计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2024,33(3):52−62 [doi: 10.15888/j.cnki.csa.009447]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

融合双分支动态偏好的会话推荐①

沈学利, 王　乐, 田学成

(辽宁工程技术大学 软件学院, 葫芦岛 125105)
通信作者: 王　乐, E-mail: 964802718@qq.com

摘　要: 针对基于会话的推荐算法仅对用户单一偏好进行静态建模而无法捕捉用户受环境影响偏好产生的波动, 从而降低推荐准确性的问题. 提出融合双分支动态偏好的会话推荐方法: 首先, 通过异构超图来建模不同类型信息, 设计双分支聚合机制获取以及整合异构超图中信息并且学习多类型节点之间的关系, 再用价格嵌入增强器来加强类别和价格之间关系; 其次, 设计双层偏好编码器, 其中采用多尺度时序 Transformer 提取用户动态价格偏好, 利用软注意机制和反向位置编码学习用户动态兴趣偏好; 最后, 用门控机制融合用户多类型动态偏好, 向用户进行推荐. 通过在 Cosmetics 和 Diginetica-buy 两个数据集上进行实验, 结果证明与其他对比算法相比在 *Precision* 和 *MRR* 评价指标中有显著的提升.

关键词: 推荐系统; 多类型动态建模; 异构超图; 双分支; 注意力机制

引用格式: 沈学利,王乐,田学成.融合双分支动态偏好的会话推荐.计算机系统应用,2024,33(3):52–62. http://www.c-s-a.org.cn/1003-3254/9447.html

Session Recommendation Incorporating Dual-branch Dynamic Preferences

SHEN Xue-Li, WANG Le, TIAN Xue-Cheng

(Software College, Liaoning Technical University, Huludao 125105, China)

Abstract: Session-based recommendation algorithms only statically model a single preference of users and fail to capture the preference fluctuation of the users affected by the environment, thus reducing the recommendation accuracy. Therefore, this study proposes a session recommendation method that integrates dual-branch dynamic preferences. First, the heterogeneous hypergraph is used to model different types of information, and a dual-branch aggregation mechanism is designed to acquire and integrate the information in the heterogeneous hypergraph and learn the relationship between multiple types of nodes. Then, a price-embedded enhancer is used to strengthen the relationship between categories and prices. Second, a two-layer preference encoder is designed, which uses a multi-scale temporal Transformer to extract the user's dynamic price preference, and a soft attention mechanism and reverse position encoding are used to learn the user's dynamic interest preference. Finally, a gating mechanism is used to integrate the user's multi-type dynamic preferences and make recommendations to users. By conducting experiments on two datasets, namely Cosmetics and Diginetica-buy, the results prove that there is a significant improvement in *Precision* and *MRR* evaluation metrics compared with other algorithms.

Key words: recommendation system; multi-type dynamic modeling; heterogeneous hypergraph; dual-branch; attention mechanism

推荐系统 (recommendation system, RS) 作为缓解信息爆炸的有效工具, 在现代电子商务系统中起着至关重要的作用[1]. 传统的推荐方法 (例如: 协同过滤[2,3]) 通常在长期历史交互中学习用户偏好. 然而在许多情

① 基金项目: 国家自然科学基金面上项目 (42271409)
收稿时间: 2023-09-27; 修改时间: 2023-10-25; 采用时间: 2023-11-09; csa 在线出版时间: 2024-01-18
CNKI 网络首发时间: 2024-01-19

况下用户存在隐私政策或未登录状态下, 数据不可用的问题导致其表现较差. 因此提出了基于会话的推荐 (session-based recommendation, SBR). 基于会话的推荐不依赖用户的个人信息或历史行为数据, 而是通过分析用户的对话内容和上下文, 为用户提供相应的个性化建议或推荐. 在淘宝, 抖音, 小红书等电子商务平台和社交媒体中发挥关键作用.

目前主要分两种方法实现基于会话的推荐: 有效捕获会话中项目之间的依赖关系以及挖掘协作信号丰富单个会话信息. SR-GNN (session-based recommendation with graph neural network)[4]利用图门控神经网络来探索每个会话中远距离项目之间的依赖关系. 因为在处理短序列方面, 门控循环单元[5]比长短期记忆[6]能够更有效地捕捉和分析项目之间的复杂关联. GCE-GNN (global context enhanced graph neural network)[7]构建了一个覆盖所有会话的全局图, 以捕捉项目的潜在高阶信号. 但是由于会话内的用户行为受限, 将会话间的项目关系纳入分析用户意图中可以更为有效. 然而, 在实际应用中会话数据过于庞大, 虽然提供了丰富的价值, 但对于建模也极具挑战, 而且这些工作重点考虑建模连续项目之间的顺序转换, 会降低模型对用户偏好变化获取的敏感程度, 无法准确地捕捉用户的动态偏好变化.

现实生活中用户的偏好往往会因时间而不断变化[8,9]并且在市场营销研究[10–13]中表示, 价格对用户最终是否决定购买起着关键作用. 例如, 用户在长时间对价格高昂的手机感兴趣, 但下一个时间段可能更倾向于便宜的服饰、食物等其他类别, 但由于用户浏览价格高昂手机次数多于低廉的服饰, 系统只考虑用户行为的转换关系而忽略用户偏好的转移. 这使得充分捕捉用户多类型偏好的动态性成为一个复杂的问题.

近年来, 图神经网络在会话推荐方面取得了显著成果[14]. 例如, SR-GNN[4]通过构建会话中不同项目间的图结构, 并利用图卷积神经网络来学习项目之间的转换关系, 从而提供更精准和个性化的推荐结果. STAMP (session-based temporal attention model for personalized recommendation)[15]结合了时间注意力机制来捕捉用户在不同时间点的兴趣变化, 并根据用户的历史行为模式提供个性化的推荐服务. 尽管它们很有效但现有的方法在处理复杂的多类型关系和多跳关系时会有局限性. 在本文中, 信息之间是高阶的关系, 如<项目、类别、价格>之间的关系. 这种复杂多类型节点之间高阶依赖关系很难使用传统的图方法建模. 研究人员提出用异构超图建模高阶关系[16]用于捕捉多类型节点信息. 现有研究表明异构超图网络能够有效结合异构图在建模异构信息方面的优势以及超图在捕获复杂高阶依赖关系方面的优势来处理 SBR 中的异构信息. 异构超图中的节点可以与多类型节点连接, 形成丰富的关系网络. 然而在本文建立的异构超图中存在多类型节点之间的复杂关联关系和多层次的关系结构, 在多次迭代的过程中可能会信息丢失从而导致节点之间的关系丢失或模糊, 影响信息的有效传播和学习. 如何有效学习各个类型节点的信息以及关系成为一个问题.

针对上述问题, 本文提出融合双分支动态偏好的会话推荐 SDBDP (session recommendation incorporating dual-branch dynamic preferences) 模型. 受 CoHHN (co-guided heterogeneous hypergraph network)[17]的启发, SDBDP 不仅考虑用户的兴趣和类别, 还考虑了不同时间用户对类别的价格偏好, 以提高推荐性能. 利用异构超图建立多类型节点的信息并设计一种双分支聚合机制来传播各个节点嵌入, SDBDP 先引入每个类型的分类树, 以确定用户各个类型的偏好嵌入, 通过使用注意力机制和聚合机制来学习各个类型节点之间的关系进行学习, 形成不同类型的超边来传播信息, 并提出价格嵌入增强器用于加强类别和价格之间的关联, 以便模型可以更好地理解它们之间的关系. 采用反向位置嵌入对位置信息进行编码, 利用软注意机制将位置嵌入和兴趣嵌入进行学习, 最终得到用户的动态兴趣偏好. 采用多尺度时序 Transformer 对价格动态建模获取局部的价格波动, 使模型更准确地预测动态价格变化, 提高模型对于价格变化的学习能力, 最终得到用户的动态价格偏好.

基于学习到的多类型动态偏好通过门控融合机制相互学习以丰富语义信息. 最后 SDBDP 根据项目特征和用户的价格和兴趣偏好进行推荐. 综上所述, 本文的工作贡献如下.

• 提出双分支聚合机制, 来有效学习异构超图中多类型节点之间的信息.

• 提出多尺度时序 Transformer, 利用多尺度动态建模来获取用户价格意识的变化, 并通过 Transformer 来学习动态价格偏好.

• 在两个公共基准上的广泛实验证明了本文提出

的 SDBDP 与最先进的方法相比的优越性.

1 相关工作

在学术界和工业界的努力下, RS 正在迅速发展来应对电子商务系统日益增长的需求和复杂性所带来的挑战. 常用的推荐方法包括传统的协同过滤方法, 基于深度学习的会话推荐方法以及基于图神经网络的推荐方法.

1.1 传统的协同过滤方法

协同过滤 (collaborative filtering, CF)[2,3]是推荐系统在早期研究中的常用方法, 它根据用户历史浏览记录来预测用户的兴趣偏好. 例如, 著名的浅层方法, 矩阵分解 (matrix decomposition, MF)[18]对整个用户-项目交互矩阵进行分解, 并对每个用户和项目进行潜在表示. 神经协同滤波 (neural collaborative filtering, NCF)[3]首次引入了多层感知器来近似矩阵分解过程, 而现代商业在线系统中许多用户信息是匿名的, 导致基于协同过滤算法的失败.

1.2 基于深度学习的会话推荐

传统的推荐方法往往无法有效捕捉项目或者用户的隐含特征, 因此在推荐过程中会忽略掉一些重要信息或关联关系. 近年来, 随着深度学习的发展, 神经网络模型得到广泛应用.

循环神经网络 (recurrent neural network, RNN)[5,6,19]是一种深度学习模型, 与传统神经网络不同, RNN 具有循环结构, 允许信息在网络中保持持续传递, 以便有效处理序列数据中的时间依赖关系, 在推荐系统中得到广泛应用. GRU4REC (gated recurrent unit for session-based recommendations)[6]简单地将数据视为时间序列, 并应用多层门控循环单元 (gated recurrent unit, GRU) 进行建模, 可以帮助网络更好地获取序列数据中的长期依赖关系和时间动态模式, 相比于传统的 RNN, GRU 能更有效地避免梯度消失和梯度爆炸的问题, 但是无法捕捉用户的长期偏好. NARM (neural attentive session-based recommendation model)[5]堆叠 GRU 作为编码器提取信息, 并堆叠自注意力层为每个隐藏状态分配权重, 形成会话嵌入, 能够灵活地根据不同用户的偏好模式来调整注意力权重, 从而更好地获取用户的兴趣变化规律, 但可能面临时间序列引入的偏差问题. SSRM (streaming session-based recommendation)[20]考虑了特定用户的历史会话, 并应用注意力机制进行组合. 尽管这些方法取得了一定的成功, 但传统的注意力机制没有考虑到时间维度, 难以捕获用户随时间变化的动态偏好.

1.3 基于图神经网络的推荐方法

图神经网络最初应用于对图像和视觉数据进行特征提取和处理[21]. 卷积神经网络 (convolutional neural network, CNN) 可以被看作是一种特殊类型的图神经网络, 它在图像处理任务中取得了重大的突破[22]. 随着对图神经网络结构和算法的深入研究, 图神经网络被引入到推荐系统[4,23]. 在推荐系统领域, 图神经网络可以用来处理用户行为图和项目关联图, 提高推荐的精准性, 从而更好地满足用户的个性化需求.

其中, 最早将图神经网络应用于工业推荐系统的 PinSage[24]模型利用图神经网络来学习用户和项目的表示向量, 以更好地获取用户兴趣和项目特征之间的关联关系. 由神经图自动编码器组成的 GC-MC (graph convolutional matrix completion)[25]模型, 用于重建用户项目评级图. NGCF (neural graph collaborative filtering)[26]中提出构建一个用户-项目二分图, 并利用多个图神经层来捕捉用户和项目之间的多跳协作信号. 虽然这些方法表明优于其他方法, 但是它们更擅长处理同构信息的图结构. 在社交推荐中, 用户之间的关系可以通过社交图来建模[27]. 例如, DiffNet[28]采用图卷积网络来对用户嵌入在其社交连接之间的扩散进行建模, 具有处理异构信息的能力, 并且能够有效地学习节点表示, 但缺乏对会话推荐特殊性的考虑, 比如对于没有用户标识的短时间会话中用户顺序行为一般无法直接使用. 一些方法将超图用于会话推荐中, SHARE (session-based hypergraph attention network for recommendation)[29]是将每一个会话构建一个超图, 并在会话序列上应用多个滑动窗口来捕获上下文信息. 综合超图在会话推荐上的优越性能, 本文采用异构超图来建模多类型节点的异构信息.

2 模型研究

基于超图的 SRS 任务一般包含超图构建, 项目嵌入学习表示关系, 目标会话的表示和生成, 预测 4 个模块. SDBDP 模型在此基础上增加一个价格嵌入增强器, 来挖掘项目类别和价格之间潜在的关系, 为获取多类型偏好提供信息.

SDBDP 通过构造异构超图进行学习, 根据会话的高阶转换关系构建异构超图来获取异构信息. 通过获

取到的异构信息, 设计一种双分支聚合机制, 在同构分支和全局分支的节点间传播各种信息, 得到不同类型的初始嵌入. 为加深价格和类别的潜在关系, 设计价格嵌入增强器, 利用初始的价格和类别嵌入通过交叉注意力机制来更好地学习用户价格嵌入. 针对用户动态价格偏好, 根据价格嵌入增强器得到价格嵌入, 利用多尺度在价格嵌入上进一步捕获短时间价格动态变化, 将得到的价格感知采用 Transformer 动态建模, 得到用户的动态价格偏好. 针对用户动态兴趣偏好, 根据双分支聚合机制得到兴趣嵌入, 将兴趣嵌入和反向位置编码衔接后利用软注意机制进行学习, 以获得动态兴趣偏好. 最终采用门控机制融合用户的兴趣和价格的动态偏好, 结合项目特征、价格和兴趣偏好对用户的行为进行预测, 模型架构如图 1 所示.

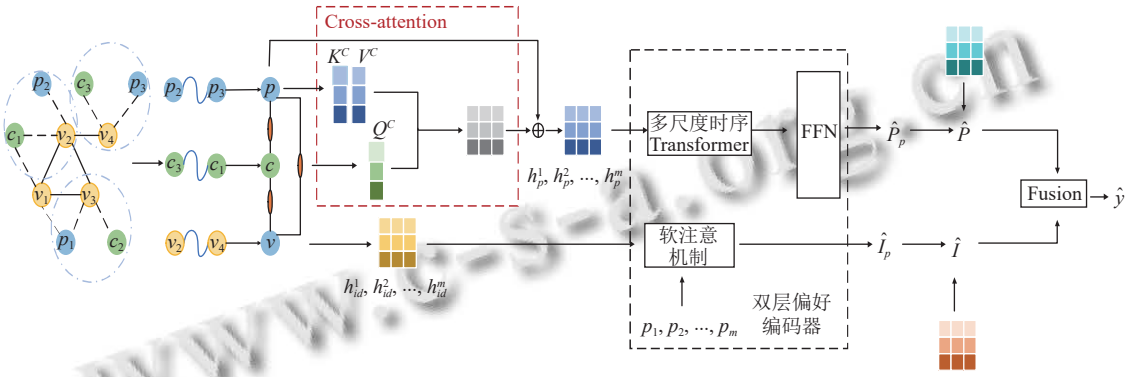


图 1 SDBDP 模型架构

2.1 问题描述

V 表示项目的集合, 设 $V=\{v_1,v_2,\cdots,v_n\}$ 表示所有会话的物品集合, $S=\{s_1,s_2,\cdots,s_m\}$ 是一个匿名用户按照时间顺序交互产生的会话, 其中 $|m|$ 是 s 的长度. 会话推荐的目的是根据短时间用户的交互信息预测用户下一次点击的物品 v_{n+1}. 即通过会话推荐模型, 输出概率向量 $\hat{y}=\left\{\hat{y}_1,\hat{y}_2,\cdots,\hat{y}_j\right\}$, 其中 $\hat{y}$ 代表用户和不同物品之间产生的交互概率, 选择 top-k 为推荐的项目列表.

商品的价格是否合适是由商品的类别来衡量的, 为提高会话推荐的准确率, 将价格看作分类变量. 根据文献[30]提出某一类商品的价格分布更接近逻辑分布, 而不是广泛使用的均匀分布. 将价格离散为 ρ 水平, 其中每个区间对应的概率相等. 形式上, 对于价格为 V_p 的项目 V_i, 其类别的价格范围为 $[\min,\max]$, 确定其价格水平如下:

$$p_i=\left\lfloor\frac{\Phi\left(x_p\right)-\Phi(\min)}{\Phi(\max)-\Phi(\min)}\times\rho\right\rfloor \tag{1}$$

其中, $\Phi(V)$ 为逻辑分布的累积分布函数.

2.2 异构超图构建

为了捕捉异构超图中高阶关系, 在异构超图中对以下异构节点进行了编码: 项目、项目价格和项目类别, 定义 4 种类型的超边用于表示节点之间的多类型关系, 如图 2 所示. 异构超图 $G=(V,E)$, 其中 V 是 N 个唯一节点的集合由类别节点 $c\in C$, 价格节点 $p\in P$ 和物品节点 $v\in V$ 组成, E 是 M 条超边的集合. 每个超边 $e\in E$ 可连接不同类型的任意数量的节点, 并被分配一个权重 $W\in R^{M\times M}$. 异构超图可以通过关联矩阵表示, 若节点在超边 e 上, 则 $h(v,e)=1$, 否则为 0.

2.3 双分支聚合机制

异构超图使 SDBDP 可对高阶关系进行建模, 并且有效缓解会话推荐中数据稀疏问题, 但是很难从异构节点中获取有用的信息. 在异构超图中, 一个目标节点能够和不同类型节点相邻, 同类型节点包含的信息为同构信息, 不同类型节点包含的信息为异构信息. 因此, 在该模块中将消息聚合汇总为两个分支, 即同构分支和全局分支. 在同构分支中, 获取与目标节点相邻的同类型节点产生的信息, 在全局分支中, 获取不同类型对目标节点产生的信息.

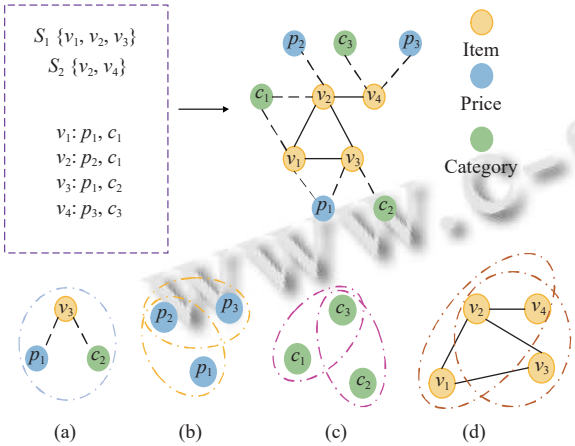
SDBDP 模型采用项目超边传播各个类型特征, $x_k\in R^d$ 是嵌入具有 k 类型的目标节点, 目标节点相邻具有 j 类型节点形成 N_k^j, 同构分支聚合用于学习同类型相邻节点给目标节点带来的信息, 假设所有目标节点嵌入维度为 d, 在同构分支中得到不同类型的嵌入表示:

$$n_k^j=\sum_{k=1}^{K}\alpha_k x_k^j \tag{2}$$

$$\beta_k = \sigma\left(\omega^j x_k^j\right) \tag{3}$$

$$\alpha_k = \frac{\exp(\beta_k)}{\sum_{x_k^j \in N_k^j} \exp(\beta_k)} \tag{4}$$

其中, x_k^j表示 k 类型相邻的 j 类型节点, 本文通过一层异构超图在一个会话中捕获同构之间的关系以缓解平滑问题, 即每个项目只从其一阶邻居中聚合信息. $\sigma(\cdot)$为 tanh 激活函数, ω^j表示不同类型的不同向量, 最后将同构分支聚合表示为$g_i\left(N_k^j\right)$.



(a) 将一个项目所有类型的特征用项目超边连接
(b) 一个会话中项目的价格节点用价格超边连接
(c) 一个会话中项目的类别节点由类别超边连接
(d) 一个会话中项目节点由兴趣超边连接

图 2 4 种超边

全局分支聚合考虑的是将同构分支学习到的不同类型嵌入n_k^j传播到目标节点为其提供异构信息. 全局分支的嵌入表示:

$$\eta_k = \sigma\left(W_k\left[x_k \| n_k^j \| n_k^m\right]\right) + W_k^j \cdot n_k^j + W_k^m \cdot n_k^m \tag{5}$$

$$h_k = x_k + \eta_k \cdot n_k^j + (1-\eta_k) \cdot n_k^m \tag{6}$$

其中, $W_k \in R^{d\times 3d}$, $W_k^j \in R^{d\times d}$和$W_k^m \in R^{d\times d}$是可学习参数, $\|$为衔接, $\sigma(\cdot)$为 Sigmoid 激活函数, 将全局分支聚合表示为$g_h\left(x_k, n_k^j, n_k^m\right)$, 所有节点更新表示为:

$$h_{id} = g_h\left(x_{id}, g_i\left(N_{id}^p\right), g_i\left(N_{id}^c\right)\right) + N_{id}^{id} \tag{7}$$

$$h_c = g_h\left(x_c, g_i\left(N_c^p\right), g_i\left(N_c^{id}\right)\right) \tag{8}$$

$$h_p = g_h\left(x_c, g_i\left(N_p^c\right), g_i\left(N_p^{id}\right)\right) \tag{9}$$

2.4 价格嵌入增强器

根据价格超边和类别超边来提取用户的价格嵌入$P = \left[h_p^1, h_p^2, \cdots, h_p^m\right]$及类别嵌入$C = \left[h_c^1, h_c^2, \cdots, h_c^m\right]$表示向量. 利用价格嵌入和类别嵌入中的关系进行交叉注意力机制 (cross-attention) 操作, 以类别嵌入作为查询向量Q^C, 对价格嵌入做注意力. 通过 cross-attention 操作, 模型可以在两者之间建立跨特征的关联, 使得模型能够更好地理解价格和类别之间的潜在关系, 用于更好学习价格的嵌入.

$$Attention\left(Q^C, K^C, V^C\right) = Softmax\left(\frac{\left(Q^C K^{C^{\mathrm{T}}}\right)}{\sqrt{d_c}} \cdot V^C\right) \tag{10}$$

2.5 动态兴趣偏好提取

根据兴趣超边来提取用户的兴趣嵌入$I = [h_{id}^1, h_{id}^2, \cdots, h_{id}^m]$表示向量. 用户的兴趣偏好会随着时间变化而出现波动[9]. 因此, 采用一种动态兴趣注意机制, 每个项目在会话的权重往往受到项目位置信息 (即会话序列中的时间序列) 的影响, 该机制动态地选择和线性组合不同项目信息. 本文利用反向位置嵌入[7]对位置信息进行编码, 将位置嵌入与学习到的兴趣嵌入衔接, 表示如下:

$$x_i^* = \tanh\left(W_f\left[h_{id}^i \| p_i\right] + b_f\right) \tag{11}$$

其中, $W_f \in R^{d\times 2d}$和b_f是可学习参数. $x_i^* \in R^d$是兴趣嵌入中第i个乘积表示, p_i为位置嵌入. 兴趣偏好可通过计算会话中各项表示的平均值来表示:

$$\bar{x}^* = \frac{1}{m}\sum_{i=1}^{m} x_i^* \tag{12}$$

其中, $x_i^* \in R^d$表示会话中第i个项目嵌入, 采用软注意机制学习会话中每个项目对应权重:

$$\beta_i = Softmax_i(\varphi_i) \tag{13}$$

$$\varphi_i = u^{\mathrm{T}}\sigma\left(W_1 x_i^* + W_2 \bar{x}^* + b\right) \tag{14}$$

其中, $\sigma(\cdot)$为 Sigmoid 激活函数, W_1, $W_2 \in R^{d\times d}$和b为可学习参数, $u^{\mathrm{T}} \in R^d$为注意向量, 用户的动态兴趣偏好表示如下:

$$\hat{I}_p = \sum_{i=1}^{k} \beta_i h_{id}^i \tag{15}$$

2.6 动态价格偏好提取

在实际应用推荐中, 用户对价格的偏好可能会随着时间的推移表现出不同的趋势. 为解决这一挑战, 采用多尺度对用户的价格偏好进行建模, 利用多头注意力机制捕获序列间项目的关系, 最终获得用户的动态

价格偏好.

2.6.1 多尺度时序 Transformer

自注意机制擅长捕获整个序列中的项目-项目全局依赖关系. 然而自注意机制的高计算代价限制模型在实际设置中的可伸缩性[31]. 基于文献[32,33]中的 Transformer 结构设计, 采用基于低秩的无二次注意操作的自注意层, 以近似线性模型的复杂度, 表示低等级的自我关注如下:

$$\hat{H} = Softmax\left(\frac{H\cdot W^{Q}\left(E\cdot H\cdot W^{K}\right)^{\mathrm{T}}}{\sqrt{d}}\cdot F\cdot H\cdot W^{V}\right) \tag{16}$$

其中, E和$F\in R^{J/C\times d}$来进行低秩嵌入变换, C 表达低秩尺度, J/C表示输入价格序列 P 上低秩潜在表示空间的数量, W^{Q}, W^{K}, W^{V}为用于映射输入到查询的参数矩阵. 利用 E 和 F 将$R^{J\times d}$维键和值转换表$H\cdot W^{K}$和$H\cdot W^{V}$投影到$\left|R^{J/C\times d}\right|$维潜在低秩嵌入$\hat{H}\in\left|R^{J/C\times d}\right|$.

多尺度方法可捕捉价格变化在不同时间尺度上的模式和趋势, 以提供更全面和准确的价格建模. 根据价格嵌入增强器获取到的用户价格嵌入$P=\left[h_p^1,h_p^2,\cdots,h_p^m\right]$表示向量.

为增强模型的多尺度学习, 采用层次结构增强低等级 Transformer 捕获动态价格, 如图 3 所示. 首先模型是按时间顺序产生的会话, 构建一个价格感知聚合器来生成特定时间段表示T_t, 表示一定时间段的价格波动. 将 t 定义为一段时间的子序列的长度, 表示如下:

$$\begin{aligned}\Gamma^{t}&=\left\{\gamma_1,\gamma_2,\cdots,\gamma_{\frac{m}{t}}\right\}\\&=\left[\eta\left(h_p^1,h_p^2,\cdots,h_p^t\right);\cdots;\eta\left(h_p^{m-t+1},\cdots,h_p^m\right)\right]\end{aligned} \tag{17}$$

其中, $\Gamma^{t}\in R^{\frac{m}{t}\times d}$和$\gamma\in R^{d}$表示价格感知的内存生成, $\eta(\cdot)$表示捕获短时间价格动态聚合器. 模型利用平均池化来进行嵌入聚合, 然后将得到的价格感知进行多头自注意建模表示如下:

$$H^{t}=Softmax\left(\frac{\Gamma^{t}\cdot W_t^{Q}\left(\Gamma^{t}\cdot W_t^{K}\right)^{\mathrm{T}}}{\sqrt{d}}\cdot\Gamma^{t}\cdot W_t^{V}\right) \tag{18}$$

本文中使用分层的 Transformer 和两组不同规模设置的t_1, t_2, 所以多尺度时序 Transformer 可以产生 3 种尺度嵌入$\hat{H}\in R^{J\times d}$, $H^{t_1}\in R^{J/t_1\times d}$, $H^{t_2}\in R^{J/t_2\times d}$.

为了将多尺度动态价格偏好合并到一个共同潜在表示空间中, 模型引入一个输入投影W_ϕ将$R^{(J/C+J/t_1+J/t_2)\times d}$维嵌入转换为$R^{J\times d}$上述编码的特定尺度的嵌入与融合层聚合表示如下:

$$\bar{H}=W_\phi\cdot\left(\hat{H}\,\|\,H^{t_1}\,\|\,H^{t_2}\right) \tag{19}$$

其中, ‖表示在不同嵌入向量上的连接操作.

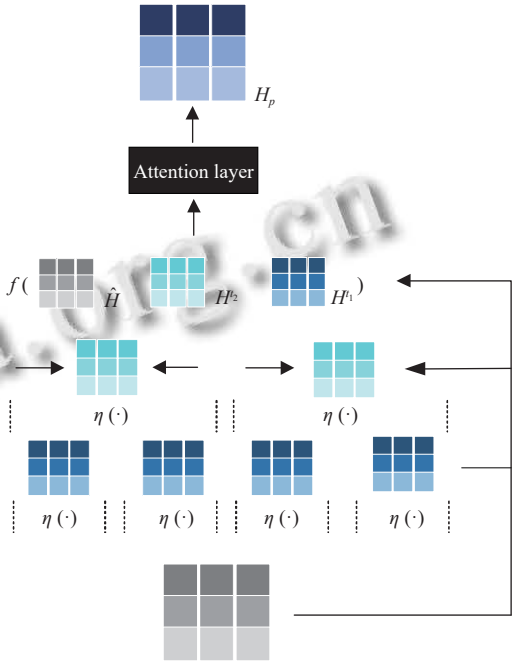


图 3 多尺度时序 Transformer

为了使价格偏好有共同参与多维交互学习的能力, 将 H 投射到 h 潜在表示空间, 并行执行与头部特定的注意操作, 表示如下:

$$H_p=[head_1,head_2,\cdots,head_h] \tag{20}$$

$$head_i=Attention\left(W_i^{Q}S_p,W_i^{K}S_p,W_i^{V}S_p\right) \tag{21}$$

其中, W_i^{Q}, W_i^{K}, $W_i^{V}\in R^{d/h\times d}$分别用于映射输入到查询参数矩阵分别是键和值, h 为头部数量, $S_p=W_\phi\cdot(\hat{H}_i\,\|\,H_i^{t_1}\,\|\,H_i^{t_2})$.

2.6.2 位置级的前馈网络

自注意子层主要基于线性投影, 为了赋予模型非线性和不同维度之间的相互作用, 在多尺度时序 Transformer 中对自注意子层的输出应用位置前馈网络. 它由两个仿射变换组成, 在一个高斯误差线性单位(Gaussian error linear unit, *GELU*) 激活之间表示如下:

$$PFFN\left(H_p\right)=\left[FFN\left(h_1^{(l)}\right)^{\mathrm{T}},\cdots,FFN\left(h_p^{(l)}\right)^{\mathrm{T}}\right] \tag{22}$$

$$FFN(u)=GELU\left(uW_{f_1}+b_{f_1}\right)W_{f_2}+b_{f_2} \tag{23}$$

$$GELU(u)=u\Phi(u) \tag{24}$$

其中, $\Phi(x)$ 为标准高斯分布的累积分布函数, W_{f_1}, $W_{f_2} \in R^{d\times d_i}$ 和 b_{f_1}, $b_{f_2} \in R^d$ 为可学习参数, 在所有位置上共享. 在这项工作中, 在 OpenAI GPT[34]和 BERT[35]之后, 使用了一个更平滑的 *GELU*[36]激活. l表示第l个多尺度时序 Transformer. H_p表示动态价格偏好$\hat{P}_p$.

2.7 全局特征融合

在一个项目中, 获取用户的动态价格偏好$\hat{P}_p$和兴趣偏好$\hat{I}_p$, 同时给定一个项目x_i的 ID 嵌入v_{id}^i和价格嵌入v_p^i, 获得最终的价格和兴趣偏好如下:

$$P_p = \hat{P}_p^{\mathrm{T}} \cdot v_p^i \tag{25}$$

$$I_p = \hat{I}_p^{\mathrm{T}} \cdot v_{id}^i \tag{26}$$

其中, I_p, $P_p \in R^d$是用户的兴趣偏好和价格偏好, 采用门控机制来形成最终的偏好表示如下:

$$\psi = \sigma\left(W_s\left[P_p \,\|\, I_p\right]\right) \tag{27}$$

$$\hat{y}_i = \psi \cdot P_p + (1-\psi)\cdot I_p \tag{28}$$

用 *Softmax* 函数对它进行处理, 得到最终的分数表示如下:

$$\hat{y}_i = \frac{\exp(y_i)}{\sum_{j=1}^{n}\exp(y_i)} \tag{29}$$

其中, $\hat{y}_i$表示在当前会话中单击$x_i \in V$的概率. 模型通过交叉熵损失训练, 为防止过拟合, 模型引入 L2 正则化, 表示如下:

$$\mathcal{L} = -\sum_{i=1}^{|V|} y_i \log_a(\hat{y}_i) + (1-y_i)\log_a(1-\hat{y}_i) + \lambda\|\mu\|^2 \tag{30}$$

其中, λ表示正则化因子, μ表示模型有效参数.

3 实验与分析

3.1 数据集

为验证 SDBDP 模型的有效性, 本文在两个公开数据集上进行实验. Cosmetics 数据集 (https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-cosmetics-shop) 来源于 kaggle 上一个化妆品购买的数据集, 数据集采用的时间范围为 2019 年 10 月; Diginetica-buy 数据集 (https://competitions.codalab.org/competitions/1116) 是来自 2016 年的 CIKM 杯, 包含了用户在电子商务平台上真实购买历史记录是典型的交易数据. 这两个数据集被广泛应用在推荐系统的会话推荐中, 均包含: 用户与项目的 ID, 项目的类别和价格以及用户交互时间信息. 过滤掉两个数据集中会话长度为 1, 项目出现次数少于 10 的会话和项目. 对于一个会话, 最后一个项目被视为标签其余的序列被用于建模用户首选项. 在两个数据集中按时间顺序选取前 70% 作为训练集, 20% 作为验证集, 最后 10% 作为测试集, 实验数据集的统计如表 1 所示.

表 1 数据集统计

数据集	Cosmetics	Diginetica-buy
项目数	23 094	24 889
交互数	1 058 263	855 070
价格水平	11	100
类别数	301	721
训练会话数	109 845	131 278
验证会话数	31 384	37 508
测试会话数	15 692	18 754
平均长度	6.74	4.56

3.2 评价指标

实验使用 *Precision@K* (*P@K*) 和 *MRR@K* 两种评价指标对模型进行评价, 其中 *P@K* 是衡量基于会话的推荐系统的预测精准度的指标, 表示在推荐列表排名前 K 个推荐项中正确物品所占的比例, 其值越大效果越好. 平均倒数排名 (mean reciprocal ranking, *MRR*) 是正确推荐物品的平均倒数排名, 在推荐物品列表中正确推荐物品的位置越高, 则其值越大效果越好, 当正确推荐物品的排名不在前 20 时, 该值为 0. 本次实验中, 取 K=10, K=20.

3.3 实验环境和参数设置

本文模型基于 PyTorch 框架实现, 开发语言采用 Python 3.8.13, 优化模型采用 Adam 优化器. 实验所用硬件的操作系统为 Windows 11, 处理器为 Intel(R) Core (TM) i5-1135G7 @ 2.40 GHz, 显卡为 NVIDIA GeForce MX450 2 GB, 运行内存为 16 GB. 实验中 Diginetica-buy 数据集和 Cosmetics 数据集的参数设置如表 2 所示, 其中 epoch 表示训练次数, batch_size 表示训练批次大小, learnRate 表示学习率, d 表示嵌入维度, L 表示使用的层数.

表 2 参数设置

参数	Diginetica-buy	Cosmetics
epoch	30	20
batch_size	100	100
learnRate	0.000 5	0.000 5
d	128	128
L	3	3

3.4 对比实验模型

(1) GRU4Rec[6]使用 GRU 通过一个会话-并行的小批处理训练过程来捕获项目序列的表示.

(2) BERT4Rec[37]采用双向自注意体系结构来编码历史交互序列.

(3) NARM[5]通过将注意力纳入会话推荐的 RNN 中, 它改进了 GRU4Rec[6].

(4) SR-GNN[4]构建会话图, 并利用图神经网络来捕获项目之间的成对转换.

(5) CoHHN[17]通过超图捕获项目之间的超成对关系, 并使用注意力层来替换之前工作中的所有 RNN 编码器.

3.5 对比实验结果

在 Diginetica-buy 和 Cosmetics 两个数据集上对所有模型进行测试, 结果见表 3, SDBDP 在两个真实数据集上的结果均优于其他对比模型. NARM 和 BERT4-Rec 通过引入注意力机制捕捉会话中的主要意图而优于 GRU4Rec, 注意力机制优点是它可以捕获序列项目之间的不一致依赖关系. 基于 GNN 的方法 SR-GNN 依靠对节点之间成对关系进行建模的能力获得竞争性能. 此外 CoHHN 虽优于单一类型建模获取用户兴趣偏好的信息, 但它没有考虑用户偏好随时间变化而产生兴趣波动, 无法自动适应数据的变化, 若用户的偏好产生变化, 模型可能会失效, 因此其性能不如所提出的 SDBDP 模型. 本模型相较于最优对比模型, 在 Diginetica-buy 数据集上, *P*@20 和 *MRR*@20 指标上分别提升 3.1%, 3.0%; 在 Cosmetics 数据集上 *P*@20 和 *MRR*@20 指标上分别提升 2.7%, 2.4%, 这表明本文模型能够有效获取用户动态偏好数据, 提升会话推荐性能.

表 3 在 Diginetica-buy 和 Cosmetics 数据集上的对比实验 (%)

Method	Diginetica-buy				Cosmetics			
	P@10	*P*@20	*MRR*@10	*MRR*@20	*P*@10	*P*@20	*MRR*@10	*MRR*@20
GRU4Rec	21.91	27.54	11.27	11.70	19.27	21.73	14.40	14.58
BERT4Rec	47.22	60.16	20.37	21.51	38.21	46.35	23.32	23.87
NARM	46.28	56.59	21.70	23.29	42.32	46.09	34.10	34.45
SR-GNN	45.67	56.1	21.25	22.89	43.80	48.09	34.52	34.89
CoHHN	50.29	63.24	24.89	25.78	47.54	53.65	36.31	36.74
SDBDP	**51.39**	**65.19**	**25.62**	**26.55**	**49.31**	**55.09**	**37.21**	**37.62**

从表 4 中可以看出, SR-GNN, GRU4Rec, BERT4Rec, NARM 的方法复杂度较低但性能不如 CoHHN 模型. SDBDP 模型的方法复杂度要低于基线最优模型, 运行时间要少于 CoHHN 模型, 而且 SDBDP 模型的性能优于其他模型, 说明本文所提出的 SDBDP 模型的性能较高.

表 4 模型方法复杂度

Method	方法复杂度
GRU4Rec	757.355M
BERT4Rec	1.28G
NARM	928.619M
SR-GNN	14M
CoHHN	4.424G
SDBDP	2.255G

3.6 参数敏感性分析

为研究模型传播层数对推荐效果的影响, 继续在 Diginetica-buy 和 Cosmetics 两个数据集上进行实验, 其他的参数保持不变的情况下, 在 1–5 层数内调节, 实验结果如图 4 所示.

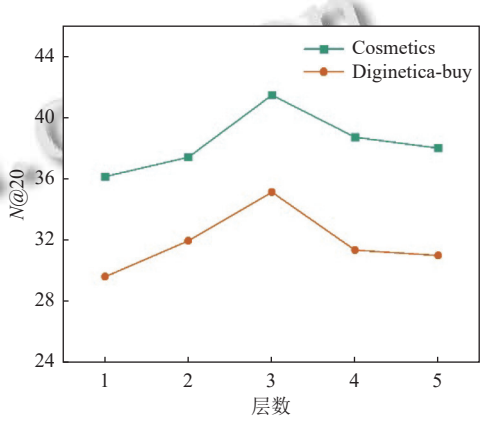


图 4 层数对两个数据集的影响

从图 4 可以看出, 随着传播层数的增加, SDBDP 模型在 *N*@20 (*N* 表示 *NDCG*, 可以通过给排名靠前物品的更高得分来确定命中的位置) 中有所提升. 当层数不大于 3 层时, 更多的传播层可以获得更好的性能, 这表明额外的消息传递层将从其他邻居获得潜在的依赖关系. 而进一步使用更多的层时, 会引入噪声, 这会导致过度平滑的问题.

学习率 (learnRate) 是模型训练中至关重要的超参数之一, 它对于确定优化目标函数是否会收敛到局部最小值以及何时会达到最小值具有重要影响. 在本研究中, 尝试了不同的学习率设置来对模型进行训练, 实验结果如表 5 和表 6 所示.

表 5 在 *P*@20 下学习率对模型的影响

数据集	0.000 1	0.000 3	0.000 5	0.000 7	0.001	0.003
Cosmetics (%)	53.38	54.59	**55.09**	50.9	53.91	51.92
Diginetica-buy (%)	64.38	59.5	**65.19**	63.95	63.91	59.6

表 6 在 *MRR*@20 下学习率对模型的影响

数据集	0.000 1	0.000 3	0.000 5	0.000 7	0.001	0.003
Cosmetics (%)	35.09	37.29	**37.62**	35.52	36.95	35.79
Diginetica-buy (%)	26.1	24.86	**26.55**	25.83	25.85	24.18

从表 5 和表 6 中可以清晰地看出, 对于不同的 n 值, 当学习率 learnRate 设置为 0.000 5 时, 在 Cosmetics 和 Diginetica-buy 数据集上, 模型表现出最佳性能. 因此, 我们选择将学习率设置为 0.000 5, 作为本研究模型的最佳参数配置.

3.7 消融实验

本文采用多类型偏好对用户行为进行预测, 对不同时间段偏好进行动态学习. 为验证设置增加多类型对推荐性能的影响, 去除对价格和类别的学习, 只学习用户兴趣偏好, 将变体命名为 SDBDP-I. 为研究动态偏好获取, 将多尺度时序 Transformer 改为用自注意学习价格嵌入, 将此变体命名为 SDBDP-P. 在消融实验中参数设置和原模型一致. 结果如图 5、图 6 所示.

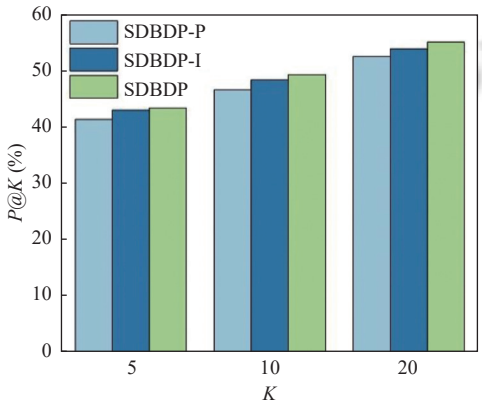


图 5 在 Cosmetics 数据集下消融实验

从图 5 和图 6 可以看出, 本模型在两个数据集上均优于两种变体, 且 SDBDP-I 性能最差, 证明增加多类型偏好学习可以更精准地为用户推荐. 通过 SDBDP-P 比 SDBDP 推荐性能差, 证明自注意关注于局部上下文而缺乏全局信息的捕捉能力, 在对用户偏好进行学习时存在一些局限性, 而且自注意不擅长对时序数据中的时间相关性进行建模, 无法动态获取用户的偏好信息. 性能最优的是本文所提出的 SDBDP 模型. 通过多类型学习用户的偏好, 利用多尺度时序 Transformer 有效地捕捉用户价格, 以及整合不同时间尺度捕捉的价格信息的局部和全局关系, 还可以减轻局部噪音, 最终提升推荐性能.

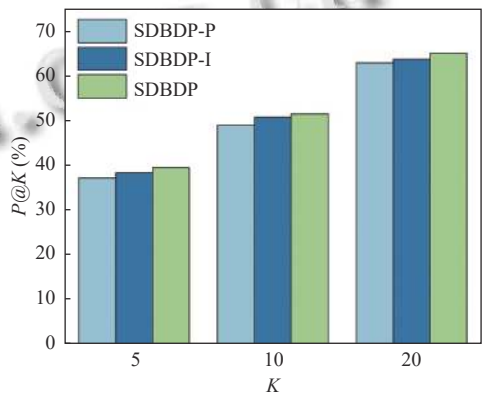


图 6 在 Diginetica-buy 数据集下消融实验

3.8 实例分析

为了直观地展示 SDBDP 模型在 Diginetica-buy 数据集上的有效性和准确性, 本文从测试集中随机选择了一段会话进行实例分析. 这个会话长度为 13, 其中前 12 个项目构成了目标会话, 而最后一个项目则是待预测项目 (即真实值).

在这个实例分析中, 本文选择了两个代表性的对比方法, 分别是 SR-GNN 和 CoHHN. SR-GNN 是首个将图神经网络融入会话推荐算法的方法, 而 CoHHN 则是在 Diginetica-buy 数据集上效果最好的基线方法. 前 16 个推荐结果如表 7 所示.

表 7 推荐实例

会话值/模型	项目编号
目标会话	58814→58572→5312→58239→58814→56209→601→58572→5312→56773→601→57731
真实值	58317
SR-GNN	57731, 58572, 59385, 56601, 56773, 58783, 58708, 57043, 54915, 7920, 5312, 58281, 56511, 28668, 601, 58423
CoHHN	57731, 58572, 601, 58281, 58365, 58783, 57043, 58281, 54615, 637, 56209, 58281, 58317, 58814, 601, 56492
SDBDP	57731, 58572, 56492, 5312, 601, 58423, 56601, 59385, 58317, 54915, 56511, 28668, 56773, 58423, 56209, 58814

从表 7 中可以观察到, CoHHN 和 SDBDP 模型都成功在前 16 个推荐结果中预测到真实值 58317 号项目, 而 SR-GNN 模型没有准确预测出真实值, 这说明了 SDBDP 模型的有效性. 此外, SDBDP 在正确预测项目的排名上达到第 9 位, 而 CoHHN 仅排在第 13 位, 这说明了 SDBDP 模型在准确性方面的优越性.

4 结论与展望

本文通过分析动态价格和兴趣偏好在用户决策中的重要性, 提出融合双分支动态偏好的会话推荐模型, 该模型考虑用户短时间内受环境影响的价格和兴趣偏好波动, 通过多尺度时序 Transformer 来学习用户动态价格偏好, 在软注意机制中添加反向位置编码提取用户动态兴趣偏好, 实验结果证明, 通过双分支聚合机制获取各个节点的嵌入, 利用多尺度对价格动态建模, 在学习兴趣偏好中添加反向位置编码对提高模型推荐准确性发挥了作用. 在未来的工作中, 笔者将更深入学习价格因素动态建模, 用其他方法进行动态价格建模, 另外可以考虑使用卷积对数据嵌入进行处理, 构建更高效精准的推荐模型.

参考文献

1 Wang SJ, Pasi G, Hu L, *et al*. The era of intelligent recommendation: Editorial on intelligent recommendation with advanced AI and learning. IEEE Intelligent Systems, 2020, 35(5): 3–6. [doi: 10.1109/MIS.2020.3026430]

2 Sarwar B, Karypis G, Konstan J, *et al*. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th International Conference on World Wide Web. Hong Kong: ACM, 2001. 285–295.

3 He XN, Liao LZ, Zhang HW, *et al*. Neural collaborative filtering. Proceedings of the 26th International Conference on World Wide Web. Perth: International World Wide Web Conferences Steering Committee, 2017. 173–182.

4 Wu S, Tang YY, Zhu YQ, *et al*. Session-based recommendation with graph neural networks. Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Honolulu: AAAI, 2019. 346–353.

5 Li J, Ren PJ, Chen ZM, *et al*. Neural attentive session-based recommendation. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. Singapore: ACM, 2017. 1419–1428.

6 Hidasi B, Karatzoglou A, Baltrunas L, *et al*. Session-based recommendations with recurrent neural networks. Proceedings of the 4th International Conference on Learning Representations. San Juan, 2016.

7 Wang ZY, Wei W, Cong G, *et al*. Global context enhanced graph neural networks for session-based recommendation. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. Xi'an: ACM, 2020. 169–178.

8 Tan QY, Zhang JW, Liu NH, *et al*. Dynamic memory based attention network for sequential recommendation. Proceedings of the 35th AAAI Conference on Artificial Intelligence. AAAI, 2021. 4384–4392.

9 Zhou HC, Tan QY, Huang X, *et al*. Temporal augmented graph neural networks for session-based recommendations. Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2021. 1798–1802.

10 Krishnamurthi L, Mazumdar T, Raj SP. Asymmetric response to price in consumer brand choice and purchase quantity decisions. Journal of Consumer Research, 1992, 19(3): 387–400. [doi: 10.1086/209309]

11 Chen SFS, Monroe KB, Lou YC. The effects of framing price promotion messages on consumers' perceptions and purchase intentions. Journal of Retailing, 1998, 74(3): 353–372. [doi: 10.1016/S0022-4359(99)80100-6]

12 Han SM, Gupta S, Lehmann DR. Consumer price sensitivity and price thresholds. Journal of Retailing, 2001, 77(4): 435–456. [doi: 10.1016/S0022-4359(01)00057-4]

13 Umberto P. Developing a price-sensitive recommender system to improve accuracy and business performance of ecommerce applications. International Journal of Electronic Commerce Studies, 2015, 6(1): 1–18.

14 Wu SW, Sun F, Zhang WT, *et al*. Graph neural networks in recommender systems: A survey. ACM Computing Surveys, 2023, 55(5): 97.

15 Liu Q, Zeng YF, Mokhosi R, *et al*. STAMP: Short-term attention/memory priority model for session-based recommendation. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. London: ACM, 2018. 1831–1839.

16 Yadati N, Nimishakavi M, Yadav P, *et al*. HyperGCN: A new method of training graph convolutional networks on hypergraphs. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 135.

17 Zhang XK, Xu B, Yang L, *et al*. Price DOES matter!:

Modeling price and interest preferences in session-based recommendation. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. Madrid: ACM, 2022. 1684–1693.

18 Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009, 42(8): 30–37. [doi: 10.1109/MC.2009.263]

19 高茂庭, 徐彬源. 基于循环神经网络的推荐算法. 计算机工程, 2019, 45(8): 198–202, 209.

20 Guo L, Yin HZ, Wang QY, *et al*. Streaming session-based recommendation. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage: ACM, 2019. 1569–1577.

21 Shi WJ, Rajkumar R. Point-GNN: Graph neural network for 3D object detection in a point cloud. Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 1708–1716.

22 Sun PZ, Zhang RF, Jiang Y, *et al*. Sparse R-CNN: End-to-end object detection with learnable proposals. Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 14449–14458.

23 Chen TW, Wong RCW. Handling information loss of graph neural networks for session-based recommendation. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2020. 1172–1180.

24 Ying R, He RN, Chen KF, *et al*. Graph convolutional neural networks for Web-scale recommender systems. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. London: ACM, 2018. 974–983.

25 van den Berg R, Kipf TN, Welling M. Graph convolutional matrix completion. arXiv:1706.02263, 2017.

26 Wang X, He XN, Wang M, *et al*. Neural graph collaborative filtering. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. Paris: ACM, 2019. 165–174.

27 Yu JL, Yin HZ, Li JD, *et al*. Self-supervised multi-channel hypergraph convolutional network for social recommendation. Proceedings of the Web Conference 2021. Ljubljana: ACM, 2021. 413–424.

28 Wu L, Sun PJ, Fu YJ, *et al*. A neural influence diffusion model for social recommendation. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. Paris: ACM, 2019. 235–244.

29 Wang JL, Ding KZ, Zhu ZW, *et al*. Session-based recommendation with hypergraph attention networks. Proceedings of the 2021 SIAM International Conference on Data Mining (SDM). SIAM, 2021. 82–90.

30 Greenstein-Messica A, Rokach L. Personal price aware multi-seller recommender system: Evidence from eBay. Knowledge-based Systems, 2018, 150: 14–26. [doi: 10.1016/j.knosys.2018.02.026]

31 Kitaev N, Kaiser Ł, Levskaya A. Reformer: The efficient Transformer. Proceedings of the 8th International Conference on Learning Representations. Addis Ababa: OpenReview.net, 2020.

32 Wang SN, Li BZ, Khabsa M, *et al*. Linformer: Self-attention with linear complexity. arXiv:2006.04768, 2020.

33 Yang YH, Huang C, Xia LH, *et al*. Multi-behavior hypergraph-enhanced Transformer for sequential recommendation. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. Washington: ACM, 2022. 2263–2274.

34 Radford A, Narasimhan K, Salimans T, *et al*. Improving language understanding by generative pre-training. https://www.mikecaptain.com/resources/pdf/GPT-1.pdf. (2023-10-19)[2023-10-25].

35 Devlin J, Chang MW, Lee K, *et al*. BERT: Pre-training of deep bidirectional Transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Minneapolis: Association for Computational Linguistics, 2019. 4171–4186.

36 Hendrycks D, Gimpel K. Gaussian error linear units (GELUs). arXiv:1606.08415, 2016.

37 Sun F, Liu J, Wu J, *et al*. BERT4Rec: Sequential recommendation with bidirectional encoder representations from Transformer. Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing: ACM, 2019. 1441–1450.

(校对责编: 孙君艳)