软件学报 ISSN 1000-9825, CODEN RUXUEW
2025,36(12):5629−5643 [doi: 10.13328/j.cnki.jos.007401] [CSTR: 32375.14.jos.007401]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

带有差异化机制的多视角归纳式知识图谱补全框架*

童翰文[1,2], 钱羽希[4], 刘井平[3], 梁祖杰[4], 肖仰华[1,2], 韦 峰[4], 郝正鸿[4], 韩 冰[4]

[1](复旦大学 计算机科学技术学院, 上海 200438)

[2](上海市数据科学重点实验室 (复旦大学), 上海 200438)

[3](华东理工大学 信息科学与工程学院, 上海 200237)

[4](蚂蚁集团, 上海 200010)

通信作者: 刘井平, E-mail: jingpingliu@ecust.edu.cn; 肖仰华, E-mail: shawyh@fudan.edu.cn

摘 要: 知识图谱补全模型需要具备归纳能力, 才能够随着知识图谱的扩充泛化到新实体上. 然而, 现有的方法都只能通过聚合知识图谱中的邻居信息, 从一个局部的视角来理解实体的语义, 从而导致无法从不同的视角捕捉到实体之间的多种有价值的关联. 在局部视角以外, 通过非显式连接实体之间和远距离连接实体之间的交互, 从而以全局视角和序列视角来进一步理解实体是至关重要的. 更重要的是, 强调通过多个不同视角聚合到的信息应当是互补的, 而不是冗余的. 因此, 提出一个带有差异化机制的多视角知识图谱补全框架, 用于归纳式知识图谱补全任务. 它能够从多个不同视角学习到互补的、互不重叠的实体表示. 具体来说, 除了通过关系图卷积网络聚合邻居信息得到实体的局部表示外, 设计一种基于注意力的差异化机制, 用于从语义相关的实体和实体相关路径中聚合得到实体的全局和序列表示. 最终, 融合这些表示, 并基于它们给三元组打分. 实验结果证明, 所提方法在归纳式的设定下超越了当前最先进的方法. 此外, 所提方法在直推式的知识图谱补全任务中也保持着有竞争力的表现.

关键词: 归纳式知识图谱补全; 多视角框架; 差异化机制; 知识图谱

中图法分类号: TP182

中文引用格式: 童翰文, 钱羽希, 刘井平, 梁祖杰, 肖仰华, 韦峰, 郝正鸿, 韩冰. 带有差异化机制的多视角归纳式知识图谱补全框架. 软件学报, 2025, 36(12): 5629–5643. http://www.jos.org.cn/1000-9825/7401.htm

英文引用格式: Tong HW, Qian YX, Liu JP, Liang ZJ, Xiao YH, Wei F, Hao ZH, Han B. Multi-view Framework for Inductive Knowledge Graph Completion with Differentiation Mechanism. Ruan Jian Xue Bao/Journal of Software, 2025, 36(12): 5629–5643 (in Chinese). http://www.jos.org.cn/1000-9825/7401.htm

Multi-view Framework for Inductive Knowledge Graph Completion with Differentiation Mechanism

TONG Han-Wen[1,2], QIAN Yu-Xi[4], LIU Jing-Ping[3], LIANG Zu-Jie[4], XIAO Yang-Hua[1,2], WEI Feng[4], HAO Zheng-Hong[4], HAN Bing[4]

[1](School of Computer Science, Fudan University, Shanghai 200438, China)

[2](Shanghai Key Laboratory of Data Science (Fudan University), Shanghai 200438, China)

[3](School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China)

[4](Ant Group, Shanghai 200010, China)

Abstract: Knowledge graph completion (KGC) models require inductive ability to generalize to new entities as the knowledge graph expands. However, current approaches understand entities only from a local perspective by aggregating neighboring information, failing to

* 基金项目: 国家自然科学基金青年科学基金 (62306112); 上海市青年科技英才扬帆计划 (23YF1409400); 上海市基础研究特区计划 (22TQ1400100-20)
童翰文和钱羽希为共同第一作者.
收稿时间: 2024-04-07; 修改时间: 2024-11-26; 采用时间: 2025-02-10; jos 在线出版时间: 2025-06-11
CNKI 网络首发时间: 2025-06-11

© 中国科学院软件研究所 http://www.jos.org.cn

capture valuable interconnections between entities across different views. This study argues that global and sequential perspectives are essential for understanding entities beyond the local view by enabling interaction between disconnected and distant entity pairs. More importantly, it emphasizes that the aggregated information must be complementary across different views to avoid redundancy. Therefore, a multi-view framework with the differentiation mechanism is proposed for inductive KGC, aimed at learning complementary entity representations from various perspectives. Specifically, in addition to aggregating neighboring information to obtain the entity's local representation through R-GCN, an attention-based differentiation mechanism is employed to aggregate complementary information from semantically related entities and entity-related paths, thus obtaining global and sequential representations of the entities. Finally, these representations are fused and used to score the triples. Experimental results demonstrate that the proposed framework consistently outperforms state-of-the-art approaches in the inductive setting. Moreover, it retains competitive performance in the transductive setting.

Key words: inductive knowledge graph completion; multi-view framework; differentiation mechanism; knowledge graph

1 引 言

知识图谱是一种多关系图, 其中节点表示实体, 边表示关系. 它们对于各种各样的知识密集型应用都至关重要, 比如问答系统、推荐系统以及对话系统[1,2]. 然而, 现实中的知识图谱通常都是不完备的, 这就需要运用知识图谱补全技术补充其中的缺失边, 从而促进知识图谱的自动构建和验证. 更重要的, 现实中的知识图谱通常是动态变化的, 因为总会有实体持续地被添加或删除. 因此, 真实可用的知识图谱补全方法需要具有归纳性, 从而能够泛化到加入知识图谱的新实体上. 例如, 在图 1 中, 给定一个三元组"(China, participated in, 2008 Summer Olympics)", 模型就需要能够泛化到新实体"2012 Summer Olympics"上, 并且能够预测"China"和"2012 Summer Olympics"之间的缺失边.

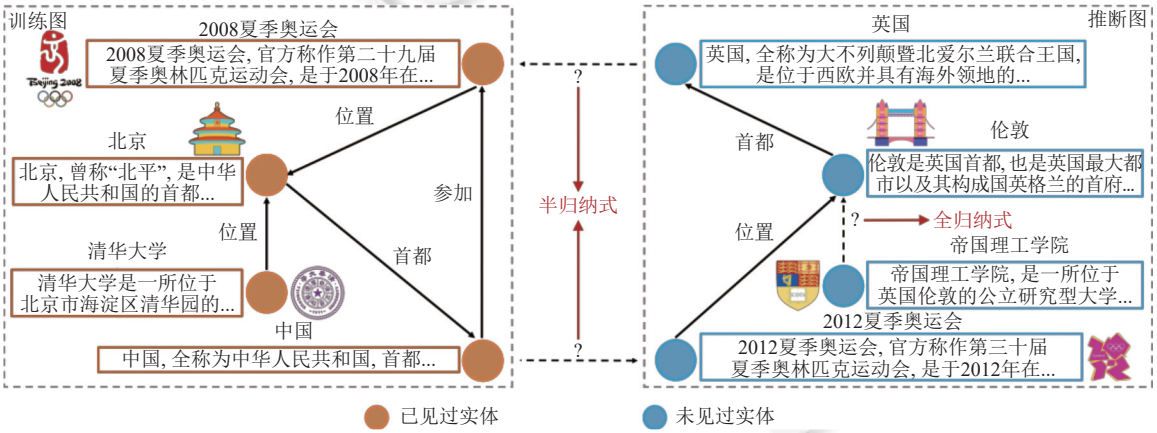


图 1 归纳式知识图谱补全的示例

为了在进行知识图谱补全时建模新的实体, 现有工作大多使用基于文本的方法. 它们使用预训练语言模型来编码每个实体相关的文本得到实体的表示, 然后利用传统的知识图谱补全模型对三元组打分, 比如 TransE[3−5]或余弦相似度[6,7]. 然而, 这些方法都有一定的局限性, 因为它们只利用了实体相关的文本信息而忽视了知识图谱天然的图结构. 为了更进一步引入知识图谱的结构信息, 一系列工作, 例如 StATIK[8]就提出了一些混合模型. 它们在预训练语言模型之上堆叠图神经网络以融合文本和结构信息. 然而, 这些工作也有局限性. 它们只能够从每个实体的邻居聚合信息, 从知识图谱的局部视角学习实体的表示. 结果是, 它们无法捕获可从知识图谱的其他视角中得出的有价值信息.

本文认为除了局部视角以外, 探索知识图谱中的全局视角和序列视角对于理解实体的语义是至关重要的. (1) 从全局视角学习意味着通过知识图谱中其他相关的实体来理解一个实体, 不管它们之间是否有边相连. 知识图谱中的实体除了会被显式边相连, 也会存在隐式的语义关联, 而这种关联往往不会以显式边的形式表达. 因此, 从

© 中国科学院软件研究所 http://www.jos.org.cn

相关的实体上聚合语义信息可以帮助增强一个实体的语义, 特别是在归纳式设定下理解未见过的实体. 例如, 在图 1 中, 尽管一个见过的实体“2008 Summer Olympics”和一个未见过的实体“2012 Summer Olympics”之间没有显式的边, 它们却拥有相同的概念和相似的语义. 对于“2012 Summer Olympics”来说, 从一个见过的实体“2008 Summer Olympics”聚合语义信息可以建立起见过和未见过实体之间的联系, 从而进一步帮助理解未见过实体的语义. (2) 从序列视角学习意味着从实体相关联的路径上理解一个实体. 知识图谱中的实体除了有局部邻居外, 路径还会将实体和距离它若干跳的实体连接起来, 显式地反映出知识图谱中实体间的多跳关联. 因此, 通过允许远距离但相连的实体之间进行交互, 便能够沿着路径聚合实体信息, 从而捕捉到实体间的多跳关联, 更好地理解实体的语义. 而且, 这种关联也是无法通过局部或全局视角捕捉到的. 例如, 在图 1 中, 沿着一条路径“(2008 Summer Olympics, location, Beijing), (Beijing, capital of, China)”聚合信息就能够保证“2008 Summer Olympics”和一个远距离但相连的实体“China”之间产生交互. 这会有助于更好地理解“2008 Summer Olympics”并且识别出它和“China”之间的关系. 更重要的是, 这其中存在的一个潜在的问题是从多个视角聚合到的信息可能会出现冗余. 例如, 一个语义关联的实体可能恰好也是一个邻居实体, 就像三元组“(The Dark Knight Rises, prequel, The Dark Knight)”. 两部电影“The Dark Knight Rises”和“The Dark Knight”既是语义相似的实体, 又是邻居实体, 这可能会导致从局部视角和全局视角聚合得到重叠的信息, 进一步导致通过多视角捕捉互补信息的目标失效. 因此, 本文强调从多个视角聚合信息时避免冗余, 确保聚合信息的多样性和互补性是至关重要的.

本文为了从多个视角全面理解实体的语义, 以更好地进行归纳式知识图谱补全, 提出了一个带有差异化机制的多视角知识图谱补全框架. 该框架包含一个带有差异化机制的多视角编码器和一个打分解码器. 编码器的目标是从局部、全局、序列这 3 个视角学习互补的实体表示, 而解码器的目标是基于这些习得的表示对三元组打分. 具体而言, 本文首先使用一个预训练语言模型来编码每个实体的名称和描述得到它的文本表示. 基于该表示, 本文使用 3 个模块来从多个视角聚合多样化的信息. (1) 结构编码模块使用关系图卷积网络[9]建模知识图谱的空间局部性[6], 聚合邻居信息以得到实体的局部表示. (2) 全局编码模块使用一种全局注意力机制来建模知识图谱中任何一对实体之间的语义相关性, 从相关实体聚合语义信息来获得实体的全局表示. 本文进一步设计了一种基于注意力的差异化机制, 以缓解局部表示和全局表示之间的过度注意问题, 避免全局视角聚合和局部视角重叠的信息. (3) 路径编码模块使用一个带有位置编码的 Transformer[10]编码层来建模实体间的多跳关联, 沿着多条实体路径聚合实体信息以得到实体的序列表示. 和全局视角类似, 序列视角依然使用差异化机制来避免它从多条路径聚合序列结构信息时的信息冗余. 最终, 本文通过一个融合层对各种各样视角学到的表示进行融合, 得到最终的实体表示. 在打分解码器中, 本文使用一个基于转移的模型 RotatE[11], 基于实体的最终表示对三元组打分. 正确的三元组应当得到尽可能高的分数.

本文的研究贡献总结如下.

(1) 为归纳式知识图谱补全任务提出了一个新颖的、带有差异化机制的多视角知识图谱补全框架. 最显著的特点是它保证了知识图谱中局部、全局、序列视角信息聚合的多样性和互补性, 从而学习到多个视角互补的实体表示.

(2) 使用一个带有差异化机制的多视角编码器和一个打分解码器实现了上述框架. 编码器使用了一个基于注意力的差异化机制来避免过度注意的问题, 从各种视角学习独特的实体表示. 解码器基于实体表示, 使用一个基于转移的模型对三元组打分.

(3) 在归纳式设定下对所提方法进行了充分的实验. 实验结果证明它超越了现有最先进的方法. 此外, 所提方法在直推式设定下也保持着有竞争力的性能.

本文第 2 节介绍知识图谱补全的相关工作和研究现状. 第 3 节给出问题定义并且对方法进行整体概述. 第 4 节详细介绍所提方法, 包括了带有差异化机制的多视角编码器、打分解码器以及模型的训练与推断过程. 第 5 节给出本文方法的实验结果, 包括了与各个基线模型的实验结果对比, 以及对各个模块的分析与消融实验. 第 6 节总结全文.

© 中国科学院软件研究所 http://www.jos.org.cn

2 相关工作

本节回顾知识图谱补全的相关工作和研究现状, 并且主要关注于归纳式的设定.

2.1 直推式知识图谱补全

直推式知识图谱补全是一种传统的知识图谱补全设定, 意味着在模型推断时出现的实体都是在训练中见过的. 传统的知识图谱补全方法是基于嵌入的[12], 天然地是直推式的. 它们学习静态的实体和关系表示, 用于给三元组打分. 根据打分函数的不同, 可以将它们划分为两类: 基于转移的模型和语义匹配模型. 基于转移的模型, 如 TransE[3]和 TransH[13], 把一个三元组视作从头实体向尾实体的一个特定于关系的转移. ComplEx[14]和 RotatE[11]将实体和关系表示引入复数空间, 进一步提高模型的表征能力. 语义匹配模型, 如 RESCAL[15]和 DistMult[16], 通过头实体、关系、尾实体之间的语义是否匹配来判断一个三元组是否成立. 近期, 基于图的方法, 如 R-GCN[17]和 CompGCN[18], 利用知识图谱的图结构, 通过对局部子图的信息聚合来学习实体表示. 然而, 这些方法往往无法直接应用到归纳式设定下, 因为它们无法为未见过的新实体产生合理的表示.

2.2 归纳式知识图谱补全

归纳式知识图谱补全意味着在模型推断时会有训练中未见过的新实体加入. 为了更好地建模未见过的实体, 近期的研究关注于基于文本的方法, 通过编码实体文本来得到实体的表示. DKRL[19]使用卷积神经网络编码实体描述得到实体的表示. KG-BERT[20]将头实体的描述、关系文本和尾实体的描述拼接成 BERT[21]的输入序列, 然后使用编码后的表示对三元组进行打分或分类. BLP[4]和 StAR[5]进一步引入转移模型, 在通过 BERT 编码实体描述的基础上给三元组打分. SimKGC[6]引入了对比学习, 将头实体的描述和关系拼接在一起, 和尾实体的描述分别通过 BERT 进行编码, 得到头实体的关系感知的表示和尾实体的表示, 然后通过它们之间的余弦相似度对三元组打分. Bi-Link[7]进一步引入了概率性的语法提示, 然后通过一个对比学习框架来学习可以泛化到大型知识图谱上的关系模式. 然而, 所有这些方法都忽视了知识图谱显式的图结构. GraIL[22]和 BERTRL[23]利用知识图谱中的局部子图和路径完成归纳式的关系预测. RMPI[24]组合了子图结构和本体模式, 从实体视角和关系视角进行消息传递, 从而基于聚合到的表示为三元组打分. 然而, 这些方法都太过耗时, 难以在一个包含了知识图谱中所有实体的大规模候选实体集上进行链接预测. StATIK[8]提出了一个更高效的双塔模型, 以语言模型为骨架, 在此之上堆叠了一种消息传播神经网络来聚合实体的邻居信息. 然而, StATIK 的局限性在于它仅能够聚合邻居信息, 而本文的方法能够在保证效率的前提下, 通过从知识图谱的局部、全局和序列视角聚合互补的信息, 全面地捕捉到知识图谱的局部结构, 以及实体之间的隐式语义关联和多跳关联.

3 问题定义与方法概述

本节首先对归纳式知识图谱补全任务进行形式化的定义, 然后简短地概述本文的方法, 该方法的架构见后文图 2.

3.1 问题定义

知识图谱可以被定义为一个有向图 $\mathcal{G}=(\mathcal{E},\mathcal{R},\mathcal{T})$. $\mathcal{E}$ 表示实体集, $\mathcal{R}$ 表示关系集, 而 $\mathcal{T}$ 表示所有三元组 $(h,r,t)\in\mathcal{E}\times\mathcal{R}\times\mathcal{E}$ 的集合, 且 h、r 和 t 分别对应头实体、关系和尾实体. 对于归纳式知识图谱补全, 存在一个训练图 $\mathcal{G}_{\text{train}}=(\mathcal{E}_{\text{train}},\mathcal{R},\mathcal{T}_{\text{train}})$, 满足 $\mathcal{E}_{\text{train}}\subset\mathcal{E}$, $\mathcal{T}_{\text{train}}\subset\mathcal{T}$ 且 $\mathcal{T}_{\text{train}}$ 中的三元组仅包含训练实体集 $\mathcal{E}_{\text{train}}$ 中的实体. 该任务的目标是利用在 $\mathcal{G}_{\text{train}}$ 上训练的模型对测试三元组 $(h',r,t')\in\mathcal{T}-\mathcal{T}_{\text{train}}$ 进行头实体或尾实体的预测, 其中满足 $(h'\notin\mathcal{E}_{\text{train}})\vee(t'\notin\mathcal{E}_{\text{train}})$. 具体而言, 给定一个查询 $(h',r,?)$ 或 $(?,r,t')$, 任务目标是对所有的尾实体候选或头实体候选进行排序, 满足目标实体的排序尽可能高.

3.2 方法概述

本文的框架包含了一个带有差异化机制的多视角编码器和一个打分解码器. 给定一个知识图谱 $\mathcal{G}=(\mathcal{E},\mathcal{R},\mathcal{T})$, 编码器的目标是通过局部、全局和序列视角聚合它们互补的信息, 从而学到好的实体表示 $\boldsymbol{E}\in\mathbb{R}^{|\mathcal{E}|\times 3d}$. 具体而言,

© 中国科学院软件研究所 http://www.jos.org.cn

本文首先将每个实体的名称和描述拼接起来, 并将其输入到一个预训练语言模型中, 得到实体的初始表示. 之后, 本文使用 3 个模块, 通过从多个视角聚合互补的信息来丰富实体的表示: (1) 结构编码模块使用关系图卷积网络来聚合邻居信息, 获得实体的局部表示 $\boldsymbol{E}^l \in \mathbb{R}^{|\mathcal{E}|\times d}$. (2) 全局编码模块利用一种全局注意力机制来允许知识图谱中任何实体之间的语义交互, 从而获得实体的全局表示 $\boldsymbol{E}^g \in \mathbb{R}^{|\mathcal{E}|\times d}$. 本文设计了一种基于注意力的差异化机制来缓解局部表示和全局表示之间的过度注意的问题, 避免全局视角聚合和局部视角重叠的信息. (3) 路径编码模块使用一个带有位置编码的 Transformer[10]编码层来编码多条实体相关的路径, 并且使用一个类似的差异化机制来聚合路径表示, 从而得到实体的序列表示 $\boldsymbol{E}^s \in \mathbb{R}^{|\mathcal{E}|\times d}$. 这种差异化机制同样避免了路径视角聚合和局部视角相冗余的信息. 最终, 本文融合 $\boldsymbol{E}^l$, $\boldsymbol{E}^g$, $\boldsymbol{E}^s$ 来得到最终的实体表示 $\boldsymbol{E}$. 在解码器中, 给定一个三元组 (h,r,t), 本文使用一个基于转移的模型 RotatE[11]来基于实体表示 $\boldsymbol{e}_h, \boldsymbol{e}_t \in \boldsymbol{E}$ 和关系表示 $\boldsymbol{r} \in \boldsymbol{R}$, $\boldsymbol{R} \in \mathbb{R}^{|\mathcal{R}|\times 3d}$ 给三元组打分. 例如, 对于尾实体预测 $(h,r,?)$, 本文会计算以每个实体 $t_i \in \mathcal{E}$ 为尾实体时的三元组的分数, 并且预测得分最高的实体作为输出, 如公式 (1) 所示:

$$\text{argmax}\phi(h,r,t_i),\ t_i \in \mathcal{E} \tag{1}$$

其中, $\phi(\cdot,\cdot,\cdot)$ 表示 RotatE 模型.

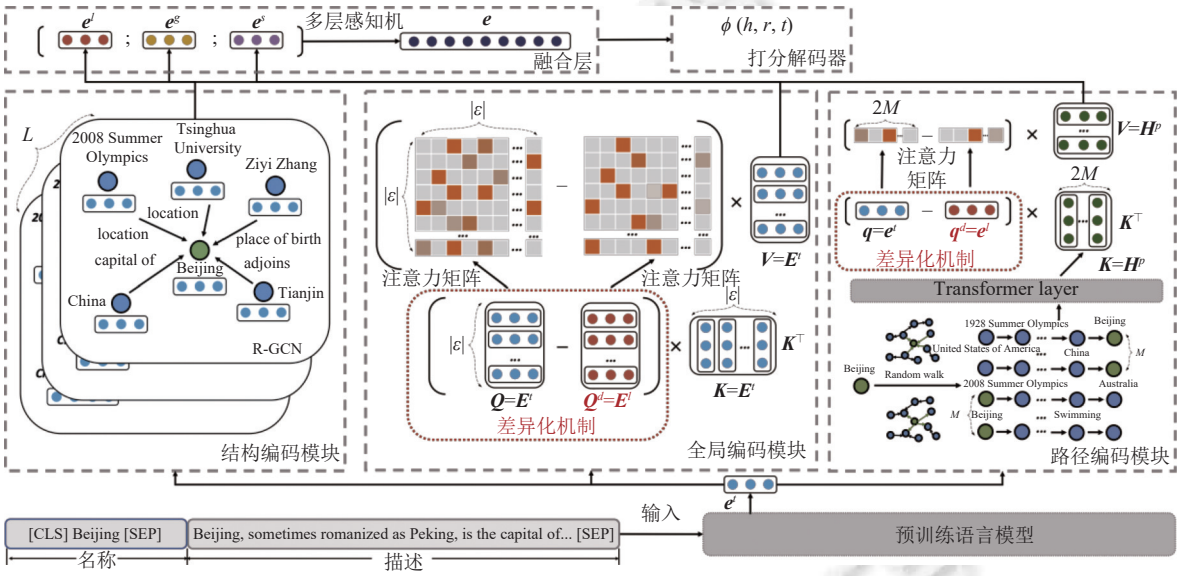


图 2 带有差异化机制的多视角知识图谱补全框架的整体架构

4 方 法

本节首先对本文的框架进行详细的介绍, 包括了带有差异化机制的多视角编码器以及打分解码器; 然后, 介绍本文框架的模型训练和推断过程.

4.1 多视角编码器

本节首先介绍利用预训练语言模型获取初始实体表示的方式; 然后, 详细阐述用于从局部、全局、序列视角聚合互补信息的模块; 最终, 介绍用于融合多视角表示得到最终实体表示的融合层.

4.1.1 预训练语言模型

考虑到预训练语言模型强大的文本编码能力, 本文使用 BERT[21]来编码每个实体相关的文本, 从而捕捉到它的语义信息并生成初始的实体表示. 具体而言, 本文将每个实体的名称和文本描述通过 [SEP] 这个特殊的 token 拼接, 构造 BERT 的输入序列. 本文将实体名称表示为 $X^m = \left[x_1^m, x_2^m, \ldots, x_{n_m}^m\right]$, 实体描述表示为 $X^d = \left[x_1^d, x_2^d, \ldots, x_{n_d}^d\right]$, 其中 n_m 和 n_d 分别是实体名称和描述的长度. 通过这种方式, 输入文本 X' 表示为 [[CLS], X^m, [SEP], X^d, [SEP]]. 本文首

© 中国科学院软件研究所 http://www.jos.org.cn

先将 X' 输入 BERT 来得到每个 token 的上下文相关的表示, 然后对 [[CLS], X^m, [SEP]] 的 token 表示进行平均池化操作, 最终得到实体的文本表示 $\boldsymbol{e}^t$ 作为它的初始表示. 文本编码过程可以描述如下:

$$\boldsymbol{e}^m_{[\mathrm{CLS}]}, \boldsymbol{e}^m_1, \ldots, \boldsymbol{e}^m_{n_m}, \boldsymbol{e}^m_{[\mathrm{SEP}]}, \boldsymbol{e}^d_1, \ldots, \boldsymbol{e}^d_{n_d}, \boldsymbol{e}^d_{[\mathrm{SEP}]} = f_{\mathrm{BERT}}(X') \tag{2}$$

$$\boldsymbol{e}^t = MEAN\text{-}POOLING\left(\boldsymbol{e}^m_{[\mathrm{CLS}]}, \boldsymbol{e}^m_1, \ldots, \boldsymbol{e}^m_{n_m}, \boldsymbol{e}^m_{[\mathrm{SEP}]}\right) \tag{3}$$

最终, 本文得到了所有实体的初始表示, 表示为 $\boldsymbol{E}^t = \left[\boldsymbol{e}^t_1, \ldots, \boldsymbol{e}^t_{|\mathcal{E}|}\right] \in \mathbb{R}^{|\mathcal{E}| \times d}$.

4.1.2 结构编码模块

结构编码模块的目标是通过从邻居实体聚合信息, 以一个局部的视角学习实体的表示. 知识图谱通常都展现出一种空间局部性[6], 表现为实体和它们的邻居实体密切相关, 而且它们之间的关系显式地表现了它们之间的关联类型. 为了捕捉到这种局部性并且更好地理解实体的语义, 本文使用 R-GCN[17]来允许带有不同关系类型的邻居的信息传播. 具体来说, 对于实体 i, 本文使用它的文本表示 $\boldsymbol{e}^t_i$ 作为它输入 R-GCN 的初始表示. 信息聚合的过程如公式 (4) 所示:

$$\boldsymbol{h}_i^{(l+1)} = \sum\nolimits_{r \in \mathcal{R}} \sum\nolimits_{j \in \mathcal{N}^r(i)} \frac{1}{|\mathcal{N}^r(i)|} \left(\boldsymbol{W}_r^{(l)} \boldsymbol{h}_j^{(l)} + \boldsymbol{W}_o^{(l)} \boldsymbol{h}_i^{(l)}\right) \tag{4}$$

其中, $\boldsymbol{h}_i^{(l)}$ 表示实体 i 在第 l 层 R-GCN 的隐状态且满足 $\boldsymbol{h}_i^{(0)} = \boldsymbol{e}^t_i$, $\mathcal{N}^r(i)$ 表示实体 i 关于关系 r 的邻居实体集, $\boldsymbol{W}_r^{(l)}$ 表示第 l 层 R-GCN 的关于关系 r 的关系感知变换矩阵, $\boldsymbol{W}_o^{(l)}$ 表示第 l 层 R-GCN 的自我循环权重. 本文将实体 i 在最后一层 R-GCN 的表示作为它的局部表示. 最终, 所有实体的局部表示为 $\boldsymbol{E}^l = \left[\boldsymbol{e}^l_1, \ldots, \boldsymbol{e}^l_{|\mathcal{E}|}\right] \in \mathbb{R}^{|\mathcal{E}| \times d}$.

4.1.3 全局编码模块

全局编码模块的目标是通过保证任何一对实体之间的语义交互, 从全局的视角学习实体的表示. 由于结构编码模块只考虑了通过显式边相连的邻居实体, 它无法捕获到没有显式边相连的实体之间的语义关联, 特别是对于一些新添加的、未见过的实体. 为了克服这样的局限性, 本文提出一种全局注意力机制, 以确保任何一对实体之间的信息聚合. 更重要的, 为了防止全局编码模块聚合了和结构编码模块重叠的信息, 本文设计了一种基于注意力的差异化机制来缓解这种过度注意的问题, 鼓励当前模块学习到一个和其他模块不同的表示.

受到 Transformer[10]架构的自注意力机制的启发, 对于每个实体, 本文提出计算任何一对实体之间的注意力权重, 并且基于该权重聚合所有实体的表示. 为了更好地区分聚合得到的全局表示和局部表示, 本文在计算注意力权重时引入了一种差异化机制来避免过度注意的问题. 在有了初始实体表示 $\boldsymbol{E}^t$ 后, 信息聚合过程如公式 (5) 和公式 (6) 所示:

$$\boldsymbol{Q} = \boldsymbol{E}^t, \boldsymbol{K} = \boldsymbol{E}^t, \boldsymbol{V} = \boldsymbol{E}^t, \boldsymbol{Q}^d = \boldsymbol{E}^l \tag{5}$$

$$\tilde{\boldsymbol{E}}^g = Softmax\left(\frac{(\boldsymbol{Q} - \boldsymbol{Q}^d)\boldsymbol{K}^{\top}}{\sqrt{d_k}} + \boldsymbol{M}\right)\boldsymbol{V}, \ \boldsymbol{M}_{ij} = \begin{cases} 0, & i \neq j \\ -\infty, & i = j \end{cases} \tag{6}$$

其中, $\tilde{\boldsymbol{E}}^g \in \mathbb{R}^{|\mathcal{E}| \times d}$ 表示所有实体从其他所有实体聚合得到的表示. 掩码矩阵 $\boldsymbol{M} \in \mathbb{R}^{|\mathcal{E}| \times |\mathcal{E}|}$ 决定了一对实体是否可以互相计算注意力. 具体来说, $\boldsymbol{M}$ 的主对角线元素都被设置为 $-\infty$ 来避免实体和其自身计算注意力, 而其他元素都被设置为 0 来保证实体可以和除它以外的所有实体计算注意力. 这种方式保证了计算得到的注意力权重不会被自身和自身之间参照的信息所主导. d_k 表示每一个自注意力头的维度. 本文设计的差异化机制的目标是使全局编码模块关注于结构编码模块所不关注的实体. 具体来说, 通过 $(\boldsymbol{Q} - \boldsymbol{Q}^d)$ 的操作, 通过 $\boldsymbol{Q}^d$ 计算出来注意力权重更高的实体会得到一个更低的注意力权重. 最终, 从这些实体通过加权求和操作聚合得到的信息就会有更低的重要性.

最终, 本文通过一个带有残差连接[25]和层归一化[26]的前馈神经网络得到所有实体的全局表示 $\boldsymbol{E}^g = [\boldsymbol{e}^g_1, \ldots, \boldsymbol{e}^g_{|\mathcal{E}|}] \in \mathbb{R}^{|\mathcal{E}| \times d}$. 具体计算过程如下所示:

$$FFN\left(\tilde{\boldsymbol{E}}^g\right) = \tilde{\boldsymbol{E}}^g \boldsymbol{W}^g + \boldsymbol{b}^g \tag{7}$$

$$\boldsymbol{E}^g = LayerNorm\left(FFN\left(\tilde{\boldsymbol{E}}^g\right) + \boldsymbol{E}^t\right) \tag{8}$$

4.1.4 路径编码模块

路径编码模块的目标是通过沿着连接多个实体的路径聚合信息, 从序列的视角学习实体的表示. 路径是一系

© 中国科学院软件研究所 http://www.jos.org.cn

列由关系实体所组成的序列, 显式地反映了实体之间的多跳关联. 通过沿着路径聚合实体表示, 路径编码模块使得远距离相连的实体之间也可以进行信息交互, 从而可以学习到一个能有效捕捉到多跳关联的实体表示. 需要注意的是, 这样的关联是无法通过结构编码模块或全局编码模块所捕捉的. 前者受到图神经网络所固有的过度平滑问题[9]的约束, 限制了它处理远距离实体的能力; 而后者则是仅考虑了实体之间的语义相关性, 却忽略了知识图谱天然的图结构.

为了沿着路径聚合实体的表示, 本文首先使用一种随机游走算法[27]的变体来采样和每个实体相关联的两种方向的路径. 然后, 本文使用一个带有位置编码的 Transformer 编码层来对采样得到的路径编码, 以得到每个路径的表示. 最终, 本文再使用和全局编码模块类似的差异化机制来聚合每个路径的表示, 得到实体的序列表示. 具体来说, 对于实体 i, 本文使用随机游走[27]来采样 M 条从 i 开始的正向路径, 以及 M 条以 i 结尾的反向路径. 每条路径的长度都设置为 K, 并且较短的路径会被填充到长度为 K. 为了增强路径上实体之间的关联性, 本文提出将每个实体关联到它能够通过随机游走到达的受欢迎的实体. 这种方式可以使得它利用这些受欢迎实体的丰富的信息来丰富它自身的表示. 本文通过在随机游走时给每条边 (u,v) 设置一个采样概率 $p_{u,v} \propto deg(u)+deg(v)$ 来实现这样的想法, 其中, $deg(\cdot)$ 表示一个实体的度. 通过这种方式, 本文获得 M 条正向的路径 $\overrightarrow{\mathcal{P}}=\{\overrightarrow{P}_1,\overrightarrow{P}_2,\ldots,\overrightarrow{P}_M\}$ 和 M 条反向的路径 $\overleftarrow{\mathcal{P}}=\{\overleftarrow{P}_1,\overleftarrow{P}_2,\ldots,\overleftarrow{P}_M\}$. 最终, 对于每个实体 i, 本文再反转它所有正向路径的实体顺序, 使得它们和所有反向路径的格式保持一致, 都以实体 i 结尾, 从而得到总共 $2M$ 条路径 $\mathcal{P}=\{P_1,P_2,\ldots,P_{2M}\}$.

本文使用一个带有旋转位置编码 (rotary positional embedding, RoPE)[28] 的 Transformer 编码层来编码采样的路径. 它的模型架构和第 4.1.1 节的预训练语言模型是保持一致的. 而且, 本文使用 RoPE 来编码路径中的位置信息, 从而更好地捕捉到它的序列结构. 对于路径 $P_j=[j_1,j_2,\ldots,i]$, 本文使用实体的文本表示 $\boldsymbol{E}^t$ 来初始化每个实体的表示, 得到 $\boldsymbol{P}_j=\left[\boldsymbol{e}^t_{j_1},\boldsymbol{e}^t_{j_2},\ldots,\boldsymbol{e}^t_i\right]\in\mathbb{R}^{K\times d}$. 然后, 本文使用带有 RoPE 的 Transformer 编码层来编码这些路径, 并且使用每条路径中实体 i 经过编码后的表示作为路径的表示. 具体计算过程如下:

$$\boldsymbol{h}^p_j=TransEnc(\boldsymbol{P}_j)[i] \tag{9}$$

$$\boldsymbol{H}^p=\left[\boldsymbol{h}^p_j\right],\ j\in[1,\ 2M] \tag{10}$$

其中, $TransEnc(\cdot)$ 表示带有 RoPE 的 Transformer 编码层, $\boldsymbol{h}^p_j\in\mathbb{R}^d$ 表示路径 P_j 的表示, $\boldsymbol{H}^p\in\mathbb{R}^{2M\times d}$ 表示所有基于实体 i 采样的 $2M$ 条路径的表示.

在得到 $\boldsymbol{H}^p$ 后, 本文使用全局编码模块中的差异化机制来聚合所有路径表示. 具体计算过程如下所示:

$$\boldsymbol{q}=\boldsymbol{e}^t_i,\boldsymbol{K}=\boldsymbol{H}^p,\boldsymbol{V}=\boldsymbol{H}^p,\boldsymbol{q}^d=\boldsymbol{e}^l_i \tag{11}$$

$$\tilde{\boldsymbol{e}}^s_i=Softmax\left(\frac{(\boldsymbol{q}-\boldsymbol{q}^d)\boldsymbol{K}^\top}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{12}$$

其中, $\tilde{\boldsymbol{e}}^s_i$ 表示实体 i 从路径表示 $\boldsymbol{H}^p$ 聚合得到的表示. 最终, 本文通过一个带有残差连接和层归一化的前馈神经网络得到实体 i 的序列表示, 计算过程如下:

$$FFN(\tilde{\boldsymbol{e}}^s_i)=\tilde{\boldsymbol{e}}^s_i\boldsymbol{W}^s+\boldsymbol{b}^s \tag{13}$$

$$\boldsymbol{e}^s_i=LayerNorm(FFN(\tilde{\boldsymbol{e}}^s_i)+\boldsymbol{e}^t_i) \tag{14}$$

最终, 所有实体的序列表示为 $\boldsymbol{E}^s=\left[\boldsymbol{e}^s_1,\ldots,\boldsymbol{e}^s_{|\mathcal{E}|}\right]\in\mathbb{R}^{|\mathcal{E}|\times d}$.

4.1.5 融合层

给定各种视角学习到的实体表示 $\boldsymbol{E}^l$, $\boldsymbol{E}^g$, $\boldsymbol{E}^s$, 融合层的目标是融合这些表示, 从而产生实体的最终表示 $\boldsymbol{E}$. 融合过程如公式 (15) 所示:

$$\boldsymbol{E}=\left[\boldsymbol{E}^l;\boldsymbol{E}^g;\boldsymbol{E}^s\right]\boldsymbol{W}^f+\boldsymbol{b}^f \tag{15}$$

其中, $\boldsymbol{E}\in\mathbb{R}^{|\mathcal{E}|\times 3d}$ 是所有实体的最终表示, $[\cdot;\cdot]$ 表示拼接操作, $\boldsymbol{W}^f\in\mathbb{R}^{3d\times 3d}$ 和 $\boldsymbol{b}^f\in\mathbb{R}^{3d}$ 分别是线性变换的权重矩阵和偏置.

© 中国科学院软件研究所 http://www.jos.org.cn

4.2 打分解码器

本节介绍打分解码器, 它的目标是基于多视角编码器所习得的实体表示对三元组打分. 给定多视角编码器产生的实体表示 $\boldsymbol{E}$, 可训练的关系表示 $\boldsymbol{R}$ 和一个三元组 (h,r,t), 打分解码器的目标是在该三元组是正确的情况下, 赋予它一个尽可能高的分数, 否则赋予一个尽可能低的分数.

本文使用一个先进的、基于转移的模型 RotatE[11], 作为打分解码器. 具体来说, 本文将 h 和 t 的表示分别标记为 $\boldsymbol{e}_h$ 和 $\boldsymbol{e}_t$, r 的表示标记为 $\boldsymbol{r}\in\boldsymbol{R}$. 打分解码器的计算过程如下所示:

$$\phi(h,r,t)=\gamma-\|\boldsymbol{e}_h\circ\boldsymbol{r}-\boldsymbol{e}_t\| \tag{16}$$

其中, "$\circ$"指哈达玛积, 它代表着复数空间中通过旋转进行平移的操作; γ 是间隔超参数.

4.3 模型训练与推断

本节详细介绍本文框架的模型训练和推断过程. 本文使用对比学习来训练模型, 期望最大化正确三元组的分数, 同时最小化错误三元组的分数. 本文将知识图谱中已有的三元组视作正样本. 然后, 对于每个正样本, 本文通过破坏它的头实体或尾实体, 采样新的三元组作为负样本. 具体而言, 给定一个正样本 (h,r,t), 本文通过从训练实体集中随机采样 h' 或 t', 替代它的头实体或尾实体, 并且确保破坏后的三元组 (h',r,t) 或 (h,r,t') 不会出现在训练图 $\mathcal{G}_{\text{train}}$ 中. 除此之外, 根据 SimKGC[6], 对于尾实体预测 $(h,r,?)$ 来说, 基于文本的方法往往由于头实体 h 和查询的文本高度重叠, 而给头实体自身一个很高的分数, 但这却导致一个错误的预测. 因此, 本文使用一种自身负样本, 即在尾实体预测时使用头实体 h, 在头实体预测时使用尾实体 t, 作为难负例. 最终, 本文对于尾实体预测可以获得负样本集合 $\mathcal{D}^-=\{(h,r,t')\notin\mathcal{G}_{\text{train}}\cup(h,r,h)\notin\mathcal{G}_{\text{train}}\}$, 而对于头实体预测可以获得负样本集合 $\mathcal{D}^-=\{(h',r,t)\notin\mathcal{G}_{\text{train}}\cup(t,r,t)\notin\mathcal{G}_{\text{train}}\}$. 之后, 可以使用 InfoNCE[29]损失来训练模型. 具体计算方式如公式 (17), 其中, τ 表示温度超参数.

$$\mathcal{L}=-\log\frac{\mathrm{e}^{\phi(h,r,t)/\tau}}{\mathrm{e}^{\phi(h,r,t)/\tau}+\sum\limits_{(h',r',t')\in\mathcal{D}^-}\mathrm{e}^{\phi(h',r',t')/\tau}} \tag{17}$$

在模型推断时, 本文以尾实体预测 $(h,r,?)$ 为例. 头实体预测 $(?,r,t)$ 的过程也是相似的. 本文计算以每个实体 $t_i\in\mathcal{E}$ 为尾实体时, 所有三元组的分数, 并将得分最高的尾实体作为输出. 该过程表示如下, $\phi(\cdot,\cdot,\cdot)$ 表示打分解码器 RotatE:

$$\operatorname{argmax}\phi(h,r,t_i),\ t_i\in\mathcal{E} \tag{18}$$

5 实 验

本节进行充分的实验来评估所提出的框架. 首先, 介绍本文的实验设置; 然后, 介绍本文方法的整体效果及其与现有方法的对比; 最后, 对本文方法进行消融实验, 并对各个模块进行详细的分析.

5.1 实验设置

5.1.1 数据集

本文使用 3 个基于 Freebase[30]的数据集来评估本文的方法. 它们代表了 3 种不同的知识图谱补全的任务设定. FB15k-237[31]是 Freebase 的一个子集, 它包含了大量现实世界的事实知识. 它代表着直推式设定, 意味着一个测试三元组 (h',r,t') 中的所有实体在训练中都被模型见过, 即满足 $(h'\in\mathcal{E}_{\text{train}})\wedge(t'\in\mathcal{E}_{\text{train}})$. Daza 等人[4]构建了一个 FB15k-237 的归纳式版本, 称作 FB15k-237-Ind. 它代表着半归纳式或动态式设定, 意味着一个测试三元组 (h',r,t') 中至少有一个实体是在训练中未见过的, 即满足 $(h'\notin\mathcal{E}_{\text{train}})\vee(t'\notin\mathcal{E}_{\text{train}})$. Teru 等人[22]提出了另一个归纳式版本的 FB15k-237. 它代表着全归纳式或迁移式设定, 意味着测试三元组中的头实体和尾实体都没有在训练图中出现过, 即满足 $(h'\notin\mathcal{E}_{\text{train}})\wedge(t'\notin\mathcal{E}_{\text{train}})$. 由于 Teru 等人[22]提出了 4 个版本的归纳式数据集, 本文按照文献 [7] 选择了其中一个子集 (v4). 需要注意的是, 本文主要关注于归纳式的设定, 但是本文依然引入了直推式设定下的数据集来更全面地评估本文的方法. 数据集的统计数据可以见表 1.

© 中国科学院软件研究所 http://www.jos.org.cn

表 1 数据集统计信息

FB15k-237类型	关系数	训练集		验证集		测试集	
		实体数	三元组数	实体数	三元组数	实体数	三元组数
半归纳式	237	11633	215082	11633 + 1454	42164	11633 + 1454 + 1454	52870
全归纳式	219	4707	27203	0 + 3051	1416	0 + 3051	1424
直推式	237	14541	272115	14541 + 0	17535	14541 + 0	20466

注: “+”指在验证集或测试集中添加的新实体

5.1.2 基线模型

本文在不同的任务设定下比较了本文方法和一些强力的基线模型.

● 对于半归纳式设定, 基线模型包括了 StATIK[8] (当前最先进的方法, 是一个同时利用知识图谱文本和结构信息的混合模型)、StAR[5]、BLP[4]. 本文同样引入了 BLP[4]中提到的其他基线模型, 包括了 DKRL[19]和一个词袋模型 (bag-of-word, BOW). 该词袋模型通过对实体的所有单词表示做平均来得到实体的表示. 以上两个模型所用到的单词表示可以通过不同的方式来获取, 这里使用到了 GloVe[32] (GloVe-BOW、GloVe-DKRL) 和 BERT (BE-BOW、BE-DKRL).

● 对于全归纳式设定, 基线模型包括了 Bi-Link[7] (当前最先进的基于文本的方法, 是一个带有概率性的、基于规则的提示的对比学习框架) 以及它的基线模型, 包括了 DKRL[19]、KG-BERT[20]和 SimKGC[6].

● 对于直推式设定, 除了以上提到的当前最先进的基于文本的归纳式方法, 本文还引入了基于嵌入的方法作为基线模型, 包括了 TransE[3]、DistMult[16]、RotatE[11]和 TuckER[33]. 本文还引入了强有力的基于图神经网络的基线模型, 包括了 R-GCN[17]和 CompGCN[18].

此外, 本文还使用 ChatGPT (GPT-3.5-Turbo) 作为一个强力的基线模型, 它可用于所有的设定下. 本文按照现有工作[34]的设定以及提示来诱导 ChatGPT 进行链接预测任务.

5.1.3 实现细节

本文在半归纳式和直推式设定下使用 bert-mini[35,36] (https://huggingface.co/prajjwal1/bert-mini) 作为预训练语言模型, 而在全归纳式设定下使用 bert-small[35,36] (https://huggingface.co/prajjwal1/bert-small). 它们相比于先前工作[6−8]所使用的 BERT 拥有更小的参数量, 而使用更大的预训练语言模型也许可以进一步提升性能. 这些模型的权重都是通过 Transformers[37]库加载的. 由于算力的限制, 本文对于半归纳式、全归纳式和直推式设定, 分别将输入文本截断到最大长度为 48、56 和 36 个 token. 本文将公式 (6) 和公式 (12) 中注意力头的维度 d_k 和 BERT 保持一致. 对于结构编码模块, R-GCN 的层数设置为 2. 对于路径编码模块, 设置路径长度 K 为 5, 半归纳式和直推式设定下采样的路径个数 $2M$ 为 2×20, 全归纳式设定下采样的路径个数 $2M$ 为 2×80. 需要注意的是, 考虑到 M 增大会带来大量的空间开销, 本文在训练时为每个批次随机采样 $2M'=2\times2$ 条路径而在推断时使用全部的 $2M$ 条路径. 除此之外, 由于 BERT 是经过了预训练的而其他模块都是随机初始化的, 本文使用差分学习率进行模型训练, 并将其他模块和 BERT 之间的学习率比例标记为 λ. 本文对于半归纳式、全归纳式、直推式设定, 将学习率 α 分别设置为 5E–5、1E–5、5E–5, 将学习率比例 λ 分别设置为 10、100、20, 将批次大小分别设置为 64、256、256, 将训练轮数分别设置为 30、200、120. 而且, 本文将训练时负样本数量也分别设置为 4096、1024、1024. 本文还将温度超参数 τ 设置为 0.1, 间隔超参数 γ 设置为 9.0. 最后, 本文使用 AdamW[38]优化器, 它的权重衰减为 0.01, 以及使用线性学习率调度策略, 预热 1% 的训练步数, 然后进行线性衰减.

5.1.4 评估方式和指标

本文遵循先前的工作, 利用实体排序任务来评估本文的方法. 对于每个测试三元组 (h,r,t), 尾实体预测指给定头实体 h 和关系 r, 对所有的候选实体进行排序, 从而预测尾实体 t, 即预测 $(h,r,?)$. 头实体预测也是类似的, 即预测 $(?,r,t)$. 对于每个查询 $(h,r,?)$ 或 $(?,r,t)$, 本文将知识图谱中的所有实体都视作候选实体. 本文使用 4 个自动化的评估指标: MRR (mean reciprocal rank) 和 Hits@k ($k\in\{1,3,10\}$). MRR 计算所有测试三元组排序倒数的平均值. Hits@k 指正确的实体被排在前 k 个位置的比例. 本文在过滤设定[3]下计算所有自动化指标, 指在预测一个三元组中, 对候

© 中国科学院软件研究所 http://www.jos.org.cn

选实体排序时, 训练集、验证集和测试集中其他所有正确的三元组的分数都会被忽略掉, 从而防止自动化指标对于一些正确预测的误判. 而且, 所有汇报的指标都是取的头实体预测和尾实体预测, 两个预测方向下指标的平均. 由于让 ChatGPT 像本文方法一样对知识图谱中所有的实体做排序是不现实的, 本文按照现有工作[34]中的设定, 随机采样了 25 个测试样例, 并且仅汇报了 Hits@1 指标.

5.2 整体效果评估

表 2–表 4 汇报了在不同设定下, 所有方法的整体效果, 最优结果加粗显示, 同时表 4 中对次优结果加下划线显示. 可以得出结论: (1) 如表 2 和表 3 所示, 本文方法在半归纳式和全归纳式的设定下, 都显著超越了现有的最先进方法, 证明了本文方法的有效性和归纳性. (2) 即使本文方法中仅使用了一个小版本的 BERT, 它依然超越了使用更大版本 BERT 的基线模型, 比如 BLP、StATIK 和 SimKGC. 这突出了多视角编码器能够综合利用知识图谱中文本和结构信息的优势. (3) 如表 4 所示, 本文方法在直推式设定下也取得了有竞争力的表现, 表明了本文框架的强大性能并不仅局限于归纳式的设定. 其中的原因可能是本文框架在邻居信息之外, 进一步捕捉到了实体间的语义相关性和多跳关联性, 促进了对于实体语义的理解. (4) 本文方法在所有的设定下都超越了 ChatGPT, 其中的原因可能是 ChatGPT 更倾向于避免给出一个直接的答案. 它往往会给出一个模糊不清的、有歧义的答案, 或者是请求获得更多的上下文信息.

表 2 半归纳式设定下的整体性能评估 (%)

模型	MRR	Hits@1	Hits@3	Hits@10
GloVe-BOW	17.2	9.9	18.8	31.6
BE-BOW	17.3	10.3	18.4	31.6
GloVe-DKRL	11.2	6.2	11.1	21.1
BE-DKRL	14.4	8.4	15.1	26.3
BLP-TransE	19.5	11.3	21.3	36.3
BLP-DistMult	14.6	7.6	15.6	28.6
BLP-ComplEx	14.8	8.1	15.4	28.3
BLP-SimplE	14.4	7.7	15.2	27.4
StAR	16.3	9.2	17.6	30.9
StATIK	22.4	14.3	24.8	38.1
ChatGPT	—	16.0	—	—
本方法	**23.9**	**16.1**	**25.7**	**40.0**

表 3 全归纳式设定下的整体性能评估 (%)

模型	MRR	Hits@1	Hits@3	Hits@10
DKRL	3.9	—	—	7.3
KG-BERT	16.1	—	—	32.1
SimKGC	10.0	—	—	21.5
Bi-Link	18.3	—	—	33.8
ChatGPT	—	12.0	—	—
本方法	**23.2**	**12.9**	**25.5**	**46.2**

表 4 直推式设定下的整体性能评估 (%)

模型	MRR	Hits@1	Hits@3	Hits@10
TransE	27.9	19.8	37.6	44.1
DistMult	28.1	19.9	30.1	44.6
RotatE	33.8	24.1	37.5	53.3
TuckER	35.8	26.6	**39.4**	**54.4**
R-GCN	24.8	15.3	25.8	41.4
CompGCN	35.5	26.4	39.0	53.5
KG-BERT	24.2	14.8	26.9	43.0
StAR	29.6	20.5	32.2	48.2
SimKGC	33.6	24.9	36.2	51.1
Bi-Link	34.3	26.5	38.6	53.8
ChatGPT	—	24.0	—	—
本方法	**36.1**	**27.2**	<u>39.2</u>	<u>54.1</u>

5.3 消融实验

为了进一步探究本文框架中各种模块对模型性能的影响, 本文遵循文献 [8] 的方式, 在半归纳式设定下进行了消融实验, 分别移除差异化机制和每个编码模块, 在表 5 中汇报了消融实验的结果.

© 中国科学院软件研究所 http://www.jos.org.cn

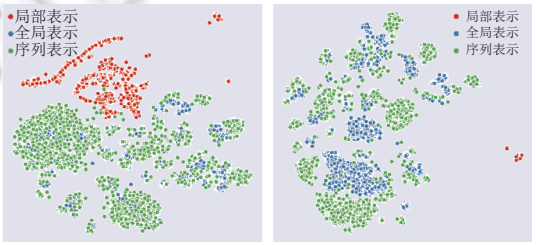
从表 5 中可见: 首先, 移除差异化机制造成了严重的模型性能下降, 证明了它在避免过度注意问题以及鼓励各种视角学习有区别的实体表示时的有效性; 其次, 移除任何的编码模块都导致了模型性能的下降, 表明从各个视角学习到的实体表示都是必要的.

5.4 差异化机制的可视化

为了定性地展示差异化机制的有效性, 本文在有和没有差异化机制两种情况下, 对不同视角学习得到的实体表示进行可视化. 具体来说, 本文在全归纳式设定下, 使用 t-SNE[39]对测试集中未见过的实体的各种视角的表示进行可视化. 可视化的结果如图 3 所示. 在图 3(a) 中, 局部表示和其他视角的表示相距很近而且甚至存在一定重叠. 这表明在没有差异化机制的情况下, 局部视角和其他视角的表示区分度较低. 然而, 使用了差异化机制后, 在图 3(b) 中, 局部表示和其他视角的表示就变得相距较远, 呈现出更明显的区分度. 这表明差异化机制有效地在全局和序列视角中鼓励学习和局部视角不同的表示, 保证了不同视角下信息聚合的多样性和互补性. 另外一个有趣的现象是在图 3(b) 中, 全局表示和序列表示之间的距离也有所增加, 其中的原因可能是在差异化机制中移除掉的局部信息也包含了全局和序列视角之间重叠的公共信息. 最终, 移除这样的冗余信息也会进一步促进全局和序列视角之间信息聚合的多样性和互补性.

表 5 半归纳式设定下的消融实验结果 (%)

模型	MRR	Hits@1	Hits@3	Hits@10
本方法	**23.9**	**16.1**	**25.7**	**40.0**
移除: 差异化机制	23.1	15.4	24.7	38.5
移除: 结构编码模块	23.2	15.5	24.7	38.8
移除: 全局编码模块	23.6	16.0	25.2	39.2
移除: 路径编码模块	23.3	15.6	24.8	39.4



(a) 实体表示 (无差异化机制) (b) 实体表示 (有差异化机制)

图 3 全归纳式设定下, 对训练中未见过实体的表示的可视化

5.5 全局注意力机制的可视化

为了更直观地展现第 4.1.3 节中的全局注意力机制是如何工作的, 本文对实体之间计算出的注意力权重进行可视化, 以此来观察每个实体会更注意哪些其他实体. 具体来说, 本文在半归纳式设定下, 对于每个实体, 选择和它计算得到的注意力权重最高的 10 个实体, 并且可视化它们的注意力权重. 后文图 4 展现了一些示例. 由此, 本文可以得出结论: 全局注意力机制能够成功捕获实体之间的语义相关性, 并且会给概念上相关的实体赋予更高的注意力权重. 例如, 对于“Kobe Bryant”来说, 实体“Allen Iverson”和“Dwyane Wade”都是著名的篮球运动员且像“Kobe Bryant”一样司职得分后卫, 且它们在注意力计算中获得了更高的权重. 类似地, 对于“GNU/Linux”来说, “OS X”和“Microsoft Windows”都是著名的操作系统, 并且它们和“GNU/Linux”的概念相关性也都反映在了它们更高的注意力权重中. 这些实体分别和“Kobe Bryant”以及“GNU/Linux”属于同样的概念, 并且和它们紧密相关. 最终, 它们的语义信息将会有助于对“Kobe Bryant”和“GNU/Linux”有一个更全面综合的理解, 而这也证明了全局注意力机制的有效性.

5.6 路径采样策略的分析

为了分析第 4.1.4 节中的路径采样策略对模型性能的影响, 本文探究采样路径的数量 M 和每条边的采样概率 $p_{u,v}$ 是如何影响模型性能的, 并且在图 5 中汇报了实验结果. 可以发现, 随着 M 的增加, 模型性能会提升; 而当 M 超过 80 后, 模型性能开始下降. 其中的原因可能是采样更多的路径将使得一个实体看见更多远距离相连的实体, 并和这些不同领域的实体发生交互, 最终获取到更广泛的上下文信息并通过它们丰富自身的表示. 然而, 随着 M 变得过大, 越来越多的无关实体会被引入并成为噪声. 这些实体虽然在图中存在, 但与目标实体的关联较弱, 使得聚合了这些无关实体信息的目标实体表示质量下降, 最终导致模型性能下降. 对于采样概率, 本文的采样概率要比 $p_{u,v} \propto \frac{1}{deg(u)} + \frac{1}{deg(v)}$ 更好. 该方式的目标是将每个实体尽可能和远距离相连的长尾实体而不是受欢迎的实体关

© 中国科学院软件研究所 http://www.jos.org.cn

联上. 其中的原因可能是实体会从几跳之内的受欢迎实体的丰富的信息中得到更多的信息增益, 而不是长尾实体. 然而, 将一个实体和远距离相连的长尾实体关联上能够使得它聚合到长尾实体的稀疏信息, 而这种稀疏信息很难通过局部或全局视角聚合到. 因此, 该策略又会优于随机采样策略.

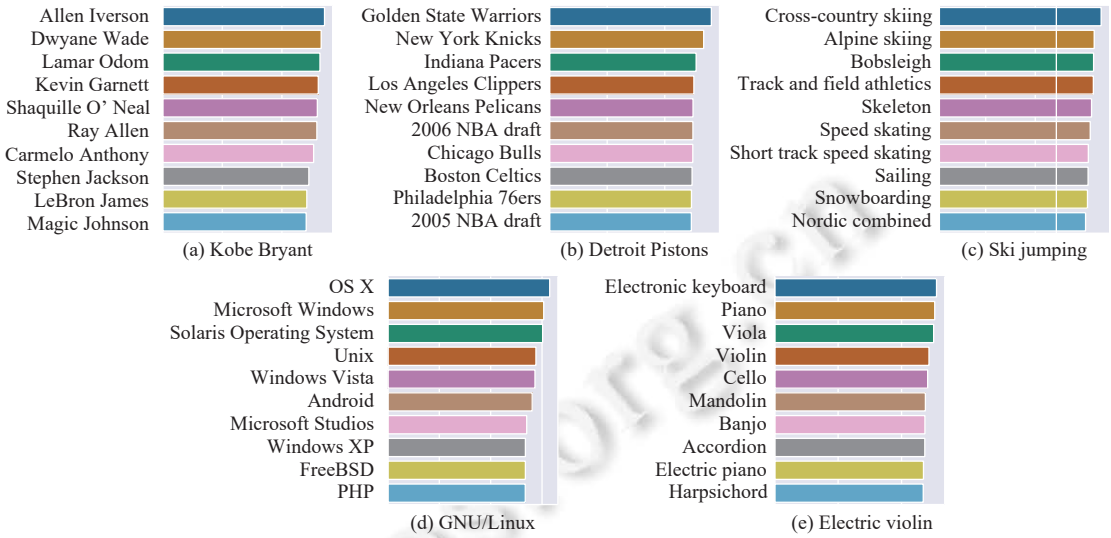
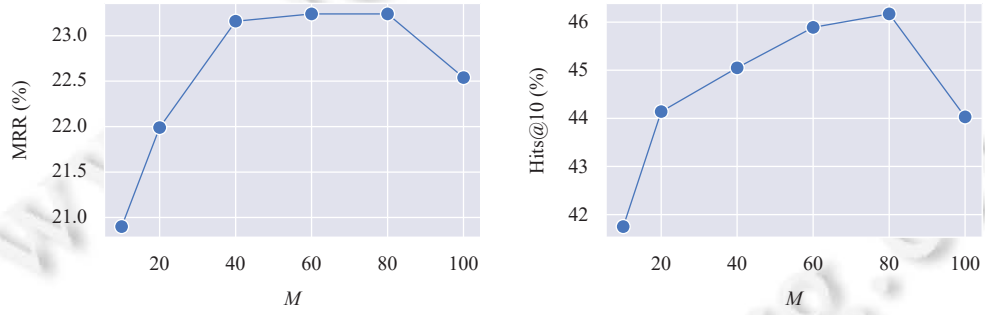
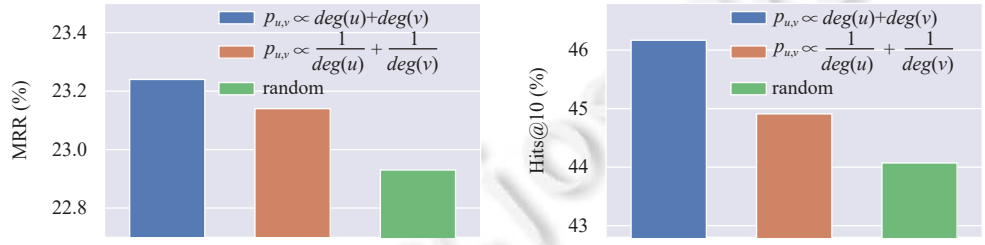


图 4 半归纳式设定下, 对全局注意力机制中注意力权重最高的实体的可视化



(a) 采样路径数 M 对模型性能的影响



(b) 边的采样概率 $p_{u,v}$ 对模型性能的影响

图 5 全归纳式设定下, 采样路径数和边的采样概率对模型性能的影响

5.7 推理效率的分析

真实世界中, 随着知识图谱的扩充, 知识图谱补全也需要考虑到实体和三元组数的增长. 因此, 推理效率是决定知识图谱补全模型是否能够落地到真实应用中的一个重要因素. 为了展现本文框架的推理效率, 本文与其他最先进的归纳式方法对每个查询的推理时间进行了比较, 并在表 6 中汇报了实验结果. 从表 6 中可以得出结论: 本文

© 中国科学院软件研究所 http://www.jos.org.cn

方法与一些交互式编码器架构的基线模型相比, 例如 KG-BERT, 推理速度是显著更快的. 更重要的是, 本文方法也超越了其他被公认为高效的、表示式编码器架构的基线模型, 例如 BLP 和 StATIK. 这些都进一步证明了本文框架的推理高效性.

表 6 半归纳式设定下, 归纳式模型对每个查询的推理时间 (ms)

模型	KG-BERT	StAR	BLP	StATIK	本文方法
推理时间	87 000	321	21	4	**2.8**

6 总 结

本文关注于归纳式知识图谱补全, 提出了一个带有差异化机制的多视角知识图谱补全框架, 从知识图谱的局部、全局和序列视角来学习互补的实体表示. 本文框架包含了一个带有差异化机制的多视角编码器和一个打分解码器. 多视角编码器使用了 1 个预训练语言模型和 3 个编码模块, 通过差异化机制来避免多视角信息聚合的冗余, 从而从多视角学习到互补的实体表示. 打分解码器使用了一个基于转移的模型, 基于多视角习得的实体表示给三元组打分. 实验结果表明, 本文方法在归纳式设定下超越了各种当前最先进的模型, 并且在直推式设定下也保持着强有力的性能.

未来, 如何将大规模语言模型 (large language model, LLM) 与知识图谱构建和补全任务深度融合是值得继续探索的研究方向. 一方面, 随着大语言模型具备更强的上下文理解[40]和更长的输入长度, 它能够更有效地理解和处理结构化数据, 从而提升对知识图谱结构的理解能力. 通过利用更有效的线性化策略以及工具学习[41], 可以使得大模型更精准地访问并理解结构数据, 进一步促进知识图谱的构建与补全. 另一方面, 知识图谱和大模型作为两种不同的知识表示形式, 未来可以结合其各自优势, 既利用知识图谱的结构化特性, 又发挥大模型在推理能力[42]上的优势, 在完成知识图谱构建和补全中复杂的推理任务时也兼具可解释性[43].

References:

[1] Ji SX, Pan SR, Cambria E, Marttinen P, Yu PS. A survey on knowledge graphs: Representation, acquisition, and applications. IEEE Trans. on Neural Networks and Learning Systems, 2022, 33(2): 494–514. [doi: 10.1109/TNNLS.2021.3070843]

[2] Liu Q, Li Y, Duan H, Liu Y, Qin ZG. Knowledge graph construction techniques. Journal of Computer Research and Development, 2016, 53(3): 582–600 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2016.20148228]

[3] Bordes A, Usunier N, Garcia-Durán A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In: Proc. of the 27th Int'l Conf. on Neural Information Processing Systems. Lake Tahoe: Curran Associates Inc., 2013. 2787–2795.

[4] Daza D, Cochez M, Groth P. Inductive entity representations from text via link prediction. In: Proc. of the 2021 Web Conf. Ljubljana: ACM, 2021. 798–808. [doi: 10.1145/3442381.3450141]

[5] Wang B, Shen T, Long GD, Zhou TY, Wang Y, Chang Y. Structure-augmented text representation learning for efficient knowledge graph completion. In: Proc. of the 2021 Web Conf. Ljubljana: ACM, 2021. 1737–1748. [doi: 10.1145/3442381.3450043]

[6] Wang L, Zhao W, Wei ZY, Liu JM. SimKGC: Simple contrastive knowledge graph completion with pre-trained language models. In: Proc. of the 60th Annual Meeting of the Association for Computational Linguistics. Dublin: ACL, 2022. 4281–4294. [doi: 10.18653/v1/2022.acl-long.295]

[7] Peng BH, Liang SH, Islam M. Bi-Link: Bridging inductive link predictions from text via contrastive learning of Transformers and prompts. arXiv:2210.14463, 2022.

[8] Markowitz E, Balasubramanian K, Mirtaheri M, Annavaram M, Galstyan A, ver Steeg G. StATIK: Structure and text for inductive knowledge graph completion. In: Proc. of the 2022 Findings of the Association for Computational Linguistics: NAACL. Seattle: ACL, 2022. 604–615. [doi: 10.18653/v1/2022.findings-naacl.46]

[9] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. arXiv:1609.02907, 2016.

[10] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

[11] Sun ZQ, Deng ZH, Nie JY, Tang J. RotatE: Knowledge graph embedding by relational rotation in complex space. arXiv:1902.10197, 2019.

© 中国科学院软件研究所 http://www.jos.org.cn

[12] Zhang TC, Tian X, Sun XH, Yu MH, Sun YH, Yu G. Overview on knowledge graph embedding technology research. Ruan Jian Xue Bao/Journal of Software, 2023, 34(1): 277–311 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6429.htm [doi: 10.13328/j.cnki.jos.006429]

[13] Wang Z, Zhang JW, Feng JL, Chen Z. Knowledge graph embedding by translating on hyperplanes. In: Proc. of the 28th AAAI Conf. on Artificial Intelligence. Québec: AAAI Press, 2014. 1112–1119.

[14] Trouillon T, Welbl J, Riedel S, Gaussier É, Bouchard G. Complex embeddings for simple link prediction. In: Proc. of the 33rd Int'l Conf. on Machine Learning. New York: JMLR.org, 2016. 2071–2080.

[15] Nickel M, Tresp V, Kriegel HP. A three-way model for collective learning on multi-relational data. In: Proc. of the 28th Int'l Conf. on Machine Learning. Bellevue: Omnipress, 2011. 809–816.

[16] Yang BS, Yih WT, He XD, Gao JF, Deng L. Embedding entities and relations for learning and inference in knowledge bases. arXiv:1412.6575, 2014.

[17] Schlichtkrull M, Kipf TN, Bloem P, van den Berg R, Titov I, Welling M. Modeling relational data with graph convolutional networks. In: Proc. of the 15th Int'l Conf. on the Semantic Web. Heraklion: Springer, 2018. 593–607. [doi: 10.1007/978-3-319-93417-4_38]

[18] Vashishth S, Sanyal S, Nitin V, Talukdar P. Composition-based multi-relational graph convolutional networks. arXiv:1911.03082, 2020.

[19] Xie RB, Liu ZY, Jia J, Luan HB, Sun MS. Representation learning of knowledge graphs with entity descriptions. In: Proc. of the 30th AAAI Conf. on Artificial Intelligence. Phoenix: AAAI Press, 2016. 2659–2665. [doi: 10.1609/aaai.v30i1.10329]

[20] Yao L, Mao CS, Luo Y. KG-BERT: BERT for knowledge graph completion. arXiv:1909.03193, 2019.

[21] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional Transformers for language understanding. In: Proc. of the 2019 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 4171–4186. [doi: 10.18653/v1/N19-1423]

[22] Teru KK, Denis EG, Hamilton WL. Inductive relation prediction by subgraph reasoning. In: Proc. of the 37th Int'l Conf. on Machine Learning. Vienna: PMLR, 2020. 9448–9457.

[23] Zha HW, Chen ZY, Yan XF. Inductive relation prediction by BERT. In: Proc. of the 36th AAAI Conf. on Artificial Intelligence. Palo Alto: AAAI Press, 2022. 5923–5931. [doi: 10.1609/aaai.v36i5.20537]

[24] Geng YX, Chen JY, Pan JZ, Chen MY, Jiang S, Zhang W. Relational message passing for fully inductive knowledge graph completion. In: Proc. of the 39th IEEE Int'l Conf. on Data Engineering. Anaheim: IEEE, 2023. 1221–1233. [doi: 10.1109/ICDE55515.2023.00098]

[25] He KM, Zhang XY, Ren SQ, Sun J. Deep residual learning for image recognition. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778. [doi: 10.1109/CVPR.2016.90]

[26] Ba JL, Kiros JR, Hinton GE. Layer normalization. arXiv:1607.06450, 2016.

[27] Lovász L. Random walks on graphs: A survey. 1993. https://cs.yale.edu/publications/techreports/tr1029.pdf

[28] Su JL, Ahmed M, Lu Y, Pan SF, Bo W, Liu YF. RoFormer: Enhanced Transformer with rotary position embedding. Neurocomputing, 2024, 568: 127063. [doi: 10.1016/j.neucom.2023.127063]

[29] Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: Proc. of the 37th Int'l Conf. on Machine Learning. Vienna: PMLR, 2020. 1597–1607.

[30] Bollacker K, Evans C, Paritosh P, Sturge T, Taylor J. Freebase: A collaboratively created graph database for structuring human knowledge. In: Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. Vancouver: ACM, 2008. 1247–1250. [doi: 10.1145/1376616.1376746]

[31] Toutanova K, Chen DQ, Pantel P, Poon H, Choudhury P, Gamon M. Representing text for joint embedding of text and knowledge bases. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: Association for Computational Linguistics, 2015. 1499–1509. [doi: 10.18653/v1/D15-1174]

[32] Pennington J, Socher R, Manning C. GloVe: Global vectors for word representation. In: Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing. Doha: ACL, 2014. 1532–1543. [doi: 10.3115/v1/D14-1162]

[33] Balažević I, Allen C, Hospedales T. TuckER: Tensor factorization for knowledge graph completion. In: Proc. of the 2019 Conf. on Empirical Methods in Natural Language Processing and the 9th Int'l Joint Conf. on Natural Language Processing. Hong Kong: ACL, 2019. 5185–5194. [doi: 10.18653/v1/D19-1522]

[34] Zhu YQ, Wang XH, Chen J, Qiao SF, Ou YX, Yao YZ, Deng SM, Chen HJ, Zhang NY. LLMs for knowledge graph construction and reasoning: Recent capabilities and future opportunities. arXiv:2305.13168, 2023.

[35] Bhargava P, Drozd A, Rogers A. Generalization in NLI: Ways (not) to go beyond simple heuristics. arXiv:2110.01518, 2021.

[36] Turc I, Chang MW, Lee K, Toutanova K. Well-read students learn better: On the importance of pre-training compact models. arXiv:1908.08962, 2019.

© 中国科学院软件研究所 http://www.jos.org.cn

[37] Wolf T, Debut L, Sanh V, *et al*. Transformers: State-of-the-art natural language processing. In: Proc. of the 2020 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations. ACL, 2020. 38–45. [doi: 10.18653/v1/2020.emnlp-demos.6]

[38] Loshchilov I, Hutter F. Decoupled weight decay regularization. arXiv:1711.05101, 2017.

[39] van der Maaten L, Hinton G. Visualizing data using t-SNE. Journal of Machine Learning Research, 2008, 9(86): 2579–2605.

[40] Dong QX, Li L, Dai DM, Zheng C, Ma JY, Li R, Xia HM, Xu JJ, Wu ZY, Chang BB, Sun X, Li L, Sui ZF. A survey on in-context learning. In: Proc. of the 2024 Conf. on Empirical Methods in Natural Language Processing. Miami: ACL, 2024. 1107–1128. [doi: 10.18653/v1/2024.emnlp-main.64]

[41] Yang S, Nachum O, Du YL, Wei J, Abbeel P, Schuurmans D. Foundation models for decision making: Problems, methods, and opportunities. arXiv:2303.04129, 2023.

[42] Qiao SF, Ou YX, Zhang NY, Chen X, Yao YZ, Deng SM, Tan CQ, Huang F, Chen HJ. Reasoning with language model prompting: A Survey. In: Proc. of the 61st Annual Meeting of the Association for Computational Linguistics. Toronto: ACL, 2023. 5368–5393. [doi: 10.18653/v1/2023.acl-long.294]

[43] Ma A, Yu YH, Yang SL, Shi C, Li J, Cai XX. Survey of knowledge graph based on reinforcement learning. Journal of Computer Research and Development, 2022, 59(8): 1694–1722 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.20211264]

附中文参考文献:

[2] 刘峤, 李杨, 段宏, 刘瑶, 秦志光. 知识图谱构建技术综述. 计算机研究与发展, 2016, 53(3): 582–600. [doi: 10.7544/issn1000-1239.2016.20148228]

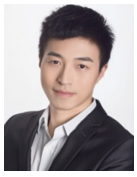
[12] 张天成, 田雪, 孙相会, 于明鹤, 孙艳红, 于戈. 知识图谱嵌入技术研究综述. 软件学报, 2023, 34(1): 277–311. http://www.jos.org.cn/1000-9825/6429.htm [doi: 10.13328/j.cnki.jos.006429]

[43] 马昂, 于艳华, 杨胜利, 石川, 李劼, 蔡修秀. 基于强化学习的知识图谱综述. 计算机研究与发展, 2022, 59(8): 1694–1722. [doi: 10.7544/issn1000-1239.20211264]

童翰文(1999－), 男, 硕士, 主要研究领域为知识图谱, 大模型, 自然语言处理.

肖仰华(1980－), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为大数据管理与挖掘, 图数据库, 知识图谱.

钱羽希(1995－), 男, 硕士, 主要研究领域为自然语言处理, 知识图谱, 智能体.



韦峰(1989－), 男, 硕士, 主要研究领域为自然语言处理, 知识图谱, 智能体.

刘井平(1991－), 男, 博士, 副教授, 主要研究领域为知识图谱, 大模型, 自然语言处理.

郝正鸿(1987－), 女, 硕士, 主要研究领域为自然语言处理, 知识图谱, 智能对话.

梁祖杰(1998－), 男, 硕士, 主要研究领域为自然语言处理, 知识图谱, 智能体.

韩冰(1985－), 女, 硕士, 主要研究领域为自然语言理解, 搜索推荐, 知识工程.

© 中国科学院软件研究所 http://www.jos.org.cn