

融合项目直接关联实体传播的IEP-CKAN推荐方法

李新松^{1,2}, 徐 涛^{1,2}, 戴玉刚^{1,2}, 刘玉佳^{1,2}

¹西北民族大学语言与文化计算教育部重点实验室, 甘肃 兰州

²西北民族大学甘肃省民族语言文化智能信息处理重点实验室, 甘肃 兰州

收稿日期: 2024年5月23日; 录用日期: 2024年8月9日; 发布日期: 2024年8月27日

摘 要

面向推荐系统的协同知识感知注意网络(Collaborative Knowledge-Aware Attentive Network for Recommender Systems, 简称CKAN)没有优化与项目直接关联实体的嵌入表示, 且没有考虑不同传播层次的重要性以及用户点击物品间的关联信息, 使推荐实体不够准确。因此, 提出一种融合项目直接关联实体传播的推荐算法(Items are Directly Associated with Entity Propagation - Collaborative Knowledge-aware Attentive Network for Recommender Systems, 简称IEP-CKAN), 在IEP-CKAN中将优化与项目直接关联实体的嵌入表示, 并融入Ripple Net模型思想在利用波纹集信息时给予靠近中心节点的层次信息更大或更小的权重, 再通过无序编码器挖掘用户点击物品间的关联信息, 最终计算出推荐结果。实验结果表明, 与CKAN相比该方法在Last. FM数据集上将AUC和F1的值提高了1.1%和1.3%, 在Book-Crossing数据集上将AUC和F1的值提高了0.9%和1.0%, 证明该方法能够进一步提高推荐模型的性能。

关键词

推荐算法, 知识图谱, 注意力机制, 无序编码器, 嵌入优化

IEP-CKAN Recommendation Method Integrating Item Directly Related Entity Propagation

Xinsong Li^{1,2}, Tao Xu^{1,2}, Yugang Dai^{1,2}, Yujia Liu^{1,2}

¹Key Laboratory of Linguistic and Cultural Computing Ministry of Education, Northwest Minzu University, Lanzhou Gansu

²Key Laboratory of Minzu Languages and Cultures Intelligent Information Processing in Gansu Province, Northwest Minzu University, Lanzhou Gansu

文章引用: 李新松, 徐涛, 戴玉刚, 刘玉佳. 融合项目直接关联实体传播的 IEP-CKAN 推荐方法[J]. 人工智能与机器人研究, 2024, 13(3): 550-564. DOI: 10.12677/airr.2024.133057

Abstract

The Collaborative Knowledge-aware Attentive Network for Recommender Systems (CKAN) does not optimize the embedded representation of entities directly associated with the project. Moreover, the importance of different communication levels and the correlation information between items clicked by users are not considered, so the recommended entity is not accurate enough. Therefore, This paper proposes a recommendation algorithm (Items are Directly Associated with Entity Propagation - Collaborative Knowledge-aware Attentive) Network for Recommender Systems (IEP-CKAN for short) puts the embedded representation of entities directly related to the optimization project in IEP-CKAN, and incorporates the idea of Ripple Net model to give a greater or smaller weight to the hierarchical information near the central node when making use of the information of the ripple set. Then through the disordered encoder mining the user clicks between the items of the association information, the final calculation of the recommendation result. The experimental results show that compared with CKAN, the proposed method increases the values of AUC and F1 by 1.1% and 1.3 % in Last. FM dataset and 0.9% and 1.0% in Book-Crossing dataset, which proves that the proposed method can further improve the performance of the recommended model.

Keywords

Recommendation Algorithm, Knowledge Graph, Attention Mechanism, Unordered Encoder, Embedded Optimization

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

推荐系统是一种能够从海量数据中提取与用户匹配信息的智能工具，可有效应对信息过载问题，实现个性化推荐服务。推荐算法是推荐系统的一部分，是实现个性化推荐的核心技术。推荐算法分为基于内容的推荐、协同过滤推荐、混合推荐 3 类[1]。基于内容的推荐通过分析物品内容信息，提取用户对物品的兴趣偏好进行推荐，常被用于评分数据稀疏和物品内容特征非常强的场景[2]，但存在缺乏多样性、特征提取难度大等缺点。协同过滤算法利用相似用户之间具有相似兴趣偏好的特点，来发现用户对项目的潜在偏好，不需要对项目进行复杂的特征提取，但在数据稀疏、冷启动等方面仍然面临难题。基于内容的推荐算法和协同过滤推荐算法各有局限性，混合推荐算法通过整合多种推荐技术弥补单一推荐模型的不足。

Transformer 模型[3]自 2017 年提出以来引起了广泛的关注，并在自然语言处理和其他领域取得了显著的成就。在推荐领域中，Transformer 模型常被应用于序列建模推荐[4]-[7]，即将用户行为序列输入到 Transformer 模型中，学习到用户的兴趣和偏好并据此进行个性化推荐。

近年来，将知识图谱作为辅助信息引入到推荐算法的研究受到了广泛关注。知识图谱中所包含的丰富数据，特别是各类有关物品的知识，可以引入推荐算法，以更好地刻画物品特征，从而增强推荐效果

[8], 有效应对传统推荐算法存在的稀疏性和冷启动等问题。基于知识图谱的推荐算法可分为基于嵌入的推荐方式、基于路径的推荐方式、基于传播的推荐方式三大类[9]。

基于嵌入的方法[10]-[14]是知识图谱推荐系统中常用的技术,其核心思想是用低维向量表示知识图谱中的实体和关系。基于嵌入的推荐方法早期提出,并已广泛应用于词嵌入、信息检索和推荐等领域。基于路径的方法[15] [16]利用知识图谱中连接实体对的线性路径来提高推荐性能。基于传播的推荐方法[17]-[21]将实体和关系的语义表示和连接信息整合在一起,以充分利用知识图谱中的信息提供更好的推荐。Wang 等[17]提出的 CKAN 模型是当前先进的基于传播的推荐方法,它将用户-项目交互和知识关联结合在知识图谱中,进行异构传播,产生多个跳跃的涟漪集合,捕捉用户潜在的兴趣爱好。

虽然 CKAN 通过协同传播的方式丰富了用户和项目的嵌入表示并取得了很好的推荐效果,但其并未对项目直接关联实体的嵌入表示进行优化,忽略了多层邻居的不同重要性以及用户点击物品之间的依赖关系,导致项目推荐的准确性不够。为了进一步提高推荐效果并优化项目的实体表示,本研究提出了一种名为 IEP-CKAN 的方法。在 IEP-CKAN 中,将优化项目直接关联实体的嵌入表示,并融入 Ripple Net 模型思想,在预测层利用波纹集信息时由内向外给予递增或递减的层次加权,再通过 Transformer 模型无序编码器捕捉对用户点击物品间关联信息,从而优化用户嵌入表示的准确性,最终计算出推荐结果。本文的主要贡献可归纳为以下几点:

- 1) 采用层次加权方法,根据推荐场景给予波纹集不同权重与层次增减因子,提高推荐效果。
- 2) 提出了直接关联实体传播的方法,探究了 CKAN 模型中直接关联实体对推荐结果的影响。

3) 将 Transformer 模型融入知识图谱推荐方法中,证明了捕捉用户点击物品间依赖关系对推荐结果的积极影响。

2. 相关工作

2.1. 基于知识图谱的推荐算法

Trans E [10]、Trans H [11]、Trans R [12]、Trans D [13]等利用基于距离的计分函数,通过 Trans 系列的嵌入方法学习到知识图谱中实体和关系的表示,可以用来补充表示用户和项目。Zhang 等人[14]提出了协同知识库嵌入(Collaborative Knowledge-Base Embedding, CKE)框架,结合了协同过滤和知识图嵌入的方法,用于提取项目特征并捕捉用户与项目之间的隐性关系。基于嵌入的推荐方法在知识图谱中的实体和关系学习低维特征向量的同时保留了知识图谱中原有的结构和语义信息,但忽略了由路径连接的实体之间的语义关联[9]。

Yu 等人[15]提出了 Hete-MF 方法,结合正则化和矩阵分解技术,提取不同元路径,计算路径中各项目之间的相似度,达到推荐效果的更高质量。Sun 等人[16]于 2018 年提出的 RKGE 模型无需手动定义元路径,就能自动挖掘用户与项目之间的路径关系。基于路径的推荐方法充分利用了知识图谱中丰富的关系信息并具有良好的可解释性,但存在着计算复杂度高、路径长度选择困难等问题。

在基于传播的推荐方式中,Ripple Net 模型[18]引入了一个自动发现用户潜在层次兴趣的偏好性传播概念。该模型提取与用户节点相连的 N-hop 实体节点,并使用这些邻居实体节点的嵌入表示来更新用户的嵌入表示,但其对关系信息的重视不足。KGCN [19]模型与 KGNN-LS [20]模型采用图卷积神经网络通过知识图谱中的邻居信息获得项目嵌入,取得了较好的推荐推荐效果,但两者都忽略了与项目相关的协同信号。KGAT [21]模型将用户兴趣图(UIG)和知识图谱(KG)结合到一个同构统一图(CKG)中,并通过图卷积神经网络递归的在 CKG 中传播以补充实体嵌入,但当有新的交互用户加入时就需要重建 CKG 并重新训练模型,计算成本过高。

2.2. 基于 Transformer 的推荐算法

现有的基于 Transformer 的推荐算法中常利用 Transformer 中位置编码特性对应用户行为序列信息, 预测出下一个用户感兴趣的物品。BST 模型[4]利用 Transformer 结构来捕获用户行为序列中的信息与用户兴趣的动态变化性。BERT4Rec 模型[5]旨在利用 Transformer 架构的预训练模型来进行个性化推荐, 利用 BERT 模型对用户行为序列进行掩码语言建模, 从而学习用户行为的表示。凤丽洲等[6]针对新闻的偏好时序性的特点, 引入 Transformer 捕捉用户点击新闻间的关联信息及用户兴趣随时间的变化情况。王明虎等[7]将图模型学习到的各个商品的特征表示以序列的形式输入 Transformer 编码层, 从用户的行为信息出发, 利用商品关联有向图构建用户对应的商品序列。但在一些推荐场景中无法获得具有时序性的用户点击物品的历史信息或此类信息对推荐性能影响较小且增加模型负担, 为此, 本文提出了基于 Transformer 的无序编码器, 即通过多头注意力机制、前馈神经网络等挖掘用户点击物品间的关联信息, 从而优化用户嵌入表示。

3. IEP-CKAN 模型

IEP-CKAN 模型由基于 Transformer 无序编码器的嵌入模块与基于知识图谱传播的嵌入模块构成。基于 Transformer 无序编码器的嵌入模块由 m 个 Transformer 无序编码器组成, Transformer 无序编码器由多头注意力机制层与非线性前馈神经网络层构成。基于知识图谱传播的嵌入模块由异构传播层、知识感知注意嵌入层和预测层构成。如图 1 所示。图中, 预测层的聚合器部分仅体现了实验效果最佳的融合层次加权的拼接聚合器($Agg_{concat+}$)。

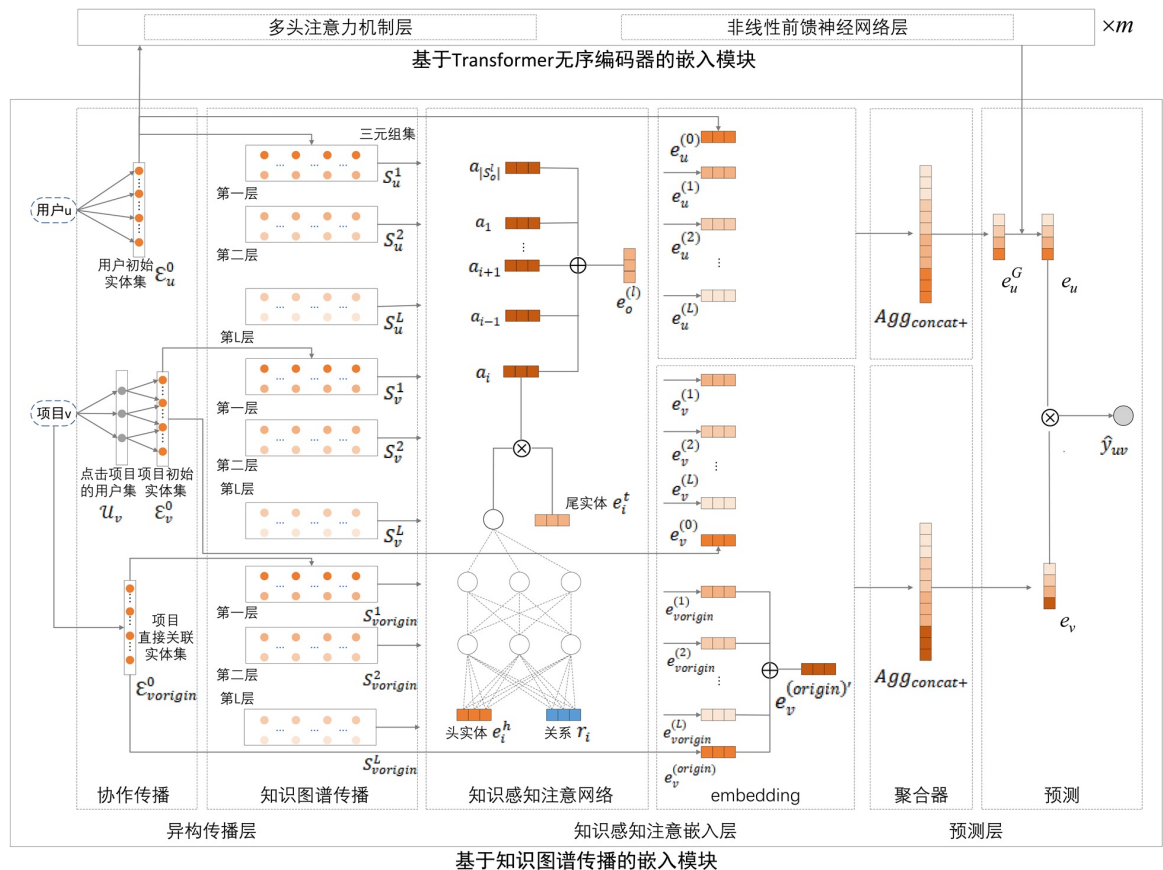


Figure 1. Architecture diagram of IEP-CKAN model

图 1. IEP-CKAN 模型架构图

3.1. 问题描述

设有 M 个用户 $u_1, u_2, \dots, u_M$ 和 N 个项目 $v_1, v_2, \dots, v_N$ ，在特定场景中，根据用户历史行为记录如点击、观看、购买等，可以得到用户 - 项目交互矩阵 $Y \in \mathbb{R}^{M \times N}$ ，当 $y_{uv} = 1$ 时表示用户与项目进行过交互，当 $y_{uv} = 0$ 时表示用户与项目未进行交互。将知识图谱定义为 $G = \{(h, r, t) | h, t \in \mathcal{E}, r \in \mathcal{R}\}$ ，三元组 (h, r, t) 包含头实体 h 、尾实体 t 与关系 r ， $\mathcal{E}$ 是知识图谱中的实体集， $\mathcal{R}$ 是知识图谱中的关系集， $\mathcal{R}$ 既包含规范方向，也包含逆方向。使用集合 $A = \{(v, e) | v \in \mathcal{V}, e \in \mathcal{E}\}$ 表示项目与知识图谱中实体的对齐关系。

已有用户交互矩阵 Y 和知识图谱 G ，目前的任务是预测用户与其没有交互过的项目进行交互的概率。进一步表示为任务的目标是学习预测函数 $\hat{y}_{uv} = F(u, v | \Theta, Y, G)$ ，其中 $\hat{y}_{uv}$ 表示用户与项目交互的预测概率， Θ 表示函数的模型参数。

3.2. 算法描述

3.2.1. 异构传播层

在异构传播层中包含协作传播和知识图谱传播两个模块。

与 CKAN 模型中相似，协作传播模块的核心观念是用户过去互动过的项目在一定程度上能够反映其偏好。使用用户 u 的相关项目来表示用户 u ，通过用户 u 历史上的交互，可以得到相关项目的集合，将其转化为初始种子集后再利用知识图谱进行传播。用户 u 的初始实体集定义如下：

$$\mathcal{E}_u^0 = \{e | (v, e) \in A \text{ and } v \in \{v | y_{uv} = 1\}\} \quad (1)$$

类似地，与同一项目进行过交互的用户也可以因为相似的行为偏好对项目的特征表示做出贡献。将同一用户交互过的项目定义为协作邻居，并将项目 v 的协作项目集表示如下：

$$\mathcal{V}_v = \{v_u | u \in \{u | y_{uv} = 1\} \text{ and } y_u v_u = 1\} \quad (2)$$

综合协作项目集和对齐集，初始化项目 v 的实体集定义如下：

$$\mathcal{E}_v^0 = \{e | (v_u, e) \in A \text{ and } v_u \in \mathcal{V}_v\} \quad (3)$$

知识图谱传播模块通过在知识图谱中的链接上进行传播，递归地扩展具有补充边信息的用户和项目的表示。通过沿着知识图谱中的链路传播，可以从初始实体集中获取不同距离的扩展实体集和三元组集，进一步扩展潜在的有效表示。得到的集合如下所示，其中 o 代表用户 u 或项目 v 。

$$\mathcal{E}_o^l = \{t | (h, r, t) \in G \text{ and } h \in \mathcal{E}_o^{l-1}\}, l = 1, 2, \dots, L \quad (4)$$

$$S_o^l = \{(h, r, t) | (h, r, t) \in G \text{ and } h \in \mathcal{E}_o^{l-1}\}, l = 1, 2, \dots, L \quad (5)$$

与 CKAN 不同的是，本文关注于与项目直接关联的实体，集合如下：

$$\mathcal{E}_{v\text{-origin}}^0 = \{e | (e, v) \in A\} \quad (6)$$

项目直接关联实体是潜在语义空间中与项目 v 本身最接近的节点。因此，本文将项目直接关联实体在知识图谱中传播，传播方式与上文 $\mathcal{E}_u^0$ 、 $\mathcal{E}_v^0$ 类似，得到集合如下：

$$\mathcal{E}_{v\text{-origin}}^l = \{t | (h, r, t) \in G \text{ and } h \in \mathcal{E}_{v\text{-origin}}^{l-1}\}, l = 1, 2, \dots, L \quad (7)$$

$$S_{v\text{-origin}}^l = \{(h, r, t) | (h, r, t) \in G \text{ and } h \in \mathcal{E}_{v\text{-origin}}^{l-1}\}, l = 1, 2, \dots, L \quad (8)$$

3.2.2. 知识感知注意嵌入层

在知识感知注意嵌入层中，考虑 (h, r, t) 是第 l 层三元组集中的第 i 个三元组，建立一个尾部实体的注

意嵌入 a_i ，如下：

$$a_i = \pi(e_i^h, r_i) e_i^t \quad (9)$$

其中 e_i^h 是头部实体的嵌入， r_i 是关系嵌入， e_i^t 是第 i 个元组的尾部实体的嵌入。 $\pi(e_i^h, r_i)$ 控制头部实体的注意力权重以及头部实体和尾部实体之间的关系。

最终，可以得到用户 u 或项目 v 的第 l 层三元组集的表示：

$$e_o^{(l)} = \sum_{i=1}^{|S_o^l|} a_i^{(o)}, l = 1, 2, \dots, L \quad (10)$$

其中下标和上标 o 是符号 u 或 v 的统一占位符， $|S_o^l|$ 是集合 S_o^l 中三元组的数量。

初始实体集中包含着较高价值的信息，因此为用户和项目添加初始实体集的表示：

$$e_o^{(0)} = \frac{\sum_{e \in \mathcal{E}_o^0} e}{|\mathcal{E}_o^0|} \quad (11)$$

在异构传播层得到了项目直接关联实体的传播集合 $\mathcal{E}_{v\text{-origin}}^l$ 与 $S_{v\text{-origin}}^l$ ，使用前文中知识感知的潜在嵌入方法，得到项目直接关联实体的第 l 层三元组集的表示：

$$e_{v\text{-origin}}^{(l)} = \sum_{i=1}^{|S_o^l|} a_i^{(v\text{-origin})}, l = 1, 2, \dots, L \quad (12)$$

项目直接关联实体嵌入表示如下：

$$e_v^{(\text{origin})} = \frac{\sum_{e \in \{e | (e, v) \in A\}} e}{|\{e | (e, v) \in A\}|} \quad (13)$$

聚合项目直接关联实体嵌入表示与其传播信息，得到优化后的项目直接关联实体嵌入表示，方法如下：

$$e_v^{(\text{origin})'} = e_v^{(\text{origin})} + \sum_{i=1}^L e_{v\text{-origin}}^{(i)} \quad (14)$$

将用户、项目表示集公式化，如下所示：

$$\tau_u = \{e_u^{(0)}, e_u^{(1)}, \dots, e_u^{(L)}\} \quad (15)$$

$$\tau_v = \{e_v^{(\text{origin})'}, e_v^{(0)}, e_v^{(1)}, \dots, e_v^{(L)}\} \quad (16)$$

3.2.3. 多头注意力机制层

由于(1)中 $\mathcal{E}_u^0$ 中的物品均能体现用户的兴趣，因此集合 $\mathcal{E}_u^0$ 可以代表用户的兴趣类， U_n 是集合 $\mathcal{E}_u^0$ 对应的矩阵，包含着用户兴趣类的信息，本文将矩阵 U_n 称为用户兴趣类向量矩阵。矩阵 U_n 表示如下：

$$U_n = \{e_1^u, e_1^u, e_1^u, \dots, e_n^u\} \quad (17)$$

其中 e_i^u 表示用户点击过物品所对应知识图谱中的实体向量， n 为用户点击的物品数。

具体来说，Transformer 输入为用户兴趣类向量矩阵 U_n ，对矩阵 U_n 进行线性变换，分别计算查询 Q_i 、键 K_i 、值 V_i 矩阵：

$$Q_i = U_n W_i^Q \quad (18)$$

$$K_i = U_n W_i^K \quad (19)$$

$$V_i = U_n W_i^V \quad (20)$$

其中, W_i^Q 、 W_i^K 、 W_i^V 分别为相应的矩阵参数。

则第 i 头注意力机制的输出 Z_i 为:

$$Z_i = V_i \text{softmax} \left(\frac{Q_i K_i^T}{\sqrt{d_k}} \right) \quad (21)$$

多头注意力机制的输出 Z 为:

$$Z = \text{Concat}(Z_1, Z_2, Z_3, \dots, Z_h) W_o \quad (22)$$

其中 W_o 为线性变换矩阵参数, h 为多头个数, Concat 为拼接操作。

Dropout 函数与和残差连接可以有效缓解过拟合与梯度下降的影响, 因此加入 Dropout 函数与和残差连接, 如图 2 所示。多头注意力机制层的输出为:

$$O^h = \text{LayerNorm}(U + \text{Dropout}(Z)) \quad (23)$$

其中, Layer Norm 为归一化层。

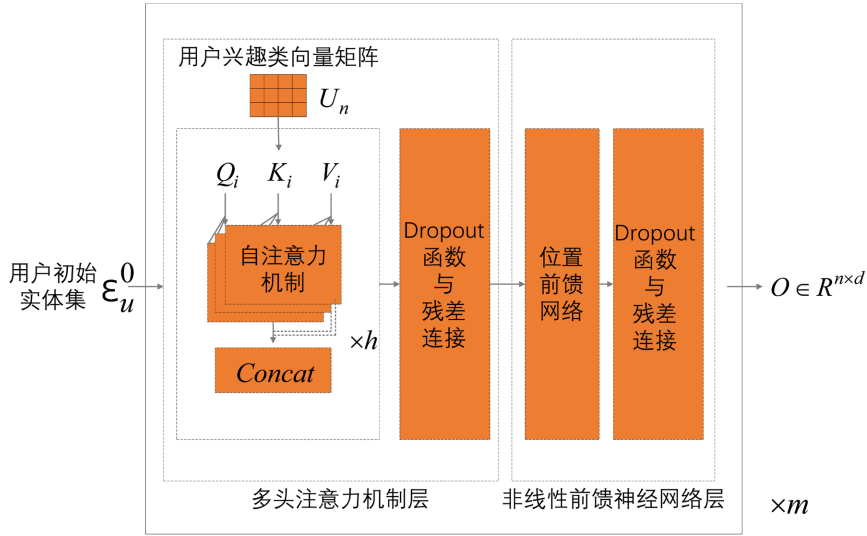


Figure 2. Embedded module diagram based on Transformer disordered encoder
图 2. 基于 Transformer 无序编码器的嵌入模块图

3.2.4. 非线性前馈神经网络层

为了增强非线性建模能力, 加入非线性前馈神经网络, 计算如下:

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (24)$$

其中, W_1 、 b_1 、 W_2 、 b_2 为可训练参数。

将 O^h 输入到非线性前馈神经网络中, 再通过 Dropout 函数与和残差连接得到编码器的输出:

$$O = \text{LayerNorm}(O^h + \text{Dropout}(\text{FFN}(O^h))) \quad (25)$$

其中, $O \in R^{n \times d}$, d 为嵌入维度。

3.2.5. 预测层

在 Ripple Net 模型中, 用户点击过的实体沿着知识图谱的边进行逐渐衰减的扩散, 类似于水中的涟

滴, Ripple Net 模型的这种传播方式体现了不同传播层次的重要性差异。本文将该思想融入到 CKAN 模型中并加以改进, 具体方法如下: 在知识感知注意嵌入层中, 获得了用户和项目在知识图谱中每个层次的表示。在稀疏性较小的数据集中, 可以给予靠近中心节点的层次更大的权重, 以强调其重要性。而在稀疏性较大的数据集中, 需要给予外围层次更大的权重, 以充分利用传播的信息, 从而优化用户和项目的嵌入表示。

本文中, 提出了融合层次加权的累加聚合器与融合层次加权的拼接聚合器, 以实现由内向外给予层次表示递增或递减的层次加权。

通过下述方法, 得到不同层次的权重:

$$\alpha_0 = \pm 1, \alpha_1 = c \times 1 \times \alpha_0, \alpha_2 = c \times 2 \times \alpha_0, \dots, \alpha_n = c \times n \times \alpha_0 \quad (26)$$

其中 c 是可训练参数, α_0 的值取为 1 或 -1, 当 $\alpha_0 = 1$ 时表示给予外围层次更大的权重, 当 $\alpha_0 = -1$ 时表示给予靠近中心节点层次更大的权重; $n = |\tau_o|$ 。

融合层次加权的累加聚合器($\text{Agg}_{\text{sum}+}$): 对获得的层次表示乘以对应权重, 然后求和并进行非线性变换。方法如下:

$$\text{Agg}_{\text{sum}+} = \sigma \left(W_a \times \left((1 + a_1) \times e_o^{(i_1)} + (1 + a_2) \times e_o^{(i_2)} + \dots + (1 + a_n) \times e_o^{(i_n)} \right) + b_a \right) \quad (27)$$

其中 $e_o^{(i_k)} \in \tau_o$, b_a 是可训练参数, W_a 是可训练权重矩阵。

融合层次加权的拼接聚合器($\text{Agg}_{\text{concat}+}$): 对获得的层次表示乘以对应权重, 然后拼接并进行非线性变换。方法如下:

$$\text{Agg}_{\text{concat}+} = \sigma \left(W_a \times \left((1 + a_1) \times e_o^{(i_1)} \parallel (1 + a_2) \times e_o^{(i_2)} \parallel \dots \parallel (1 + a_n) \times e_o^{(i_n)} \right) + b_a \right) \quad (28)$$

其中 $\parallel$ 表示拼接操作。

需要注意的是, 拼接聚合器需要保证用户与项目拥有相同数量的表示来进行最终的预测计算, 因此去掉项目中的 $e_o^{(0)}$ 来保持数量一致。

算法的关键步骤如下:

算法 1 层次加权算法

输入: user_embeddings 用户表示集, item_embeddings 项目表示集, c 层次加权参数

输出: e_u 用户表示向量, e_v 项目表示向量

1. $n = \text{item_embeddings.length}$, 创建数组 $a[n]$;

2. $a[0] = 1$ or -1

3. $e_u = \text{user_embeddings}[0]$

4. $e_v = \text{item_embeddings}[0]$

5. for $i = 1$ to $n - 1$ do

$a[i] = c \cdot i \cdot a[0]$

end for

6. if $\text{Agg} = \text{sum} + \text{do}$

for $i = 1$ to $\text{user_embeddings.length}$ do

$e_u = a[i] \cdot \text{user_embeddings}[i] + e_u$

end for

for $i = 1$ to $\text{item_embeddings.length}$ do

$e_v = a[i] \cdot \text{item_embeddings}[i] + e_v$


```

end for
end if
if Agg = concat + do
for  $i = 1$  to  $user\_embeddings.length$  do
 $e_u = a[i] \cdot user\_embeddings[i] \parallel e_u$ 
end for
for  $i = 1$  to  $item\_embeddings.length$  do
 $e_v = a[i] \cdot item\_embeddings[i] \parallel e_v$ 
end for
end if

```

为体现层次加权方法的有效性,本文在后续实验中将上述两种聚合器与 CKAN 模型中提出的累加聚合器、拼接聚合器、池聚合器进行对比。

累加聚合器:

$$Agg_{sum} = \sigma \left(W_a \times \sum_{e_o \in \tau_o} e_o + b_a \right) \quad (29)$$

拼接聚合器:

$$Agg_{concat} = \sigma \left(W_a \times \left(e_o^{(i_1)} \parallel e_o^{(i_2)} \parallel \dots \parallel e_o^{(i_n)} \right) + b_a \right) \quad (30)$$

池聚合器:

$$Agg_{pool} = \sigma \left(W_a \times pool_{\max}(\tau_o) + b_a \right) \quad (31)$$

通过基于 Transformer 无序编码器的嵌入模块得到矩阵 O , 为了得到用户的嵌入表示, 将矩阵 O 中的项目信息取平均值, 得到:

$$e_u^T = \frac{\sum_{i \leq n} o_i}{n} \quad (32)$$

其中, e_u^T 表示经过 Transformer 模块后得到的用户嵌入表示, o_i 表示用户点击物品经过无序编码器后的表示向量。

使用 e_u^G 表示经过聚合器后得到的用户嵌入表示, 聚合 e_u^T 与 e_u^G 信息得到最终用户嵌入表示:

$$e_u = \frac{e_u^T + e_u^G}{2} \quad (33)$$

其中, e_u 表示最终用户嵌入表示, e_u^T 、 e_u^G 的信息聚合方式有待进一步研究改进。

使用 e_u 与 e_v 表示用户与项目的聚量。最终通过向量的内积运算来预测用户对项目的偏好得分:

$$\hat{y}_{uv} = e_u^T e_v \quad (34)$$

3.3. 损失函数

采用与 CKAN 相同的损失函数与采样方法。损失函数如下:

$$L = \sum_{u \in \mathcal{U}} \left(\sum_{v \in \{v | (u,v) \in \mathcal{P}^+\}} \mathcal{J}(y_{uv}, \hat{y}_{uv}) - \sum_{v \in \{v | (u,v) \in \mathcal{P}^-\}} \mathcal{T}(y_{uv}, \hat{y}_{uv}) \right) + \lambda \|\Theta\|_2^2 \quad (35)$$

其中, $\mathcal{L}$ 为交叉熵损失, $\mathcal{P}^+$ 为正用户 - 项目集, $\mathcal{P}^-$ 为负用户 - 项目集。
 $\Theta = \{\varepsilon, \mathcal{R}, \alpha_0, c, W_l, b_l, \forall l \in \{0, 1, 2, a\}\}$ 是模型参数的集合, ε 和 $\mathcal{R}$ 是所有实体和关系的嵌入表, $\|\Theta\|_2^2$ 是由 λ 参数化的 L2 正则化。

4. 实验结果与分析

4.1. 实验数据集

本文在 Last. FM 数据集与 Book-Crossing 数据集上进行了实验。对应数据集的知识图谱数据是从 CKAN 公开的预处理知识图谱获得的, 两个数据集经过预处理后的统计数据如表 1 所示。

Table 1. Statistics for the two datasets
表 1. 两个数据集的统计数据

数据集	Last. FM	Book-Crossing
用户数	1872	17,860
项目数	3846	14,967
交互记录数	42,346	139,746
用户平均交互数	23	8
KG 实体数	9366	77,903
KG 关系数	60	25
KG 三元组数	15,518	151,500
项目平均链接数	4	10

4.2. 实验设计

Table 2. Values of hyperparameters
表 2. 超参数取值

超参数	Last. FM	Book-Crossing
d	64	64
$K-u$	8	16
$K-i$	64	64
L	3	2
λ	$1e-5$	$1e-5$
batch	2 048	2 048
η	$2e-3$	$2e-3$
c	0.1	0.1
α_0	-1	1
h	8	8
m	1	1

为了验证提出模型的有效性，选取了一些具有代表性的推荐模型进行比较：

Ripple Net [11]：使用用户的历史记录来挖掘用户对知识图谱实体的潜在高阶偏好。

KGCN [14]：将 GCN 方法扩展到知识图谱，通过有选择性和有偏向性地聚合邻域信息。

KGNN-LS [16]：将异构的知识图谱转换为用户特定的加权图，使用标签平滑正则化方法在图神经网络中计算个性化的项目嵌入。

KGAT [17]：将用户兴趣图(UIG)和知识图谱(KG)结合到一个同构统一图(CKG)中。

CKAN [15]：使用协同传播丰富用户和项目的 0-hop 的嵌入表示，并利用注意力机制有效聚合用户与项目的信息。

采用 AUC 和 F1 来评价模型的性能，对于每个数据集，按照 6:2:2 的比例随机构成训练集、验证集和测试集。

本模型的超参数取值如表 2 所示。

其中， d 表示实体和关系的嵌入维度； $K-u$ 表示用户三元组的数量； $K-i$ 表示项目三元组的数量； L 表示图层深度； λ 表示正则化的参数；batch 表示批处理大小； η 表示学习率； α_0 表示层次增减因子； c 表示层次加权参数； h 表示注意力头数； m 表示无序编码器个数。

4.3. 结果与分析

将提出的模型与其他不同模型进行了对比，并在表 3 中列出了各模型在不同数据集上的 AUC 和 F1 指标结果。

Table 3. Results of model experiments

表 3. 超参数取值模型实验结果

模型	Last. FM		Book-Crossing	
	AUC	F1	AUC	F1
Ripple Net	0.776	0.702	0.721	0.647
KGCN	0.802	0.708	0.684	0.631
KGNN-LS	0.805	0.722	0.676	0.631
KGAT	0.829	0.742	0.731	0.654
CKAN	0.842	0.769	0.753	0.673
IEP-CKAN	0.853	0.782	0.762	0.683

从表 3 中可以看出，IEP-CKAN 模型的实验结果是普遍优于对比实验的，在 Last. FM 以及 Book-Crossing 两个数据集上性能都有所提升。与上述模型相比，IEP-CKAN 模型对推荐项目的图谱信息进行了有效的利用，对项目直接关联实体的嵌入表示进行了优化，通过层级加权与挖掘用户点击物品间关联信息的方式使推荐效果得到了进一步的提高。本文所提出的模型相比 CKAN 模型，在 Last. FM 数据集上 AUC 与 F1 分别提升了 1.1%与 1.3%，在 Book-Crossing 数据集上 AUC 与 F1 分别提升了 0.9%与 1.0%。

4.4. 参数分析

1) 层次加权参数 c 取值分别为 0.025、0.05、0.1、0.15、0.2，在 Last. FM 以及 Book-Crossing 两个数据集上对应的 AUC 取值如表 4 所示。当层次加权参数 c 取值为 0.1 时，模型在 Last. FM 与 Book-Crossing 数据集上效果最好。可以观察到层次加权参数 c 的变化对 Last. FM 数据集影响相对较大，对 Book-Crossing 影响相对较小，这是由于 Last. FM 的最佳传播层数是 3，Book-Crossing 的最佳传播层数是 2。由此可知，

层次权重方法在最佳传播层数大的数据集上效果更好。

Table 4. Influence of hierarchical weighted parameters on recommendation quality

表 4. 层次加权参数对推荐质量的影响

层次加权参数	Last. FM	Book-Crossing
0.025	0.849	0.760
0.05	0.852	0.761
0.1	0.853	0.762
0.15	0.851	0.760
0.2	0.848	0.759

2) 层次增减因子 α_0 取值为 1、-1，在 Last. FM 以及 Book-Crossing 两个数据集上对应的 AUC 取值如表 5 所示。层次增减因子 α_0 在 Last. FM 数据集与 Book-Crossing 数据集上的最佳值分别是 -1、1。出现这样的原因可能是 Last. FM 数据集稀疏性较小，实体的最初嵌入信息更加准确，因此可以适当减少涟漪集的权重提高实体嵌入的准确性。而 Book-Crossing 数据集稀疏性较大，需要适当增加涟漪集的权重丰富实体信息，以提高实体嵌入的准确性。

Table 5. The influence of level increase and decrease factors on recommendation quality

表 5. 层次增减因子对推荐质量的影响

层次增减因子	Last. FM	Book-Crossing
1	0.851	0.762
-1	0.853	0.759

3) 五种聚合器在 Last. FM 以及 Book-Crossing 两个数据集上对应的 AUC 取值如表 6 所示。融合层次加权的累加聚合器在 Book-Crossing 数据集上取得了良好的效果，但在 Last. FM 数据集上效果欠佳，这可能是由于两种数据集在数据稀疏性和数据准确性上的差异造成的。融合层次加权的拼接聚合器在 Last. FM 数据集与 Book-Crossing 数据集上均有着更好的效果，这可能是由于使用融合层次加权的拼接聚合器可以在嵌入向量中保留更多的信息。

Table 6. Effects of different aggregators on recommendation quality

表 6. 不同聚合器对推荐质量的影响

聚合器	Last. FM	Book-Crossing
Agg _{sum+}	0.811	0.733
Agg _{concat+}	0.853	0.762
Agg _{sum}	0.824	0.721
Agg _{concat}	0.847	0.754
Agg _{pool}	0.827	0.735

4) Transformer 无序编码器个数 m 在 Last. FM 以及 Book-Crossing 两个数据集上对应的 AUC 取值如表 7 所示。Transformer 无序编码器个数在两个数据集上取值为 1 时有着良好的推荐效果，这可能是由于较多的无序编码器个数会使复杂度过高，同时使用户兴趣类中的项目信息丢失，导致性能下降。

Table 7. Effect of number of unordered encoders on recommendation quality
表 7. 无序编码器个数对推荐质量的影响

无序编码器个数	Last. FM	Book-Crossing
1	0.853	0.762
2	0.841	0.756
3	0.835	0.753
4	0.843	0.752

4.5. 消融实验

为验证项目直接关联实体传播和融合层次加权方法对推荐任务的有效性, 本文分别对项目直接关联实体传播模块(M1)、融合层次加权方法模块(M2)和基于 Transformer 无序编码器的嵌入模块(M3)进行消融实验, 实验结果如表 8 所示(“×”表示未应用, “✓”表示应用)。

从实验中可以看出, 融合层次权重方法在 Last. FM 数据集上将 AUC 与 F1 提高了 0.4%、0.5%, 在 Book-Crossing 数据集上将 AUC 提高了 0.1%, 表明层次权重方法能够识别传播过程中不同层次的重要性, 提高推荐效果。加入项目直接关联实体传播的方法在 Last. FM 数据集上将 AUC 与 F1 提高了 0.6%、0.6%, 在 Book-Crossing 数据集上将 AUC 与 F1 提高了 0.4%、0.4%, 表明加入项目直接关联实体传播的方法能够优化项目的嵌入表示, 从而使推荐效果得到提升。基于 Transformer 无序编码器的嵌入方法在 Last. FM 数据集上将 AUC 与 F1 提高了 0.3%、0.4%, 在 Book-Crossing 数据集上将 AUC 与 F1 提高了 0.5%、0.3%, 表明捕捉用户点击物品间关联信息能够使用户嵌入表示更加准确, 从而提高推荐效果。说明本文提出的推荐算法可以更好地捕捉项目信息, 注意到不同层次在传播中的重要性, 并有效捕捉用户点击物品间关联信息, 提供了基于传播的推荐方式的优化思路。

Table 8. Results of ablation experiment
表 8. 消融实验结果

M1	M2	M3	Last. FM		Book-Crossing	
			AUC	F1	AUC	F1
×	×	×	0.842	0.769	0.753	0.673
✓	×	×	0.848	0.775	0.757	0.677
×	✓	×	0.846	0.774	0.754	0.673
×	×	✓	0.845	0.773	0.758	0.676
✓	✓	✓	0.853	0.782	0.762	0.683

5. 结束语

本文针对 CKAN 模型中未能有效利用项目直接关联实体信息, 未能关注不同传播层次重要性以及未能捕捉用户点击物品间关联信息的问题, 提出了 IEP-CKAN 模型加以解决。在 IEP-CKAN 模型中, 采用异构传播的方式优化项目直接关联实体的嵌入表示。通过这种方式, 模型能够更全面地理解实体之间的语义关联, 从而提高项目的表示质量和准确性。此外, IEP-CKAN 模型还根据数据集的特点, 给予波纹集层次信息不同的影响, 对不同层次传播信息进行加权处理, 关键层次将得到更多的重视, 更好地捕捉传播过程中不同层次的重要性。最后, 使用 Transformer 无序编码器捕捉用户点击物品间的关联信息, 挖

掘用户兴趣类信息,从而优化用户嵌入表示。未来的研究中将尝试降低算法复杂度以提高模型运算效率,并将层次加权方法与基于 Transformer 无序编码器的嵌入方法应用到其他推荐模型中。

基金项目

中央高校基本科研业务费“科研创新平台能力提升专项项目”(3192024125-02)。

参考文献

- [1] 刘华玲, 马俊, 张国祥. 基于深度学习的内容推荐算法研究综述[J]. 计算机工程, 2021, 47(7): 1-12.
- [2] 王俊淑, 张国明, 胡斌. 基于深度学习的推荐算法研究综述[J]. 南京师范大学学报(工程技术版), 2018, 18(4): 33-43.
- [3] Vaswani, A., Shazeer, N., Parmar, N., et al. (2017) Attention Is All You Need. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*, MIT Press, Long Beach, 6000-6010.
- [4] Chen, Q., Zhao, H., Li, W., Huang, P. and Ou, W. (2019) Behavior Sequence Transformer for E-Commerce Recommendation in Alibaba. *Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data*, Anchorage, 5 August 2019, 1-4. <https://doi.org/10.1145/3326937.3341261>
- [5] Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., et al. (2019) BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, Beijing, 3-7 November 2019, 1441-1450. <https://doi.org/10.1145/3357384.3357895>
- [6] 凤丽洲, 杨阳, 王友卫, 等. 基于 Transformer 和知识图谱的新闻推荐新方法[J]. 浙江大学学报(工学版), 2023, 57(1): 133-143.
- [7] 王明虎, 石智奎, 苏佳, 等. 基于 RoBERTa 和图增强 Transformer 的序列推荐方法[J]. 计算机工程, 2024, 50(4): 121-131.
- [8] 程开原, 姚俊萍, 李晓军, 等. 时态网络中知识图谱推荐: 关键技术与研究进展[J]. 中国电子科学研究院学报, 2021, 16(2): 174-183.
- [9] 罗承天, 叶霞. 基于知识图谱的推荐算法研究综述[J]. 计算机工程与应用, 2023, 59(1): 49-60.
- [10] Bordes, A., Usunier, N., Garcia-Duran, A., et al. (2013) Translating Embeddings for Modeling Multi-Relational Data. In: *Advances in Neural Information Processing Systems*, MIT Press, Vancouver, 2787-2795.
- [11] Wang, Z., Zhang, J., Feng, J. and Chen, Z. (2014) Knowledge Graph Embedding by Translating on Hyperplanes. *Proceedings of the AAAI Conference on Artificial Intelligence*, **28**, 1112-1119. <https://doi.org/10.1609/aaai.v28i1.8870>
- [12] Lin, Y., Liu, Z., Sun, M., Liu, Y. and Zhu, X. (2015) Learning Entity and Relation Embeddings for Knowledge Graph Completion. *Proceedings of the AAAI Conference on Artificial Intelligence*, **29**, 2181-2187. <https://doi.org/10.1609/aaai.v29i1.9491>
- [13] Ji, G., He, S., Xu, L., Liu, K. and Zhao, J. (2015) Knowledge Graph Embedding via Dynamic Mapping Matrix. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Volume 1, 687-696. <https://doi.org/10.3115/v1/p15-1067>
- [14] Zhang, F., Yuan, N.J., Lian, D., Xie, X. and Ma, W. (2016) Collaborative Knowledge Base Embedding for Recommender Systems. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, 13-17 August 2016, 353-362. <https://doi.org/10.1145/2939672.2939673>
- [15] Yu, X., Ren, X., Gu, Q., et al. (2013) Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks. *Proceedings of IJCAI-HINA*, Beijing, 3-9 August 2013, 1-6.
- [16] Sun, Z., Yang, J., Zhang, J., Bozzon, A., Huang, L. and Xu, C. (2018) Recurrent Knowledge Graph Embedding for Effective Recommendation. *Proceedings of the 12th ACM Conference on Recommender Systems*, Vancouver, 2 October 2018, 297-305. <https://doi.org/10.1145/3240323.3240361>
- [17] Wang, Z., Lin, G., Tan, H., Chen, Q. and Liu, X. (2020) CKAN: Collaborative Knowledge-Aware Attentive Network for Recommender Systems. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 25-30 July 2020, 219-228. <https://doi.org/10.1145/3397271.3401141>
- [18] Wang, H., Zhang, F., Wang, J., Zhao, M., Li, W., Xie, X., et al. (2018) RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems. *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, Torino, 22-26 October 2018, 417-426. <https://doi.org/10.1145/3269206.3271739>
- [19] Wang, H., Zhao, M., Xie, X., Li, W. and Guo, M. (2019) Knowledge Graph Convolutional Networks for Recommend-

- er Systems. *The World Wide Web Conference*, San Francisco, 13-17 May 2019, 3307-3313.
<https://doi.org/10.1145/3308558.3313417>
- [20] Wang, H., Zhang, F., Zhang, M., Leskovec, J., Zhao, M., Li, W., *et al.* (2019) Knowledge-Aware Graph Neural Networks with Label Smoothness Regularization for Recommender Systems. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, 4-8 August 2019, 968-977.
<https://doi.org/10.1145/3292500.3330836>
- [21] Wang, X., He, X., Cao, Y., Liu, M. and Chua, T. (2019) KGAT. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Anchorage, 4-8 August 2019, 950-958.
<https://doi.org/10.1145/3292500.3330989>