aligned}\boldsymbol{F} = \mathrm{softmax}\left(\frac{\boldsymbol{Q}\cdot\boldsymbol{K}'^{\mathrm{T}}}{\sqrt{d}}\right)\boldsymbol{V}' =\\ \mathrm{softmax}\left(\frac{(\boldsymbol{H}\boldsymbol{W}_Q)\cdot f(\boldsymbol{H}^{(\mathrm{I})}\boldsymbol{W}_K)^{\mathrm{T}}}{\sqrt{d}}\right)f(\boldsymbol{H}^{(\mathrm{I})}\boldsymbol{W}_V)\end{aligned} \tag{8}$$

通过上述变换操作，自注意力机制的计算复杂度从$O(n^2)$降低到$O(nk)$。

其次，通过两层带有 GeLU 激活函数的线性变换，从而使得模型具有学习非线性特征的能力。由于在自注意力计算过程中会出现梯度消失情况，故而本文在前馈神经网络后加入了残差连接。其计算过程为：

$$\begin{aligned}\boldsymbol{E} = \mathrm{FFN}(\boldsymbol{F}) =\\ \mathrm{GeLU}(\boldsymbol{W}_1\boldsymbol{F}+\boldsymbol{b}_1)\boldsymbol{W}_2+\boldsymbol{b}_2+\boldsymbol{F}\end{aligned} \tag{9}$$

式中，$\boldsymbol{W}_1,\boldsymbol{W}_2\in\mathbb{R}^{d\times d}$是可学习的参数矩阵；$\boldsymbol{b}_1$和$\boldsymbol{b}_2$是偏置向量。此外，本文在训练过程中加入了 Dropout[23] 来防止过拟合。为方便描述，上述过程形式化定义为：

$$\boldsymbol{E} = \mathrm{IAN}(\boldsymbol{H}) \tag{10}$$

通过堆叠多个兴趣注意力网络层，可以捕获到更深层次的特征信息。第一层定义为$\boldsymbol{E}^{(1)}=\boldsymbol{E}$，第$L$层兴趣注意力网络定义为：

$$\boldsymbol{E}^{(L)} = \mathrm{IAN}(\boldsymbol{E}^{(L-1)}) \tag{11}$$

式中，$\boldsymbol{E}^{(L)}\in\mathbb{R}^{n\times d}$是兴趣注意力网络最后一层的输出。

2.5 注意力层模块

然后通过一层注意力层，得到每一个图嵌入向量的权重，和图神经网络的嵌入向量进行相乘，从而区分每个图嵌入向量对推荐结果产生的不同影响。通过注意力层后得到用户的长期兴趣偏好表示：

$$\alpha = \mathrm{softmax}\left(w\boldsymbol{E}^{(L)^{\mathrm{T}}}\right) \tag{12}$$

$$\boldsymbol{S}_{\mathrm{global}} = \alpha\boldsymbol{H} \tag{13}$$

式中，w是可学习参数；$\boldsymbol{S}_{\mathrm{global}}$即为用户的长期兴趣表示。

2.6　预测模块

通过兴趣注意力网络和注意力层后，得到了用户的长期兴趣偏好表示向量。但用户对物品的选择不仅取决于其长期兴趣偏好，还取决于当前的短期兴趣偏好，短期兴趣偏好定义为在当前会话中最后一次点击的物品[9,17]，即：

$$\boldsymbol{S}_{\text{local}} = \boldsymbol{h}_t \tag{14}$$

通过将长期兴趣偏好和短期兴趣偏好进行加权，得到用户最终的会话表示向量：

$$\boldsymbol{S}_f = \omega \boldsymbol{S}_{\text{global}} + (1-\omega)\boldsymbol{S}_{\text{local}} \tag{15}$$

式中，ω是控制长期兴趣偏好和当前兴趣偏好在用户会话表示中所占比重的权重因子。最后，使用 softmax 函数得到每一个候选物品v_i点击的概率值：

$$\hat{y}_i = \text{softmax}(\boldsymbol{S}_f^{\text{T}} v_i) \tag{16}$$

式中，$\hat{y}_i$表示物品v_i在当前会话中成为下一个点击物品的概率值。经过排序，将点击概率值最大的前 K 个物品推荐给用户。

模型训练的损失函数使用的是交叉熵损失函数。为防止过拟合，本文在损失函数中加入了 L2 正则化约束：

$$\text{Loss} = -\sum_{i=1}^{n} y_i \ln(\hat{y}_i) + (1-y_i)\ln(1-\hat{y}_i) + \lambda \|\theta\|_2 \tag{17}$$

式中，y_i表示物品v_i经 one-hot 编码后的真实标签；θ是所有可学习的参数集；λ是 L2 正则化的惩罚系数。

3　实验结果与分析

3.1　实验环境

本实验是在 Windows10 操作系统下，处理器为 Intel® Core™ i9-10920X CPU @ 3.50 GHz，运行内存为 32 GB，显卡型号为 NVIDIA GeForce RTX 3080Ti，显存为 12 GB，实验中使用的是 Python 3.8 编程语言，深度学习计算框架为 Pytorch 1.11.1 版本，推荐算法模型使用的是 Recbole 1.0.1[24] 框架。

3.2　数据集及预处理

本实验使用的是电子商务环境下的 3 个推荐算法公开数据集：Diginetica（http://cikm2016.cs.iupui.edu/cikm-cup）、Retailrocket（https://www.kaggle.com/datasets/retailrocket/ecommerce-dataset）和 Tmall（https://tianchi.aliyun.com/dataset/dataDetail?dataId=42）。Diginetica 数据集来自于 CIKM2016 比赛，包含电商平台 5 个月内用户的点击日志；Retailrocket 数据集包含电商平台 6 个月的用户浏览活动；Tmall 数据集来自于 IJCAI-15 比赛，包含了匿名用户在天猫电商购物平台上的购物日志。数据集具体信息如下表 1 所示。

表 1　实验数据集

Dataset	Interactions	Items	Sessions	Avg.Length
Diginetica	786 582	42 862	204 532	4.12
Retailrocket	871 637	51 428	321 032	6.40
Tmall	427 797	37 367	66 909	10.62

首先对数据集进行预处理，过滤掉所有数据集中长度为 1 的会话和出现次数少于 5 次的点击物品[9,17]，并将每个数据集中的会话序列按时间先后顺序及 8:1:1 的比例划分为训练集、验证集和测试集。

3.3　评价指标

本文采用 Recall@K 和 MRR@K 评价指标对实验结果进行评估。本实验中 K 值设置为 10 和 20。

Recall@K 是用于衡量算法模型预测准确性的指标，表示推荐结果排名列表中排在前 K 个推荐物品中正确样本数量占所有测试数的比例，其计算公式为：

$$\text{Recall@K} = \frac{n_{\text{hit}}}{N} \tag{18}$$

式中，N表示测试集中的样本数量；n_{hit}表示前 K 个推荐物品中有此样本正确答案的样本数。

MRR@K（Mean Reciprocal Rank, MRR）是平均倒数排名，当排名超过 K 时，MRR@K 的值设置为 0。MRR 的值考虑到了推荐排名的顺序，其值越大，表明正确的推荐结果排名越靠前。其计算公式为：

$$\text{MRR@K} = \frac{1}{N}\sum_{i \in M} \frac{1}{\text{rank}_i} \tag{19}$$

式中，N表示测试集中样本总数；M表示前 K 个推荐物品中包含正确推荐物品的样本集；rank_i表示物品i在推荐列表中的排名。

3.4　对比模型

Pop：传统的基于会话的推荐模型，只推荐训练集中出现次数最频繁的 Top-N 个物品。

FPMC[7]：结合矩阵分解和一阶马尔可夫链来进行下一个物品推荐。

GRU4Rec[8]：使用 GRU 对用户序列进行建模，并利用会话并行的小批量训练过程来模拟用户的行为序列。

NARM[17]：一种在 RNN 的基础上加入注意力

机制的模型，通过注意力机制从隐藏态捕获用户的意图，并结合用户点击顺序行为信息得到最终的推荐结果。

SASRec[11]：一种使用自注意力网络对物品序列中用户的长期兴趣偏好进行提取的序列模型。

LightSANs[19]：通过使用低阶分解的自注意力网络对物品信息进行提取的序列模型。

SR-GNN[9]：通过使用图神经网络来捕获物品之间的转换关系，并结合用户最后一次的点击物品进行推荐。

TA-GNN[21]：在 SR-GNN 的基础上加入目标注意力网络，从而能够准确地捕捉到不同候选物品的优先级。

GC-SAN[13]：使用图神经网络和自注意力网络分别捕获物品的上下文依赖和长期依赖关系，最后通过门控机制将长期兴趣和当前兴趣进行融合。

3.5 参数设置

上述所有模型均使用 Pytorch 深度学习框架和 Recbole 推荐算法框架实现。实验中为了统一对比不同模型，epoch 统一设置为 30，batch_size 大小统一设置为 100，Embedding 嵌入维度设置为 100，模型学习率设置为 0.001，使用 Adam 优化器[25] 进行模型训练，其他超参数根据模型原论文建议进行了调整。以验证集的 MRR@20 为模型的早停标准，如果 MRR@20 评价指标在 10 个 epoch 没有提升，则结束训练。

3.6 实验结果及分析

为了评估本文所提算法模型的有效性，本文从不同角度设计了对比实验和消融实验并对实验结果进行了分析讨论。

3.6.1 模型的性能对比与分析

为了验证本文所提算法模型的有效性，将 SR-IAN 模型与上述对比模型进行了实验分析，实验结果如表 2 所示。

由表 2 可知，传统的推荐方法 Pop 在基于会话的推荐任务中表现较差，因为其忽略了用户在当前会话中的兴趣偏好，仅仅考虑了前 K 个最受欢迎的物品；FPMC 由于引入了一阶马尔可夫链来建模会话中物品的转移序列，故而其表现相对 Pop 方法而言更好。

表 2 不同数据集上的实验对比结果 %

Model	Diginetica				Retailrocket				Tmall			
	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20
Pop	0.514	0.181	0.832	0.202	1.012	0.214	1.424	0.241	0.147	0.037	6.649	0.443
FPMC	21.816	8.127	32.904	8.888	39.363	22.496	46.538	22.993	18.864	13.144	19.981	13.222
GRU4Rec	33.964	14.700	46.487	15.562	50.960	34.217	57.387	34.663	28.779	21.752	31.875	21.968
NARM	34.703	14.770	47.226	15.634	51.750	34.980	58.208	35.431	31.065	23.058	34.288	23.281
SASRec	34.979	15.350	47.768	16.233	52.503	35.059	58.938	35.507	34.310	25.231	37.824	25.474
LightSANs	35.724	15.530	48.311	16.398	52.598	35.140	59.085	35.593	35.412	25.970	39.062	26.223
SR-GNN	34.860	14.949	47.475	15.818	51.078	33.858	57.334	34.294	31.460	23.608	34.446	23.816
TA-GNN	34.974	15.149	47.657	16.026	51.185	33.607	57.635	34.055	30.591	22.846	33.767	23.067
GC-SAN	35.566	15.704	47.955	16.561	52.435	35.270	58.772	35.710	33.027	24.714	36.213	24.934
SR-IAN	**37.763**	**16.614**	**50.660**	**17.503**	**53.586**	**36.416**	**60.423**	**36.893**	**36.135**	**27.635**	**40.131**	**27.911**

与传统模型相比，基于深度学习的方法具有更好的表现。GRU4Rec 方法是一种基于 RNN 的方法，其性能相较于传统推荐方法有明显提升，说明 RNN 对序列数据有一定的建模能力；NARM 将 RNN 和注意力机制相结合，使用 RNN 最后的隐藏态作为用户的主要兴趣偏好，其结果相较于 GRU4Rec 有一定的提升，说明在会话推荐任务中加入注意力机制可以捕获会话中用户的主要偏好，并且提高了推荐结果的准确性。SASRec 模型相较于其他序列模型具有更好的表现，其使用自注意力网络捕捉远距离物品之间的依赖关系，表明自注意力网络对长期依赖关系具有良好的捕获能力；LightSANs 模型通过使用低阶分解的自注意力网络来加速模型训练，并且使用解耦位置编码从而更有效建模物品之间的顺序关系，其性能相较于直接使用自注意力网络的 SASRec 模型更优。

基于图神经网络的模型通过引入图神经网络来捕获物品之间的转换关系，并且结合注意力机制来捕获用户的长期兴趣偏好，取得了不错的效果。SR-GNN 将会话序列建模为会话图，然后使用

GNN 捕获相邻物品之间的转换关系，其性能表现优于多数基于 RNN 的模型；TA-GNN 通过在 SR-GNN 基础上加入目标注意力网络，使得模型能够捕获到候选目标物品和当前会话中物品的依赖关系，从而增强了模型的表达能力，但是由于模型需要对每一个候选物品进行注意力计算，导致模型需要耗费更长的训练时间；GC-SAN 模型通过使用图神经网络来捕获物品的上下文依赖关系，并且引入自注意力网络来捕获用户历史物品中的全局依赖关系，通过将两部分上下文局部依赖关系进行加权融合，从而提升推荐结果的准确性。

由表 2 可知，在 3 个电商平台的数据集上，本文所提的模型相比其他模型而言表现更优。具体来说，在 3 个公开数据集上，MRR@20 指标相比表现最佳的方法模型分别有着 5.68%、3.31% 和 6.43% 的相对提升。本文所提的方法通过使用兴趣注意力网络将所有历史物品映射到很少的几类用户的主要兴趣偏好中，再通过注意力层对图嵌入向量进行加权区分，从而使得用户的会话表示更加准确，故模型性能表现更优，验证了本文所提模型的有效性。

3.6.2　兴趣注意力网络对模型性能的影响

在基准模型 GC-SAN 基础上将自注意力网络替换为兴趣注意力网络，得到 GC-IAN 模型，通过对兴趣注意力网络的消融实验来验证其有效性。消融实验结果如表 3 所示。

表 3　消融实验结果　　%

Model	Diginetica				Retailrocket				Tmall			
	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20	R@10	M@10	R@20	M@20
GC-SAN	35.566	15.704	47.955	16.561	52.435	35.270	58.772	35.710	33.027	24.714	36.213	24.934
GC-IAN	35.924	16.048	48.271	16.901	52.524	35.428	58.967	35.965	33.202	24.696	36.509	24.926
SR-IAN	**37.763**	**16.614**	**50.660**	**17.503**	**53.586**	**36.416**	**60.423**	**36.893**	**36.135**	**27.635**	**40.131**	**27.911**

由表 3 可知，加入兴趣注意力网络的模型 GC-IAN 相较于基准模型 GC-SAN 在 Recall 和 MRR 两个评价指标上，在大多数情况下表现更优，说明兴趣注意力网络在基于图神经网络的会话推荐算法中发挥了其能够提取用户主要兴趣偏好的作用。同时，在加入注意力层对图嵌入向量进行加权区分后，模型性能达到最佳，说明不同物品的图嵌入向量对推荐结果确实存在不同影响，需要对其进行区分。

3.6.3　模型复杂度对比

为了验证本文所加入的兴趣注意力网络对模型复杂度的影响，本文从时间复杂度和空间复杂度两个方面进行对比分析。为了进行客观公正的对比，自注意力网络部分使用相同的超参数，通过和基于自注意力网络的模型进行对比，从而分析兴趣注意力网络对模型复杂度的影响。其中，空间复杂度主要是通过实验过程中统计到的模型可学习参数量进行对比分析；时间复杂度主要是以一个 epoch 的平均训练时间来进行对比。对比结果如表 4 所示。

由表 4 可知，本文所提模型和使用自注意力网络的模型相比参数量虽有所增加，但总体而言参数量相差不是很大，即模型在空间复杂度上是可接受的；但是，从模型训练耗时来看，本文所提模型在训练时间上相较其他模型而言用时更少。具体而言，在 3 个数据集上，本文所提模型相比基准模型 GC-SAN 而言，其训练时间分别是基准模型的 54.9%、69.9% 和 51.1%。说明本文加入的兴趣注意力网络加速了模型的训练速度，从而降低了模型的计算复杂度。故本文所提模型在时间复杂度上优于其他对比模型。

表 4　模型复杂度对比结果

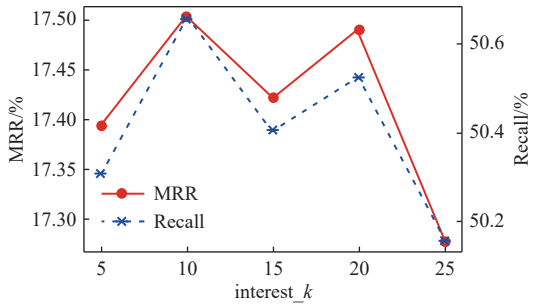
Model	Diginetica		Retailrocket		Tmall	
	Params/M	Times/s	Params/M	Times/s	Params/M	Times/s
SASRec	**4.66**	1 654.01	**5.50**	1 224.95	**4.11**	402.46
LightSANs	4.75	1 318.62	5.58	1 463.06	4.20	729.33
GC-SAN	4.76	1 539.46	5.60	1 279.20	4.21	673.69
SR-IAN	4.77	**845.76**	5.62	**894.30**	4.22	**378.30**

综上，本文所提算法模型相比其他模型，在时间和空间复杂度上均是可接受的，从而验证了加入

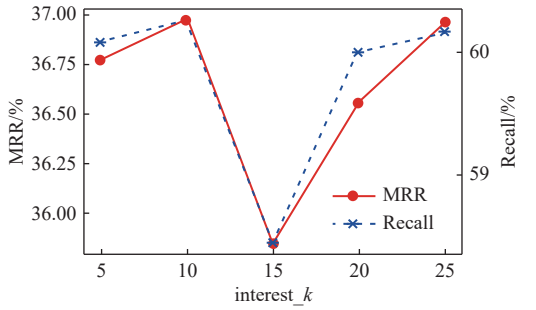
兴趣注意力网络的可行性。

3.6.4 关键超参数对模型性能的影响

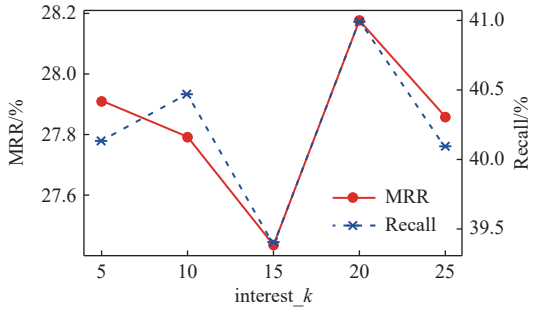
本文所提模型包含两个关键超参数：兴趣注意力网络中用户的主要兴趣个数 interest_k 和控制长期兴趣偏好和当前兴趣偏好所占比重的权重因子 ω。超参数 interest_k 在 3 个数据集上的实验结果如图 3 所示。



a. Diginetica数据集实验结果



b. Retailrocket数据集实验结果



c. Tmall数据集实验结果

图 3 不同 interest_k 值下的模型性能对比

由图 3 可知，在 Diginetica 和 Retailrocket 数据集上，interest_k 的最佳取值均为 10，在 Tmall 数据集上，interest_k 的最佳取值为 20。通过分析数据集可知，由于 Diginetica 和 Retailrocket 数据集中会话序列平均长度较短且相差不大，因此会话中所包含的物品个数也差不多，从而会话序列中用户的主要兴趣偏好也相近；而 Tmall 数据集中会话序列平均长度为前两个数据集的两倍之多，故而在一个会话中所包含的用户长期兴趣也会随之增多，所以超参数 interest_k 的取值也会变大。

控制长期兴趣偏好和当前兴趣偏好所占比重的权重因子ω在 3 个数据集上 MRR@20 指标的变化趋势如图 4 所示。

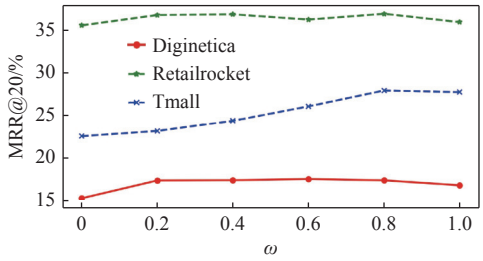


图 4 不同权重因子ω下的模型性能对比

由图 4 可知，在仅仅使用当前兴趣偏好（$\omega=0.0$）的情况下，模型性能表现较差，说明用户的长期兴趣偏好对推荐结果至关重要。随着长期兴趣偏好所占比例的不断上升，模型性能也随之提升，当权重因子ω在 0.4～0.8 之间时，模型性能表现最优，当只使用长期兴趣偏好进行推荐时，MRR@20 评价指标却降低了，说明当前兴趣偏好对推荐结果也发挥了一定的作用。

4 结束语

针对现有会话推荐算法对用户主要兴趣偏好建模不准确不充分的问题，提出了一种基于兴趣注意力的会话推荐算法 SR-IAN。算法模型首先通过图神经网络捕获物品节点的上下文转换关系，其次使用兴趣注意力网络捕获用户的主要兴趣偏好，从而更准确地建模用户的长期兴趣偏好，然后使用注意力层对物品的图嵌入向量进行加权，以区分出不同物品节点对推荐结果的不同影响，最后通过预测层得到每一个候选物品的点击概率值并对其进行排序，将概率值最大的物品推荐给用户。通过在多个推荐算法公开数据集上的实验，证明了本文所提方法的有效性和高效性。

在后续研究工作中，主要是使用不同的序列模型对用户长期兴趣偏好进行更有效地建模，结合图神经网络捕获的上下文转换关系来生成用户的会话表示，从而提高基于会话的推荐算法的准确性和效率，更好地为用户提供服务。

参 考 文 献

[1] ZHANG S, YAO L, SUN A, et al. Deep learning based recommender system: A survey and new perspectives[J]. ACM Computing Surveys (CSUR), 2019, 52(1): 1-38.

[2] 刘君良, 李晓光. 个性化推荐系统技术进展[J]. 计算机科学, 2020, 47(7): 47-55.

LIU J L, LI X G. Techniques for recommendation system: A survey[J]. Computer Science, 2020, 47(7): 47-55.

[3] 黄立威, 江碧涛, 吕守业, 等. 基于深度学习的推荐系统研究综述[J]. 计算机学报, 2018, 41(7): 1619-1647.
HUANG L W, JIANG B T, LYU S Y, et al. Survey on deep learning based recommender systems[J]. Chinese Journal of Computers, 2018, 41(7): 1619-1647.

[4] 李孟浩, 赵学健, 余云峰, 等. 推荐算法研究进展[J]. 小型微型计算机系统, 2022, 43(3): 544-554.
LI M H, ZHAO X J, YU Y F, et al. Survey on research progress of recommendation algorithms[J]. Journal of Chinese Computer Systems, 2022, 43(3): 544-554.

[5] 曾义夫, 牟其林, 周乐, 等. 基于图表示学习的会话感知推荐模型[J]. 计算机研究与发展, 2020, 57(3): 590-603.
ZENG Y F, MU Q L, ZHOU L, et al. Graph embedding based session perception model for next-click recommendation[J]. Journal of Computer Research and Development, 2020, 57(3): 590-603.

[6] WANG S J, CAO L B, WANG Y, et al. A survey on session-based recommender systems[J]. ACM Computing Surveys (CSUR), 2021, 54(7): 1-38.

[7] 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. North Carolina: ACM, 2010: 811-820.

[8] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks[C]//Proceedings of the 4th International Conference on Learning Representations. San Juan, Puerto Rico: ICLR, 2016, 1-10.

[9] WU S, TANG Y Y, ZHU Y Q, et al. Session-based recommendation with graph neural networks[C]// Proceedings of the AAAI Conference on Artificial Intelligence. Hawaii: AAA Press, 2019, 33(1): 346-353.

[10] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Advances in Neural Information Processing Systems. Long Beach: NeurIPS, 2017: 5998-6008.

[11] KANG W C, MCAULEY J. Self-attentive sequential recommendation[C]//2018 IEEE International Conference on Data Mining (ICDM). Singapore: IEEE Computer Society, 2018: 197-206.

[12] SUN F, LIU J, WU J, et al. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing: ACM, 2019: 1441-1450.

[13] XU C F, ZHAO P P, LIU Y C, et al. Graph contextualized self-attention network for session-based recommendation [C]//Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao, China: AAAI Press, 2019: 3940-3946.

[14] 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. Hong Kong, China: ACM, 2001: 285-295.

[15] SHANI G, HECKERMAN D, BRAFMAN R I. An MDP-based recommender system[J]. Journal of Machine Learning Research, 2005, 6(9): 1265-1295.

[16] GU W R, DONG S B, ZENG Z Z. Increasing recommended effectiveness with Markov chains and purchase intervals[J]. Neural Computing and Applications, 2014, 25(5): 1153-1162.

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

[18] LIU Q, ZENG Y F, 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. London: ACM, 2018: 1831-1839.

[19] FAN X Y, LIU Z, LIAN J X, et al. Lighter and better: Low-rank decomposed self-attention networks for next-item recommendation[C]//Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. [S.l.]: ACM, 2021: 1733-1737.

[20] ZHANG Z W, CUI P, ZHU W W. Deep learning on graphs: A survey[J]. IEEE Transactions on Knowledge & Data Engineering, 2020, 34(1): 249-270.

[21] YU F, ZHU Y Q, 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 Information Retrieval. [S.l.]: ACM, 2020: 1921-1924.

[22] LI Y J, ZEMEL R, BROCKSCHMIDT M, et al. Gated graph sequence neural networks[C]//Proceedings of the 4th International Conference on Learning Representations. San Juan, Puerto Rico: ICLR, 2016: 1-20.

[23] 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.

[24] ZHAO W X, MU S L, HOU Y P, et al. Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. [S.l.]: ACM, 2021: 4653-4664.

[25] KINGMA D, BA J. Adam: A method for stochastic optimization[C]//Proceedings of the 3rd International Conference on Learning Representations. San Diego: ICLR, 2015: 1-15.

编　辑　张　莉