UniHR: Hierarchical Representation Learning for Unified Knowledge Graph Link Prediction

Anonymous ACL submission

Abstract

Beyond-triple fact representations including hyper-relational facts with auxiliary key-value pairs, temporal facts with additional timestamps, and nested facts implying relationships between facts, are gaining significant attention. However, existing link prediction models are usually designed for one specific type of facts, making it difficult to generalize to other fact representations. To overcome this limitation, we propose a **Uni**fied **H**ierarchical **R**epresentation learning framework (**UniHR**) for unified knowledge graph link prediction. It consists of a unified Hierarchical Data Representation (HiDR) module and a unified Hierarchical Structure Learning (HiSL) module as graph encoder. The HiDR module unifies hyper-relational KGs, temporal KGs, and nested factual KGs into triple-based representations. Then HiSL incorporates intra-fact and inter-fact message passing, focusing on enhancing the semantic information within individual facts and enriching the structural information between facts. Experimental results across 7 datasets from 3 types of KGs demonstrate that our UniHR outperforms baselines designed for one specific kind of KG, indicating strong generalization capability of HiDR form and the effectiveness of HiSL module. Code and data are available at `https://anonymous.4open.science/r/UniHR-BDCB/`.

1 Introduction

Large-scale knowledge graphs (KGs) such as WordNet (Miller, 1995), Freebase (Bollacker et al., 2008), and Wikidata (Vrandečić and Krötzsch, 2014) have been widely applied in many areas like question answering (Kaiser et al., 2021), recommendation systems (Guo et al., 2020), and natural language processing (Annervaz et al., 2018). However, the presence of missing facts within these KGs inevitably limit their applications. Therefore, the link prediction task has been introduced to predict missing elements within factual data. Current link prediction methods mainly focus on facts in the form of triple $(head\,entity, relation, tail\,entity)$.

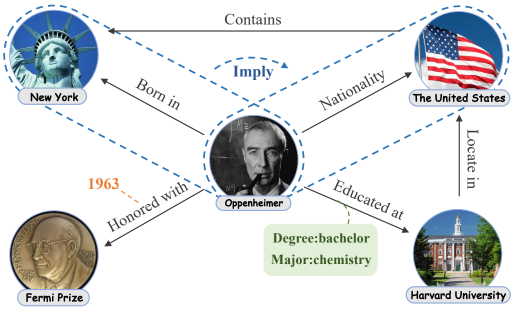

Figure 1: A special KG consists of triple-based fact, hyper-relational fact, nested fact and temporal fact.

Despite the simplicity and unity of triple-based representation, it is difficult to adequately express complex facts, such as "*Oppenheimer is educated at Harvard University for a bachelor degree in chemistry*" shown in Figure 1. Therefore, existing researches (Wang et al., 2021; Xiong et al., 2024; Xu et al., 2019) contribute to focusing on semantically richer facts. Figure 1 illustrates three specific types of facts: hyper-relational fact ((*Oppenheimer, educated at, Harvard University*), *degree: bachelor, major: chemistry*), temporal fact (*Oppenheimer, honored with, Fermi Prize, 1963*), nested fact ((*Oppenheimer, born in, New York*), *imply*, (*Oppenheimer, nationality, The United States*)). These forms of facts allow for expression of complex semantics and revelation of relationships between facts, extending beyond the triple-based representation. Thus in recent years, Hyper-relational KGs (HKG) (Chung et al., 2023), Temporal KGs (TKG) (Xu et al., 2023a), and Nested factual KGs (NKG) (Xiong et al., 2024) attract wide research interests.

Recent studies have demonstrated the effectiveness of various embedding strategies for these beyond-triple representations (Xiong et al., 2023).

However, these methods are usually designed for specific representation forms, e.g., StarE (Galkin et al., 2020) customizes graph neural network to implement message passing on hyper-relational facts, For nested factual KGs, BiVE (Chung and Whang, 2023) connects two levels of facts throgh a simple linear layer. In addition, GeomE+ (Xu et al., 2023a) et al. temporal KG embedding methods contain time-aware scoring functions to adapt timestamps. Although these methods perform well on specific type of facts, it is evident that such customized methods are difficult to generalize to other types of KGs. *Therefore, establishing a unified representation learning method for multiple types of KGs is worth to investigate.*

To overcome the challenges mentioned above, we propose a **Uni**fied **H**ierarchical **R**epresentation learning method **(UniHR)**, which includes a Hierarchical Data Representation (HiDR) module and a Hierarchical Structure Learning (HiSL) module as the graph encoder. HiDR module standardizes hyper-relational facts, nested factual facts, and temporal facts into the form of triples without loss of information. Furthermore, HiSL module captures local semantic information during intra-fact message passing and then utilizes inter-fact message passing to enrich the global view of nodes to obtain better node embeddings based on HiDR form. Finally, the updated embeddings are fed into decoders for link prediction. Experimental results demonstrate that our UniHR achieves state-of-the-art performance on HKG and NKG datasets, and competitive performance on TKG datasets, revealing strong generalization capability of HiDR form and effectiveness of HiSL module. Our contributions can be summarized as follows.

1. We emphasize the value of investigating unified KG representation method, including unified symbolic representation and unfied representation learning method for different KGs.

2. To our knowledge, we propose the first unified KG representation learning framework UniHR, across different types of KGs, including a hierarchical data representation module and a hierarchical structure learning module.

3. We conduct link prediction experiments on 7 datasets across 3 types of KGs. Compared to methods designed for one kind of KG, UniHR achieves the best or competitive results, verifying strong generalization capability.

2 Preliminaries

In this section, we introduce the definition of four types of existing knowledge graphs (KGs): triple-based KG, hyper-relational KG, nested factual KG and temporal KG, along with link prediction tasks on these types of KGs.

Triple-based Knowledge Graph. A common triple-based KG $\mathcal{G}_{KG} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}\}$ represents facts as triples, denoted as $\mathcal{F} = \{(h, r, t) \,|h, t \in \mathcal{V}, r \in \mathcal{R}\}$, where $\mathcal{V}$ is the set of entities and $\mathcal{R}$ is the set of relations.

Link Prediction on Triple-based KG. The link prediction on triple-based KGs involves answering a query $(h, r, ?)$ or $(?, r, t)$, where the missing element '?' is an entity in $\mathcal{V}$.

Hyper-relational Knowledge Graph. A hyper-relational KG (HKG) $\mathcal{G}_{HKG} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}\}$ consists of hyper-relational facts, abbreviated as H-Facts, denoted as $\mathcal{F} = \{((h, r, t), \{(k_i : v_i)\}_{i=1}^{m}) \,|\, h, t, v_i \in \mathcal{V}, r, k_i \in \mathcal{R}\}$. Typically, we refer to (h, r, t) as the main triple in the H-Fact and $\{(k_i : v_i)\}_{i=1}^{m}$ as m auxiliary key-value pairs.

Link Prediction on Hyper-relational KG. Similar to link prediction on triple-based KGs, the link prediction on HKGs aims to predict entities in the main triple or the key-value pairs. Symbolically, the aim is to predict the missing element, denoted as '?' for queries $((h, r, t), (k_1 : v_1), \ldots (k_i : ?))$, $((?, r, t), \{(k_i : v_i)\}_{i=1}^{m})$ or $((h, r, ?), \{(k_i : v_i)\}_{i=1}^{m})$.

Nested Factual Knowledge Graph. A nested factual KG (NKG) can be represented as $\mathcal{G}_{NKG} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}, \hat{\mathcal{R}}, \hat{\mathcal{F}}\}$, which is composed of two levels of facts, called atomic facts and nested facts. $\mathcal{F} = \{(h, r, t) \,|h, t \in \mathcal{V}, r \in \mathcal{R}\}$ is the set of atomic facts, where $\mathcal{V}$ is a set of atomic entities and $\mathcal{R}$ is a set of atomic relations. $\hat{\mathcal{F}} = \{(\mathcal{F}_i, \hat{r}, \mathcal{F}_j) \,|\, \mathcal{F}_i, \mathcal{F}_j \in \mathcal{F}, \hat{r} \in \hat{\mathcal{R}}\}$ is the set of nested facts, where $\hat{\mathcal{R}}$ is the set of nested relations.

Link Prediction on Nested Factual KG. The link prediction on the NKGs is performed on the atomic facts or nested facts. We refer to the link prediction on atomic facts as *Base Link Prediction*, and the link prediction on nested facts as *Triple Prediction*. For base link prediction, given a query $(h, r, ?)$ or $(?, r, t)$, the aim is to predict the missing atomic entity '?' from $\mathcal{V}$. For triple prediction, given a query $(?, \hat{r}, \mathcal{F}_j)$ or $(\mathcal{F}_i, \hat{r}, ?)$, the aim is to predict the missing atomic fact '?' from $\mathcal{F}$.

Temporal Knowledge Graph. A temporal KG (TKG) $\mathcal{G}_{TKG} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}, \mathcal{T}\}$ is composed of quadruple-based facts, which can be represented as $\mathcal{F} = \{(h, r, t, [\tau_b, \tau_e])|h, t \in \mathcal{V}, r \in \mathcal{R}, \tau_b, \tau_e \in \mathcal{T}\}$, where τ_b is the begin time, τ_e is the end time, $\mathcal{V}$ is the set of entities, $\mathcal{R}$ is the set of relations and $\mathcal{T}$ is the set of timestamps.

Link Prediction on Temporal KG. The link prediction on TKGs aims to predict missing entities '?' in $\mathcal{V}$ for two types of queries $(?, r, t, [\tau_b, \tau_e])$ or $(h, r, ?, [\tau_b, \tau_e])$.

3 Related Works

Link Prediction on Hyper-relational Knowledge Graph. Earlier HKG representation learning methods e.g., m-TransH (Wen et al., 2016), RAE (Zhang et al., 2018) have generalized the triple-based approach to HKG and loosely represent the combinations of key-value pairs. Galkin et al. customize StarE (Galkin et al., 2020) based on CompGCN (Vashishth et al., 2019) for hyper-relational facts to capture the interaction information of key-value pairs with the main triple in the message passing stage, and achieves impressive results, demonstrating that the structural information of the graph in HKGs is also important. GRAN (Guan et al., 2021) introduces edge-aware bias into the vanilla transformer attention (Vaswani et al., 2017), while HyNT (Chung et al., 2023) designs a qualifier encoder for HKG. They both focus on intra-fact dependencies but ignore the global structural information. Due to the existence of its particular key-value pairs on H-Facts, there are many limitations in capturing global structural information through the widely available triple-based GNNs.

Link Prediction on Nested Factual Knowledge Graph. Chung et al. (Chung and Whang, 2023) first introduced the concept of nested facts. They also propose BiVE which projects atomic facts to fact nodes in the encoding phase via a simple linear layer and scores both atomic facts and nested facts using the quaternion-based KGE scoring functions like QuatE (Zhang et al., 2019) or BiQUE (Guo and Kok, 2021). Based on BiVE, NestE (Xiong et al., 2024) represents the fact nodes as a 1×3 embedding matrix and the nested relations as a 3×3 matrix to avoid information loss, embedding them into hyperplanes with different dimensions. These methods only capture the semantic information between atomic facts and nested facts while ignoring global structural information. Meanwhile, due to the complexity of this representation, common triple-based GNNs have difficulty in message passing between atomic fact and nested fact.

Link Prediction on Temporal Knowledge Graph. Recent studies in temporal knowledge graph representation learning have focused on enhancing performance by designing special time-aware scoring functions. Models such as TTransE (Leblay and Chekol, 2018), HyTE (Dasgupta et al., 2018), TeRo (Xu et al., 2020), and TGeomE+ (Xu et al., 2023a) incorporate temporal-aware module into the KGE score function in various ways. However, existing models for link prediction seldom directly utilize GNNs to perceive time information for enhancing entity and relation embeddings. Therefore, we believe it is a direction worth exploring.

4 Methodology

In this section, we introduce our method, a **Uni**fied **H**ierarchical **R**epresentation learning framework **(UniHR)**, which includes a Hierarchical Data Representation (HiDR) module and a Hierarchical Structure Learning (HiSL) module. Our workflow can be divided into the following three steps: **1)** Given a KG $\mathcal{G}$ of any type, we represent it into $\mathcal{G}^{HiDR}$ under the HiDR form. **2)** The $\mathcal{G}^{HiDR}$ will be encoded by HiSL module with the enhancement of semantic information within individual facts and structural information between facts on the whole graph. **3)** In the phase of decoding, the updated embeddings of nodes and edges are fed into transformer decoders to obtain the plausibility score of facts.

4.1 Hierarchical Data Representation

To overcome the differences in the representation of multiple types KGs, we introduce a Hierarchical Data Representation module, abbreviated as **HiDR**. Different from labelled RDF representation (Ali et al., 2022), we constrain "triple" to be considered as the basic units of HiDR form, then HiDR could continuous benefit from the model developments of triple-based KGs, which is the most active area about link prediction over KGs.

As shown in Fig. 2, in order to ensure comprehensive representation of facts, we introduce three hierarchical types of nodes and three connected relations in HiDR. Inspired by the nested fact form (Xiong et al., 2024), we denote original entities within three types of KGs as *atomic nodes* and complement *fact nodes* for HKGs and

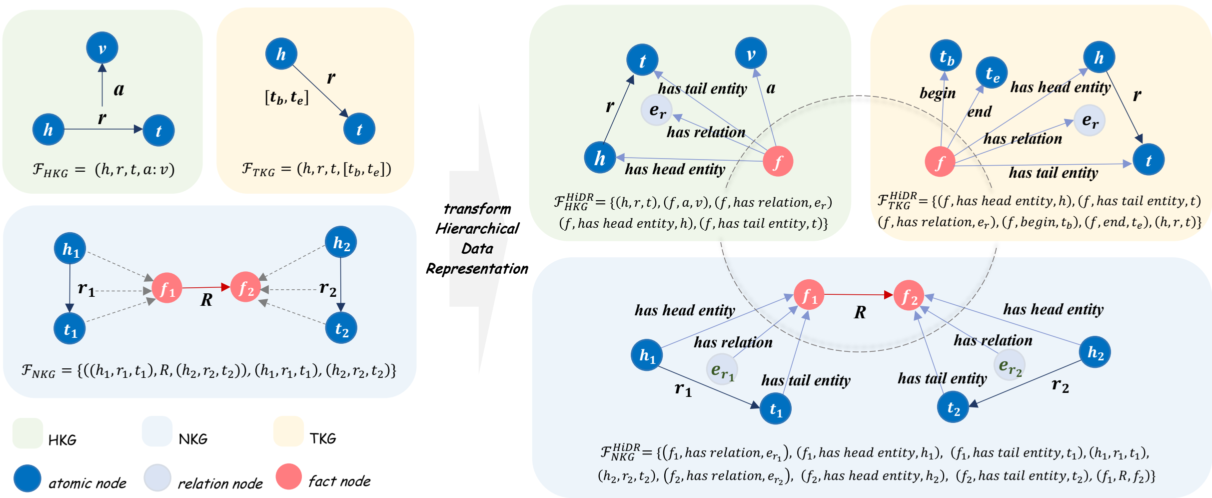

Figure 2: Diverse facts are translated into the HiDR form. Blue, light blue, and red circles denote *atomic nodes*, *relation nodes*, and *fact nodes*. Black, light blue, and red arrows denote *atomic relations*, *connected relations*, and *nested relations*. The triples they connect correspond to *atomic facts*, *connected facts*, and *nested facts*.

TKGs lacking a designated fact node. To facilitate the interaction between fact nodes and relations explicitly, we incorporate *relation nodes* into the graph, represented as e_r for each r. These relation nodes are derived from transforming the relation edges in the original KG. It is important to facilitate direct access of fact nodes to the relevant atomic nodes during message passing process. To achieve this, we introduce three *connected relations*: `has relation`, `has head entity` and `has tail entity`, which establish directly connections between atomic nodes and fact nodes. Ultimately, we denote the (main) triple (h, r, t) in original fact as three *connected facts*: $(f, has\,relation, e_r)$, $(f, has\,head\,entity, h)$, $(f, has\,tail\,entity, t)$, and a *atomic fact* (h, r, t), where f is *fact node*. Formally, the definition of HiDR form is as follows:

Definition 1. *Hierarchical Data Representation: A KG represented as the HiDR form is denoted as $\mathcal{G}^{HiDR} = \{\mathcal{V}^{HiDR}, \mathcal{R}^{HiDR}, \mathcal{F}^{HiDR}\}$, where $\mathcal{V}^{HiDR} = \mathcal{V}_a \cup \mathcal{V}_r \cup \mathcal{V}_f$ is a joint set of atomic node set ($\mathcal{V}_a$), relation node set ($\mathcal{V}_r$), fact node set ($\mathcal{V}_f$). $\mathcal{R}^{HiDR} = \mathcal{R}_a \cup \mathcal{R}_n \cup \mathcal{R}_c$ is a joint set of atomic relation set ($\mathcal{R}_a$), nested relation set ($\mathcal{R}_n$), connected relation set $\mathcal{R}_c = \{has\,relation, has\,head\,entity, has\,tail\,entity\}$. The fact set $\mathcal{F}^{HiDR} = \mathcal{F}_a \cup \mathcal{F}_c \cup \mathcal{F}_n$ is jointly composed of three types of triple-based facts: atomic facts ($\mathcal{F}_a$), connected facts ($\mathcal{F}_c$) and nested facts ($\mathcal{F}_n$), where $\mathcal{F}_a = \{(v_1, r, v_2) | \, v_1, v_2 \in \mathcal{V}_a, r \in \mathcal{R}_a\}$, $\mathcal{F}_c = \{(v_1, r, v_2) | \, v_1 \in \mathcal{V}_f, r \in \mathcal{R}_c, v_2 \in \mathcal{V}_a\}$, $\mathcal{F}_n = \{(v_1, r, v_2) | \, v_1, v_2 \in \mathcal{V}_f, r \in \mathcal{R}_n\}$.*

Next, we introduce how to transform different types of KGs into HiDR form.

For hyper-relational knowledge graphs, we regard key-value pairs as complementary information for facts. Thus, we translate H-Facts $\mathcal{F}_{HKG} = \{((h, r, t), \{(k_i\!: v_i)\}_{i=1}^{m}\}$ into the HiDR form that $\mathcal{G}_{HKG}^{HiDR} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}_{HKG}^{HiDR}\}$ following the definition, where $\mathcal{F}_c$=$\{(f, has\,relation, e_r), (f, has\,head\,entity, h), (f, has\,tail\,entity, t), (f, k_1, v_1), \ldots, (f, k_m, v_m)\}$, $\mathcal{F}_a = \{(h, r, t) \,|\, ((h, r, t), \{(k_i\!: v_i)\}_{i=1}^{m}) \in \mathcal{F}_{HKG})\}$ and $\mathcal{F}_n$=$\varnothing$.

For nested factual knowledge graphs, HiDR can naturally represent hierarchical facts, so we translate the atomic facts $\mathcal{F}_{NKG} = \{(h_i, r_i, t_i)\}$ and the nested facts $\hat{\mathcal{F}}_{NKG} = \{((h_1, r_1, t_1), R, (h_2, r_2, t_2)) | (h_i, r_i, t_i) \in \mathcal{F}_{NKG}\}$ into the form of HiDR that $\mathcal{G}_{NKG}^{HiDR} = \{\mathcal{V}, \mathcal{R}, \mathcal{F}_{NKG}^{HiDR}\}$ following the definition, where $\mathcal{F}_a = \{(h_i, r_i, t_i) \,|\, (h_i, r_i, t_i) \in \mathcal{F}_{NKG}\}$, $\mathcal{F}_c$=$\{(f_i, has\,head\,entity, h_i), (f_i, has\,tail\,entity, t_i), (f_i, has\,relation, e_{r_i}) | f_i = (h_i, r_i, t_i) \in \mathcal{F}_{NKG}\}$ and $\mathcal{F}_n$=$\{(f_1, R, f_2) | f_i \in \mathcal{F}_{NKG}\}$.

For temporal knowledge graphs, we regard the TKG as a special HKG, and convert timestamps to auxiliary key-value pairs in HKGs by adding two special *atomic relations*: `begin` and `end`, regarding timestamps as special numerical atomic nodes. Thus, we firstly translate the temporal facts in TKGs $\mathcal{F}_{TKG}$=$\{(h, r, t, [\tau_b, \tau_e])\}$ into H-Facts form $\mathcal{F}_{TKG}^{HKG}$=$\{(h, r, t, begin\!:\!\tau_b, end\!:\!\tau_e)\}$. Then according to the previous transformation in HKG, it can be translated into the HiDR form that $\mathcal{G}_{TKG}^{HiDR}$=$\{\mathcal{V}, \mathcal{R}, \mathcal{F}_{TKG}^{HiDR}\}$ following the definition, where $\mathcal{F}_a$ =$\{(h, r, t) \,|\, (h, r, t, begin\!: \tau_b, end\!:$

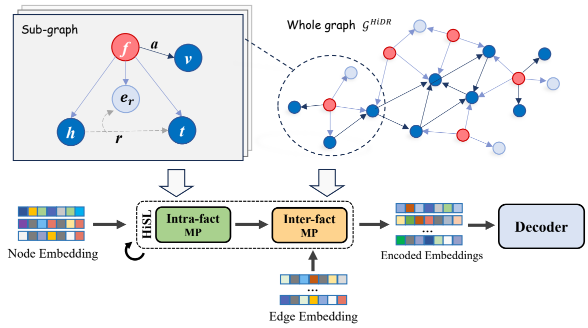

Figure 3: HiSL module for intra-fact and inter-fact MP.

$\tau_e) \in \mathcal{F}_{TKG}^{HKG}\}$, $\mathcal{F}_c = \{(f, has\,relation, e_r), (f, has\,head\,entity, h), (f, has\,tail\,entity, t), (f, begin, \tau_b), (f, end, \tau_e) \mid f = (h, r, t, begin{:}\, \tau_b, end{:}\, \tau_e) \in \mathcal{F}_{TKG}^{HKG}\}$ and $\mathcal{F}_n = \varnothing$.

In summary, we could convert all above KGs into the HiDR form, and preserve the semantics in the original KGs without loss of information.

4.2 Hierarchical Structure Learning

In this section, we illustrate how various KGs in the form of HiDR can be encoded by the Hierarchical Structure Learning module, abbreviated as **HiSL** shown in Fig. 3.

Representation Initialization. We first initialize the embedding matrices $\mathbf{H}_a \in \mathbb{R}^{|\mathcal{V}_a| \times d}$ and $\mathbf{E} \in \mathbb{R}^{|\mathcal{R}^{HiDR}| \times d}$ for atomic nodes and all relation edges. Then we also initialize the embedding of relation node $\mathbf{H}_r \in \mathbb{R}^{|\mathcal{V}_r| \times d}$, which can be transformed from the relation edge r, define as:

$$\mathbf{H}_r = \mathbf{E_a} \cdot \mathbf{W_r}, \tag{1}$$

where $\mathbf{E_a} \subseteq \mathbf{E}, \mathbf{W_r} \in \mathbb{R}^{d \times d}$ denote the atomic relation embeddings and a projection matrix. Then we initialize the fact node embeddings $\mathbf{H}_f$ to explicitly capture key information within facts by utilizing the embedding of (main) triple:

$$\mathbf{h}_f = f_m([\mathbf{h}_h; \mathbf{h}_r; \mathbf{h}_t]), \tag{2}$$

where $(h, r, t) \in \mathcal{F}_a$, the operation $[\cdot;\cdot]$ is the vector concatenation, $\mathbf{h}_h, \mathbf{h}_t \subseteq \mathbf{H}_a, \mathbf{h}_r \subseteq \mathbf{H}_r$ denote (main) triple embedding and f_m: $\mathbb{R}^{3d} \rightarrow \mathbb{R}^d$ is a 1-layer MLP.

For numerical atomic nodes, namely timestamps in temporal knowledge graphs, we utilize the Time2Vec (Kazemi et al., 2019) to encode the timestamp τ into an embedding:

$$\mathbf{h}_\tau = \omega_p \sin(f_p(\tau)) + f_{np}(\tau), \tag{3}$$

where f_p: $\mathbb{R}^1 \rightarrow \mathbb{R}^d$ is a 1-layer MLP as periodic function, f_{np}: $\mathbb{R}^1 \rightarrow \mathbb{R}^d$ is a 1-layer MLP as non-periodic function, and $\omega_p \in \mathbb{R}^1$ is a learnable parameter for scaling the periodic features.

Intra-fact Message Passing. In this stage, message passing is conducted for fact nodes. Given a fact node $f_k \in \mathcal{V}_f$, we construct its constituent elements, i.e., one-hop neighbors, as the node set $\mathcal{V}_k = \{v \in \mathcal{N}_{f_k} \mid v \in \mathcal{V}_a \cup \mathcal{V}_r\}$, where $\mathcal{N}_{f_k}$ is the set of one-hop neighbors of fact node f_k. Then we retain the edges directly connected to fact node f_k, thereby constructing a subgraph $\mathcal{G}_k = \{\mathcal{V}_k, \mathcal{R}_k, \mathcal{F}_k\} \subseteq \mathcal{G}^{HiDR}$. For this subgraph, we employ the graph attention network (GAT) (Brody et al., 2021) to aggregate local information, computing the attention score $\alpha_{i,j}$ between node $i \in \mathcal{V}_k$ and its neighbor j. The formula for calculating $\alpha_{i,j}$ in the l-th layer is as follows:

$$\alpha_{i,j}^l = \frac{\exp\left(\mathbf{W}^l\left(\sigma\left(\mathbf{W}_{in}^l\mathbf{h}_i^l + \mathbf{W}_{out}^l\mathbf{h}_j^l\right)\right)\right)}{\sum\limits_{j' \in \mathcal{N}_i} \exp\left(\mathbf{W}^l\left(\sigma\left(\mathbf{W}_{in}^l\mathbf{h}_i^l + \mathbf{W}_{out}^l\mathbf{h}_{j'}^l\right)\right)\right)}, \tag{4}$$

where $\mathbf{h}_i^l, \mathbf{h}_j^l \in \mathbb{R}^d$ represent the embeddings of node i and its neighbor j in l-th layer. And there are three learnable weight matrices $\mathbf{W}_{in}^l, \mathbf{W}_{out}^l \in \mathbb{R}^{d \times d}$ and $\mathbf{W}^l \in \mathbb{R}^d$. We choose LeakyReLU as activation function σ. Then, the updated node embeddings are obtained by aggregating the information of neighbors according to the attention scores:

$$\mathbf{h}_i^l = \mathbf{h}_i^l + \sum_{j \in \mathcal{N}_i} \alpha_{i,j}^l \cdot \mathbf{W}_{out}^l \mathbf{h}_j^l. \tag{5}$$

Inter-fact Message Passing. At this stage, message passing is conducted on the whole graph $\mathcal{G}^{HiDR}$. Similar to previous work (Vashishth et al., 2019), we use a non-parametric aggregation operator $\phi(\cdot): \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ to obtain messages from neighbouring nodes and edges. Specifically, we employ the circular-correlation operator inspired from HolE (Nickel et al., 2016), defined as:

$$\phi(\mathbf{h}_j, \mathbf{e}_r) = \mathbf{h}_j \star \mathbf{e}_r = \mathbf{F}^{-1}\left((\mathbf{F}\mathbf{h}_j) \odot \overline{(\mathbf{F}\mathbf{e}_r)}\right) \tag{6}$$

where $\mathbf{F}$ and $\mathbf{F}^{-1}$ denote the discrete fourier transform (DFT) matrix and its inverse matrix, the $\odot$ is the element-wise (Hadamard) product. Furthermore, in order to fully capture the heterogeneity of the graph, we classify edges along two dimensions: $\lambda(r) \in \{forward, reverse\}$ and $\tau(r) \in \{connected\ relation, atomic\ relation, nested\ relation\}$ and adopt two relation-type specific learnable parameters $\mathbf{W}_{\lambda(r)} \in \mathbb{R}^{d \times d}$ and $\omega_{\tau(r)} \in \mathbb{R}^1$ for more fine-grained aggregation:

$$\mathbf{h}_i^{l+1} = \sum_{(r,j)\in\mathcal{N}(i)} \sigma\left(\omega_{\tau(r)}^l\right) \mathbf{W}_{\lambda(r)}^l \phi\left(\mathbf{h}_j^l, \mathbf{e}_r^l\right) + \mathbf{W}_{self}^l \mathbf{h}_i^l \quad (7)$$

$$\mathbf{e}_r^{l+1} = \mathbf{W}_{rel}^l \mathbf{e}_r^l \quad (8)$$

where $\mathbf{W}_{self}^l, \mathbf{W}_{rel}^l \in \mathbb{R}^{d\times d}$, σ is a sigmoid activation function and $\mathcal{N}(i)$ is a set of immediate neighbors of i for its outgoing edges r. We utilize $\phi(\cdot)$ to combine the information from both edge r and node j, and then passes it to node i. Subsequently, node i aggregates the information according to the types of edge r separately to update its embedding, while edge r is also projected into the same embedding space as the updated nodes.

4.3 Link Prediction Decoder

Since the query varies across different settings, we use the transformer (Vaswani et al., 2017) as the decoder with the *mask* pattern. Specifically, we convert the updated node and edge embeddings into a sequence of fact embeddings, mask the elements to be predicted in facts with the $[M]$ token as the input to the transformer. Finally, we obtain the embedding of output $[M]$ in the last layer to measure the plausibility of the fact, denoted as $\mathbf{h}_{pre}$, and calculate the probability distribution of candidates, followed by training it using the cross-entropy loss function:

$$P = \text{Softmax}\left(f\left(\mathbf{h}_{pre}\right)[\mathbf{E};\mathbf{H}]^\top\right), \quad (9)$$

$$\mathcal{L} = \sum_{t=0}^{|\mathcal{R}|+|\mathcal{V}|} y_t \log P_t \quad (10)$$

where $P \in \mathbb{R}^{|\mathcal{R}|+|\mathcal{V}|}$ represents the confidence scores of all candidates, f: $\mathbb{R}^d \to \mathbb{R}^d$ is a 1-layer MLP, and $[\mathbf{E};\mathbf{H}] \in \mathbb{R}^{(|\mathcal{R}|+|\mathcal{V}|)\times d}$ is the embedding matrix of all candidate edges or nodes. The P_t and y_t are probability and ground truth of the t-th candidate. The final loss function $\mathcal{L}$ includes both node loss and edge loss during the predictions.

5 Experiment

5.1 Experiment Settings

Datasets. For link prediction on HKGs, we select three benchmark datasets: WikiPeople (Guan et al., 2021) , WD50K (Galkin et al., 2020) and JF17K (Wen et al., 2016). As for the NKGs, we choose FBH, FBHE and DBHE constructed by (Chung and Whang, 2023). Lastly, we use wikidata12k (Dasgupta et al., 2018), a subset of wikidata (Vrandečić and Krötzsch, 2014) for link prediction on TKGs. The statics of datasets are given in Appendix D.

Evaluation metric. We conduct link prediction across multiple settings, evaluating performance based on the rank of predicted facts. We use the MR (Mean Rank), MRR (Mean Reciprocal Rank) and Hits@K (K=1,3,10) as our evaluation metrics. And we choose to adopt the metrics used in prior works. Additionally, we employ filtering settings (Bordes et al., 2013) during the evaluation process to eliminate existing facts in the dataset.

Baselines. For link prediction on HKG, we compare our UniHR against NaLP (Guan et al., 2021), tNaLP (Guan et al., 2021), RAM (Liu et al., 2021), HINGE (Rosso et al., 2020), NeuInfer (Guan et al., 2020), StarE (Galkin et al., 2020), HyTransformer (Yu and Yang, 2021), GRAN (Wang et al., 2021) and HyNT (Chung et al., 2023). For link prediction on NKG, QuatE (Zhang et al., 2019), BiQUE (Guo and Kok, 2021), Neural-LP (Yang et al., 2017), DRUM (Sadeghian et al., 2019), AnyBURL (Meilicke et al., 2019), BiVE (Chung and Whang, 2023) and NestE (Xiong et al., 2024) are chosen as baselines. BiVE and NestE are especially designed for NKG. We compare against following TKG link prediction methods: ComplEx-N3 (Lacroix et al., 2018), TTransE (Leblay and Chekol, 2018), HyTE (Dasgupta et al., 2018), TA-DistMult (Garcia-Duran et al., 2018), ATiSE (Xu et al., 2019), TeRo (Xu et al., 2020), TASTER (Wang et al., 2023), TGeomE+ (Xu et al., 2023a).

Implementation details. All experiments are conducted on a single Nvidia 80G A800 GPU and implemented with PyTorch. For base link prediction on NKGs, we also use augmented triples from (Chung and Whang, 2023) for training to ensure fairness. For triple prediction, due to the small size of training set, we conduct training based on fixed embeddings of entities obtained from the base link prediction and set $\omega_{nested\ relation}$=0 to prevent overfitting. We employ AdamW (Kingma and Ba, 2015) optimizer. Hyperparameters are chosen by using a grid search based on the MRR performance and details can be found in Appendix F.

5.2 Main Results

Link prediction on HKG. We compare our method with previous methods on the WD50K and WikiPeople datasets shown in Table 1. We can observe that the models based on transformers (i.e., StarE, GRAN, HyNT) demonstrate significantly better performance compared to other models. We attribute this to the transformer's supe-

Model	WikiPeople							WD50K						
	# Tr. Params	subject/object			all entities			# Tr. Params	subject/object			all entities		
		MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10		MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
NaLP	-	0.356	0.271	0.499	0.360	0.275	0.503	-	0.230	0.170	0.347	0.251	0.187	0.375
tNaLP	-	0.358	0.288	0.486	0.361	0.290	0.490	-	0.221	0.163	0.331	0.243	0.182	0.360
RAM	27.34M	0.459	0.384	0.584	0.461	0.386	0.585	-	0.276	0.210	0.399	0.296	0.232	0.416
HINGE	-	0.393	0.309	0.547	0.395	0.311	0.549	-	0.264	0.187	0.410	0.277	0.200	0.424
NeuInfer	-	0.357	0.247	0.533	0.357	0.248	0.532	-	0.220	0.154	0.347	0.225	0.158	0.355
StarE	7.84M	0.458	0.364	<u>0.611</u>	-	-	-	10.39M	0.309	0.234	0.452	-	-	-
HyTransformer	-	0.460	0.382	0.594	-	-	-	-	0.304	0.231	0.443	-	-	-
GRAN	15.26M	0.462	0.366	0.610	0.465	0.371	<u>0.613</u>	18.51M	0.330	0.255	0.472	<u>0.361</u>	0.286	<u>0.501</u>
HyNT	23.02M	<u>0.482</u>	<u>0.415</u>	0.602	<u>0.481</u>	<u>0.414</u>	0.603	29.61M	<u>0.333</u>	<u>0.259</u>	<u>0.474</u>	0.360	<u>0.287</u>	0.500
UniHR	8.02M	**0.491**	**0.417**	**0.618**	**0.493**	**0.420**	**0.621**	10.55M	**0.348**	**0.278**	**0.482**	**0.382**	**0.313**	**0.513**

Table 1: Results of link prediction on HKG datasets. All baselines' results are taken from (Chung et al., 2023). The best results are written bold, while the second are underlined. # Tr. Params denotes the number of learnable parameters during training.

Model	FBHE/FBH		DBHE		FBH				FBHE				DBHE			
	MRR	Hits@10	MRR	Hits@10	#Tr. Params	MR	MRR	Hits@10	#Tr. Params	MR	MRR	Hits@10	#Tr. Params	MR	MRR	Hits@10
	Base link prediction				*Triple prediction*											
QuatE	0.354	0.581	0.264	0.440	-	145603.8	0.103	0.114	-	94684.4	0.101	0.209	-	26485.0	0.157	0.179
BiQUE	0.356	0.583	0.274	0.446	-	81687.5	0.104	0.115	-	61015.2	0.135	0.205	-	19079.4	0.163	0.185
Neural-LP	0.315	0.486	0.233	0.357	-	115016.6	0.070	0.073	-	90000.4	0.238	0.274	-	21130.5	0.170	0.209
DRUM	0.317	0.490	0.237	0.359	-	115016.6	0.069	0.073	-	90000.3	0.261	0.274	-	21130.5	0.166	0.209
AnyBURL	0.310	0.526	0.220	0.364	-	108079.6	0.096	0.108	-	83136.8	0.191	0.252	-	20530.8	0.177	0.214
BiVE	0.370	0.607	0.274	0.422	12.66M	6.20	0.855	0.941	12.67M	8.35	0.711	0.866	10.74M	3.63	0.687	0.958
NestE	<u>0.371</u>	0.608	<u>0.289</u>	<u>0.443</u>	12.11M	<u>3.34</u>	<u>0.922</u>	<u>0.982</u>	12.46M	**3.05**	**0.851**	**0.962**	10.22M	<u>2.07</u>	**0.862**	<u>0.984</u>
UniHR	**0.401**	**0.619**	**0.300**	**0.455**	4.12M	**2.46**	**0.946**	**0.993**	4.12M	<u>5.20</u>	<u>0.793</u>	<u>0.890</u>	3.67M	**1.90**	**0.862**	**0.987**

Table 2: Results of base link prediction (left) and triple prediction (right). All baselines' results are taken from (Xiong et al., 2024). For BiVE (Chung and Whang, 2023) and NestE (Xiong et al., 2024), we pick their best variants.

Model	wikidata12k				
	# Tr. Params	MRR	Hits@1	Hits@3	Hits@10
ComplEx-N3	-	0.248	0.143	-	0.489
TTransE	-	0.172	0.096	0.184	0.329
HyTE	1.69M	0.253	0.147	-	0.483
TA-DistMult	-	0.230	0.130	-	0.461
TeRo	12.91M	0.299	0.198	0.329	0.507
ATiSE	31.46M	0.252	0.148	0.288	0.462
TASTER	5.04M	0.327	<u>0.239</u>	-	0.514
TGeomE+	-	**0.333**	0.232	<u>0.361</u>	**0.546**
UniHR	3.68M	**0.333**	**0.240**	**0.367**	<u>0.527</u>

Table 3: Results of link prediction on wikidata12k.

rior ability to capture long distance dependencies within H-Facts. Among these methods, it can be seen that our proposed UniHR achieves state-of-the-art results, which means our method effectively captures global structural information. Compared to the GNN-based method StarE, we achieve improvements of 3.9 points (12.6%) in MRR, 4.4 points (18.8%) in Hits@1 and 3.0 points (6.6%) in Hits@10 on WD50K. This indicates that the performance of StarE's customized graph neural network is limited by its inability to flexibly capture key-value pair information. Moreover, since the embeddings for newly added fact nodes and relation nodes are computed from atomic facts, so our training parameters do not significantly increase.

Link prediction on NKG. Our experiments on the NKGs consist of two tasks: base link prediction and triple prediction. Unlike previous quaternion-based methods (Chung and Whang, 2023; Xiong et al., 2024), UniHR significantly reduces the number of required training parameters. From the results in Table 2, we can see that our proposed UniHR obtains competitive results as the first method to capture global structural information of NKGs. For base link prediction task, UniHR achieves considerable improvements on all datasets. Of particular note, the MRR of FBHE/FBH and DBHE increases by 8.1% and 3.8%, respectively.

For triple prediction, we perform best on FBH and DBHE datasets, especially obtaining an improvement of 2.4 points in MRR on FBH, and achieve the second-best performance on FBHE, which suggests that structural information is also valuable for NKG and UniHR can effectively capture the heterogeneity of NKG to enhance node embeddings. In particular, as a unified method, we do not use the customized decoder for triples, while previous state-of-the-art methods do. We will further illustrate the effectiveness of UniHR with other decoders in Appendix C.

Link prediction on TKG. As shown in Table 3, we achieve competitive results on the wikidata12k, even surpassing TGeomE+ by 3.4% on Hits@1 and 1.7% on Hits@3. However, existing temporal knowledge graph embedding methods, such

Variant	FBH			FBHE			DBHE		
	MR	MRR	Hits@10	MR	MRR	Hits@10	MR	MRR	Hits@10
w/o initial $\mathbf{h}_f$	5.22	0.909	0.980	5.86	0.767	0.885	2.56	0.794	0.978
w/o $\mathbf{W}_r$	3.06	0.944	0.992	5.98	0.792	0.885	2.30	0.850	0.976
w/o intra-fact MP	3.97	0.897	0.972	5.26	0.754	0.883	2.02	0.842	0.983
w/o $\omega_{\tau(r)}$	2.70	0.934	0.992	5.46	0.782	0.888	2.69	0.810	0.973
w/o $\mathbf{W}_{\lambda(r)}$	2.50	0.941	0.992	5.69	0.778	0.889	2.37	0.810	0.978
w/o inter-fact MP	2.61	0.913	0.991	**4.56**	0.776	0.887	2.11	0.827	0.986
UniHR	**2.46**	**0.946**	**0.993**	5.20	**0.793**	**0.890**	**1.90**	**0.862**	**0.987**

Table 4: Ablation studies on the HiSL module for triple prediction. Best results are boldfaced.

Model	WikiPeople$^-$										wikidata12k$^-$				
	subject/object					all entities					subject/object				
	MR	MRR	Hits@1	Hits@3	Hits@10	MR	MRR	Hits@1	Hits@3	Hits@10	MR	MRR	Hits@1	Hits@3	Hits@10
UniHR	835.8	0.486	**0.412**	0.528	0.617	829.0	0.488	0.414	0.531	0.620	818.7	0.314	0.220	0.345	**0.509**
UniHR$_{Joint}$	**692.7**	**0.488**	0.409	**0.533**	**0.629**	**686.5**	**0.490**	**0.414**	**0.536**	**0.632**	**489.5**	**0.315**	**0.222**	**0.346**	0.498

Table 5: Results of separate training and joint training on the HKG and TKG dataset, where identical entities and relations share the same embeddings. WikiPeople$^-$ and wikidata12k$^-$ represent the filtered test sets.

as TGeomE+, often employ time-aware decoders, which are challenging to generalize to other types of KGs. In contrast, our approach efficiently encodes timestamps as atomic nodes only during initialization and learns temporal information through message passing on graph structure, demonstrating that graph structure information is also beneficial for temporal knowledge graphs, highlighting the effectiveness of our UniHR.

5.3 Further Analysis

Ablation study on HiSL. We conduct ablation experiments on triple prediction, the most relevant task to fact nodes, using three NKG datasets. As shown in Table 4, all variant models with certain modules or parameters removed exhibit a decrease in performance. We can conclude that intra-fact and inter-fact message passing modules both play crucial roles, allowing UniHR to more fully represent the current fact node with enhanced structural information. We also change the initialization of fact node embeddings to learnable embeddings (w/o initial $\mathbf{h}_f$) in HiSL. There is a performance decrease, indicating that initializing fact node representations is essential. It highlights key information in the facts, mitigating the noise introduced by excessive neighbors.

Potential of Joint Learning on Different Types of KGs. We suppose that unified representation makes it possible to develop pre-trained models that integrate multiple types of KGs. To explore its potential, we conduct joint learning on different types of KGs. Therefore, we construct a hybrid dataset called **wikimix** which includes two subsets of Wikidata (Vrandečić and Krötzsch, 2014), namely HKG dataset WikiPeople and TKG dataset wikidata12k, which encompass 3547 identical entities and 18 identical relations. Due to the different types of facts, there are no identical facts in these two subsets. To further prevent data leakage, we filter out 537 entries from the wikidata12k test set whose main triples appear in the H-Facts of WikiPeople train set, and 384 entries from the WikiPeople test set whose main triples appear in wikidata12k train set. Statics of wikimix are given in Appendix D.

From the results in Table 5, it is evident that joint learning outperforms separate learning across most metrics. Notably, there are improvements of 17.1% and 40.2% in MR metric on wikipeople$^-$ and wikidata12k$^-$ datasets, respectively. This indicates that more complex structural interactions and diverse types of training data are beneficial. Moreover, our UniHR demonstrates good scalability and effectiveness in integrating multiple types of KGs.

6 Conclusion

In this paper, we propose UniHR, a unified hierarchical knowledge graph representation learning framework consisting of a Hierarchical Data Representation (HiDR) module and a Hierarchical Structure Learning (HiSL) module. The HiDR form unifies the hyper-relational facts, nested facts and temporal facts into the form of triples, overcoming the limitations of customized encoders for different forms of facts. Moreover, HiSL captures local semantic information within facts and global structural information between facts. Our approach achieves the best or competitive performance on link prediction tasks across three types of KGs.

Limitations

The limitations of our paper are summarized as follows:

Our UniHR In this paper, our UniHR framework focuses on link prediction tasks under transductive settings with a single modality. In the future, we will investigate how to generalize our HiDR form to more complex tasks such as inductive reasoning (Teru et al., 2020) and multi-modal scenarios (Zhang et al., 2024), etc.

Joint Learning on Different Types of KGs Constrained by computational resources, our analysis of the potential for joint training across multiple types of knowledge graphs focuse only on HKG and TKG. We believe the unification of knowledge graph representation learning methods is a developing trend that makes it possible to develop unified pre-trained models based on multiple types of KGs. In the future, we aim to explore joint training across more types of KG to demonstrate the advantages of integrating multi-type KG data.

Ethics Statement

In this paper, we explore the unified knowledge graph link prediction problem, aiming to complete various types of knowledge graphs using a unified model with deep learning techniques. Our training and evaluation are based on publicly available and widely used datasets of different types of knowledge graphs. Therefore, we believe this does not violate any ethics.

References

Waqas Ali, Muhammad Saleem, Bin Yao, Aidan Hogan, and Axel-Cyrille Ngonga Ngomo. 2022. A survey of RDF stores & SPARQL engines for querying knowledge graphs. *VLDB J.*, 31(3):1–26.

KM Annervaz, Somnath Basu Roy Chowdhury, and Ambedkar Dukkipati. 2018. Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 313–322.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Shaked Brody, Uri Alon, and Eran Yahav. 2021. How attentive are graph attention networks? In *International Conference on Learning Representations*.

Chanyoung Chung, Jaejun Lee, and Joyce Jiyoung Whang. 2023. Representation learning on hyper-relational and numeric knowledge graphs with transformers. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pages 310–322. ACM.

Chanyoung Chung and Joyce Jiyoung Whang. 2023. Learning representations of bi-level knowledge graphs for reasoning beyond link prediction. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 4208–4216. AAAI Press.

Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018. Hyte: Hyperplane-based temporally aware knowledge graph embedding. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2001–2011.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1811–1818. AAAI Press.

Mikhail Galkin, Priyansh Trivedi, Gaurav Maheshwari, Ricardo Usbeck, and Jens Lehmann. 2020. Message passing for hyper-relational knowledge graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7346–7359.

Alberto Garcia-Duran, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4821.

Saiping Guan, Xiaolong Jin, Jiafeng Guo, Yuanzhuo Wang, and Xueqi Cheng. 2020. Neuinfer: Knowledge inference on n-ary facts. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6141–6151.

Saiping Guan, Xiaolong Jin, Jiafeng Guo, Yuanzhuo Wang, and Xueqi Cheng. 2021. Link prediction on n-ary relational data based on relatedness evaluation. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):672–685.

Jia Guo and Stanley Kok. 2021. Bique: Biquaternionic embeddings of knowledge graphs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8338–8351.

Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568.

Magdalena Kaiser, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Reinforcement learning from reformulations in conversational question answering over knowledge graphs. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 459–469.

Seyed Mehran Kazemi, Rishab Goel, Sepehr Eghbali, Janahan Ramanan, Jaspreet Sahota, Sanjay Thakur, Stella Wu, Cathal Smyth, Pascal Poupart, and Marcus A. Brubaker. 2019. Time2vec: Learning a vector representation of time. *CoRR*, abs/1907.05321.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. 2018. Canonical tensor decomposition for knowledge base completion. In *International Conference on Machine Learning*, pages 2863–2872. PMLR.

Julien Leblay and Melisachew Wudage Chekol. 2018. Deriving validity time in knowledge graph. In *Companion proceedings of the the web conference 2018*, pages 1771–1776.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Ye Liu, Hui Li, Alberto García-Durán, Mathias Niepert, Daniel Oñoro-Rubio, and David S. Rosenblum. 2019. MMKG: multi-modal knowledge graphs. In *The Semantic Web - 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2-6, 2019, Proceedings*, volume 11503 of *Lecture Notes in Computer Science*, pages 459–474. Springer.

Yu Liu, Quanming Yao, and Yong Li. 2021. Role-aware modeling for n-ary relational knowledge bases. In *Proceedings of the Web Conference 2021*, pages 2660–2671.

Christian Meilicke, Melisachew Wudage Chekol, Daniel Ruffinelli, and Heiner Stuckenschmidt. 2019. Anytime bottom-up rule learning for knowledge graph completion. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3137–3143.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. 2019. Learning attention-based embeddings for relation prediction in knowledge graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4710–4723.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A review of relational machine learning for knowledge graphs. *Proc. IEEE*, 104(1):11–33.

Paolo Rosso, Dingqi Yang, and Philippe Cudré-Mauroux. 2020. Beyond triplets: hyper-relational knowledge graph embedding for link prediction. In *Proceedings of the web conference 2020*, pages 1885–1896.

Ali Sadeghian, Mohammadreza Armandpour, Patrick Ding, and Daisy Zhe Wang. 2019. Drum: End-to-end differentiable rule mining on knowledge graphs. *Advances in Neural Information Processing Systems*, 32.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2018. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

Komal Teru, Etienne Denis, and Will Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning*, pages 9448–9457. PMLR.

Kristina Toutanova and Danqi Chen. 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, CVSC 2015, Beijing, China, July 26-31, 2015*, pages 57–66. Association for Computational Linguistics.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based multi-relational graph convolutional networks. In *International Conference on Learning Representations*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Quan Wang, Haifeng Wang, Yajuan Lyu, and Yong Zhu. 2021. Link prediction on n-ary relational facts: A graph-based approach. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 396–407.

Xin Wang, Shengfei Lyu, Xiangyu Wang, Xingyu Wu, and Huanhuan Chen. 2023. Temporal knowledge graph embedding via sparse transfer matrix. *Inf. Sci.*, 623:56–69.

Jianfeng Wen, Jianxin Li, Yongyi Mao, Shini Chen, and Richong Zhang. 2016. On the representation and embedding of knowledge bases beyond binary relations. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, pages 1300–1307.

Bo Xiong, Mojtaba Nayyeri, Daniel Daza, and Michael Cochez. 2023. Reasoning beyond triples: Recent advances in knowledge graph embeddings. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 5228–5231. ACM.

Bo Xiong, Mojtaba Nayyeri, Linhao Luo, Zihao Wang, Shirui Pan, and Steffen Staab. 2024. Neste: Modeling nested relational structures for knowledge graph reasoning. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 9205–9213. AAAI Press.

Chengjin Xu, Mojtaba Nayyeri, Fouad Alkhoury, Hamed Shariat Yazdi, and Jens Lehmann. 2019. Temporal knowledge graph embedding model based on additive time series decomposition. *arXiv preprint arXiv:1911.07893*.

Chengjin Xu, Mojtaba Nayyeri, Fouad Alkhoury, Hamed Shariat Yazdi, and Jens Lehmann. 2020. Tero: A time-aware knowledge graph embedding via temporal rotation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1583–1593.

Chengjin Xu, Mojtaba Nayyeri, Yung-Yu Chen, and Jens Lehmann. 2023a. Geometric algebra based embeddings for static and temporal knowledge graph completion. *IEEE Trans. Knowl. Data Eng.*, 35(5):4838–4851.

Hongcai Xu, Junpeng Bao, and Wenbo Liu. 2023b. Double-branch multi-attention based graph neural network for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15257–15271.

Fan Yang, Zhilin Yang, and William W Cohen. 2017. Differentiable learning of logical rules for knowledge base reasoning. *Advances in neural information processing systems*, 30.

Donghan Yu and Yiming Yang. 2021. Improving hyper-relational knowledge graph completion. *arXiv preprint arXiv:2104.08167*.

Richong Zhang, Junpeng Li, Jiajie Mei, and Yongyi Mao. 2018. Scalable instance reconstruction in knowledge bases via relatedness affiliated embedding. In *Proceedings of the 2018 world wide web conference*, pages 1185–1194.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. 2019. Quaternion knowledge graph embeddings. *Advances in neural information processing systems*, 32.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, and Huajun Chen. 2024. Native: Multi-modal knowledge graph completion in the wild. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 91–101. ACM.

A Related Works

A.1 Link Prediction Methods for Triple-based KGs

Most existing techniques in KG representation learning are proposed for triple-based KGs. Among these techniques, knowledge graph embedding (KGE) models (Bordes et al., 2013; Sun et al., 2018) have received extensive attention due to their effectiveness and simplicity. The idea is to project entities and relations in the KG to low-dimensional vector spaces, utilizing KGE scoring functions to measure the plausibility of triples in the embedding space. Typical methods include TransE (Bordes et al., 2013), RotatE (Sun et al., 2018), and ConvE (Dettmers et al., 2018).

Depending on the KGE model alone has limitation of capturing complex graph structures, whereas augmenting global structural information with a graph neural network (GNN) (Vashishth et al., 2019; Nathani et al., 2019; Xu et al., 2023b) proves to be an effective approach for enhancement. The paradigm of combining GNN as encoder with KGE scoring function as decoder helps to enhance the performance of KGE scoring function. These GNN methods design elaborate message passing mechanisms to capture the global structural features. Typically, CompGCN (Vashishth et al., 2019) aggregates the joint embedding of entities and relations in the neighborhood via a parameter-efficient way and MA-GNN (Xu et al., 2023b) learns global-local structural information based on multi-attention. These methods achieve impressive results on triple-based KGs but are hard to generalize to beyond-triple KGs.

B Results on JF17K

Table 6 shows the experimental results on JF17K. Due to the absence of a validation set in the JF17K dataset and the different ways of dividing the dataset across various baselines, we adopt the results reported in the original paper. Consistent with previous experiments on hyper-relational knowledge graphs, we also achieve state-of-the-art performance on JF17K among all baselines. In particular, we achieved 1.7 (2.9%) points in MRR and 1.5 (2.9%) points in Hits@1 compared to the method StarE which also utilises a graph neural network encoding and a simple transformer decoding, indicating that our hierarchical GNN HiSL could better capture the structure of hyper-relational facts.

Model	JF17K					
	subject/object			all entities		
	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10
m-TransH	0.206	0.206	0.462	-	-	-
NaLP	0.221	0.165	0.331	0.366	0.290	0.516
HINGE	0.431	0.342	0.611	0.517	0.436	0.675
NeuInfer	0.449	0.361	0.624	0.473	0.397	0.618
StarE	0.574	0.496	0.725	-	-	-
HyTransformer	0.582	0.501	0.742	-	-	-
UniHR	**0.591**	**0.511**	**0.745**	**0.621**	**0.545**	**0.768**

Table 6: Link prediction on JF17K. All results of baselines are taken from the original paper. Best results are in bold.

C Decoder Analysis

To explore the effectiveness of our UniHR encoding further, we pair UniHR with different decoders and evaluated them on triple prediction task. In addition to the previously mentioned unified framework **UniHR + Transformer**, we also experiment on **UniHR + ConvE** with two scoring strategies. The ConvE (Dettmers et al., 2018) is the decoder customized for triples and its scoring function is $vec\left(\sigma\left(\left[\tilde{\mathbf{h}}_h; \tilde{\mathrm{e}}_r\right] * \psi\right)\right)$, where $\tilde{\mathbf{h}}_h$ and $\tilde{\mathrm{e}}_r$ represent reshaped 2D embeddings of head entity h and relation r, and $*$ is a convolution operator. The $vec(\cdot)$ and ψ are denoted as the vectorization function and a set of convolution kernels.

Due to our special representation, there exists two scoring methods for atomic triples, thus we present the base link prediction results separately for each scoring method. The s_f represents scoring triples $(f, has\ head\ entity, h)$ and $(f, has\ tail\ entity, t)$, and s_t represents scoring (h, r, t). The performance of base link prediction is shown in Table 7. Notably, FBH and FBHE share identical atomic facts, resulting in the same performance. It can be observed that regardless of the scoring method employed, we both achieve competitive performance, especially with scoring (h, r, t) on FBH and scoring $(f, has\ head\ entity, h)$ $(f, has\ tail\ entity, t)$ on DBHE. We attribute the differences in performance under different scoring methods to dataset characteristics. The DBHE dataset is relatively smaller, and scoring method s_f effectively alleviates overfitting problem. Conversely, for larger datasets FBH, scoring based on (h, r, t) minimizes information loss.

Table 8 shows the results of triple prediction on three benchmark datasets. Among all baselines, Quate, Bique, Neural-LP, Drum, and AnyBURL struggle to model the mapping relationship between atomic facts and nested facts. Furthermore, prior works (Chung and Whang, 2023) do not guarantee that all atomic facts in the nested fact test set are present in the training set as entities, which shifts the problem from a transductive setting to an inductive setting, leading to significant performance gaps between these baselines. On most metrics, our method outperforms BiVE and NestE which are specifically modeled for nested facts. Notably, NestE fully preserves the semantics of atomic facts. However, on the FBHE dataset, UniHR + ConvE achieves an improvement of 0.58 (6.4%) points in MRR and 0.24 (2.4%) points in Hits@10 compared to the state-of-the-art model NestE and the second-best performance after UniHR + Transformer on the FBH and DBHE datasets, demonstrating UniHR's powerful graph structure encoding capabilities. We also carry out ablation experiments on UniHR + ConvE as shown in Table 8. Performance declines are observed after removing any part of the HiSL module, showing the significance of HiSL for hierarchical encoding.

D Datasets Statistics

Table 9 shows the details of the three hyper-relational knowledge graph benchmark datasets: WikiPeople, WD50K, JF17K, three nested factual knowledge graph benchmark datasets: FBH, FBHE, DBHE,and the temporal knowledge graph benchmark dataset wikidata12k. Among them, WikiPeople is a dataset derived from Wikidata

Algorithm 1 Message passing process of HiSL

Input: $\mathcal{G}^{HiDR} = \{\mathcal{V}_a \cup \mathcal{V}_r \cup \mathcal{V}_f, \mathcal{R}_a \cup \mathcal{R}_n \cup \mathcal{R}_c, \mathcal{F}_a \cup \mathcal{F}_c \cup \mathcal{F}_n\}$; the number of encoder layers L.
Output: The node embedding matrix $\mathbf{H}^L$; the edge embedding matrix $\mathbf{E}^L$.

Initialize the embedding matrix of atomic nodes $\mathbf{H}_a^0 \in \mathbb{R}^{|\mathcal{V}_a| \times d}$.
Initialize the embedding matrix of three types of relations $\mathbf{E}^0 = \{\mathbf{E}_a^0, \mathbf{E}_c^0, \mathbf{E}_n^0\} \in \mathbb{R}^{|\mathcal{R}_a \cup \mathcal{R}_c \cup \mathcal{R}_n| \times d}$.
Initialize the embedding matrix of relation nodes $\mathbf{H}_r^0 \leftarrow \mathbf{E}_a^0 \cdot \mathbf{W}_r \in \mathbb{R}^{|\mathcal{V}_r| \times d}$.
for $h, r, t \in \mathcal{F}_a$ **do**
  Initialize the embedding of fact nodes $\mathbf{h}_f^0 \leftarrow f_m([\mathbf{h}_h; \mathbf{h}_r; \mathbf{h}_t])$. ▷ Representation Initialization
end for
for $l \leftarrow 1$ to L **do**
  for $i \in \mathcal{V}$ **do**
    $\mathcal{M}_i^l \leftarrow \mathbf{h}_i^{l-1}$. ▷ Intra-fact Message Passing
    for $i, r, j \in \mathcal{F}_c$ **do**
      $\alpha_{i,j}^{l-1} \leftarrow \frac{\exp\left(\mathbf{W}^{l-1}\left(\sigma\left(\mathbf{W}_{in}^{l-1}\mathbf{h}_i^{l-1} + \mathbf{W}_{out}^{l-1}\mathbf{h}_j^{l-1}\right)\right)\right)}{\sum_{j' \in \mathcal{N}_i} \exp\left(\mathbf{W}^{l-1}\left(\sigma\left(\mathbf{W}_{in}^{l-1}\mathbf{h}_i^{l-1} + \mathbf{W}_{out}^{l-1}\mathbf{h}_{j'}^{l-1}\right)\right)\right)}$.
      $\mathcal{M}_i^l \leftarrow \{\alpha_{i,j}^{l-1} \cdot \mathbf{W}_{out}^{l-1}\mathbf{h}_j^{l-1}\} \cup \mathcal{M}_i^l$.
    end for
  end for
  for $i \in \mathcal{V}$ **do**
    $\mathbf{h}_i^{l-1} \leftarrow \mathtt{Aggregate}(\mathcal{M}_i^l)$.
  end for ▷ Inter-fact Message Passing
  for $i \in \mathcal{V}$ **do**
    $\mathcal{M}_i^l \leftarrow \{\mathbf{W}_{self}^{l-1}\mathbf{h}_i^{l-1}\}$.
    for $r, j \in \mathcal{N}_i$ **do**
      $\mathbf{m}_{(i,r,j)}^{l-1} \leftarrow \sigma\left(\omega_{\tau(r)}^{l-1}\right)\mathbf{W}_{\lambda(r)}^{l-1}\phi\left(\mathbf{h}_j^{l-1}, \mathbf{e}_r^{l-1}\right)$.
      $\mathcal{M}_i^l \leftarrow \{\mathbf{m}_{(i,r,j)}^{l-1}\} \cup \mathcal{M}_i^l$.
    end for
  end for
  for $i \in \mathcal{V}$ **do**
    $\mathbf{h}_i^l \leftarrow \mathtt{Aggregate}(\mathcal{M}_i^l)$.
  end for
  $\mathbf{E}^l \leftarrow \mathbf{W}_{rel}^{l-1}\mathbf{E}^{l-1}$.
end for
return $\mathbf{H}^L, \mathbf{E}^L$

(Vrandečić and Krötzsch, 2014) concerning entities type “human”. WikiPeople filter out the elements which have at least 30 mentions as key-value pairs. WD50K is a high-quality dataset extracting from Wikidata statements and avoiding the potential data leakage which allows triple-based models to memorize main fact in the H-Facts of test set. The “with Q(%)" column in Table 9 denote the number of facts with auxiliary key-value pairs and the “Arity" column denote range of the number of entities in hyper-relational facts. The nested factual knowledge graph datasets FBH and FBHE (Chung and Whang, 2023) are constructed based on FB15k237 (Toutanova and Chen, 2015) from Freebase (Bollacker et al., 2008) while DBHE is based on DB15K (Liu et al., 2019) from DBpedia (Lehmann et al., 2015). FBH contains nested facts that can be only inferred inside the atomic facts, while FBHE and DBHE contain externally-sourced nested relation crawling from Wikipedia articles, e.g., NextAlmaMater and SucceededBy. Temporal knowledge graph dataset wikidata12K is also a subset of Wikidata (Vrandečić and Krötzsch, 2014), which represents the time information $\tau \in \mathcal{T}$ as time intervals.

E Pseudo-code of HiSL

The pseudo code of HiSL is outlined in Algorithm 1.

Model	FBHE/FBH		DBHE	
	MRR	Hits@10	MRR	Hits@10
QuatE	0.354	0.581	0.264	0.440
BiQUE	0.356	0.583	0.274	0.446
Neural-LP	0.315	0.486	0.233	0.357
DRUM	0.317	0.490	0.237	0.359
AnyBURL	0.310	0.526	0.220	0.364
BiVE	0.370	0.607	0.274	0.422
NestE	0.371	0.608	0.289	0.443
UniHR + ConvE s_h	0.397	**0.622**	0.289	0.443
UniHR + ConvE s_f	0.375	0.596	**0.307**	**0.471**
UniHR + Transformer	**0.401**	0.619	0.300	0.455

Table 7: Base link prediction on FBHE, FBH and DBHE. All baselines' results are taken from (Xiong et al., 2024). The best results among all models are written bold, while the second are underlined. The s_f and s_h denote $(f, has\ head\ entity, h)$ $(f, has\ tail\ entity, t)$ and (h, r, t) two types of scoring method respectively. For BiVE (Chung and Whang, 2023) and NestE (Xiong et al., 2024), we pick their variants with best performance.

F Hyperparameter Settings

Here, we show the hyperparameter details for each link prediction task. To be specific, we tune the learning rate using the range $\{0.0001, 0.0005, 0.001\}$, the embedding dim using the range $\{50, 100, 200, 400\}$, the GNN layer using the range $\{1, 2, 3\}$ and dropout using the range $\{0.1, 0.2, 0.3, 0.4\}$. Additionally, we use smoothing label in the training phase from range $\{0.1, 0.2, 0.3\}$. The best hyperparameters obtained from the experiments are presented in Table 10.

Model	FBH			FBHE			DBHE		
	MR	MRR	Hits@10	MR	MRR	Hits@10	MR	MRR	Hits@10
QuatE (Zhang et al., 2019)	145603.8	0.103	0.114	94684.4	0.101	0.209	26485.0	0.157	0.179
BiQUE (Guo and Kok, 2021)	81687.5	0.104	0.115	61015.2	0.135	0.205	19079.4	0.163	0.185
Neural-LP (Yang et al., 2017)	115016.6	0.070	0.073	90000.4	0.238	0.274	21130.5	0.170	0.209
DRUM (Sadeghian et al., 2019)	115016.6	0.069	0.073	90000.3	0.261	0.274	21130.5	0.166	0.209
AnyBURL (Meilicke et al., 2019)	108079.6	0.096	0.108	83136.8	0.191	0.252	20530.8	0.177	0.214
BiVE (Chung and Whang, 2023)	6.20	0.855	0.941	8.35	0.711	0.866	3.63	0.687	0.958
NestE (Xiong et al., 2024)	3.34	0.922	0.982	**3.05**	0.851	0.962	2.07	0.862	0.984
UniHR + Transformer	**2.46**	**0.946**	**0.993**	5.20	0.793	0.890	**1.90**	0.862	**0.987**
UniHR + ConvE	3.00	0.900	0.983	6.27	**0.909**	**0.986**	2.06	**0.876**	0.978
UniHR + ConvE w/o $\mathbf{h}_f$	4.39	0.887	0.979	10.10	0.865	0.970	2.76	0.798	0.961
UniHR + ConvE w/o intra-fact	6.54	0.859	0.959	18.10	0.871	0.968	5.82	0.665	0.900
UniHR + ConvE w/o inter-fact	12.56	0.864	0.961	20.56	0.864	0.966	10.75	0.764	0.951

Table 8: Triple prediction on FBHE, FBH and DBHE. All baselines' results are taken from (Xiong et al., 2024). The best results among all models are written bold. For BiVE (Chung and Whang, 2023) and NestE (Xiong et al., 2024), we pick their variants with best performance.

Dataset	Atomic Fact	Entities	Relations	Train	Valid	Test	with Q(%)	Arity	Nested Fact	Nested Relation	with AF(%)	Period
Hyper-relational Knowledge Graph												
WikiPeople	369866	34825	178	294439	37715	37712	9482(2.6%)	2-7	-	-	-	-
WD50K	236507	47155	531	166435	23913	46159	32167(13.6%)	2-67	-	-	-	-
JF17K	100947	28645	501	76379	-	24568	46320(45.9%)	2-6	-	-	-	-
Nested Factual Knowledge Graph												
FBH	310116	14541	237	248094	31011	31011	-	-	27062	6	33157	-
FBHE	310116	14541	237	248094	31011	31011	-	-	34941	10	33719	-
DBHE	68296	12440	87	54636	6830	6830	-	-	6717	8	8206	-
Temporal Knowledge Graph												
wikidata12k	40621	12554	24	32497	4062	4062	-	-	-	-	-	19-2020
Multiple types of Knowledge Graph												
wikimix	409566	43832	184	326936	41777	3525(TKG)/37328(HKG)	9098(2.2%)	2-7	-	-	-	19-2020

Table 9: The statistics of diverse knowledge graphs dataset, where "with Q(%)" and "Arity" column respectively denote the number of facts with auxiliary key-value pairs and the range of arity of hyper-relational facts, the "with AF(%)" column denotes the number of atomic facts in nested facts.

Hyperparameter	WikiPeople	WD50K	wikidata12k	FBHE$_{base}$	FBH$_{base}$	DBHE$_{base}$	FBHE$_{triple}$	FBH$_{triple}$	DBHE$_{triple}$
batch_size	2048	2048	2048	2048	2048	2048	2048	2048	2048
embedding dim	200	200	200	200	200	200	200	200	200
hidden dim	200	200	200	200	200	200	200	200	200
GNN_layer	2	2	2	2	2	2	2	2	2
GNN_intra-fact heads	4	4	4	4	4	4	4	4	4
GNN_intra-fact dropout	0.1	0.1	0.1	0.3	0.1	0.3	0.2	0.1	0.1
GNN_inter-fact activation	tanh	tanh	tanh	tanh	tanh	tanh	tanh	tanh	tanh
GNN_dropout	0.1	0.1	0.1	0.3	0.1	0.3	0.2	0.1	0.1
transformer layers	2	2	2	2	2	2	2	2	2
transformer heads	4	4	4	4	4	4	4	4	4
transfomer activation	gelu	gelu	gelu	gelu	gelu	gelu	gelu	gelu	gelu
decoder dropout	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
soft label for entity	0.2	0.2	0.4	0.2	0.2	0.3	0.2	0.2	0.2
soft label for relation	0.1	0.1	0.3	0.2	0.2	0.3	0.2	0.2	0.2
weight_decay	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
learning rate	5e-4	5e-4	5e-4	5e-4	5e-4	5e-4	5e-4	5e-4	5e-4

Table 10: The major hyperparameters of our approach for all link prediction tasks.