

基于图神经网络和时间注意力的会话序列推荐

孙鑫, 刘学军⁺, 李斌, 梁珂

(南京工业大学 计算机科学与技术学院, 江苏 南京 211816)

摘要: 为解决基于循环神经网络及其改进的方法在处理会话序列数据时只考虑序列行为, 无法从有限的点击中获得准确的会话向量表示的问题, 提出一种基于图神经网络和时间注意力的会话序列推荐算法。结合门控图神经网络和项目浏览时间信息, 有效建模会话中所有点击项目之间的复杂转换, 更充分利用用户浏览信息, 使会话向量表示的计算更准确、区分度更高。实验结果表明, 该方法能够提高推荐结果的准确性, 更为有效地预测用户的下一次点击。

关键词: 基于会话的推荐; 会话图; 门控图神经网络; 注意力机制; 时间注意力因子

中图法分类号: TP301.6 **文献标识号:** A **文章编号:** 1000-7024 (2020) 10-2913-08

doi: 10.16208/j.issn1000-7024.2020.10.036

Graph neural networks with time attention mechanism for session-based recommendations

SUN Xin, LIU Xue-jun⁺, LI Bin, LIANG Ke

(School of Computer Science and Technology, Nanjing Tech University, Nanjing 211816, China)

Abstract: To solve the problem that the recurrent neural network and its improved methods only consider the sequence behavior when processing the sequence data, and cannot obtain the accurate session vector representation from the limited click, the graph neural networks with time attention mechanism for session-based recommendations algorithm was proposed. Gated graph neural network and the item browsing time information were used to model the complex conversion between all the click items in the session effectively and make more full use of user browsing information. The calculation of the session vector representation was then more accurate and discriminative. Experimental results show that the proposed method can improve the accuracy of the recommendation results and predict the user's next click more effectively.

Key words: session-based recommendation; session graph; gated graph neural networks; attention mechanism; time attention factors

0 引言

在基于会话的推荐问题中, 经典的协同过滤和矩阵分解^[1,2]方法并不适用。近年来, 大多数研究将循环神经网络(recurrent neural network, RNN)应用于基于会话的推荐系统中^[3-6], 取得了良好的效果。然而, 这些算法仍然存在以下问题: ①只使用用户点击序列对连续项目之间的单向转换建模, 忽略了会话中的其它项目, 即忽略了远距离项目之间的复杂转换。②忽略项目的浏览时间信息。人们普遍认为, 用户倾向于花更多时间在他们感兴趣的项目上,

而这些感兴趣的项目总是与用户当前的目标密切相关。因此, 项目的浏览时间是会话序列推荐中的一个重要特征。③在基于会话的推荐中, 会话大多是匿名的, 而且数量众多, 再加上在会话点击中涉及的用户行为通常是有限的, 因此很难准确地估计每个会话的向量表示。为了解决上述问题, 提出了一种基于图神经网络和时间注意力机制的会话序列推荐方法, 称为 GNN-TA-SR (graph neural networks with time attention mechanism for session-based recommendations)。该方法利用门控图神经网络, 建模项目之间的复杂转换并得到项目隐含向量, 然后, 基于得到的

收稿日期: 2019-07-31; 修订日期: 2019-10-23

基金项目: 国家重点研发计划基金项目 (2018YFC0808505、2017YFC0805605); 江苏省重点研发计划基金项目 (BE2017617)

作者简介: 孙鑫 (1993-), 女, 山东泰安人, 硕士研究生, 研究方向为深度学习、推荐系统等; 通讯作者: 刘学军 (1970-), 男, 吉林长春人, 博士, 教授, CCF 高级会员, 研究方向为数据库、数据挖掘、传感器网络、推荐系统等; 李斌 (1979-), 男, 江苏南京人, 讲师, 研究方向为传感器网络、智能信息处理等; 梁珂 (1993-), 男, 浙江绍兴人, 本科, 研究方向为机器学习、计算机应用等。

E-mail: 209743962@qq.com

项目隐含向量,利用一个时间注意力网络生成准确的会话向量表示。其中,时间注意力因子(time attention factors, TAF)是根据用户浏览过程中项目的持续时间信息创造性地计算出来的。

1 相关工作

1.1 传统的推荐方法

传统的基于会话的推荐方法主要包括基于项目的邻域方法和基于马尔可夫链的序列方法。

基于项目的邻域方法在会话期间根据项目的相似性矩阵来向用户推荐与当前点击的项目最相似的项目,其中项目相似性是根据同一会话中的共现计算的,即在会话中经常一起点击的项目被认为是相似的。这些方法难以考虑项目的顺序并仅基于最后点击生成预测。

然后,提出了基于马尔可夫链的序列方法,该方法基于先前的点击预测用户的下一步行为。基于马尔可夫链的序列方法将推荐生成作为序列优化问题处理,文献[7]采用马尔可夫决策过程(MDPs)作为解决方案。文献[8]基于马尔可夫链提出了一种个性化的序列推荐模型,将矩阵分解和马尔可夫链方法结合起来,同时适应长期和短期动态。这些方法的主要问题是,当试图包括用户所有可能选择的序列时,状态空间很快变得难以管理。

1.2 基于深度学习的方法

深度神经网络最近被验证在建模序列数据方面非常有效^[9]。受自然语言处理领域最新研究进展的启发,已经开发了一些基于深度学习的解决方案。Hidasi等^[3]首先提出了一种循环神经网络方法,然后通过数据增强技术和考虑用户行为的时间转换来改进模型^[4]。Li等^[6]提出一种基于RNN的编码器-解码器模型(NARM),它来自RNN的最后隐藏状态作为序列行为,并使用先前点击的隐藏状态进行注意力计算,以捕获给定会话中的主要目的。Jannach和Ludewig^[10]将循环方法和基于邻域的方法结合在一起,以混合顺序模式和共现信号。Tuan和Phuong^[5]将会话点击与内容特征(例如项目描述和项目类别)结合起来,通过使用三维卷积神经网络生成推荐。Liu等^[11]提出了一个短期注意优先模型(STAMP),通过采用简单的MLP网络和注意力网络来捕捉用户的一般兴趣和当前兴趣。

近年来,图神经网络(graph neural network, GNN)在社交网络、知识图谱、推荐系统,甚至生命科学等各个领域越来越受欢迎^[12-15],它是一种直接在图结构上运行的神经网络。门控GNN^[16]是GNN的一种改进,它使用门控循环单元,通过BPTT(back-propagation through time)计算梯度。最近,GNN被用于建模自然语言处理和计算机视觉应用的图形结构依赖性,例如,脚本事件预测^[17]、情境识别^[18]和图像分类^[19]。图神经网络能够在考虑丰富节点连接的情况下自动提取会话图的特征,非常适合基于会话

的推荐。

2 模型框架

2.1 公式化描述

基于会话的推荐算法的任务是通过给定用户的当前点击序列,预测用户接下来可能点击的项目。在基于会话的推荐中,设会话序列 $s = [v_1, v_2, \dots, v_n]$ ($v_i \in V, 1 \leq i \leq n$)是按时间戳排序的用户点击项目列表, $V = \{v_1, v_2, \dots, v_m\}$ 为所有会话中涉及的项目总空间。基于会话的推荐问题可以表述为基于序列的预测问题,构建一个模型,使得对于给定的会话点击序列作为输入,通过模型能够得到输出 \hat{y} ,其中 $\hat{y} = \{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{m-1}, \hat{y}_m\}$,即在基于会话的推荐模型下,对于会话 s ,模型输出所有可能项目的概率。 \hat{y} 可以看作是一个排序列表,包括可能会出现在输入会话后的下一个点击项目的所有可能的情况, \hat{y}_j ($1 \leq j \leq m$)对应于项目 j 在此会话中的推荐得分。 \hat{y} 中排名最高的前 K 个项目将是推荐的候选项目。

2.2 总体框架

对于基于会话的推荐,首先从历史会话序列构造有向图。基于会话图,GNN能够捕获项目的转换并相应地生成精确的项目隐含向量,这是以前的序列方法难以做到的,如基于马尔可夫链和基于RNN的方法。基于准确的项目隐含向量,所提出的GNN-TA-SR可以构造更可靠的会话表示,并且可以推断出下一次点击项目。

图1为所提出的GNN-TA-SR方法的总体框架。首先,所有会话序列都被建模为有向会话图 $G = (V, \epsilon)$, V 是节点的集合,由项目空间所有且唯一的项目组成,一个项目就是有向图中的一个节点 v , ϵ 是有向边的集合, $v \rightarrow v' \in \epsilon$ 是有向边,表示用户在会话 s 中点击项目 v 之后又点击了项目 v' 。其中每个会话序列可以被视为子图,如图1会话图中虚线框部分所表示的会话序列 $s = [v_3, v_1, v_3, v_5, v_7]$ 。然后,依次对每个会话图进行处理,通过门控图神经网络得到每个图中涉及的所有节点的隐含向量。之后,将每个会话表示为全局偏好和该会话中用户的当前兴趣的组合,其中全局偏好向量和用户当前兴趣均由节点的隐含向量组成。最后,对于每个会话,模型预测每个项目成为下一个点击的概率。

2.3 在会话图上学习项目隐含向量

每个会话序列 $s = [v_1, v_2, \dots, v_m]$ 都可以被视为有向图 G 的子图 $G_s = (V_s, \epsilon_s)$,在此会话图中,每个节点代表一个项目 $v_i \in V$ 。每条边 $(v_i, v_{i+1}) \in \epsilon_s$ 表示用户在会话 s 中的 v_i 之后点击项目 v_{i+1} 。由于会话序列中可能出现多个重复点击的项目,所以需要给每条边分配一个归一化加权值,该值被计算为该边的出现次数除以该边的起始节点的出度。将每个项目 $v \in V$ 映射到一个统一的嵌入空间中,节点向

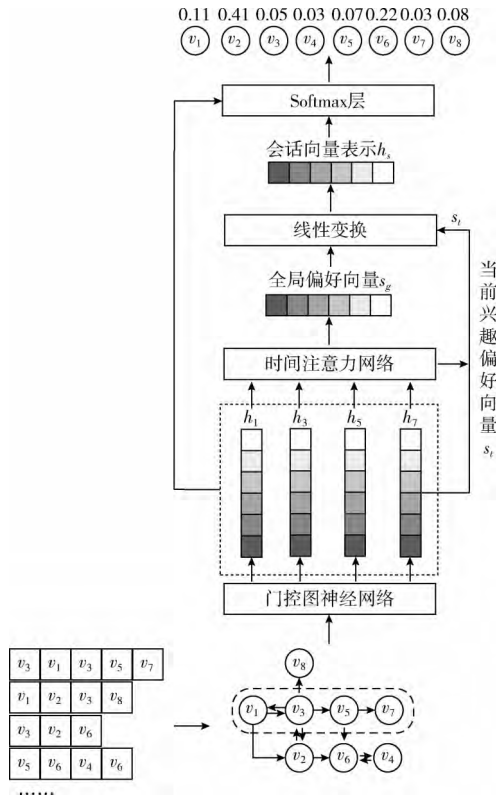
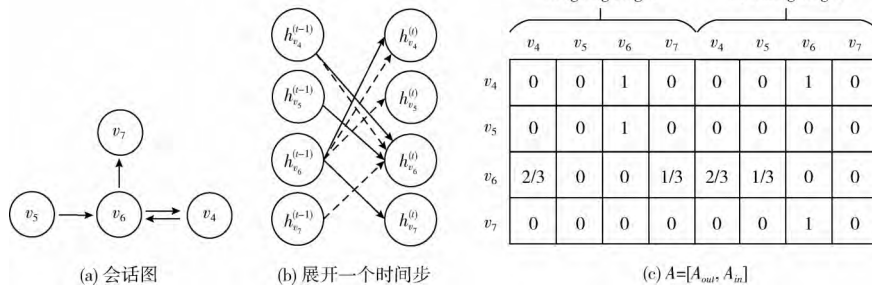


图 1 GNN-TA-SR 的总体框架

量 $h_v \in \mathbb{R}^d$ 表示通过图神经网络学习的项目 v 的隐含向量, 其中 d 是维度。

图 2 会话 $s = [v_5, v_6, v_4, v_6, v_4, v_6, v_7]$

2.4 计算时间注意力因子

在浏览过程中, 用户总是倾向于在感兴趣的项目上花费更多的时间, 而对于不喜欢的项目或者无意间点击的项目, 他们可能只会浏览很短的时间。此外, 用户的下一个浏览项目总是与用户当前的兴趣相关联。因此, 我们使用项目浏览时间来计算时间注意力因子, 它可以反映之前点击的项目与用户当前兴趣的联系。这些计算出的注意力因子将被添加到时间注意力网络中用于生成会话向量表示。图 3 显示了时间注意力因子的计算过程。计算过程分为 4 个步骤: 项目浏览时间的计算、设置浏览时间的阈值、 z 分数变换和归一化操作。

具体地, 对于图 G_s 的节点 v_i , 节点向量的学习更新函数如下

$$a_i^{(t)} = A_{i,:} [h_{v_1}^{(t-1)}, \dots, h_{v_n}^{(t-1)}]^T H + b \quad (1)$$

$$z_i' = \sigma(W_z a_i^{(t)} + U_z h_{v_i}^{(t-1)}) \quad (2)$$

$$r_i' = \sigma(W_r a_i^{(t)} + U_r h_{v_i}^{(t-1)}) \quad (3)$$

$$\tilde{h}_{v_i}^{(t)} = \tanh(W a_i^{(t)} + U(r_i' \odot h_{v_i}^{(t-1)})) \quad (4)$$

$$h_{v_i}^{(t)} = (1 - z_i') \odot h_{v_i}^{(t-1)} + z_i' \odot \tilde{h}_{v_i}^{(t)} \quad (5)$$

其中, $H \in \mathbb{R}^{d \times 2d}$ 控制权重, h_{v_i} 表示节点 v_i 的隐含向量, 矩阵 $A \in \mathbb{R}^{n \times 2n}$ 决定图中的节点如何相互通信。以会话序列 $s = [v_5, v_6, v_4, v_6, v_4, v_6, v_7]$ 为例, 如图 2 所示, 图 (a) 是会话 s 对应的会话图, 图 (b) 是展开的一个时间步, 实线边对应于图 (a) 中的有向边, 虚线边对应于有向边的反向边, 每条边的权重分别对应于图 (c) 矩阵 A 中的非零参数。 A 定义为两个邻接矩阵 A_{out} 和 A_{in} 的串联, 它们分别表示会话图中输出和输入边的加权连接。 $A_{i,:} \in \mathbb{R}^{1 \times 2n}$ 是矩阵 A 中对应于节点 v_i 的一行。对于每个会话图 G_s , 门控图神经网络同时处理节点。在矩阵 A 给出的限制条件下, 等式 (1) 是在图的不同节点之间通过传入和传出的边传递信息的步骤, 式 (2)~式 (5) 类似于门控循环单元 (gated recurrent unit, GRU) 的更新, 它们包含来自其它节点和前一个时间步的信息, 以更新每个节点的隐藏状态。其中, z 和 r 是更新门和重置门, $\sigma(\cdot)$ 是 sigmoid 函数, \odot 是点乘运算符。通过对会话图中所有节点进行更新直至收敛, 可以获得最终的节点向量。

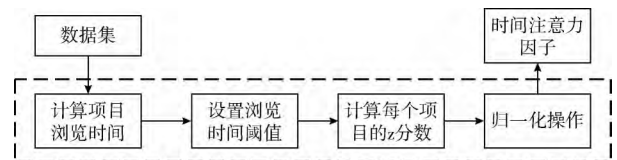


图 3 时间注意力因子的计算过程

从数据集中每个事件的时间戳, 可以获得每个项目的浏览时间。图 4 显示了 RecSys Challenge 2015 数据集中的项目浏览时间分布, 只有约 1% 项目的浏览时间超过 1500 s。这些长时间的浏览可能是由于浏览行为被某些东西打断, 并

且在他(或她)离开之前浏览页面没有关闭。为了避免异常的长时间浏览的影响,应该设置一个时间阈值 T_{th} 。假设在正常情况下,每个项目的浏览行为都可以在 T_{th} 时间范围内完成,如果用户对一个项目的浏览时间超过该阈值,则视为异常行为并设置浏览时间为 T_{th} 。 T_{th} 的具体取值取决于实际应用的需要,根据 RecSys Challenge 2015 数据集的特点,下面的分析将 T_{th} 设置为 1500 s。

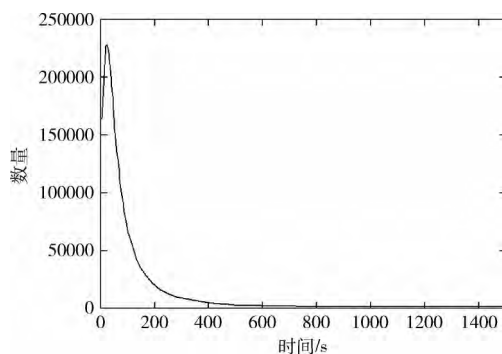


图4 RecSys Challenge 2015 数据集中的项目浏览时间分布

在 GNN-TA-SR 模型中,我们不关心项目浏览时间的绝对值,而关心的是每个项目浏览时间所占的比例。例如,在会话 s_1 中,用户在项目 v_1 、 v_2 、 v_3 上的浏览时间分别为 30 s、60 s 和 180 s,而在另一条会话 s_2 中,用户在 v_1 、 v_2 、 v_3 上的浏览时间分别为 250 s、168 s 和 180 s。显然,在第一条会话中用户喜欢 v_3 的程度要大于在第二条会话中用户喜欢 v_3 的程度,尽管他们在 v_3 上的浏览时间相同。如果直接使用项目的浏览时间作为它们的注意力因子,系统可能会错误地认为项目 v_3 对会话 s_1 和会话 s_2 同等重要。为了避免这个错误,更准确地衡量会话序列中每个项目的重要性,使用项目的浏览时间信息和它所属会话的全局信息计算项目的 z 分数。 z 分数值定义如下

$$z_i = \frac{t_i - \bar{t}}{\sigma} \quad (6)$$

其中, z_i 表示会话序列 $s = [v_1, v_2, \dots, v_n]$ 中项目 v_i 的 z 分数, t_i 表示 v_i 的浏览时间, \bar{t} 是项目所属会话中所有项目浏览时间的平均值, σ 是会话序列项目浏览时间的标准差。 z_i 的绝对值表示以标准差 σ 为单位的 t_i 和 \bar{t} 之间的距离,如果计算的 z_i 值为负,则表示用户对此项目的关注程度低于平均水平,如果 z_i 值为正,则表示用户对此项目的兴趣高于其它项目。 z_i 值越大,用户对此项目的兴趣就越大。

为了使每条会话的兴趣表示保持一致,通过使用 sigmoid 函数将计算得到的 z 分数进行归一化。项目的时间注意力因子由下式表示

$$\alpha_{ti} = \frac{1}{1 + e^{-z_i}} \quad (7)$$

2.5 生成会话向量表示

基于节点向量,每个会话 s 可以表示为一个嵌入向量

h_s , 该向量由该会话中涉及的节点向量直接表示。为了更好地预测用户的下一次点击,我们制定了几种策略以结合会话的全局(长期)偏好和当前兴趣,并将此混合向量用作会话向量。

GNN-SR-AVE: 将会话中点击的项目视为同等重要,忽略浏览时间差异, s_g 表示用户在当前会话的长期偏好,被定义为节点向量的平均值, s_t 表示该会话中用户的当前兴趣,在该方法中,将最后一次点击 v_n 视为的用户当前兴趣

$$s_g = \frac{1}{n} \sum_{i=1}^n h_{v_i} \quad (8)$$

$$s_t = h_{v_n} \quad (9)$$

GNN-SR-TAF: 考虑在一个会话中,用户对每个项目的喜爱程度不同,或者点击某个项目是属于意外点击,这些情况大都可以根据项目浏览时间反映出来,因此,引入时间注意力因子来计算用户在当前会话的长期偏好和当前兴趣

$$s_g = \sum_{i=1}^n \alpha_{ti} h_{v_i} \quad (10)$$

$$s_t = \alpha_{tn} h_{v_n} \quad (11)$$

GNN-SR-TAF₀: 为了控制变量,对比在计算当前会话的长期偏好和当前兴趣引入时间注意力因子对推荐性能的影响,令

$$s_g = \sum_{i=1}^n \alpha_{ti} h_{v_i} \quad (12)$$

$$s_t = h_{v_n} \quad (13)$$

GNN-TA-SR: 考虑会话中每个点击的项目对整条会话和最后一次点击的贡献度不同,给每个项目赋予不同的权重

$$\alpha_i = q^T \sigma(W_1 h_{v_i} + W_2 h_{v_n} + W_3 s_{g'} + c) \quad (14)$$

其中,令

$$s_{g'} = \sum_{i=1}^n \alpha_{ti} h_{v_i} \quad (15)$$

从等式(14)可以看出,会话序列中每个项目的注意力系数是基于目标项目 v_i 、最后点击 v_n 和会话全局偏好的初步表示 $s_{g'}$ 来计算的。

则用户在当前会话的全局偏好

$$s_g = \sum_{i=1}^n \alpha_i h_{v_i} \quad (16)$$

用户的当前兴趣

$$s_t = \alpha_n h_{v_n} \quad (17)$$

其中,参数 $q \in \mathbb{R}^d$ 、 $W_1, W_2, W_3 \in \mathbb{R}^{d \times d}$ 用于控制项目向量的权重。

最后,以上3种策略均通过 s_g 和 s_t 向量的串联进行线性变换来计算混合向量 h_s

$$h_s = W_4 [s_g; s_t] \quad (18)$$

其中,矩阵 $W_4 \in \mathbb{R}^{d \times 2d}$ 将两个组合向量映射到潜在空间

\mathbb{R}^d 中。

2.6 生成推荐

在获得每个会话的向量表示之后, 通过将项目向量 h_{v_i} 乘以会话向量 h_s 来计算每个候选项目 v_i 的得分 \hat{r}_i , 其可以被定义为

$$\hat{r}_i = h_s^T h_{v_i} \quad (19)$$

然后通过 softmax 层来获得模型的输出向量 \hat{y}

$$\hat{y} = \text{softmax}(\hat{r}) \quad (20)$$

其中, $\hat{r} \in \mathbb{R}^m$ 表示所有候选项目的推荐分数, $\hat{y} \in \mathbb{R}^m$ 表示节点成为会话 s 中的下一次点击的概率。

对于每个会话图, 损失函数定义为预测值与真实值的交叉熵, 然后使用 BPTT 算法来训练模型

$$L(\hat{y}) = - \sum_{i=1}^m y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \quad (21)$$

其中, y 是真实的会话序列中下一个点击项目的 one-hot 向量。注意, 在基于会话的推荐场景中, 大多数会话的长度相对较短。因此, 建议选择相对较少的训练步骤来防止过拟合。

3 实验与分析

在本节中, 首先描述了实验中使用的数据集, 数据预处理和评价指标。然后, 将提出的 GNN-TA-SR 与其它方法进行了比较。最后在不同的实验设置下对 GNN-TA-SR 进行了详细的分析。

3.1 数据集和数据预处理

GNN-TA-SR 模型在 RecSys Challenge 2015 发布的公开数据集 Yoochoose 中进行训练和评估, 该数据集包括从电子商务网站收集的 6 个月的点击流, 其中训练集仅包含会话事件。从文献[20]中分析, 该数据集中约 90% 的会话有 7 个或更少的事件, 最长的会话包含 262 个事件。所有项目的平均交互次数为 23, 其中最受欢迎的项目在数据集中被点击 162 622 次。

首先, 从 Yoochoose 数据集中过滤掉长度为 1 的会话和点击次数小于 5 次的项目, 并划分出 Yoochoose 数据集最后一天的会话数据作为测试集, 之前的会话数据分别作为训练集, 然后从测试集中删除那些不曾出现在训练集中的项目。最终, Yoochoose 数据集保留了 7 966 257 条会话和 37 483 个项目, 共计 31 637 239 次点击。

为了公平比较, 遵循文献 [4] 对会话序列进行拆分处理, 对于会话 $s = [v_1, v_2, \dots, v_n]$, 拆分成序列和相应的标签 $([v_1], v_2)$, $([v_1, v_2], v_3)$, \dots , $([v_1, v_2, \dots, v_{n-1}], v_n)$ 。因为 Yoochoose 训练集非常大, 并且根据文献 [6] 的实验, 在距离测试集较近的数据的训练会产生比在整个数据集的训练更好的结果, 因此, 实验中使用距离测试集时间最近的 1/64 和 1/4 的训练序列。两个数据集的统计数据见表 1。

表 1 实验数据集统计

数据集	Yoochoose 1/64	Yoochoose 1/4
训练集	375 073	5 969 416
测试集	55 898	55 898
点击数	565 552	7 980 529
项目数	17 694	30 660
平均长度	6.16	5.71

3.2 评价指标

P @ 20 (Precision) 被广泛用作基于会话的推荐领域中预测准确度的度量。它代表前 20 项中正确推荐的项目的比例。

MRR @ 20 (mean reciprocal rank, MRR) 是正确推荐项目的倒数排名的平均值。当排名超过 20 时, 倒数排名被设置为 0。MRR 度量是范围 $[0, 1]$ 的归一化分数, 考虑推荐排名的顺序, 其中较大的 MRR 值表示正确的推荐位于排名列表的顶部, 这表明相应推荐系统的性能更好。

3.3 参数设置

通过对所有数据集进行广泛的网格搜索来优化超参数, 并且通过基于验证集上的 P @ 20 得分的早期停止来选择最佳模型, 验证集是训练集的随机 10% 子集。网格搜索的超参数范围如下: 隐含向量维度 d : $\{50, 100, 150, 200\}$, 学习率 η : $\{0.001, 0.005, 0.01, 0.1, 1\}$ 。根据平均性能, 在本研究中, 两个数据集隐含向量的维度 d 设置为 100。所有参数都使用均值为 0、标准差为 0.1 的高斯分布初始化, 并使用小批量 Adam 优化器对这些参数进行优化, 其中初始学习率 η 设置为 0.001, 每 3 个周期后衰减 0.1。此外, 批量大小和 L2 正则化参数分别设置为 100 和 10^{-5} 。

3.4 与相关方法比较

为了验证所提出算法的整体性能, 将 GNN-TA-SR 算法同以下 8 种现有的解决会话型推荐问题的算法——POP、S-POP、Item-KNN、FPMC、GRU4Rec、GRU4Rec+、NARM、STAMP^[7]进行了比较:

POP 和 S-POP 分别推荐训练集和当前会话中的前 N 个频繁项目。

Item-KNN 推荐与会话中先前点击的项目类似的项目, 其中相似性被定义为会话向量之间的余弦相似度。

FPMC 是一种基于马尔可夫链的序列预测方法。

GRU4Rec^[3]使用 RNN 对基于会话的推荐的用户序列进行建模。

GRU4Rec+^[4]进一步研究了 RNN 在基于会话推荐领域的应用, 提出了一种数据增强技术, 并改变输入数据的数据分布, 用于改进模型的性能。

NARM^[6]采用具有注意机制的 RNN 来捕获用户的主要目的和序列行为。

STAMP^[11]捕捉用户当前会话的一般兴趣和最后一次点击的当前兴趣。

表 2 展示了所有算法在两个数据集上 P@20 和 MRR@20 的总体性能。由于内存不足,无法初始化 FPMC,所以没有报告 Yoochoose 1/4 的性能^[6]。

表 2 实验结果对比

方法	Yoochoose1/64		Yoochoose1/4	
	P@20	MRR@20	P@20	MRR@20
POP	6.71	1.65	1.33	0.30
S-POP	30.44	18.35	27.08	17.75
Item-KNN	51.60	21.81	52.31	21.70
FPMC	45.62	15.01	—	—
GRU4Rec	60.64	22.89	59.53	22.60
GRU4Rec+	67.84	29.00	69.11	29.22
NARM	68.32	28.76	69.73	29.23
STAMP	68.74	29.67	70.44	30.00
GNN-SR-AVE	67.92	28.71	69.33	29.13
GNN-SR-TAF	68.36	29.10	69.75	29.22
GNN-SR-TAF ₀	68.29	29.11	69.71	29.03
GNN-TA-SR	70.86	31.42	71.53	31.87

基于图神经网络的会话序列推荐算法将分离的会话序列聚合到图结构数据中,通过门控图神经网络得到项目隐含向量。从表 2 可以看出,GNN-TA-SR 在 Yoochoose1/64 和 Yoochoose1/4 数据集上以 P@20 和 MRR@20 实现了最先进的性能,验证了所提模型的有效性。在该模型中,考虑了项目之间的复杂转换并利用了项目浏览时间信息,综合考虑了会话的全局偏好和当前兴趣。在本文所提出的 4 种混合策略中,GNN-TA-SR 性能最佳,GNN-SR-TAF 次之,GNN-SR-TAF₀ 又优于 GNN-SR-AVE,这表明将会话中的所有项目视为同等重要是不合理的,通过添加时间注意力因子,以及计算每个项目对整条会话和最近点击项目的贡献度来得到会话的全局偏好可以提高推荐性能。此外,GNN-SR-TAF₀ 略优于 GNN-SR-AVE 表明,在当前兴趣 s_t 引入时间注意力因子,对最后点击的项目赋予一个权重,也可以提高推荐性能。

对于 POP、SPOP、Item-KNN 和 FPMC 等传统方法,其性能相对较差。这种简单的模型仅基于重复的共同点击的项目或连续的项目进行推荐,这在基于会话的推荐场景中是有问题的。即便如此,S-POP 仍然优于 POP 和 FPMC 等,验证了会话上下文信息的重要性。与基于马尔可夫链的 FPMC 相比,Item-KNN 具有更好的性能。Item-KNN 仅利用项目之间的相似性而不考虑序列信息,这表明传统的基于马尔可夫链的方法主要依赖的连续项目的独立性假设

是不现实的。此外,这样的全局解决方案可能耗费时间和内存,使得它们无法扩展到大规模数据集。

所有的神经网络基线都明显优于传统模型,从而验证了深度学习技术在该领域的有效性。GRU4Rec+ 通过使用将单个会话拆分为多个子会话的数据增强技术来进行训练,以提高 GRU4Rec 的性能。虽然 GRU4Rec+ 不修改 GRU4Rec 的模型结构,但它们都只考虑了序列行为,这可能会遇到用户兴趣漂移的困难。NARM 在基线中实现了比较好的性能,因为它不仅使用具有 GRU 单元的 RNN 来建模序列行为,而且还使用注意力机制来捕获主要目的,这表明主要目的信息在推荐中的重要性。这是合理的,因为当前会话中的项目的一部分可能反映用户的主要目的并且与下一个项目相关。而 STAMP 通过使用最后点击的项目来改进短期内存。这些方法显式地对用户的全局行为偏好进行建模,并考虑用户以前的动作和下一次点击之间的转换,从而获得优于这些传统方法的性能。然而,它们的性能仍然低于所提出的方法。与 NARM 和 STAMP 等最先进的方法相比,GNN-TA-SR 进一步考虑了会话序列项目之间的复杂转换,从而将每个会话建模为一个图,它可以捕获用户点击之间更复杂和隐式的连接。而在 NARM 和 GRU4Rec 中,它们显式地为每个用户建模,并通过分离的会话序列获得用户表示,而忽略了项目之间可能的交互关系。因此,本文所提出的模型对会话行为的建模能力更强。

此外,GNN-TA-SR 采用时间注意力机制生成会话向量表示,该会话向量可以自动选择最重要的项目转换,并忽略当前会话中的噪声和无效的用户动作。相反,STAMP 仅使用最后点击的项目和先前的动作之间的转换,这可能是不够的。其它的 RNN 模型,如 GRU4Rec 和 NARM,在传播过程中也不能选择有效的信息。它们使用先前所有的项目来获得一个表示用户一般兴趣的向量。当用户的行为漫无目的,或者他的兴趣在当前会话中快速变化时,传统的模型无法处理嘈杂的会话。

3.5 会话向量不同组成部分之间的比较

为了分析会话向量的不同组成部分对推荐性能的影响,将 GNN-TA-SR 与以下 4 种方法进行了比较:①只考虑用户的当前兴趣(GNN-SR-L),即只考虑用户最后点击的项目;②只考虑平均池化的全局偏好向量(GNN-SR-AVEG);③引入时间注意力因子的全局偏好向量(GNN-SR-TAFG);④通过时间注意力网络得到的全局偏好向量(GNN-TA-SRG),即 GNN-TA-SR 的全局偏好向量。图 5 给出了对比结果。

从图 5 中可以看出,混合向量方法 GNN-TA-SR 在两个数据集上都获得了最佳结果,这证实了将当前会话兴趣与长期偏好相结合的重要性。图 5 中还显示了在两个数据集上带时间注意力因子的 GNN-SR-TAFG 的性能明显优于平均池化的 GNN-SR-AVEG,这表明考虑项目浏览时间是

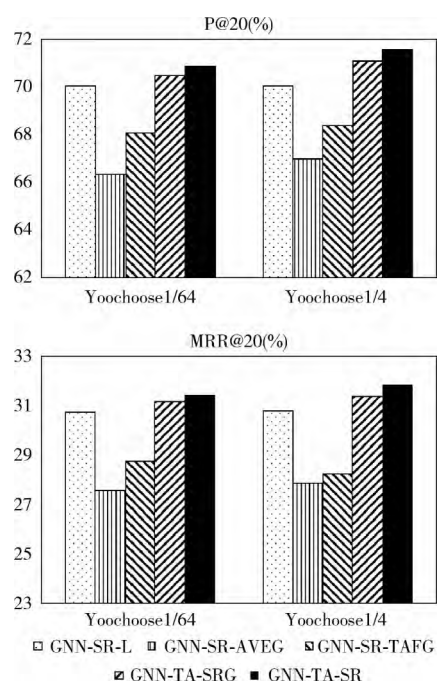


图5 会话向量不同组成部分之间的比较

有效的。项目浏览时间在一定程度上反映了不同项目在会话中的重要性不同,引入时间注意力因子有利于识别会话中用户真正喜欢的项目,也可以在一定程度上弱化意外点击等噪声行为对获取会话向量表示的影响。此外,GNN-TA-SRG的性能优于GNN-SR-AVEG,表明GNN-TA-SRG更有助于从会话数据中提取重要行为来构建长期偏好;GNN-SR-L的性能优于GNN-SR-AVEG和GNN-SR-TAFG,并且性能仅次于GNN-TA-SRG,这表明当前兴趣和长期偏好对于基于会话的推荐都是至关重要的。

4 结束语

在用户偏好和历史记录难以获取的情况下,基于会话的推荐是必不可少的。本文提出了一种基于图神经网络和时间注意力机制的会话序列推荐算法,不仅考虑了会话序列项目之间的复杂结构和转换,将每个会话建模为一个有向图来捕获用户点击之间更复杂和隐式的连接,而且考虑项目的浏览时间信息,提出了一种结合会话的长期偏好和当前兴趣的策略,以更好地预测用户的下一步动作。综合实验结果表明,所提出的算法优于其它最先进的方法。

参考文献:

[1] GU Huan, YANG Changchun, WU Yun, et al. Collaborative filtering recommendation algorithm combining community structure and personal interests [J]. Computer Engineering and Design, 2018, 39 (11): 128-132 (in Chinese). [顾寰, 杨长

春, 吴云, 等. 融合社区结构和个人兴趣的协同过滤推荐算法 [J]. 计算机工程与设计, 2018, 39 (11): 128-132.]

- [2] DING Shaohong, JI Donghong, WANG Lulu. Collaborative filtering recommendation algorithm based on user attributes and scores [J]. Computer Engineering and Design, 2015, 36 (2): 487-491 (in Chinese). [丁少衡, 姬东鸿, 王路路. 基于用户属性和评分的协同过滤推荐算法 [J]. 计算机工程与设计, 2015, 36 (2): 487-491.]
- [3] Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based recommendations with recurrent neural networks [J]. arXiv Preprint arXiv: 1511.06939, 2015.
- [4] Tan Y K, Xu X, Liu Y. Improved recurrent neural networks for session-based recommendations [C] //Proceedings of DLRS. Boston: ACM, 2016: 17-22.
- [5] Tuan T X, Phuong T M. 3D convolutional networks for session-based recommendation with content features [C] //Proceedings of the Eleventh ACM Conference on Recommender Systems. Como: ACM, 2017: 138-146.
- [6] Li Jing, Ren Pengjie, Chen Zhumin, et al. Neural attentive session-based recommendation [C] //Proceedings of ACM CIKM. Singapore: ACM, 2017: 1419-1428.
- [7] Xingang W, Chenghao L. Collaborative filtering recommendation algorithm based on MDP model [C] //14th International Symposium on Distributed Computing and Applications for Business Engineering and Science. Washington: IEEE, 2015: 110-113.
- [8] He R, McAuley J. Fusing similarity models with markov chains for sparse sequential recommendation [C] //IEEE 16th International Conference on Data Mining. Barcelona: IEEE, 2016: 191-200.
- [9] Lecun Y, Bengio Y, Hinton G. Deep learning [J]. Nature, 2015, 521 (7553): 436-444.
- [10] 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. Como: ACM, 2017: 306-310.
- [11] 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. London: ACM, 2018: 1831-1839.
- [12] Shen Y, Li H, Yi S, et al. Person re-identification with deep similarity-guided graph neural network [C] //Proceedings of the European Conference on Computer Vision, 2018: 486-504.
- [13] Thekumparampil K K, Wang C, Oh S, et al. Attention-based graph neural network for semi-supervised learning [J]. arXiv Preprint arXiv: 1803.03735, 2018.
- [14] Selvan R, Kipf T, Welling M, et al. Extraction of airways using graph neural networks [J]. arXiv Preprint arXiv:

- 1804.04436, 2018.
- [15] HU Yankang, FAN Xin, YU Letian, et al. Graph based neural network regression strategy for facial image super-resolution [J]. Journal of Software, 2018, 29 (4): 22-33 (in Chinese). [呼延康, 樊鑫, 余乐天, 等. 图神经网络回归的人脸超分辨率重建 [J]. 软件学报, 2018, 29 (4): 22-33.]
- [16] Li Y, Tarlow D, Brockschmidt M, et al. Gated graph sequence neural networks [J]. arXiv Preprint arXiv: 1511.05493, 2015.
- [17] Li Z, Ding X, Liu T. Constructing narrative event evolutionary graph for script event prediction [J]. arXiv Preprint arXiv: 1805.05081, 2018.
- [18] Li R, Tapaswi M, Liao R, et al. Situation recognition with graph neural networks [C] //Proceedings of the IEEE International Conference on Computer Vision, 2017: 4173-4182.
- [19] Marino K, Salakhutdinov R, Gupta A. The more you know: Using knowledge graphs for image classification [C] //IEEE Conference on Computer Vision and Pattern Recognition, 2017: 20-28.
- [20] Pampín H J C, Peleteiro A. A time-aware exploration of recsys15 challenge dataset [C] //Proceedings of the 4th Spanish Conference on Information Retrieval. Granada: ACM, 2016: 1-4.