

融合 DeepE 和对比学习的链路预测模型^①



翁慧敏¹, 郭躬德¹, 林世水^{2,3}

¹(福建师范大学 计算机与网络空间安全学院, 福州 350117)

²(福建医科大学 临床医学部, 福州 350122)

³(福州大学附属省立医院, 福州 350001)

通信作者: 郭躬德, E-mail: ggd@fjnu.edu.cn

摘 要: 现有的知识图谱链路预测方法在学习语义信息的过程中大多只关注单个三元组中头实体 h 、关系 r 和尾实体 t 之间的语义关系, 没有考虑不同三元组中相关实体和实体关系之间的联系. 针对此问题, 本文提出了 DeepE_CL 模型. 首先, 通过 DeepE 模型学习相关三元组的语义信息和具有相同实体关系对的实体或具有相同实体的实体关系对的语义信息. 其次, 利用提取的相关三元组语义信息计算相应的评分函数和交叉熵损失, 并采用对比学习模型对提取的具有相同实体关系对的实体或具有相同实体的实体关系对的语义信息进行优化, 从而实现对相关三元组缺失信息的预测. 本文在 4 个常见的数据集上进行验证, 运用 MR 、 MRR 、 $Hit@1$ 和 $Hit@10$ 这 4 个评价指标对所提方法和其他基线模型进行对比, 实验结果表明 DeepE_CL 模型在所有指标上都取得了最好的结果. 为了进一步验证模型的实用性, 本文还将模型应用到了 1 个真实的中成药数据集, 实验结果显示 DeepE_CL 模型比 DeepE 模型在 MR 指标上降低了 18, 在 MRR 、 $Hit@1$ 指标上分别提升了 0.8%、1.1%, 在 $Hit@10$ 指标上维持不变. 实验证明了引入对比学习模型的 DeepE_CL 模型在提升知识图谱链路预测性能方面的有效性.

关键词: 链路预测; 对比学习; 知识图谱; 中成药数据集

引用格式: 翁慧敏, 郭躬德, 林世水. 融合 DeepE 和对比学习的链路预测模型. 计算机系统应用, 2025, 34(2): 206-215. <http://www.c-s-a.org.cn/1003-3254/9769.html>

Link Prediction Model Integrating DeepE and Contrastive Learning

WENG Hui-Min¹, GUO Gong-De¹, LIN Shi-Shui^{2,3}

¹(College of Computer and Cyber Security, Fujian Normal University, Fuzhou 350117, China)

²(School of Clinical, Fujian Medical University, Fuzhou 350122, China)

³(Fuzhou University Affiliated Provincial Hospital, Fuzhou 350001, China)

Abstract: Most of the existing knowledge graph link prediction methods focus only on the semantic relationships between a head entity h , a relationship r , and a tail entity t in a single triad in learning semantic information. They do not consider the links between related entities and entity relationships in different triads. To address this problem, this study proposes the DeepE_CL model. Firstly, the study uses the DeepE model to learn the semantic information of related triads and entities with the same entity relationship pairs or entity relationship pairs with the same entities. Secondly, the extracted semantic information of the related triads is used to calculate the corresponding scoring function and cross-entropy loss, and the extracted semantic information of entities with the same entity relationship pairs or entity relationship pairs with the same entities is optimized through the comparative learning model, so as to predict the missing information of the related triads. This paper validates the proposed method through four common datasets and compares the proposed method with other baseline models by applying four evaluation indicators, including MR , MRR , $Hit@1$, and $Hit@10$. The experimental results show that the DeepE_CL model achieves the best results in all indicators. To further validate the

① 基金项目: 国家自然科学基金 (62171131); 福建省自然科学基金 (2022J01398)

收稿时间: 2024-07-15; 修改时间: 2024-08-13, 2024-08-27; 采用时间: 2024-09-03; csa 在线出版时间: 2024-12-16

CNKI 网络首发时间: 2024-12-17

usefulness of the model, this study also applies the model to a real traditional Chinese medicine (TCM) dataset, and the experimental results show that compared with the DeepE model, the DeepE_CL model reduces the *MR* indicators by 18, and improves the *MRR*, *Hit@1* indicators by 0.8%, 1.1%, and the *Hit@10* indicators remain unchanged. The experiments demonstrate that the DeepE_CL model, introducing a comparative learning model, is very effective in improving the performance of knowledge graph link prediction.

Key words: link prediction; contrastive learning; knowledge graph; traditional Chinese medicine (TCM) dataset

知识图谱链路预测方法是一种用于预测知识图谱 (knowledge graph, KG) 中缺失事实的方法^[1], 可以有效地对知识图谱进行补全, 并挖掘隐含知识. 知识图谱链路预测方法在推荐系统^[2]、知识问答^[3]、类型检测^[2]等领域有着广泛的应用. KG 中包含大量的事实三元组 (h, r, t) , 其中 h 和 t 分别代表头实体和尾实体, r 代表的是实体间的关系. 知识图谱链路预测的核心思想是将已知实体和关系映射到向量空间中, 以保留三元组的语义信息, 然后计算相应的评分函数来预测缺失事实成立的可信度. 然而目前的知识图谱链路预测方法在学习其语义信息的过程中大多只关注 h 、 r 和 t 之

间的语义关系, 而忽略了不同三元组中相关实体和实体关系之间的联系. 例如, 图 1(a) 中, (杭州, 属于, 中国) 和 (上海, 属于, 中国) 共享同一个尾实体对 (属于, 中国), 所以杭州和上海应该具有相似的语义向量. 图 1(c) 中, (北京, 首都, 中国) 和 (杭州, 属于, 中国) 共享尾实体中国, 那么 (北京, 首都) 和 (杭州, 属于) 应该具有相似的语义向量. 在知识图谱中, 具有相同实体关系对的实体以及具有相同实体的实体-关系对应该具有相似的语义向量即语义相似性, 捕获这些实体或实体对的语义相似性可以更全面地考虑不同三元组之间的关联性, 以提高模型学习的能力.

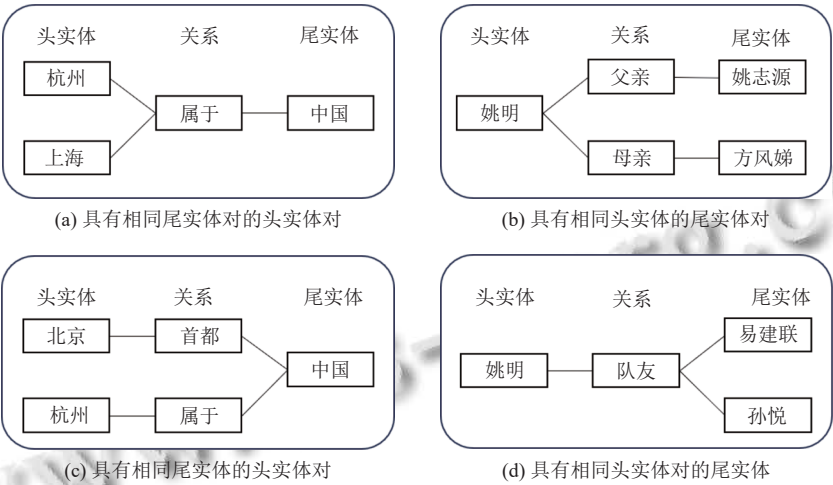


图 1 具有相同实体或实体-关系对的三元组示例

针对上述问题, 本文将 DeepE^[4]模型和对比学习模型相结合提出了 DeepE_CL 模型, 该模型不仅保留了 DeepE 模型在处理简单知识图谱时的优势, 还通过引入对比学习模型, 增强了模型对复杂关系的学习能力, 使其在面对知识图谱中的多样性和复杂性时, 能够更加精准地进行链路预测. DeepE_CL 模型首先利用 DeepE 模型提取头实体 h 和关系 r 之间的语义信息以及具有相同实体关系对的实体或者具有相同实体的实体-关系对的语义信息. 随后, 采用有监督对比学习模

型优化 DeepE 模型中提取的语义信息, 确保正样本对的特征尽可能相似, 负样本对的特征尽可能不同. 最后, 在 4 个公共数据集和中成药数据集上与众多知识图谱链路预测基线模型进行实验对比, 实验结果表明: 本文所提的知识图谱链路预测方法能够取得较好预测效果.

1 相关工作

1.1 知识图谱链路预测

现有的知识图谱链路预测方法通常分类为以下

3类: 基于平移距离的方法、基于语义匹配的方法以及基于神经网络的方法。

基于平移距离的链路预测方法通过计算两个实体在向量空间中的距离来衡量它们之间的相关性, 该方法虽然在计算大规模知识图谱的时候计算效率较高, 但是为每个实体和关系使用单一的向量表示, 这不足以捕捉其不同上下文中的多样性。例如, TransE^[5]模型通过平移的向量 r 连接 h 和 t , 来预测它们之间的误差, 但它难以对复杂关系进行建模。Wang 等人^[6]引入了超平面机制, 提出了 TransH 模型。RotatE^[7]模型则将关系定义为复数空间中头实体到尾实体的旋转。此外, 许多研究者还将上述模型的思想应用于动态的知识图谱, 并在链路预测中提出了相应的嵌入模型^[8,9]。

基于语义匹配的链路预测方法通过评估实体和关系的语义向量来预测三元组成立的可能性, 该方法能较好地理解和预测实体间复杂的语义关系, 具有较好的泛化性, 但是该方法对数据质量和计算成本要求较高。例如, Rescal 模型^[10]通过优化一个多维的三阶张量来学习实体和关系的语义向量, 但在大型知识图谱上, Rescal 的计算效率显著下降。DistMult^[11]和 HolE^[12]在 Rescal 基础上针对该问题进行了改进。ComplEx^[13]模型在复数空间保留点积的计算优势, 使得空间和时间复杂度为线性关系。

基于神经网络的链路预测方法能够增强模型学习深层次语义特征的能力, 但是训练时间较长, 存在过拟合的风险。ConvE^[14]将 h 和 r 的语义特征向量拼接成一个二维矩阵, 然后通过卷积神经网络提取语义向量并将该向量与 t 的语义向量进行评分函数的计算。InteractE^[15]模型解决 ConvE 模型捕获实体和关系的语义交互限制的问题。Zhu 等人^[4]提出了 DeepE 模型, 该模型借鉴了 ResNet 模型, 使得深度函数学习深层特征, 而浅层函数则学习浅层特征。Zhou 等人^[16]针对多重卷积层的非线性特性在一定程度上可能导致表层知识的流失的问题提出了 JointE 模型, 该模型降低参数、提升模型效率。同时, 为了增强知识图谱链路预测的准确性, LMKE^[17]模型和 LP-BERT^[18]模型利用 BERT 作为编码器对文本进行编码, 为模型预测提供了丰富的上下文信息。

上述模型都是通过评分函数来预测 h 、 r 、 t 之间的关系, 虽然可以学习到较强的表示向量并且都取得了不错的成绩但是它们都忽略了不同三元组中相关实体和实体关系对之间的关系。

1.2 对比学习

对比学习使得正样本对之间的特征表示尽可能接近, 负样本对之间的特征表示尽可能不同。它的核心问题是正负例的设计以及损失函数的设计。

近年来, 对比学习方法在计算机视觉领域和自然语言处理任务中得到了广泛的应用, 在计算机视觉领域中通过放大、缩小以及旋转图片来构造正负例, SimCLR^[19]通过对同一图片进行颜色变换、高斯模糊、裁剪等方法生成互为正例对的图片。相反, 不同图片则构成互为负例对。Kalantidis 等人^[20]使用 mix-up 方法获得模型难以识别的负样本。在自然语言处理任务中通过加词、漏词、换位和乱序等方式构建正负例。Yang 等人^[21]采用省略单词数量、词频和词性的方法构造正样本。Kachuee 等人^[22]在每个训练批次中以洗牌的方式生成负样本。

损失函数是用来评估模型预测值和真实值之间的不一致程度。目前大多数对比学习模型都是采用交叉熵损失函数来计算正负样本的相似度。以 SimCLR^[19]为例, 其损失函数如下:

$$CL_{\text{SimCLR}} = -\log \frac{\exp(\text{sim}(z_i, z_i^+)/\tau)}{\sum_{a \in A(i)} \exp(\text{sim}(z_i, z_a^+)/\tau)} \quad (1)$$

该方法通过计算正样本正确分类的类别交叉熵作为损失函数来控制判别效果。其中 z_i^+ 是 z_i 的正样本, $A(i)$ 为 mini-batch 中所有负样本集合, τ 为对比损失温度系数。然而, 当已知多个样本属于同一类时, 式 (1) 将无法有效处理这种情况。因此, Khosla 等人^[23]提出了有监督对比学习, 使得模型在学习过程中能够有效地利用标签信息, 该模型能够将属于同一类别的样本在嵌入空间中拉到一起, 同时将来自不同类别的样本分开。有监督对比损失函数^[23]公式如下:

$$CL_{\text{Sup}} = \frac{-1}{|P(i)|} \sum_{z_i^+ \in P(i)} \log \frac{\exp(\text{sim}(z_i, z_i^+)/\tau)}{\sum_{a \in A(i)} \exp(\text{sim}(z_i, z_a)/\tau)} \quad (2)$$

其中, $P(i)$ 是 mini-batch 中所有正例的集合。

2 DeepE_CL 方法与模型

2.1 DeepE 模型介绍

DeepE 模型具有在学习深层特征的同时不丢失浅层特征信息的特点。该模型由特征提取网络 (feature

extraction network) 和投影网络 (project network) 两部分组成, 整体框架如图2所示。

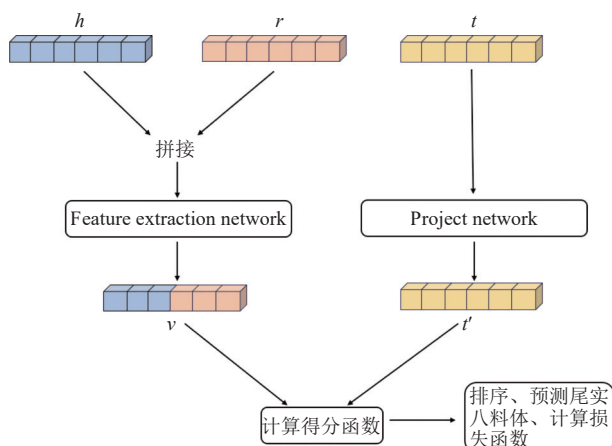


图2 DeepE 模型框架图

2.1.1 特征提取网络

特征提取网络首先拼接 h 和 r , 然后将得到的向量作为多个堆叠的 DeepE 构建块的输入, 构建块返回一个特征向量 $v \in R_d$. 每个 DeepE 构建块由 1 个线性函数和 1 个非线性函数进行加法得到, DeepE 构建块如图3(a)所示, x 和 $f(x)$ 为 DeepE 的输入和输出向量, 形式上每个 DeepE 构建块的定义如下:

$$f(x) = x + w_2 \sigma(w_1 x) \quad (3)$$

其中, w_1 和 w_2 是权重矩阵, σ 是非线性函数, 采用 ReLU 作为激活函数。

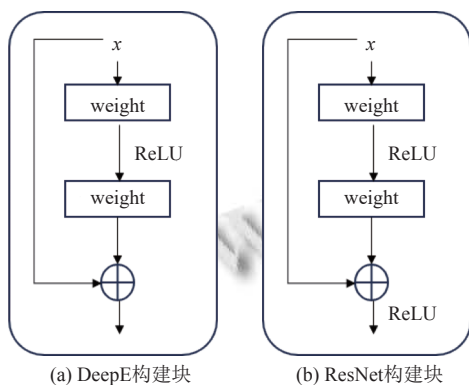


图3 构建块

2.1.2 投影网络

投影网络将 t 投影到与 v 接近的空间, t 作为一个堆叠的 ResNet 构建块的输入, 并返回一个向量 t' , ResNet 构建块如图3(b)所示。

2.1.3 评分函数

该模型的评分函数是 v 和 t' 之间的点积。评分函数

如下:

$$\phi(h, r, t) = f(h||r) \cdot g(t) = v \cdot t' \quad (4)$$

其中, $||$ 为级联算子, f 和 g 分别为 feature extraction network 和 project network 模块。

2.2 对比学习方法

2.2.1 正负例的选择

在对比学习中正负例的构建是十分关键的, 然而在计算机视觉领域和自然语言处理领域中构建正负例的方法并不适用于知识图谱。因此, 本文采用以下方法来构造 KG 的正例^[24]。对于一个三元组 (h, r, t) , 本文定义 h 的正例 h^+ 是与 h 具有相同关系和尾实体的头实体, t 的正例 t^+ 是与 t 具有相同头实体和关系的尾实体。对于 (h, r) 对应的正例是 $(h, r)^+$, (t, r) 对应的正例是 $(t, r)^+$ 。因此, 给定一个三元组 (h, r, t) , 本文构造 4 种正例: h^+ , t^+ , $(h, r)^+$ 和 $(t, r)^+$, 分别对应图1中的 4 个实例。对于实例 h , 本文将其所有正例 h^+ 用在同一个 mini-batch 中。此外, 对于在一个小批量中没有正例的三元组, 本文将为每个三元组添加一个正例, 该正例从训练集中随机采样并且本文将不属于同一类别的样本作为对比学习的负例对。

2.2.2 对比损失函数

对于给定的三元组 (h, r, t) , 本文使用式(2)来计算 4 类正例的对比损失, 因此三元组 (h, r, t) 的整体对比损失函数^[23]如下:

$$L_{cl}(h, r, t) = CL_{Sup}(h, h^+) + CL_{Sup}(t, t^+) + CL_{Sup}((h, r), (h, r)^+) + CL_{Sup}((t, r), (t, r)^+) \quad (5)$$

式(5)由 4 个对比损失项组成, 在实验过程中发现不同的对比损失项对不同的知识图谱有不同的影响。出现这种情况的原因可能是因为不同的知识图谱具有不同的实体数量与关系数量。因此, 本文针对特定的 KG 分配了一个权重超参数 α 。加入权重 α 后的对比损失函数^[22]如下:

$$L_{cl}^w(h, r, t) = \alpha_h CL_{Sup}(h, h^+) + \alpha_t CL_{Sup}(t, t^+) + \alpha_{hr} CL_{Sup}((h, r), (h, r)^+) + \alpha_{tr} CL_{Sup}((t, r), (t, r)^+) \quad (6)$$

2.3 DeepE_CL 模型整体框架

本文在 DeepE 模型的基础上引入有监督对比学习模型, 以提高模型提取实体和关系语义信息的能力。DeepE_CL 整体框架如图4所示。首先, 利用 feature

extraction network 提取 h 和 r 的深层语义信息 v 并使用 project network 提取 t 特征向量 t' . 接着, 根据提取的语义信息 v 和特征向量 t' 计算相应的评分函数和交叉熵损失 L_s . 然后, 将相同实体关系对的实体或具有相同实体的实体-关系对输入 feature extraction network

提取深层语义信息 v' , 并采用式 (6) 权重对比损失函数对 DeepE 模型中提取的语义信息 v 和 v' 进行优化并得到损失函数 $L_{cl}^w(h, r, t)$. 最后, 得到总训练目标 $L_{all} = L_s + L_{cl}^w(h, r, t)$, 并利用反向传播算法和 Adam 优化器更新模型中的参数.

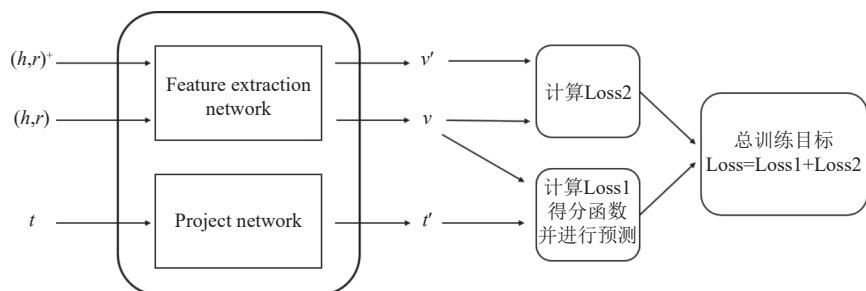


图4 DeepE_CL 模型整体框架图

2.4 DeepE_CL 时间复杂度分析

DeepE_CL 模块的时间复杂度主要有 3 部分.

(1) feature extraction network 中 DeepE 构建块中刚开始头实体特征向量和尾实体特征向量进行拼接导致输入维度不等于输出维度的, 因此执行线性映射, 时间复杂度 $O(Im)$, 每个构建块顺序执行每个线性层、批归一化层、Dropout 层和激活函数, 时间复杂度为 $O(km^2)$, 因此该模块总体的时间复杂度为 $O(Im + km^2)$, 其中 I 是输入维度, m 是输出维度, k 是 DeepE 构建块的个数.

(2) project network 中 ResNet 构建块, 每个构建块顺序执行每个线性层、批归一化层、Dropout 层和激活函数, 时间复杂度为 $O(tm^2)$, t 是 ResNet 构建块的个数.

(3) 对比损失函数模块, 该模块中相似度计算的时间复杂度为 $O((2n)^2 \cdot I)$, 正负样本计算的时间复杂度为 $O((2n)^2)$, 该模块总的时间复杂度为 $O(n^2I)$, n 是 batch_size 的大小, I 是输入维度的大小.

因此, DeepE_CL 模型的时间复杂度为 $O(Im + km^2 + n^2I)$.

3 实验

3.1 数据集

本文使用 FB15k-237、WN18RR、Kinship 和 UMLS 这 4 个常见的基准数据集评估 DeepE_CL 模型在知识图谱链路预测中的实验性能. 其中 FB15k-237 数据集是 Freebase 的子集, 描述了知识事实的知识图

谱. WN18RR 数据集是 WordNet 的子集, 描述英语单词之间关联特征的数据集. Kinship 数据集是亲属关系数据集, UMLS 数据集是医疗数据集, 数据集的统计信息如表 1 所示.

表1 数据集统计信息

数据类型	FB15k-237	WN18RR	Kinship	UMLS
#实体	14541	40943	104	135
#关系	237	11	25	46
#训练集	272 115	86 835	8 544	5 216
#验证集	17 535	3 034	1 068	652
#测试集	20 446	3 134	1 074	661

3.2 基线模型

本文将 DeepE_CL 模型与非线性模型 DistMult^[11] 和 RotatE^[7], 神经网络模型 ConvE^[14]、HypE^[25]、InteractE^[15]、AcrE^[26] 和 JointE^[16], 以及采用多层感知器模型 ER-MLP2d^[27] 和 DeepE^[2] 进行比较以表明本文提出的 DeepE_CL 模型在提升知识图谱链路预测性能方面是有效的.

3.3 评价指标

本文采用以下 4 个常用指标来对性能进行评价.

(1) MR 是平均排名, 具体的计算公式如下:

$$MR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} rank_i \quad (7)$$

其中, Q 表示三元组, $|Q|$ 表示三元组集合的个数, $rank_i$ 表示第 i 个三元组链路预测的排名, MR 值越小越好.

(2) MRR 是平均倒数排名, 它与 MR 相反, MRR 值越大越好, 具体计算公式如下:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \quad (8)$$

(3) $Hit@N$ 指的是在链路预测中排名小于等于 N 的三元组的平均占比, 具体计算公式如下:

$$Hit@N = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \prod (rank_i \leq N) \quad (9)$$

其中, $\prod(\cdot)$ 是 indicator 函数 (若条件真则函数值为 1, 否则为 0)。一般地, 取 n 为 1、3 或者 10, $Hit@N$ 指标越大越好。

3.4 实验设置

本文基于 PyTorch 库实现了 DeepE_CL 模型, 并在单个 NVIDIA GeForce RTX 3090 GPU 上运行了所有实验。此外, 为保证实验公平性, FB15k-237 和 WN18RR 数据集中与 DeepE 模型相关的参数均与文献[4]一致, 训练过程中均使用 Adam 优化器更新模型的参数并且当 MRR 在验证集上连续 10 个历元没有提高时, 训练将终止。当在验证集上取得最佳 MRR 时, 选择最佳模型。本文使用网格搜索来搜索各个数据集的最佳参数并报告了 5 次不同随机初始化的平均结果。DeepE_CL 模型包含以下参数, 其中 DeepE 模型的参数具体如下: d 表示模型隐藏层的嵌入维度, b 表示 batch_size 的大小, $l2$ 表示 L2 正则化, ns 表示 DeepE 构建块的数量, nt 表示 ResNet 构建块的数量, ni 表示 ResNet 构建块内部层数, hd 表示 DeepE 构建块中输入层和全连接层的

丢弃率, id 表示 DeepE 构建模块中的丢弃率, td 表示 ResNet 构建块中输入层和全连接层的丢弃率; 有监督对比损失模型的参数具体如下: t 表示温度, α_h 、 α_t 、 α_{hr} 、 α_{tr} 分别表示模型中各自的权重参数。

在实验过程中由于不同的对比损失项对不同的知识图谱有不同的影响, 如表 2 和表 3 所示, 其中加粗代表最高分。在 FB15k-237 数据集上, 使用 $CL_{Sup}(h, h^+)$ 取得了最高的预测结果, 而在 WN18RR 数据集上, 使用 $CL_{Sup}((h, r), (h, r)^+)$ 取得了最高的预测结果, 在 Kinship 和 UMLS 数据集上使用 $CL_{Sup}(t, t^+)$ 取得了最高的预测结果, 出现这种现象可能是因为不同的知识图谱具有不同的属性, 如实体数量与关系数量以及三元组数量的不同等。

根据表 2 和表 3 的结果, 本文选择对应预测结果最高的对比损失项进行实验, 每个数据集具体的实验参数设置如表 4 所示, 其中 FB15k-237 数据集 $\alpha_h = 2$, α_t 、 α_{hr} 和 α_{tr} 为 0, 代表 FB15k-237 数据集选择 $CL_{Sup}(h, h^+)$ 作为对比选择项, 选择项的权重参数为 2; WN18RR 数据集 $\alpha_{hr} = 2$, α_h 、 α_t 和 α_{tr} 为 0, 代表 WN18RR 数据集选择 $CL_{Sup}((h, r), (h, r)^+)$ 作为对比选择项, 选择项的权重参数为 2; Kinship 和 UMLS 数据集 $\alpha_t = 2$, α_h 、 α_{hr} 和 α_{tr} 为 0, 代表 Kinship 和 UMLS 数据集选择 $CL_{Sup}(t, t^+)$ 作为对比选择项, 选择项的权重参数为 2。

表 2 仅使用一个特定的对比损失项的 DeepE_CL 的链路预测结果 1

对比损失项	FB15k-237				WN18RR			
	MR	MRR	Hit@1	Hit@10	MR	MRR	Hit@1	Hit@10
$CL_{Sup}(h, h^+)$	154	0.364	0.271	0.55	2498	0.487	0.444	0.571
$CL_{Sup}(t, t^+)$	153	0.360	0.267	0.546	1942	0.488	0.445	0.569
$CL_{Sup}((h, r), (h, r)^+)$	161	0.354	0.261	0.540	1577	0.489	0.445	0.574
$CL_{Sup}((t, r), (t, r)^+)$	162	0.350	0.256	0.539	1689	0.482	0.438	0.567

表 3 仅使用一个特定的对比损失项的 DeepE_CL 的链路预测结果 2

对比损失项	Kinship				UMLS			
	MR	MRR	Hit@1	Hit@10	MR	MRR	Hit@1	Hit@10
$CL_{Sup}(h, h^+)$	1.74	0.882	0.813	0.988	1.17	0.956	0.924	0.999
$CL_{Sup}(t, t^+)$	1.73	0.884	0.816	0.990	1.18	0.958	0.930	0.998
$CL_{Sup}((h, r), (h, r)^+)$	1.88	0.879	0.810	0.987	1.19	0.953	0.919	0.998
$CL_{Sup}((t, r), (t, r)^+)$	1.83	0.873	0.801	0.987	1.15	0.954	0.921	0.998

3.5 实验结果与分析

3.5.1 训练过程分析实验

图 5 为本文模型在训练过程中训练损失的变化情况。本文采用 Adam 优化器并设置了权重衰退 weight_

decay 对模型进行了优化, 图像中横坐标表示训练次数, 纵坐标表示损失值。可以看出, 本文模型在训练过程中的损失值在下降, 并且 FB15k-237 和 WN18RR 数据集在 400 epoch 左右趋于收敛, Kinship 数据集在 200

epoch 左右趋于收敛, UMLS 在 100 epoch 左右趋于收敛.

表 4 参数设置

参数简称	FB15k-237	WN18RR	Kinship	UMLS
<i>d</i>	300	250	300	250
<i>b</i>	512	512	512	512
<i>l2</i>	5E-08	5E-05	5E-03	5E-05
<i>ns</i>	40	1	10	1
<i>nt</i>	1	2	1	1
<i>ni</i>	2	3	2	2
<i>hd</i>	0.4	0.4	0.4	0.4
<i>id</i>	0.01	0	0.01	0.01
<i>td</i>	0.4	0	0	0
<i>t</i>	0.1	0.1	0.5	0.9
α_h	2	0	0	0
α_t	0	0	2	2
α_{hr}	0	2	0	0
α_{tr}	0	0	0	0

3.5.2 对比实验

为验证在 DeepE 模型上引入有监督对比学习模型

能够提高模型提取实体和关系的语义信息. 将以上 4 个数据集分别放入基线模型和本文提出的 DeepE_CL 模型进行实验, 实验结果如表 5 和表 6 所示, 其中加粗代表最高分, 下划线代表次高分. 本文除 DeepE 和 DeepE_CL, FB15k-237 和 WN18RR 的其余模型的实验结果来源于文献[4], Kinship 和 UMLS 的其余模型的实验结果来源于文献[28].

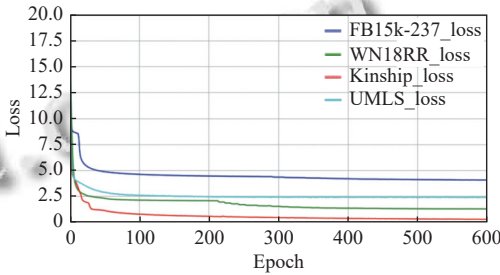


图 5 损失曲线

表 5 FB15k-237 和 WN18RR 对比实验

Model	FB15k-237				WN18RR			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
Dismult	254	0.241	0.155	0.419	5 110	0.43	0.39	0.49
RotatE	177	0.338	0.241	0.533	3 340	0.476	0.428	0.571
ConvE	244	0.325	0.237	0.501	4 187	0.430	0.400	0.520
HypER	250	0.341	0.252	0.520	5 798	0.465	0.436	0.522
InteractE	172	0.354	0.263	0.535	5 202	0.463	0.430	0.528
AcrE	—	<u>0.358</u>	<u>0.266</u>	0.545	—	0.459	0.422	0.532
JointE	177	0.356	0.262	0.543	4 655	0.471	<u>0.438</u>	0.537
ER-MLP2d	234	0.338	—	<u>0.547</u>	4 233	0.358	—	0.421
DeepE	<u>167</u>	0.354	0.263	0.536	<u>2 560</u>	<u>0.479</u>	<u>0.438</u>	<u>0.550</u>
DeepE_CL	154	0.364	0.271	0.55	1 577	0.489	0.445	0.574

表 6 Kinship 和 UMLS 对比实验

Model	Kinship				UMLS			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
Dismult	7.90	0.48	0.377	0.708	18.8	0.164	0.061	0.403
RotatE	2.90	0.738	0.617	0.954	<u>2.10</u>	0.822	0.703	0.969
ConvE	3.00	0.772	0.665	0.950	3.20	0.836	0.764	0.946
HypER	1.96	0.868	0.790	0.981	—	—	—	—
DeepE	<u>1.92</u>	<u>0.873</u>	<u>0.800</u>	<u>0.988</u>	1.18	<u>0.952</u>	<u>0.919</u>	<u>0.997</u>
DeepE_CL	1.73	0.884	0.816	0.99	1.18	0.958	0.930	0.998

本文首先在 FB15k-237 和 WN18RR 上评估 DeepE_CL 模型. 以非线性模型 (DistMult、RotatE) 和卷积神经网络模型 (ConvE、HypER、InteractE、AcrE、JointE) 以及多层感知器模型 (ER-MLP2d、DeepE) 为基准, 结果如表 5 所示. 总体而言, DeepE_CL 模型在两个数据集上的所有指标上都取得了相当有竞争力的表现. 从实验结果表 5 可以看出, 与非线性模型相比较,

DeepE_CL 模型在 FB15k-237 和 WN18RR 数据集上 *MR* 指标分别降低了 23–100、1763–3533; *MRR* 指标分别提升了 2.6%–12.3%、1.3%–5.9%; *Hit@1* 指标分别提升了 0.3%–1.16%、1.7%–5.5%; *Hit@10* 分别提升了 1.7%–13.1%、0.3%–8.4%. 与基于卷积神经网络模型的基线进行比较, DeepE_CL 模型在 FB15k-237 和 WN18RR 数据集上 *MR* 指标分别降低了 18–96、2610–

4221; *MRR* 指标分别提升了 0.6%–3.9%、5.1%–5.9%; *Hit@1* 指标分别提升了 0.5%–3.4%、0.7%–4.5%; *Hit@10* 分别提升了 0.5%–4.9%、3.7%–5.4%。与多层感知机模型的基线进行了比较, DeepE_CL 模型在 FB15k-237 和 WN18RR 数据集上 *MR* 指标分别降低了 13–83、98–2656; *MRR* 指标分别提升了 1%–2.6%、1%–13.1%; *Hit@1* 指标分别提升了 0.8%、0.7%; *Hit@10* 分别提升了 0.3%–1.4%、2.4%–15.3%。特别地, 与基于多层感知机的 DeepE 模型相比较, FB15k-237 和 WN18RR 在 DeepE_CL 模型上所有指标都取得了较好的结果。

此外, 本文还在 Kinship、UMLS 这两个小型数据集上评估 DeepE_CL 模型。表 6 列出了几种经典的知识图谱链路预测模型, 如 DistMult、RotatE、ConvE、HypER、DeepE。不出所料, DeepE_CL 模型在 *MR*、*MRR*、*Hit@3* 和 *Hit@10* 方面都优于其他基线模型。

综上所述, DeepE_CL 模型在 4 个基准数据集上几乎优于所有基线模型。也就是说, 无论数据集的大小和复杂程度如何, DeepE_CL 模型都能达到最好的性能, 这充分显示了 DeepE_CL 的鲁棒性和良好的泛化能力。

3.5.3 消融实验

为了验证本文所提出模型的有效性, 在 DeepE 模型上增加对比损失函数进行消融实验, w/o loss 表示在计算过程中不加入对比损失函数, w/o 表示计算过程中加入对比损失函数。消融实验结果如表 7 和表 8 所示, 实验结果显示在 4 个数据集上引入对比损失函数后, 模型的性能得到了提升, 在 *MRR*、*Hit@1* 和 *Hit@10* 指标分别提升了 0.6%–1.1%、0.7%–1.6%、0.1%–2.4%, 验证了加入对比损失函数可以提高模型学习实体和关系的语义信息, 也证实了模型的有效性。

表 7 FB15k-237 和 WN18RR 模型消融实验

数据集	FB15k-237				WN18RR			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
w/o loss	167	0.354	0.263	0.536	2560	0.479	0.438	0.550
w/o	154	0.364	0.271	0.550	1577	0.489	0.445	0.574

表 8 Kinship 和 UMLS 模型消融实验

数据集	Kinship				UMLS			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
w/o loss	1.92	0.873	0.800	0.988	1.18	0.952	0.919	0.997
w/o	1.73	0.884	0.816	0.990	1.18	0.958	0.930	0.998

为了验证在对比损失函数中正例对模型的影响, 本文在 4 个数据集上进行了消融实验, 其中 w/o pos 表

示在计算对比损失时不加入正例即在一个小批量中删除了所有原始的和额外添加的正例, w/o 表示在计算对比损失时加入正例。为保证实验结果公平性, 保持各模型参数一致。消融实验结果如表 9 和表 10 所示, 实验结果显示在 4 个数据集上计算对比损失时加入正例的所有指标都取得了较好的实验结果。因此, 在对比损失中引入正例是有效的, 它能够提高 DeepE_CL 的性能。

表 9 FB15k-237 和 WN18RR 对比损失函数中引入正例消融实验

数据集	FB15k-237				WN18RR			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
w/o pos	166	0.357	0.265	0.541	2250	0.484	0.443	0.562
w/o	154	0.364	0.271	0.550	1577	0.489	0.445	0.574

表 10 Kinship 和 UMLS 对比损失函数中引入正例消融实验

数据集	Kinship				UMLS			
	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>	<i>MR</i>	<i>MRR</i>	<i>Hit@1</i>	<i>Hit@10</i>
w/o pos	1.82	0.876	0.805	0.989	1.17	0.954	0.921	0.999
w/o	1.73	0.884	0.816	0.990	1.18	0.958	0.930	0.998

4 DeepE_CL 模型在中成药数据集中的应用

4.1 数据集的来源

中成药作为我国传统医学的重要组成部分, 蕴含着丰富的知识和智慧。知识图谱链路预测模型作为一种强大的数据挖掘工具, 能够在复杂的中成药数据集发现潜在的联系和规律。本文采用的中成药数据集来源于文献[29]里的新药方剂, 它来源于中成药医局, 共包含 17045 个实体和 6 个关系。训练集、验证集和测试集的个数分别是 118314、14790、14789。

该数据集实验设置和第 3.4 节实验设置一致, 进行网格搜索后可得该数据集最佳参数设置如下: $d = 500$, $b = 512$, $l2 = 5E-05$, $ns = 2$, $nt = 2$, $ni = 2$, $hd = 0.4$, $id = 0.01$, $td = 0$, $t = 0.9$, $\alpha_n = 0$, $\alpha_t = 0$, $\alpha_{hr} = 1$, $\alpha_{tr} = 0$ 。

4.2 实验结果

本文将 DeepE_CL 模型应用在中成药数据集上进行药物重定位, 并以 RotatE、Dismult、InteractE 和 DeepE 这 4 个模型为基准进行对比, 实验结果如表 11 所示。本文除 DeepE_CL 模型外其余模型的实验数据来源于文献[30]。

从实验结果可以看出, 在中成药数据集中 DeepE_CL 模型在所有指标上都取得了最好的结果。与基线模型相比较, 该数据集在 DeepE_CL 模型上 *MR* 指标降

低了18–130; MRR 、 $Hit@1$ 和 $Hit@10$ 指标分别提升了0.8%–10.9%、1.1%–14.1%、0–4.6%。特别地,与DeepE模型相比,该数据集在DeepE_CL模型上 MR 指标降低了18; MRR 、 $Hit@1$ 指标分别提升了0.8%、1.1%; $Hit@10$ 维持不变。实验结果证明了DeepE_CL模型在中成药数据集的应用是有效的。

表 11 中成药数据集对比实验结果

指标	MR	MRR	$Hit@1$	$Hit@10$
RotatE	273	0.739	0.694	0.823
Dismult	273	0.803	0.771	0.861
InteractE	200	0.803	0.776	0.851
DeepE	161	0.840	0.824	0.869
DeepE_CL	143	0.848	0.835	0.869

5 结论与展望

本文提出了一种融合DeepE和对比学习的知识图谱链路预测模型,该模型有助于弥补现有知识图谱链路预测方法在学习语义信息时忽略不同三元组中相关实体和实体关系之间联系的问题。该模型利用DeepE模型学习三元组的语义信息,并对具有相同实体关系对的实体或具有相同实体的实体关系对的语义信息进行学习,使得相关的语义信息能够更好地用低维向量表示其主要特征。同时设计了一种加权对比损失函数进行优化,使其在训练过程中考虑不同三元组中实体和关系的相似性。本文所提模型在常用的数据集和中成药数据集上进行了对比分析,证明了本文提出的模型能够提升知识图谱链路预测的学习效果。

尽管DeepE_CL模型在实验中表现优异,但它在某些方面仍有改进空间。一个问题是,在计算对比损失函数时,模型需要构建一个样本间的相似度矩阵,这在处理大规模数据集时会导致模型的计算复杂性增加。因此,在未来的研究过程中将探索更先进的对比学习方法在知识图谱链路预测中的应用,同时也将关注模型的复杂度问题,寻求降低计算成本并提高模型训练速度的有效策略。

参考文献

- Hilman D, Şerban O. A unified link prediction architecture applied on a novel heterogenous knowledge base. *Knowledge-based Systems*, 2022, 241: 108228. [doi: [10.1016/j.knsys.2022.108228](https://doi.org/10.1016/j.knsys.2022.108228)]
- 封皓君, 段立, 张碧莹. 面向知识图谱的知识推理综述. *计算机系统应用*, 2021, 30(10): 21–30. [doi: [10.15888/j.cnki.csa.008137](https://doi.org/10.15888/j.cnki.csa.008137)]
- 张栩翔, 马华. 知识图谱与图嵌入在个性化教育中的应用综述. *计算机系统应用*, 2022, 31(3): 48–55. [doi: [10.15888/j.cnki.csa.008377](https://doi.org/10.15888/j.cnki.csa.008377)]
- Zhu DH, Shen S, Huang SJ, *et al.* DeepE: A deep neural network for knowledge graph embedding. *arXiv:2211.04620*, 2022.
- Bordes A, Usunier N, Garcia-Durán A, *et al.* Translating embeddings for modeling multi-relational data. *Proceedings of the 26th International Conference on Neural Information Processing Systems*. Lake Tahoe: Curran Associates Inc., 2013. 2787–2795.
- Wang Z, Zhang JW, Feng JL, *et al.* Knowledge graph embedding by translating on hyperplanes. *Proceedings of the 28th AAAI Conference on Artificial Intelligence*. Québec City: AAAI, 2014. 1112–1119.
- Sun ZQ, Deng ZH, Nie JY, *et al.* RotatE: Knowledge graph embedding by relational rotation in complex space. *Proceedings of the 7th International Conference on Learning Representations*. New Orleans: OpenReview.net, 2019.
- Sadeghian A, Armandpour M, Colas A, *et al.* ChronoR: Rotation based temporal knowledge graph embedding. *Proceedings of the 35th AAAI Conference on Artificial Intelligence*. Washington: AAAI, 2014. 6471–6479.
- Dasgupta SS, Ray SN, Talukdar P. HyTE: Hyperplane-based temporally aware knowledge graph embedding. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels: Association for Computational Linguistics, 2018. 2001–2011.
- Riedel S, Yao LM, McCallum A, *et al.* Relation extraction with matrix factorization and universal schemas. *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta: Association for Computational Linguistics, 2013. 74–84.
- Yang BS, Yih WT, He XD, *et al.* Embedding entities and relations for learning and inference in knowledge bases. *Proceedings of the 3rd International Conference on Learning Representations*. San Diego: OpenReview.net, 2015.
- Nickel M, Rosasco L, Poggio T. Holographic embeddings of knowledge graphs. *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. Phoenix: AAAI, 2016. 1955–1961.
- Trouillon T, Welbl J, Riedel S, *et al.* Complex embeddings for simple link prediction. *Proceedings of the 33rd*

- International Conference on Machine Learning. New York City: PMLR, 2016. 2071–2080.
- 14 Dettmers T, Minervini P, Stenetorp P, *et al.* Convolutional 2D knowledge graph embeddings. Proceedings of the 32nd AAAI Conference on Artificial Intelligence. New Orleans: AAAI, 2018. 1811–1818.
- 15 Vashishth S, Sanyal S, Nitin V, *et al.* InteractE: Improving convolution-based knowledge graph embeddings by increasing feature interactions. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI, 2020. 3009–3016.
- 16 Zhou ZH, Wang C, Feng Y, *et al.* JointE: Jointly utilizing 1D and 2D convolution for knowledge graph embedding. Knowledge-based Systems, 2022, 240: 108100. [doi: 10.1016/j.knsys.2021.108100]
- 17 Wang XT, He QY, Liang JQ, *et al.* Language models as knowledge embeddings. Proceedings of the 31st International Joint Conference on Artificial Intelligence. Vienna: IJCAI, 2022. 2291–2297.
- 18 Li D, Zhu BQ, Yang S, *et al.* Multi-task pre-training language model for semantic network completion. ACM Transactions on Asian and Low-resource Language Information Processing, 2023, 22(11): 250.
- 19 Chen T, Kornblith S, Norouzi M, *et al.* A simple framework for contrastive learning of visual representations. Proceedings of the 37th International Conference on Machine Learning. Baltimore: PMLR, 2020. 1597–1607.
- 20 Kalantidis Y, Sariyildiz MB, Pion N, *et al.* Hard negative mixing for contrastive learning. Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 1829.
- 21 Yang ZH, Cheng Y, Liu Y, *et al.* Reducing word omission errors in neural machine translation: A contrastive learning approach. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence: Association for Computational Linguistics, 2019. 6191–6196.
- 22 Kachuee M, Yuan H, Kim YB, *et al.* Self-supervised contrastive learning for efficient user satisfaction prediction in conversational agents. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2020. 4053–4064.
- 23 Khosla P, Teterwak P, Wang C, *et al.* Supervised contrastive learning. Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2020. 1567.
- 24 Luo ZP, Xu WT, Liu WQ, *et al.* KGE-CL: Contrastive learning of tensor decomposition based knowledge graph embeddings. Proceedings of the 29th International Conference on Computational Linguistics. Gyeongju: Association for Computational Linguistics, 2021. 2598–2607.
- 25 Balažević I, Allen C, Hospedales TM. Hypernetwork knowledge graph embeddings. Proceedings of the 28th International Conference on Artificial Neural Networks and Machine Learning. Munich: Springer, 2019. 553–565.
- 26 Ren FL, Li JC, Zhang HH, *et al.* Knowledge graph embedding with atrous convolution and residual learning. Proceedings of the 28th International Conference on Computational Linguistics. Barcelona: Association for Computational Linguistics, 2020. 1532–1543.
- 27 Ravishankar S, Chandras, Talukdar PP. Revisiting simple neural networks for learning representations of knowledge graphs. Proceedings of the 6th Workshop on Automated Knowledge Base Construction. Long Beach, 2017.
- 28 Tan ZX, Chen ZL, Feng SB, *et al.* KRACL: Contrastive learning with graph context modeling for sparse knowledge graph completion. Proceedings of the 2023 ACM Web Conference. Austin: ACM, 2023. 2548–2559.
- 29 周忠眉, 林宝德, 肖青. 古代方剂与新药方剂高频药组配情况分析. 漳州师范学院学报(自然科学版), 2004, 17(1): 19–21.
- 30 翁慧敏. 链路预测方法在药物重定位中的应用研究. 福建电脑, 2024, 40(3): 39–43.

(校对责编: 张重毅)