

# Towards Continual Knowledge Graph Embedding via Incremental Distillation

## 通过增量蒸馏实现持续知识图谱嵌入

Jiajun Liu <sup>1\*</sup>, Wenjun Ke <sup>1,2\*†</sup>, Peng Wang <sup>1,2†</sup>, Ziyu Shang <sup>1</sup>, Jinhua Gao <sup>3</sup>, Guozheng Li <sup>1</sup>, Ke Ji <sup>1</sup>, Yanhe Liu <sup>1</sup>

Jiajun Liu <sup>1\*</sup>, Wenjun Ke <sup>1,2\*†</sup>, Peng Wang <sup>1,2†</sup>, Ziyu Shang <sup>1</sup>, Jinhua Gao <sup>3</sup>, Guozheng Li <sup>1</sup>, Ke Ji <sup>1</sup>, Yanhe Liu <sup>1</sup>

<sup>1</sup> School of Computer Science and Engineering, Southeast University

<sup>1</sup> 东南大学计算机科学与工程学院

<sup>2</sup> Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications

<sup>2</sup> 新一代人工智能技术及其交叉应用重点实验室

(Southeast University), Ministry of Education, China

(东南大学), 教育部, 中国

<sup>3</sup> Institute of Computing Technology, Chinese Academy of Sciences

<sup>3</sup> 中国科学院计算技术研究所

{jiajliu, kewenjun, pwang, ziyus1999, liguozheng, keji, liuyanhe} @seu.edu.cn, gaojinhua@ict.ac.cn

{jiajliu, kewenjun, pwang, ziyus1999, liguozheng, keji, liuyanhe} @seu.edu.cn, gaojinhua@ict.ac.cn

## Abstract

### 摘要

Traditional knowledge graph embedding (KGE) methods typically require preserving the entire knowledge graph (KG) with significant training costs when new knowledge emerges. To address this issue, the continual knowledge graph embedding (CKGE) task has been proposed to train the KGE model by learning emerging knowledge efficiently while simultaneously preserving decent old knowledge. However, the explicit graph structure in KGs, which is critical for the above goal, has been heavily ignored by existing CKGE methods. On the one hand, existing methods usually learn new triples in a random order, destroying the inner structure of new KGs. On the other hand, old triples are preserved with equal priority, failing to alleviate catastrophic forgetting effectively. In this paper, we propose a competitive method for CKGE based on incremental distillation (IncDE), which considers the full use of the explicit graph structure in KGs. First,

to optimize the learning order, we introduce a hierarchical strategy, ranking new triples for layer-by-layer learning. By employing the inter- and intra-hierarchical orders together, new triples are grouped into layers based on the graph structure features. Secondly, to preserve the old knowledge effectively, we devise a novel incremental distillation mechanism, which facilitates the seamless transfer of entity representations from the previous layer to the next one, promoting old knowledge preservation. Finally, we adopt a two-stage training paradigm to avoid the over-corruption of old knowledge influenced by under-trained new knowledge. Experimental results demonstrate the superiority of IncDE over state-of-the-art baselines. Notably, the incremental distillation mechanism contributes to improvements of 0.2%-6.5% in the mean reciprocal rank (MRR) score. More exploratory experiments validate the effectiveness of IncDE in proficiently learning new knowledge while preserving old knowledge across all time steps.

传统的知识图谱嵌入 (KGE) 方法在新知识出现时通常需要保留整个知识图谱 (KG), 伴随高昂的训练成本。为解决此问题, 提出了持续知识图谱嵌入 (CKGE) 任务, 旨在高效学习新增知识同时保持已有知识。然而, 现有 CKGE 方法在实现上述目标时严重忽略了 KG 中对目标至关重要的显式图结构。一方面, 现有方法通常以随机顺序学习新三元组, 破坏了新 KG 的内部结构; 另一方面, 旧三元组以同等优先级被保留, 未能有效缓解灾难性遗忘。本文提出了一种基于增量蒸馏 (IncDE) 的竞争性 CKGE 方法, 充分利用 KG 的显式图结构。首先, 为优化学习顺序, 我们引入分层策略, 对新三元组进行分层排序以逐层学习。通过结合层间与层内顺序, 新三元组基于图结构特征被分组到不同层。其次, 为有效保持旧知识, 我们设计了新颖的增量蒸馏机制, 促进实体表示从上一层向下一层的无缝传递, 推进旧知识的保存。最后, 我们采用两阶段训练范式, 避免因新知识训练不足而对旧知识造成过度破坏。实验结果表明 IncDE 优于最先进基线方法。值得注意的是, 增量蒸馏机制在平均倒数排名 (MRR) 得分上带来了 0.2%-6.5% 的提升。更多探索性实验验证了 IncDE 能在各时间步高效学习新知识同时保留旧知识的有效性。

## Introduction

### 引言

Knowledge graph embedding (KGE) (Bordes et al. 2013; Wang et al. 2017; Rossi et al. 2021) aims to embed entities and relations from knowledge graphs (KGs) (Dong et al. 2014) into continuous vectors in a low-dimensional space, which is crucial for various knowledge-driven tasks, such as question answering (Bordes, Weston, and Usunier 2014), semantic search (Noy et al. 2019), and relation extraction (Li et al. 2022). Traditional KGE models (Bordes et al. 2013; Trouillon et al. 2016; Sun et al. 2019; Liu et al. 2020) only focus on obtaining embeddings of entities and relations in static KGs. However, real-world KGs constantly evolve, especially emerging new knowledge, such as new triples, entities, and relations. For example, during the evolution of DB-pedia (Bizer et al. 2009) from 2016 to 2018, about 1 million new entities, 2,000 new relations, and 20 million new triples emerged (DBpedia 2021). Traditionally, when a KG evolves, KGE models need to retrain the models with the entire KG, which is a non-trivial process with huge training costs. In domains such as bio-medical and financial fields, it is significant to update the KGE models to support medical assistance and informed market decision-making with rapidly evolving KGs, especially with substantial new knowledge.

知识图谱嵌入 (KGE)(Bordes et al. 2013; Wang et al. 2017; Rossi et al. 2021) 旨在将知识图谱 (KGs)(Dong et al. 2014) 中的实体和关系嵌入到低维连续向量空间, 这对多种以知识为驱动的任务至关重要, 例如问答 (Bordes, Weston, and Usunier 2014)、语义检索 (Noy et al. 2019) 和关系抽取 (Li et al. 2022)。传统 KGE 模型 (Bordes et al. 2013; Trouillon et al. 2016; Sun et al. 2019; Liu et al. 2020) 仅关注在静态 KG 中获取实体和关系的嵌入。然而, 真实世界的 KG 不断演化, 尤其是新知识的出现, 如新的三元组、实体和关系。例如, 在 DBpedia (Bizer et al. 2009) 从 2016 到 2018 的演化过程中, 约产生了 100 万个新实体、2000 个新关系和 2000 万个新三元组 (DBpedia 2021)。传统上, 当 KG 演化时, KGE 模型需要用整个 KG 重新训练模型, 这是一项代价巨大的复杂过程。在生物医药和金融等领域, 随着 KG 快速演化并伴随大量新知识, 及时更新 KGE 模型以支持医疗辅助和明智的市场决策具有重要意义。

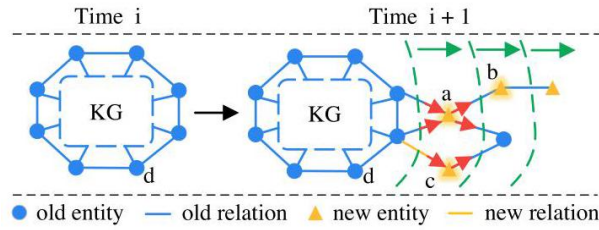


Figure 1: Illustration of a growing KG. Two specific learning orders should be considered: entities closer to the old KG should be prioritized (  $a$  is prioritised over  $b$  ); entities influenced heavier to new triples (e.g., connecting with more relations) should be prioritized (  $a$  is prioritised over  $c$  ).

图 1: 增长中的 KG 示意。应考虑两类具体学习顺序: 更接近旧 KG 的实体应优先 (  $a$  优先于  $b$  ); 受新三元组影响更大的实体 (例如连接更多关系) 应优先 (  $a$  优先于  $c$  )。

To this end, the continual KGE (CKGE) task has been proposed to alleviate this problem by using only the emerging knowledge for learning (Song and Park 2018; Daruna et al. 2021). In comparison with the traditional KGE, the key of CKGE lies in learning emerging knowledge well while preserving old knowledge effectively. As shown in Figure 1, new entities and relations (i.e., the new entity  $a$ ,  $b$ , and  $c$ ) should be learned to adapt to the new KG. Meanwhile, knowledge in the old KG (such as old entity  $d$ ) should be preserved. Generally, existing CKGE methods can be categorized into three families: dynamic architecture-based, replay-based, and regularization-based methods. Dynamic architecture-based methods (Rusu et al. 2016; Lomonaco and Maltoni 2017) preserve all old parameters and learn the emerging knowledge through new architectures. However, retaining all old parameters hinders the adaptation of old knowledge to the new knowledge. Replay-based methods (Lopez-Paz and Ranzato 2017; Wang et al. 2019; Kou et al. 2020) replay KG subgraphs to remember old knowledge, but recalling only a portion of the subgraphs leads to the destruction of the overall old graph structure. Regularization-based methods (Zenke, Poole, and Ganguli 2017; Kirkpatrick et al. 2017; Cui et al. 2023) aim to preserve old knowledge by adding regularization terms. However, only adding regularization terms to the old parameters makes it infeasible to capture new knowledge well.

为此, 提出了持续知识图嵌入 (CKGE) 任务, 仅使用新出现的知识进行学习以缓解该问题 (Song and Park 2018; Daruna et al. 2021)。与传统 KGE 相比, CKGE 的关键在于在有效保留旧知识的同时, 良好地学习新出现的知识。如图 1 所示, 新实体和关系 (即新实体  $a$ 、 $b$  和  $c$ ) 应被学习以适应新的 KG。同时, 旧 KG 中的知识 (例如旧实体  $d$ ) 应得到保留。总体而言, 现有 CKGE 方法可分为三类: 基于动态架构的方法、基于重放的方法和基于正则化的方法。基于动态架构的方法 (Rusu et al. 2016; Lomonaco and Maltoni 2017) 保留所有旧参数并通过新架构学习新出现的知识。然而, 保留所有旧参数会阻碍旧知识向新知识的适配。基于重放的方法 (Lopez-Paz and Ranzato 2017; Wang et al. 2019; Kou et al. 2020) 通过重放 KG 子图来记住旧知识, 但仅回忆部分子图会破坏整体旧图结构。基于正则化的方法 (Zenke, Poole, and Ganguli 2017; Kirkpatrick et al. 2017; Cui et al. 2023) 通过添加正则项来保留旧知识, 然而仅对旧参数添加正则项使得难以很好地捕捉新知识。

---

\*These authors contributed equally.

\* 这些作者贡献相同。

† Corresponding authors.

† 通讯作者。

Copyright ©2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

版权所有 ©2024, 人工智能促进协会 (www.aaai.org)。保留所有权利。

---

Despite achieving promising effectiveness, current CKGE methods still perform poorly due to the explicit graph structure of KGs being heavily ignored. Meanwhile, previous research has emphasized the crucial role of the graph structure in addressing graph-related continual learning tasks (Zhou and Cao 2021; Liang et al. 2022; Febrinanto et al. 2023). Specifically, existing CKGE methods suffer from two main drawbacks: (1) First, regarding the new emerging knowledge, current CKGE methods leverage a random-order learning strategy, neglecting the significance of different triples in a KG. Previous studies have demonstrated that the learning order of entities and relations can significantly affect continual learning on graphs (Wei et al. 2022). Since knowledge in KGs is organized in a graph structure, a randomized learning order can undermine the inherent semantics conveyed by KGs. Hence, it is essential to consider the priority of new entities and relations for effective learning and propagation. Figure 1 illustrates an example where entity  $a$  should be learned before entity  $b$  since the representation of  $b$  is propagated through  $a$  from the old KG. (2) Second, regarding the old knowledge, current CKGE methods treat the memorization at an equal level, leading to inefficient handling of catastrophic forgetting (Kirkpatrick et al. 2017). Existing studies have demonstrated that preserving knowledge by regularization or distillation from important nodes in the topology structure is critical for continuous graph learning (Liu, Yang, and Wang 2021). Therefore, old entities with more essential graph structure features should receive higher preservation priority. In Figure 1, entity  $a$  connecting more other entities should be prioritized for preservation at time  $i + 1$  compared to entity  $c$ .

尽管取得了有希望的效果, 现有 CKGE 方法仍表现不佳, 因为 KG 的显式图结构被严重忽视。同时, 先前研究强调了图结构在解决图相关持续学习任务中的关键作用 (Zhou and Cao 2021; Liang et al. 2022; Febrinanto et al. 2023)。具体地, 现有 CKGE 方法存在两大缺陷:(1) 首先, 针对新出现的知识, 当前 CKGE 方法采用随机顺序学习策略, 忽视了 KG 中不同三元组的重要性。先前研究表明, 实体和关系的学习顺序会显著影响图上的持续学习 (Wei et al. 2022)。由于 KG 中的知识以图结构组织, 随机化的学习顺序可能破坏 KG 所传达的固有语义。因此, 考虑新实体和关系的优先级以实现有效学习与传播是必要的。图 1 举例说明实体  $a$  应在实体  $b$  之前学习, 因为  $b$  的表示是通过旧 KG 中的  $a$  传播的。(2) 其次, 针对旧知识, 现有 CKGE 方法对记忆一视同仁, 导致灾难性遗忘的处理低效 (Kirkpatrick et al. 2017)。现有研究表明, 通过对拓扑结构中重要节点进行正则化或蒸馏来保留知识对连续图学习至关重要 (Liu, Yang, and Wang 2021)。因此, 具有更重要图结构特征的旧实体应获得更高的保留优先级。在图 1 中, 连接更多其它实体的实体  $a$  在时间  $i + 1$  相较于实体  $c$  应被优先保留。

In this paper, we propose IncDE, a novel method for the CKGE task that leverages incremental distillation. IncDE aims to enhance the capability of learning emerging knowledge while efficiently preserving old knowledge simultaneously. Firstly, we employ hierarchical ordering to determine the optimal learning sequence of new triples. This involves dividing the triples into layers and ranking them through the inter-hierarchical and intra-hierarchical orders. Subsequently, the ordered emerging knowledge is learned layer by layer. Secondly, we introduce a novel incremental distillation mechanism to preserve the old knowledge considering the graph structure effectively. This mechanism incorporates the explicit graph structure and employs a layer-by-layer paradigm to distill the entity representation. Finally, we use a two-stage training strategy to improve the preservation of old knowledge. In the first stage, we fix the representation of old entities and relations. In the second stage, we train the representation of all entities and relations, protecting the old KG from disruption by under-trained emerging knowledge.

本文提出了 IncDE, 一种用于 CKGE 任务的新方法, 利用增量蒸馏以同时增强新出现知识的学习能力并高效保留旧知识。首先, 我们采用层级排序来确定新三元组的最佳学习顺序, 将三元组划分为层并通过层间与层内顺序对其排序, 随后按层逐步学习有序的新出现知识。其次, 我们引入了一种新颖的增量蒸馏机制, 有效地结合图结构以保留旧知识。该机制纳入显式图结构并采用逐层范式对实体表示进行蒸馏。最后, 我们使用两阶段训练策略以改善旧知识的保留。在第一阶段, 固定旧实体和关系的表示; 在第二阶段, 训练所有实体和关系的表示, 以防止未充分训练的新出现知识破坏旧 KG。

To evaluate the effectiveness of IncDE, we construct three new datasets with varying scales of new KGs. Extensive experiments are conducted on both existing and new datasets. The results demonstrate that IncDE outperforms all strong baselines. Furthermore, ablation experiments reveal that incremental distillation provides a significant performance enhancement. Further exploratory experiments verify the ability of IncDE to effectively learn emerging knowledge while efficiently preserving old knowledge.

为评估 IncDE 的有效性, 我们构建了三套规模不同的新知识图数据集。在已有数据集与新数据集上进行了大量实验。结果表明 IncDE 超过了所有强基线。此外, 消融实验表明增量蒸馏带来了显著的性能提升。进一步的探索性实验验证了 IncDE 在高效保留旧知识的同时, 有效学习新兴知识的能力。

To sum up, the contributions of this paper are three-fold:

总之，本文的贡献有三点：

- We propose a novel continual knowledge graph embedding framework IncDE, which learns and preserves the knowledge effectively with explicit graph structure.

- 我们提出了一种新颖的连续知识图嵌入框架 IncDE，利用显式图结构有效学习并保留知识。

- We propose hierarchical ordering to get an adequate learning order for better learning emerging knowledge. Moreover, we propose incremental distillation and a two stage training strategy to preserve decent old knowledge.

- 我们提出分层排序以获得更合适的学习顺序，从而更好地学习新兴知识。此外，我们提出了增量蒸馏和两阶段训练策略以更好地保留旧知识。

- We construct three new datasets based on the scale changes of new knowledge. Experiments demonstrate that IncDE outperforms strong baselines. Notably, incremental distillation improves 0.2%-6.5% in MRR.

- 我们基于新知识规模变化构建了三套新数据集。实验表明 IncDE 优于强基线。值得注意的是，增量蒸馏在 MRR 上提高了 0.2%-6.5%。

## Related Work

### 相关工作

Different from traditional KGE (Bordes et al. 2013; Trouil-lon et al. 2016; Kazemi and Poole 2018; Pan and Wang 2021; Shang et al. 2023), CKGE (Song and Park 2018; Daruna et al. 2021) allows KGE models to learn emerging knowledge while remembering the old knowledge. Existing CKGE methods can be divided into three categories. (1) Dynamic architecture-based methods (Rusu et al. 2016; Lomonaco and Maltoni 2017) dynamically adapt to new neural resources to change architectural properties in response to new information and preserve old parameters. (2) Memory replay-based methods (Lopez-Paz and Ranzato 2017; Wang et al. 2019; Kou et al. 2020) retain the learned knowledge by replaying it. (3) Regularization-based methods (Zenke, Poole, and Ganguli 2017; Kirkpatrick et al. 2017; Cui et al. 2023) alleviate catastrophic forgetting by imposing constraints on updating neural weights. However, these methods overlook the importance of learning new knowledge in an appropriate order for graph data. Moreover, they ignore how to preserve appropriate old knowledge for better integration of new and old knowledge. Several datasets for CKGE (Hamaguchi et al. 2017; Kou et al. 2020; Daruna et al. 2021; Cui et al. 2023) have been constructed. However, most of them restrict the new triples to contain at least one old entity, neglecting triples without old entities. In the evolution of real-world KGs like Wikipedia (Bizer et al. 2009) and Yago (Suchanek, Kasneci, and Weikum 2007), numerous new triples emerge without any old entities.

与传统 KGE (Bordes et al. 2013; Trouillon et al. 2016; Kazemi and Poole 2018; Pan and Wang 2021; Shang et al. 2023) 不同, CKGE (Song and Park 2018; Daruna et al. 2021) 允许 KGE 模型在学习新兴知识的同时记住旧知识。现有 CKGE 方法可分为三类。(1) 基于动态架构的方法 (Rusu et al. 2016; Lomonaco and Maltoni 2017) 动态适应新神经资源以响应新信息并改变架构属性, 同时保留旧参数。(2) 基于记忆重放的方法 (Lopez-Paz and Ranzato 2017; Wang et al. 2019; Kou et al. 2020) 通过重放保留已学知识。(3) 基于正则化的方法 (Zenke, Poole, and Ganguli 2017; Kirkpatrick et al. 2017; Cui et al. 2023) 通过对神经权重更新施加约束来缓解灾难性遗忘。然而, 这些方法忽视了为图数据按适当顺序学习新知识的重要性。此外, 它们也未考虑如何保留恰当的旧知识以便更好地整合新旧知识。已构建了若干 CKGE 数据集 (Hamaguchi et al. 2017; Kou et al. 2020; Daruna et al. 2021; Cui et al. 2023)。但大多数数据集限制新三元组至少包含一个旧实体, 忽略了不含旧实体的三元组。在像 Wikipedia (Bizer et al. 2009) 和 Yago (Suchanek, Kasneci, and Weikum 2007) 这样的真实世界知识图演化中, 大量新三元组是在不含任何旧实体的情况下出现的。

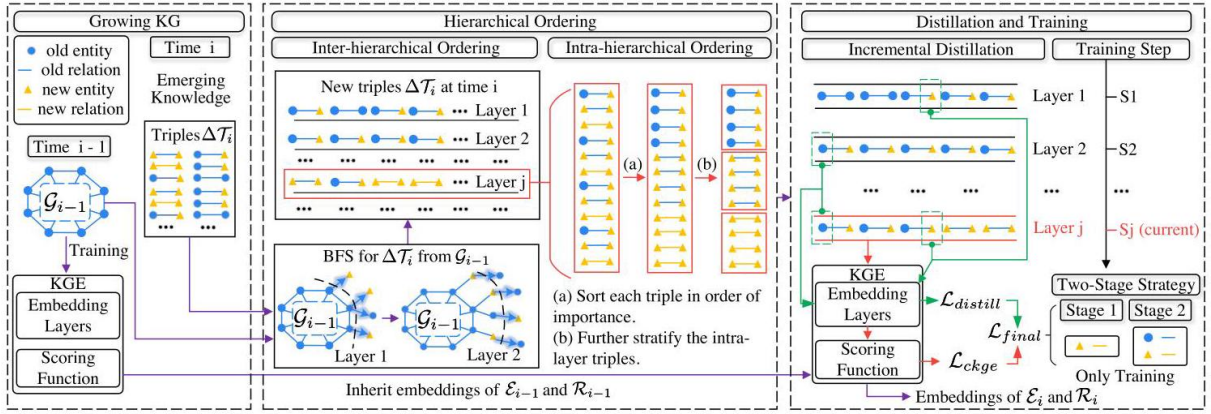


Figure 2: An overview of our proposed IncDE framework.

图 2: 我们提出的 IncDE 框架概览。

## Preliminary and Problem Statement

### 预备知识与问题陈述

## Growing Knowledge Graph

### 增长的知识图

A knowledge graph (KG)  $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$  contains the collection of entities  $\mathcal{E}$ , relations  $\mathcal{R}$ , and triples  $\mathcal{T}$ . A triple can be denoted as  $(h, r, t) \in \mathcal{T}$ , where  $h, r$ , and  $t$  represent the head entity, the relation, and the tail entity, respectively. When a KG grows with emerging knowledge at time  $i$ , it is denoted as  $\mathcal{G}_i = (\mathcal{E}_i, \mathcal{R}_i, \mathcal{T}_i)$ , where  $\mathcal{E}_i, \mathcal{R}_i, \mathcal{T}_i$  are the collection of entities, relations, and triples in  $\mathcal{G}_i$ . Moreover, we denote  $\Delta \mathcal{T}_i = \mathcal{T}_i - \mathcal{T}_{i-1}$ ,  $\Delta \mathcal{E}_i = \mathcal{E}_i - \mathcal{E}_{i-1}$  and  $\Delta \mathcal{R}_i = \mathcal{R}_i - \mathcal{R}_{i-1}$  as new triples, entities, and relations, respectively.

一个知识图 (KG)  $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$  包含实体集合  $\mathcal{E}$ 、关系  $\mathcal{R}$  和三元组  $\mathcal{T}$ 。一个三元组可表示为  $(h, r, t) \in \mathcal{T}$ ，其中  $h, r, t$  分别表示头实体、关系和尾实体。当知识图在时间  $i$  随新兴知识增长时，表示为  $\mathcal{G}_i = (\mathcal{E}_i, \mathcal{R}_i, \mathcal{T}_i)$ ，其中  $\mathcal{E}_i, \mathcal{R}_i, \mathcal{T}_i$  是  $\mathcal{G}_i$  中的实体、关系和三元组集合。此外，我们将  $\Delta\mathcal{T}_i = \mathcal{T}_i - \mathcal{T}_{i-1}$ ， $\Delta\mathcal{E}_i = \mathcal{E}_i - \mathcal{E}_{i-1}$  和  $\Delta\mathcal{R}_i = \mathcal{R}_i - \mathcal{R}_{i-1}$  分别表示为新的三元组、实体和关系。

## Continual Knowledge Graph Embedding

### 连续知识图嵌入

Given a KG  $\mathcal{G}$ , knowledge graph embedding (KGE) aims to embed entities and relations into low-dimensional vector space  $\mathbb{R}$ . Given head entity  $h \in \mathcal{E}$ , relation  $r \in \mathcal{R}$ , and tail entity  $t \in \mathcal{E}$ , their embeddings are denoted as  $\mathbf{h} \in \mathbb{R}^d$ ,  $\mathbf{r} \in \mathbb{R}^d$ , and  $\mathbf{t} \in \mathbb{R}^d$ , where  $d$  is the embedding size. A typical KGE model contains embedding layers and a scoring function. Embedding layers generate vector representations for entities and relations, while the scoring function assigns scores to each triple in the training stage.

给定知识图谱 KG  $\mathcal{G}$ ，知识图谱嵌入 (KGE) 旨在将实体和关系嵌入到低维向量空间中  $\mathbb{R}$ 。给定头实体  $h \in \mathcal{E}$ 、关系  $r \in \mathcal{R}$  和尾实体  $t \in \mathcal{E}$ ，它们的嵌入分别记作  $\mathbf{h} \in \mathbb{R}^d$ 、 $\mathbf{r} \in \mathbb{R}^d$  和  $\mathbf{t} \in \mathbb{R}^d$ ，其中  $d$  为嵌入维度。典型的 KGE 模型包含嵌入层和评分函数。嵌入层为实体和关系生成向量表示，评分函数在训练阶段为每个三元组分配分数。

Given a growing KG  $\mathcal{G}_i$  at time  $i$ , continual knowledge graph embedding (CKGE) aims to update the embeddings of old entities  $\mathcal{E}_{i-1}$  and relations  $\mathcal{R}_{i-1}$  while obtaining the embeddings of new entities  $\Delta\mathcal{E}_i$  and relations  $\Delta\mathcal{R}_i$ . Finally, embeddings of all entities  $\mathcal{E}_i$  and relations  $\mathcal{R}_i$  are obtained.

对于在时间点  $i$  不断增长的知识图谱  $\mathcal{G}_i$ ，持续知识图谱嵌入 (CKGE) 旨在在获得新实体  $\Delta\mathcal{E}_i$  和新关系  $\Delta\mathcal{R}_i$  的嵌入的同时更新旧实体  $\mathcal{E}_{i-1}$  和旧关系  $\mathcal{R}_{i-1}$  的嵌入。最终得到所有实体  $\mathcal{E}_i$  和关系  $\mathcal{R}_i$  的嵌入。

## Methodology

### Methodology

## Framework Overview

### Framework Overview

The framework of IncDE is depicted in Figure 2. Initially, when emerging knowledge appears at time  $i$ , IncDE performs hierarchical ordering on new triples  $\Delta\mathcal{T}_i$ . Specifically, inter-hierarchical ordering is employed to divide  $\Delta\mathcal{T}_i$  into multiple layers using breadth-first search (BFS) expansion from the old graph  $\mathcal{G}_{i-1}$ . Subsequently, intra-hierarchical ordering is applied within each layer to further sort and divide the triples. Then, the grouped  $\Delta\mathcal{T}_i$  is trained layer by layer, with the embeddings of  $\mathcal{E}_{i-1}$  and  $\mathcal{R}_{i-1}$  inherited from the KGE model in previous time  $i - 1$ . During training, incremental distillation is introduced. Precisely, if an entity



in layer  $j$  has appeared in a previous layer, its representation is distilled with the closest layer to the current one. Additionally, a two-stage training strategy is proposed. In the first stage, only the representations of new entities  $\Delta\mathcal{E}_i$  and relations  $\Delta\mathcal{R}_i$  are trained. In the second stage, all entities  $\mathcal{E}_i$  and relations  $\mathcal{R}_i$  are trained in the training process. Finally, the embeddings of  $\mathcal{E}_i$  and  $\mathcal{R}_i$  at time  $i$  are obtained.

IncDE 的框架如图 2 所示。最初，当在时间  $i$  出现新知识时，IncDE 对新三元组  $\Delta\mathcal{T}_i$  进行分层排序。具体地，先通过从旧图  $\mathcal{G}_{i-1}$  进行广度优先搜索 (BFS) 扩展采用层间排序，将  $\Delta\mathcal{T}_i$  划分为多层。随后，在每一层内采用层内排序对三元组进行进一步排序和划分。然后，分组后的  $\Delta\mathcal{T}_i$  按层训练， $\mathcal{E}_{i-1}$  和  $\mathcal{R}_{i-1}$  的嵌入从前一时刻的 KGE 模型继承 (时间为  $i-1$ )。在训练过程中引入了增量蒸馏。准确地说，如果层  $j$  中的一个实体曾在先前层出现，则其表示会向与当前层最近的一层进行蒸馏。此外，提出了两阶段训练策略。第一阶段仅训练新实体  $\Delta\mathcal{E}_i$  和新关系  $\Delta\mathcal{R}_i$  的表示。第二阶段在训练过程中训练所有实体  $\mathcal{E}_i$  和关系  $\mathcal{R}_i$ 。最终得到时间为  $i$  时刻的  $\mathcal{E}_i$  和  $\mathcal{R}_i$  的嵌入。

## Hierarchical Ordering

### Hierarchical Ordering

To enhance the learning of the graph structure for emerging knowledge, we first order the triples  $\Delta\mathcal{T}_i$  at time  $i$  in an inter-hierarchical way and an intra-hierarchical way, based on the importance of entities and relations, as shown in Figure 2. Ordering processes can be pre-calculated to reduce training time. Then, we learn the new triples  $\Delta\mathcal{T}_i$  layer by layer and in order. The specific ordering strategies are as follows.

为加强对新兴知识的图结构学习，我们首先基于实体和关系的重要性，对时间  $i$  的三元组  $\Delta\mathcal{T}_i$  进行层间与层内排序，如图 2 所示。排序过程可预先计算以减少训练时间。随后我们按层并按序学习新的三元组  $\Delta\mathcal{T}_i$ 。具体的排序策略如下。

**Inter-Hierarchical Ordering** For inter-hierarchical ordering, we split all new triples  $\Delta\mathcal{T}_i$  into multiple layers  $l_1, l_2, \dots, l_n$  at time  $i$ . Since the representations of new entities  $\Delta\mathcal{E}_i$  are propagated from the representations of the old entities  $\mathcal{E}_{i-1}$  and old relations  $\mathcal{R}_{i-1}$ , we split new triples  $\Delta\mathcal{T}_i$  based on the distance between new entities  $\Delta\mathcal{E}_i$  and old graph  $\mathcal{G}_{i-1}$ . We use the bread-first search (BFS) algorithm to progressively partition  $\Delta\mathcal{T}_i$  from  $\mathcal{G}_{i-1}$ . First, we take the old graph as  $l_0$ . Then, we take all the new triples that contain old entities as the next layer,  $l_1$ . Next, we treat the new entities in  $l_1$  as the seen old entities. Repeat the above two processes until no triples can be added to a new layer. Finally, we use all remaining triples as the final layer. This way, we initially divide all the new triples  $\Delta\mathcal{T}_i$  into multiple layers.

跨层次排序对于跨层次排序，我们将所有新三元组  $\Delta\mathcal{T}_i$  在时间  $l_1, l_2, \dots, l_n$  划分为多层  $i$ 。由于新实体  $\Delta\mathcal{E}_i$  的表示是从旧实体  $\mathcal{E}_{i-1}$  和旧关系  $\mathcal{R}_{i-1}$  的表示传播得到的，我们根据新实体  $\Delta\mathcal{E}_i$  与旧图  $\mathcal{G}_{i-1}$  的距离来划分新三元组  $\Delta\mathcal{T}_i$ 。我们使用广度优先搜索 (BFS) 算法从  $\mathcal{G}_{i-1}$  逐步划分  $\Delta\mathcal{T}_i$ 。首先，我们将旧图作为  $l_0$ 。然后，将所有包含旧实体的新三元组作为下一层， $l_1$ 。接着，将  $l_1$  中的新实体视为已见的旧实体。重复上述两步，直到无法添加三元组为止。最后，我们将所有剩余三元组作为最终一层。以此方式，我们初步将所有新三元组  $\Delta\mathcal{T}_i$  划分为多层。

**Intra-Hierarchical Ordering** The importance of the triples in graph structure is also critical to the order in which entities  $\mathcal{E}_i$  and relations  $\mathcal{R}_i$  are learned or updated at time  $i$ .

层内排序图结构中三元组的重要性对于在时间  $i$  学习或更新实体  $\mathcal{E}_i$  和关系  $\mathcal{R}_i$  的顺序也至关重要。

So for the triples of each layer, we further order them based on the importance of entities and relations in the graph structure, as shown in Figure 2 (a). To measure the importance of entities  $\mathcal{E}_i$  in the new triples  $\Delta\mathcal{T}_i$ , we first calculate the node centrality of an entity  $e \in \mathcal{E}_i$  as  $f_{nc}(e)$  as follow:

因此对于每层的三元组，我们进一步根据图结构中实体和关系的重要性对它们排序，如图 2(a) 所示。为衡量新三元组  $\Delta\mathcal{T}_i$  中实体  $\mathcal{E}_i$  的重要性，我们首先将实体  $e \in \mathcal{E}_i$  的节点中心性计算为  $f_{nc}(e)$ ，具体如下：

$$f_{nc}(e) = \frac{f_{\text{neighbor}}(e)}{N - 1} \quad (1)$$

where  $f_{\text{neighbor}}(e)$  denotes the number of the neighbors of  $e$ , and  $N$  denotes the number of entities in the new triples  $\Delta\mathcal{T}_i$  at time  $i$ . Then, in order to measure the importance of relations  $\mathcal{R}_i$  in the triples of each layer, we compute the betweenness centrality of a relation  $r \in \mathcal{R}_i$  as  $f_{bc}(r)$ :

其中  $f_{\text{neighbor}}(e)$  表示  $e$  的邻居数， $N$  表示在时间  $i$  的新三元组  $\Delta\mathcal{T}_i$  中的实体数。然后，为衡量每层三元组中关系  $\mathcal{R}_i$  的重要性，我们将关系  $r \in \mathcal{R}_i$  的介数中心性计算为  $f_{bc}(r)$ ：

$$f_{bc}(r) = \sum_{s,t \in \mathcal{E}_i, s \neq t} \frac{\sigma(s, t | r)}{\sigma(s, t)} \quad (2)$$

where  $\sigma(s, t)$  is the number of shortest paths between  $s$  and  $t$  in the new triples  $\Delta\mathcal{T}_i$ , and  $\sigma(s, t | r)$  is the number of  $\sigma(s, t)$  passing through relation  $r$ . Specifically, we only compute  $f_{nc}$  and  $f_{bc}$  of emerging KGs, avoiding the graph being excessive. To obtain the importance of the triple  $(h, r, t)$  in each layer, we compute the node centrality of the head entity  $h$ , the node centrality of the tail entity  $t$ , and the betweenness centrality of the relation  $r$  in this triple. Considering the overall significance of entities and relations within the graph structure, we adopt  $f_{nc}$  and  $f_{bc}$  together. The final importance of each triple can be calculated as:

其中  $\sigma(s, t)$  是新三元组  $\Delta\mathcal{T}_i$  中  $s$  与  $t$  之间最短路径的数量， $\sigma(s, t | r)$  是经过关系  $r$  的  $\sigma(s, t)$  的数量。具体而言，我们仅计算新兴知识图谱的  $f_{nc}$  和  $f_{bc}$ ，以避免图过度膨胀。为获得每层中三元组  $(h, r, t)$  的重要性，我们计算该三元组中头实体  $h$  的节点中心性、尾实体  $t$  的节点中心性以及该关系  $r$  的介数中心性。考虑到实体与关系在图结构中的整体重要性，我们同时采用  $f_{nc}$  与  $f_{bc}$ 。每个三元组的最终重要性可计算为：

$$IT_{(h,r,t)} = \max(f_{nc}(h), f_{nc}(t)) + f_{bc}(r) \quad (3)$$

We sort the triples of each layer according to the values of their  $IT$  values. The utilization of intra-hierarchical ordering guarantees the prioritization of triples that are important to the graph structure in each layer. This, in turn, enables more effective learning of the structure of the new graph.

我们根据每个三元组的  $IT$  值对每层的三元组进行排序。层内排序的应用确保在每层中优先处理对图结构重要的三元组，从而更有效地学习新图的结构。

Moreover, the intra-hierarchical ordering can help further split the intra-layer triples, as shown in Figure 2(b). Since the number of triples in each layer is determined by the size of the new graph, it could be too large to learn. To prevent the number of triples in a particular layer from being too large, we set the maximum number of triples in each layer to be  $M$ . If the number of triples in one layer exceeds  $M$ , it can split into several layers not exceeding  $M$  triples in the intra-hierarchical ordering.

此外，层内层次排序还能帮助进一步划分层内三元组，如图 2(b) 所示。由于每层三元组的数量由新图的大小决定，可能过多难以学习。为防止某一层的三元组数过大，我们将每层的最大三元组数设为  $M$ 。若某层的三元组数超过  $M$ ，则可在层内层次排序中拆分成若干不超过  $M$  三元组的层。

## Distillation and Training

### 蒸馏与训练

After hierarchical ordering, we train new triples  $\Delta\mathcal{T}_i$  layer by layer at time  $i$ . We take TransE (Bordes et al. 2013) as the base KGE model. When training the  $j$ -th layer ( $j > 0$ ), the loss for the original TransE model is:

在完成层次排序后，我们按时间  $i$  逐层训练新的三元组  $\Delta\mathcal{T}_i$ 。我们采用 TransE (Bordes et al. 2013) 作为基础 KGE 模型。训练第  $j$  层 ( $j > 0$ ) 时，原始 TransE 模型的损失为：

$$\mathcal{L}_{\text{ckge}} = \sum_{(h,r,t) \in l_j} \max(0, f(h, r, t) - f(h', r, t') + \gamma) \quad (4)$$

where  $(h', r, t')$  is the negative triple of  $(h, r, t) \in l_j$ , and  $f(h, r, t) = |h + r - t|_{L1/L2}$  is the score function of TransE. We inherit the embeddings of old entities  $\mathcal{E}_{i-1}$  and relations  $\mathcal{R}_{i-1}$  from the KGE model at time  $i-1$  and randomly initialize the embeddings of new entities  $\Delta\mathcal{E}_i$  and relations  $\Delta\mathcal{R}_i$ .

其中  $(h', r, t')$  是  $(h, r, t) \in l_j$  的负三元组，而  $f(h, r, t) = |h + r - t|_{L1/L2}$  是 TransE 的评分函数。我们从时间  $i-1$  的 KGE 模型继承旧实体  $\mathcal{E}_{i-1}$  和关系  $\mathcal{R}_{i-1}$  的嵌入，并对新实体  $\Delta\mathcal{E}_i$  和关系  $\Delta\mathcal{R}_i$  的嵌入进行随机初始化。

During training, we use incremental distillation to preserve the old knowledge. Further, we propose a two-stage training strategy to prevent the embeddings of old entities and relations from being overly corrupted at the start of training.

训练过程中，我们使用增量蒸馏以保留旧知识。此外，我们提出了两阶段训练策略，以防止旧实体和关系的嵌入在训练开始阶段被过度破坏。

**Incremental Distillation** In order to alleviate catastrophic forgetting of the entities learned in previous layers, inspired by the knowledge distillation for KGE models (Wang et al. 2021; Zhu et al. 2022; Liu et al. 2023), we distill the entity representation in the current layer with the entities that have appeared in previous layers as shown in Figure 2. Specifically, if entity  $e$  in the  $j$ -th ( $j > 0$ ) layer has appeared in a previous layer, we distill it with the representation of  $e$  from the nearest layer. The loss of distillation for entity  $e_k$  ( $k \in [1, |\mathcal{E}_i|]$ ) is:

增量蒸馏为减轻先前层中已学习实体的灾难性遗忘，受知识蒸馏在 KGE 模型中的启发 (Wang et al. 2021; Zhu et al. 2022; Liu et al. 2023)，我们对当前层的实体表示进行蒸馏，使用已在先前层出现的实体，如图 2 所示。具体地，若实体  $e$  在第  $j$ -th ( $j > 0$ ) 层已在先前层出现，我们就用最近一层中  $e$  的表示对其进行蒸馏。实体  $e_k$  ( $k \in [1, |\mathcal{E}_i|]$ ) 的蒸馏损失为：

$$\mathcal{L}_{\text{distill}}^k = \begin{cases} \frac{1}{2}(\mathbf{e}'_k - \mathbf{e}_k)^2, & |\mathbf{e}'_k - \mathbf{e}_k| \leq 1 \\ |\mathbf{e}'_k - \mathbf{e}_k| - \frac{1}{2}, & |\mathbf{e}'_k - \mathbf{e}_k| > 1 \end{cases} \quad (5)$$

where  $\mathbf{e}_k$  denotes the representation of entity  $e_k$  in layer  $j$ ,  $\mathbf{e}'_k$  denotes the representation of entity  $e_k$  from the nearest previous layer. By distilling entities that have appeared in previous layers, we remember old knowledge efficiently. However, different entities should have different levels of memory for past representations. Entities with higher importance in the graph structure should be prioritized and preserved to a greater extent during distillation. Besides the node centrality of the entity  $f_{nc}$ , similar to the betweenness centrality of the relation, we define the betweenness centrality  $f_{bc}(e)$  of an entity  $e$  at time  $i$  as:

其中  $\mathbf{e}_k$  表示第  $j$  层中实体  $e_k$  的表示， $\mathbf{e}'_k$  表示来自最近先前层的实体  $e_k$  的表示。通过蒸馏已在先前层出现的实体，我们能高效记忆旧知识。然而，不同实体对过去表示的记忆程度应不同。在图结构中重要性更高的实体应在蒸馏过程中被优先且更大程度地保留。除了节点的中心性  $f_{nc}$ ，类似于关系的中介中心性，我们将时间  $i$  时实体  $e$  的中介中心性定义为  $f_{bc}(e)$ ：

$$f_{bc}(e) = \sum_{s,t \in \mathcal{E}_i, s \neq t} \frac{\sigma(s, t | e)}{\sigma(s, t)} \quad (6)$$

We combine  $f_{bc}(e)$  and  $f_{nc}(e)$  to evaluate the importance of an entity  $e$ . Concretely, when training the  $j$ -th layer, for each new entity  $e_k$  appearing at the time  $i$ , we compute  $f_{bc}(e_k)$  and  $f_{nc}(e_k)$  to get the preliminary weight  $\lambda_k$  as:

我们结合  $f_{bc}(e)$  和  $f_{nc}(e)$  来评估实体  $e$  的重要性。具体地，在训练第  $j$  层时，对于在时间  $i$  出现的每个新实体  $e_k$ ，我们计算  $f_{bc}(e_k)$  和  $f_{nc}(e_k)$  以得到初步权重  $\lambda_k$ ，其计算为：

$$\lambda_k = \lambda_0 \cdot (f_{bc}(e_k) + f_{nc}(e_k)) \quad (7)$$

where  $\lambda_0$  is 1 for new entities that have already appeared in previous layers, and  $\lambda_0$  is 0 for new entities that have not appeared. At the same time, we learn a matrix  $\mathbf{W} \in \mathbb{R}^{1 \times |\mathcal{E}_i|}$  to dynamically change the weights of distillation loss for different entities. The dynamic distillation weights is:

其中  $\lambda_0$  对于已在前几层出现的新实体为 1，对于尚未出现的新实体为 0。同时，我们学习一个矩阵  $\mathbf{W} \in \mathbb{R}^{1 \times |\mathcal{E}_i|}$  来动态调整不同实体的蒸馏损失权重。动态蒸馏权重为：

$$[\lambda'_1, \lambda'_2, \dots, \lambda'_{|\mathcal{E}_i|}] = [\lambda_1, \lambda_2, \dots, \lambda_{|\mathcal{E}_i|}] \circ \mathbf{W} \quad (8)$$

where  $\circ$  denotes the Hadamard product. The final distillation loss for each layer  $j$  at the time  $i$  is:

其中  $\circ$  表示哈达玛积。时刻  $i$  下第  $j$  层的最终蒸馏损失为：

$$\mathcal{L}_{\text{distill}} = \sum_{k=1}^{|\mathcal{E}_i|} \lambda'_k \cdot \mathcal{L}_{\text{distill}}^k \quad (9)$$

When training the  $j$ -th layer, the final loss function can be calculated as:

在训练第  $j$  层时，最终损失函数可计算为：

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{ckge}} + \mathcal{L}_{\text{distill}} \quad (10)$$

After layer-by-layer training for new triples  $\Delta\mathcal{T}_i$ , all representations of entities  $\mathcal{E}_i$  and relations  $\mathcal{R}_i$  are obtained.

经过对新三元组  $\Delta\mathcal{T}_i$  的逐层训练后，所有实体  $\mathcal{E}_i$  与关系  $\mathcal{R}_i$  的表示均被获得。

Dataset	Time 1			Time 2			Time 3			Time 4			Time 5		
	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$
ENTITY	2,909	233	46,388	5,817	236	72,111	8,275	236	73,785	11,633	237	70,506	14,541	237	47,326
RELATION	11,560	48	98,819	13,343	96	93,535	13,754	143	66,136	14,387	190	30,032	14,541	237	21,594
FACT	10,513	237	62,024	12,779	237	62,023	13,586	237	62,023	13,894	237	62,023	14,541	237	62,023
HYBRID	8,628	86	57,561	10,040	102	20,873	12,779	151	88,017	14,393	209	103,339	14,541	237	40,326
GraphEqual	2,908	226	57,636	5,816	235	62,023	8,724	237	62,023	11,632	237	62,023	14,541	237	66,411
GraphHigher	900	197	10,000	1,838	221	20,000	3,714	234	40,000	7,467	237	80,000	14,541	237	160,116
GraphLower	7,505	237	160,000	11,258	237	80,000	13,134	237	40,000	14,072	237	20,000	14,541	237	10,116

数据集	时间 1			时间 2			时间 3			时间 4			时间 5		
	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$	$N_E$	$N_R$	$N_T$
实体	2,909	233	46,388	5,817	236	72,111	8,275	236	73,785	11,633	237	70,506	14,541	237	47,326
关系	11,560	48	98,819	13,343	96	93,535	13,754	143	66,136	14,387	190	30,032	14,541	237	21,594
事实	10,513	237	62,024	12,779	237	62,023	13,586	237	62,023	13,894	237	62,023	14,541	237	62,023
混合	8,628	86	57,561	10,040	102	20,873	12,779	151	88,017	14,393	209	103,339	14,541	237	40,326
图相等	2,908	226	57,636	5,816	235	62,023	8,724	237	62,023	11,632	237	62,023	14,541	237	66,411
图更高	900	197	10,000	1,838	221	20,000	3,714	234	40,000	7,467	237	80,000	14,541	237	160,116
图更低	7,505	237	160,000	11,258	237	80,000	13,134	237	40,000	14,072	237	20,000	14,541	237	10,116

Table 1: The statistics of datasets.  $N_E, N_R$  and  $N_T$  denote the number of cumulative entities, cumulative relations and current triples at each time  $i$ .

表 1: 数据集统计。 $N_E, N_R$  和  $N_T$  分别表示在每个时间  $i$  的累计实体数、累计关系数和当前三元组数。

**Two-Stage Training** During the training process, when incorporating the new triples  $\Delta\mathcal{T}_i$  into the existing graph  $\mathcal{G}_{i-1}$  at time  $i$ , the embeddings of old entities and relations that are not present in the new triples  $\Delta\mathcal{T}_i$  remain unchanged. However, the embeddings of old entities and relations that are included in the new triples  $\Delta\mathcal{T}_i$  are updated. Therefore, in the initial stage of each time  $i$ , part of the representations of entities  $\mathcal{E}_{i-1}$  and relations  $\mathcal{R}_{i-1}$  in the old graph  $\mathcal{G}_{i-1}$  will be corrupted by the new entities  $\Delta\mathcal{E}_i$  and relations  $\Delta\mathcal{R}_i$  that are not fully trained. To solve this problem, IncDE uses a two-stage training strategy to preserve the knowledge in the old graph better, as shown in Figure 2. In the first training stage, IncDE freezes the embeddings of all old entities  $\mathcal{E}_{i-1}$  and relations  $\mathcal{R}_{i-1}$  and trains only the embeddings of new entities  $\Delta\mathcal{E}_i$  and relations  $\Delta\mathcal{R}_i$ . Then, IncDE trains the embeddings of all entities  $\mathcal{E}_i$  and relations  $\mathcal{R}_i$  in the new graph in the second training stage. With the two-stage training strategy, IncDE prevents the structure of the old graph from disruption by new

triples in the early training phase. At the same time, the representations of entities and relations in the old graph and those in the new graph can be better adapted to each other during training.

两阶段训练在训练过程中，当在时间  $i$  将新三元组  $\Delta\mathcal{T}_i$  并入已有图  $\mathcal{G}_{i-1}$  时，不出现在新三元组  $\Delta\mathcal{T}_i$  中的旧实体和旧关系的嵌入保持不变。但出现在新三元组  $\Delta\mathcal{T}_i$  中的旧实体和旧关系的嵌入会被更新。因此，在每个时间  $i$  的初始阶段，旧图  $\mathcal{G}_{i-1}$  中部分实体  $\mathcal{E}_{i-1}$  和关系  $\mathcal{R}_{i-1}$  的表示会被尚未充分训练的新实体  $\Delta\mathcal{E}_i$  和新关系  $\Delta\mathcal{R}_i$  所损坏。为了解决这一问题，IncDE 采用两阶段训练策略以更好地保留旧图中的知识，如图 2 所示。第一阶段，IncDE 冻结所有旧实体  $\mathcal{E}_{i-1}$  和旧关系  $\mathcal{R}_{i-1}$  的嵌入，仅训练新实体  $\Delta\mathcal{E}_i$  和新关系  $\Delta\mathcal{R}_i$  的嵌入。然后，IncDE 在第二阶段训练新图中所有实体  $\mathcal{E}_i$  和关系  $\mathcal{R}_i$  的嵌入。通过两阶段训练策略，IncDE 防止在早期训练阶段新三元组破坏旧图结构，同时使旧图与新图中实体和关系的表示在训练过程中更好地相互适应。

## Experiments

### 实验

## Experimental Setup

### 实验设置

**Datasets** We use seven datasets for CKGE, including four public datasets (Cui et al. 2023): ENTITY, RELATION, FACT, HYBRID, as well as three new datasets constructed by us: GraphEqual, GraphHigher, and GraphLower. In ENTITY, RELATION, and FACT, the number of entities, relations, and triples increases uniformly at each time step. In HYBRID, the sum of entities, relations, and triples increases uniformly over time. However, these datasets constrain knowledge growth, requiring new triples to include at least one existing entity. To address this limitation, we relax these constraints and construct three new datasets: GraphE-qual, GraphHigher, and GraphLower. In GraphEqual, the number of triples consistently increases by the same increment at each time step. In GraphHigher and GraphLower, the increments of triples become higher and lower, respectively. Detailed statistics for all datasets are presented in Table 1. The time step is set to 5. The train, valid, and test sets are allocated 3:1:1 for each time step. The datasets are available at <https://github.com/seukgcode/IncDE>.

数据集我们为 CKGE 使用七个数据集，包括四个公开数据集 (Cui et al. 2023): ENTITY、RELATION、FACT、HYBRID，以及我们构建的三个新数据集: GraphEqual、GraphHigher 和 GraphLower。在 ENTITY、RELATION 和 FACT 中，实体、关系和三元组的数量在每个时间步均匀增加。在 HYBRID 中，实体、关系和三元组的总和随时间均匀增长。然而，这些数据集限制了知识增长，要求新三元组至少包含一个已存在的实体。为了解决此限制，我们放宽这些约束并构建了三个新数据集: GraphEqual、GraphHigher 和 GraphLower。在 GraphEqual 中，三元组数在每个时间步以相同增量持续增加。在 GraphHigher 和 GraphLower 中，三元组的增量分别变大和变小。所有数据集的详细统计见表 1。时间步设置为 5。每个时间步的训练、验证和测试集按 3:1:1 分配。数据集可在 <https://github.com/seukgcode/IncDE> 获取。

**Baselines** We select two kinds of baseline models: non-continual learning methods and continual learning-based methods. First, we select a non-continual learning method, Fine-tune (Cui et al. 2023), which is fine-tuned with the new triples each time. Then, we select three kinds of continual learning-based methods: dynamic architecture-based, memory replay-based baselines, and regularization-based. Specifically, the dynamic architecture-based methods are PNN (Rusu et al. 2016) and CWR (Lomonaco and Maltoni 2017). The memory replay-based methods are GEM (Lopez-Paz and Ranzato 2017), EMR (Wang et al. 2019), and DiC-GRL (Kou et al. 2020). The regularization-based methods are SI (Zenke, Poole, and Ganguli 2017), EWC (Kirkpatrick et al. 2017), and LKGE (Cui et al. 2023).

基线我们选择两类基线模型: 非连续学习方法和基于连续学习的方法。首先, 选择一种非连续学习方法 Fine-tune(Cui et al. 2023), 每次用新三元组进行微调。然后, 选择三类基于连续学习的方法: 基于动态架构的、基于记忆重放的和基于正则化的。具体而言, 基于动态架构的方法有 PNN(Rusu et al. 2016) 和 CWR(Lomonaco and Maltoni 2017)。基于记忆重放的方法有 GEM(Lopez-Paz and Ranzato 2017)、EMR(Wang et al. 2019) 和 DiC-GRL(Kou et al. 2020)。基于正则化的方法有 SI(Zenke, Poole, and Ganguli 2017)、EWC(Kirkpatrick et al. 2017) 和 LKGE(Cui et al. 2023)。

**Metrics** We evaluate our model performance on the link prediction task. Particularly, we replace the head or tail entity of the triples in the test set with all other entities and then compute and rank the scores for each triple. Then, we compute MRR, Hits@1, and Hits@10 as metrics. The higher the MRR, Hits@1, Hits@3, and Hits@10, the better the model works. At time  $i$ , we use the mean of the metrics tested on all test sets at the time  $[1, i]$  as the final metric. The main results are obtained from the model generated at the last time.

我们在链路预测任务上评估模型性能。具体地, 我们将测试集中三元组的头实体或尾实体替换为所有其他实体, 然后为每个三元组计算并排序得分。随后, 我们计算 MRR、Hits@1 和 Hits@10 作为度量。MRR、Hits@1、Hits@3 和 Hits@10 值越高, 模型表现越好。在时间  $i$ , 我们采用在时间  $[1, i]$  上对所有测试集测得的度量的平均值作为最终指标。主要结果来自最后时间生成的模型。

**Settings** All experiments are implemented on the NVIDIA RTX 3090Ti GPU with the PyTorch (Paszke et al. 2019). In all experiments, we set TransE (Bordes et al. 2013) as the base KGE model and the max size of time  $i$  as 5. The embedding size for entities and relations is 200. We tune the batch size in  $[512, 1024, 2048]$ . We choose Adam as the optimizer and set the learning rate from  $[1e-5, 1e-4, 1e-3]$ . In our experiments, we set the max number of triples in each layer  $M$  in  $[512, 1024, 2048]$ . To ensure fairness, all experimental results are averages of 5 running times.

设置所有实验在配备 PyTorch (Paszke et al. 2019) 的 NVIDIA RTX 3090Ti GPU 上实现。在所有实验中, 我们将 TransE (Bordes et al. 2013) 设为基础 KGE 模型, 时间的最大大小  $i$  设为 5。实体和关系的嵌入维度为 200。我们在  $[512, 1024, 2048]$  中调节批量大小。优化器选择 Adam, 学习率在  $[1e-5, 1e-4, 1e-3]$  中设置。在实验中, 我们将每层的最大三元组数  $M$  设在  $[512, 1024, 2048]$ 。为确保公平, 所有实验结果均为 5 次运行的平均值。

## Results

### 结果

Main Results The results of the main experiments on the seven datasets are reported in Table 2 and Table 3.

主要结果七个数据集上的主要实验结果列于表 2 和表 3。

Firstly, it is worth noting that IncDE exhibits a considerable improvement when compared to Fine-tune. Specifically, IncDE demonstrates enhancements ranging from 2.9%-10.6% in MRR, 2.4%-7.2% in Hits@1, and 3.7%- 17.5% in Hits@10 compared to Fine-tune. The results suggest that direct fine-tuning leads to catastrophic forgetting.

首先，值得注意的是，与 Fine-tune 相比，IncDE 展现出明显提升。具体而言，IncDE 在 MRR 上比 Fine-tune 提升 2.9%–10.6%，在 Hits@1 上提升 2.4%–7.2%，在 Hits@10 上提升 3.7%–17.5%。结果表明直接微调会导致灾难性遗忘。

Method	ENTITY			RELATION			FACT			HYBRID			GraphEqual		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
Fine-tune	0.165	0.085	0.321	0.093	0.039	0.195	0.172	0.090	0.339	0.135	0.069	0.262	0.183	0.096	0.358
PNN	0.229	0.130	0.425	0.167	0.096	0.305	0.157	0.084	0.290	0.185	0.101	0.349	0.212	0.118	0.405
CWR	0.088	0.028	0.202	0.021	0.010	0.043	0.083	0.030	0.192	0.037	0.015	0.077	0.122	0.041	0.277
GEM	0.165	0.085	0.321	0.093	0.040	0.196	0.175	0.092	0.345	0.136	0.070	0.263	0.189	0.099	0.372
EMR	0.171	0.090	0.330	0.111	0.052	0.225	0.171	0.090	0.337	0.141	0.073	0.267	0.185	0.099	0.359
DiCGRL	0.107	0.057	0.211	0.133	0.079	0.241	0.162	0.084	0.320	0.149	0.083	0.277	0.104	0.040	0.226
SI	0.154	0.072	0.311	0.113	0.055	0.224	0.172	0.088	0.343	0.111	0.049	0.229	0.179	0.092	0.353
EWC	0.229	0.130	0.423	0.165	0.093	0.306	0.201	0.113	0.382	0.186	0.102	0.350	0.207	0.113	0.400
LKGE	0.234	0.136	0.425	0.192	0.106	0.366	0.210	0.122	0.387	0.207	0.121	0.379	0.214	0.118	0.407
IncDE	0.253	0.151	0.448	0.199	0.111	0.370	0.216	0.128	0.391	0.224	0.131	0.401	0.234	0.134	0.432

方法	实体			关系			事实			混合			图相等		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
微调	0.165	0.085	0.321	0.093	0.039	0.195	0.172	0.090	0.339	0.135	0.069	0.262	0.183	0.096	0.358
PNN	0.229	0.130	0.425	0.167	0.096	0.305	0.157	0.084	0.290	0.185	0.101	0.349	0.212	0.118	0.405
CWR	0.088	0.028	0.202	0.021	0.010	0.043	0.083	0.030	0.192	0.037	0.015	0.077	0.122	0.041	0.277
GEM	0.165	0.085	0.321	0.093	0.040	0.196	0.175	0.092	0.345	0.136	0.070	0.263	0.189	0.099	0.372
EMR	0.171	0.090	0.330	0.111	0.052	0.225	0.171	0.090	0.337	0.141	0.073	0.267	0.185	0.099	0.359
DiCGRL	0.107	0.057	0.211	0.133	0.079	0.241	0.162	0.084	0.320	0.149	0.083	0.277	0.104	0.040	0.226
SI	0.154	0.072	0.311	0.113	0.055	0.224	0.172	0.088	0.343	0.111	0.049	0.229	0.179	0.092	0.353
EWC	0.229	0.130	0.423	0.165	0.093	0.306	0.201	0.113	0.382	0.186	0.102	0.350	0.207	0.113	0.400
LKGE	0.234	0.136	0.425	0.192	0.106	0.366	0.210	0.122	0.387	0.207	0.121	0.379	0.214	0.118	0.407
IncDE	0.253	0.151	0.448	0.199	0.111	0.370	0.216	0.128	0.391	0.224	0.131	0.401	0.234	0.134	0.432

Table 2: Main experimental results on ENTITY, RELATION, FACT, HYBRID, and GraphEqual. The bold scores indicate the best results and underlined scores indicate the second best results.

表 2: 在 ENTITY、RELATION、FACT、HYBRID 和 GraphEqual 上的主要实验结果。加粗分数为最好结果，带下划线的分数为第二好结果。



Method	GraphHigher			GraphLower		
	MRR	H@1	H@10	MRR	H@1	H@10
Fine-tune	0.198	0.108	0.375	0.185	0.098	0.363
PNN	0.186	0.097	0.364	0.213	0.119	0.407
CWR	0.189	0.096	0.374	0.032	0.005	0.080
GEM	0.197	0.109	0.372	0.170	0.084	0.346
EMR	0.202	0.113	0.379	0.188	0.101	0.362
DiCGRL	0.116	0.041	0.242	0.102	0.039	0.222
SI	0.190	0.099	0.371	0.186	0.099	0.366
EWC	0.198	0.106	0.385	0.210	0.116	0.405
LKGE	0.207	0.120	0.382	0.210	0.116	0.403
IncDE	0.227	0.132	0.412	0.228	0.129	0.426

方法	图-更高			图-更低		
	MRR	H@1	H@10	MRR	H@1	H@10
微调	0.198	0.108	0.375	0.185	0.098	0.363
PNN	0.186	0.097	0.364	0.213	0.119	0.407
CWR	0.189	0.096	0.374	0.032	0.005	0.080
GEM	0.197	0.109	0.372	0.170	0.084	0.346
EMR	0.202	0.113	0.379	0.188	0.101	0.362
DiCGRL	0.116	0.041	0.242	0.102	0.039	0.222
SI	0.190	0.099	0.371	0.186	0.099	0.366
EWC	0.198	0.106	0.385	0.210	0.116	0.405
LKGE	0.207	0.120	0.382	0.210	0.116	0.403
IncDE	0.227	0.132	0.412	0.228	0.129	0.426

Table 3: Main experimental results on GraphHigher and GraphLower.

表 3:GraphHigher 与 GraphLower 上的主要实验结果。

Secondly, IncDE outperforms all CKGE baselines. Notably, IncDE achieves improvements of 1.5%-19.6%, 1.0%- 12.4%, and 1.9%-34.6%, respectively, in MRR, Hits@1, and Hits@10 compared to dynamic architecture-based approaches (PNN and CWR). Compared to replay-based baselines (GEM, EMR, and DiCGRL), IncDE improves 2.5%- 14.6%, 1.9%-9.4%, and 3.3%-23.7% in MRR, Hits@1, and Hits @ 10. Moreover, IncDE obtains 0.6%-11.3%, 0.5%- 8.2%, and 0.4%-17.2% improvements in MRR, Hits@1, and Hits@10 compared to regularization-based methods (SI, EWC, and LKGE). These results demonstrate the superior performance of IncDE on growing KGs.

其次，IncDE 优于所有 CKGE 基线。值得注意的是，与基于动态架构的方法 (PNN 和 CWR) 相比，IncDE 在 MRR、Hits@1 和 Hits@10 上分别提升了 1.5%-19.6%、1.0%-12.4% 和 1.9%-34.6%。与基于重放的基线 (GEM、EMR 和 DiCGRL) 相比，IncDE 在 MRR、Hits@1 和 Hits@10 上分别提高了 2.5%-14.6%、1.9%-9.4% 和 3.3%-23.7%。此外，与基于正则化的方法 (SI、EWC 和 LKGE) 相比，IncDE 在 MRR、Hits@1 和 Hits@10 上分别获得了 0.6%-11.3%、0.5%-8.2% 和 0.4%-17.2% 的提升。这些结果证明了 IncDE 在增长中的知识图谱上的优越性能。

Thirdly, IncDE exhibits distinct improvements across different types of datasets when compared to the strong baselines. In datasets with equal growth of knowledge (ENTITY, FACT, RELATION, HYBRID, and GraphEqual), IncDE has an average improvement of 1.4% in MRR over the state-of-the-art methods. In datasets with unequal growth of knowledge (GraphHigher and GraphLower), IncDE demonstrates an improvement of 1.8%-2.0% in MRR over the optimal methods. It means that IncDE is particularly well-suited for scenarios involving unequal knowledge growth. Notably, when dealing with a more real-scenario-aware dataset, GraphHigher, where a substantial amount of new knowledge emerges, IncDE demonstrates the most apparent advantages compared to other strongest baselines by 2.0% in MRR. It indicates that IncDE performs well when a substantial amount of new knowledge is emerging. Therefore, we verify the scalability of IncDE in datasets (GraphHigher, GraphLower, and GraphEqual) with varying sizes (triples from 10K to 160K, from 160K to 10K, and the remaining 62K). In particular, we observe that IncDE only improves by 0.6%-0.7% in MRR on RELATION and FACT compared to the best results among all baselines, where the improvements are insignificant as other datasets. This can be attributed to the limited growth of new entities in these two datasets, indicating that IncDE is highly adaptable to situations where the number of entities varies significantly. In real life, the number of relations between entities remains relatively stable, while it is the entities themselves that appear in large numbers. This is where IncDE excels in its adaptability. With its robust capabilities, IncDE can effectively handle the multitude of entities and their corresponding relations, ensuring seamless integration and efficient processing.

第三，与强基线相比，IncDE 在不同类型的数据集上表现出显著提升。在知识均衡增长的数据集 (ENTITY、FACT、RELATION、HYBRID 和 GraphEqual) 中，IncDE 在 MRR 上平均较最先进方法提高了 1.4%。在知识不均衡增长的数据集 (GraphHigher 和 GraphLower) 中，IncDE 在 MRR 上较最优方法提升了 1.8%-2.0%，表明 IncDE 尤其适用于知识不均衡增长的场景。值得注意的是，在更贴近真实场景的数据集 GraphHigher(大量新知识涌现) 上，IncDE 相较其他最强基线在 MRR 上优势最明显，达到 2.0%。这表明当大量新知识出现时，IncDE 的表现优异。因此，我们在不同规模的数据集 (GraphHigher、GraphLower 和 GraphEqual，三者的三元组分别从 10K 到 160K、从 160K 到 10K 以及剩余的 62K) 上验证了 IncDE 的可扩展性。尤其是，在 RELATION 和 FACT 上，IncDE 相较所有基线的最好结果仅在 MRR 上提升了 0.6%-0.7%，比其他数据集的提升不显著。这可归因于这两个数据集中新增实体的增长有限，表明 IncDE 对实体数量显著变化的情形具有良好适应性。在现实中，实体间的关系数目相对稳定，而大量出现的正是实体本身，这正是 IncDE 适应性强的地方。凭借其强健能力，IncDE 能有效处理大量实体及其对应关系，确保无缝整合与高效处理。

**Ablation Experiments** We investigate the effects of hierarchical ordering, incremental distillation, and the two-stage training strategy, as depicted in Table 4 and Table 5. Firstly, when we remove the incremental distillation, there is a significant decrease in the model performance. Specifically, the metrics decrease by 0.2%-6.5% in MRR, 0.1%-5.2% in Hits@1, and 0.2%-11.6% in Hits@10. These findings highlight the crucial role of incremental distillation in effectively preserving the structure of the old graph while simultaneously learning the representation of the new graph. Secondly, there is a slight decline in model performance when we eliminate the hierarchical ordering and two-stage training strategy. Specifically, the metrics of MRR decreased by 0.2%-1.8%, Hits@1 decreased by 0.1%-1.8%, and Hits@10 decreased by 0.2%-4.4%. The results show that the hierarchical ordering and the two-stage training improve the performance of IncDE.

消融实验我们研究了层级排序、增量蒸馏和两阶段训练策略的作用，如表 4 和表 5 所示。首先，当去除增量蒸馏时，模型性能显著下降。具体而言，MRR 下降了 0.2%-6.5%，Hits@1 下降了 0.1%-5.2%，Hits@10 下降了 0.2%-11.6%。这些发现凸显了增量蒸馏在有效保留旧图结构的同时学习新图表示方面的关键作用。其次，去除层级排序和两阶段训练策略时，模型性能略有下降。具体而言，MRR 下降了 0.2%-1.8%，Hits@1 下降了 0.1%-1.8%，Hits@10 下降了 0.2%-4.4%。结果表明，层级排序和两阶段训练提升了 IncDE 的性能。

**Performance of IncDE in Each Time** Figure 3 shows how well IncDE remembers old knowledge at different times. First, we observe that on several test data (D1, D2, D3, D4 in ENTITY; D3, D4 in HYBRID), the performance of IncDE decreases slightly by 0.2%-3.1% with increasing time. In particular, the performance of IncDE does not undergo significant degradation on several datasets, such as D1 of HYBRID (Time 2 to Time 4) and D2 of GraphLower (Time 2 to Time 5). It means that IncDE can remember old knowledge well on most datasets. Second, on a few datasets, the performance of IncDE unexpectedly gains as it continues to be trained. Specifically, the performance of IncDE gradually increases by 0.6% on D3 of GraphLower in MRR. This demonstrates that IncDE learns emerging knowledge well and enhances the old knowledge with emerging knowledge.

IncDE 在各时间点的表现图 3 展示了 IncDE 在不同时间对旧知识的记忆情况。首先，我们观察到在若干测试数据上 (ENTITY 的 D1、D2、D3、D4；HYBRID 的 D3、D4)，随着时间增加，IncDE 的性能略微下降了 0.2%-3.1%。特别地，在若干数据集上，IncDE 的性能并未出现显著退化，例如 HYBRID 的 D1(时间 2 到时间 4) 和 GraphLower 的 D2(时间 2 到时间 5)。这表明 IncDE 能在大多数数据集上很好地记住旧知识。其次，在少数数据集上，随着持续训练，IncDE 的性能出乎意料地有所提升。具体来说，IncDE 在 GraphLower 的 D3(MRR) 上性能逐步提高了 0.6%。这说明 IncDE 能很好地学习新兴知识，并用新兴知识增强旧知识。

Method	ENTITY			RELATION			FACT			HYBRID			GraphEqual		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
IncDE w/o HO	0.248	0.148	0.441	0.186	0.105	0.344	0.197	0.119	0.347	0.210	0.122	0.380	0.230	0.131	0.426
IncDE w/o ID	0.188	0.099	0.354	0.134	0.070	0.254	0.167	0.090	0.321	0.185	0.105	0.340	0.199	0.107	0.383
IncDE w/o TS	0.250	0.149	0.444	0.186	0.099	0.354	0.213	0.126	0.389	0.220	0.127	0.397	0.231	0.132	0.430
IncDE	0.253	0.151	0.448	0.199	0.111	0.370	0.216	0.128	0.391	0.224	0.131	0.401	0.234	0.134	0.432

方法	实体			关系			事实			混合			图相等		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
不含 HO 的 IncDE	0.248	0.148	0.441	0.186	0.105	0.344	0.197	0.119	0.347	0.210	0.122	0.380	0.230	0.131	0.426
不含 ID 的 IncDE	0.188	0.099	0.354	0.134	0.070	0.254	0.167	0.090	0.321	0.185	0.105	0.340	0.199	0.107	0.383
不含 TS 的 IncDE	0.250	0.149	0.444	0.186	0.099	0.354	0.213	0.126	0.389	0.220	0.127	0.397	0.231	0.132	0.430
IncDE	0.253	0.151	0.448	0.199	0.111	0.370	0.216	0.128	0.391	0.224	0.131	0.401	0.234	0.134	0.432

Table 4: Ablation experimental results on ENTITY, RELATION, FACT, HYBRID and GraphEqual. HO is the hierarchical ordering. ID is the incremental distillation. TS is the two-stage. We learn the new KG in randomized order w/o HO.

表 4: 在 ENTITY、RELATION、FACT、HYBRID 和 GraphEqual 上的消融实验结果。HO 表示分层排序。ID 表示增量蒸馏。TS 表示两阶段。我们在无 HO 的随机顺序中学习新 KG。

Method	GraphHigher			GraphLower		
	MRR	H@1	H@10	MRR	H@1	H@10
IncDE w/o HO	0.221	0.129	0.405	0.224	0.126	0.424
IncDE w/o ID	0.225	0.131	0.410	0.196	0.105	0.377
IncDE w/o TS	0.225	0.130	0.408	0.225	0.128	0.423
IncDE	0.227	0.132	0.412	0.228	0.129	0.426

方法	图上界			图下界		
	MRR	H@1	H@10	MRR	H@1	H@10
无 HO 的 IncDE	0.221	0.129	0.405	0.224	0.126	0.424
无 ID 的 IncDE	0.225	0.131	0.410	0.196	0.105	0.377
无 TS 的 IncDE	0.225	0.130	0.408	0.225	0.128	0.423
IncDE	0.227	0.132	0.412	0.228	0.129	0.426

Table 5: Ablation experimental results on GraphHigher and GraphLower.

表 5: 在 GraphHigher 和 GraphLower 上的消融实验结果。

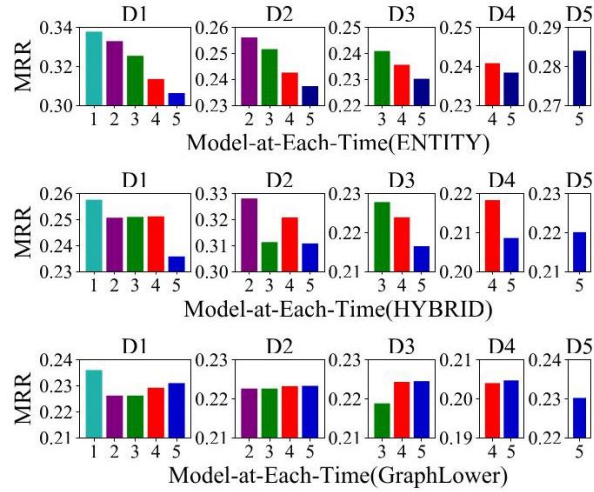


Figure 3: Effectiveness of IncDE at Each Time on ENTITY, HYBRID, and GraphLower. Different colors represent the performance of models generated at different times.  $D_i$  denotes the test set at time  $i$ .

图 3: IncDE 在 ENTITY、HYBRID 和 GraphLower 上各时间点的有效性。不同颜色代表不同时间生成的模型性能。 $D_i$  表示时间  $i$  的测试集。

**Effect of Learning and Memorizing** In order to verify that IncDE can learn emerging knowledge well and remember old knowledge efficiently, we study the effect of IncDE and Fine-tune each time on the new KG and old KGs, respectively, as shown in Figure 4. To assess the performance on old KGs, we calculated the mean value of the MRR across all past time steps. Firstly, we observe that IncDE outperforms Fine-tune on the new KG, with a higher MRR ranging from 0.5% to 5.5%. This indicates that IncDE is capable of effectively learning emerging knowledge. Secondly, IncDE has 3.8%-11.2% higher than Fine-tune on old KGs in MRR. These findings demonstrate that IncDE mitigates the issue of catastrophic forgetting and achieves more efficient retention of old knowledge.

学习与记忆的效果为验证 IncDE 能否有效学习新兴知识并高效记忆旧知识，我们分别研究了每次在新 KG 和旧 KGs 上的 IncDE 与 Fine-tune 的效果，如图 4 所示。为评估在旧 KGs 上的表现，我们计算了所有过去时间步 MRR 的平均值。首先，我们观察到 IncDE 在新 KG 上优于 Fine-tune，MRR 提高 0.5% 到 5.5%，表明 IncDE 能够有效学习新兴知识。其次，IncDE 在旧 KGs 上的 MRR 比 Fine-tune 高 3.8%–11.2%。这些结果表明 IncDE 缓解了灾难性遗忘问题，并能更高效地保留旧知识。

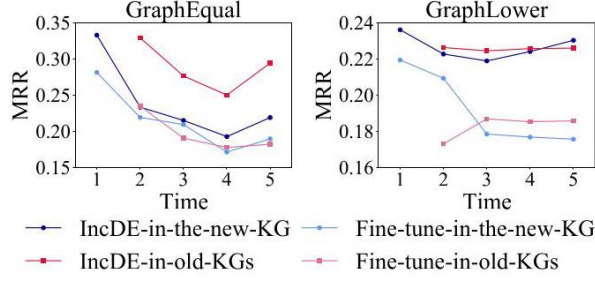


Figure 4: Effectiveness of learning emerging knowledge and memorizing old knowledge.

图 4: 学习新兴知识与记忆旧知识的有效性。

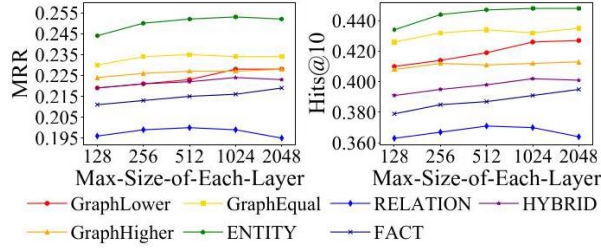


Figure 5: Results of MRR and Hits@10 with different max sizes of layers in all datasets.

图 5: 在所有数据集中，不同最大层大小下的 MRR 与 Hits@10 结果。

**Effect of Maximum Layer Sizes** To investigate the effect of the max size of each layer  $M$  in incremental distillation on model performance, we study the performances of In-cDE models at the last time with different  $M$ , as shown in Figure 5. First, we find that the model performance on all datasets rises with  $M$  in the range of  $[128, 1024]$ . This indicates that, in general, the higher  $M$ , the more influential the incremental distillation becomes. Second, we observe a significant performance drop on some datasets when  $M$  reaches 2048. It implies that too large an  $M$  could lead to too few layers and limit the performance of incremental distillation. Empirically,  $M = 1024$  is the best size in most datasets. This further proves that it is necessary to limit the number of triples learned in each layer.

最大层大小的影响为探究增量蒸馏中每层最大大小  $M$  对模型性能的影响，我们研究了最后时刻不同  $M$  下 In-cDE 模型的性能，如图 5 所示。首先，我们发现所有数据集的模型性能在  $M$  位于 [128, 1024] 范围内随  $M$  增大而上升。这表明一般而言， $M$  越大，增量蒸馏的影响越显著。其次，当  $M$  达到 2048 时，我们在部分数据集上观察到显著的性能下降，说明过大的  $M$  可能导致层数过少，从而限制增量蒸馏的性能。经验上， $M = 1024$  在大多数数据集中为最佳大小。这进一步证明了限制每层学习三元组数量的必要性。

Query	(Arizona State University, major_field_of_study, ?)
Methods	Top 3 Candidates
EWC	Medicine, Electrical engineering, Computer Science
PNN	Medicine, Electrical engineering, Computer Science
LKGE	English Literature, Computer Science, Political Science
IncDE	Computer Science, University of Tehran, Medicine
w/o HO	Computer Science, Medicine, University of Tehran
w/o ID	Political Science, English Literature, Theatre
w/o TS	Computer Science, Medicine, University of Tehran

查询	(Arizona State University, major_field_of_study, ?)
方法	前三候选项
EWC	医学, 电气工程, 计算机科学
PNN	医学, 电气工程, 计算机科学
LKGE	英国文学, 计算机科学, 政治学
IncDE	计算机科学, 德黑兰大学, 医学
不含 HO	计算机科学, 医学, 德黑兰大学
不含 ID	政治学, 英国文学, 戏剧
不含 TS	计算机科学, 医学, 德黑兰大学

Table 6: Results of the case study. We use the model generated at time 5 and randomly select a query appearing in ENTITY at time 1 for prediction. The italic one is the query, and the bold ones are true prediction results.

表 6: 案例研究结果。我们使用时间 5 生成的模型并随机选择在时间 1 出现在 ENTITY 中用于预测的查询。斜体为查询，粗体为真实的预测结果。

Case Study To further explore the capacity of IncDE to preserve old knowledge, we conduct a comprehensive case study as shown in Table 6. In the case of predicting the major field of study of Arizona State University, IncDE ranks the correct answer Computer Science in the first position, outperforming other strong baselines such as EWC, PNN, and LKGE, which rank it second or third. It indicates that although other methods forget knowledge in the past time to some degree, IncDE can remember old knowledge at each time accurately. Moreover, when incremental distillation (ID) is removed, IncDE fails to predict the correct answer within the top three positions. This demonstrates that the performance of IncDE significantly declines when predicting old knowledge without the incremental distillation. Conversely, after removing hierarchical ordering (HO) and the two-stage training strategy (TS), IncDE still accurately predicts the correct answer in the first position. This observation strongly supports the fact that the incremental distillation provides IncDE with a crucial advantage over alternative strong baselines in preserving the old knowledge.

案例研究为进一步探讨 IncDE 保留旧知识的能力, 我们进行了如表 6 所示的全面案例研究。在预测亚利桑那州立大学的主修领域时, IncDE 将正确答案计算机科学排在第一位, 优于将其排在第二或第三位的其他强基线 (如 EWC、PNN 和 LKGE)。这表明尽管其他方法在过去的时间段中在一定程度上遗忘了知识, IncDE 能在每个时间准确记住旧知识。此外, 当去掉增量蒸馏 (ID) 时, IncDE 无法在前三名内预测出正确答案。这表明在没有增量蒸馏的情况下, IncDE 在预测旧知识时性能显著下降。相反, 在去掉分层排序 (HO) 和两阶段训练策略 (TS) 之后, IncDE 仍能将正确答案准确预测在第一位。这一观察有力地支持了增量蒸馏使 IncDE 在保留旧知识方面相比其他强基线拥有关键优势的事实。

## Discussion

### 讨论

**Novelty of IncDE** The novelty of IncDE can be summarized by the following two aspects. (1) Efficient knowledge-preserving distillation. Although IncDE utilizes distillation methods, it is different from previous KGE distillation methods (Wang et al. 2021; Zhu et al. 2022; Liu et al. 2023). For one thing, compared to other KGE distillation methods that mainly distill final distribution, incremental distillation (ID) distills the intermediate hidden states. Such a manner skillfully preserves essential features of old knowledge, making it adaptable to various downstream tasks. For another thing, only ID transfers knowledge from the model itself, thus mitigating error propagation compared to transferring knowledge from other models. (2) Explicit graph-aware mechanism. Compared to other CKGE baselines, IncDE stands out by incorporating the graph structure into continual learning. This explicit graph-aware mechanism allows IncDE to leverage the inherent semantics encoded within the graph, enabling it to intelligently determine the optimal learning order and effectively balance the preservation of old knowledge.

**IncDE 的新颖性** IncDE 的新颖性可归纳为以下两点。(1) 高效的知识保留蒸馏。尽管 IncDE 使用了蒸馏方法, 但它不同于以往的 KGE 蒸馏方法 (Wang 等, 2021; Zhu 等, 2022; Liu 等, 2023)。一方面, 与主要蒸馏最终分布的其他 KGE 蒸馏方法相比, 增量蒸馏 (ID) 蒸馏的是中间隐藏状态。此类方式巧妙地保留了旧知识的关键特征, 使其适应多种下游任务。另一方面, ID 仅从模型自身转移知识, 相较于从其他模型转移知识可减轻误差传播。(2) 显式图感知机制。与其他 CKGE 基线相比, IncDE 的突出之处在于将图结构纳入持续学习。该显式图感知机制使 IncDE 能利用图中编码的内在语义, 智能地确定最优学习顺序并有效平衡旧知识的保留。

**Three Components in IncDE** The three components of IncDE, hierarchical ordering (HO), incremental distillation (ID), and two-stage training (TS) are inherently dependent on each other and necessary to be used together. We explain it in the following two aspects. (1) Designing Principle. The fundamental motivation of IncDE lies in effectively learning emerging knowledge while simultaneously preserving old knowledge. This objective is accomplished by all three components: HO, ID, and TS. On the one hand, HO plays a role in dividing new triples into layers, optimizing the process of learning emerging knowledge. On the other hand, ID and TS try to distill and preserve the representation of entities, ensuring the effective preservation of old knowledge. (2) Inter Dependence. The three components are intrinsically interrelated and should be employed together. For one thing, HO plays a vital role in generating a partition of new triples, which are subsequently fed into ID. For another thing, by employing TS, ID prevents old entities from being disrupted in the early training stages.

IncDE 的三大组成部分 IncDE 的三大组成部分——分层排序 (HO)、增量蒸馏 (ID) 和两阶段训练 (TS) 本质上相互依赖, 必须共同使用。我们从以下两方面说明。(1) 设计原则。IncDE 的根本动机在于在有效学习新兴知识的同时保留旧知识。该目标由三部分共同实现:HO、ID 和 TS。一方面, HO 在将新三元组划分为多个层次方面发挥作用, 优化学习新兴知识的过程。另一方面, ID 和 TS 力求蒸馏并保留实体表示, 确保旧知识的有效保存。(2) 相互依赖性。三部分本质上相互关联, 应当协同使用。一方面, HO 在生成新三元组的划分中起关键作用, 这些划分随后被输入到 ID。另一方面, 通过采用 TS, ID 可防止旧实体在训练早期阶段受到干扰。

**Significance of Incremental Distillation** Even though the three proposed components of IncDE: incremental distillation (ID), hierarchical ordering (HO), and two-stage training (TS) are all effective for the CKGE task, ID serves as the central module among them. Theoretically, the primary challenge in the continual learning task is catastrophic forgetting that occurs when learning step by step, which is also suitable for the CKGE task. To tackle this challenge, ID introduces the explicit graph structure to distill entity representations, effectively preserving old knowledge layer by layer during the whole training time. However, HO focuses on learning new knowledge well, and TS can only alleviate catastrophic forgetting in the early stages of training. Therefore, ID plays the most important role among all components in the CKGE task. In experiments, we observe that ID exhibits significant improvements (4.1% in MRR on average) compared to HO (0.9% in MRR on average) and TS (0.5% in MRR on average) from Table 4 and Table 5. Such results further verify ID as the pivotal component compared with HO and TS. The three components interact with each other and work together to complete the CKGE task.

增量蒸馏的重要性尽管 IncDE 提出的三项组件: 增量蒸馏 (ID)、分层排序 (HO) 和两阶段训练 (TS) 对 CKGE 任务均有效, ID 是其中的核心模块。从理论上讲, 持续学习任务的主要挑战是逐步学习时发生的灾难性遗忘, 这同样适用于 CKGE 任务。为应对这一挑战, ID 引入显式图结构来蒸馏实体表示, 在整个训练过程中逐层有效地保留旧知识。然而, HO 专注于更好地学习新知识, TS 只能在训练的早期阶段缓解灾难性遗忘。因此, 在 CKGE 任务中, ID 在所有组件中起到最重要的作用。在实验中, 我们观察到与 HO(平均 MRR 提升 0.9%) 和 TS(平均 MRR 提升 0.5%) 相比, ID 带来了显著改进 (平均 MRR 提升 4.1%), 见表 4 和表 5。此类结果进一步验证了 ID 相较于 HO 和 TS 的关键地位。三者相互作用、协同工作以完成 CKGE 任务。

## Conclusion

### 结论

This paper proposes a novel continual knowledge graph embedding method, IncDE, which incorporates the graph structure of KGs in learning emerging knowledge and remembering old knowledge. Firstly, we perform hierarchical ordering for the triples in the new knowledge graph to get an optimal learning sequence. Secondly, we propose incremental distillation to preserve old knowledge when training the new triples layer by layer. Moreover, We optimize the training process with a two-stage training strategy. In the future, we will consider how to handle the situation where old knowledge is deleted as knowledge graphs evolve. Also, it is imperative to address the integration of cross-domain and heterogeneous data into expanding knowledge graphs.



本文提出了一种新颖的持续知识图谱嵌入方法 IncDE, 该方法在学习新兴知识和记忆旧知识时融合了知识图谱的图结构。首先, 我们对新知识图谱中的三元组进行分层排序以获得最优学习序列。其次, 我们提出增量蒸馏以在分层训练新三元组时保留旧知识。此外, 我们用两阶段训练策略优化训练过程。未来, 我们将考虑在知识图谱演化中如何处理旧知识被删除的情况。同时, 必须解决将跨域和异构数据整合入扩展知识图谱的问题。

## Acknowledgments

### 致谢

We thank the anonymous reviewers for their insightful comments. This work was supported by National Science Foundation of China (Grant Nos.62376057) and the Start-up Research Fund of Southeast University (RF1028623234). All opinions are of the authors and do not reflect the view of sponsors.

我们感谢匿名评审的见解性意见。该工作由中国国家自然科学基金 (项目编号 62376057) 和东南大学创业研究基金 (RF1028623234) 资助。以上观点仅代表作者, 不代表资助方立场。

## References

### 参考文献

Bizer, C.; Lehmann, J.; Kobilarov, G.; Auer, S.; Becker, C.; Cyganiak, R.; and Hellmann, S. 2009. Dbpedia—a crystallization point for the web of data. *Journal of web semantics*, 7(3): 154-165.

Bizer, C.; Lehmann, J.; Kobilarov, G.; Auer, S.; Becker, C.; Cyganiak, R.; and Hellmann, S. 2009. Dbpedia——数据网的结晶点。 *Journal of web semantics*, 7(3): 154-165.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. 将嵌入用于建模多关系数据。 In *NIPS*.

Bordes, A.; Weston, J.; and Usunier, N. 2014. Open question answering with weakly supervised embedding models. In *ECML-PKDD*.

Bordes, A.; Weston, J.; and Usunier, N. 2014. 使用弱监督嵌入模型的开放式问答。 In *ECML-PKDD*.

Cui, Y.; Wang, Y.; Sun, Z.; Liu, W.; Jiang, Y.; Han, K.; and Hu, W. 2023. Lifelong embedding learning and transfer for growing knowledge graphs. In *AAAI*.

Cui, Y.; Wang, Y.; Sun, Z.; Liu, W.; Jiang, Y.; Han, K.; and Hu, W. 2023. 面向增长知识图谱的终身嵌入学习与迁移。 In *AAAI*.

Daruna, A.; Gupta, M.; Sridharan, M.; and Chernova, S. 2021. Continual learning of knowledge graph embeddings. *IEEE Robotics and Automation Letters*, 6(2): 1128-1135.

Daruna, A.; Gupta, M.; Sridharan, M.; and Chernova, S. 2021. 持续学习知识图谱嵌入。 *IEEE Robotics and Automation Letters*, 6(2): 1128-1135.

DBpedia. 2021. DBpedia - A community-driven knowledge graph. <https://wiki.dbpedia.org/>. Accessed: 2023-08-01.

DBpedia. 2021. DBpedia - 一个社区驱动的知识图谱。 <https://wiki.dbpedia.org/>. Accessed: 2023-08-01.

Dong, X.; Gabrilovich, E.; Heitz, G.; Horn, W.; Lao, N.; Murphy, K.; Strohmman, T.; Sun, S.; and Zhang, W. 2014. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *SIGKDD*.

Dong, X.; Gabrilovich, E.; Heitz, G.; Horn, W.; Lao, N.; Murphy, K.; Strohmman, T.; Sun, S.; and Zhang, W. 2014. 知识金库: 一种面向网络规模的概率知识融合方法。 In *SIGKDD*.

Febrinanto, F. G.; Xia, F.; Moore, K.; Thapa, C.; and Aggarwal, C. 2023. Graph lifelong learning: A survey. *IEEE Computational Intelligence Magazine*, 18(1): 32-51.

Febrinanto, F. G.; Xia, F.; Moore, K.; Thapa, C.; and Aggarwal, C. 2023. 图终身学习: 综述。 *IEEE Computational Intelligence Magazine*, 18(1): 32-51.

Hamaguchi, T.; Oiwa, H.; Shimbo, M.; and Matsumoto, Y. 2017. Knowledge transfer for out-of-knowledge-base entities: a graph neural network approach. In *IJCAI*.

Hamaguchi, T.; Oiwa, H.; Shimbo, M.; and Matsumoto, Y. 2017. 将知识转移到知识库外实体: 一种图神经网络方法。 In *IJCAI*.

Kazemi, S. M.; and Poole, D. 2018. Simple embedding for link prediction in knowledge graphs. *NeurIPS*.

Kazemi, S. M.; and Poole, D. 2018. 用于知识图链接预测的简单嵌入。 *NeurIPS*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Des-jardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13): 3521-3526.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Des-jardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; et al. 2017. 克服神经网络的灾难性遗忘。 *Proceedings of the national academy of sciences*, 114(13): 3521-3526.

Kou, X.; Lin, Y.; Liu, S.; Li, P.; Zhou, J.; and Zhang, Y. 2020. Disentangle-based Continual Graph Representation Learning. In *EMNLP*.

Kou, X.; Lin, Y.; Liu, S.; Li, P.; Zhou, J.; and Zhang, Y. 2020. 基于解缠的持续图表示学习。 In *EMNLP*.

Li, G.; Chen, X.; Wang, P.; Xie, J.; and Luo, Q. 2022. Fastre: Towards fast relation extraction with convolutional encoder and improved cascade binary tagging framework. In IJCAI.

Li, G.; Chen, X.; Wang, P.; Xie, J.; and Luo, Q. 2022. Fastre: 借助卷积编码器与改进的级联二元标注框架实现快速关系抽取。In IJCAI.

Liang, K.; Meng, L.; Liu, M.; Liu, Y.; Tu, W.; Wang, S.; Zhou, S.; Liu, X.; and Sun, F. 2022. A Survey of Knowledge Graph Reasoning on Graph Types: Static, Dynamic, and Multimodal.

Liang, K.; Meng, L.; Liu, M.; Liu, Y.; Tu, W.; Wang, S.; Zhou, S.; Liu, X.; and Sun, F. 2022. 不同图类型上的知识图推理综述: 静态、动态与多模态。

Liu, H.; Yang, Y.; and Wang, X. 2021. Overcoming catastrophic forgetting in graph neural networks. In AAAI.

Liu, H.; Yang, Y.; and Wang, X. 2021. 克服图神经网络中的灾难性遗忘。In AAAI.

Liu, J.; Wang, P.; Shang, Z.; and Wu, C. 2023. IterDE: An Iterative Knowledge Distillation Framework for Knowledge Graph Embeddings. In AAAI.

Liu, J.; Wang, P.; Shang, Z.; and Wu, C. 2023. IterDE: 一种用于知识图嵌入的迭代知识蒸馏框架。In AAAI.

Liu, Y.; Wang, P.; Li, Y.; Shao, Y.; and Xu, Z. 2020. AprilE: Attention with pseudo residual connection for knowledge graph embedding. In COLING.

Liu, Y.; Wang, P.; Li, Y.; Shao, Y.; and Xu, Z. 2020. AprilE: 用于知识图嵌入的伪残差连接注意力。In COLING.

Lomonaco, V.; and Maltoni, D. 2017. Core50: a new dataset and benchmark for continuous object recognition. In CoRL.

Lomonaco, V.; and Maltoni, D. 2017. Core50: 用于持续对象识别的新数据集与基准。In CoRL.

Lopez-Paz, D.; and Ranzato, M. 2017. Gradient episodic memory for continual learning. In NeurIPS.

Lopez-Paz, D.; and Ranzato, M. 2017. 用于持续学习的梯度历时记忆。In NeurIPS.

Noy, N.; Gao, Y.; Jain, A.; Narayanan, A.; Patterson, A.; and Taylor, J. 2019. Industry-scale Knowledge Graphs: Lessons and Challenges: Five diverse technology companies show how it's done. Queue, 17(2): 48-75.

Noy, N.; Gao, Y.; Jain, A.; Narayanan, A.; Patterson, A.; and Taylor, J. 2019. 工业规模知识图: 教训与挑战: 五家不同技术公司展示了实现方法。Queue, 17(2): 48-75.

Pan, Z.; and Wang, P. 2021. Hyperbolic hierarchy-aware knowledge graph embedding for link prediction. In EMNLP.

Pan, Z.; and Wang, P. 2021. 面向链路预测的双曲层级感知知识图嵌入。 In EMNLP.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; and et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In NeurIPS.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; and et al. 2019. PyTorch: 一种命令式风格的高性能深度学习库。 In NeurIPS.

Rossi, A.; Barbosa, D.; Firmani, D.; Martinata, A.; and Meri-aldo, P. 2021. Knowledge graph embedding for link prediction: A comparative analysis. ACM Transactions on Knowledge Discovery from Data (TKDD), 15(2): 1-49.

Rossi, A.; Barbosa, D.; Firmani, D.; Martinata, A.; and Meri-aldo, P. 2021. 用于链路预测的知识图嵌入: 比较分析。 ACM Transactions on Knowledge Discovery from Data (TKDD), 15(2): 1-49.

Rusu, A. A.; Rabinowitz, N. C.; Desjardins, G.; Soyer, H.; Kirkpatrick, J.; Kavukcuoglu, K.; Pascanu, R.; and Had-sell, R. 2016. Progressive neural networks. arXiv preprint arXiv:1606.04671.

Rusu, A. A.; Rabinowitz, N. C.; Desjardins, G.; Soyer, H.; Kirkpatrick, J.; Kavukcuoglu, K.; Pascanu, R.; and Had-sell, R. 2016. 进阶神经网络。 arXiv preprint arXiv:1606.04671.

Shang, Z.; Wang, P.; Liu, Y.; Liu, J.; and Ke, W. 2023. ASKRL: An Aligned-Spatial Knowledge Representation Learning Framework for Open-World Knowledge Graph. In ISWC.

Shang, Z.; Wang, P.; Liu, Y.; Liu, J.; and Ke, W. 2023. ASKRL: 一种面向开放世界知识图的对齐空间知识表示学习框架。 In ISWC.

Song, H.-J.; and Park, S.-B. 2018. Enriching translation-based knowledge graph embeddings through continual learning. IEEE Access, 6: 60489-60497.

Song, H.-J.; and Park, S.-B. 2018. 通过持续学习丰富基于平移的知识图嵌入。 IEEE Access, 6: 60489-60497.

Suchanek, F. M.; Kasneci, G.; and Weikum, G. 2007. Yago: A core of semantic knowledge. In WWW.

Suchanek, F. M.; Kasneci, G.; and Weikum, G. 2007. Yago: 语义知识核心。 In WWW.

Sun, Z.; Deng, Z.-H.; Nie, J.-Y.; and Tang, J. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In ICLR.

Sun, Z.; Deng, Z.-H.; Nie, J.-Y.; and Tang, J. 2019. RotatE: 通过复数空间的关系旋转进行知识图嵌入。 In ICLR.

Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; and Bouchard, G. 2016. Complex embeddings for simple link prediction. In ICML.

Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; and Bouchard, G. 2016. 复杂嵌入用于简明链路预测。 In ICML.

Wang, H.; Xiong, W.; Yu, M.; Guo, X.; Chang, S.; and Wang, W. Y. 2019. Sentence Embedding Alignment for Lifelong Relation Extraction. In NAACL.

Wang, H.; Xiong, W.; Yu, M.; Guo, X.; Chang, S.; and Wang, W. Y. 2019. 面向终身关系抽取的句子嵌入对齐。 In NAACL.

Wang, K.; Liu, Y.; Ma, Q.; and Sheng, Q. Z. 2021. Mulde: Multi-teacher knowledge distillation for low-dimensional knowledge graph embeddings. In WWW.

Wang, K.; Liu, Y.; Ma, Q.; and Sheng, Q. Z. 2021. Mulde: 面向低维知识图谱嵌入的多教师知识蒸馏。 In WWW.

Wang, Q.; Mao, Z.; Wang, B.; and Guo, L. 2017. Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering, 29(12): 2724-2743.

Wang, Q.; Mao, Z.; Wang, B.; and Guo, L. 2017. 知识图谱嵌入: 方法与应用综述。 IEEE Transactions on Knowledge and Data Engineering, 29(12): 2724-2743.

Wei, D.; Gu, Y.; Song, Y.; Song, Z.; Li, F.; and Yu, G. 2022. IncreGNN: Incremental Graph Neural Network Learning by Considering Node and Parameter Importance. In DASFAA.

Wei, D.; Gu, Y.; Song, Y.; Song, Z.; Li, F.; and Yu, G. 2022. IncreGNN: 通过考虑节点和参数重要性的增量图神经网络学习。 In DASFAA.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual learning through synaptic intelligence. In ICML.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. 通过突触智力实现持续学习。 In ICML.

Zhou, F.; and Cao, C. 2021. Overcoming catastrophic forgetting in graph neural networks with experience replay. In AAAI.

Zhou, F.; and Cao, C. 2021. 通过经验回放克服图神经网络的灾难性遗忘。 In AAAI.

Zhu, Y.; Zhang, W.; Chen, M.; Chen, H.; Cheng, X.; Zhang, W.; and Chen, H. 2022. Dualde: Dually distilling knowledge graph embedding for faster and cheaper reasoning. In WSDM.

Zhu, Y.; Zhang, W.; Chen, M.; Chen, H.; Cheng, X.; Zhang, W.; and Chen, H. 2022. Dualde: 双向蒸馏知识图谱嵌入以实现更快更廉价的推理。 In WSDM.