Towards Global-Topological Relation Graph for Inductive Knowledge Graph Completion

Ling Ding[1], Lei Huang[1], Zhizhi Yu[1*], Di Jin[1], Dongxiao He[1]

[1]College of Intelligence and Computing, Tianjin University, Tianjin, China.
{dltjdx2022, huanglei, yuzhizhi, jindi, hedongxiao}@tju.edu.cn

Abstract

Knowledge Graphs (KGs) are structured data presented as directed graphs. Due to the common issues of incompleteness and inaccuracy encountered during construction and maintenance, completing KGs becomes a critical task. Inductive Knowledge Graph Completion (KGC) excels at inferring patterns or models from seen data to be applied to unseen data. However, existing methods mainly focus on new entities, while relations are usually randomly initialized. To this end, we propose TARGI, a simple yet effective inductive method for KGC. Specifically, we first construct a global relation graph for each topology from a global graph perspective, thus leveraging the in-variance of relation structures. We then utilize this graph to aggregate the rich embeddings of new relations and new entities, thereby performing KGC robustly in inductive scenarios. This successfully addresses the excessive reliance on the degree of relations and resolves the high complexity and limited scope of enclosing subgraph sampling in existing fully inductive algorithms. We conduct KGC experiments on six inductive datasets using inference data where entities are entirely new and new relations at 100%, 50%, and 0% ratios. Extensive results demonstrate that our model accurately learns the topological structures and embeddings of new relations, and guides the embedding learning of new entities. Notably, our model outperforms 15 state-of-the-art methods, especially in two fully inductive datasets.

Introduction

Knowledge Graphs (KGs) are structured directed graph datasets, where data is organized in triples. In these graphs, nodes denote the head or tail entities, and edges capture the relations between nodes (Ji et al. 2022). KGs serve a foundational role in various applications, such as recommendation systems (Wang et al. 2019) and question-answering systems (Huang et al. 2019), as they enhance data interoperability, reasoning capabilities, and semantic query abilities (Paulheim 2017). Owing to inherent incompleteness and inaccuracies during construction and maintenance, Knowledge Graph Completion (KGC) came into being.

Traditional KGC methods primarily focus on link prediction or entity completion by leveraging existing knowledge to address missing information, such as embedding-based methods TransE (Bordes et al. 2013) and RotatE (Sun et al. 2019). However, these methods fall under transductive learning with all entities and relations known during the training and inference stages, struggling to handle the emergence of inductive scenarios with new entities and relations.

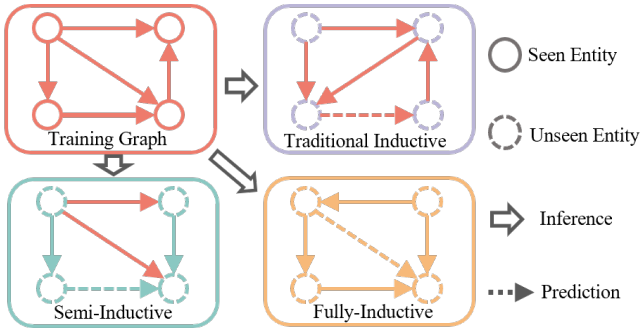

Figure 1: Inductive KGC scenario (Lee, Chung, and Whang 2023). For instance, if the inference data contains only new entities, it falls under the traditional inductive scenario. If both relations and entities are new, it means fully inductive.

(Teru, Denis, and Hamilton 2020) introduce inductive KGC to predict missing triples between new entities not observed during training. They categorize entities and relations by seen entity or relation (observed in training) and unseen entity or relation (not observed in training) (Liang et al. 2024). However, unlike handling unseen entities, existing inductive KGC methods assume that all relations are known, thus performing transductive reasoning on the relations. This assumption poses limitations in real-world applications as new relations may continue to emerge. More importantly, we find that most inductive methods focus on learning embeddings of new entities such as (Liang et al. 2023). In contrast, embeddings of relations are often obtained through random initialization or straightforward processing, inevitably overlooking the semantic correlations between relations. To this end, we emphasize a more realistic scenario of inductive learning that simultaneously addresses new relations and new entities. As shown in Figure 1, the fully-inductive scenario involves entirely unseen relations, while the semi-inductive scenario involves both seen and unseen relations during inference.

Recently, some research indicates that the semantic corre-

*Corresponding author

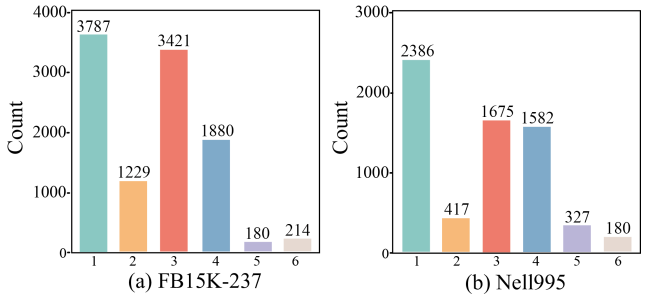

Figure 2: Topological patterns for relation pairs, where x-axis represents the number of relation pairs that simultaneously exhibit 1 to 6 types of topological structures, and the y-axis denotes the count of relation pairs.

lation between two connected relations is highly associated with their topological structure, such as TACT (Chen et al. 2021) and RMPI (Geng et al. 2023). For instance, "/people/person/nationality" is strongly related to "/people/ethnicity/languages speaking", but less so to "film/film/location". Considering this, RMPI (Geng et al. 2023) employs enclosing subgraph sampling to construct relation message-passing for handling KGC in fully inductive scenarios. However, they only consider the topological structure of the neighborhood subgraphs and ignore the global topological structures. InGram (Lee, Chung, and Whang 2023) uses the degree of relations to construct relation graphs but leads the model to favor frequently occurring relations and entities, thus affecting its generalization capability.

Moreover, we instinctively realize that employing weights to signify the complexity of relations within a relation graph, as done by InGram, is insufficient. We analyze the topology on FB15K-237 (Toutanova and Chen 2015) and NELL995 (Xiong, Hoang, and Wang 2017) datasets and find multiple distinct topological structures within the same relation pairs. As shown in Figure 2, we can observe the following: 1) The complexity of topological patterns varies significantly across different datasets. 2) Most patterns are sharply concentrated in one, three, or four types, with the mentioned single type containing any given pattern. Then, we categorize TACT-defined topological structures into two types: complex (including PARA and LOOP) and normal (others). And our further analysis indicates that complex topological patterns are fewer in number. However, in subsequent experiments, such as in certain layers of the FB-100 dataset, the attention weights of complex topological patterns significantly exceed normal patterns. This further validates our intuition and underscores the need to incorporate additional dimensions in topological structure analysis, and more accurately capture and reflect the complexity of relations.

Considering the aforementioned conclusion, we propose an inductive KGC model based on graph neural networks (GNNs) named TARGI. Specifically, we first construct global relation graphs for each topological structure. Then, we incorporate topological structures into the above relation graphs to develop global relation embeddings, which are used to update entity embeddings. Our global relation graph spans different topological patterns, which ensures that even if the relations vary between training and inference, the relation interaction graph remains consistent. When the relations differ during inference, their interactions are preserved and captured by the global relation graph. This enables our model to be effectively generalized to inductive scenarios.

The main contributions of our work are as follows:

- We construct relation graphs for each topology from the global view (entire graph), leveraging the invariance of the relation structure and using relative relation embeddings to parameterize any unseen relations.
- We propose an inductive KGC model, which is able to aggregate relation-topological graphs and the embeddings of new entities and relations in fully inductive scenarios to resolve the high complexity and limitations associated with enclosing subgraph sampling.
- We conduct extensive experiments on 6 different inductive scenarios and outperform 15 baseline models.

Preliminaries

We first give the problem formulation, and then introduce the definition of KGC and inductive learning.

Problem Formulation. A KG can be defined as $G = \{V, R, E\}$, where V is a finite set of entities, R is a finite set of relations, and E is a set of facts in the form of triples $(V \times R \times V)$, where $E = \{(v_i, r, v_j) \mid v_i, v_j \in V; r \in R\}$. v_i, r and v_j are respectively called the head entity (subject), relation (predicate), and tail entity (object) of the triple, let $|R| = m$ and $|V| = n$. Link prediction involves predicting a triple $E_{pre}(v_i, r, v_j)$, where $r \in R_{\text{inf}}$ and $v_i, v_j \in V_{inf}$.

Knowledge Graph Completion. The task of KGC (Arnaout et al. 2021) involves predicting whether a given candidate triple is true or false (i.e., triple classification) or predicting the missing entity or relation within a triple (i.e., entity/relation prediction). The prediction score for true (positive) triples is higher than that for false (negative) triples. The training graph $G_{tr} = (V_{tr}, R_{tr}, E_{tr})$ typically refers to the KG used for model training, which contains known entities, relations, and triples. (Lee, Chung, and Whang 2023) further process the traditional KG dataset by dividing E_{tr} into two disjoint sets, that is, $E_{tr} := F_{tr} \cup T_{tr}$, where F_{tr} is the set of known facts and T_{tr} is the set of triples the model needs to predict. The inference graph G_{inf} typically refers to triples composed of unseen entities and relations, represented as $G_{inf} = (V_{inf}, R_{inf}, E_{inf})$.

Inductive Learning. In the traditional inductive scenario, the inference graph includes new entities, but the relations are known and shared between the training and the inference graph. Specifically, a set of unseen entities V_{inf} (where $V_{tr} \cap V_{inf} = \emptyset$) (Teru, Denis, and Hamilton 2020) appears during prediction. While the relations are not new, they are represented as $G_{tr} = (V_{tr}, R, E_{tr})$ and $G_{inf} = (V_{inf}, R, E_{inf})$ (Zhang and Yao 2022). This implies that if $V_{train} \subseteq V_{inf}$, the inference graph can be interpreted as an extension of the training graph. And if $V_{tr} \cap V_{inf} = \emptyset$, we consider the inference graph to be a separate disjoint graph with the same set of relations. In the fully inductive KGC task, both entities and relations are new, meaning

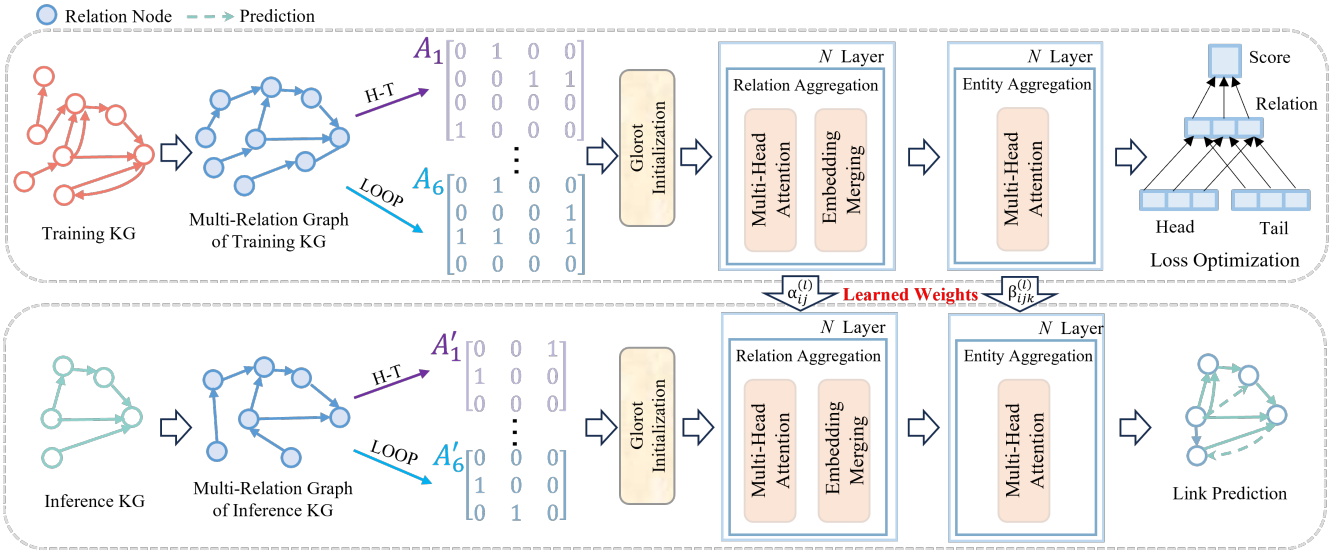

Figure 3: An overview of TARGI, which consists of three modules: 1) global-topological relation graph construction. 2) relation embedding learning. 3) entity embedding learning. In the inference stage, the reconstructed global topology graph can utilize the learned attention weights to compute embeddings for both new relations and entities, thereby enabling link prediction.

$V_{tr} \cap V_{inf} = \emptyset$ and $R_{tr} \cap R_{inf} = \emptyset$ are satisfied simultaneously. For the edges in the inference graph E_{inf}, we divide E_{inf} into three disjoint sets: $E_{inf} := F_{inf} \cup T_{val} \cup T_{test}$ in a ratio of 3:1:1. Our approach of using $F_{inf} \cup T_{val} \cup T_{test}$ is the same as the classical methods that employed by QBLP (Ali et al. 2022) and NodePiece (Galkin et al. 2022).

Methodology

We first give a brief overview of our approach, and then introduce the proposed new method in specific including detailed descriptions of each component.

Overview

To let KGC tasks essentially suitable for fully inductive scenarios, we introduce a novel global topology-guided GNNs approach, as shown in Figure 3. The core idea is to learn transferable, invariant relation graph structures from KGs accommodating partial or entirely new relations. We incorporate topological structures into relation graphs to develop global relation embeddings, which are used to update entity embeddings. During inference, a global topology graph is reconstructed to compute embeddings for unseen relations and new entities. Our method comprises three stages: 1) Global-topological relation graph construction. We analyze global relation patterns in the input knowledge graph to construct multi-relation graphs that encompass six patterns. 2) Relation embedding learning. We utilize attention mechanisms on the topology graph to learn and compute relation embeddings. 3) Entity embedding learning. We combine the original knowledge graph with relation embeddings and update entity embeddings through neighbor aggregation using attention mechanisms. During the inference stage, the reconstructed global topology graph allows us to compute embeddings for new relations and new entities using previously learned attention weights.

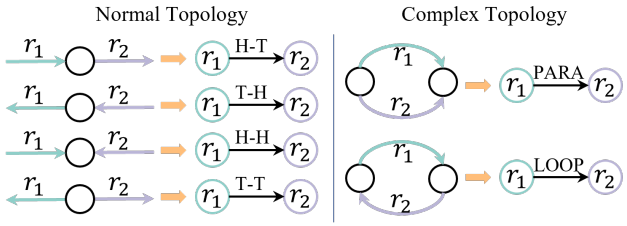

Figure 4: Topological patterns of relation pairs (Chen et al. 2021), we divide these patterns into normal and complex.

Global-Topological Relation Graph Construction

To model correlation patterns between any relation pairs, (Chen et al. 2021) categorize the relational topological patterns into seven types: "head-to-tail", "tail-to-tail", "head-to-head", "tail-to-head", "parallel", "loop", and "not connected". Since "not connected" is not relevant to our link prediction tasks, we exclude this pattern. Following the definitions of (Chen et al. 2021), these six patterns are named "H-T", "T-T", "H-H", "T-H", "PARA", and "LOOP" as shown in Figure 4. For example, $(r_1, \text{H-T}, r_2)$ signifies the correlation between r_1 and r_2, where r_2 has an "H-T" pattern, indicating that r_1 and r_2 are linked head-to-tail. (r_1, PARA, r_2) indicates that the two relations share the same head and tail entities, while (r_1, LOOP, r_2) denotes that these relations like a loop.

Based on this, we begin constructing a multi-relation graph. In the given knowledge graph, we first identify patterns. Each relation is treated as a node v, and the connections (topological patterns) between different pairs of relations are considered as edges e, thereby constructing a directed multi-graph. Due to the varying topological structures in various datasets, the number of relations (i.e., nodes in the multi-graph) differs (as shown in Figure 1). The multi-graph

helps us extract adjacency matrices of consistent shape, facilitating subsequent operations. Specifically, we define a three-dimensional tensor $\mathbb{A} \in \mathbb{R}^{K \times M \times M}$, where $|K| = 6$ represents the types of topology and M represents the number of edges. Each slice corresponds to the adjacency matrix of relation pairs under a specific type. The adjacency tensor $\mathbb{A}_e(v_h, v_t)$ indicates the presence of an edge of type e between the nodes v_h and v_t, which can be easily obtained by traversing the multi-graph (Lee, Chung, and Whang 2023).

By constructing a global relation graph across different topological patterns, we ensure that even though relations at inference time differ, their interactions stay constant and are captured by the global relation graph. This enables the model to generalize to inductive scenarios.

Relation Embedding Learning

By using the method mentioned above, we derive six distinct patterns of topological relation graphs. Each graph is uniquely characterized by its adjacency matrix A, which specifies the adjacency relations for each type of relation. For undefined relations and entities, we utilize Glorot initialization (Glorot and Bengio 2010) to generate their initial feature vectors, while applying GNN message-passing (Wang et al. 2022) to iteratively update the feature embeddings of both relation and entity nodes. The detailed process is as follows:

First, we update the relation embedding vectors by aggregating them based on the multi-relation graph. Each relation node r_i has a feature vector $x_i \in \mathbb{R}^d$ $(i = 1, \cdots, m)$, where d represents the dimension of the feature vector. Let $s_i^{(l)} \in \mathbb{R}^{d'}$ denote the hidden embedding of the relation node r_i, where d' is the dimension of the hidden layer, and the superscript l indicates the l-th layer with $l = 0, \cdots, L-1$, where L is count of layers for relations. Initially, we project the initial feature vector x_i onto a hidden embedding vector through a linear mapping. This process can be described as $s_i^{(0)} = Hx_i$, where $H \in \mathbb{R}^{d' \times d}$ is the projection matrix. Unless specified, all vectors are column vectors. This projection transforms the original feature vectors into a new form of embedding, enabling them to better capture and represent the characteristics of relation nodes in subsequent layers.

Then, we update the embedding of each relation by aggregating the embedding vectors of each relation node and its adjacent relations. Specifically, the forward propagation process can be defined as:

$$s_i^{rel} = \sigma\left(\sum_{r_j \in \mathcal{N}_i} \alpha_{ij}^{(l)} W_{rel}^{(l)} s_j^{(l)}\right), \tag{1}$$

where $\sigma(\cdot)$ is an element-wise activation function, such as LeakyReLU (Maas et al. 2013), $s_j^{(l)}$ represents the embedding vector of the adjacent relation r_j at layer l, $W_{rel}^{(l)} \in \mathbb{R}^{d' \times d'}$ is the weight matrix for that layer, and $\mathcal{N}_i$ denotes the set of neighbors of relation r_i in the relation graph. The attention score $\alpha_{ij}^{(l)}$ can be defined as:

$$\alpha_{ij}^{(l)} = \frac{\exp\left(y^{(l)}\sigma\left(W^{(l)}[s_i^{(l)} || s_j^{(l)}]\right)\right)}{\sum_{r_{j'} \in \mathcal{N}_i} \exp\left(y^{(l)}\sigma\left(W^{(l)}[s_i^{(l)} || s_j^{(l)}]\right)\right)}, \tag{2}$$

where $||$ denotes the concatenation of vectors, $y^{(l)} \in \mathbb{R}^{1 \times d'}$ is a row weight vector used to address the static attention issues suggested by (Velickovic et al. 2018) and (Brody, Alon, and Yahav 2022). $W^{(l)} \in \mathbb{R}^{d' \times 2d'}$ denotes the weight matrix used for processing the concatenated embeddings. $r_{j'}$ is the index of neighboring node j'. The core idea is to assign different attention scores to each relation node to more effectively aggregate information from adjacent relations.

To further stabilize the learning process of the self-attention mechanism, we find it advantageous to extend our approach to a multi-head attention mechanism, similar to (Vaswani et al. 2017). Specifically, we utilize K independent attention heads to perform transformations separately, and then concatenate the feature embeddings from these heads to obtain the final comprehensive output feature embedding.

$$s_i^{rel} = \Big\|_{k=1}^{K} \sigma\left(\sum_{r_j \in \mathcal{N}_i} (\alpha_{ij}^{(l)})^k (W_{rel}^{(l)})^k s_j^{(l)}\right), \tag{3}$$

where $||$ denotes the concatenation operation, $(\alpha_{ij}^{(l)})^k$ is the normalized attention coefficient computed by the k-th attention mechanism. $(W_{rel}^{(l)})^k$ is the weight matrix associated with the linear transformation of the k-th attention head.

Through the above aggregation, we obtain relation embeddings for each specific pattern. As we construct six distinct relation graphs, we concatenate these relation embeddings and apply an adaptive weighted aggregation strategy to form the final comprehensive relation embedding is:

$$s_i^{(l+1)} = \sum_{r=1}^{\mathcal{R}} \beta_r \cdot (s_i^{rel})_r, \tag{4}$$

where β_r is the aggregation weight and $(s_i^{rel})_r$ is the relation embedding obtained in the r-th relation graph. During the update process at each layer, multi-relation topological graphs enable relation embeddings to include multiple types of relations and their interactions, capturing diverse and complex structural features in relation-rich networks.

Entity Embedding Learning

Furthermore, we utilize the generated relation embedding to guide entity-level aggregation. Entity v_i is initialized by Glorot initialization denote as $\widehat{x}_i \in \mathbb{R}^{\widehat{d}}$ $(i = 1, \cdots, n)$, where $\hat{d}$ is the feature dimension. The hidden embedding of v_i is defined as $h_i^{(l)} \in \mathbb{R}^{\widehat{d'}}$, where d' is hidden layer dimension and the superscript l indicates the l-th layer, $l = 0, \cdots, L-1$. Similar to the initial processing of relations, we project the initial feature vector to the hidden space using a linear mapping as $\mathrm{h}_i^{(0)} = \widehat{H}\hat{\mathrm{x}}_i$, where $\widehat{H} \in \mathbb{R}^{\widehat{d'} \times \widehat{d}}$.

A GNN layer updates every node embedding by dynamically aggregating its neighbors' embeddings (Brody,

Alon, and Yahav 2022), we define the neighbors v_j for v_i as $\widehat{\mathcal{N}}_i = \{v_j | (v_j, r_k, v_i) \in \mathcal{F}, v_j \in \mathcal{V}, r_k \in \mathcal{R}\}$. Entity level aggregation includes three embeddings from entity neighborhood, self-loop (Hamilton, Ying, and Leskovec 2017), and the adjacent relations. Specifically, we perform frequency calculation on entity v_j as $fq(v_j) = \sum_{v_j \in \widehat{\mathcal{N}}_i} \sum_{r_k \in \mathcal{R}_{ji}} \delta(v_j = index)$, where $\delta = 0, 1$ is an indicator function and $\mathcal{R}_{ji}$ denotes the set of relations from v_j to v_i. The calculation of the self-loop can be defined as:

$$\overline{s}_i^{(L)} = \sum_{v_j \in \widehat{\mathcal{N}}_i} \sum_{r_k \in \mathcal{R}_{ji}} \frac{s_k^{(L)}}{fq(v_j)}, \quad (5)$$

where $s_k^{(L)}$ represents the relation vector at layer L for a specific relation r_k linking v_j and v_i. This approach combines initial features with the aggregation of neighborhoods, enabling entities to capture more complex and diverse structural features within multi-layered relation graphs.

The entire attention mechanism and node update process are divided into four main steps. Inspired by GATv2 (Brody, Alon, and Yahav 2022), we compute the attention scores for each entity pair and associated relations in the first step as:

$$e(h_j, h_i, r_j)^{(l)} = v_{attn}^{\top} \sigma(W^{(l)} \cdot [{h_j}^{(l)} \| {h_i}^{(l)} \| \overline{s}_i^{(L)}]), \quad (6)$$

where $[{h_j}^{(l)} \| {h_i}^{(l)} \| \overline{s}_i^{(L)}]$ denotes that concatenate the embeddings of entity v_i, v_j and the final relation r_j, $v_{attn}^{\top}$ denotes the dynamic learnable attention weights by GATv2, and $W^{(l)} \in \mathbb{R}^{\hat{d'} \times (2\hat{d'} + d')}$ denotes learnable weight matrix.

Then, we transform these raw attention scores into interpretable weights by normalizing them using the Softmax function which can be defined as:

$$\beta_{ijk}^{(l)} = \mathrm{softmax}(e(h_i, h_j, r_k)^{(l)}), \quad (7)$$

With these normalized attention weights, we perform aggregation, which adopts a weighted sum of all neighboring entities and related relations as:

$$\mathrm{aggr}(h_i)^{(l)} = \sum_{v_j \in \widehat{\mathcal{N}}_i} \sum_{r_k \in \mathcal{R}_{ji}} \beta_{ijk}^{(l)} \cdot W^{(l)} [h_j^{(l)} \| s_k^{(L)}], \quad (8)$$

Finally, we update the target node's embedding using the aggregated information by combining the node's information with that of its neighbors as:

$$h_i^{(l+1)} = \sigma(W^{(l)} [h_i^{(l)} \| s_i^{(L)}] + \mathrm{aggr}(h_i)^{(l)}) \quad (9)$$

Optimization Objective

Negative triplets T_{neg} are generated by altering a head or tail entity of positive triplet T_{tr}, where $T_{neg} = T_{neg}^i \cup T_{neg}^j$. For instance, negative triplets of head entity are $T_{neg}^i = \{(v_i', r_k, v_j) \mid (v_i, r_k, v_j) \in T_{tr}, v_i \in \mathcal{E}, v_i' \neq v_i\}$. We initialize entities and relations using their respective learning embeddings to set up for forward propagation and obtain embedding vectors $s_k := M s_k^{(L)}$ $(k = 1, \cdots, m)$ for relations and $h_i := \widehat{M} h_i^{(\widehat{L})}$ $(i = 1, \cdots, n)$ for entities.

Since we aim to simplify the problem, we convert all link prediction tasks into the task of predicting tail entities. Thus, our KGC link prediction is viewed as a $query(v_i, r_k, ?)$. We insert each candidate entity $v_j \in V_{\text{inf}}$ into the query triple. Then we use a knowledge graph embedding scoring function, denoted by $f(v_i, r_k, v_j)$, returns a scalar value representing the plausibility of a given triplet (v_i, r_k, v_j) (Schlichtkrull et al. 2018). The entity with the highest score is defined as the missing entity (Sun et al. 2019). The model calculates scores for both positive and negative samples, employing a specific scoring function, applied to both sample types. Then, the margin-based ranking loss is defined as:

$$\mathcal{L} = \sum_{(v_i, r_k, v_j) \in T_{tr}} \max(0, \gamma - f(T_{tr}) + f(T_{neg})), \quad (10)$$

where γ is a margin separating the positive and negative.

Following loss computation, gradients are obtained via back-propagation, with gradient clipping applied to maintain stable training dynamics. Parameters are updated at the end of each epoch using the Adam optimizer (Kingma and Ba 2015). In essence, the process leverages an optimization loop for updating model parameters, integrating loss evaluation, and back-propagation via Adam.

Experiments

We first give the experimental datasets and evaluation metrics, and then compare our TARGI with the state-of-the-art methods on inductive, traditional inductive link prediction, and ablation studies. In the inductive link prediction experiments, we examine both semi-inductive and fully inductive scenarios for relations. Finally, we conduct ablation studies to provide a thorough analysis of TARGI.

Experimental Setup

Datasets. We conduct KGC link prediction experiments on two commonly used public datasets: NELL995 (Xiong, Hoang, and Wang 2017) and FB15K-237 (Toutanova and Chen 2015). Due to the differences in experimental scenarios compared to traditional KGC methods, we adopt the approach by (Lee, Chung, and Whang 2023) to design and process NELL995 and FB15K-237 into six distinct datasets, corresponding to traditional inductive learning scenarios (without new relations) and inductive scenarios. These datasets can be categorized into two types: traditional inductive datasets NL-0 and NELL995-v1 as $V_{train} \neq V_{inf}$ and $R_{train} = R_{inf}$. Inductive datasets NL-50, FB-50, NL-100, and FB-100, where $V_{train} \neq V_{inf}$ and $R_{train} \neq R_{inf}$. The dataset suffix -100 indicates that 100% relations are new, -50 indicates that 50% relations are new, and -0 indicates that all relations are known. For the entity part, we follow the assumption made by (Teru, Denis, and Hamilton 2020), where all entities in G_{inf} are considered new entities. Intuitively, TARGI is expected to be better suited for the FB dataset. However, to facilitate broader comparisons, we chose the NELL for traditional inductive scenarios.

Baselines. We compare the performance with 15 models: GraIL (Teru, Denis, and Hamilton 2020), CoMPILE (Mai et al. 2021), RMPI (Geng et al. 2023), CompGCN (Vashishth et al. 2020), NodePiece (Galkin et al. 2022), NeuralLP (Yang, Yang, and Cohen 2017), DRUM (Sadeghian

	FB-100				NL-100				FB-50				NL-50			
	MR ↓	MRR	Hit@10	Hit@1	MR ↓	MRR	Hit@10	Hit@1	MR ↓	MRR	Hit@10	Hit@1	MR ↓	MRR	Hit@10	Hit@1
GraIL	-	-	-	-	915.4	0.133	0.178	0.115	-	-	-	-	818.7	0.155	0.258	0.101
CoMPILE	-	-	-	-	743.8	0.125	0.212	0.070	-	-	-	-	449.6	0.187	0.307	0.135
RMPI	-	-	-	-	142.8	0.222	0.377	0.140	-	-	-	-	470.1	0.188	0.307	0.100
CompGCN	1239.2	0.014	0.022	0.010	887.3	0.007	0.017	0.003	2331.9	0.003	0.006	0.002	1188.7	0.003	0.006	0.000
NodePiece	1111.1	0.007	0.008	0.001	777.2	0.015	0.017	0.004	1311.2	0.019	0.048	0.005	835.7	0.037	0.081	0.012
NeuralLP	977.5	0.025	0.061	0.009	558.9	0.078	0.191	0.030	1451.8	0.081	0.188	0.048	789.4	0.099	0.191	0.065
DRUM	980.0	0.033	0.070	0.010	583.8	0.073	0.133	0.040	833.8	0.066	0.127	0.035	453.1	0.155	0.328	0.072
BLP	890.2	0.018	0.037	0.007	561.1	0.021	0.047	0.009	571.3	0.070	0.157	0.035	428.5	0.037	0.095	0.013
QBLP	812.6	0.015	0.023	0.004	693.3	0.003	0.003	0.000	532.4	0.068	0.134	0.033	341.6	0.051	0.101	0.029
NBFNet	431.4	0.076	0.166	0.030	221.2	0.090	0.203	0.035	735.8	0.133	0.255	0.068	318.3	0.213	0.335	0.151
RED-GNN	361.3	0.113	0.258	0.050	196.8	0.210	0.378	0.111	1153.8	0.128	0.250	0.073	631.3	0.171	0.287	0.116
RAILD	698.3	0.038	0.061	0.022	571.5	0.023	0.047	0.011	-	-	-	-	-	-	-	-
SE-GNN	936.5	0.004	0.011	0.005	395.8	0.020	0.035	0.001	461.1	0.008	0.009	0.001	1717.7	0.002	0.003	0.000
GreenKGC	4953.3	0.001	0.002	0.000	1524.1	0.004	0.006	0.001	1994.6	0.138	0.254	0.083	1215.9	0.197	0.343	0.126
InGram	172.7	0.228	0.373	0.158	**96.8**	0.295	0.457	0.205	539.3	0.111	0.201	0.060	113.8	0.273	0.422	0.185
TARGI	**136.6**	**0.253**	**0.429**	**0.172**	161.7	**0.314**	**0.489**	**0.230**	**332.2**	**0.159**	**0.307**	**0.088**	**98.5**	**0.314**	**0.533**	**0.207**

Table 1: Inductive link prediction. "-" denotes result not available: Due to scalability issues with enclosing subgraph sampling, we cannot provide the results of GraIL, CoMPILE, and RMPI on FB consistent with the findings in (Lee, Chung, and Whang 2023). The prohibition against new and unknown relations prevents us from offering RAILD results on FB-50 and NL-50.

et al. 2019), BLP (Daza, Cochez, and Groth 2021), QBLP (Ali et al. 2022), NBFNet (Zhu et al. 2021), RED-GNN (Zhang and Yao 2022), RAILD (Gesese, Sack, and Alam 2022), SE-GNN (Li et al. 2022), GreenKGC (Wang et al. 2023) and InGram (Lee, Chung, and Whang 2023).

Implementation Details. To ensure a fair comparison, we set the dimensions of our method and all baseline methods at $d = 32$ and $\hat{d} = 32$. Each experiment is run five times, and the average results are reported for robust comparison. We use the common Glorot initialization (Glorot and Bengio 2010) to initialize the initial features of relations and entities and we choose Adam optimizer (Kingma and Ba 2015) and the default initialization in pytorch.

Inductive Link Prediction

We conduct fully inductive KGC link prediction on FB-100 and NL-100 and semi-inductive on FB-50 and NL-50. As shown in Table 2, our method shows overall superior performance compared to baselines, especially on FB dataset. This suggests that our aggregation function effectively enhances relation-embedding learning by integrating relation topology patterns. Results support that TARGI can adapt to varying proportions of new relations and is not confined to strictly fully inductive scenarios.

GraIL, CoMPILE, and RMPI employ subgraph sampling techniques, we extend these methods to enable them to learn from all entities in V_{inf} but are still limited to the NELL dataset. BLP, QBLP, RAILD, and GreenKGC depend on pre-trained embeddings. For example, GreenKGC requires pretraining and feature pruning with RotatE to fix low-dimensional triple features. RAILD results are not available for semi-inductive scenarios because they cannot simultaneously process known and unknown relations. The results of GreenKGC with previous reports are unlikely due to the sharp decline in performance caused by pre-training inability to handle new relations.

Traditional Inductive Link Prediction

As shown in Table 2, we can observe that our proposed TARGI performs well among most traditional inductive link prediction methods. Our results frequently rank first and often rank second, which is primarily because we focus on enhancing the model's ability to generalize to new relations rather than simply memorizing the specific topology of known relations. In contrast, methods like NBFNet identify and directly apply predefined rules or patterns between known relations for reasoning. Note that the results of reproduced comparison algorithms may differ from previously reported values. To ensure a fair comparison, we uniformly set the embedding dimension to 32, which may impact some methods like SE-GNN. Additionally, for the validation and testing, we use the "ind-test" provided by the NELL dataset, which might differ from other reported methods. Methods like GreenKGC, which involve pre-training on the data, might potentially enhance model performance. We may explore this approach in future work for improvements.

Ablation Study

To evaluate the importance of each module in our method, we conduct ablation studies on datasets under fully-inductive scenarios. We specifically substitute the relation-weighted aggregator with mean and sum aggregators and use a traditional graph attention network (GAT) for relation embedding learning. Additionally, we evaluate the model by applying each of the six topological relation graphs individually and conducting specific experiments on normal topology (excluding LOOP and PARA). As shown in Table 3, the results reveal that the ablation models consistently fall short of the performance achieved by the complete model. Moreover, the impact of these topological structures varies across different datasets. The study affirms our experimental findings: neglecting the topological structure between re-

Model	NELL995-v1				NL-0			
	MR↓	MRR	Hit@10	Hit@1	MR↓	MRR	Hit@10	Hit@1
GraIL	18.9	0.489	0.588	0.398	555.8	0.191	0.335	0.118
CoMPILE	19.9	0.477	0.558	0.393	566.2	0.222	0.358	0.144
RMPI	52.2	0.468	0.557	0.422	391.1	0.218	0.333	0.155
CompGCN	13.9	0.289	0.740	0.006	938.8	0.008	0.009	0.001
NodePiece	10.2	0.672	**0.858**	0.543	355.8	0.098	0.220	0.039
NeuralLP	35.4	0.557	0.787	0.411	577.8	0.178	0.343	0.098
DRUM	33.0	0.525	0.732	0.400	535.1	0.198	0.333	0.128
BLP	40.1	0.164	0.474	0.050	435.4	0.041	0.104	0.013
QBLP	17.5	0.332	0.575	0.245	332.7	0.071	0.154	0.015
NBFNet	**7.2**	0.611	0.834	0.503	153.2	**0.258**	**0.419**	**0.170**
RED-GNN	15.6	0.552	0.698	0.488	318.5	0.218	0.365	0.144
RAILD	115.3	0.063	0.221	0.001	433.7	0.048	0.113	0.018
SE-GNN	76.1	0.027	0.035	0.005	411.3	0.013	0.016	0.006
GreenKGC	56.5	0.179	0.345	0.085	**28.7**	0.176	0.418	0.060
InGram	9.8	0.688	0.725	0.530	148.1	0.238	0.401	0.154
TARGI	10.8	**0.692**	0.855	**0.580**	128.5	0.241	0.413	0.159

Table 2: Traditional inductive link prediction.

lations is inadequate. Moreover, replacing our relation layer with GAT significantly affects results, causing around a 57% drop in NL-100 and a 39% decrease in MRR on the FB-100 dataset, indicating that our relation embedding layer has a significant advantage in modeling the relation topology.

	FB-100			NL-100		
	MRR	Hit@10	Hit@1	MRR	Hit@10	Hit@1
w/o mean agg.	0.244	0.384	0.170	0.279	0.467	0.178
w/o sum agg.	0.182	0.280	0.124	0.211	0.334	0.146
w/o relation layer	0.155	0.225	0.113	0.135	0.299	0.071
only T-H	0.188	0.317	0.122	0.223	0.405	0.141
only H-H	0.216	0.365	0.140	0.263	0.443	0.182
only H-T	0.177	0.199	0.071	0.259	0.404	0.173
only T-T	0.219	0.352	0.149	0.258	0.440	0.168
only LOOP	0.236	0.385	0.160	0.253	0.428	0.173
only PARA	0.204	0.328	0.136	0.165	0.286	0.100
only normal	0.233	0.387	0.157	0.233	0.387	0.157
TARGI	**0.253**	**0.429**	**0.172**	**0.314**	**0.489**	**0.230**

Table 3: Ablation study of TARGI.

Related Work

Traditional Knowledge Graph Completion. Traditional KGC can be categorized into three main approaches. First, embedding-based methods transform entities and relations into low-dimensional vector spaces, enabling efficient computation and reasoning (Nathani et al. 2019) include TransE (Bordes et al. 2013) and RotatE (Sun et al. 2019). Second, path-based methods focus on reasoning over paths or contexts within the graph for link prediction, with PRA (Sun et al. 2019) as a notable example. Finally, GNN-based methods integrate the structure information of KGs into neural networks including RED-GNN (Zhang and Yao 2022) and SE-GNN (Li et al. 2022), effectively capturing both local and global structural features for improved performance.

Knowledge Embedding Learning. Existing inductive methods based on embedding learning often lack a specific focus on relations. For instance, in BLP (Chen et al. 2021), relations are represented through random initialization, while QBLP (Daza, Cochez, and Groth 2021) adopts an inductive approach using hyper-relations within a hyper-GNN. However, since it does not generate specific embeddings for relations, it still relies on BERT-based pre-trained vectors. Similarly, GreenKGC (Wang et al. 2023) leverages KG embedding methods like RotatE for feature pruning and generating pre-trained vectors, which are derived from the names and textual descriptions of entities or relations. Nevertheless, due to the specific nature of KG datasets, these textual descriptions are often brief and fail to provide sufficient details about relations, limiting their effectiveness.

Inductive Knowledge Graph Completion. Existing methods for inductive Knowledge Graph Completion can be broadly categorized based on their focus. Some methods extract entity-independent relation patterns using logical inference (Teru, Denis, and Hamilton 2020), while others leverage GNNs to capture neighborhood information, enabling feature extraction for unseen entities and relations (Schlichtkrull et al. 2018). Most current approaches assume that new entities, not included during training, will emerge during inference, and thus perform inductive inference on entities. However, methods like R-GCN, CompGCN, NodePiece (Galkin et al. 2022), and GraIL (Teru, Denis, and Hamilton 2020) assume that all relations are known during training, applying transductive learning to relations rather than inductive learning. Recently, some approaches have shifted focus to fully inductive scenarios for both entities and relations. For instance, RMPI (Geng et al. 2023) utilizes enclosing subgraph sampling to construct relation message-passing networks, while InGram (Lee, Chung, and Whang 2023) leverages the degree of relations to construct relation graphs, enabling generalization to unknown domains. ULTRA (Galkin et al. 2024) introduces a unique pre-training and fine-tuning paradigm, benefiting from large-scale training data but requiring significant computational resources.

Conclusion

In this paper, we propose a GNN-based method called TARGI for KGC, applicable to fully inductive, semi-inductive, and traditional inductive scenarios for link prediction tasks. We first construct global relation graphs for each topological structure from a holistic view, effectively leveraging the invariance of relation structures. We then adaptively construct a global-topological relation graph according to the ratio of new to known relations in different inductive scenarios. Finally, we aggregate relation and entity embeddings, updating them during inference through the global relation topological graph to robustly perform link prediction tasks in knowledge graphs. Our method addresses the over-reliance on "degree" in existing fully inductive algorithms and mitigates the high complexity and limitations of enclosing subgraph sampling. Experiments on six real-world datasets demonstrate that our approach outperforms existing methods in inductive KGC tasks and especially shows competitive performance in fully inductive scenarios.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (No. 2023YFC3304503), the National Natural Science Foundation of China (No. 92370111, No. 62422210, No. 62272340, and No. 62276187, No. 62402337), the Postdoctoral Fellowship Program of CPSF under Grant No. GZC20241207, the China Postdoctoral Science Foundation under Grant No. 2024M752367, and Hebei Natural Science Foundation No. F2024202047.

References

Ali, M.; Berrendorf, M.; Galkin, M.; Thost, V.; Ma, T.; Tresp, V.; and Lehmann, J. 2022. Improving Inductive Link Prediction Using Hyper-Relational Facts. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 5259–5263.

Arnaout, H.; Razniewski, S.; Weikum, G.; and Pan, J. Z. 2021. Negative Statements Considered Useful. *J. Web Semant.*, 71: 100661.

Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems*, 2787–2795.

Brody, S.; Alon, U.; and Yahav, E. 2022. How Attentive are Graph Attention Networks? In *the Tenth International Conference on Learning Representations*.

Chen, J.; He, H.; Wu, F.; and Wang, J. 2021. Topology-Aware Correlations Between Relations for Inductive Link Prediction in Knowledge Graphs. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, 6271–6278.

Daza, D.; Cochez, M.; and Groth, P. 2021. Inductive Entity Representations from Text via Link Prediction. In *Proceedings of the 21th ACM Web Conference*, 798–808.

Galkin, M.; Denis, E. G.; Wu, J.; and Hamilton, W. L. 2022. NodePiece: Compositional and Parameter-Efficient Representations of Large Knowledge Graphs. In *the Tenth International Conference on Learning Representations*.

Galkin, M.; Yuan, X.; Mostafa, H.; Tang, J.; and Zhu, Z. 2024. Towards Foundation Models for Knowledge Graph Reasoning. In *The Twelfth International Conference on Learning Representations*.

Geng, Y.; Chen, J.; Pan, J. Z.; Chen, M.; Jiang, S.; Zhang, W.; and Chen, H. 2023. Relational Message Passing for Fully Inductive Knowledge Graph Completion. In *39th IEEE International Conference on Data Engineering*, 1221–1233.

Gesese, G. A.; Sack, H.; and Alam, M. 2022. RAILD: Towards Leveraging Relation Features for Inductive Link Prediction In Knowledge Graphs. In *Proceedings of the 11th International Joint Conference on Knowledge Graphs*, 82–90.

Glorot, X.; and Bengio, Y. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9, 249–256.

Hamilton, W. L.; Ying, Z.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. In *Advances in Neural Information Processing Systems*, 1024–1034.

Huang, X.; Zhang, J.; Li, D.; and Li, P. 2019. Knowledge Graph Embedding Based Question Answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 105–113.

Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; and Yu, P. S. 2022. A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE Trans. Neural Networks Learn. Syst.*, 33(2): 494–514.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations*.

Lee, J.; Chung, C.; and Whang, J. J. 2023. InGram: Inductive Knowledge Graph Embedding via Relation Graphs. In *International Conference on Machine Learning*, volume 202, 18796–18809.

Li, R.; Cao, Y.; Zhu, Q.; Bi, G.; Fang, F.; Liu, Y.; and Li, Q. 2022. How Does Knowledge Graph Embedding Extrapolate to Unseen Data: A Semantic Evidence View. In *Thirty-Sixth AAAI Conference on Artificial Intelligence*, 5781–5791.

Liang, K.; Liu, Y.; Zhou, S.; Tu, W.; Wen, Y.; Yang, X.; Dong, X.; and Liu, X. 2023. Knowledge graph contrastive learning based on relation-symmetrical structure. *IEEE Transactions on Knowledge and Data Engineering*, 36(1): 226–238.

Liang, K.; Meng, L.; Zhou, S.; Tu, W.; Wang, S.; Liu, Y.; Liu, M.; Zhao, L.; Dong, X.; and Liu, X. 2024. MINES: Message Intercommunication for Inductive Relation Reasoning over Neighbor-Enhanced Subgraphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 10645–10653.

Maas, A. L.; Hannun, A. Y.; Ng, A. Y.; et al. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of International Conference on Machine Learning*, 3.

Mai, S.; Zheng, S.; Yang, Y.; and Hu, H. 2021. Communicative Message Passing for Inductive Relation Reasoning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence*, 4294–4302.

Nathani, D.; Chauhan, J.; Sharma, C.; and Kaul, M. 2019. Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, 4710–4723.

Paulheim, H. 2017. Knowledge Graph Refinement: A Survey of Approaches and Evaluation Methods. *Semantic Web*, 8(3): 489–508.

Sadeghian, A.; Armandpour, M.; Ding, P.; and Wang, D. Z. 2019. DRUM: End-To-End Differentiable Rule Mining On Knowledge Graphs. In *Advances in Neural Information Processing Systems*, 15321–15331.

Schlichtkrull, M. S.; Kipf, T. N.; Bloem, P.; van den Berg, R.; Titov, I.; and Welling, M. 2018. Modeling Relational Data with Graph Convolutional Networks. In *the Semantic*

Web - 15th International Conference, volume 10843, 593–607.

Sun, Z.; Deng, Z.; Nie, J.; and Tang, J. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *7th International Conference on Learning Representations*.

Teru, K. K.; Denis, E. G.; and Hamilton, W. L. 2020. Inductive Relation Prediction by Subgraph Reasoning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, 9448–9457.

Toutanova, K.; and Chen, D. 2015. Observed Versus Latent Features for Knowledge Base and Text Inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, 57–66.

Vashishth, S.; Sanyal, S.; Nitin, V.; and Talukdar, P. P. 2020. Composition-based Multi-Relational Graph Convolutional Networks. In *8th International Conference on Learning Representations*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *6th International Conference on Learning Representations*.

Wang, X.; Wang, D.; Xu, C.; He, X.; Cao, Y.; and Chua, T. 2019. Explainable Reasoning over Knowledge Graphs for Recommendation. In *the Thirty-Third AAAI Conference on Artificial Intelligence*, 5329–5336.

Wang, Y.; Ge, X.; Wang, B.; and Kuo, C. J. 2023. GreenKGC: A Lightweight Knowledge Graph Completion Method. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 10596–10613.

Wang, Z.; Mu, C.; Hu, S.; Chu, C.; and Li, X. 2022. Modelling the Dynamics of Regret Minimization in Large Agent Populations: a Master Equation Approach. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 534–540.

Xiong, W.; Hoang, T.; and Wang, W. Y. 2017. DeepPath: A Reinforcement Learning Method for Knowledge Graph Reasoning. *CoRR*, abs/1707.06690.

Yang, F.; Yang, Z.; and Cohen, W. W. 2017. Differentiable Learning of Logical Rules for Knowledge Base Reasoning. In *Advances in Neural Information Processing Systems*, 2319–2328.

Zhang, Y.; and Yao, Q. 2022. Knowledge Graph Reasoning with Relational Digraph. In *Proceedings of the 22th ACM Web Conference*, 912–924.

Zhu, Z.; Zhang, Z.; Xhonneux, L. A. C.; and Tang, J. 2021. Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction. In *Advances in Neural Information Processing Systems*, 29476–29490.