

一种融合知识图注意神经网络的推荐算法

李瑞征 赵加坤

(西安交通大学软件学院 陕西 西安 710048)

摘要 为了提高推荐算法的准确性和可解释性,通常会在推荐算法中添加并利用用户和项目的一些辅助信息。大量实验表明,在推荐算法中添加知识图谱作为辅助信息,通过挖掘实体之间的相关属性可以有效地获取项目之间的相关性,从而较大地提高推荐的性能。受到图注意神经网络和KGCN的启发,设计一个注意嵌入传播层来计算知识图谱中实体的邻居信息,以丰富项目表示。在三个真实的数据集上进行实验,结果分析表明,在电影和书籍推荐中该算法推荐性能最佳,在音乐推荐中也取得了较好的推荐效果。

关键词 推荐系统 知识图谱 图注意神经网络 CTR

中图分类号 TP301.6

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2024.03.042

KNOWLEDGE GRAPH ATTENTION NETWORK FOR RECOMMENDATION SYSTEMS

Li Ruizheng Zhao Jiakun

(School of Software Engineering, Xi'an Jiaotong University, Xi'an 710048, Shaanxi, China)

Abstract In order to improve the accuracy and interpretability of the recommendation algorithm, some auxiliary information of users and items is usually added and used in the recommendation algorithm. A large number of experiments show that adding knowledge graphs as auxiliary information to the recommendation algorithm can effectively obtain the correlation between items by mining the relevant attributes between entities, thereby greatly improving the performance of the recommendation. Inspired by the graph attention neural network and KGCN, an attention embedding propagation layer was designed to calculate the neighbor information of entities in the knowledge graph to enrich project representation. Experiments were conducted on three real data sets. The analysis of the results shows that this algorithm has the best recommendation performance in movie and book recommendation; in music recommendation, it has also achieved a high recommendation effect.

Keywords Recommendation system Knowledge graphs Graph attention neural network CTR

0 引言

随着互联网技术的飞速发展,各类信息迅速增长造成的信息过载、用户搜索迷向已经成为了互联网不得不面对的一个问题。为了解决信息过载带来的一系列负面影响,推荐系统便应运而生了。推荐系统的目的是通过最短时间内为用户检索到一组满足其个性化需求的项目,在快速解决信息过载的过程中发挥了极其重要的作用。今天,不管是从搜索引擎、电子商务,还是社交媒体和新闻网站,都可以发现推荐系统的身影。推荐系统不仅可以从大量繁杂的项目中筛选出最

符合用户需求的极少量项目来供用户选择,而且还可以满足用户的个性化兴趣推荐的要求。

目前,最常使用的推荐算法是基于协同过滤的方法(CF)^[1-3],但是该算法在使用过程中通常会遇到用户-项目交互稀疏和冷启动的问题,导致推荐效果不佳。为了克服这个难题,研究人员通常会在算法中添加一些辅助信息来提高算法的推荐效果,例如用户的基本属性或者是项目的基本属性。最近的一些相关研究表明,异构多关系信息网络-知识图谱有助于提高推荐系统的性能,在知识图谱(KG)中,项目的属性不是孤立的,而是相互联系的。将知识图谱应用到推荐系统中,可以提供如下三点帮助^[4]:(1) 提高推荐系统的

准确性:使用知识图谱中项目之间丰富的语义关系有助于探索发现项目之间潜在的联系,从而提高推荐的准确性;(2) 提高推荐的多样性:知识图谱中的各种关系有助于合理地扩展用户的兴趣,从而提高推荐的多样性;(3) 提供了推荐的可解释性:知识图谱中链接了用户的历史偏好和历史交互项目,为推荐提供了可解释性。

在推荐算法中融合知识图谱的方法主要有两种:基于路径的方法和基于嵌入的方法。基于路径的方法^[5]主要是通过设计元路径/元图计算潜在特征,以此来表示用户和项目之间沿不同关系路径/图的连通性,提取高阶信息,然后将其输入到推荐框架中进行推荐。但是这种方法严重依赖手动设计元路径或元图,而在某些应用场景,例如某些实体和关系不在同一个域中,手动设计元路径是非常困难的,甚至是不可能实现的。基于嵌入的方法主要是通过利用 transE^[6]、transH^[7]和 transR^[8]等方法将实体和关系映射为低维表示向量,然后将表示向量使用到推荐框架中。这类 KGE 方法在使用时仅需要考虑实体的直接关系,并不需要考虑多跳的关系路径,因此更适合进行链接预测和 KG 完成。

现今,图神经网络^[9-10]在知识图谱中的使用更加频繁,例如图卷积网络和图注意网络。受到图卷积网络的启发,Wang 等^[4]提出了使用知识图卷积网络来推荐项目,其关键思想是:在计算 KG 中给定实体的表示时,将带有偏差的邻域信息进行汇总和合并,可以成功地捕获局部临近结构并将其存储在每个实体中。同时,邻域的权重取决于连接关系和特定的用户,既表征了 KG 的语义信息又体现了用户对关系的个性化兴趣。

本文受到 KGCN^[4]和 GAT^[10]的启发,提出知识图注意网络的推荐算法(KGAT2Rec)。图注意网络采用按照邻居实体节点对给定实体节点的不同重要程度分配权重的方法,区别于图卷积神经网络中将所有邻居实体的信息赋给相同权重。图注意网络用于推荐算法的设计思路如下:在计算 KG 中给定实体的表示时,设计一个注意嵌入传播层计算实体的邻居权重;然后将邻居信息进行汇总合并,并用来参与之后的推荐计算。这样做的好处包括:(1) 可以通过关系与用户区分不同邻居实体的重要程度,为实体聚合更重要的信息;(2) 同 KGCN,通过邻居聚合操作,可以捕获实体的局部临近结构。如图 1 所示,对于电影 X-Men,通过注意嵌入传播层获得该电影邻居(演员、导演和流派等)对其的不同影响,对于不同用户而言,有些用户可能更多关注该电影中的演员 Huge Jackman,则会更多地为其推荐该演员的相关电影 X-Men:Days of future past,有

些用户可能更多关注该电影的导演 Bryan Singer,则会更多地为其推荐该导演的相关电影 Jack the giant slayer,从而实现为不同的用户推荐不同的电影。在计算过程中,需要注意:每个实体邻居的大小各不相同,因此同样采用 KGCN 中的方法,对每个实体的邻居进行固定采样。

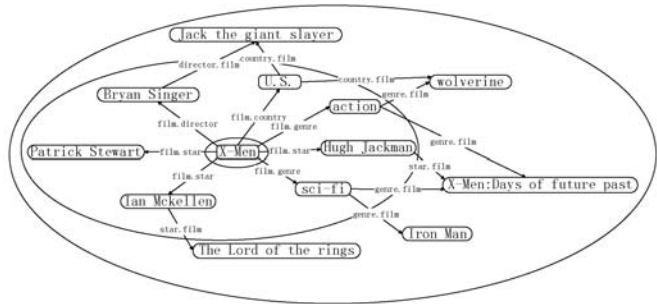


图 1 电影 X-Men 的知识图谱

将本文算法应用于三个数据集:MovieLens-1M、Book-Crossing 和 Last. FM。实验结果表明,与大多数基线算法相比,本文算法具有较高的推荐性能。

1 相关内容

1.1 知识图谱

在知识图谱中包含很多有用的事实和与项目相关的信息,KG 是一种有向异构图,节点表示实体,边表示关系,通常使用三元组(head, relation, tail)的形式进行保存。现在,研究人员提出了一些用于学术和工业应用的知识图谱:NELL、DBpedia、Google Knowledge Graph 和 Microsoft Satoria。最近提出一种方法试图将知识图谱嵌入到连续的向量空间中,同时保留原始图的某些特征。例如基于翻译的方法(TransE^[6]、TransH^[7]和 TransR^[8]),其中 TransE^[6]的主要思想如图 2 所示,在空间中存在 head + relation \approx tail 的假设,三元组中的一对嵌入实体可以以较低的误差通过平移向量 r 进行连接。最后通过正样本的得分高于负样本作为损失函数进行训练得到实体和关系的嵌入。其他算法 TransH 和 TransR 是对 TransE 算法的改进。

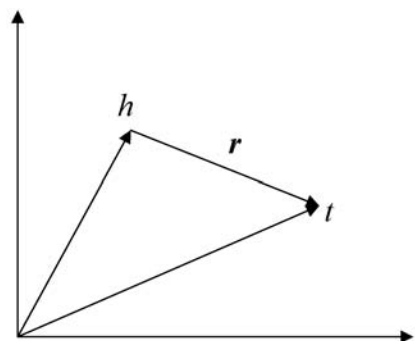


图 2 TransE

1.2 图神经网络

最近有很多研究将卷积推广到图域,主要分为两个方向:频谱方法和非频谱方法。频谱方法可用于图形的频谱表示,Bruna 等^[11]通过计算图 Laplacian 的本征分解,在傅里叶域中定义了卷积运算,但是也导致了密集的运算和非空间局部化的滤波器。而之后 Kipf 等^[12]通过限制滤波器在每个节点周围的 1 步邻域中进行计算简化了之前的方法。但是在基于频谱的方法中,学习滤波器都依赖于 Laplacian 特征根,因此在特定结构上训练的模型不能直接应用于具有不同结构的图,而非频谱方法却可以直接在图上定义卷积,并在空间上相邻的组中进行计算。为了处理具有不同大小的邻域并保持 CNN 的权重共享属性,研究人员建议为每个节点度学习权重矩阵,从图中提取局部连接的区域,或提取并归一化包含固定数量节点的邻域。Hamilton 等^[13]提出了 GraphSAGE,是一种利用归纳方法进行节点计算表示的方法。通过对每个节点固定大小的邻域进行采样,然后再执行特定的聚合,这种方法在大型归纳测试中都表现出了很高的性能。

Velickovic 等^[10]受到图卷积网络和注意机制的启发,提出了基于注意力体系结构的神经网络。该网络遵循自注意策略,通过邻居来计算图中每个节点的隐藏表示,该模型可直接用于归纳学习问题,包括将模型推广到完全看不见的图形的任务中。

现在有很多推荐算法使用了图神经网络,如 deepInf^[14]通过设计将网络结构和特殊用户特征整合到卷积神经网络和注意力网络中,以此学习用户的潜在特征表示来预测社交影响。KGAT^[15]将用户项目图和知识图谱中融合构成一个协作知识图,然后在图神经网络框架下通过端到端的方法实现高阶关系建模来进行推荐。

2 模型框架

2.1 问题描述

用户项目交互矩阵 \mathbf{Y} : 用户集表示为 $U = \{u_1, u_2, \dots, u_n\}$, 项目集表示为 $V = \{v_1, v_2, \dots, v_n\}$, 根据用户隐式反馈定义用户和项目交互矩阵 $\mathbf{Y} = \{y_{uv} \mid u \in U, v \in V\}$:

$$y_{uv} = \begin{cases} 1 & u \text{ 和 } v \text{ 之间存在隐式交互} \\ 0 & \text{其他} \end{cases} \quad (1)$$

式中: $y_{uv} = 1$ 表示用户和项目之间存在隐式交互(例如购买和点击等行为), 否则, $y_{uv} = 0$ 。

知识图谱 \mathcal{G} : 由大量形如 entity-relation-entity 的三元组 (h, r, t) 的形式组成, 其中 $h \in \mathcal{E}, r \in \mathcal{R}, t \in \mathcal{E}$, \mathcal{E} 是知识图谱中的实体集, \mathcal{R} 是知识图谱中的关系集。

问题描述: 给定用户项目交互矩阵 \mathbf{Y} 和知识图谱 \mathcal{G} , 模型的目标是预测用户 u 对之前没有交互过的项目 v 是否有潜在兴趣。目的是学习一个预测函数 $\hat{y}_{uv} = \mathcal{F}(u, v \mid \mathbf{Y}, \mathcal{G}, \Theta)$, \hat{y}_{uv} 表示用户 u 点击项目 v 的概率, Θ 表示函数 \mathcal{F} 中的模型参数。

模型的整体框架如图 3 所示, 从知识图谱中采样 n 跳邻居项目, 然后与用户表示在注意嵌入传播层中进行计算, 获取项目表示后进行得分预测。

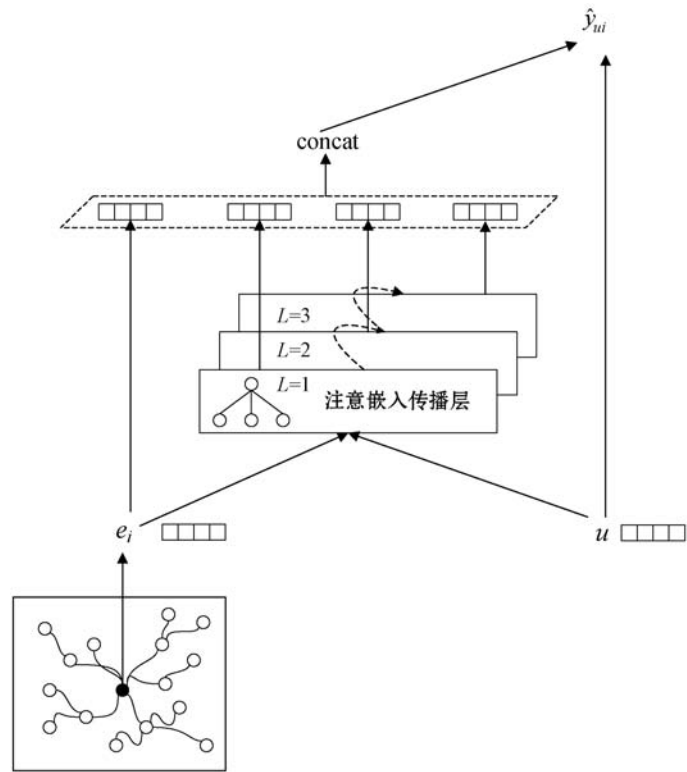


图3 模型框架

2.2 注意嵌入传播层

在图卷积网络的体系结构中,使用图神经网络的思想来捕获知识图谱中实体之间的高阶结构邻近性,下面主要介绍单层注意嵌入传播层方法。

实体邻域集: 对于每个实体 h , 它的邻域集表示为:

$$\mathcal{N}_{N(h)} = \{(h, r, t) \mid (h, r, t) \in \mathcal{G}\} \quad (2)$$

式中: h 表示头实体; t 表示与 h 一跳相连的尾实体。在本文中,为每个实体采样固定大小的邻域,即 $|\mathcal{N}_{(v)}| = K$ 。

对于每一个用户项目对 (u, v) , $\mathcal{N}_{(v)}$ 表示项目对应的实体邻域集,使用关系注意力机制来计算项目 v 的对应实体的邻居实体的权重。通过式(3)计算项目

对应实体的邻居实体的得分。

$$\pi(h, r, t) = \mathbf{u}^\top \tanh(\mathbf{e}_h + \mathbf{e}_r) \quad (3)$$

式中: \mathbf{e}_h 和 \mathbf{e}_r 分别是项目 v 对应的邻居实体向量和关系向量; \mathbf{u} 是用户 u 的向量表示; \tanh 为非线性激活函数,注意力得分取决于在空间中目标实体和邻居实体的距离,即 $\mathbf{e}_h + \mathbf{e}_r$ 。然后使用 softmax 函数来归一化计算邻居实体的权重:

$$\tilde{\pi}(h, r, t) = \text{softmax}(\pi(h, r, t)) = \frac{\exp(\pi(h, r, t))}{\sum_{(h, r, t) \in \mathcal{N}(v)} \exp(\pi(h, r, t))} \quad (4)$$

计算得到的注意力得分能够使得模型每个节点关注应该从哪个邻居实体获得更多的信息。然后通过线性组合获得项目 v 的实体邻域结构:

$$\mathbf{v}_{\mathcal{N}(v)}^u = \sum_{e \in \mathcal{N}(v)} \tilde{\pi}(v, r, t) \mathbf{e} \quad (5)$$

式中: \mathbf{e} 是项目 v 的邻居实体的嵌入表示。

最后,进一步聚合项目的实体表示 \mathbf{v} 和它的邻域表示 $\mathbf{v}_{\mathcal{N}(v)}^u$ 为一个向量。设计了两种聚合函数^[2]:

(1) concat aggregator:将两个向量串联,然后进行非线性变换。

$$\text{agg}_{\text{concat}} = \text{LeakyReLU}(\mathbf{W}(\mathbf{v} \parallel \mathbf{v}_{\mathcal{N}(v)}^u) + \mathbf{b}) \quad (6)$$

式中: \mathbf{W} 是转换权重; \mathbf{b} 是偏置。

(2) sum aggregator:将两个向量相加,然后就进行线性变换:

$$\text{agg}_{\text{sum}} = \text{LeakyReLU}(\mathbf{W}(\mathbf{v} + \mathbf{v}_{\mathcal{N}(v)}^u) + \mathbf{b}) \quad (7)$$

聚合将项目及其邻域显式地连接在一起来表示项目嵌入,在该嵌入中包含了项目本身的信息和知识图谱中的邻域信息。

为了稳定注意力机制的学习过程,本文采用多头注意力^[10],在 K 个独立的关注机制上执行式(6)、式(7)的转换,然后将这些特征连接起来,从而得到如下的输出特征表示:

$$\mathbf{v}' = \parallel_{k=1}^K \sigma(\mathbf{v}_{\text{agg}}) \quad (8)$$

式中: $\mathbf{v}_{\text{agg}} = \text{agg}_{\text{concat}}$ 或者 $\mathbf{v}_{\text{agg}} = \text{agg}_{\text{sum}}$ 。

2.3 预测函数

实体的最终表示为本身的嵌入表示和它的邻域结构,将其视为该项目的一阶表示,即为单层注意嵌入传播的计算方法。这里可以将实体计算扩展到多层以获得目标项目的高阶结构信息,从而更广泛和更深入地探索用户的潜在兴趣。对于一个 n 层模型而言,最多可到达该节点的 n 跳邻居,即节点的高阶结构信息是通过 n 跳邻居节点结算得出的,学习新的实体嵌入向量的聚合过程和相邻多跳节点间信息的传递如图4所示。

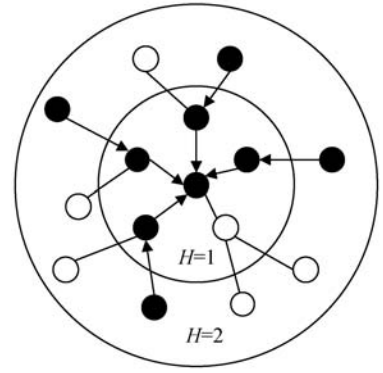


图4 聚合过程和信息传递

最后,将用户表示向量和最终得到的项目表示向量进行内积来计算预测概率:

$$\hat{y}(u, v) = \mathbf{u}^\top \mathbf{v}' \quad (9)$$

2.4 模型优化

在训练期间采用负采样策略,完整的损失函数定义为:

$$\mathcal{L} = \sum_{u \in U} \left(\sum_{v: y_{uv}=1} \mathcal{J}(y_{uv}, \hat{y}_{uv}) - \sum_{i=1}^{T_u} \mathbb{E}_{v_i \sim P(v_i)} \mathcal{J}(y_{uv}, \hat{y}_{uv}) \right) + \lambda \|\mathcal{F}\|_2^2 \quad (10)$$

式中: \mathcal{J} 是交叉熵损失函数; $P(v_i)$ 是负采样分布; T_u 是对于用户 u 的负采样数量, $T_u = |\{v: y_{uv}=1\}|$ 和 P 是遵循统一分布;最后一项是L2正则化。

3 实验与结果分析

3.1 数据集

本文中使用了三个数据集来进行实验比较,分别是电影数据集(movieLens-1M)、书籍推荐数据集(Book-Crossing)和音乐推荐数据集(Last.FM),下面分别介绍:

(1) MovieLens-1M 是电影推荐中广泛使用的基准数据集,它是由 MovieLens 用户提供的电影评分(评分为1~5)数据。

(2) Book-Crossing 是从 book-crossing 社团中收集的书籍评分(评分为0~10)信息。

(3) Last.FM 包含了从 Last.FM 在线音乐系统中收集的用户音乐评分数据。

在三个原始数据集中,用户的偏好都是通过用户评分体现,但是用户评分通常是显式反馈,在推荐算法中需要将显式反馈转换为隐式反馈。本文实验将用户评分较高的项目使用1进行标记,表示用户对该项目感兴趣,然后在用户未与之交互的项目集合中采样一个集合,标记为0。对于 MovieLens-1M,将评分阈值设置为4,对于 Book-Crossing 和 Last.FM 数据集本身的

稀疏性,将评分阈值设置为 0。本文使用原论文中的知识图谱进行计算。在表 1 中列出了实验中使用的三个数据集的基本数据信息。

表 1 三个数据集基本信息

基本数据	MovieLens-1M	Book-Crossing	Last. FM
#users	6 036	17 860	1 872
#items	2 442	14 967	3 846
#interactions	753 556	139 746	42 346
#entities	182 011	77 903	9 366
#relations	12	25	60
#KG triple	1 241 995	151 500	15 518

3.2 基线算法

使用四个推荐算法与本文算法进行比较,分别为 BPRMF、FM、CKE 和 KGCN。其中前两个推荐算法不使用知识图谱,后两个推荐算法均使用知识图谱作为辅助信息进行推荐。下面进行简单介绍:

(1) BPRMF^[16]是一个具有贝叶斯个人排名的矩阵分解算法。

(2) FM^[17]考虑了输入之间的二阶特征相互作用。

(3) CKE^[18]是一个将 CF 与知识图谱中的结构化内容、文本内容和视觉内容结合的统一推荐框架。

(4) KGCN^[4]是一个端到端的框架,可以在知识图谱上挖掘项目之间的关联属性来有效地捕获项目之间关联性,在计算知识图谱中给定实体的表示时,将带有偏差的邻域信息进行汇总和合并。

3.3 参数设置

对于三个数据集,将训练集、验证集和测试集的比率设置为 6:2:2。在实验过程中的三个数据集的参数设置如表 2 所示。其他基线算法的参数设置如下:对于 BPRMF,三个数据集的维度和学习率设置为:movieLens-1M: $d = 64, \eta = 2E - 3$, Book-Crossing: $d = 128, \eta = 2E - 1$, Last. FM: $d = 16, \eta = 2E - 2$ 。对于 FM:movieLens-1M: $d = 16, \eta = 2E - 2$, Book-Crossing: $d = 2, \eta = 5E - 4$, Last. FM: $d = 16, \eta = 2E - 2$ 。对于 CKE:三个数据集的维数分别为 64、128 和 64。所有数据集的 KG 部分训练权重为 0.1。对于 KGCN:使用与原论文相同的参数。其他超参数与原论文中相同或在其代码中的默认参数。

表 2 实验参数设置

数据集	参数
MovieLens-1M	$k = 8, d = 32, H = 1, \lambda = 1E - 7, \eta = 2E - 2$
Book-Crossing	$k = 8, d = 32, H = 2, \lambda = 1E - 5, \eta = 2E - 2$
Last. FM	$k = 8, d = 16, H = 1, \lambda = 1E - 6, \eta = 4E - 2$

使用两个评估方法:

(1) 在点击率预测中,通常使用经过训练的模型来预测测试集中的每个用户互动,通过 AUC 和 F1 来评估点击率预测效果。

(2) 在 top-k 推荐中,使用训练好的模型为测试集中的每个用户选择 n 个预测点击率最高的项目,然后使用 recall@k 评估推荐效果。

所有可训练的参数均由 Adam 算法优化。

3.4 实验结果分析

CTR 预测结果和 Top-K 推荐结果分别显示在表 3 和图 5 - 图 7 中,可以得到以下的观察结果:

(1) KGAT2Rec 在电影推荐和书籍推荐中具有最好的性能,相较于最佳基线算法分别提高了 3.67% 和 4.75%。在音乐推荐中推荐性能仅次于 KGCN,说明通过堆叠注意嵌入传播层,KGAT2Rec 能够有效捕获项目的协作信息来传递知识,从而提高算法的推荐效果。

(2) 对于 CKE 而言,其推荐性能仅优于 BPRMF,与 FM 的性能差距较大,这表明 CKE 无法通过类似 TransR 的方式充分使用 KG 中的知识信息。为了丰富项目的表示,FM 利用其连接实体的嵌入,而 CKE 仅使用其对齐实体的嵌入。此外,FM 中的交叉功能实际上充当用户和实体之间的第二级连接,CKE 在三元组的粒度上对连接进行建模,而高阶连接则保持不变。

(3) 与 FM 相比,KGCN 性能略胜于 FM 算法,说明在 KG 中捕获邻居信息可以提高推荐性能。

(4) KGAT2Rec-concat 相比于 KGAT2Rec-sum 在三个数据集上总体表现更好,这可能是因为 concat 相比于 sum 更好地保存了用户信息。

表 3 CTR 预测的 AUC 和 F1 结果

模型	MovieLens-1M		Book-Crossing		Last. FM	
	AUC	F1	AUC	F1	AUC	F1
BPRMF	0.500 7	0.666 4	0.544 7	0.615 2	0.507 7	0.641 0
FM	0.842 7	0.794 7	0.566 1	0.670 1	0.729 7	0.681 3
CKE	0.502 5	0.666 1	0.545 4	0.603 2	0.511 6	0.636 5
KGCN	0.885 9	0.808 7	0.662 5	0.630 4	0.760 3	0.694 0
KGAT2Rec-sum	0.910 5	0.839 7	0.693 8	0.625 3	0.731 3	0.682 8
KGAT2Rec-concat	0.918 4	0.849 2	0.694 0	0.632 1	0.735 7	0.686 7

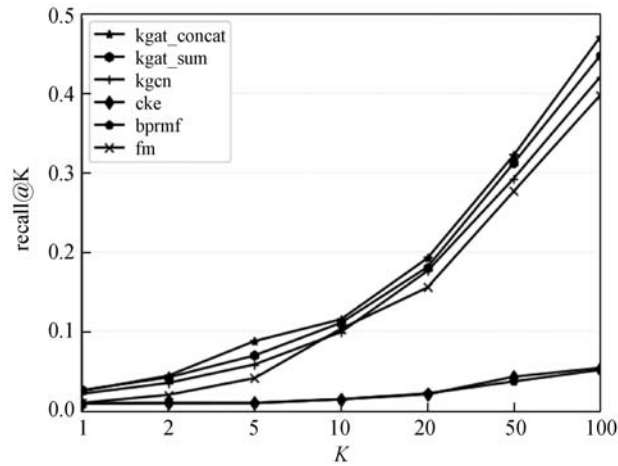


图 5 MovieLens-1M 数据集 top-K 推荐的 recall@K 值

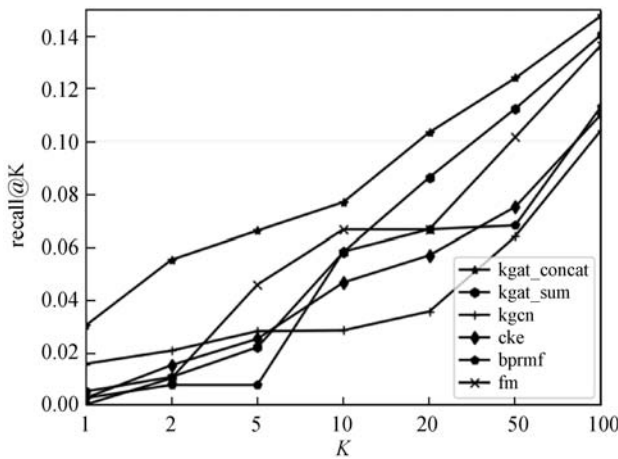


图 6 Book-Crossing 数据集 top-K 推荐的 recall@K 值

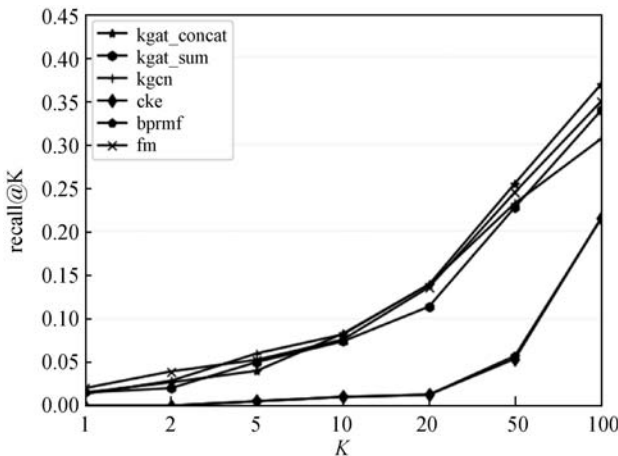


图 7 Last.FM 数据集 top-K 推荐的 recall@K 值

3.4.1 邻居数量的影响

通过改变邻居抽样的数量来考察使用 KG 的有效性。从表 4 中可以发现:

(1) 当 $K = 8$ 时, KGAT2Rec 可以获得最佳性能, 可能是因为, 邻居数量太少不能获得足够的邻域信息来进行推荐, 而邻居数量太多会带来更多的噪声使推荐效果下降。

(2) 联合表 3、表 4, 可以发现, 在大多数情况下, 使用 KG 中的邻居信息可以提高推荐的性能。

表 4 不同邻居抽样数量 K 的 AUC 值

数据集	K					
	2	4	8	16	32	64
MovieLens-1M	0.903	0.905	0.914	0.911	0.907	0.906
Book-Crossing	0.684	0.685	0.687	0.686	0.680	0.683
Last.FM	0.708	0.713	0.725	0.714	0.705	0.702

3.4.2 传播层数量的影响

改变 KGAT2Rec 中 KG 的邻居层数量, 来考察使用多个嵌入传播层的效率。将结果汇总到表 5 中, 可以发现:

(1) 当 $H = 1$ 或 $H = 2$ 时, 模型足以满足实际使用。较大的 H 会给模型带来大量的噪声, 也就是说在计算项目的相似性时, 使用较长的关系链意义很小, 甚至会造成错误的结果。

(2) 联合表 3 和表 5, 使用一层注意嵌入传播层的性能优于大多数基线算法, 从而表明了使用注意嵌入传播层可以提高推荐的性能。

表 5 不同传播层数量 H 的 AUC 值

数据集	N			
	1	2	3	4
MovieLens-1M	0.909 4	0.904 5	0.904 9	0.896 0
Book-Crossing	0.683 8	0.685 7	0.681 5	0.678 4
Last.FM	0.720 9	0.714 8	0.717 0	0.702 3

3.4.3 多头注意机制数量的影响

改变计算过程中注意力机制的数量, 来考察模型的性能。如表 6 所示, 当注意机制的数量为 2 时, 模型的性能最佳, 这说明使用多头注意机制可以稳定学习过程, 并且提高推荐的性能。

表 6 不同注意机制数量 N 的 AUC 值

数据集	N		
	1	2	3
MovieLens-1M	0.910 0	0.916 2	0.908 3
Book-Crossing	0.687 3	0.694 0	0.685 7
Last.FM	0.702 7	0.725 7	0.724 5

4 结 语

本文使用注意嵌入传播层来学习项目嵌入表示, 用来进行知识感知推荐。它可以自适应地学习来自邻居节点的嵌入, 用以更新项目对应实体的表示。同时在模型计算过程中, 也探索了用户的个性化和用户潜在兴趣。在三个真实世界数据集上进行的大量实验证

明了 KGAT2Rec 在推荐方面的有效性。

这项工作探索了图神经网络在推荐中的潜力,说明通过获得知识图谱中的结构信息可以提高推荐的性能。在现实世界中,还有其他的一些辅助信息,如项目内容信息、上下文信息和视觉信息,可以将其与知识图谱进行结合,提高推荐的性能,为推荐提供更多的可解释性。

参 考 文 献

- [1] He X N, He Z K, Song J K, et al. NAIS: Neural attentive item similarity model for recommendation[J]. IEEE Transactions on Knowledge & Data Engineering, 2018, 30 (12): 2354 - 2366.
- [2] He X N, Liao L Z, Zhang H W, et al. Neural collaborative filtering[C]//26th International Conference on World Wide Web, 2017: 173 - 182.
- [3] Wang X, He X N, Wang M, et al. Neural graph collaborative filtering[C]//42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019: 165 - 174.
- [4] Wang H W, Zhao M, Xie X, et al. Knowledge graph convolutional networks for recommender systems [EB]. arXiv: 1904. 12575, 2019.
- [5] Wang X, Wang D X, Xu C R, et al. Explainable reasoning over knowledge graphs for recommendation[C]//33rd AAAI Conference on Artificial Intelligence, 2019: 5329 - 5336.
- [6] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[C]//26th International Conference on Neural Information Processing, 2013: 2787 - 2795.
- [7] Wang Z, Zhang J W, Feng J W, et al. Knowledge graph embedding by translating on hyperplanes[C]//28th AAAI Conference on Artificial Intelligence, 2014: 1112 - 1119.
- [8] Lin Y K, Liu Z Y, Sun M, et al. Learning entity and relation embeddings for knowledge graph completion[C]//29th AAAI Conference on Artificial Intelligence, 2015: 2181 - 2187.
- [9] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[C]//International Conference on Logic Programming, 2017.
- [10] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks[EB]. arXiv: 1710. 10903, 2018.
- [11] Bruna J, Zaremba W, Szlam A, et al. Spectral networks and locally connected networks on graphs [EB]. arXiv: 1312. 6203v1, 2013.
- [12] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks [EB]. arXiv: 1609. 02907v1, 2016.
- [13] Hamilton W L, Ying R, Leskovec J. Inductive representation learning on large graphs[C]//31st International Conference on Neural Information Processing Systems, 2017: 1025 - 1035.
- [14] Qiu J Z, Tang J, Ma H, et al. DeepInf: Social influence prediction with deep learning[C]//24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018: 2110 - 2119.
- [15] Wang X, He X N, Cao Y X, et al. KGAT: Knowledge graph attention network for recommendation[C]//25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019: 950 - 958.
- [16] Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback [C]//25th Conference on Uncertainty in Artificial Intelligence, 2009: 452 - 461.
- [17] Rendle S. Factorization machines[C]//IEEE International Conference on Data Mining, 2010: 995 - 1000.
- [18] Zhang F Z, Yuan N J, Lian D F, et al. Collaborative knowledge base embedding for recommender systems [C]//22nd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2016: 353 - 362.
- ~~~~~
- (上接第 252 页)
- [12] Fan W Q, Li Q, Cheng M. Deep modeling of social relations for recommendation[C]//32nd AAAI Conference on Artificial Intelligence, 2018: 8075 - 8076.
- [13] Wang X, Hoi S, Ester M, et al. Learning personalized preference of strong and weak ties for social recommendation [C]//26th International Conference on World Wide Web, 2017: 1601 - 1610.
- [14] 吴国栋, 查志康, 涂立静, 等. 图神经网络推荐研究进展 [J]. 智能系统学报, 2020, 15 (1): 14 - 24.
- [15] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[EB]. arXiv: 1609. 02907, 2016.
- [16] Wu L, Sun P J, Hong R, et al. SocialGCN: An efficient graph convolutional network based model for social recommendation[EB]. arXiv: 1811. 02815v1, 2018.
- [17] Wu S, Tang Y, Zhu Y Q, et al. Session-based recommendation with graph neural networks[C]//33rd AAAI Conference on Artificial Intelligence, 2019: 346 - 353.
- [18] Yu J L, Yin H Z, Li J D, et al. Enhancing social recommendation with adversarial graph convolutional networks [EB]. arXiv: 2004. 02340, 2020.
- [19] Zhou G R, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction [C]//33rd AAAI Conference on Artificial Intelligence, 2019: 5941 - 5948.
- [20] Hamilton W, Ying R, Leskovec J. Inductive representation learning on large graphs [C]//31st International Conference on Neural Information Processing Systems, 2017: 1025 - 1035.
- [21] Linden G, Smith B, York J. Amazon. com recommendations: Item-to-item collaborative filtering[J]. IEEE Internet computing, 2003, 7 (1): 76 - 80.