

## 融合本体和实例的知识图谱嵌入模型

秦晶, 李贯峰, 陈昱胤, 肖毓航

### 引用本文

秦晶, 李贯峰, 陈昱胤, 肖毓航. [融合本体和实例的知识图谱嵌入模型](#)[J]. 计算机科学, 2026, 53(3): 331-340.

QIN Jing, LI Guanfeng, CHEN Yuyin, XIAO Yuhang. [Embedding Model of Knowledge Graph via Jointly Modeling Ontology and Instances](#) [J]. Computer Science, 2026, 53(3): 331-340.

---

## 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)**

### [基于多任务学习的眼科视频特征融合与多维画像](#)

Multi-task Learning-based Ophthalmic Video Feature Fusion and Multi-dimensional Profiling  
计算机科学, 2026, 53(3): 383-391. <https://doi.org/10.11896/jsjcx.260200058>

### [基于背景结构感知的小样本知识图谱补全](#)

Background Structure-aware Few-shot Knowledge Graph Completion  
计算机科学, 2026, 53(2): 331-341. <https://doi.org/10.11896/jsjcx.250100107>

### [知识图谱的复杂逻辑查询方法研究综述](#)

Survey on Complex Logical Query Methods in Knowledge Graphs  
计算机科学, 2026, 53(2): 273-288. <https://doi.org/10.11896/jsjcx.250400033>

### [个性化教育资源推荐综述](#)

Review of Personalized Educational Resource Recommendations  
计算机科学, 2026, 53(2): 1-15. <https://doi.org/10.11896/jsjcx.250700184>

### [基于元学习的跨语言知识图谱实体对齐框架](#)

Cross-language Knowledge Graph Entity Alignment Based on Meta-learning  
计算机科学, 2026, 53(1): 271-277. <https://doi.org/10.11896/jsjcx.241100069>

# 融合本体和实例的知识图谱嵌入模型

秦晶 李贯峰 陈昱胤 肖毓航

宁夏大学信息工程学院 银川 750021

宁夏“东数西算”人工智能与信息安全重点实验室 银川 750021

(2720753621@qq.com)

**摘要** 知识图谱嵌入通过将实体和关系投影到连续的低维向量空间中,为机器学习模型提供更强大的知识表示输入,从而支撑更多的知识图谱应用场景。近年来,研究人员试图利用知识图谱中的本体和实例之间的潜在语义信息来增强知识图谱的嵌入。然而,它们未能有效融合概念的层次结构和实例的特定信息,并且忽略了 isA 关系之间的传递性,导致模型在处理知识图谱中的长尾实体时的性能和泛化能力受限。为了弥补上述不足,提出了一个融合了本体和实例的知识图谱嵌入模型 JMOI (Representation Learning of Knowledge Graph via Jointly Modeling Ontology and Instances)。该模型通过引入自注意力机制,能够捕捉到概念和实例之间复杂的语义关系,并增加了一个可学习的参数来调整概念嵌入的邻域范围,以区分不同概念的层次信息,从而对 isA 关系的传递性进行建模。在 YAGO26K-906 和 DB111K-174 数据集上的实验结果表明,与现有技术相比, JMOI 在大多数情况下都达到了最佳性能,与次优模型相比,在链接预测 Hits@1 指标上最大提升了 6.5%,在三元组分类中召回率指标最大提升了 6.9%。

**关键词:** 知识图谱;知识图谱嵌入;概念和实例;链接预测;三元组分类

**中图分类号** TP391.1

## Embedding Model of Knowledge Graph via Jointly Modeling Ontology and Instances

QIN Jing, LI Guanfeng, CHEN Yuyin and XIAO Yuhang

School of Information Engineering, Ningxia University, Yinchuan 750021, China

Ningxia Key Laboratory of Artificial Intelligence and Information Security for Channeling Computing Resources from the East to the West, Yinchuan 750021, China

**Abstract** Knowledge graph embedding provides a more powerful knowledge representation input to machine learning models by projecting entities and relationships into a continuous low-dimensional vector space, thereby supporting more knowledge graph application scenarios. In recent years, researchers have tried to use the potential semantic information between ontology and instance in knowledge graph to enhance the embedding of knowledge graph. However, they fail to effectively integrate the hierarchical structure of concepts and the specific information of instances, and ignore the transitivity between isA relationships, resulting in limited performance and generalization ability of the models when dealing with long-tail entities in the knowledge graph. In order to solve the above shortcomings, this paper proposes a knowledge graph embedding model (Representation Learning of Knowledge Graph via Jointly Modeling Ontology and Instances, JMOI), which integrates ontology and instance. By introducing self-attention mechanism, this model captures the complex semantic relationship between concepts and instances, and adds a learnable parameter to adjust the neighborhood range of concept embedding, so as to distinguish the hierarchical information of different concepts. The transitivity of isA relationship is modeled. Experimental results on the YAGO26K-906 and DB111K-174 datasets show that JMOI achieves the best performance in most cases compared with the prior art, with a maximum improvement of 6.5% in the link prediction Hits@1 and 6.9% in the Recall in triple classification compared with the suboptimal model.

**Keywords** Knowledge graph, Knowledge graph embedding, Concepts and instances, Link prediction, Triple classification

到稿日期:2025-02-25 返修日期:2025-04-28

基金项目:宁夏全职引进高层次人才科研启动项目(2023BSB03066);宁夏自然科学基金(2024AAC03098);国家自然科学基金(62066038);宁夏大学 2025 研究生创新项目(CXXM2025-038)

This work was supported by the Full-time Introduction of High-level Talent Research Start-up Project Foundation of Ningxia(2023BSB03066), Natural Science Foundation of Ningxia Province(2024AAC03098), National Natural Science Foundation of China (62066038) and 2025 Graduate Innovation Project of Ningxia University(CXXM2025-038).

通信作者:李贯峰(ligf@nxu.edu.cn)

## 1 引言

随着人工智能技术的快速发展,知识图谱(Knowledge Graphs, KGs)作为一种结构化的知识表示方法<sup>[1]</sup>,已成为连接感知智能与认知智能的桥梁。知识图谱以三元组( $h, r, t$ )的形式(即头实体、关系和尾实体),将现实世界中的实体及其相互关系进行编码,为信息检索<sup>[2]</sup>、个性化推荐<sup>[3]</sup>和智能问答<sup>[4]</sup>等应用提供了基础。然而,三元组的结构化表示并不利于计算,因此知识图谱嵌入得到了广泛的研究。知识图谱嵌入(Knowledge Graph Embedding, KGE)技术旨在将知识图谱中的实体和关系映射到低维向量空间中<sup>[5]</sup>,以保留其语义信息和拓扑结构,从而实现更高效的知识图谱补全和关系预测。

然而,知识图谱存在数据稀疏和不平衡问题<sup>[6]</sup>,导致学习的知识嵌入性能不佳。知识图谱中的实体是类型化的,类型通常以层次结构组织,每个实体都可以与一个或多个类型相关联<sup>[7]</sup>。实体不仅包括具体的实例,如“故宫”,还包括抽象的概念,如“建筑”。基于概念和实例的知识图谱嵌入模型利用图中概念和实例之间的潜在语义关系来学习其表示,它可以在不增加训练样本额外要求的情况下,补充实体和关系的语义信息。概念可以为属于它的实例提供基本的类型信息,实例可以为相应的概念提供详细信息,从而缓解数据稀疏性导致的学习不足问题。

知识图谱嵌入领域的研究逐渐关注双视图知识图谱(实例视图和本体视图)的联合建模问题。基于概念和实例的知识图谱嵌入的代表性研究有 JECI++<sup>[8]</sup>, TransC<sup>[9]</sup> 和 JOIE<sup>[10]</sup> 等。最新的研究有 DGS<sup>[11]</sup>, CIST<sup>[12]</sup>, Concept2-Box<sup>[13]</sup>, TransO<sup>[14]</sup> 和 EIKE<sup>[15]</sup> 等。

JECI++模型通过基于实例邻域信息与实例所属概念的交互信息构建预测函数来预测实例,并通过最小化预测值与实际实例嵌入之间的差距来迭代学习实例和概念的嵌入。然而,由于知识图谱的稀疏性,并不是所有的实例都有足够的邻居信息用于训练,因此模型的泛化性不够。TransC 尝试在同一空间内对概念和实例进行建模,将每个概念建模为一个球体,将每个实例建模为同一语义空间中的一个点。但是,球体是一个高度对称的空间几何体,因此无法捕捉概念的复杂语义。JOIE 通过跨视图关联模型来学习实例和概念之间的关系,并利用视图内模型来学习实例和概念的内部结构,但它无法处理实例属于多个概念的情况,也不能对 subClassOf 关系的传递性进行建模。

为了解决传统单几何空间模型无法同时捕捉实例的循环关系和概念的层次关系的问题,DGS 将实例视图(循环结构)嵌入到球面空间,将本体视图(层次结构)嵌入到双曲空间,并设计桥接空间处理两者的交集节点。封闭球面空间算子直接在球面空间中进行旋转操作,避免投影到切空间的近似误差。但是,双曲空间和球面空间的联合优化复杂度较高,导致训练效率受限。CIST 将概念和实例分别嵌入到不同维度的空间,并通过非线性映射函数连接两者。但是,模型对复杂关系,如重叠和包含的建模能力有限,而且实例空间仅使用传统向量嵌入,难以捕捉实例间的复杂关联。传统盒嵌入方法未区分

概念与实例的几何特性,导致语义桥接能力不足。在 Concept2Box 模型中,概念使用盒嵌入(Box Embedding)捕捉层次和重叠关系,实例使用向量嵌入,并通过向量到盒的距离度量桥接两者。利用 BERT 编码概念和关系的文本信息,生成动态的盒变换参数。但是,盒嵌入需要处理多维边界参数,计算复杂度较高。TransO 在单一空间中通过投影矩阵结合本体约束,但不区分概念的几何表示。EIKE 将本体表示为外延空间和内延空间,外延空间使用几何方法表示概念,使用点向量表示实例;内延空间使用预训练语言模型编码概念的文本属性,捕捉语义关联。

现有模型待解决的问题主要有以下 3 点。

(1)概念与实例的语义混淆:大多数现有的 KGE 模型未能充分区分概念与实例,而将它们全部都视为实体,忽略了它们在语义和结构上的差异,导致模型无法准确捕捉实体的层次化语义信息。

(2)长尾实体性能受限:数据稀疏性导致模型对长尾实体的嵌入质量不足,泛化能力较差。

(3)跨视图关系建模能力有限:现有方法不支持复杂的跨视图关系建模,如重叠性、传递性、对称性等,限制了模型在处理复杂知识图谱结构时的性能和泛化能力。

为了解决上述问题,本文提出了一个融合本体和实例的知识图谱嵌入模型 JMOI,将嵌入空间划分为概念嵌入空间和实例嵌入空间,分别对概念和实例的嵌入进行建模。将知识图谱中的三元组分为实例三元组、子类三元组和关系三元组(包括实例通过关系连接的三元组和概念通过关系连接的三元组,不包括子类三元组)3 种类型,并分别建模。针对实例三元组,将实例和概念映射到同一个向量空间,引入注意力机制自适应地调整实例与概念的映射关系,解决多归属实例的嵌入冲突。对于子类三元组,利用关系子类的传递性进行建模。对于关系三元组,选择经典的嵌入模型 TransE 来学习嵌入。通过融合层次化类别信息和注意力机制,模型能够更全面地捕捉知识图谱中的复杂结构和语义信息,从而提高知识图谱嵌入的性能。

本文的主要贡献如下:

(1)将嵌入空间划分为概念嵌入空间和实例嵌入空间,分别对概念和实例进行建模,充分考虑了它们在语义和结构上的差异,能够更准确地捕捉实体的层次化语义信息;

(2)利用层次化类别信息增强实体的表示,能够有效缓解数据稀疏性问题,使得模型在处理长尾实体时具有更好的性能和泛化能力;

(3)通过引入自注意力机制,模型能够自适应地调整实例与概念之间的映射关系,准确地捕捉实例与其所属概念之间的复杂语义关系,提高嵌入质量和模型性能;

(4)以 TransE<sup>[17]</sup>, DistMult<sup>[18]</sup>, HolE<sup>[19]</sup>, TransC, JOIE, DGS 和 Concept2Box 为基准, JMOI 模型在链接预测和三元组分类任务上均取得了优越性能,验证了其有效性和实用性。

## 2 相关工作

### 2.1 基于三元组结构的知识图谱嵌入模型

目前的知识图谱嵌入模型主要分为 3 类。

(1)基于平移距离的模型。传统的平移距离嵌入模型有 TransE, TransH<sup>[20]</sup>, TransR<sup>[21]</sup> 和 TransD<sup>[22]</sup> 等, Trans 系列的改进模型有 STransH<sup>[23]</sup> 和 MvTransE<sup>[24]</sup> 等。STransH 采用单层神经网络和超平面模型来提升翻译模型对复杂关系的处理能力。多视图翻译嵌入模型 MvTransE 能够结合全局视图和局部视图来完成事实三元组的嵌入表示。这类模型算法简单, 计算效率高, 可扩展性强, 但是对于复杂的实体和关系, 难以捕捉其语义信息。

(2)基于语义匹配的模型。传统的语义匹配嵌入模型有 RESCAL<sup>[25]</sup>, DistMult 和 HolE 等。GETD<sup>[26]</sup> 将张量分解方法从二元关系知识图谱拓展到多元关系知识图谱, 将 Tucker 分解拓展到对应于多元关系的  $n$ -Tucker 分解, 重塑得到  $(n+1)$  阶的核心张量  $\mathbf{W}$ 。S2S<sup>[27]</sup> 分割实体和关系嵌入得到  $M(M \leq N)$  个片段, 允许模型在不同元数事实之间共享嵌入。这类模型通过分解张量来学习实体和关系之间的复杂交互模式, 表现力更强, 但存在过度拟合和数据稀疏的问题。

(3)基于神经网络的模型。传统的神经网络嵌入模型有 ConvE<sup>[28]</sup>, ConvKB<sup>[29]</sup> 和 RSN<sup>[30]</sup> 等。DRR-GAT<sup>[31]</sup> 模型使用图注意力网络得到实体的局部嵌入表示, 通过注意力机制对邻居节点加权来更新目标节点的表示。MA-GNN<sup>[32]</sup> 是一种基于多注意力的双分支支神经网络, 利用多注意力机制分别处理知识图谱中的局部信息和全局信息。SR-GNN<sup>[33]</sup> 模型设计了一个实体语义聚合模块, 通过循环神经网络聚合同一中心实体的邻居实体的语义相似性信息, 再通过图神经网络将不同的关系类型的邻居信息传递给中心实体。这类模型可以学习到实体和关系的非线性表示, 并且可以处理复杂的关系和层次结构。其缺点是参数量大, 计算复杂度偏高, 且存在数据稀疏问题和过拟合问题。

## 2.2 基于概念和实例的知识图谱嵌入模型

这类模型利用概念和实例之间的关系来增强知识图谱嵌入学习。现有工作大致可分为两类。

(1)在同一嵌入空间中建模: 这类工作对同一维度向量空间中不同层次的实体进行建模。代表方法有 TKRL<sup>[34]</sup>, JECI++, TransC, Concept2Box 和 TransO 等。TKRL 通过充分利用实体类型到实体的投影矩阵来捕获层次信息, 将概念建模为投影矩阵, 并使用线性组合或矩阵乘法对分层结构进行编码。JECI++ 简化概念层次结构并重新将实例链接到简化后的概念上, 然后使用循环卷积在嵌入空间中定位实例。然而, 该模型只利用了知识图谱的一部分。TransC 将每个概念建模为语义空间中的一个球体, 将每个实例建模为同一语义空间中的一个点, 并通过点与球体的相对位置来建模实例与概念之间潜在的语义关系。Concept2Box 使用对偶几何表示联合嵌入 KGs 的两个视图, 将概念建模为潜在空间中的盒子, 盒子嵌入学习层次结构和它们之间的复杂关系, 如重叠和不相交。传统的知识图谱嵌入忽略本体信息约束, TransO 引入本体信息(实体类型和层次)约束, 可以显式地对关系进行建模, 增强模型的推理能力并保持较低的模型复杂度。

这些方法将实体放在同一向量空间的不同层次上, 而不区分它们的语义差异, 并且经常引入额外的参数, 增加了时间

和空间的复杂性。

(2)在不同嵌入空间中建模: 这类工作在不同的维度向量空间中对不同层次的实体进行建模。代表方法有 JOIE, KEC<sup>[35]</sup>, HiTE<sup>[36]</sup>, CIST, DGS 和 EIKE 等。JOIE 使用非线性仿射变换将实例向量映射到概念向量空间, 通过视图内模型分别捕获嵌入空间中实例和概念的结构化知识, 并使用跨视图模型学习实例和概念之间的语义关系。KEC 将三元组的头尾实体向量投影到概念空间中对应概念的超平面上, 保证了概念的语义约束。HiTE 通过约束头部和尾部实体的概念嵌入之间的有序关系来建模实体的层次结构。CIST 在不同的嵌入空间中对概念和实例进行建模, 并为概念引入可学习的参数来建模其层次结构。然而, 这些方法通常通过多次人工超参数搜索来确定不同层次实体的嵌入维数, 导致更高的训练成本。双几何空间嵌入模型 DGS 使用复杂的非欧几里德几何空间来建模双视图 KGs。EIKE 首次同时建模外延(结构)与内延(文本)知识, 采用基于几何的方法建模外延知识。这些方法通过几何区域直观表示概念的包含关系, 支持层次推理且计算效率较高, 但是对复杂语义的建模能力受限。与上述方法相比, 本文 JMOI 模型区分了不同层次的实体, 不仅可以缓解数据稀疏性问题, 而且可以很好地对概念的层次结构和 isA 关系<sup>[16]</sup>的传递性进行建模。

## 3 问题描述

知识图谱: 知识图谱描述实体(概念、实例)以及它们之间的关系。它可以形式化为  $KG = (C, I, R, S)$ , 其中  $C, I, R, S$  分别表示概念集、实例集、关系集和三元组集。关系集  $R$  可以形式化为  $R = \{r_i, r_c\} \cup R_I$ , 其中  $r_i$  是一个 instanceOf 关系,  $r_c$  是一个 subClassOf 关系,  $R_I$  是实例关系集。本文将三元组集  $S$  分为 3 个不相交的子集。

(1)实例三元组集  $S_I = \{(i, r_i, c)_k\}_{k=1}^{n_i}$ , 其中  $i \in I$  是一个实例,  $c \in C$  是一个概念,  $n_i$  是  $S_I$  的大小。

(2)子类三元组集  $S_C = \{(c_i, r_c, c_j)_k\}_{k=1}^{n_c}$ , 其中  $c_i, c_j \in C$  是概念,  $c_i$  是  $c_j$  的子概念,  $n_c$  是  $S_C$  的大小。

(3)关系三元组集  $S_R = \{(h, r, t)_k\}_{k=1}^{n_r}$ , 其中  $h, t \in I$  是头实例和尾实例,  $r \in R_I$  是一个实例关系,  $n_r$  是  $S_R$  的大小。

知识图谱嵌入: 知识图谱嵌入记为  $f: C, I, R \rightarrow \mathbb{R}^k$ , 是将概念  $C$ 、实例  $I$  和关系  $R$  投影到向量空间  $\mathbb{R}^k$  中的映射,  $f$  分别为实体和关系的嵌入函数, 将实体和关系转换为  $k$  维空间的矢量表示。

## 4 JMOI 模型

### 4.1 模型描述

知识图谱大多可分为两种基本视图, 即实例视图及其本体视图, 如图 1 所示。实例视图主要包含三元组特定实体及其之间的关系, 本体视图主要描述的是抽象概念及其语义元关系。两个视图之间通过本体和实例的类属关系链接。实例三元组包含概念和实例, 子类三元组描述了概念之间的层次关系, 关系三元组是普通三元组。每一种三元组都应采用不同的嵌入机制进行训练, 以充分捕获每一

种三元组中存在的潜在语义链接。

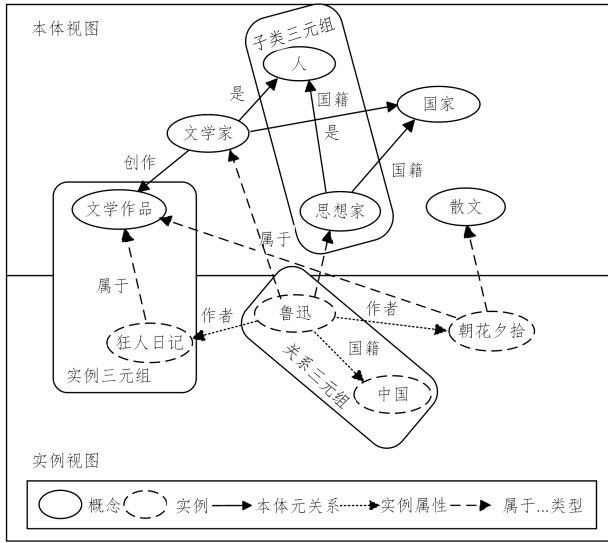


图1 知识图谱中本体和实例视图示例

Fig. 1 Example of ontology and instance views in knowledge graph

基于  $S = \{S_I \cup S_C \cup S_R\}$  对知识图谱中的 3 种类型的三元组进行建模,具体描述如下:

对于实例三元组  $(i, r_i, c)$ , 实例  $i$  和概念  $c$  的嵌入分别为  $n$  和  $m$ 。实例三元组建模模型如图 2 所示。为了有效地利用实例与概念之间的潜在关联性,本文使用自注意力机制将实例嵌入  $n$  映射到概念嵌入  $m$ 。这种方法不仅将实例与概念映射到同一个空间中,而且还能够捕捉它们之间的长距离依赖关系。

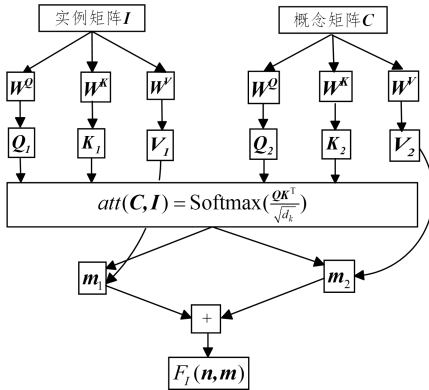


图2 实例三元组建模模型图

Fig. 2 Diagram of instance of triples building model

在这一过程中,将实例  $i$  作为 query 向量,而概念  $c$  同时作为 key 向量和 value 向量。这种设计使得模型能够聚焦于实例和概念之间的直接关系,同时考虑整个知识图谱的上下文信息。在实际实现中,为了提高计算效率,通过矩阵的形式计算注意力,将多个实例的表示向量及其对应的概念表示向量分别拼接为实例矩阵  $I$  和概念矩阵  $C$ 。然后,引入 3 个待训练的权重矩阵  $W^Q$ ,  $W^K$  和  $W^V$ ,通过权重矩阵与实例矩阵和概念矩阵分别做矩阵相乘操作,得到 query, key 和 value 对应的矩阵  $Q, K$  和  $V$ 。计算过程如下:

$$Q = I \times W^Q \quad (1)$$

$$K = C \times W^K \quad (2)$$

$$V = C \times W^V \quad (3)$$

其中,  $C \in \mathbb{R}^{|C| \times k}$ ,  $I \in \mathbb{R}^{|I| \times k}$ ,  $W^Q \in \mathbb{R}^{k \times d_k}$ ,  $W^K \in \mathbb{R}^{k \times d_k}$ ,  $W^V \in \mathbb{R}^{k \times d_k}$ ,  $W^Q \in \mathbb{R}^{k \times d_k}$ ,  $W^V \in \mathbb{R}^{k \times k}$ ,  $k$  是实体和概念的向量维度。

接着,计算查询  $(Q)$  和键  $(K)$  之间的点积,并除以  $\sqrt{d_k}$  (其中  $d_k$  是权重矩阵的维度)来缩放点积,以防止梯度消失。然后应用 Softmax 函数来获取注意力分数:

$$att(C, I) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \quad (4)$$

最后,将注意力分数乘以值  $(V)$  来获取最终的概念嵌入表示:

$$m = \text{Selfatt}(Q, K)V \quad (5)$$

其中,  $\text{Selfatt}(Q, K)$  表示自注意力机制,突出重要的信息并抑制不重要的信息。目标函数可以定义为最小化实例嵌入  $n$  通过自注意力机制后的表示和概念嵌入  $m$  之间的距离,同时考虑概念嵌入的邻域范围  $\delta_i$ 。

$$F_1(n, m) = \|m - \text{Selfatt}(n)\|_2 - \delta_i \quad (6)$$

通过最小化这个目标函数,模型可以学习到如何将实例嵌入映射到概念嵌入,同时捕捉它们之间的复杂关系。这种方法不仅提高了嵌入的质量,还增强了模型对知识图谱中实体和概念之间关系的理解和预测能力。

对于子类三元组  $(c_i, r_c, c_j)$ , 概念  $c_i$  是概念  $c_j$  的子概念。 $m_i$  和  $m_j$  代表的是概念的嵌入向量,其中  $m_i$  是子概念的嵌入向量,而  $m_j$  是父概念的嵌入向量。在本文模型中,所有概念的嵌入向量都在同一个  $d$  维向量空间中建模。概念在层次结构中越高,其嵌入向量中每个元素的绝对值越小;反之,层次越低的概念,其嵌入向量中每个元素的绝对值越大。更具体的类别(即层次较低的类别)通常具有更多的属性和特征,因此在向量空间中应该有更显著表示;而更抽象的类别(即层次较高的类别)则具有更少的特定属性,因此在向量空间中的表示应该更为模糊。嵌入向量  $m_j$  中每个元素的绝对值小于嵌入向量  $m_i$  中每个元素的绝对值。因此,本文定义具有层次关系的子类三元组  $(c_i, r_c, c_j)$  的评分函数为:

$$F_C(m_i, m_j) = \sum_{k=1}^d [m_{j_k}^2 - m_{i_k}^2]_+ \quad (7)$$

其中,  $m_{j_k}$  表示嵌入向量  $m_j$  的第  $k$  个元素;  $m_{i_k}$  表示嵌入向量  $m_i$  的第  $k$  个元素;  $[z]_+ = \max\{0, z\}$  表示  $[z]_+$  只保留  $z$  为正的部分,其余部分设为 0。

通过最小化这个评分函数,可以确保模型学习到的嵌入向量能够反映出概念之间的层次关系。得到上面的嵌入之后, JMOI 可以很容易地处理 isA 关系的传递性。实例三元组的传递性如式(8)所示:

$$(i, r_i, c_1) \in S_I \wedge (c_1, r_c, c_2) \in S_C \rightarrow (i, r_i, c_2) \in S_I \quad (8)$$

如果实例  $i$  属于概念  $c_1$ , 且  $c_1$  是  $c_2$  的子概念, 那么实例  $i$  也属于概念  $c_2$ 。

子类三元组的传递性如式(9)所示:

$$(c_3, r_c, c_4) \in S_C \wedge (c_4, r_c, c_5) \in S_C \rightarrow (c_3, r_c, c_5) \in S_C \quad (9)$$

如果概念  $c_3$  是  $c_4$  的子概念, 且  $c_4$  是  $c_5$  的子概念, 那么  $c_3$  也是  $c_5$  的子概念。

对于关系三元组  $(h, r, t)$ , 采用经典的 TransE 模型。TransE 模型将关系视为头实体和尾实体之间的平移向量, 目标函数定义为:

$$F_R(h, r, t) = \| \mathbf{h} + \mathbf{r} - \mathbf{t} \|_2 \quad (10)$$

其中,  $\mathbf{h}$  和  $\mathbf{t}$  分别是头实体和尾实体的嵌入向量,  $\mathbf{r}$  是关系的嵌入向量。这个目标函数的目的是使得头实体和关系嵌入向量的和尽可能接近尾实体的嵌入向量。

## 4.2 训练方法

使用基于均匀分布的负抽样策略随机替换正三元组  $(h, r, t)$  中的  $h, t$ , 生成负三元组, 且新生成的负三元组  $(h', r, t)$  在 KGs 中不存在。

对于实例三元组, 使用  $\delta$  和  $\delta'$  来表示正三元组和负三元组, 用  $S_I$  和  $S_I'$  来表示正三元组集和负三元组集。对于实例三元组集  $S_I$ , 定义基于边际的排序损失函数  $L_I$ :

$$L_I = \sum_{\delta \in S_I} \sum_{\delta' \in S_I'} [\gamma_I + F_I(\delta) - F_I(\delta')]_+ \quad (11)$$

其中,  $[x]_+ = \max(0, x)$ ,  $\gamma_I$  是实例三元组的边界超参数。

对于子类三元组集  $S_C$ , 定义基于边际的排序损失函数  $L_C$ :

$$L_C = \sum_{\delta \in S_C} \sum_{\delta' \in S_C'} [\gamma_C + F_C(\delta) - F_C(\delta')]_+ \quad (12)$$

对于关系三元组集  $S_R$ , 定义基于边际的排序损失函数  $L_R$ :

$$L_R = \sum_{\delta \in S_R} \sum_{\delta' \in S_R'} [\gamma_R + F_R(\delta) - F_R(\delta')]_+ \quad (13)$$

最后, 将整体损失函数定义为以下 3 个损失函数的线性组合:

$$L = \beta_1 \cdot L_I + \beta_2 \cdot L_C + L_R \quad (14)$$

其中,  $\beta_1, \beta_2 > 0$  是平衡  $L_I, L_C$  和  $L_R$  之间的超参数。为了学习模型参数, 利用随机梯度下降 (SGD) 算法对上述损失函数进行优化。

## 5 实验与结果分析

本章对 JMOI 模型在两项任务, 即链路预测和三元组分类上的性能进行验证和评价。首先, 简要地介绍实验设置、数据集。然后, 讨论这两项任务的评价指标、实验的具体配置以及相应的实验结果, 对 JMOI 模型的实验性能进行分析, 并将其与其他知识图谱嵌入模型进行比较。

### 5.1 实验设置

实验在 Windows Server 2022 Standard 操作系统下进行, CPU 为 Intel Xeon Gold 6248R CPU @ 3.0 GHz, GPU 为 NVIDIA A100-PCIE-40GB, 软件框架是基于 PyTorch 1.12.1 的深度学习统一训练框架, 开发语言为 Python。

#### 5.1.1 数据集

之前的工作大多使用 FB15K 和 WN18 进行评估, 但 FB15K 主要由实例组成, WN18 主要由概念组成, 因此不适合评估本文模型。本文使用 JOIE<sup>[10]</sup> 中提出的两个数据集<sup>1)</sup>: 从 YAGO 中提取的 YAGO26K-906 和从 DBpedia 中提取的 DB111K-174。数据集的统计信息如表 1 所列。

表 1 数据集的统计信息

Table 1 Statistical information of the datasets		
	YAGO26K-906	DB111K-174
实例	26078	111762
概念	906	174
实例关系	34	305
概念关系	29	19
实例三元组	9962	99748
子类三元组	1410	163
实例关系三元组	390738	863643
概念关系三元组	7552	600

#### 5.1.2 对比模型

本文将对对比模型分为两类: (1) 基础的知识图谱嵌入模型, 即不需要任何辅助信息知识图谱嵌入模型, 选择 TransE, DistMult 和 HolE 等经典模型作为对比模型; (2) 基于实体层次结构信息知识图谱嵌入模型, 选择 TransC, JOIE, DGS, CIST 和 Concept2Box 作为对比模型。部分对比模型的实验结果直接取自 JOIE 和 CIST。

### 5.2 实验对比与分析

#### 5.2.1 链接预测

链接预测的主要目的是在关系三元组  $(h, r, t)$  中, 给定  $(h, r)$  预测缺失的尾实体  $t$ , 或给定  $(r, t)$  预测缺失的头实体  $h$ 。在这个任务中, 需要给出知识图谱中候选实体的排序列表, 而不是直接获得最佳答案。本文在这项任务中使用了两个评估指标: 一个是所有测试三元组的平均倒数秩 (MRR), 另一个是排名不高于  $N$  的测试三元组的比例 (Hits@ $N$ )。模型的性能越好, MRR 和 Hits@ $N$  的值越高。在本文实验中, 链接预测分为两个子任务, 即实例关系三元组的链接预测和概念关系三元组的链接预测, 在 YAGO26K-906 和 DB111K-174 两个数据集上执行了这两个子任务。

首先, 将三元组数据集的 85% 作为训练集, 5% 作为验证集, 10% 作为测试集。对于每个测试三元组  $(h, r, t)$ , 用知识图谱中的实体  $x$  替换尾实体  $t$ , 以生成一组负三元组  $(h, r, x)$ 。计算负三元组的目标函数得分  $F(h, r, x)$ , 然后按升序对得分进行排序, 以获得测试三元组  $(h, r, t)$  在所有负三元组中的排名。同样, 也可以得到一组候选三元组  $(x, r, t)$  的目标函数得分排序, 这些候选三元组  $(x, r, t)$  取代了测试三元组  $(h, r, t)$  的头部实体  $h$ 。此外, 如果知识图谱中存在候选三元组 (即替换了头或尾的候选三元组仍然是正确的), 则其目标函数得分低于原始测试三元组的得分是合理的。为了消除这种干扰因素的影响, 采用了先前研究工作中使用的“过滤器”设置<sup>[18]</sup>, 以确保候选三元组不属于任何数据集。

#### (1) 参数设置

在训练过程中, 本文使用了随机梯度下降法中的以下超参数设置: 学习率  $\lambda$  为  $\{0.0005, 0.001, 0.01, 0.1\}$ , 边缘超参数  $\gamma_I, \gamma_C, \gamma_R$  为  $\{0.1, 0.2, 0.5, 1, 2, 5, 7\}$ , 嵌入空间维数  $d_e$  为  $\{25, 50, 75, 100\}$ ,  $d_r$  为  $\{100, 300, 500, 1000\}$ ,  $\beta_1, \beta_2$  为  $\{0.5, 1\}$ , 阈值超参数  $\delta_e$  为  $\{0, 3, 5, 8, 10\}$ , 批量大小  $B$  为 128。通过验证集上的 Hits@10 确定最佳参数。自对抗负采样参数  $\alpha$  为  $\{0.5, 1.0\}$ 。两个数据集的最佳参数为相同值, 配置如表 2 所列。在每个数据集上迭代训练模型 5000 次。

<sup>1)</sup> <https://github.com/JunhengH/joie-kdd19>

表2 链接预测任务的最优参数配置

Table 2 Optimal parameter configuration for link prediction task

数据集	$\lambda$	$\gamma_i$	$\gamma_c$	$\gamma_r^i$	$\gamma_r^c$	$d_c$	$d_e$	$\beta_1$	$\beta_2$	$B$	$\alpha$
YAGO26K-906	0.001	2	1	5.0	5.0	50	300	1	1	128	1
DB111K-174	0.001	2	1	5.0	5.0	50	300	1	1	128	1

## (2) 实验结果

实验结果如表3所列。从表中可以看出, JMOI 几乎在所有的评价指标方面都比其他对比模型表现得更好, 特别是

使用自对抗负抽样方法的 JMOI(self-adv)。在 Hits@1 和 Hits@10 上, JMOI(self-adv) 的得分几乎是其他对比模型的两倍, 从而证明了本文模型的有效性。

表3 链接预测结果

Table 3 Results of link prediction

模型	YAGO26K-906						DB111K-174					
	实例关系三元组			概念关系三元组			实例关系三元组			概念关系三元组		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
TransE	0.160	8.66	30.17	0.145	12.29	20.59	0.212	9.66	41.85	0.164	13.43	19.78
DisMult	0.223	10.91	32.76	0.197	17.72	25.08	0.163	8.07	32.54	0.153	12.43	16.32
HoIE	0.235	12.90	34.31	0.192	18.70	20.29	0.206	9.53	37.68	0.144	15.50	20.11
TransC	0.159	8.65	30.04	0.210	20.24	26.85	0.217	9.94	42.74	0.197	16.60	23.88
JOIE	0.176	8.64	32.68	0.263	16.72	38.46	0.208	9.38	41.90	0.461	25.00	78.73
DGS	<u>0.366</u>	23.15	56.06	0.290	25.88	44.38	0.226	10.31	60.85	0.470	26.88	78.82
CIST	0.364	<u>23.97</u>	61.54	0.291	30.03	61.13	0.322	15.50	62.41	0.529	34.33	79.11
Concept2Box	0.344	23.33	<u>62.01</u>	<u>0.293</u>	<u>31.16</u>	<u>62.92</u>	<u>0.325</u>	<u>16.52</u>	<u>63.11</u>	<u>0.530</u>	<u>34.82</u>	80.01
JMOI(unif)	0.158	8.59	31.15	0.237	25.70	54.45	0.229	10.96	45.76	0.488	27.60	<u>80.11</u>
JMOI(self-adv)	<b>0.375</b>	<b>24.62</b>	<b>63.91</b>	<b>0.324</b>	<b>32.14</b>	<b>63.35</b>	<b>0.331</b>	<b>18.73</b>	<b>64.63</b>	<b>0.549</b>	<b>36.17</b>	<b>81.09</b>

注: 最佳结果用粗体显示, 次佳结果用下划线显示。

首先, 在两个数据集上的实验结果表明, 与 TransE, DistMult 和 HoIE 等基本知识图谱嵌入模型相比, JMOI 模型在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 94.7%, 84.9% 和 150.3%。这表明, JMOI 模型能更好地利用实例和概念之间潜在的语义关系来学习实例和概念的具体特征, 提高了 JMOI 模型在链接预测任务中的性能。其次, 与 TransC 相比, JMOI 在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 73.4%, 69.1% 和 111.1%。这是因为与 TransC 相比, JMOI 使用自注意力机制将实例与概念映射到同一个空间中, 能够捕捉它们之间的长距离依赖关系。实验结果证明了在同一嵌入空间中对实例和概念建模的有效性。

此外, 与 JOIE 相比, JMOI 在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 26.4%, 62.8% 和 33.2%。这是因为 JMOI 在概念的嵌入中添加了一个邻域范围参数, 这可以对 isA 关系的传递性以及同一实例属于不同概念的情况进行建模。再次, 与 DGS 相比, JMOI 在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 19.4%, 36.7% 和 16.5%。这可能是因为 JMOI 模型本体中的层次结构通过子类三元组的传递性比双曲空间能更好地建模。另外, 与 CIST 相比, JMOI 在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 4.7%, 9.0% 和 3.5%。这表明, JMOI 模型能够更好地捕捉实体的层次结构信息, 在链路预测任务中的性能得到提高, 证明了所提出的实体层次信息建模方法的有效性。最后, 与 Concept2Box 相比, JMOI 在 MRR, Hits@1 和 Hits@10 上的性能分别平均提高了 6.3%, 6.5% 和 1.9%。这表明引入自注意力机制可以很好地捕获实例与概念之间的潜在关联性。

在链接预测任务中, “JMOI(unif)” 和 “JMOI(self-adv)”

分别表示采用不同负采样策略的 JMOI 模型。“JMOI(unif)” 表示采用均匀负采样策略 (Uniform Negative Sampling) 的 JMOI 模型。均匀负采样是一种简单的随机采样方法, 它从负样本候选集中随机选择样本, 而不考虑样本的具体特征或分布。这种方法的优点是实现简单, 但其可能无法有效捕捉到对模型训练更有价值的负样本, 从而影响模型的性能。“JMOI(self-adv)” 表示采用自对抗负采样策略 (Self-Adversarial Negative Sampling) 的 JMOI 模型。自对抗负采样是一种更先进的采样方法, 它通过生成与正样本在语义上相似但不相同的负样本, 增强模型的训练效果, 从而提高模型对负样本的区分能力。从表中可以观察到, 使用 “self-adv” 抽样的 JMOI 模型在多个指标上都有所提升, 这表明自对抗负采样策略可能更有效地生成了负样本, 提高了模型的预测性能。

综上所述, 在链接预测任务上的对比实验结果表明, JMOI 在此任务上表现良好。进一步分析结果得出, JMOI 模型能够有效地捕捉知识图谱中实体之间的层次结构信息并对其进行建模, 从而提高知识图谱嵌入的有效性。

## 5.2.2 三元组分类

三元组分类的主要任务是确定给定的三元组是正三元组还是负三元组。测试三元组是否被标记为 “正确” 或 “不正确”。三元组可以是实例关系三元组、概念关系三元组、实例三元组或子类三元组。这是一个二元分类任务, 其评价指标包括二元分类任务中常用的正确率、准确率、召回率和 F1 值。

首先, 将三元组数据集的 85% 作为训练集, 5% 作为验证集, 10% 作为测试集。对于关系三元组数据集中的每个关系  $r$ , 设置一个阈值  $\delta_r$ , 通过最大化验证集的分类准确度来确定阈值  $\delta_r$ 。对于给定的测试关系三元组  $(h, r, t)$ , 计算它的得分

函数  $F(h, r, t)$ , 如果函数得分小于阈值  $\delta_r$ , 则预测该三元组的标签为“正”, 否则为“负”。对于实例三元组  $(x, r_i, c)$ , 如果式(6)的分数小于  $\delta_{ri}$ , 则预测它为“正”。对于子类三元组  $(x, r_c, c)$ , 如果式(7)的分数小于  $\delta_{rc}$ , 则预测该三元组为“正”。

## (1) 参数设置

此任务的参数优化设置与链路预测任务相同。最佳配置由验证集的精度确定。两个数据集的最佳参数配置如表 4 所列。在每个数据集上迭代训练模型 5000 次。

表 4 三元组分类任务的最优参数配置

Table 4 Optimal parameter configuration for triple classification task

数据集	$\lambda$	$\gamma_i$	$\gamma_c$	$\gamma_r^i$	$\gamma_r^c$	$d_c$	$d_e$	$\beta_1$	$\beta_2$	$B$	$\alpha$
YAGO26K-906	0.001	2	1	5.0	5.0	50	300	1	1	128	0.5
DB111K-174	0.001	2	1	5.0	5.0	50	300	1	1	128	1

## (2) 实验结果

数据集中的三元组可以分为 4 种不同的类型, 即实例关系三元组、概念关系三元组、实例三元组和子类三元组。分别对这 4 种类型的三元组进行实验, 实验结果如表 5—表 8 所列, 其中最佳结果用粗体显示, 次佳结果用下划线

显示。JMOI 模型在两个数据集上均获得了最优或接近最优的实验结果, 具有最高的准确率和 F1 值, 表明它在三元组分类任务中优于先进的模型。尽管一些对比模型在精确率和召回率上优于 JMOI, 但 JMOI 仍然获得了最高的 F1 值。

表 5 实例关系三元组的三元组分类结果

Table 5 Experiment results on triple classification for instance relational triples

(%)

模型	YAGO26K-906				DB111K-174			
	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
TransE	92.92	94.91	90.70	92.76	68.01	72.61	57.85	64.39
TransC	93.73	96.94	90.31	93.50	69.76	72.27	64.12	67.95
JOIE	93.37	<b>96.99</b>	89.52	93.11	71.57	72.30	69.93	71.09
DGS	94.21	95.18	91.75	93.46	71.61	72.29	69.97	69.16
CIST	96.53	96.38	96.68	96.53	71.90	72.18	71.28	71.73
Concept2Box	96.79	96.52	96.91	96.40	71.93	72.38	71.50	71.12
JMOI(unif)	<u>98.51</u>	96.78	<u>98.64</u>	<u>98.73</u>	<u>75.86</u>	<u>75.29</u>	<u>77.36</u>	<u>76.63</u>
JMOI(self-adv)	<b>98.62</b>	<u>96.98</u>	<b>99.68</b>	<b>99.65</b>	<b>76.50</b>	<b>75.37</b>	<b>78.42</b>	<b>77.01</b>

表 6 概念关系三元组的三元组分类结果

Table 6 Experiment results of triple classification for concept relational triples

(%)

模型	YAGO26K-906				DB111K-174			
	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
TransE	52.50	51.81	71.50	60.08	58.73	57.53	66.67	61.76
JOIE	74.75	<b>77.36</b>	73.50	75.38	81.75	81.25	82.53	81.89
DGS	74.90	71.26	73.70	75.52	82.13	81.28	82.31	81.93
CIST	75.00	<u>73.81</u>	77.50	75.61	82.74	81.30	<u>84.13</u>	82.69
Concept2Box	75.11	71.45	78.60	76.12	83.25	82.73	83.22	82.84
JMOI(unif)	<u>75.30</u>	72.18	<u>85.20</u>	<u>77.41</u>	<u>85.59</u>	<u>88.34</u>	<b>84.23</b>	<u>83.60</u>
JMOI(self-adv)	<b>76.35</b>	71.38	<b>87.21</b>	<b>77.95</b>	<b>86.65</b>	<b>89.47</b>	83.40	<b>85.79</b>

表 7 实例三元组的三元组分类结果

Table 7 Experiment results of triple classification for instance of triples

(%)

模型	YAGO26K-906				DB111K-174			
	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
TransE	50.53	50.47	57.74	53.86	53.57	52.38	78.57	62.86
TransC	62.38	63.79	57.29	60.36	52.54	52.40	55.49	53.90
JOIE	58.44	61.22	46.07	52.58	54.88	53.85	68.31	60.22
DGS	60.34	62.08	52.47	61.35	55.06	53.91	65.26	60.31
CIST	74.75	<u>77.65</u>	69.5	73.35	55.69	54.59	67.65	60.43
Concept2Box	74.80	70.15	70.17	72.93	56.43	55.76	65.79	62.08
JMOI(unif)	<u>76.75</u>	<b>77.91</b>	<u>72.30</u>	<u>74.35</u>	<u>68.74</u>	<u>71.65</u>	<u>67.75</u>	<u>66.53</u>
JMOI(self-adv)	<b>77.50</b>	70.26	<b>95.60</b>	<b>81.52</b>	<b>69.38</b>	<b>71.72</b>	<b>68.90</b>	<b>67.49</b>

首先, 与 TransE 相比, JMOI 在准确率、精确率、召回率和 F1 值上分别平均高出 33.9%, 32.6%, 20.5% 和 25.8%。这进一步证明了实体层次信息作为知识图谱嵌入的外部信息的有效性。其次, 从表 5、表 7 和表 8 可以看出, 与 TransC 相

比, JMOI 在准确率、精确率、召回率和 F1 值上分别平均高出 21.0%, 19.9%, 17.2% 和 17.7%。实验结果进一步证明了将实例和概念建模到同一嵌入空间的有效性。TransC 不能对概念关系三元组建模, 因此表 6 中不包括 TransC。此外,

与 JOIE 相比, JMOI 在准确率、精确率、召回率和 F1 值上分别平均高出 11.4%, 8.9%, 18.5% 和 12.5%。这进一步证明了 JMOI 可以更好地建模 isA 关系的传递性, 学习实例和概念之间潜在的语义联系。

再次, 与 DGS 相比, JMOI 在准确率、精确率、召回率和 F1 值上分别平均高出 10.6%, 9.5%, 16.0% 和 8.9%, 进一步证明了 JMOI 能够学习较好的实体和关系的向量表示。另外, 与 CISI 相比, JMOI 在准确率、精确率、召回

率和 F1 值上分别平均高出 4.4%, 5.1%, 6.0% 和 3.9%。这进一步证明了 JMOI 在学习实体嵌入时, 可以利用实例和概念之间潜在的语义关系对实例和概念的具体特征进行建模。最后, 与 Concept2Box 相比, JMOI 在准确率、精确率、召回率和 F1 值上分别平均高出 5.3%, 6.5%, 6.9% 和 4.2%。这表明通过对知识图谱嵌入模型中的实体层次结构信息进行精确建模, JMOI 模型能够获得良好的嵌入表示效果。

表 8 子类三元组的三元组分类结果

Table 8 Experiment results of triple classification for subclass of triples

模型	YAGO26K-906				DB111K-174			
	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
	(%)							
TransE	52.54	51.92	63.12	56.97	52.20	51.79	63.58	57.08
TransC	54.22	53.27	68.75	60.03	60.71	57.14	85.71	68.57
JOIE	72.79	73.13	72.06	72.59	64.29	64.29	64.29	64.29
DGS	72.86	73.54	72.22	72.63	64.55	65.84	65.45	69.84
CIST	74.82	74.02	<u>76.47</u>	75.22	71.43	68.75	78.57	73.33
Concept2Box	73.32	74.13	75.81	74.49	70.05	66.71	71.85	72.35
JMOI(unif)	74.83	<u>75.12</u>	<b>76.57</b>	<u>75.26</u>	<u>71.45</u>	<u>68.83</u>	<u>78.60</u>	<b>73.34</b>
JMOI(self-adv)	<b>75.82</b>	<b>75.91</b>	75.47	<b>75.73</b>	<b>71.56</b>	<b>68.85</b>	<b>78.82</b>	<u>74.33</u>

JMOI 模型在 YAGO26K-906 数据集上的 Precision 值较低, 可能是因为 YAGO26K-906 数据集具有较高的复杂度, 包含丰富的语义关系和层次结构, 模型在处理如此复杂的数据时, 容易出现过拟合或欠拟合现象, 导致预测结果不准确, Precision 值降低; 也可能是因为数据集中存在噪声数据, 这些低质量的数据会影响模型的训练效果, 使其在预测时产生更多错误, 从而导致 Precision 值下降。

在 JMOI 模型中, 与均匀负采样 (“unif”) 策略相比, 采用自对抗负采样 (“self-adv”) 的 JMOI 模型在准确率、召回率和 F1 值上分别平均提升了 0.9%, 1.2% 和 2.2%。这表明, 自对抗负采样能够通过生成更具挑战性的负样本, 有效增强模型对复杂样本的区分能力。然而, 在精确率指标上, “self-adv” 策略的表现略有下降, 这可能是模型对语义模糊的负样本存在误判, 导致部分负样本被预测为正样本。综上所述, JMOI 可以更好地嵌入概念和实例, 并且可以更好地建模 isA 关系的传递性, 使得 JMOI 模型在三元组分类任务中取得更好的效果, 从而进一步验证了模型的有效性。

### 5.2.3 消融实验

为了研究各个模块对 JMOI 的有效性, 在 YAGO26K-906 和 DB111K-174 数据集上对该模型进行了消融实验。首先, 通过在实例三元组建模中删除自注意力模块 (SA) 来探索它们的影响。然后, 将概念嵌入中原本可学习的邻域参数设为固定值, 分析动态调整概念层次的影响 (表示为 NP)。表 9 列出了消融实验的性能比较结果。

通过对 JMOI 中部分模块进行消去, 发现模型中每个模块对性能都有影响, 从而证明了模型中每个模块都具有有效性。从实验数据中可以发现, 在 YAGO26K-906 数据集和 DB111K-174 数据集上, 通过消去自注意力机制模块, Hits@10 性能分别下降了 2.1% 和 1.1%, MRR 性能分别下降了 3.4% 和 5.4%。这表明自注意力机制能有效捕捉实例与概念间的复杂语义关联, 显著提升模型对长尾实体的泛化能力。

固定邻域参数后, Hits@10 性能分别下降了 0.3% 和 0.9%, MRR 性能分别下降了 2.2% 和 6.3%。这表明动态邻域参数通过调整概念嵌入范围, 能够有效区分层次化语义信息。上述消融实验验证了模型中各个模块对最终性能都有一定贡献, 证明了模型中各模块的必要性。

表 9 YAGO26K-906 和 DB111K-174 数据集上的消融实验

Table 9 Ablation experiments on YAGO26K-906 and DB111K-174

模型	datasets			
	YAGO26K-906		DB111K-174	
	<i>Hits@10</i>	<i>MRR</i>	<i>Hits@10</i>	<i>MRR</i>
JMOI-SA	64.27	0.353	80.19	0.516
JMOI-NP	65.42	0.357	80.35	0.512
JMOI	<b>65.61</b>	<b>0.365</b>	<b>81.09</b>	<b>0.544</b>

### 5.3 模型复杂度

本节分析和解释了 JMOI 模型参数的复杂度。用  $N_i$  和  $N_c$  分别表示实例和概念的数量, 用  $d$  来表示嵌入向量的维度。对于实例三元组, 实例和概念的嵌入向量的模型参数复杂度为  $O(N_i d + N_c d)$ , 自注意力机制引入了 3 个权重矩阵的复杂度为  $O(3d^2)$ , 所以实例三元组的总参数复杂度为  $O(N_i d + N_c d + d^2)$ 。对于子类三元组, 总参数复杂度为  $O(N_c d)$ 。关系三元组的总参数复杂度为  $O(N_i d + N_c d)$ 。JMOI 模型的总参数复杂度为  $O(N_i d + N_c d + N_c d + d^2)$ 。

自注意力机制和可学习邻域参数的引入, 以及整体损失函数的复杂计算, 导致训练时间相应增加。特别是对于大规模知识图谱数据集, 如 DB111K-174, 训练时间可能会显著延长。在预测阶段, 需要进行更多的计算来处理自注意力机制和邻域参数, 导致推理时间略有增加。

尽管计算复杂性增加, 但模型在链接预测和三元组分类任务上的性能显著提升。例如, 在链接预测任务中, JMOI 模型在 Hit@1 和 Hits@10 上的得分几乎是其他对比模型的两倍, 这表明计算复杂性的增加是值得的。通过自注意力机制

和动态邻域参数,模型能够更好地捕捉知识图谱中的复杂结构和语义信息,提高了对长尾实体的泛化能力,这在实验结果中得到了验证。

**结束语** 本文提出了一个新的知识图谱嵌入模型 JMOI,该模型探索了概念和实例之间潜在的语义关系,通过引入自注意力机制为实例和概念之间的关系提供了一种灵活和强大的建模方式,它能够捕捉序列内部的复杂关系,提高模型的性能和泛化能力;同时增加了一个可学习的参数来调整概念嵌入的邻近范围,从而能够区分不同概念的层次信息。模型增强了知识图谱的嵌入,可以有效缓解数据稀疏导致的实体嵌入不足的问题。实验结果表明,JMOI在大多数情况下优于 TransC 和 JOIE 等基准模型,可以处理 isA 关系的传递性。尽管 JMOI 在实验中表现出色,但其仍存在以下局限性:邻域参数的学习依赖超参数调整,可能在小规模数据集上过拟合;自注意力机制增加了模型复杂度,在大规模知识图谱上的扩展性须进一步优化。虽然可利用实体层次结构信息表示知识图谱中的局部特征,但未来将尝试为知识图谱嵌入寻找更多的特征,并探索构建一个符合实验设置的较大的新数据集,以验证所提出的模型的有效性。

## 参 考 文 献

- [1] LIANG K, MENG L, LIU M, et al. A survey of knowledge graph reasoning on graph types: static, dynamic, and multi-modal [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024(12): 46.
- [2] ZHENG Z, XU Y, LIU Y, et al. Distantly supervised relation extraction based on residual attention and self learning[J]. *Neural Processing Letters*, 2024, 56(3): 1-16.
- [3] TIAN L, ZHOU X, WU Y, et al. Knowledge graph and knowledge reasoning: A systematic review[J]. *Journal of Electronic Science and Technology*, 2022, 20(2): 159-186.
- [4] WANG D, ZHOU S, HUANG J, et al. Triple alignment-enhanced complex question answering over knowledge bases[J]. *Neurocomputing*, 2024, 588: 127709.
- [5] LI Y, ZHU C. TransE-MTP: A new representation learning method for knowledge graph embedding with multi-translation principles and transE [J]. *Electronics*, 2024, 13(16): 3171.
- [6] ZHANG Z, GUAN Z, ZHANG F, et al. Weighted knowledge graph embedding[C]//SIGIR'23: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2023: 867-877.
- [7] WANG S, FU K, SUN X, et al. Representation learning of knowledge graphs with the interaction between entity types and relations[J]. *Neurocomputing*, 2022, 508: 305-314.
- [8] WANG P, ZHOU J. JECI++: A modified joint knowledge graph embedding model for concepts and instances[J]. *Big Data Research*, 2020, 24, 100160.
- [9] LYU L, HOU L, LIU Z. Differentiating concepts and instances for knowledge graph embedding[J]. arXiv: 1811. 04588, 2018.
- [10] HAO J, CHEN M, YU W, et al. Universal representation learning of knowledge bases by jointly embedding instances and ontological concepts[C]//KDD: ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2019: 1709-1719.
- [11] BAI Y, WANG W, SUN Y. Dual-Geometric space embedding model for two-view knowledge graphs[C]//KDD '22: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2022: 5033-5041.
- [12] ZHANG P, CHEN D, FANG Y, et al. CIST: Differentiating concepts and instances based on spatial transformation for knowledge graph embedding[J]. *Mathematics*, 2022, 10: 1-18.
- [13] HUANG Z, SUN Y, WANG W, et al. Concept2Box: joint geometric embeddings for learning two-view knowledge graphs [C]//ACL-Findings: Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2023: 10105-10118.
- [14] LI Z, LIU X, WANG X, et al. TransO: a knowledge-driven representation learning method with ontology information constraints[J]. *World Wide Web*, 2023, 26(1): 297-319.
- [15] WANG K, QI G, WU T, et al. Embedding ontologies via incorporating extensional and intensional knowledge[J]. *Data Intelligence*, 2024, 6(2): 175-190.
- [16] MILLER G. WordNet: A lexical database for english[J]. *Communications of the ACM*, 1995, 38: 39-41.
- [17] BORDES A, USUNIER N, GARCIA-DURAN A, et al. Translating embeddings for modeling multi-relational data[C]//NIPS: 26th International Conference on Neural Information Processing Systems. Lake Tahoe, Nevada: MIT Press, 2013: 2787-2795.
- [18] YANG B, YIH W, HE X, et al. Embedding entities and relations for learning and inference in knowledge bases[C]//ICLR: 3rd International Conference on Learning Representations. San Diego, CA: Computational and Biological Learning Society, 2015.
- [19] NICKEL M, ROSASCO L, POGGIO T A. Holographic embeddings of knowledge graphs[J]. arXiv: 1510. 04935, 2015.
- [20] WANG Z, ZHANG J, FENG J, et al. Knowledge graph embedding by translating on hyperplanes[C]//AAAI: Twenty-Eighth AAAI Conference on Artificial Intelligence. AAAI, 2014: 1112-1119.
- [21] LIN Y, LIU Z, SUN M, et al. Learning entity and relation embeddings for knowledge graph completion[C]//AAAI: Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI, 2015: 2181-2187.
- [22] JI G, HE S, XU L, et al. Knowledge graph embedding via dynamic mapping matrix[C]//ACL-IJCNLP: 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. Association for Computational Linguistics, 2015: 687-696.
- [23] CHEN X J, XIANG Y. STransH: A revised translation-based model for knowledge representation [J]. *Computer Science*, 2019, 46(9): 184-189.
- [24] BIN C, QIN S, RAO G, et al. Multiview translation learning for knowledge graph embedding[J]. *Scientific Programming*, 2020, 2020: 1-9.
- [25] NICKEL M, TRESP V, KRIEGEL H P, et al. A three-way mo-

- del for collective learning on multi-relational data[C]//ICML: International Conference on Machine Learning, Bellevue, WA: ICML, 2011; 3104482-3104584.
- [26] LIU Y, YAO Q, LI Y. Generalizing tensor decomposition for n-ary relational knowledge bases[C]//WWW: The Web Conference. ACM, 2020; 1104-1114.
- [27] DI S, YAO Q, CHEN L. Searching to sparsify tensor decomposition for nary relational data[C]//WWW: The Web Conference. ACM, 2021; 4043-4054.
- [28] DETTMERS T, MINERVINI P, STENETORP P, et al. Convolutional 2d knowledge graph embeddings[C]//AAAI: Thirtieth AAAI Conference on Artificial Intelligence. AAAI, 2018.
- [29] NGUYEN D, NGUYEN T, NGUYEN D, et al. A Novel embedding model for knowledge base completion based on convolutional neural network[C]//NAACL-HLT: 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2018; 327-333.
- [30] GUO L, SUN Z, HU W. Learning to exploit long-term relational dependencies in knowledge graphs[J]. arXiv:1905.04914, 2019.
- [31] ZHANG X, ZHANG C, GUO J, et al. Graph attention network with dynamic representation of relations for knowledge graph completion[J]. Expert Systems With Applications, 2023, 219: 119616.
- [32] XU H, BAO J, LIU W. Double-Branch multi-attention based graph neural network for knowledge graph completion[C]//ACL: 61st Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2023; 15257-15271.
- [33] LI X, TIAN Y, JI S. Semantic-and relation-based graph neural network for knowledge graph completion[J]. Applied Intelligence, 2024, 54(8): 6085-6107.
- [34] WANG Z, LI J. Text-Enhanced representation learning for knowledge graph[C]//IJCAI: Twenty-Fifth International Joint Conference on Artificial Intelligence. New York: IJCAI, 2016: 1293-1299.
- [35] GUAN N, SONG D, LIAO L. Knowledge graph embedding with concepts[J]. Knowledge-Based Systems, 2019, 164: 38-44.
- [36] ZHANG J D, LI J. Knowledge graph embedding combining with hierarchical type information[J]. Journal of Software, 2022, 33: 3331-3346.



**QIN Jing**, born in 2000, postgraduate, is a member of CCF (No. Y8902G). Her main research interest is knowledge graph embedding and reasoning.



**LI Guanfeng**, born in 1979, Ph.D, associate professor, is a member of CCF (No. A0519M). His main research interests include knowledge engineering and intelligent computing.

(责任编辑:柯颖)