



计算机科学

COMPUTER SCIENCE

基于知识图谱与邻域感知注意力机制的推荐算法研究

陈珊珊, 姚苏滨

引用本文

陈珊珊, 姚苏滨. 基于知识图谱与邻域感知注意力机制的推荐算法研究[J]. 计算机科学, 2024, 51(8): 313-323.

CHEN Shanshan, YAO Subin. [Study on Recommendation Algorithms Based on Knowledge Graph and Neighbor Perception Attention Mechanism](#) [J]. Computer Science, 2024, 51(8): 313-323.

相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

[基于动态图表示的设备故障推理预测方法](#)

Device Fault Inference and Prediction Method Based on Dynamic Graph Representation

计算机科学, 2024, 51(7): 310-318. <https://doi.org/10.11896/jsjcx.231000223>

[融合多图卷积与层级池化的文本分类模型](#)

Text Classification Method Based on Multi Graph Convolution and Hierarchical Pooling

计算机科学, 2024, 51(7): 303-309. <https://doi.org/10.11896/jsjcx.230400164>

[融入多影响力与偏好的图对比学习社交推荐算法](#)

Graph Contrastive Learning Incorporating Multi-influence and Preference for Social Recommendation

计算机科学, 2024, 51(7): 146-155. <https://doi.org/10.11896/jsjcx.230400147>

[动态路网下城市交通事故风险预测模型研究与实现](#)

Research and Implementation of Urban Traffic Accident Risk Prediction in Dynamic Road Network

计算机科学, 2024, 51(6A): 230500118-10. <https://doi.org/10.11896/jsjcx.230500118>

[融合多源图特征的Kcore-GCN反欺诈算法研究](#)

Study on Kcore-GCN Anti-fraud Algorithm Fusing Multi-source Graph Features

计算机科学, 2024, 51(6A): 230600040-7. <https://doi.org/10.11896/jsjcx.230600040>

基于知识图谱与邻域感知注意力机制的推荐算法研究

陈珊珊 姚苏滨

南京邮电大学计算机学院/软件学院/网络空间安全学院 南京 210003

摘要 为解决传统推荐算法在面对数据稀疏的推荐任务时产生的冷启动问题,本研究将知识图谱引入推荐算法,结合一种新的邻域感知注意力机制代替传统图注意力机制来挖掘实体间的高阶连通信息,提出了基于知识图谱和邻域感知注意力机制的推荐模型 KGNPAN。得益于知识图谱可使推荐具有精准、多样和可解释的特点,该模型能够很好地缓解数据稀疏与冷启动问题。首先利用基于自对抗负采样的图嵌入方法 RotatE 对原有物品和用户表征的语义信息进行扩充,将实体和关系向量映射成低维嵌入向量;其次,根据协同邻居的不同类型分别应用邻域感知注意力机制聚合邻居节点信息,丰富目标节点语义,并以卷积形式递归挖掘高阶连通信息;最后对用户与项目向量应用内积操作计算交互概率,得到推荐结果。在 Amazon-book 和 Last-FM 两个公共基准数据集上进行实验,结果表明,在与 CKE, BPRMF, RippleNet, KGAT, KGCN 和 CAKN 6 个基准模型的对比中, KGNPAN 相较于基准模型中结果最优的 CAKN 模型,在召回率(Recall)上分别提升了 1.30% 和 1.37%,在归一化折损累计增益上(NDCG)分别提升了 1.26% 和 1.14%,充分验证了其有效性和可解释性。

关键词: 推荐算法;邻域感知注意力机制;知识图谱;图神经网络;冷启动

中图分类号 TP181

Study on Recommendation Algorithms Based on Knowledge Graph and Neighbor Perception Attention Mechanism

CHEN Shanshan and YAO Subin

School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Abstract In order to solve the cold start problem caused by traditional recommendation algorithms when they face the recommendation task with sparse data, this paper introduces the knowledge graph into the recommendation algorithm, combining a new neighbor perception attention mechanism to replace the traditional graph attention mechanism to mine the higher-order connected information between entities, and proposes a recommendation model KGNPAN based on the knowledge graph and neighbor perception attention mechanism. Thanks to the knowledge graph, recommendations can be accurate, diverse and interpretable. This model can effectively alleviate issues of data sparsity and cold start. Firstly, this model utilizes the graph embedding method RotatE based on self adversarial negative sampling to expand the semantic information of the original item and user representations, mapping entity and relationship vectors into low dimensional embedding vectors. Secondly, based on the different types of collaborative neighbors, neighbor perception attention mechanisms are applied to aggregate neighbor node information, enrich the semantics of target nodes, and recursively mine high-order connected information in convolutional form. Finally, the inner product operation is applied to calculate the interaction probability between the user and the project vector, and the recommendation result is obtained. Experiments are conducted on two common benchmark datasets, Amazon-book and Last-FM, and compared with six benchmark models, namely CKE, BPRMF, RippleNet, KGAT, KGCN, and CAKN, KGNPAN. The results show that KGNPAN improves the recall rate by 1.30% and 1.37%, and normalized discounted cumulative gain(NDCG) increases by 1.26% and 1.14%, respectively, compared with CAKN model, which has the best performance in the benchmark modes, verifying the effectiveness and interpretability of the model.

Keywords Recommended algorithm, Neighbor perception attention mechanism, Knowledge graph, Graph neural network, Cold start

1 引言

随着互联网时代的飞速发展,新兴信息技术在方便人类

社会生活的同时也带来了一定的困扰。其中,指数级增长的数据所引发的信息超负荷问题最为突出,致使人们从海量数据中挑选所需信息面临重重困难。推荐算法作为一种能够

有效解决信息泛滥的过滤手段,有助于快速且精准地挑选出偏好内容,给予用户生活极大的便利,目前受到了国内外众多研究者的关注,现阶段已被应用于多个领域,如电影^[1]、商品^[2]、新闻^[3]、音乐推荐^[4]等,产生了丰富的实际成果。

传统的个性化推荐算法主要包括基于内容的推荐算法^[5]协同过滤^[6-8]的推荐算法。基于内容的推荐算法本质上是先对物品表示进行特征抽取,根据用户历史数据学习其偏好,再对比计算候选物品表示和用户偏好间的相似性,最终产生推荐结果;基于协同过滤的推荐算法利用用户群中过去的行为来预测当前用户潜在的偏好物品,能够产生多样且新颖的推荐结果。然而传统的推荐算法在面临数据稀疏和冷启动场景时,因缺少用户历史数据,很难达到精准推荐的效果。

知识图谱作为一种结构化的语义知识表示方式,具有丰富的语义知识,能够有效刻画实体之间的关联性,应用于推荐算法能够协助捕捉到用户与项目之间的潜在联系和相似性,从而增加推荐结果的多样性且提供对推荐项目的合理解释。例如,知识图谱可以帮助推荐算法发现不同电影之间的多种关系,如导演、演员、类型等,并根据这些关系给用户生成多样的、可解释性的推荐结果。例如,图1展示了基于知识图谱的电影推荐示例,其中存在推荐路径:(小王,喜欢,九品芝麻官)⇒(九品芝麻官,类型,喜剧片)⇒(九品芝麻官,演员,周星驰)⇒(小王,喜欢,西游降魔篇),西游降魔篇得以被推荐是因为知识图谱挖掘了人物及电影实体之间的潜在关系,并基于潜在联系做出推荐,使得推荐结果具有更高的可解释性。

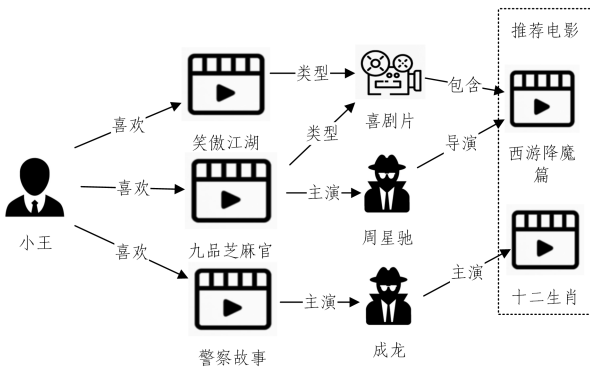


图1 基于知识图谱的电影推荐示例图

Fig. 1 Example of movie recommendation based on knowledge graph

目前,利用知识图谱作为辅助手段来挖掘项目之间的潜在联系正成为目前推荐算法^[9-11]研究的主流之选。例如Wang等提出基于知识图谱与卷积神经网络的新闻推荐模型DKN^[12],引入知识嵌入对实体表示语义信息进行扩充,有效地提高了模型的准确性和泛化能力;Wang等提出端对端模型KGAT^[13],其融合知识图谱与图注意力机制^[14]进行推荐,在一定程度上缓解了数据稀疏以及冷启动带来的不利影响。虽然目前基于知识图谱的推荐算法研究很多,但是在知识嵌入时没有对反转、对称、反对称等关系模式进行完整建模,不能充分挖掘图谱中潜在的连通信息,并且在信息传播聚合时

未对节点类型进行区分,仅通过施加传统的图注意力机制对邻域内的关系信息进行显式建模,忽略了邻域间的隐式关联。

综上所述,为了更好地缓解数据稀疏带来的冷启动问题,给予用户更好的推荐结果,本文提出了一种基于知识图谱与邻域感知注意力机制的推荐算法模型。本文的主要贡献如下:

1)将用户-项目交互矩阵与知识图谱相结合,利用知识图谱作辅助,采用RotatE对原有实体语义进行扩充,生成低维嵌入表示。

2)根据邻居类型的不同分别施加一种新的邻域感知注意力机制,充分考虑中心节点与其相邻节点之间的关系程度以及每个节点的邻居之间的隐含相关性,并根据注意力大小区分信息的重要程度,增强了模型对重要信息的关注程度。

3)提出了一种推荐算法KGNPAN,在缓解数据稀疏和用户冷启动问题上取得明显成效,其采用最大子图采样策略有效地缩减了训练过程,提高了模型性能。在Amazon-book和Last-FM两个大型公开数据集上进行了实验,结果表明KGNPAN在Recall和NDCG指标上均优于基线模型。

2 相关工作

2.1 基于知识图谱的推荐算法

知识图谱通过对实体和关系进行学习分析,能够帮助推荐算法在没有历史数据的情况下更好地推荐物品给新用户,即缓解了用户和物品冷启动问题,并且通过解释实体之间的关系提高了算法的可解释性。目前基于知识图谱的推荐算法包括以下三大类:

1)基于路径的方法。知识图谱中每个节点和边都表示一个不同的实体或者关系类型,本质上可以被为一个异构信息网络,因此此类方法通常的做法是将知识图谱与user-item交互图结合构成一个异构信息网络,并根据图中预先定义好的元路径来计算目标节点和候选节点之间的相似度,不同路径之间的不同权重反映了知识图谱中用户的偏好。例如Zhao等^[15]采用元图(Meta-Graph)替代网络中的元路径(Meta-path)并基于FMG方法来获得异质图中丰富的语义信息,完成学习推荐。但预先构造大量的元路径工作难度较大,需要根据特定的图谱信息指定不同的元构造,当应用场景或者知识图谱产生变化时,需要对元结构进行重新构建,违反了“端到端”的思想理念。因此,一些方法抛弃预先定义元结构的过程,通过深度学习手段对知识图谱中异构图中的语义路径直接进行挖掘,例如文献^[16]和文献^[17]中都利用了循环神经网络结合两种不同的注意力机制来对关系和实体进行表征建模,以此量化语义路径对结果的重要程度,使得推荐具有高可解释性。

2)基于嵌入的方法。该类方法通常由两个模块构成:图嵌入模块和推荐模块。图嵌入模块主要根据图嵌入算法将知识图谱中的实体和关系嵌入低维向量空间中,在保留原有的语义相关性的同时,对原有项目和用户表征进行语义扩充,丰富了用户和项目信息。推荐模块通过推荐算法对嵌入向量

进行表征学习,实现个性化推荐。例如 Zhang 等^[18]首先利用 TransR 完成低维向量映射,结合传统的协同过滤算法学习用户和物品的表示向量,得到交互关系后进行预测。Wang 等着眼于新闻信息的特殊性,采用 TransD^[19]对实体和关系进行扩展,将扩展后的实体和关系向量融入深度学习网络中,获得交互信号,充分考虑了隐藏知识层面的关系,达到了较好的新闻推荐效果。基于嵌入的方法有效降低了图谱的高维性,不存在手动设计元路径的困扰,但低维映射向量中无法体现图中的信息连通性,忽略了知识图谱中的高阶语义信息,缺乏可解释性。

3)基于混合的推荐^[20]。上述两种方法都将知识图谱纳入模型作辅助信息完成推荐,与传统的协同过滤推荐算法等相比,在冷启动问题及可解释性方面都有不错的改进效果,但两者皆只利用了知识图谱中的片面信息,未能充分挖掘利用知识图谱中的信息。因此,部分学者将两者联合考虑,根据嵌入思想将实体和关系进行映射得到低维向量,通过交互计算知识图谱中多跳邻居节点信息来更新实体表示,最后完成对用户偏好的预测推荐。例如 Lin 等^[21]提出 MBGCN 模型,针对用户多行为推荐进行优化,利用用户行为传播、项目相关性感知等多层网络捕获不同用户行为的隐含语义,从而实现针对用户多行为评估的商品推荐。KGCN^[22]将知识图谱与图卷积神经网络结合,在进行低维嵌入的同时捕获它们之间的语义关系,利用多头注意力机制挖掘局部邻域信息,并根据邻居节点权重进行推荐。基于混合的推荐算法将前述两种算法结合起来充分考虑,既对知识图谱中原有的项目和用户表征的语义信息进行扩充,又挖掘了图谱中的语义信息,表现出更好的推荐效果和可解释性。

2.2 知识嵌入方法

知识图谱作为一种语义知识丰富的网络图,应用到推荐系统中可以使推荐具有精准、多样和高可解释的特点。然而知识图谱通常存在链接缺失问题,即某些实体没有被添加到图谱中或者某些存在的实体缺乏足够的描述信息,这种知识图谱的不完整性会影响下游应用,例如知识推理、实体链接。作为一种常用的解决知识图谱链接缺失问题的手段,知识图谱嵌入将实体和关系嵌入连续低维向量空间,在低维向量空间中完成计算分析,在提高计算效率的同时,还保留了结构信息。

知识图谱嵌入方法主要包括基于翻译距离和基于语义匹配的模型。基于翻译距离的模型的核心思想是将关系看作头尾两个实体之间的翻译,将两个实体在关系空间下的距离作为得分函数,距离越小代表越有可能存在关系,应该得到更多的关注。基于翻译距离的模型实现简单,主要包括 TransE^[23],TransH^[24],TransR^[25],RotatE^[26]等。

TransE 作为基于翻译距离的嵌入方法中最经典的向量表示方法,简单且有不错的性能。其受 word2vec 启发,利用词向量的平移不变性,将实体和关系都嵌入向量空间中,把每个三元组实例中的关系看作从头实体到尾实体的翻译,并通过重新构建三元组,以最大限度地提高 $\mathbf{h} + \mathbf{r}$ 和

\mathbf{t} 之间的平衡性,得分函数如式(1)所示:

$$f_r(\mathbf{h}, \mathbf{t}) = -\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_2^2 \quad (1)$$

其中 $\mathbf{h}, \mathbf{r}, \mathbf{t} \in R^k$, \mathbf{h} 为头实体, \mathbf{r} 为关系实体, \mathbf{t} 为尾实体, k 表示向量维度; $\|\cdot\|_2^2$ 表示 L1 范数或者 L2 范数。TransE 的目标是让相互之间存在关系的实体之间的距离尽可能小,应该更加受到重视;不相关的实体之间的距离尽可能大,应该适当被忽略。

TransE 及 TransH 等是将实体和关系映射到同一个语义空间,并在该空间中构建向量模型,这种方式在处理单对单关系时表现出色,但当处理一对多、多对多关系时,其表现就会大打折扣。TransR 认为一个实体包含多种属性,不同关系关注的实体属性也各不相同,它将每个关系矩阵映射到一个不同的嵌入空间中,并在这个空间中学习实体和关系之间的向量表示,在大部分知识图谱学习表示任务中有较好的效果。

为了更好地建模知识图谱中的关系模式,RotatE 将每个关系定义为在复向量空间中从头实体到尾实体的旋转。相比之前的模型,RotatE 使用复数向量来表示实体和关系,这使得其能更好地对关系进行建模,进而挖掘知识图谱中的隐含信息,具有更好的伸缩性,重点解决了前面 Trans 系列不能对对称、反对称、反转、组合等关系模式进行完整建模的问题。其打分函数如式(2)所示:

$$f_r(\mathbf{h}, \mathbf{t}) = -\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|_2^2 \quad (2)$$

基于语义匹配模型通常采用基于相似性的打分函数,通过计算比较嵌入向量空间中实体和关系的潜在语义相似度来判断三元组是否成立。该类模型的典型代表有 RESCAL^[27], DistMult^[28]等。RESCAL 用向量表示实体,用矩阵表示关系,并通过自定义打分函数 $f_r(\mathbf{h}, \mathbf{t}) = \mathbf{h}^T \mathbf{M} \mathbf{t}$ 捕捉三元组的内部交互。DisMult 在 RESCAL 的基础上放松对关系矩阵的约束,利用对角矩阵代替关系矩阵,虽然解决了 RESCAL 中复杂度高的问题,但是仅能对知识图中存在的对称关系进行建模,不能像 RotatE 一样对多种关系模式进行完整建模。

3 本文模型

本文提出了一种融合知识图谱和邻域感知注意力机制的推荐模型 KGNPAN(Knowledge Graph Neighbor Perception Attention Network for Recommendation),整体架构如图 2 所示。该模型大致由 3 部分组成:1)嵌入层。该层首先将知识图谱中的实体和关系节点转换为更低维度的向量,并通过 RotatE 嵌入方法对图谱中存在的反转、对称、反对称等关系进行完整建模,从而对原有的项目、用户和关系实体的语义进行扩充,一定程度上弥补了知识图谱中缺失的链路,有效缓解了数据稀疏和冷启动下交互信息稀少的问题。2)邻域感知传播层。该层基于最大 M 子图采样策略,首先对目标节点的邻居节点进行类型区分,再对属性节点和非属性节点分别施加邻域感知注意力机制,将目标实体与中心实体的邻居节点两两使用余弦相似度计算关注系数,挖掘邻域间的隐式关联,堆叠多层后获得多跳邻居信息,最后通过聚合器将节点本身信息和自我中心网络信息进行聚合,更新节点表示。3)预测层。将模型训练得到的项目及用户的各层

表示串联成单个向量,再利用两者内积的结果来预测用户

和项目之间的匹配分数,最终生成推荐列表。

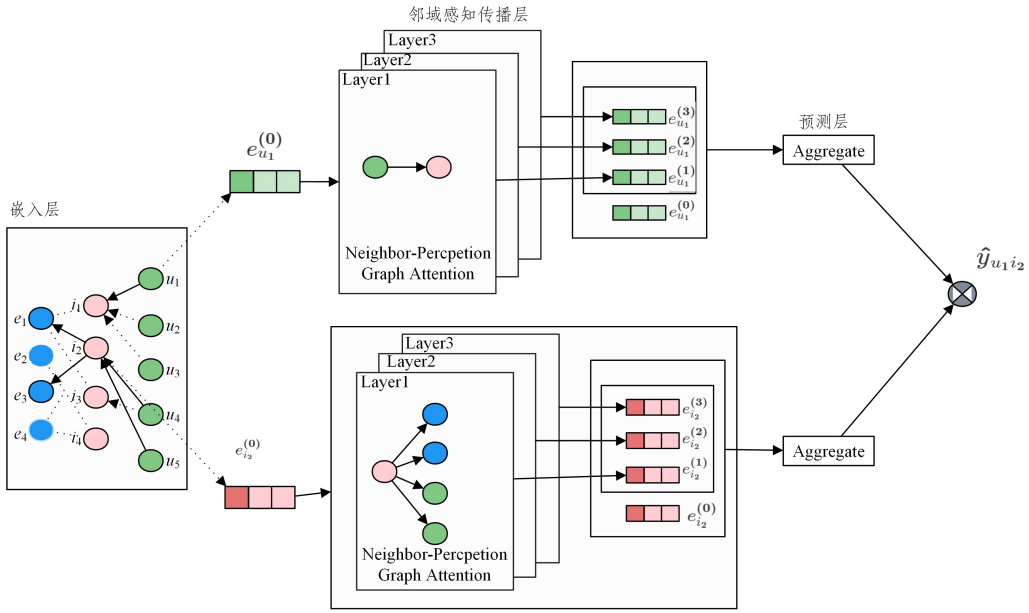


图2 KGNPAN模型整体框架

Fig. 2 Overall framework of KGNPAN model

3.1 嵌入层

知识图谱嵌入方法在将实体和关系参数化为矢量表示的同时,能够保留图谱中的信息。本文采用 RotatE 模型来为实体和关系生成低维嵌入表示,该模型能有效处理一对一、一对多、多对多的关系映射,并且弥补了 TransE 及 TransH 等在对对称、反对称、反转、组合等关系模式进行隐式或显式建模时的不完整性。TransE 可以用来描述复杂的反转、组合关系,而 DisMult 则可以用来描述头部、中间部分以及尾部部分之间复杂的三维相互作用,从而更好地刻画对称关系。现有模型都不能对上述所有模式进行推断建模。不同于 Trans 系列“平移”的做法, RotatE 主要是通过旋转操作来建模实体之间的关系。具体地,它首先将实体和关系映射到复数向量空间,然后定义每个关系 r 引起的函数映射为 h 到 t 的逐元素旋转。通过对实体向量进行旋转, RotatE 可以更好地捕捉实体之间的关系,同时,旋转操作也可以控制实体向量的方向,以便更好地匹配关系的语义。

具体而言,对于知识图谱中的每一个三元组 (h, r, t) ,在任一维度下, RotatE 总是期望满足条件:

$$t_i = h_i \cdot r_i, \text{ where } h_i, r_i, t_i \in \mathbb{R} \quad (3)$$

RotatE 的距离函数如下:

$$d_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\| \quad (4)$$

其中, $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^k$, k 表示向量维度; \circ 表示 Hadamard 乘积。当三元组越接近事实时,头实体与尾实体距离越近,得分函数值就越低,反之如果三元组与事实不符,则距离越远,得分越高。

RotatE 采用一种自对抗负采样方法代替传统的负采样方式。具体来说, RotatE 首先使用当前模型生成一些“伪负例”样本,其次将生成的负例样本和实际的负例样本一起作为负例样本进行对比训练,以此来优化模型对负例样本的区分能力,提高模型的泛化能力,减少模型在训练过程中出现的过拟合问题。其损失函数如下:

$$\mathcal{L}_{KG} = -\log \sigma(\gamma - d_r(\mathbf{h}, \mathbf{t})) - \sum_{i=1}^n p(\mathbf{h}_i', \mathbf{r}, \mathbf{t}_i') \log \sigma(d_r(\mathbf{h}_i', \mathbf{t}_i') - \gamma) \quad (5)$$

其中, σ 是激活函数, $(\mathbf{h}', \mathbf{r}', \mathbf{t}')$ 是通过当前实体和关系嵌入生成的无效三元组。 P 是负样本的权重,定义如下:

$$P(\mathbf{h}_i', \mathbf{r}, \mathbf{t}_i' | \{(\mathbf{h}_i, \mathbf{r}_i, \mathbf{t}_i)\}) = \frac{\exp \alpha f_r(\mathbf{h}_i', \mathbf{t}_i')}{\sum_i \exp \alpha f_r(\mathbf{h}_i', \mathbf{t}_i')} \quad (6)$$

通过不断优化上面的损失函数来训练模型得到一个更好的 embedding 表示。

3.2 邻域感知传播层

KGNPAN 的核心理念是利用知识图谱 KG 中实体之间的相互关联性来学习实体表示,基于邻域感知注意力机制对实体进行聚合更新,进而推动模型的发展。本层分为 4 个阶段:信息传播、邻居采样、聚合实体特、挖掘多跳邻居信息。

3.2.1 信息传播

为了刻画实体的一阶连通性结构,本层采用了 Ego 网络进行信息传播。Ego Network 又称自我中心网络,由一个独特的中心节点和它的相关邻居共同构成。对于一个实体 h ,传统的 Ego 网络采用 \mathcal{N}_h 来表示所有以 h 为头节点的三元组的集合,这里我们采用最大 M 子图采样策略来对训练过程进行优化,具体做法在 3.4 节中说明。衡量 h 的 ego 网络的 embedding 表达式定义为:

$$\mathbf{e}_{\mathcal{N}_h} = \sum_{(h, r, t) \in \mathcal{N}_h} \pi(\mathbf{h}, \mathbf{r}, \mathbf{t}) \mathbf{e}_t \quad (7)$$

其中, \mathbf{e}_t 表示尾实体的嵌入表示; $\pi(\mathbf{h}, \mathbf{r}, \mathbf{t})$ 是这个三元组对应的权重,用来控制尾节点 t 有多少信息可以通过关系 r 传递给头节点 h 。

3.2.2 邻居采样

以往进行邻居信息传播时都是将项目 and 用户节点两者同等看待,通过施加注意力机制计算权重得分,再根据权重大小控制信息传播,最终得到新的实体表示。传统的注意力机制仅计算中心节点与其邻居节点之间的传播系数,且未对邻域

内的关系信息进行显式建模,忽略了邻域间的隐式关联。但是在基于用户-项目交互图的推荐任务中,用户的嵌入在训练过程中聚合了其邻居节点的信息,但很难避免其邻居节点信息的丢失,因此我们定义了一种新的邻域感知注意力机制,直接使用用户的邻居节点之间的配对相似性来获得关注系数。具体来说,就是项目 i 对用户 u 的重要性可以用项目 i 与用户 u 周围邻居项目之间的配对相似度来表示。

此外,为了对不同类型的协作消息的重要性进行区分,采样时先将头实体 h 的邻居集分为属性邻居集 \mathcal{N}_{h-a} 和非属性邻居集 \mathcal{N}_{h-c} :

$$\begin{aligned}\mathcal{N}_{h-a} &= \{t | (h, r, t) \in G, h \in \mathcal{N}_h, r \in \mathcal{R}_a\} \\ \mathcal{N}_{h-c} &= \{t | (h, r, t) \in G, h \in \mathcal{N}_h, r \in \mathcal{R}_c\}\end{aligned}\quad (8)$$

其中, \mathcal{R}_a 和 \mathcal{R}_c 分别是属性邻居和非属性邻居的关系类型集合,非属性邻居由用户节点和项目节点构成。对于用户节点来说,只有项目节点与之相连,而项目节点除了可以与 \mathcal{N}_{h-c} 中的属性节点相连,也可与 \mathcal{N}_{h-a} 中的用户节点相连。

接下来针对两种类型节点分别运用邻域感知注意力机制,经过计算得到关注系数 $\pi_{h-a}(h, r, t)$ 和 $\pi_{h-c}(h, r, t)$, 如式(9)所示:

$$\begin{aligned}\pi_{h-a}(h, r, t) &= G(e_t, \{e_r | t' \in \mathcal{N}_{h-a}\}) \\ &= g(\{f(e_t, e_r) | t' \in \mathcal{N}_{h-a}\}) \\ &= \frac{1}{|\mathcal{N}_{h-a}|} \sum_{r' \in \mathcal{N}_{h-a}} f(e_t, e_{r'}) \\ \pi_{h-c}(h, r, t) &= G(e_t, \{e_r | t' \in \mathcal{N}_{h-c}\}) \\ &= g(\{f(e_t, e_r) | t' \in \mathcal{N}_{h-c}\}) \\ &= \frac{1}{|\mathcal{N}_{h-c}|} \sum_{r' \in \mathcal{N}_{h-c}} f(e_t, e_{r'})\end{aligned}\quad (9)$$

其中, e_h, e_r, e_t 分别表示头实体、关系和尾实体的嵌入表示, $e_{r'}$ 表示当前头实体的邻居尾实体。 $G(\dots)$ 是整体注意力函数, $g(\dots)$ 是注意力池化函数。平均每个成对注意函数计算的结果如式(10)所示:

$$f(e_t, e_r) = \text{ReLU}(\cos(e_t \cdot e_r, e_r)) \quad (10)$$

其中, f 是成对注意函数, e_r 代表 e_t 和 $e_{r'}$ 之间的关系实体。这里我们使用余弦相似度计算两两之间的相似度,并通过 ReLU 激活函数来确保 f 的结果不是负值:

$$\cos(x, y) = \frac{x^T y}{\|x\|_2 \|y\|_2} \quad (11)$$

与以往不同的是,这里并没有使用 Softmax 的点积来作归一化操作,而是采用 ReLU 函数。这是因为 Softmax 计算指数非常耗时,而 ReLU 函数具有线性和非饱和的特点,容易计算,且在处理深度神经网络时有助于缓解梯度消失问题。

在对两种类型邻居采样按系数聚合后,得到如下表示:

$$\begin{aligned}e_{\mathcal{N}_{h-a}} &= \sum_{(h, r, t) \in \mathcal{N}_{h-a}} \pi_{h-a}(h, r, t) e_t \\ e_{\mathcal{N}_{h-c}} &= \sum_{(h, r, t) \in \mathcal{N}_{h-c}} \pi_{h-c}(h, r, t) e_t\end{aligned}\quad (12)$$

3.2.3 聚合实体特征

经过信息传播后得到节点 h 对应 Ego 网络的 embedding,再将其与节点 h 本身的嵌入信息通过聚合器进行聚合,通过权重系数 w_1 和 w_2 进行加权求和,可以准确地评估两种邻居的重要性,从而确定节点在本层的表示形式:

$$e_h^{(l)} = w_1 \text{Agg}(e_h, e_{\mathcal{N}_{h-a}}) + w_2 \text{Agg}(e_h, e_{\mathcal{N}_{h-c}}) \quad (13)$$

目前比较常用的聚合器有如下 3 种:

1) GCN 聚合器将两个特征向量求和之后的结果经过一个非线性函数得到最终的聚合表示,这里使用 LeakyReLU 作为激活函数。ReLU 通过给负值区间赋予非零斜率,减少静默神经元的出现,解决梯度消失问题。GCN 聚合函数如下:

$$\text{Agg}_{(\text{GCN})} = \text{ReLU}(\mathbf{W} \cdot (e_h + e_{\mathcal{N}_h}) + b) \quad (14)$$

其中, \mathbf{W} 是可训练的权重矩阵, b 为偏置参数,可以根据邻居节点对头实体的注意力权重自适应调整参数,提高模型的特征表示能力。

2) GraphSage 聚合器将两个独立的特征组合在一起,并通过非线性变换来获取最终的结果:

$$\text{Agg}_{(\text{GS})} = \text{ReLU}(\mathbf{W} \cdot (e_h \odot e_{\mathcal{N}_h}) + b) \quad (15)$$

3) 双交互聚合器 (Bi-Interaction aggregator) 考虑了节点之间的交互作用,本质上是上面两者的结合。

$$\begin{aligned}\text{Agg}_{(\text{BI})} &= \text{ReLU}(\mathbf{W}_1 \cdot (e_h + e_{\mathcal{N}_h})) + \text{ReLU}(\mathbf{W}_2 \cdot \\ &\quad (e_h \odot e_{\mathcal{N}_h}))\end{aligned}\quad (16)$$

其中, \mathbf{W}_1 和 \mathbf{W}_2 是可训练的权重矩阵。前半部分是 GCN 聚合器的形式,后半部分是将两个特征进行 Hadamard 积后进行 ReLU 变换。Bi-Interaction 聚合器能更好地保留邻居节点之间的交互信息,传递更多和当前实体节点相似的信息,从而更加准确地捕捉节点之间的关系。

我们对上述 3 种聚合器分别进行实验,选择效果最好的双交互聚合器来进行信息聚合,这里以用户节点 u_1 为例,聚合流程如图 3 所示。首先通过最大 M 子图采样策略对 u_1 进行邻居采样,假设选择 i_1, i_2, i_3 这三个项目节点,通过式(12)计算得到衡量当前节点的 Ego 网络 embedding,上半部分直接将两者相加,即 Bi-Interaction 聚合器的前半部分,下半部分计算两者的 Hadamard 积,即 Bi-Interaction 聚合器的后半部分,再将两者经过 ReLU 变换获得最终表示。非属性节点聚合过程与上述过程类似,分别计算之后只需将两者进行聚合操作即可。

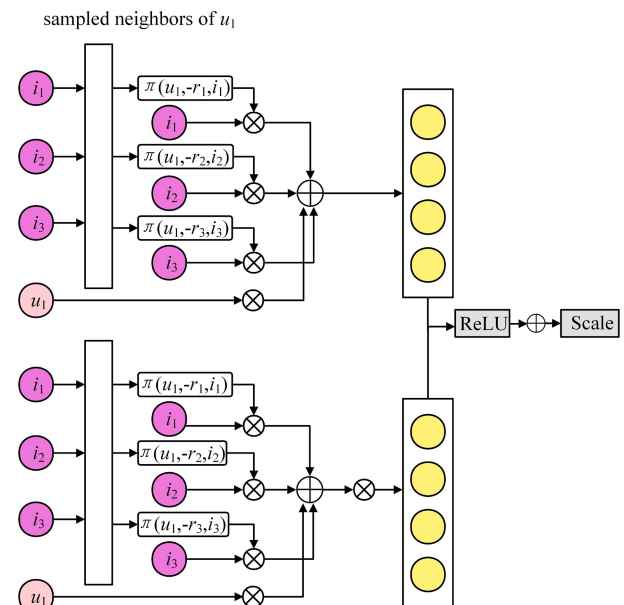


图 3 用户 u_1 的聚合流程图

Fig. 3 Aggregation flow chart of user u_1

3.2.4 挖掘多跳邻居信息

为了挖掘高阶实体信息,进一步堆叠多个邻域感知传播

聚合层以计算实体 L 阶向量,刻画当前节点的 embedding 表示,更好地描述实体的特征和行为,且利用邻域感知注意力机制区分不同邻居信息的重要程度。在此,我们将 l 层的实体表示如下:

$$\mathbf{e}_h^{(l)} = \text{Agg}(\mathbf{e}_h^{(l-1)}, \mathbf{e}_{v_h}^{(l-1)}) \quad (17)$$

当 l 为 0 时,直接设置 \mathbf{e}_h 为初始信息进行传播,之后的传播迭代按照式(17)计算即可。

3.3 预测层

经过邻域感知传播 L 层后,可得到用户的各层表示 $\{\mathbf{e}_u^{(1)}, \mathbf{e}_u^{(2)}, \dots, \mathbf{e}_u^{(L)}\}$ 以及项目的各层表示 $\{\mathbf{e}_i^{(1)}, \mathbf{e}_i^{(2)}, \dots, \mathbf{e}_i^{(L)}\}$ 。为提升传播强度,使用级联操作将原始的 embedding 表示与各个传播层获得的用户和项目表示进行融合,得到最终的向量表示,记作 $\mathbf{e}_u^*, \mathbf{e}_i^*$:

$$\begin{aligned} \mathbf{e}_u^* &= \mathbf{e}_u^{(0)} \parallel \dots \parallel \mathbf{e}_u^{(L)} \\ \mathbf{e}_i^* &= \mathbf{e}_i^{(0)} \parallel \dots \parallel \mathbf{e}_i^{(L)} \end{aligned} \quad (18)$$

其中, \parallel 是串联操作,基于层聚合机制得到的用户和项目的最终表示能够保留原始节点的特征,同时通过调整 L 可以控制信息传播的强度。

预测时将用户和项目二者进行内积得到两者间的匹配分数,再根据匹配分数对候选项目排序,即可生成推荐列表,完成项目推荐。

$$\hat{y}(u, i) = \mathbf{e}_u^{*T} \mathbf{e}_i^* \quad (19)$$

3.4 最大 M 子图采用策略

以往融合知识图谱的推荐算法中,通常采用中心节点的全邻域集进行传播聚合,但真实情况下,知识图谱中项目 and 用户数量巨大,且实体的邻居数目大概率是不相同的,极端情况下数量差异会很大。如果采用这种策略,会加重模型训练的负担。文献[29]提出固定 M 大小采样策略,该策略虽能一定程度上缓解上述问题,但针对邻居数量少于 M 的中心节点会进行邻居重复采样,重复节点的引入带来了额外的计算费用。

为了提高模型的训练效率,避免重复采样,本文采用最大 M 子图采样策略来选择性进行邻居采样。具体来说,对于邻居数大于 M 的节点,我们会从它的邻居中随机选择 M 个进行传播,而对于邻居数小于等于 M 的节点,则会选择它的所有邻居,而不是为了使其采样数达到 M 而对其进行重复采样。

如图 4 所示,假设设定 M 为 3,那么对于用户 u_1 来说,它有 4 个邻居,根据最大 M 子图采样策略,我们只会随机采样它的 3 个邻居,例如 i_1, i_2, i_3 这 3 个邻居节点。而对于 i_2 和 i_3 来说,它的邻居节点少于设定的 M ,于是采用它的全部邻居。

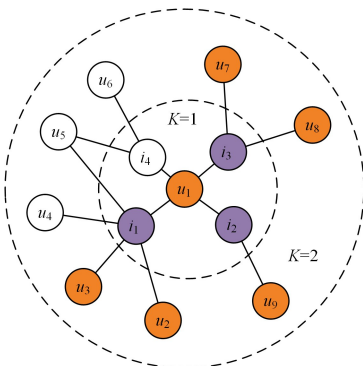


图 4 最大 M 子图采样策略

Fig. 4 Max- M sampling strategy

3.5 损失函数

为了更好地优化模型,推荐模块采用贝叶斯个性化排序 BPR (Bayesian Personalized Ranking) 中的损失函数,如式(20)所示:

$$\mathcal{L}_{CF} = \sum_{(u, i, j) \in O} -\ln \sigma(\hat{y}(u, i) - \hat{y}(u, j)) \quad (20)$$

该损失考虑了用户物品交互中的可观察与不可观察项的相对顺序,它假定更能反映出用户偏好的可观察项的交互项相较于那些不可观察项来说应该被赋予更高的预测值。

KGNPAN 模型的整体损失函数即嵌入模块的损失与推荐模块的损失之和,如式(21)所示:

$$\mathcal{L}_{KGNPAN} = \mathcal{L}_{KG} + \mathcal{L}_{CF} + \lambda \|\Theta\|_2^2 \quad (21)$$

其中, $\|\Theta\|_2$ 是正则化项,利用 Adam 优化算法调整参数,Dropout 随机舍弃一些神经元,从而有效地抑制过拟合的发生。

3.6 算法流程和复杂度分析

KGNPAN 算法流程如算法 1 所示。

算法 1 KGNPAN 算法

输入:用户-物品交互矩阵 \mathbf{Y} ,知识图谱 \mathcal{G}

输出:预测函数 $y_{uv}^* = F(u, v | \theta, \mathbf{Y}, \mathcal{G})$

1. 初始化所有参数
2. for epoch $\leftarrow 0, 1, \dots, \text{total_epoch_number}$ do
3. for each triple $(\mathbf{h}, \mathbf{r}, \mathbf{t})$ in \mathcal{G} do
4. $\mathbf{e}^{(0)} \leftarrow$ Update in Equation 5 by Adam.
5. $\pi_{h-a}(\mathbf{h}, \mathbf{r}, \mathbf{t}) \leftarrow$ Compute in Equation 9.
6. $\pi_{h-c}(\mathbf{h}, \mathbf{r}, \mathbf{t}) \leftarrow$ Compute in Equation 9.
7. for $l=1, \dots, L$ do
8. $\mathbf{e}_{h-a}^{(l)}, \mathbf{e}_{h-c}^{(l)} \leftarrow$ According to Equation 12.
9. $\mathbf{e}_u^{(l)}, \mathbf{e}_i^{(l)} \leftarrow$ According to Equation 17.
10. $\mathbf{e}_u^*, \mathbf{e}_i^* \leftarrow$ Compute according to Equation 18.
11. Calculate predicted probability $\hat{y}(u, i) = \mathbf{e}_u^{*T} \mathbf{e}_i^*$
12. Compute the target loss in Equation 21 and
13. update parameters by Adam.
14. return F .

本文采用交替训练的方式对 KGNPAN 模型进行训练, KGNPAN 的时间复杂度主要由 3 部分组成:1)知识图谱嵌入。此部分采用 RotatE 模型进行嵌入,其时间复杂度为 $O(|G|d^2)$ 。2)邻域感知传播。由式(9)可知,此部分的时间复杂度为 $O(|G|M^2 d_l d_{l-1})$,其中 d_l 和 d_{l-1} 是当前层以及上一层的特征维度, M 是邻居采样个数。3)预测层。此部分采用全层次(共 L 层)生成的嵌入做内积操作,每次更新需要对所有图上的节点进行更新,每层的嵌入大小为 d_l ,因此预测层时间复杂度为 $O(\sum_{l=1}^L |G|d_l)$ 。因此,总时间复杂度为 $O(|G|d^2 + \sum_{l=1}^L M^2 |G|d_l d_{l-1} + |G|d_l)$ 。

4 实验结果及分析

4.1 数据集描述

本文选择 Amazon-book 和 Last-FM 两个公共基准数据集对 KGNPAN 模型的可解释性及有效性进行全面评估,它们不仅包含用户-项目之间的历史交互信息,还可以利用其中

的知识图谱辅助信息与交互信息来构造用户-项目二部图。基本数据如表 1 所列。

表 1 数据集用户项目信息描述

数据集	用户	项目	交互	实体	关系
Amazon-book	70679	24915	847733	88572	39
Last-FM	23566	48123	3034796	58266	9

Last-FM 数据集是一个在音乐推荐中广泛使用的基准数据集,其收集了 Last.fm 在线音乐系统上 2015 年 1 月至 2015 年 6 月期间约 23566 个用户的收听记录、用户信息和音乐属性。

Amazon-book 作为 Amazon-review 数据集的子集,是推荐算法中经典的图书数据集,其中大约收录了 192403 个用户、63001 本书籍以及 1689188 次交互信息。

由于数据集规模较大且稀疏,为了确保数据集的质量,在一般场景下的实验事先采用 10-core 过滤方法来优化数据,即过滤掉数据集中交互次数少于 10 的用户和项目,优化后的数据集包含 70769 个用户、24915 个项目以及 847733 次交互信息。

对于所有数据集,随机选取每个数据集中 80% 的用户-项目交互数据作为训练集,10% 的数据作为测试集,剩余的 10% 交互数据视为验证集。实验中设置正实例为真实交互信息,负实例为用户与其未交互过的物品构造的交互数据。

4.2 测试指标

在对本文模型进行 Top-K 推荐实验时,采用 NDCG@K 和 Recall@K(本文默认设定 $K=20$) 两种指标作为评价算法优劣的标准。

针对原样本而言,召回率的含义是在实际为正的样本中被预测为正样本的概率。在推荐系统中,对于用户 u ,假设 $A(u)$ 是在训练集上根据用户交互行为给出的推荐列表, $T(u)$ 表示用户在测试集上的真实交互行为列表,则该用户的 Recall 计算如下:

$$Recall@K = \frac{\sum_{u \in U} |A(u) \cap T(u)|}{\sum_{u \in U} |T(u)|} \quad (22)$$

召回率的值越大,代表推荐算法命中用户感兴趣项目的概率越大,推荐效果越佳。

NDCG 作为排序结果的评价指标,能够评价推荐的排序列表与用户真实交互列表之间的差距,是折扣累计增益 DCG 除以理想折扣累计增益 iDCG 的值,值越接近于 1,推荐效果越好。计算式如下:

$$NDCG@K = \frac{DCG@K}{IDCG@K} \quad (23)$$

其中, IDCG 为理想情况下的最大 DCG 值,即最优推荐列表。

本文算法进行点击率 CTR 预测实验时,采用了 ACC 和 AUC 来衡量算法的优劣。ACC 描述预测正确的样本数占有所有样本数的比例,即推荐算法所产生的推荐列表中用户偏好的项目所占的比例,计算式如下:

$$ACC = \frac{\sum_{u \in U} |A(u) \cap T(u)|}{\sum_{u \in U} |A(u)|} \quad (24)$$

AUC 是 ROC 曲线与横坐标(即假阳性率)所夹的面积,

用来描述推荐模型的性能好坏。AUC 越大,表明推荐的效果越好。

4.3 基准方法与实验设置

在 KGNPAN 的对比实验中,使用了如下的基准方法:

CKE^[30]:将协同过滤与结构信息、文本数据、图像数据等相结合,在协同过滤的框架中集成了各种辅助信息,通过在 TransR 中使用语义嵌入进而增强矩阵分解,提升推荐系统的质量。

BPRMF^[31]:作为传统的矩阵分解推荐算法,BPRMF 通过不断优化不同用户-项目对之间的成对偏好学习模型,进而优化更新物品之间的相对排名位置来为每个用户生成推荐列表。

RippleNet^[20]:第一次将基于嵌入和基于路径的方法进行融合的混合推荐模型。RippleNet 把 Ripple 类比为用户传播的过程,利用知识图谱中的关联性,系统可以自动不断地拓展用户的潜在兴趣,从而激发他们的偏好,并将其传播出去。

KGAT^[13]:一种将知识图谱和图神经网络相结合的端到端的推荐模型。KGAT 通过注意力机制对知识图谱中实体之间的相似性进行计算,并对图中高阶连通性信息进行建模。

KGCN^[32]:利用图卷积神经网络(GCN)对知识图谱中的实体和关系进行表示学习,利用低维表示进行实体链接和预测任务,通过 Adam 优化器对参数进行优化。

CAKN^[33]:利用异质传播策略对知识关联和协作信号两种信息自然编码并结合,应用知识感知注意机制来刻画不同邻居节点的贡献。

为了证明 KGNPAN 模型的有效性,本文在 Linux ubuntu 系统、Python3.8、PyTorch1.7 环境下对本文模型和基准算法进行对比实验。KGNPAN 模型及其他基准模型的学习轮数都设置为 500,采用 Early-Stopping 早停止策略,即如果 Recall 在 10 个 epoch 内无明显变化则提前结束训练。KGNPAN 具体超参数设置如表 2 所列,基准模型中超参数默认值与 KGNPAN 保持一致,其他 KGNPAN 不包括的超参值与原始论文一致。

表 2 KGNPAN 模型超参数设置

Table 2 Hyperparameter setting of KGNPAN model

超参数名称	参数值
学习轮数	500
用户、项目嵌入维数	128
实体、关系嵌入维数	64
优化器	Adam
学习率	1×10^{-3}
批处理大小	2048
L2 正则项系数	1×10^{-4}
传播层数	3
Dropout	0.1

4.4 实验结果

4.4.1 与基准方法的整体比较

在 Amazon-book 和 Last-FM 两个公共基准数据集上对本文提出的 KGNPAN 模型与其他基准方法进行对比实验,整体结果如表 3 所列。

表3 KGNPAN与基准方法在Recall与NDCG上的比较

Table 3 Recall and NDCG comparison of KGNPAN and benchmark

Model	methods			
	Amazon-book		Last-FM	
	Recall	NDCG	Recall	NDCG
CKE	0.1390	0.0733	0.0865	0.0734
BPRMF	0.1255	0.0676	0.0781	0.0728
RippleNet	0.1415	0.0751	0.0887	0.0749
KGAT	0.1451	0.0793	0.0941	0.0787
KGCN	0.1443	0.0764	0.0893	0.0752
CKAN	0.1458	0.0802	0.0946	0.0785
KGNPAN	0.1477	0.0813	0.0958	0.0794

可以发现, KGNPAN相比效果最好的基线方法性能都有所提升。其中Recall在Amazon-book和Last-FM上分别提高了1.30%和1.37%, NDCG分别提升了1.26%和1.14%。这充分验证了采用RotatE模型能够更加高效地挖掘知识图谱中蕴含的多种关系信息,进一步刻画实体表示;结合领域感知注意力机制对邻居节点进行类型区分,差异化地聚合传播信息,并通过最大 M 子图采样策略优化训练过程,可以达到较好的推荐效果。

基于矩阵分解FM的推荐模型BPRMF性能相比其他模型表现不佳,这证实了利用知识图谱作为辅助手段相比传统的推荐算法能够更好地扩充节点语义信息。CKE是所有基于知识图谱的模型中表现最糟糕的,这与TransE模型不能有效地学习嵌入每个实体和关系信息有关。RippleNet基于水波纹的思想挖掘多跳关联实体信息, KGCN和KGAT采用注意力机制控制信息传播聚合,三者皆优于CKE和BPRMF模型,这充分验证了在知识图中传播信息以保持全局相似性的有效性。CKAN不仅考虑到了KG中的知识关联,还加入了协作信号进行结合编码,获得了更加丰富的节点信息。

4.4.2 不同聚合器的效果

为了探究聚合器对模型的影响,我们分别使用GCN, GraphSage, Bi-Interaction这3种聚合器进行实验,结果如表4所列。实验结果表明, Bi-Interaction聚合器相较于其他两个聚合器效果更好,这主要得益于Bi-Interaction聚合器在聚合过程中更加重视自身的特征信息以及传递邻居节点中与自身信息高度类似的特征信息,进而挖掘出用户和物品之间的交互特征,达到了更好的效果; GCN与GraphSage都是采取将结构信息与节点自身信息进行等比重聚合的做法,聚合效果相较于Bi-Interaction稍差,这也印证了节点本身信息在建模过程中的重要性。

表4 聚合器对比实验结果

Table 4 Results of aggregators comparison experiment

Aggregator	Results of aggregators comparison experiment			
	Amazon-book		Last-FM	
	Recall	NDCG	Recall	NDCG
GCN	0.1428	0.0788	0.0914	0.0790
GraphSage	0.1440	0.0786	0.0911	0.0786
Bi-Interaction	0.1477	0.0813	0.0958	0.0794

4.4.3 模型深度的影响

KGNPAN通过堆叠多个领域感知传播层挖掘节点的高阶邻居信息,而邻居跳数直接影响了推荐性能。 L 过小会导致部分有效实体信息在聚合过程中被忽略, L 过大意味着会

存在许多与中心节点毫无联系的实体信息被聚合,许多噪声被引进,从而削弱了模型的表现。我们分别将模型深度设置为1~4进行实验,实验结果如表5所列。

表5 嵌入传播层数 L 的影响Table 5 Effect of the number of embedding propagation layer L

Layer	Effect of the number of embedding propagation layer L			
	Amazon-book		Last-FM	
	Recall	NDCG	Recall	NDCG
KGNPAN-1	0.1371	0.0745	0.0908	0.0737
KGNPAN-2	0.1453	0.0802	0.0942	0.0781
KGNPAN-3	0.1477	0.0813	0.0958	0.0794
KGNPAN-4	0.1484	0.0818	0.0963	0.0801

从表中可以看出, KGNPAN对 L 的改变比较敏感,在 L 为3时模型达到较好的效果,而在 L 增加到4时,只是取得了微小的改进,这表明考虑实体之间的三阶关系就足以捕获协作信号,这和KGAT中的研究结果一致。

4.4.4 超参数影响分析

为了探究学习率和最大 M 采样策略中 M 的大小对模型性能的影响,本文将KGNPAN模型的学习率设置为 $\{1 \times 10^{-1}, 1 \times 10^{-2}, 1 \times 10^{-3}, 1 \times 10^{-4}, 1 \times 10^{-5}\}$, M 设置为 $\{5, 10, 15, 20, 25\}$,同时将TOP-K推荐任务中 K 设置为 $\{20, 40, 60, 80, 100\}$ 进行对比实验,实验结果如表6所列。

表6 超参数对比实验结果

Table 6 Results of hyper-parameters comparison experiment

超参数	Recall	NDCG	
M	5	0.1396	0.0749
	10	0.1429	0.0774
	15	0.1462	0.0800
	20	0.1477	0.0813
	25	0.1458	0.0796
学习率	1×10^{-1}	0.0657	0.0384
	1×10^{-2}	0.0923	0.0523
	1×10^{-3}	0.1477	0.0813
	1×10^{-4}	0.0954	0.0536
	1×10^{-5}	0.0641	0.0349
K	20	0.1477	0.0813
	40	0.2094	0.0978
	60	0.2537	0.1079
	80	0.2903	0.1166
	100	0.3254	0.1229

实验结果表明,增加 M 的大小可以使更多的邻居节点信息在传播聚合过程中得到传递,更好地刻画中心节点的实体表示。然而当 M 设置过大时,会引入过多的噪声信息,导致模型学习到错误的特征规律,引入偏差,出现过拟合现象,模型因此无法达到预期的性能水平。

设置过小的学习率会严重限制模型在训练过程中的收敛速度,需要更多的迭代次数来寻求最优解,增加了计算和时间成本。而学习率设置过大可能会导致模型在训练初期就过于发散,无法收敛到最优解,或者在训练过程中震荡不定,影响模型精度。由表6可知,随着学习率的增大,模型的性能得到显著提升,但增至一定程度后,由于过早达到拟合状态,模型精度会受到影响,导致性能下降。

Top-K推荐通过从候选项中选择评分最高的 K 个项目来向用户提供个性化推荐,评价指标NGCG@ K 和Recall@ K 的结果如图5、图6所示。

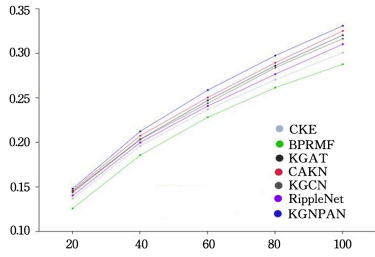


图5 KGNPAN与基准模型 Recall 的比较

Fig. 5 Comparison of Recall between KGNPAN and benchmark models

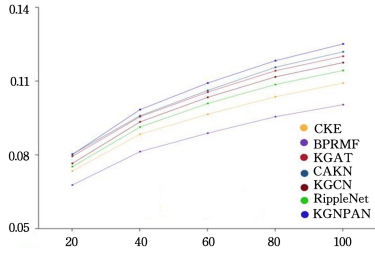


图6 KGNPAN与基准模型 NDCG 的比较

Fig. 6 Comparison of NDCG between KGNPAN and benchmark models

由图5、图6可知, Recall和NDCG都与Top-K中的K大小成正比, 较大的K值可以更全面地覆盖候选项空间, 使用户能接触到更多最新的且更有可能是热门的项目; 较小的K值更关注当前最热门或最相关的项, 使推荐结果更具个性化。

4.4.5 模块影响分析

为了探究知识图谱嵌入和邻域感知注意力机制对模型的影响, 以及区分邻居类型对信息聚合的作用, 本文对以下3种变体进行消融实验, 以此来验证模型的有效性。

1) KGNPAN-I: 不采用RotatE模型对原有知识图谱进行语义扩充, 即禁用嵌入层。

2) KGNPAN-II: 摒弃领域感知注意力机制按权重聚合信息的做法, 采用目标节点的所有邻居信息进行聚合。

3) KGNPAN-III: 同等对待节点所有类型邻居的信息, 直接对全部邻居信息进行聚合。

表7 消融实验结果

Table 7 Results of ablation experiment

Module	Amazon-book		Last-FM	
	Recall	NDCG	Recall	NDCG
KGNPAN-I	0.1355	0.0734	0.0902	0.0753
KGNPAN-II	0.1351	0.0728	0.0894	0.0745
KGNPAN-III	0.1372	0.0744	0.0912	0.0763
KGNPAN	0.1477	0.0813	0.0958	0.0794

4.4.6 冷启动和数据稀疏场景实验分析

为了验证KGNPAN在冷启动和数据稀疏场景下的有效性, 本文对Amazon-book数据集进行过滤处理, 即训练集中只保留项目交互个数少于5的用户数据, 得到冷启动数据集Amazon-book-Cold。在Cold数据集上进行实验, 结果如表8所列。

表8 Amazon-book-Cold数据集实验结果

Table 8 Experiment results on Amazon-book-Cold dataset

Model	Amazon-book		Last-FM	
	Recall	NDCG	Recall	NDCG
CKE	0.1044	0.0604	0.0715	0.0622
BPRMF	0.1013	0.0521	0.0629	0.0577
RippleNet	0.1167	0.0634	0.0755	0.0634
KGAT	0.1198	0.0699	0.0806	0.0689
KGCN	0.1180	0.0691	0.0796	0.0685
CKAN	0.1211	0.0703	0.0811	0.0694
KGNPAN	0.1246	0.0727	0.0843	0.0712

实验结果表明, KGNPAN在冷启动场景下的推荐效果在Recall和NDCG指标上都优于其他基准算法, 这充分表明新用户交互数据很少的情况下, KGNPAN模型可以通过RotatE先对原有的交互数据进行语义扩充, 再根据邻域感知注意力机制对目标节点进行邻居信息聚合, 得到语义丰富的新实体表示, 进而帮助冷启动用户构建项目偏好, 达到较好的推荐效果。

数据稀疏作为目前推荐算法遇到的另一大难题, 指用户只与很少一部分项目有反馈, 与绝大部分项目都无交互信息, 这就导致在用户-项目交互矩阵中存在大量空值, 出现严重稀疏的现象, 通常稀疏度可达99%以上; 而推荐算法需要依赖用户对项目的评分或者交互数据来挖掘其潜在偏好, 交互数据越多, 推荐结果越合理。数据的稀疏性会严重影响推荐算法的性能。

为了验证本文模型在数据稀疏场景下的有效性, 本文根据每个用户项目交互次数的不同对Last-FM数据集进行抽取划分, 在4个不同稀疏度的训练集上对KGNPAN模型进行实验, 实验结果如图7、图8所示。

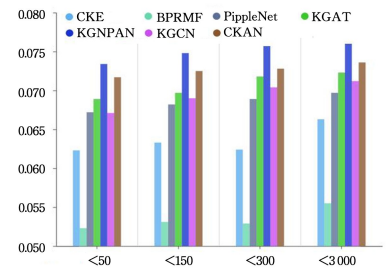


图7 KGNPAN与基准模型在稀疏场景下NDCG的比较

Fig. 7 Comparison of NDCG between KGNPAN and benchmark models in sparse scenes

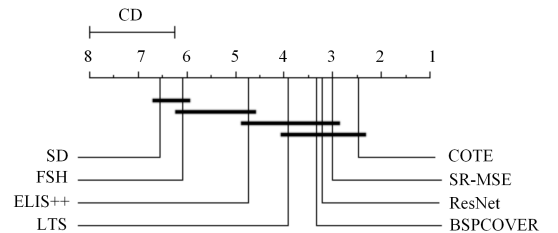


图8 KGNPAN与基准模型在稀疏场景下Recall的比较

Fig. 8 Comparison of Recall between KGNPAN and benchmark models in sparse scenes

由实验结果可知, KGNPAN在稀疏度较高的数据集集中的表现明显优于其他基线模型, 即使是在最为稠密的数据集

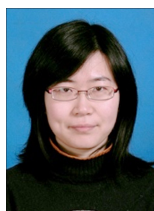
中, KGNPAN 的表现也稍优一些。实验结果验证了 KGNPAN 通过 RotatE 对多种关系进行完整建模来扩充语义信息, 并通过领域感知注意力机制来挖掘高阶连通性的有效性。

结束语 缓解推荐算法中的数据稀疏和冷启动问题一直是推荐领域着力研究的棘手难题, Wang 等提出的 KGAT 虽然将知识图谱和图注意力机制进行了初步结合, 但是在邻居信息聚合以及训练过程优化方面仍存在一定不足。本文针对其存在的缺陷, 对 KGAT 进行了多方面优化, 提出了一种融合知识图谱与邻域感知注意力机制的推荐模型 KGNPAN。一是使用 RotatE 和自对抗采样方法对嵌入模型进行优化; 二是采用了一个新的邻域感知注意力机制对给定节点的不同邻居之间的关注进行成对平均, 为这些邻居分配不同的重要性, 从而更好地聚合节点信息; 三是采用最大 M 子图采样策略, 避免每次对当前节点的整个邻居集进行传播和重复采样, 提高了模型的有效性和可解释性。KGNPAN 针对静态知识图谱进行研究, 即根据某一时间片段上的数据进行推荐, 但部分应用需要实时获取最新的知识和动态信息, 并且支持实时更新和演化, 因此根据动态图谱为用户提供最新的知识和洞见是进一步研究的方向。

参 考 文 献

- [1] CHENG Y H, LIU N, LU Y, et al. Recurrent knowledge attention network for movie recommendation[C]//2020 3rd International Conference on Electron Device and Mechanical Engineering(ICEDME). IEEE, 2020:648-651.
- [2] MA W, ZHANG M, CAO Y, et al. Jointly learning explainable rules for recommendation with knowledge graph [C] // The World Wide Web Conference. 2019:1210-1221.
- [3] LIU D Y, LIAN J X, WANG S Y, et al. KRED: Knowledge-aware document representation for news recommendations[C]// Proceedings of the 14th ACM Conference on Recommender Systems. 2020:200-209.
- [4] HANSEN C, HANSEN C, MAYSTRE L, et al. Contextual and sequential user embeddings for large-scale music recommendation[C]//Proceedings of the 14th ACM Conference on Recommender Systems. 2020:53-62.
- [5] SHU J B, SHEN X X, LIU H, et al. A content-based recommendation algorithm for learning resources [J]. Multimedia Systems, 2018, 24(2): 163-173.
- [6] NILASHI M, OTHMAN BIN I, NORAFIDA I. Multi-criteria collaborative filtering with high accuracy using higherorder singular value decomposition and Neuro-Fuzzysystem[J]. Knowledge-Based Systems, 2014, 60:82-101.
- [7] ZHANG J D, CHOW C Y, XU J. Enabling kernel-based attribute-aware matrix factorization for rating prediction[J]. IEEE Transactions on Knowledge and Data Engineering, 2016, 29(4): 798-812.
- [8] LU Q B, GUO F P. Personalized information recommendation model based on context contribution and item correlation[J]. Measurement, 2019, 142:30-39.
- [9] YE J M, SONG J Q, ZHANG K, et al. Research on Exercise Recommendation Algorithm for Online Judge System Enhanced by Knowledge Graph[J]. Journal of Chinese Computer Systems, 2023, 44(11):2558-2565.
- [10] ZHAGN X X, TANG Y Q, ZHAO W, et al. Personalized Learning Resource Recommendation Based on Knowledge Graph and Graph Embedding[J]. Computer Systems & Applications, 2023, 32(5):180-187.
- [11] LUO C T, YE X. Survey on Knowledge Graph-Based Recommendation Methods [J]. Computer Engineering and Applications, 2023, 59(1):49-60.
- [12] WANG H, ZHANG F, XIE X, et al. DKN: Deep knowledge-aware network for news recommendation[C]// Proceedings of the 2018 World Wide Web Conference. 2018:1835-1844.
- [13] WANG X, HE X, CAO Y, et al. Kgat: Knowledge graph attention network for recommendation[C]// Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019:950-958.
- [14] VELICKOVIC P, CUCURULL G, CASANOVA A, et al. Graph attention networks[J]. arXiv:1710.10903, 2017.
- [15] ZHAO H, YAO Q, LI J, et al. Meta-Graph Based Recommendation Fusion over Heterogeneous Information Networks [J]. SIGKDD Explorations, 2017(CD/ROM):584-593.
- [16] XIAN Y, FU Z, MUTHUKRISHNAN S, et al. Reinforcement knowledge graph reasoning for explainable recommendation [C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019:285-294.
- [17] HAMILTON W, YING Z, LESKOVEC J. Inductive representation learning on large graphs[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017:1025-1035.
- [18] ZHANG F, YUAN N J, LIAN D, et al. Collaborative knowledge base embedding for recommender systems[C]// Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016:353-362.
- [19] JI G, HE S, XU L, et al. Knowledge graph embedding via dynamic mapping matrix [C] // Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (volume 1: Long papers). 2015:687-696.
- [20] WANG H, ZHANG F, WANG J, et al. RippletNet: Propagating user preferences on the knowledge graph for recommender systems[C]// Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018:417-426.
- [21] LIN X, QUAN Z, WANG Z J, et al. KGNN: Knowledge Graph Neural Network for Drug-Drug Interaction Prediction[C]// IJ-CAI. 2020:2739-2745.
- [22] WANG H, ZHAO M, XIE X, et al. Knowledge graph convolutional networks for recommender systems [C] // The World Wide Web Conference. 2019:3307-3313.
- [23] BORDES A, USUNIER N, GARCIA-DURAN A, et al. Translating embeddings for modeling multi-relational data [C] // Proceedings of the 26th International Conference on Neural Information Processing Systems— Volume 2. 2787-2795.

- [24] WANG Z,ZHANG J,FENG J,et al. Knowledgegraph embedding by translating on hyperplanes[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2014,28(1).
- [25] LIN Y,LIU Z,SUN M,et al. Learning entity and relation embeddings for knowledge graph completion[C]// Twenty-ninth AAAI Conference on Artificial Intelligence. 2015.
- [26] SUN Z,DENG Z H,NIE J Y,et al. Rotate: Knowledge graph embedding by relational rotation in complex space[J]. arXiv: 1902.10197,2019.
- [27] NICKEL M,TRESP V,KRIEGEL H P. A three-way model for collective learning on multi-relational data[C]//ICML. 2011.
- [28] YANG B,YIH S W,HE X,et al. Embedding Entities and Relations for Learning and Inference in Knowledge Bases[C]// Proceedings of the International Conference on Learning Representations(ICLR). 2015.
- [29] HAMILTON W,YING Z T,LESKOVEC J. Inductive representation learning on large graphs[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. 2017:1025-1035.
- [30] ZHANG F Z,YUAN N J,LIAN D F,et al. Collaborative knowledge base embedding for recommender systems[C]// Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.
- [31] RENDLE S,FREUDENTHALER C,GANTNER Z,et al. BPR: Bayesian personalized ranking from implicit feedback[C]// Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence. 2012:452-461.
- [32] WANG H,ZHAO M,XIE X,et al. Knowledge graph convolutional networks for recommender systems[C]// The World Wide Web Conference. 2019:3307-3313.
- [33] WANG Z,LIN G,TAN H,et al. CKAN:collaborative knowledge-aware attentive network for recommender systems[C]// Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2020: 219-228.



CHEN Shanshan, born in 1980, Ph. D., associate professor. Her main research interests include large-scale distributed storage systems and architectures and so on.

(责任编辑:何杨)