

刁昕. 会话增强图神经网络推荐模型[J]. 智能计算机与应用, 2024, 14(4): 244-250. DOI: 10.20169/j.issn.2095-2163.240440

会话增强图神经网络推荐模型

刁昕

(上海理工大学 光电信息与计算机工程学院, 上海 200082)

摘要: 基于会话推荐的目标在于预测匿名用户的行为操作。然而,许多现有的会话推荐模型未能充分考虑相似会话对推荐性能的影响。尽管图神经网络能更好地捕捉会话内项目的转换关系,但一些会话内的转换关系却不利于推荐,因此模型需要学习更具代表性的会话嵌入。为了解决这些问题,本文提出了一种改进的会话增强图神经网络推荐模型——SE-GNNRM。该模型在编码阶段通过图神经网络和自注意力机制,捕捉复杂的项目转换关系和更具代表性的项目特征;利用注意机制将短期偏好和长期偏好相结合,构建全局会话图,并利用相似度融合的图注意网络捕获相似会话信息。为了证明所提模型的有效性,经在3个公共数据集上的实验结果表明,SE-GNNRM 优于现有的基线模型,是一种有效的基于会话的推荐模型。

关键词: 会话推荐; 匿名; 图神经网络; 相似度融合

中图分类号: TP181

文献标志码: A

文章编号: 2095-2163(2024)04-0244-07

Session enhanced graph neural network recommendation model

DIAO Xin

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200082, China)

Abstract: The main objective of session-based recommendation is to predict the behavior of anonymous users. However, many conversational recommendation models neglect the impact of similar conversations, which can affect recommendation performance. Although graph neural networks can capture the transitional relationships between items within a session, some intra-session transitions are not useful for recommendation, thereby requiring the model to acquire more representative session embeddings. To address these limitations, this paper proposes an improved model, called SE-GNNRM, which is a session-enhanced graph neural network recommendation model. SE-GNNRM uses a graph neural network and self-attention mechanism to capture complex item transformation relationships and more representative item features during the encoding stage. Then, the model employs an attention mechanism to combine short-term and long-term preferences for building a global session graph, while a similarity-fused graph attention network captures analogous session information. To showcase SE-GNNRM's efficacy, the model is compared to existing baseline models on three public datasets. Results indicate that SE-GNNRM outperforms baseline models and is an efficient session-based recommendation method.

Key words: session recommendation; anonymity; graph neural network; similarity fusion

0 引言

随着互联网的普及和信息时代的到来,网民每天面对和处理的信息呈指数级增长。为了选择感兴趣的内容,推荐系统应运而生。以往大多数研究都是基于不间断的用户历史记录进行个性化推荐,但在许多互联网服务中(如:Bilibili、抖音),由于存储资源有限,不可能或没有必要跟踪用户的身份行为。换句话说,用户档案和长期历史交互是不可用的,系统中只有匿名用户的短期会话数据。针对于此,

基于会话的推荐算法研究变得越来越重要^[1]。用户在一定时间间隔内产生的点击行为信息为会话序列,为了处理这些行为信息,基于会话的推荐系统提取序列中多个点击项的依赖关系,并捕获序列预测^[2]的重要性。用户的兴趣在一定时间跨度内展现出相对的稳定性,这对于推荐领域来说很重要,因此基于会话序列的推荐算法近年来受到了更多的关注^[3]。

目前,已经诞生了许多基于会话的推荐模型。如:早期的马尔可夫链会话推荐,主要根据用户最后一次与系统交互的行为,来预测用户的下一个行

为^[4]。随着深度学习的发展,循环神经网络(RNN)方法逐渐应用于会话推荐,此类方法将用户会话点击序列输入 RNN 生成推荐结果^[5]。虽然循环神经网络在一定程度上是有效的,但是大多数现有的基于 RNN 的模型在捕捉项目之间的复杂转换时,表现能力有限。最近,基于图神经网络(GNN)的推荐算法开始流行。Wu 等^[6]采用 GNN 方法进行用户推荐,解决了会话间依赖关系弱的问题,使得推荐效果大大提高。其方法是将会话数据转换为图结构,并结合多层门控循环单元(GRU),来捕捉用户的长期兴趣和当前偏好。该方法将会话图构造的会话点击项矩阵,以及会话点击序列输入到 GNN 中进行序列建模,但不能很好地过滤无用的依赖关系。随着深度神经网络的日益普及,注意力网络在许多领域(如:计算机视觉、自然语言处理、推荐系统等)得到了广泛的发展,使用源自自然语言处理领域的自注意力机制,来处理会话序列的算法也开始兴起。

为了克服上述方法的局限性,本文结合图神经网络(GNN)、自注意力网络(SAN)、软注意机制等相关技术,提出了会话增强图神经网络推荐模型(Session Enhanced Graph Neural Network Recommendation Model, SE-GNNRM)。该模型包括项目嵌入层、会话嵌入层、会话交互层和预测层。其中,项目嵌入层包括 GNN 和 SAN 两个模块,GNN 和 SAN 得到的项目嵌入,通过联合函数进行组合。在会话嵌入层中,将

项目嵌入层的输出项目嵌入与位置信息相结合,然后利用注意力学习会话嵌入。此时的会话嵌入仅表示单个会话嵌入,无法充分捕捉到用户的长期偏好。为了解决这一问题,该模型设计了一个动态全局会话交互层来学习全局会话嵌入。最后,通过预测层计算候选选项得分,得到最终推荐结果。

1 会话增强图神经网络推荐模型

1.1 模型定义

基于会话的推荐,是指在给定的行为序列后,只考虑当前行为信息,不考虑用户历史点击或属性等信息的情况下,预测用户下一步可能点击的项目。问题定义如下:令 $V = [v_1, v_2, v_3, \dots, v_m]$ 表示所有项目的集合, m 是项目总数; $S = [v_{s,1}, v_{s,2}, v_{s,3}, \dots, v_{s,n}]$ 表示按时间戳排序的会话序列,其中 $v_{s,n} \in V$ 是用户在第 n 步点击的项目。基于会话的推荐预测的是下一次点击为 $v_{s,n}$ 的项目。具体来说,推荐模型返回一个预测项目得分的列表,这些得分是 $\hat{y} = [\hat{y}_1, \hat{y}_2, \hat{y}_3, \dots, \hat{y}_m]$, 其中 \hat{y}_i 表示与项目集 v_i 相关的预测得分,并推荐得分最高的前 k 个候选项目。

1.2 模型结构

SE-GNNRM 包括项目嵌入层、会话嵌入层、会话卷积层和预测层,模型结构如图 1 所示。

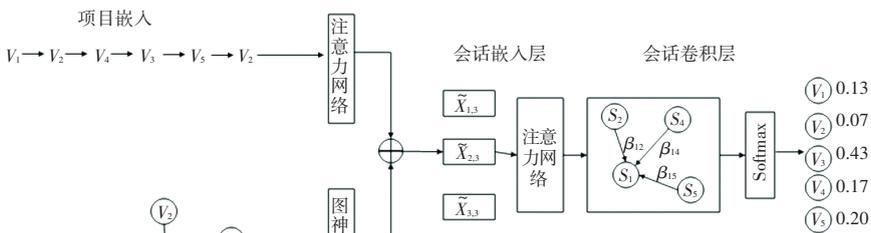


图 1 SE-GNNRM 的详细结构

Fig. 1 Detailed structure of SE-GNNRM

1.2.1 项目嵌入层

模型的输入是经过预处理的会话序列,并构造了所有会话序列的有向图。项目嵌入层包含两个模块,一个是图神经网络(GNN)模块,一个是自注意力(SAN)模块。将会话序列的有向图输入到 GNN 模块中,利用其对图结构节点强大的嵌入学习能力学习项目嵌入。会话序列被输入到 SAN 模块中,通过自注意机制学习会话序列之间的相关性,以获得更有代表性的项目嵌入。

GNN 在拓扑图上生成节点嵌入,以建模项目特征转换的依赖关系,适用于基于会话的推荐。每个经过预处理的会话序列 S 被建模为有向图 $G_s = (V_s, E_s)$, 作为 GNN 模块的输入。其中,每个节点表示一个项目 $v_{s,i} \in V_s$, 有向图中的每个连接表示会话中项目之间的连接。将每一项表示在一个统一的空间中,得到的向量 $v_i \in R^d$ 是通过 GNN 得到的 v_i 的 d 维嵌入。Li 等^[7]在 RNN 思想的基础上,将 GRU 单元引入 GNN,提出了一种可变门控图神经网络

(GGNN),使其更适合处理序列数据。该模型使用GGNN提取节点特征,更新后的会话 S 中,节点的 $v_{s,i}$ 传播公式如下。

$$a_{s,i}^{(t)} = A_{s,i}: [v_1^{(t-1)}, \dots, v_n^{(t-1)}]^T W_H + b_1 \quad (1)$$

$$z_{s,i}^{(t)} = \sigma(W_z a_{s,i}^{(t)} + U_z v_i^{(t-1)} + b_2) \quad (2)$$

$$r_{s,i}^{(t)} = \sigma(W_r a_{s,i}^{(t)} + U_r v_i^{(t-1)} + b_3) \quad (3)$$

$$\tilde{v}_i^{(t)} = \tanh(W_o a_{s,i}^{(t)} + U_o (r_{s,i}^{(t)} \odot v_i^{(t-1)})) \quad (4)$$

$$v_i^{(t)} = (1 - z_{s,i}^{(t)}) \odot v_i^{(t-1)} + z_{s,i}^{(t)} \odot \tilde{v}_i^{(t)} \quad (5)$$

其中, $W_H \in R^{d \times 2d}$ 、 $W_z, U_z, W_r \in R^{d \times 3d}$ 、 $W_o, U_o \in R^{d \times d}$ 为可学习权矩阵。 $b_1 \in R^d$ 以及 $b_2, b_3 \in R^{3d}$ 为偏置向量, $z_{s,i}^{(t)}$ 和 $r_{s,i}^{(t)}$ 分别为重置门和更新门。复位门决定如何将新的输入信息与以前的信息结合起来;更新门定义了将以前的信息保存到当前时间步长的内容。 $[v_1^{(t-1)}, \dots, v_n^{(t-1)}]$ 是嵌入会话 s 中的节点向量, $\sigma(\cdot)$ 是一个 sigmoid 函数, \odot 表示逐个元素的乘法。 $A_s \in R^{n \times 2n}$ 定义为两个相邻阶矩阵 $A_s^{(\text{out})}$ 和 $A_s^{(\text{in})}$ 的拼接,表示会话图的入边和出边的加权连接。邻接矩阵 A_s 区分了有向图中边的类型,并定义了有向图中的节点如何通过入边和出边相互连接。 $A_{s,i} \in R^{1 \times 2n}$ 是 A_s 中的第 i 行,是节点 $v_{s,i}$ 所对应的部分。此外,由于会话序列中可能出现重复项,因此对每条边计算一个归一化赋值,即边的初始值除以节点的入边或出边的数量。

自注意力机制 (SAN) 是一种通过计算序列中不同位置的嵌入值,来增强序列表示的注意力机制,已成功应用于多个领域。通过对序列内相关性的精确建模,自注意力机制减少了对外部信息的依赖。在自然语言处理中,其主要通过计算单词之间的相互影响,过滤掉序列中不重要的信息,并赋予重要信息更大的权重以解决长距离依赖问题。为了学习具有代表性的会话特征,本文采用了自注意力机制对会话序列进行编码。

本文的自注意网络由一个注意网络和一个前馈网络组成。会话序列 $S = [v_{s,1}, v_{s,2}, v_{s,3}, \dots, v_{s,n}]$ 是注意网络的输入,注意函数 C 由查询和键值对输出的映射组成:

$$C = \text{softmax}\left(\frac{(SW^Q)(SW^K)}{\sqrt{d}}\right)(SW^V) \quad (6)$$

其中, $W^Q, W^K, W^V \in R^{d \times d}$ 为学习权值矩阵, \sqrt{d} 为防止内积计算值过大的归一化因子。

前馈网络由两层神经网络组成。选择 Relu 作为第一层激活函数,为自注意模块增加非线性。同时,自注意力网络中也可能出现传输损失,利用残差

连接可以充分利用底层信息,从而缓解传输损失的问题。前馈网络层公式如下:

$$F = H + \text{Relu}(CW_1 + b_4)W_2 + b_5 \quad (7)$$

其中, $W_1 \in R^{d \times 4d}$ 、 $W_2 \in R^{4d \times d}$ 是可以学习的权值矩阵, $b_4, b_5 \in R^d$ 是偏置向量。对于会话 S , $V_s = [v_{s,1}, v_{s,2}, v_{s,3}, \dots, v_{s,n}]$ 和 $F_s = [f_{s,1}, f_{s,2}, f_{s,3}, \dots, f_{s,n}]$ 分别由 GNN 和 SAN 模块编码后得到。然后将两种项目嵌入相结合,形成互补。链接函数描述如下。

$$O_s = V_s + \theta F_s \quad (8)$$

其中, θ 为权重参数,表示 SAN 模块学习到的项目嵌入的重要性,用于调整两个模块的比例。

1.2.2 会话嵌入层

虽然模型的项目嵌入层能够学习会话中项目之间的转换依赖关系,但其本身并不包含任何循环或卷积机制,从而无法充分考虑会话中项目的顺序性,也就是先前项目的位置^[8]。在会话推荐中,每个会话是一组有序项的序列组合,其中位置顺序编码考虑了句子或序列中不同元素之间相对位置的不同影响。为了加强学习到的会话嵌入的顺序性,本文提出的模型中通过位置矩阵 $P = [p_1, p_2, p_3, \dots, p_n]$,将反向位置信息纳入会话嵌入。其中, $p_i \in R^d$ 是位置 i 的位置向量。会话序列 $O = [o_{s,1}, o_{s,2}, o_{s,3}, \dots, o_{s,n}]$ 由项目嵌入层的联合函数输出。在位置信息中添加会话序列 O 后,第 i 项可表示为

$$z_i = \tanh(W_3[o_{s,i} \| p_{n-i+1}] + b_6) \quad (9)$$

其中, $W_3 \in R^{d \times 2d}$ 为可学习的权值矩阵; $b_6 \in R^d$ 为偏置向量; $\tanh(\cdot)$ 表示双元激活函数; $\|$ 表示拼接操作,即表示 $v_{o,i}$ 的项目与位置向量 p_{n-i+1} 连接,其中 n 为当前会话的长度。受 STAMP 和 NARM 的启发,会话嵌入融合了会话的长期和短期兴趣偏好,可以更好地反映当前会话的特征,其目的是强调当前偏好对适应用户复杂多变的交互行为的重要性。

利用软注意机制计算会话中项目的注意系数 α_i 与项目嵌入 z_i 的加权和,得到当前会话嵌入 h_c ,具体描述如下:

$$h_c = \sum_{i=1}^n \alpha_i z_i \quad (10)$$

$$\alpha_i = \mathbf{q}^T \sigma(W_4 z_i + W_5 z_n + W_6 z_{\text{avg}} + b_7) \quad (11)$$

$$z_{\text{avg}} = \frac{1}{n} \sum_{i=1}^n z_i \quad (12)$$

其中, $\sigma(\cdot)$ 是一个 sigmoid 函数, $W_4, W_5, W_6 \in R^{d \times d}$ 、 $\mathbf{q} \in R^d$ 是用于计算注意系数 α_i 的权重矩阵, $b_7 \in R^d$ 是偏置向量。目标嵌入 z_n 定义为输入会话序列的最后一项, z_{avg} 为会话中各项所表示的

平均值,表示会话的一个中心特征^[9]。

1.2.3 会话交互层

会话的目标嵌入和全局嵌入实际上只关注当前会话的特征,而忽略了会话之间相似而不同的特征,因此本文设计了会话交互层。模型受 FGNN^[10] 的启发,计算现有会话中相同交互项的共现度作为共现参数 ϕ_{ij} , 并集成图注意力网络进行特征传播,使得相似度高的会话可以相互学习对推荐结果有重要影响的特征。会话的最终嵌入,取决于会话本身及其邻近会话的组合。

在会话交互层,首先构造全局会话图,其中每个节点代表一个会话,使用会话项的共现参数 ϕ_{ij} 确定全局会话图中每个节点的连接。然后,使用会话嵌入层输出的当前会话嵌入 h_c 作为计算全局会话图中节点的内积及其共现参数,得到特征传递函数的初始输入以及节点与相邻节点之间的相似系数 e_{ij} 。最后,利用 softmax 函数对 e_{ij} 进行归一化,得到会话间注意系数 β_{ij} , 具体描述如下:

$$e_{ij} = \text{LeakyRelu}([h_i \odot h_j \parallel \phi_{ij}] \mathbf{W}_7) \mathbf{W}_8 \quad (13)$$

$$\beta_{ij} = \frac{\exp(e_{ij})}{\sum_{h_k \in N_{h_i}} \exp(e_{ik})} \quad (14)$$

式中: $\text{LeakyRelu}()$ 为非线性函数,其负斜率设为 0.2, $\mathbf{W}_7 \in R^{(d+1) \times d}$, $\mathbf{W}_8 \in R^d$ 是一个可学习的权重矩阵, \odot 是按元素相乘, N_{h_i} 表示 h_i 邻域会话的集合, h_i, h_j, h_k 是会话嵌入层的输出。注意感知邻域会话嵌入 h_i^s 如下所示:

$$h_i^s = \sum_{h_k \in N_{h_i}} \beta_{ij} h_j \quad (15)$$

最后,会话交互层将依赖于注意感知邻居会话嵌入 h_i^s 与当前会话嵌入 h_i 进行聚合。邻域聚合公式采用门控机制,自主选择更有意义的会话特征,具体描述如下:

$$\gamma = \sigma(\mathbf{W}_9 h_i + \mathbf{W}_{10} h_i^s) \quad (16)$$

$$h_f = \gamma h_i + (1 - \gamma) h_i^s \quad (17)$$

其中, $\mathbf{W}_9, \mathbf{W}_{10} \in R^{d \times d}$ 是一个可学习的权值矩阵; γ 是通过计算 h_i 和 h_i^s 的线性变换后的非线性激活函数得到的门; h_f 是全局嵌入会话。

1.2.4 预测层

在预测层,模型根据会话交互层获得的会话嵌入值 h_f , 评估会话中下一个被点击项目的概率。计算 v_i 与 h_f 的内积作为每个候选项目的推荐得分,并使用 softmax 函数获得最终的推荐概率:

$$\hat{y} = \text{softmax}(h_f^T \cdot \mathbf{v}_i) \quad (18)$$

其中, \mathbf{v}_i 是项目的初始向量, $\hat{y} \in R^m$ 表示候选项目是下一个被点击的项目的概率,在 \hat{y} 中选择前 k 个候选项目,将被选为推荐项目。为了训练模型,将模型的损失函数定义为预测值与真值的交叉熵:

$$\mathcal{L}(\tilde{y}) = - \sum_{i=1}^m y_i \log(\tilde{y}_i) + (1 - y_i) \log(1 - \tilde{y}_i) + \lambda \|\psi\|^2 \quad (19)$$

其中, y_i 表示下一个点击的真实值的独热编码向量; ψ 是所有可学习参数的集合; λ 是 L_2 正则化参数;模型使用时间反向传播(BPTT)方法训练。

2 实验结果分析

2.1 数据集

实验中使用的真实数据集为 Yoochoose、Diginetica 和天猫。Yoochoose 数据集来自 RecSys Challenge 2015,包含了6个月内电商网站的用户点击数据。本文采用 Yoochoose 数据集的 1/64 作为数据集,防止 Yoochoose 数据集过大导致内存溢出。Diginetica 的数据集来自 CIKM Cup 2016,本文仅使用其交易数据。天猫数据集来自 IJCAI-15 比赛数据集,其中包含了天猫购物平台上匿名用户的购物数据。在数据预处理中,所有长度为1的会话和在数据集中出现少于5次的项都被过滤掉,并按照时间顺序对数据集进行重新排序。对于 Yoochoose 数据集,最后一天的数据作为测试集,剩余数据作为训练集;对于 Diginetica 数据集,最近7天的数据作为测试集,剩余数据作为训练集;对于天猫数据集,最后100s作为测试集,其余数据作为训练集。对输入序列进行拆分,生成序列及对应的标签^[11]。对于会话 $S = [v_{s,1}, v_{s,2}, v_{s,3}, \dots, v_{s,k}]$, 通过序列预处理,生成序列及其对应的标签,即 $([v_{s,1}], v_{s,2}), ([v_{s,1}, v_{s,2}], v_{s,3}) \dots ([v_{s,1}, v_{s,2}, \dots, v_{s,k-1}], v_{s,k})$ 是这3个数据集的训练集和测试集。预处理数据集的统计数据见表1。

表1 数据集的统计数据

Table 1 Statistics of the dataset

| 数据集 | Yoochoose1/64 | Diginetica | 天猫 |
|------|---------------|------------|---------|
| 点击数 | 557 248 | 982 961 | 818 479 |
| 训练样本 | 369 859 | 719 470 | 351 268 |
| 测试样本 | 55 898 | 60 858 | 25 898 |
| 项目 | 16 766 | 43 097 | 40 728 |
| 平均长度 | 6.16 | 5.12 | 6.69 |

2.2 基准模型

为验证所提模型的有效性,与以下几种方法进

行了比较。基线法的描述如下:

(1) Item-KNN^[11] 使用余弦相似度定义会话之间的相似度,并推荐与之前点击相似的项目。

(2) FPMC^[4] 是一种基于马尔科夫链和矩阵分解的序列推荐方法。

(3) GRU4REC^[5] 使用 RNN 对基于会话的提议的用户序列进行建模。

(4) NARM^[12] 使用带有注意机制的 RNN 来捕获用户的主要目的和顺序行为,增加了注意机制来捕获用户在当前会话中的一般兴趣和当前兴趣。

(5) CSR^[13] 由一个内部和一个外部内存编码器组成,用于对当前会话和邻近会话的首选项进行建模,以便更好地预测当前会话的意图。

(6) SR-GNN^[6] 使用 GNN 捕获复杂项目信息转换,并使用注意机制捕获长期和短期利益。

(7) FGNN^[10] 通过加权注意层学习项目向量,通过会话图特征提取层学习会话特征。

(8) STAMP^[14] 使用一个注意力网络来捕获用户在当前会话中的一般兴趣和当前兴趣。

(9) CoSAN^[15] 使用自注意网络从邻居会话中学习当前会话嵌入和物品的协同特征信息。

2.3 参数设置

本文的实验参数设置如下:项目嵌入维数和训练批次大小设置为 100;会话交互层选择范围的共现参数 ϕ_{ij} 为 [0.0-0.9];对每个向量使用随机初始化,所有参数的平均值为 0,标准偏差为 0.1。采用交叉熵损失函数和小批量 Adam 算法对所有参数进行优化,初始学习率设置为 0.001,epoch 数设置为 30,每 3 个 epoch 后衰减为 0.1;采用 L2 正则化防止算法过拟合,L2 正则化系数选择范围为 $[1 \times 10^{-3}, 5 \times 10^{-4}, 1 \times 10^{-4}, 5 \times 10^{-5}, 1 \times 10^{-5}]$ 。为了公平的比较,基线算法的项目嵌入维数设置为 100,其他基本超参数遵循基线算法文献中显示的内容。GRU4REC 和 NARM 的隐藏单元个数设置为 100,CSR 的内存设置为 100,FGNN 的注意层数设置为 3,使用的多头机制设置为 8;CoSAN 邻居会话数设置为 5。

2.4 评价指标

本文选择了两个指标进行实验,以比较各种方法之间的效果。其中, $P@20$ (精确度) 被广泛用作预测精度的衡量标准,其代表了前 20 个项目中正确推荐项目的比例。 $MRR@20$ (平均倒数排名) 则计算正确推荐条目在排名列表中倒数排名的平均值, MRR 值越大表示在排名列表顶部的正确推荐越多。

为了验证模型的性能,本文设计了两组实验,并对实验结果进行了分析。第一组,比较了 SE-GNN 与 9 个基线模型的性能,第二组超参数实验,研究了共现参数 ϕ_{ij} 对模型性能的影响。

2.5 结果和分析

2.5.1 对比实验

实验使用 TensorFlow 和 PyTorch 框架,并使用 Python 语言模拟现有的 9 个基线模型和 SE-GNNRM,并对其性能进行比较和分析。对比结果见表 2。

表 2 对比实验结果

| 模型 | Yoochoose1/64 | | Diginetica | | 天猫 | |
|----------|---------------|----------|------------|----------|--------|----------|
| | $P@20$ | $MRR@20$ | $P@20$ | $MRR@20$ | $P@20$ | $MRR@20$ |
| FPMC | 45.42 | 15.03 | 26.48 | 6.95 | 16.11 | 7.32 |
| Item-KNN | 51.68 | 21.75 | 35.75 | 11.49 | 9.15 | 3.30 |
| GRU4REC | 60.84 | 22.69 | 30.79 | 15.46 | 10.83 | 5.89 |
| NARM | 68.52 | 28.62 | 48.46 | 16.21 | 23.32 | 10.68 |
| CSR | 71.25 | 30.29 | 50.52 | 16.38 | 29.46 | 13.96 |
| SR-GNN | 70.67 | 30.88 | 50.73 | 17.59 | 27.57 | 13.72 |
| FGNN | 71.32 | 31.57 | 51.32 | 18.39 | 25.26 | 10.42 |
| TAMP | 68.54 | 28.59 | 46.59 | 15.13 | 26.47 | 13.36 |
| CoSAN | 69.81 | 29.86 | 48.36 | 15.31 | 27.21 | 13.12 |
| SE-GNN | 72.53 | 32.24 | 52.76 | 18.46 | 32.68 | 15.54 |

从表 2 可以看出,本文提出的 SE-GNNRM 在 $P@20$ 和 $MRR@20$ 两个评价指标上表现最好。在 Yoochoose1/64 数据集上对次优算法的提升分别为 1.42% 和 1.58%,在 Diginetica 数据集上分别为 1.53% 和 0.1%,在天猫数据集上分别为 10.9% 和 11.8%。SE-GNNRM 有效地展示了其推荐三个数据集的能力。

在早期的推荐模型中,FPMC 模型限制了会话序列的依赖关系,推荐效果较差。Item-KNN 优于基于马尔科夫链的算法,但只推荐高度相似的物品,没有充分考虑会话中的顺序转换问题。与传统算法相比,基于深度学习的模型明确地模拟了用户的全局行为偏好,并考虑了用户上一个行为和下一次点击之间的过渡,有效地证明了其捕获会话序列的能力,并具有更强的顺序转换关系的能力。使用 RNN 建模会话序列的经典著作 GRU4REC 证明了用 GRU 单元编码当前会话比传统的 RNN 和 LSTM 更适合会话推荐问题,并将最后一层作为会话的一般偏好。但是,会话嵌入表示的兴趣首选项是可变的,并且 GRU4REC 不区分当前首选项和一般首选项。NARM 在 GRU4REC 的基础上很好地解决了这一问

题。在使用 GRU 单元对当前会话进行编码获得项目嵌入后,通过软注意机制对输入会话序列的不同部分进行动态选择和线性组合。实验结果也证明了该机制在会话偏好学习中的有效性。CSRM 还使用基于 GRU 的 RNN 对会话序列建模,同时还考虑会话的当前和一般偏好。此外,CSRM 设计了一个外部内存编码层,用于学习相邻会话的协作信息,使其具有比许多建模单个会话的模型更好的推荐性能。

在对比实验结果中,基于 GNN 的算法比基于 RNN 的算法更具竞争力。SR-GNN 通过将每个会话建模为有向图,进一步考虑会话中项目转换的依赖关系。会话中的项是图中的节点,其可以捕获会话中更复杂和隐式的依赖关系。实验结果表明,SR-GNN 虽然基于单会话建模,但其性能几乎与考虑多个会话的 CSRM 算法相当,证明使用 GNN 对会话序列建模是有意义的。FGNN 设计了权重图注意层学习会话图中的项目嵌入,并使用基于 GRU 的读出函数对会话嵌入进行抽象。FGNN 使用了多种深度学习技术,在 Yoochoose1/64 和 Diginetica 中都取得了相当不错的性能。

STAMP 是一种算法,其使用注意机制显式地表示会话的最后一项、会话的平均特征以及嵌入在会话中的项,以捕获会话的偏好。近年来,基于自我关注的会话推荐算法在许多领域都有突出的表现。CoSAN 利用序列数据较强的自注意编码能力,从项

目之间的相似度和位置信息的角度学习会话序列,结合相邻会话识别的协作信息,也得到了相当合适的结果。实验结果表明,该注意机制算法优于现有的一些深度学习算法,是处理会话推荐问题的有力选择。

对比而言,本文提出的 SE-GNNRM 算法同时使用 GNN 和自注意机制对会话序列进行编码,并设计了融合函数将两个会话向量进行融合,实现互补。同时,采用加入融合反向位置信息的软注意机制来获得合理的会话嵌入。在会话交互层中,引入了多会话协同信息。在构建的全局会话图上,通过图注意网络聚合邻居会话的偏好信息,使其更全面有效地处理会话序列信息。

综上所述,实验结果和分析证明了本文所提 SE-GNNRM 的合理性和有效性。

2.5.2 共现参数分析

本小节旨在展示 3 个数据集在 0~0.9 不同共现参数 ϕ_{ij} 对推荐性能的影响。在会话交互层,需要创建动态全局会话图,并根据共现参数确定节点之间的连接。只有达到同现参数阈值的节点之间存在边,这样才能根据全局会话图确认当前会话的邻居会话。因此,如何选择合适的共现参数需要通过实验进行验证和讨论。图 2 显示了在 3 个数据集中选择不同的共现参数时模型性能的变化。

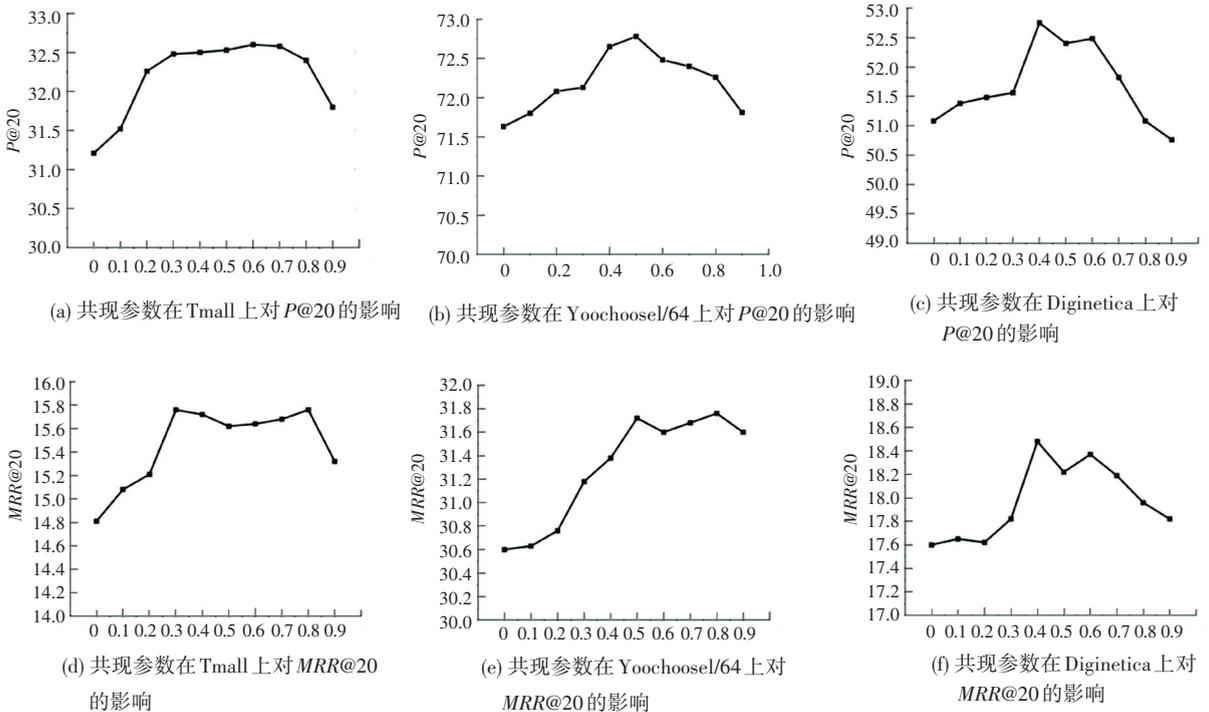


图 2 共现参数 ϕ_{ij} 对 3 种数据集的影响

Fig. 2 The effect of co-occurrence parameter ϕ_{ij} in three datasets

从图2可以看出,天猫数据集的趋势在0.3至0.8之间比较平缓。因此,模型最终选择了将天猫数据集的共现参数设为0.6。在Yoochoose1/64和Diginetica数据集上,分别在0.5和0.4处获得最佳结果。可以得出,3个数据集的最优值都趋向于区间的中间范围。这一结果可以解释为,如果邻居会话数选择过低,则可能会引入噪声影响推荐算法的性能;而如果选择过高,则可能无法完全学习相似邻居节点的特征,甚至可能无邻居节点可学习。

3 结束语

对于许多现有的会话推荐模型,更多地考虑会话内的顺序转换,忽略了相似会话对推荐性能的影响。本文提出了一种基于GNN和注意网络的会话推荐模型,即会话增强图神经网络(SE-GNNRM)。在3个真实数据集上的大量实验证明了该模型的有效性。但是,各项指标仍有着很大的进步空间。在未来的研究中,希望通过更高效的图表示学习和引入更有意义的辅助信息来提高模型的推荐性能。

参考文献

- [1] GAO C, WANG X, HE X, et al. Graph neural networks for recommender system [C]//Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining. IEEE, 2022: 1623-1625.
- [2] WANG S, CAO L, WANG Y, et al. A survey on session-based recommender systems[J]. ACM Computing Surveys (CSUR), 2021, 54(7): 1-38.
- [3] 郭栋. 基于会话序列的推荐算法研究[D]. 北京: 北京建筑大学, 2020.
- [4] 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. IEEE, 2010: 811-820.
- [5] HIDASI B, QUADRANA M, KARATZOGLOU A, et al. Parallel

- recurrent neural network architectures for feature - rich session - based recommendations [C]//Proceedings of the 10th ACM Conference on Recommender Systems. IEEE, 2016: 241-248.
- [6] WU S, TANG Y, ZHU Y, et al. Session-based recommendation with graph neural networks [C]//Proceedings of the AAAI Conference on Artificial Intelligence. IEEE, 2019: 346-353.
- [7] LI Y, TARLOW D, BROCKSCHMIDT M, et al. Gated graph sequence neural networks[J]. arXiv preprint arXiv:1511.05493, 2015.
- [8] KANG W C, MCAULEY J. Self-attentive sequential recommendation [C]//Proceedings of 2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018: 197-206.
- [9] DUAN H, ZHU J. Context-aware short-term interest first model for session-based recommendation[J]. arXiv preprint arXiv:2103.15514, 2021.
- [10] 邱瑞红,李晶晶,黄姿,等. 使用图神经网络重新思考基于会话的推荐中的项目顺序[C]//第28届信息与知识管理国际会议论文集. 2019: 579-588.
- [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. IEEE, 2016: 17-22.
- [12] 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. IEEE, 2017: 1419-1428.
- [13] 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. IEEE, 2019: 345-354.
- [14] 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 and Data Mining. IEEE, 2018: 1831-1839.
- [15] LUO A, ZHAO P, LIU Y, et al. Collaborative self - attention network for session-based recommendation [C]//Proceedings of the Twenty - Ninth International Joint Conference on Artificial Intelligence. IEEE, 2020: 2591-2597.