mathematics

MDPI

Article

Semantic-Enhanced Knowledge Graph Completion

Xu Yuan [1], Jiaxi Chen [1], Yingbo Wang [1], Anni Chen [2], Yiou Huang [3], Wenhong Zhao [4,*] and Shuo Yu [5]

[1] School of Software, Dalian University of Technology, Dalian 116620, China; david@dlut.edu.cn (X.Y.); chenjiaxii@mail.dlut.edu.cn (J.C.); yingbo.wang.dlut@outlook.com (Y.W.)
[2] School of Computer Science, University of Wollongong, Wollongong 2522, Australia; ac851@uowmail.edu.au
[3] School of Mathematics and Physics, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China; yiouh08@icloud.com
[4] Ultraprecision Machining Center, Zhejiang University of Technology, Hangzhou 310014, China
[5] School of Computer Science and Technology, Dalian University of Technology, Dalian 116024, China; shuo.yu@ieee.org
* Correspondence: zhaowh@zjut.edu.cn

Abstract: Knowledge graphs (KGs) serve as structured representations of knowledge, comprising entities and relations. KGs are inherently incomplete, sparse, and have a strong need for completion. Although many knowledge graph embedding models have been designed for knowledge graph completion, they predominantly focus on capturing observable correlations between entities. Due to the sparsity of KGs, potential semantic correlations are challenging to capture. To tackle this problem, we propose a model entitled semantic-enhanced knowledge graph completion (SE-KGC). SE-KGC effectively addresses the issue by incorporating predefined semantic patterns, enabling the capture of semantic correlations between entities and enhancing features for representation learning. To implement this approach, we employ a multi-relational graph convolution network encoder, which effectively encodes the KG. Subsequently, we utilize a scoring decoder to evaluate triplets. Experimental results demonstrate that our SE-KGC model outperforms other state-of-the-art methods in link-prediction tasks across three datasets. Specifically, compared to the baselines, SE-KGC achieved improvements of 11.7%, 1.05%, and 2.30% in terms of MRR on these three datasets. Furthermore, we present a comprehensive analysis of the contributions of different semantic patterns, and find that entities with higher connectivity play a pivotal role in effectively capturing and characterizing semantic information.

Keywords: knowledge graph completion; relational graph convolutional network; higher-order semantic pattern

MSC: 05C85

check for updates

Citation: Yuan, X.; Chen, J.; Wang, Y.; Chen, A.; Huang, Y.; Zhao, W.; Yu, S. Semantic-Enhanced Knowledge Graph Completion. *Mathematics* **2024**, *12*, 450. https://doi.org/10.3390/math12030450

Academic Editor: Pedro A. Castillo Valdivieso

Received: 8 December 2023
Revised: 19 January 2024
Accepted: 24 January 2024
Published: 31 January 2024

1. Introduction

Knowledge graphs (KGs) are instrumental in various intelligence applications, including recommendation systems [1,2], information retrieval [3,4], and question answering [5,6]. To facilitate these knowledge-driven applications, numerous types of KGs have been developed in the past decades. KGs comprise structured human-understandable knowledge, typically represented as triplets [7]. Each triplet, denoted as (e_h, r, e_t), comprises a head entity e_h, a tail entity e_t, and a relation r. By leveraging this structured representation, KGs can be represented as multi-relation graphs, with entities acting as nodes and relations serving as edges [8]. This graphical representation facilitates the exploration and analysis of interconnections and semantic relationships within the knowledge domain, enabling sophisticated knowledge-driven applications to leverage the rich information network within KGs [9].

KGs are usually constructed from various data sources (e.g., structured databases or text corpora), through methods such as relation extraction [10], or by manually checking facts by domain experts [11,12]. Considering the quality of existing facts and the need for newly added knowledge, KGs are always incomplete and sparse, which limits their utility in various applications [13–15]. To address this challenge, knowledge graph embedding (KGE) models have emerged as a powerful solution [16–19]. These models aim to learn low-dimensional vector representations or embeddings of relations and entities within KGs. By utilizing these embeddings, KGE models can effectively predict missing links in KGs, thereby enabling the completion of the graph [20].

KGs consist of human-understanding knowledge organized by incomplete graphs, hence containing undiscovered relations between entities [21]. Due to the incompleteness and sparsity of KGs, most KGE models learn representation with insufficient information [22–24]. Precisely, these methods complete KGs mainly based on observable correlations between entities and overlook potential semantic correlations. Figure 1 illustrates that even unconnected entities may have probable relationships. Furthermore, interactions within networks are not limited to pairs, but rather occur in larger groups [25]. Atom-groups in a molecule represent specific chemical functionalities and user-groups in a social network serve particular sociological purposes [26]. Similarly, triplets in KGs only depict observable entity-pairs, but unobservable correlations reside in entity-groups (e.g., triangles). Entity-groups in KGs serve as semantic patterns, and slight structural variations (e.g., absence of a triplet) within these patterns intricately shape distinct characteristics. Although many KGE models have achieved significant success, capturing these unobserved relations remains challenging. Graph convolutional networks (GCNs) offer powerful graph-learning capabilities and can be employed to extract structural characteristics beyond pairwise relations [27–30], GCNs are also effective in various domains (e.g., traffic flow prediction [31]). R-GCN [32] introduced relation-specific transformations to incorporate relational information during the aggregation of neighbors. Shang et al. considered the influence of relation weights and proposed SACN [33]. Vashishth et al. designed CompGCN to improve relation modeling by jointly embedding entities and relations. HRAN [34] use heterogeneous relation–attention networks to improve relation modeling and prediction in complex and heterogeneous KGs. All of these demonstrate the effectiveness of GCNs in capturing higher-order interactions and dependencies between entities, which also provide valuable insights for relation modeling and prediction tasks. In the context of message passing scheme of GCNs, they effectively propagate semantic information within KGs. Moreover, GCNs perform well in relation modeling and link-prediction tasks, showcasing their capabilities of handling complex relational data. However, GCNs encounter difficulties in aggregating limited neighborhood information within sparse KGs. Consequently, GCN-based KGE models also face challenges in capturing potential semantic correlations.

To address this problem, we propose a KGE model, namely semantic-enhanced knowledge graphcompletion (SE-KGC). The core of SE-KGC lies in a semantic enhancement module, capturing potential correlations through semantic patterns. Considering these patterns exhibit diverse behaviors across different domains, we predefine them by selecting several higher-order structures (i.e., motifs) and then adapting the attention mechanism. Accordingly, the enhancement module operates like sliding windows on KGs, empowering information propagation on underlying relations by mimicking convolution operations. We then apply a multi-relational GCN encoder to learn powerful representations in an enriched local neighborhood. Additionally, we utilize a scoring decoder to evaluate triplets. In summary, the main contributions of this paper are as follows:

- We propose SE-KGC, a knowledge graph embedding model that captures potential semantic correlations between entities.
- We develop an entity feature enhancement module that incorporates both semantic and structural information for inherently sparse KGs, which adaptively enriches local neighborhoods for the GCN encoder.
- We conduct experiments on several real-world KGs from different domains, demonstrating the effectiveness of SE-KGC. We thoroughly analyze the learned weights of different semantic patterns and find that those with higher connectivity are more vital.

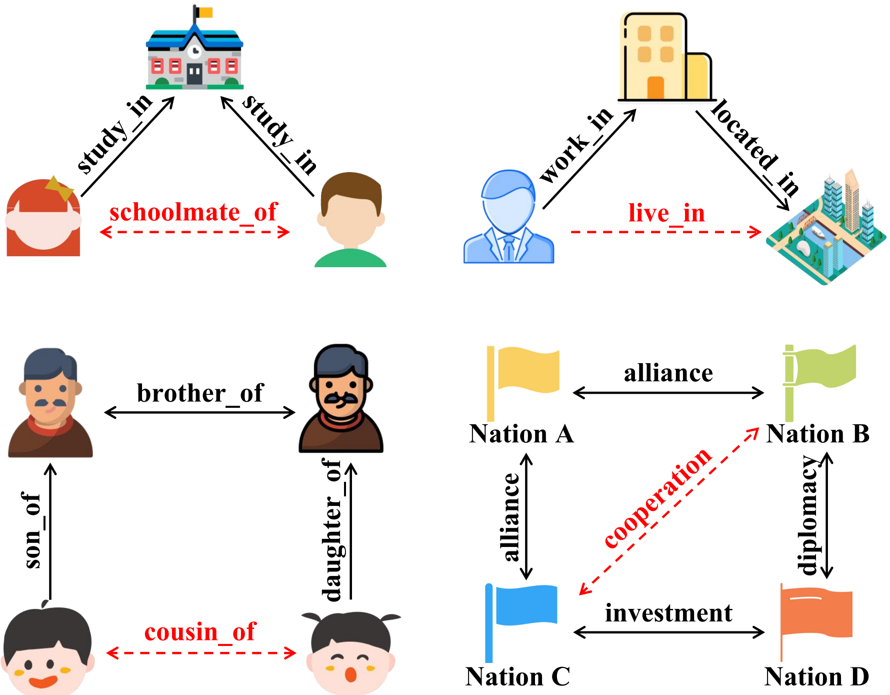


Figure 1. An example of semantic patterns. Black solid lines are observable relations and red dashed lines are potential semantic correlations.

The subsequent sections of this paper are organized as follows. Section 2 deals with the previous relevant research related to our study. Section 3 introduces some preliminaries. Section 4 elaborates on the intricate details of our proposed model, highlighting its key components and functionalities. Section 5 offers a comprehensive exposition of our experimental design, including an in-depth introduction to the experiments conducted and the corresponding results obtained. Finally, Section 6 summarises the contributions, emphasizing its significance and potential implications.

2. Related Work

2.1. Knowledge Graph Embedding Methods

KGE models aim to learn low-dimensional embeddings for relations and entities and subsequently predict the validity of given triplets. Translational distance models (e.g., TransE [22]) measure the distance between two entities based on their relationship in the embedding space. However, these models are limited in their ability to handle complex relations (e.g., symmetric relations and asymmetric relations). Bilinear matching models (e.g., DISTMULT [23] and ComplEx [24]) satisfy various relations through powerful matrix multiplication, which is always time-consuming. Convolutional neural network (CNN)-based methods (e.g., ConvE [35], ConvR [36], and ConvKB [37]) adopt convolution operations to perform multiplication on local regions of embedding matrices. While these methods are scalable, they face challenges in capturing global relational patterns and may struggle to effectively represent long-tail relations.

With the success of graph convolutional networks (GCNs) in handling graph-structured data, numerous GCN-based models have been proposed for learning representations of knowledge graphs (KGs). R-GCN [32] introduces learnable weights for each relation, integrating them into the entity aggregation process. SACN [33] leverages the entity structure and relation types to encode entities and relations using a weighted GCN. CompGCN [38] leverages entity-relation composition operations and jointly learns representations for entities and relations. However, a common limitation of most GCN-based models is that they primarily rely on observable correlations between entities while disregarding unobservable correlations. Our work captures these potential correlations through predefined semantic patterns and enriches neighborhood information for the GCN encoder. This enables our model to capture both observable and unobservable relationships, leading to improved representation learning for KGs.

2.2. Structure Enhancement Methods

Most traditional KGE models mainly consider the triplet-form of KGs and overlook structural information. GCNs have the ability to learn structural characteristics during training by aggregation formulation. However, GCN cannot meet the need for tedious neighborhoods in inherently sparse KGs, leading to learn uncertain structural characteristics. To solve this problem, some GCN-based KGE models aim to integrate structural information. SCAN [33] takes this issue by uniquifying edges in each subgraphs through relation types, thereby improve the ability to identify structures. Similarly, KGEL [39] treats each entity as a subject and an object simultaneously and leverages a cluster for both roles. In comparison, DRGI [40] incorporates mutual information maximization to learn the structural distribution as complement information. Nevertheless, these three methods heavily rely on the learning capacity of neural networks to capture the structural characteristic, which introduces uncertainties in model explainability. In contrast, HRAN [34] explicitly aggregates information based on relation paths in KGs for GCN encoders. Relation paths are highly human-understandable, but the path structure is not complex enough to capture most structural information in KGs. Our methods adopt more complex structures (i.e., motifs) for aggregation, maintaining good explainability simultaneously.

3. Preliminaries

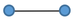
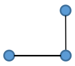
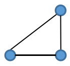
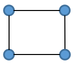
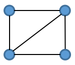
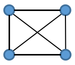
3.1. Problem Definition

Let $\mathcal{G} = (\mathcal{E}, \mathcal{R}, X, \mathcal{Z})$ denote a knowledge graph containing triplets (e_h, r, e_t). $\mathcal{E}$ is the set of entities and $\mathcal{R}$ is the set of relations such that $e_h, e_t \in \mathcal{E}$ and $r \in \mathcal{R}$. For a triplet (e_h, r, e_t), suppose $\mathbf{e_h}, \mathbf{r}, \mathbf{e_t} \in \mathbb{R}^d$ are the vector representations of e_h, r, e_t, respectively. We use $X \in \mathbb{R}^{|\mathcal{E}| \times d}$ to represent the d-dimensional initial features of all entities in $\mathcal{G}$. And $\mathcal{Z} \in \mathbb{R}^{|\mathcal{R}| \times d}$ denotes the initial relation features. Link prediction refers to the task of predicting missing head $(?, r, e_t)$ or tail $(e_h, r, ?)$ to infer new unobserved triplets for knowledge graph completion. For a given KG, the objective of the scoring function $\phi(e_h, r, e_t)$ is to look up the embedding of entities and relations upon the embedding matrices $\mathbf{E} \in \mathbb{R}^{|\mathcal{E}| \times d}$, $\mathbf{R} \in \mathbb{R}^{|\mathcal{R}| \times d}$, and then utilize them to score triplets. Generally, correct triplets receive higher truthfulness scores than incorrect ones.

3.2. Network Motif

A network motif refers to a specific higher-order structure that frequently appears in networks [26,41]. Motifs contain rich structural information and have been extensively studied in biological, social, and technological networks. It is worth noting that the number of selected motifs should encounter diverse domains for different KGs. However, higher-order motifs actually contain lower-order ones, leading to redundant expressiveness and increased computational complexity. Balancing the trade-off, our model uses six predefined motifs with orders of 2, 3, and 4 as semantic patterns for semantic enhancement. The specific definitions of these six motifs adopted in our model can be found in Table 1.

Table 1. The six predefined semantic patterns given by motifs.

Name	M2	M31	M32	M41	M42	M43
Id	1	2	3	4	5	6
Motif						

4. Method

This section introduces our proposed method SE-KGC, as shown in Figure 2. We begin by describing how we capture semantic correlations between entities using predefined semantic patterns. Subsequently, we outline the process of enhancing features with semantic information to facilitate representation learning. Next, we present the graph convolutional network employed for representation learning and the scoring function utilized for triplet evaluation. Finally, we provide comprehensive training details for our model.

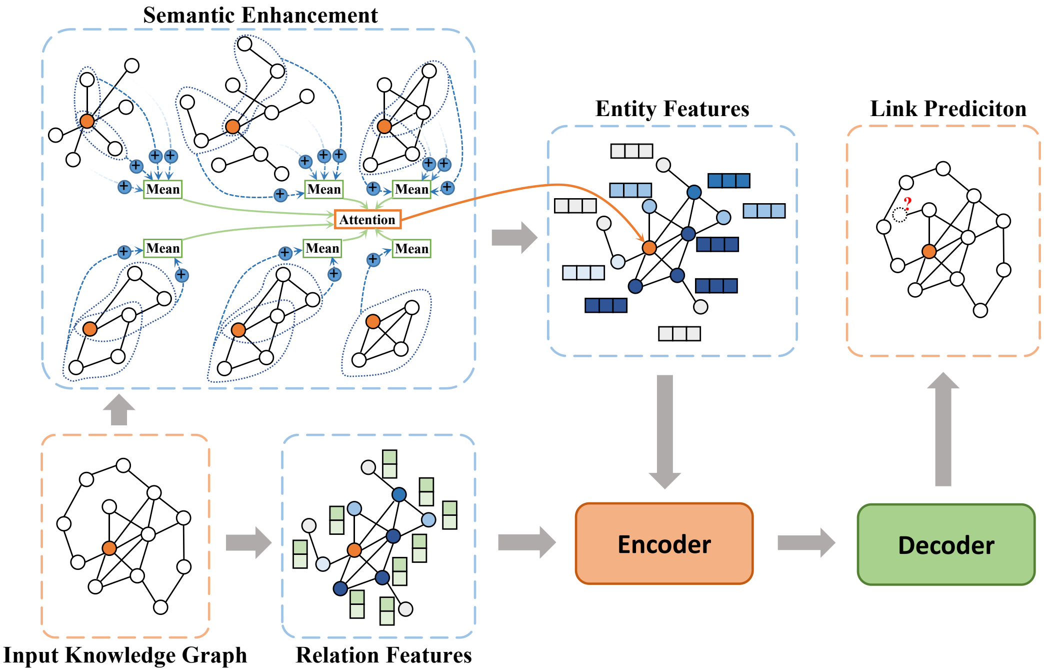


Figure 2. An overview of SE-KGC. First, the input knowledge graph is supplied to the model, where entities undergo semantic enhancement to extract refined features while relation features are obtained directly. Subsequently, these extracted features are then fed into the encoder for subsequent processing. Finally, the decoder effectively utilizes the encoded features for accurate link prediction.

4.1. Semantic Enhancement

A knowledge graph consists of structured human-understanding facts; thus, motifs frequently occurring in a knowledge graph naturally represent semantic patterns. Take the triangle motif $M32$ in Figure 1 as an example; it indicates that two entities linked by another entity also have a relation between them. Consequently, utilizing motifs to exploit semantic patterns within a knowledge graph holds great promise for capturing semantic correlation.

We regard motifs in Table 1 as six different semantic patterns, denoted as $M = \{M_1, M_2, \ldots, M_6\}$. A semantic pattern M_i contains substructures with the number of k in $\mathcal{G}$, denoted as $S_{M_i} = \{s_{M_{i,1}}, s_{M_{i,2}}, \ldots, s_{M_{i,k}}\}$. Each substructure comprises multiple entities, and the k-th substructure that follows M_i is represented as $s_{M_i,k} = \{s_{M_i,k,e_1}, s_{M_i,k,e_2}, s_{M_i,k,e_3}, \ldots\}$. We define substructure feature $\mathbf{s}_{M_i,k} \in \mathbb{R}^d$ of $s_{M_i,k}$ by summing the features of all the entities it contains:

$$\mathbf{s}_{M_i,k} = \sum_{e \in s_{M_i,k}} \mathbf{e}. \tag{1}$$

An entity e resides in multiple substructures with the number of $|h_{M_i,e}|$ under the semantic pattern M_i.

We define semantic feature $\mathbf{m}_{M_i,e} \in \mathbb{R}^d$ of entity e under pattern M_i by averaging the features of all substructures it lies in:

$$\mathbf{m}_{M_i,e} = \frac{1}{|h_{M_i,e}|} \sum_{s_{M_i,j} \in S_{M_i} \wedge e \in s_{M_i,j}} \mathbf{s}_{M_i,j}. \tag{2}$$

Concatenating semantic features for entity e under all semantic patterns M_i, we obtain $\mathbf{m}_{M,e} = Concat[\mathbf{m}_{M_1,e}, \mathbf{m}_{M_2,e}, \ldots, \mathbf{m}_{M_6,e}] \in \mathbb{R}^{6\times d}$. However, entities should exhibit varying sensitivities to distinct semantic patterns. To account for this diversity, our model adaptively learns the contribution of each pattern for all entities. Accordingly, we employ the attention mechanism, which allows us to calculate the enhanced feature $\mathbf{x}_{enh,e} \in \mathbb{R}^d$ for entity e:

$$\mathbf{x}_{enh,e} = Att(\mathbf{m}_{M,e}) = Softmax(f(\mathbf{m}_{M,e})) \cdot \mathbf{m}_{M,e}. \tag{3}$$

where $f(\cdot)$ represents a learnable linear transformation function. Finally, we obtain the final enhanced features $\mathbf{X}_{enh} \in \mathbb{R}^{|\mathcal{R}|\times d}$ of $\mathcal{G}$ from enhanced features of all entities in $\mathcal{R}$.

Our semantic enhancement closely resembles convolution operation on images. Substructures (e.g., triangles and rectangles) defined by motifs can be seen as analogies of sliding windows on KGs. The calculation of the substructure feature in Equation (1) and the semantic feature in Equation (2) can be regarded as convolution filter and mean pooling, respectively. We perform these two formulas with non-parameterized equal weights because KGs have no canonical entity ordering. Furthermore, the six semantic patterns play a similar role to multi-channels. The attention mechanism enables our model to learn the contributions of different patterns for different KGs automatically.

We can also conceptualize our semantic enhancement as the reconstruction of different neighbor subgraphs according to various semantic patterns. We expand the entity neighborhood or highlight influential entities for the target entity through these rebuilding subgraphs. As a result, our semantic enhancement enables entities to obtain refined information even if they are initially unconnected. Enriching local neighborhood information also encourages us to leverage strongly localized graph learning models (i.e., GCNs).

4.2. Graph Encoder

The semantic enhancement allows us to capture potential correlations between entities directly. Incorporating this expanded information with original information through representation learning is intuitive. To this end, we employ a graph neural network designed for multi-relational graphs as the graph encoder in our model.

For input of the encoder, we combine the enhanced features X_{enh} with the initial features X through three optional operations: Attention, Replace, and Concat(Concatenate).

$$X' = Att(X_{enh}, X), \tag{4}$$

$$X' = X_{enh}, \tag{5}$$

$$X' = X_{enh}||X. \tag{6}$$

where $Att(\cdot)$ and $|\cdot|$ represent Attention and Concat operations, respectively. To enable KGs to learn entity and relation representations simultaneously, we utilize an entity-relation composition operation:

$$\phi(x'_u, z_r) = \mathbf{x_u}' * \mathbf{z_r}, \tag{7}$$

where $*$ is multiplication operator [23], and $\mathbf{x}'_u$ and $\mathbf{z}_r$ denote features of entity u and relation r, respectively. The operation redefines the messages passed in GNN by combining entity and relation information. Then, the entity update equation of our GCN encoder is given as:

$$h_e = f\Big(\sum_{(u,r)\in\mathcal{N}(e)} W_e \phi(x'_u, z_r)\Big). \tag{8}$$

where $\mathcal{N}(e)$ is a set of immediate neighbors of e for its outgoing edges, and $W_e \in \mathbb{R}^{d_w\times d}$ is a parameter matrix given by adding inverse relations and self-loops. Similarly, the relation update equation of our GCN encoder is given as follows:

$$h_r = W_r z_r, \tag{9}$$

where $W_r \in \mathbb{R}^{d_w\times d}$ is a learnable parameter matrix across all relations.

Since GCNs are always multi-layer structures, we expand Equations (8) and (9) as:

$$h_e^{k+1} = f\left(\sum_{(u,r)\in\mathcal{N}(e)} W_e^k \phi(h_e^k, h_r^k)\right), \tag{10}$$

$$h_r^{k+1} = W_r^k h_r^k. \tag{11}$$

The entity-relation composition operation $\phi(\cdot)$ in our encoder is non-parameterized. Usually, untrainable operations may not offer significant insights for redefining messages in a unique entity-relation fashion. It is worth noting that semantic enhancement in our model bridges an inner correlation between entities and relations. Hence, it is effective to incorporate simple composition operations with our semantic enhancement.

4.3. Graph Decoder

In this paper, we adopt ConvE [35] as the decoder to evaluate triplets. ConvE models the interaction between input entities and relations through convolutional and fully connected layers. Taking a triplet (e_h, r, e_t) as input, ConvE reshapes entity embeddings and applies convolution to obtain a score:

$$p_{(e_h,r,e_t)} = ReLU(vec(ReLU([\mathbf{e}_h; \mathbf{r}] * \omega))\mathbf{W})\mathbf{e}_t. \tag{12}$$

The operator $*$ denotes element-wise multiplication, and ω denotes the filter of a 2D convolutional layer. We use $\mathbf{W} \in \mathbb{R}^{1\times d}$ to represent trainable parameters, and $vec(\cdot)$ represents the flatten operation. These scores reflect the similarity or association between the tail entity and the given head entity and relation.

4.4. Optimization Target

We jointly train our encoder-decoder model. For a triplet (e_h, r, e_t), we obtain its score $p_{(e_h,r,e_t)}$ outputted by graph decoder to predict its truthfulness:

$$\hat{y} = sigmoid(p_{(e_h,r,e_t)}). \tag{13}$$

We train the model by minimizing binary cross-entropy (BCE) loss:

$$\mathcal{L} = -\frac{1}{N}\sum^{N}(y \cdot log(\hat{y}) + (1-y) \cdot log(1-\hat{y})). \tag{14}$$

where N represents the number of candidate tail entities, and label y is set as either 1 or 0 to represent whether a triplet is true. The training procedure of SE-KGC is illustrated in Algorithm 1.

Algorithm 1 SE-KGC

```
Input: $\mathcal{G} = (\mathcal{E}, \mathcal{R}, X, \mathcal{Z})$; Sum function $Sum(\cdot)$; mean function $Mean(\cdot)$; attention function
    $Att(\cdot)$; update function $f(\cdot)$; sigmoid function $sigmoid(\cdot)$; scoring function $p_{(e_h,r,e_t)}$.
Output: entity embedding $\mathbf{e} = h_e^L$; relation embedding $\mathbf{r} = h_r^L$.
for each substructure in the semantic pattern do
    $\mathbf{s}_{M_i,k} = Sum(\mathbf{e})$
end for
for each semantic pattern in the KG do
    $\mathbf{m}_{M_i,e} = Mean(\mathbf{s}_{M_i,j})$
end for
$\mathbf{m}_{M,e} = Concat[\mathbf{m}_{M_1,e}, \mathbf{m}_{M_2,e}, \ldots, \mathbf{m}_{M_6,e}]$
$\mathbf{x}_{enh,e} = Att(\mathbf{m}_{M,e})$
$X' = Att(X_{enh}, X)$, $X' = X_{enh}$ or $X' = X_{enh}||X$
$\phi(x'_u, z_r) = \mathbf{x_u}' * \mathbf{z_r}$
```

Algorithm 1 *Cont.*

```
for epoch=1 $\to$ n do
    for $\forall e \in \mathcal{E}$ do and $\forall r \in \mathcal{R}$
        for k=1 $\to$ L do
            $h_e^{k+1} = f(\sum_{(u,r)\in\mathcal{N}(e)} W_e^k \phi(h_e^k, h_r^k))$
            $h_r^{k+1} = W_r^k h_r^k$
        end for
    end for
    $\mathcal{L} = -\frac{1}{N}\sum^N (y \cdot log(sigmoid(p_{(e_h,r,e_t)})) + (1-y) \cdot log(1 - sigmoid(p_{(e_h,r,e_t)})))$
end for
```

5. Experiments

We evaluated SE-KGC against multiple baseline methods in the link-prediction task. Furthermore, we performed an extensive analysis of different semantic patterns to gain deeper insights into their effectiveness and contributions.

5.1. Datasets and Baselines

5.1.1. Datasets

To validate the effectiveness of our approach, extensive experiments are conducted on the following benchmark datasets: WN18RR [35], Kinship [42], and Nations [43]. Each dataset was divided into the train, valid, and test sets. The statistical information for these datasets is presented in Table 2, and additional detailed information is provided as follows.

- WN18RR [35]: WN18RR is a subset of WN18, which is built by removing reversible relations from WN18 dataset. It contains 40,559 entities and 11 relations.
- Kinship [42]: Aboriginal Australian kinship systems are important in traditional aboriginal cultures as customary laws for social interactions among kin. The Alyawarre system from Central Australia has 104 entities and 25 relations, particularly relevant to marriages between aboriginal people.
- Nations [43]: Nations is a small knowledge graph and focuses on countries and their political relations, providing valuable insights into international affairs and diplomatic connections.

Table 2. Statistics of datasets.

Dataset	WN18RR	Kinship	Nations
#Entities	40,599	104	14
#Relations	11	25	55
#Train set	86,835	8544	1592
#Valid set	2924	1068	199
#Test set	2824	1074	201

# represents the quantity.

5.1.2. Baselines

We evaluate our model with three different combination operations in Equations (4)–(6), denoted as SE-KGC-attention, SE-KGC-replace, and SE-KGC-concat, respectively. We then compare their performance with several methods developed in recent years.

- Translational distance models: TransE [22], RotatE [44].
- Bilinear matching models: DISTMULT [23], ComplEx [24].
- CNN-based models: ConvE [35], ConvR [36], ConvKB [37], LTE-ConvE [45].
- GCN-based models: R-GCN [32], SACN [33], CompGCN [38], HRAN [34], KGEL [39], DRGI [40].

By comparing with these existing models, we aim to assess the effectiveness and competitiveness of our proposed approach in knowledge graph representation and link-prediction tasks.

5.2. Experimental Setup

5.2.1. Implementation Details

Each model is trained for 500 epochs using an ADAM optimizerwith a learning rate of 0.001. ADAM is a suitable choice for training knowledge graph models due to its ability to handle sparse gradients, scalability, adaptive learning rate, and faster convergence. To prevent overfitting, the training process ends when the value of the loss function no longer decreases. In order to accommodate the dataset size, strike a balance between efficiency and performance, and mitigate the risk of overfitting, we opt for a single GCN layer with a hidden state dimension of 200. Additionally, aligning the hidden state dimension with the dimensions of the entity and relation embeddings contributes to the seamless integration of information across various model components. The same hyperparameters are used to maintain consistency across different datasets. All experiments are conducted on Intel(R) Xeon(R) Gold 5120 CPU @ 2.20 GHz (Intel Corporation, Santa Clara, United States) and Tesla V100-SXM2-32 GB GPU (NVIDIA Corporation, Santa Clara, United States), which can accelerate the computational process and improve the speed of model training and inference.

5.2.2. Metrics

Consistent with most baseline methods, we evaluate the performance of our model using the ranks of triplets. Specifically, we conduct both head and tail evaluations for all correct triplets. Taking the tail evaluation of a triplet as an example, we first construct its correct triplet by replacing its head and tail entities with other entities. Subsequently, the proposed model predicts the scores of correct and corrupted triples using the graph decoder (ConvE). These scores are then sorted in descending order to determine the corresponding ranks. Based on these ranks, we assess the model's performance using several rank-based metrics, including Mean Rank (MR), Mean Reciprocal Rank (MRR), and Hits@n, which represents the proportion of correct entities ranking in the top n (where n = 1, 3, 10).

- $MR = \frac{1}{|S|}\sum_{i=1}^{|S|} rank_i$
  $= \frac{1}{|S|}(rank_1 + rank_2 + \cdots + rank_{|S|})$
- $MRR = \frac{1}{|S|}\sum_{i=1}^{|S|} \frac{1}{rank_i}$
  $= \frac{1}{|S|}(\frac{1}{rank_1} + \frac{1}{rank_2} + \cdots + \frac{1}{rank_{|S|}})$
- $Hits@n = \frac{1}{|S|}\sum_{i=1}^{|S|} \mathrm{II}(rank_i) \leq n$

Given the set of triplets S, $rank_i$ indicates the predicted rank of the ith triplet. The indicator function $\mathrm{II}(\cdot)$ is utilized to determine if a rank is the correct one. The MR metric quantifies the average rank of the model within the ranking for each test example. On the other hand, the MRR metric captures the speed at which the model identifies the correct answer within the ranking. Meanwhile, the Hits@n metric assesses whether the model includes the correct triples within its top n predictions. A higher value of MRR and Hits@n indicates better model performance, while a higher MR value indicates poorer performance. Again, the procedure for head evaluation is the same as for tails. The final evaluation result is obtained by averaging the results of both head and tail evaluations. In the realm of real-world KGs, relations between entities typically exhibit bidirectional characteristics, while the identities of missing entities remain unknown (head or tail). To facilitate the application of KGC in link-prediction tasks, a model must possess the capability to concurrently predict both head and tail entities. As an example, consider a KG comprising entities A, B, and C. Given the correct triplet (A, Relation1, B), head evaluation involves fixing Relation1 and entity B to predict the missing head entity A. If the model accurately predicts entity A, it can be concluded that the model exhibits a good understanding of Relation1 and the patterns and dependencies between entities A and B. The same principle applies to tail evaluation. By simultaneously considering both head and tail predictions, the model gains a more comprehensive understanding of the interrelationships among entities, thereby enhancing its ability to capture entity interconnectedness, leading to improved accuracy in predicting missing entities.

5.3. Link Prediction

The results from experiments are shown in Tables 3–5. SE-KGC outperforms all other methods on the three datasets. Notably, for the WN18RR dataset, SE-KGC exhibits significant performance improvement compared to SACN and CompGCN, with MRR increased by 11.3% and 11.7%, respectively, while Hits@10 increased by 17.9% and 19.2%, respectively. Such effectiveness is obtained due to the semantic enhancement step. In addition, the Hits@1 and MRR metrics of SE-KGC-concat are better than those of SE-KGC-replace. It is worth noting that SE-KGC-replace does not use the original feature, which still contributes to the overall performance. Comparing SE-KGC to the single decoder model ConvE, SE-KGC demonstrates significant improvements across all metrics, confirming the effectiveness of our encoder model and semantic enhancement approach. The above results show that the proposed SE-KGC can generate representation embedding for entities and relations and can be used in link-prediction tasks. It can also improve the effectiveness of the method through semantic enhancement. The results show that it is necessary to pay attention to the potential semantic correlations between entities.

Table 3. Performance of link-prediction tasks evaluated on the WN18RR dataset.

Methods	MRR	MR	Hits@10	Hits@3	Hits@1
TransE [22]	0.226	3384	0.501	-	-
RotatE [44]	0.476	3340	0.571	0.492	0.428
DISTMULT [23]	0.430	5510	0.490	0.440	0.390
ComplEx [24]	0.440	5261	0.510	0.460	0.410
ConvE [35]	0.430	4187	0.520	0.440	0.400
ConvR [36]	0.475	-	0.537	0.489	0.443
ConvKB [37]	0.249	3324	0.524	0.417	0.057
LTE-ConvE [45]	0.472	3434	0.544	0.485	0.437
R-GCN [32]	-	-	-	-	-
SACN [33]	0.470	-	0.540	0.480	0.430
CompGCN [38]	0.479	3533	0.546	0.494	0.443
HRAN [34]	0.479	2113	0.542	0.494	0.450
KGEL [39]	0.476	-	0.547	0.467	0.446
DRGI [40]	0.479	3223	0.543	0.496	0.445
SE-KGC-attention	0.486	2443	0.570	0.500	0.446
SE-KGC-replace	0.523	**1330**	**0.644**	**0.560**	0.456
SE-KGC-concat	**0.535**	1526	0.626	0.555	**0.487**

Bold values indicate optimal quantities.

Table 4. Performance of link-prediction tasks evaluated on the Kinship dataset.

Methods	MRR	MR	Hits@10	Hits@3	Hits@1
TransE [22]	0.309	6.800	0.841	0.643	0.009
RotatE [44]	0.738	2.900	0.954	0.827	0.617
DISTMULT [23]	0.516	5.260	0.867	0.581	0.367
ComplEx [24]	0.823	2.480	0.971	0.899	0.733
ConvE [35]	0.830	2.000	0.980	0.910	0.730
ConvR [36]	-	-	-	-	-
ConvKB [37]	0.614	3.300	0.953	0.755	0.436
LTE-ConvE [45]	0.840	1.997	0.982	0.919	0.752
R-GCN [32]	0.109	25.920	0.239	0.088	0.030
SACN [33]	0.799	2.500	0.964	0.878	0.699
CompGCN [38]	0.850	1.951	0.981	0.920	0.769
HRAN [34]	-	-	-	-	-
KGEL [39]	0.844	-	0.983	0.919	0.764
DRGI [40]	0.847	1.900	0.981	0.915	0.765
SE-KGC-attention	**0.859**	1.942	0.981	**0.926**	**0.781**
SE-KGC-replace	0.844	**1.863**	**0.987**	0.918	0.758
SE-KGC-concat	0.849	1.968	0.981	0.919	0.766

Bold values indicate optimal quantities.

Table 5. Performance of link-prediction tasks evaluated on the Nations dataset.

Methods	MRR	MR	Hits@10	Hits@3	Hits@1
TransE [22]	0.422	3.114	0.988	0.749	0.050
RotatE [44]	0.667	2.366	0.998	0.799	0.502
DISTMULT [23]	0.692	2.480	0.988	0.794	0.555
ComplEx [24]	0.511	3.425	0.975	0.617	0.313
ConvE [35]	0.820	2.000	1.000	0.880	0.720
ConvR [36]	-	-	-	-	-
ConvKB [37]	-	-	-	-	-
LTE-ConvE [45]	0.742	2.112	0.992	0.831	0.622
R-GCN [32]	0.787	1.843	0.998	0.866	0.674
SACN [33]	-	-	-	-	-
CompGCN [38]	0.796	1.796	1.000	0.866	0.689
HRAN [34]	-	-	-	-	-
KGEL [39]	-	-	-	-	-
DRGI [40]	-	-	-	-	-
SE-KGC-attention	**0.839**	**1.562**	**1.000**	**0.928**	**0.740**
SE-KGC-replace	0.799	1.861	0.993	0.878	0.692
SE-KGC-concat	0.833	1.627	**1.000**	0.901	0.739

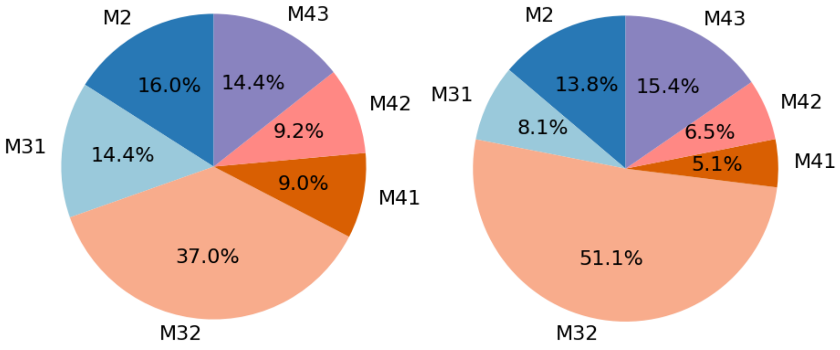
Bold values indicate optimal quantities.

This paper also addresses the performance of SE-KGC on two small datasets (Kinship, Nations). For Kinship, SE-KGC showed a 1.6% improvement in the Hits@1 metrics compared to CompGCN. For Nations, SE-KGC showed a greater improvement in the Hits@3 and Hits@1 metrics compared to CompGCN, with 7% and 7.9%, respectively. Our model has limited improvement in performance due to the small size of these two datasets. The MRR index of SE-KGC-attention is 24.4% higher than that of ConvE, which confirms the effectiveness of our encoder model and shows that the neighborhood information and semantic enhancement of SE-KGC-attention aggregation are valuable.

The best results for WN18RR were observed with the replace and concat optional operations, as they effectively preserved the original enhanced features and initial features. Compared with WN18RR, Kinship and Nations exhibit better performance on SE-KGC-attention rather than SE-KGC-replace. We contend the reason is that SE-KGC-attention easily fits trainable attention weights for small datasets.

5.4. Analysis of Semantic Patterns

As shown in Figure 3, both SE-KGC-replace and SE-KGC-concat exhibit a similar distribution of the learned weights of the six motifs in the WN18RR dataset. Motifs $M32$ and $M43$ occupy the top two highest proportions in two SE-KGC variants, which can be explained by semantic patterns given by structures of motifs. Structurally, motifs $M32$ and $M43$ connect every entity to all other entities in the graph. Semantic patterns behind these two structures imply that entities with the same neighbors probably also have relations. This observation highlights the semantic patterns ingrained within the structures of these motifs, reinforcing the importance of considering such patterns in link-prediction tasks.



(**a**) SE-KGC-replace (**b**) SE-KGC-concat

Figure 3. Proportion distribution of different semantic patterns in the WN18RR dataset.

We can also obtain the distribution of the SE-KGC-attention model from Figure 4. For third-order motifs, $M32$ holds greater significance compared to $M31$. Similarly, in the case of fourth-order motifs, $M43$ exhibits higher importance than $M42$, and likewise, $M42$ surpasses $M41$ in significance. This observation can be attributed to the semantic patterns determined by the motifs. The patterns can be explained by the fact that entities with higher connectivity tend to play a more crucial role in characterizing semantic information. We show the effectiveness of semantic enhancement guided by motifs, and different motif structures stand for various semantic patterns.

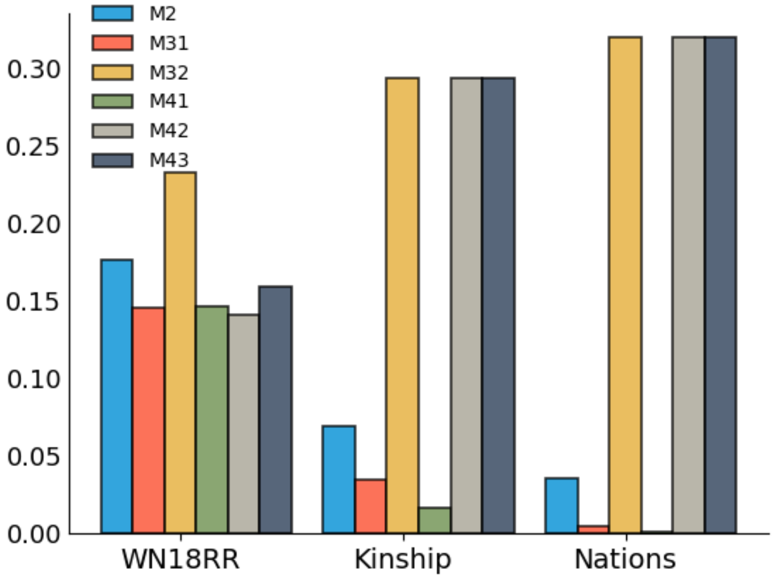


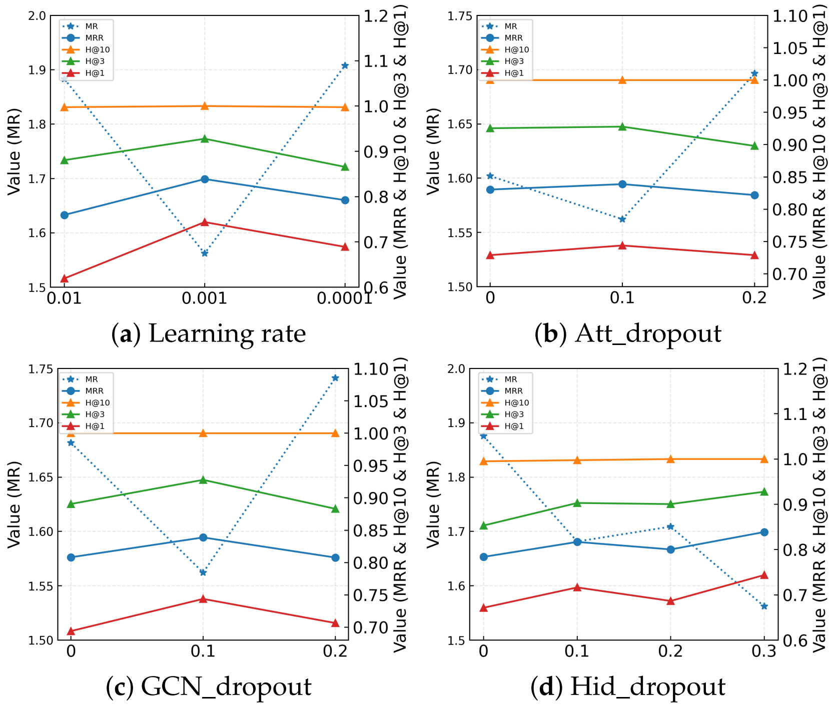
Figure 4. Proportion distribution of different semantic patterns in SE-KGC-attention.

5.5. Parameter Sensitivity

We conducted extensive experiments on the Nations dataset to investigate the sensitivity of four important parameters, including the impact of the learning rate, the settings of attention dropout, hidden dropout and GCN dropout. By systematically varying these parameters and analyzing their effects on the experimental outcomes, our objective was to gain profound insights into their influence and identify the optimal configurations for our model.

5.5.1. The Value of Learning Rate

The learning rate is a crucial hyperparameter in machine learning that plays a pivotal role in determining the step size for updating model parameters during training. It directly influences the speed at which the model converges when minimizing the loss function using gradient descent. Setting the appropriate learning rate is essential for improving the accuracy and convergence speed of the model. We conducted experiments with learning rates set to 0.01, 0.001, and 0.0001 to ascertain the most suitable learning rate. The experimental results are shown in Figure 5a. Notably, our model achieved the best results when the learning rate was set to 0.001. As observed, when the learning rate is too small, the model tends to converge at a sluggish pace. Conversely, when using a large learning rate, the model may either fail to converge or converge to a suboptimal solution.



(**a**) Learning rate

(**b**) Att_dropout

(**c**) GCN_dropout

(**d**) Hid_dropout

Figure 5. Parameter sensitivity.

5.5.2. The Settings of Dropout

Attention dropout, GCN dropout, and hidden dropout are regularization techniques widely employed in deep learning to mitigate the problem of overfitting. These techniques play a crucial role in improving the generalization capability of models. Attention dropout sets a proportion of attention weights to zero during training to reduce the model's reliance on specific attention areas and improve generalization. Hidden dropout randomly drops out neurons during training. This dropout scheme reduces the model's dependence on specific features and promotes robustness by encouraging the remaining neurons to collectively contribute to the learning process. GCN dropout operates on the adjacency matrix of graph data during training to reduce dependence on certain nodes or edges and improve generalization. This regularization technique enhances the generalization capability of graph convolutional networks by encouraging the model to consider a wider range of node and edge relationships. These techniques are important for preventing overfitting and improving generalization in deep learning models. The experimental results, depicted in Figure 5b–d, confirm the effectiveness of these techniques. Notably, the optimal experimental results are obtained when att_dropout is and set to 0.1, GCN_dropout is set to 0.1, and hid_dropout is set to 0.2.

6. Conclusions

We have proposed a KGE model, namely SE-KGC, that can capture various semantic information within knowledge graphs. The key idea is to present a semantic enhancement under different semantic patterns. These patterns are predefined by a group of typical higher-order structures, incorporating human-understanding knowledge. The semantic enhancement not only captures potential semantic correlations between entities but also enriches the local neighborhood for our GCN encoder. We apply a decoder to evaluate triplet scores for link-prediction tasks. Experimental results demonstrate that our SE-KGC is superior to other models on three datasets, and we also comprehensively analyze the contributions of different semantic patterns.

Capturing semantic information in KGs is crucial for advancing various information-retrieval applications. Although large language models have achieved significant success in understanding semantemes, their underlying mechanisms remain opaque and difficult to interpret. Exploring human-understanding higher-order semantic patterns holds significant value for model transparency. In this paper, we attempted to formulate these semantic patterns for knowledge graph completion and achieved significant improvements. The follow-up research could pay more attention to higher-order complex semantic information, which will definitely prompt practical applications in various fields such as mining graphs and data science.

Author Contributions: Conceptualization, X.Y., J.C. and W.Z.; investigation, J.C. and S.Y.; methodology, J.C., Y.W. and S.Y.; supervision, X.Y., W.Z. and A.C.; validation, X.Y., J.C. and W.Z.; writing—original draft, X.Y., J.C., A.C. and Y.W.; writing—review and editing, X.Y., W.Z., Y.H. and S.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is partially supported by Bintuan Science and Technology Program No. 2021AA006, the Fundamental Research Funds for the Central Universities under Grant No. DUT23LAB101, and the "High-level Talent Team" Project of Dalian Science and Technology Talent Innovation Support Policy Program under Project No. 2022RG11.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The three datasets used in this paper are publicly available.

Acknowledgments: The authors would like to thank Ahsan Shehzad and Huafei Huang for their help.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional neural network
GCN	Graph convolutional networks
KG	Knowledge graph
KGE	Knowledge graph embedding

References

1. Wang, J.; Shi, Y.; Yu, H.; Wang, X.; Yan, Z.; Kong, F. Mixed-Curvature Manifolds Interaction Learning for Knowledge Graph-aware Recommendation. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei, Taiwan, 23–27 July 2023; pp. 372–382.
2. Qin, Y.; Gao, C.; Wei, S.; Wang, Y.; Jin, D.; Yuan, J.; Zhang, L.; Li, D.; Hao, J.; Li, Y. Learning from Hierarchical Structure of Knowledge Graph for Recommendation. *ACM Trans. Inf. Syst.* **2024**, *42*, 1–24. [CrossRef]
3. Wu, T.; Bai, X.; Guo, W.; Liu, W.; Li, S.; Yang, Y. Modeling Fine-grained Information via Knowledge-aware Hierarchical Graph for Zero-shot Entity Retrieval. In Proceedings of the 16th ACM International Conference on Web Search and Data Mining, Singapore, 27 February–3 March 2023; pp. 1021–1029.
4. Zhou, Y.; Chen, X.; He, B.; Ye, Z.; Sun, L. Re-thinking Knowledge Graph Completion Evaluation from an Information Retrieval Perspective. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, 11–15 July 2022; pp. 916–926.
5. Liu, L.; Chen, Y.; Das, M.; Yang, H.; Tong, H. Knowledge Graph Question Answering with Ambiguous Query. In Proceedings of the ACM Web Conference 2023, Austin, TX, USA, 30 April–4 May 2023; pp. 2477–2486.
6. Chen, Z.; Liao, J.; Zhao, X. Multi-granularity Temporal Question Answering over Knowledge Graphs. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; pp. 11378–11392.
7. Zhang, Y.; Qian, Y.; Ye, Y.; Zhang, C. Adapting Distilled Knowledge for Few-shot Relation Reasoning over Knowledge Graphs. In Proceedings of the 2022 SIAM International Conference on Data Mining, Alexandria, VA, USA, 28–30 April 2022; pp. 666–674.
8. Qi, Z.; Wang, H.; Zhang, H. A Dual-Store Structure for Knowledge Graphs. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 1104–1118. [CrossRef]
9. Wang, J.; Wang, B.; Gao, J.; Hu, S.; Hu, Y.; Yin, B. Multi-Level Interaction Based Knowledge Graph Completion. *IEEE ACM Trans. Audio Speech Lang. Process.* **2024**, *32*, 386–396. [CrossRef]
10. Zhao, X.; Yang, M.; Qu, Q.; Xu, R.; Li, J. Exploring Privileged Features for Relation Extraction with Contrastive Student-Teacher Learning. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 7953–7965. [CrossRef]

11. Deng, S.; Wang, C.; Li, Z.; Zhang, N.; Dai, Z.; Chen, H.; Xiong, F.; Yan, M.; Chen, Q.; Chen, M.; et al. Construction and Applications of Billion-Scale Pre-Trained Multimodal Business Knowledge Graph. In Proceedings of the 39th IEEE International Conference on Data Engineering, Anaheim, CA, USA, 3–7 April 2023; pp. 2988–3002.
12. Zhou, X.; Zheng, X.; Cui, X.; Shi, J.; Liang, W.; Yan, Z.; Yang, L.T.; Shimizu, S.; Wang, K.I. Digital Twin Enhanced Federated Reinforcement Learning with Lightweight Knowledge Distillation in Mobile Networks. *IEEE J. Sel. Areas Commun.* **2023**, *41*, 3191–3211. [CrossRef]
13. Nandi, A.; Kaur, N.; Singla, P.; Mausam. Simple Augmentations of Logical Rules for Neuro-Symbolic Knowledge Graph Completion. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; pp. 256–269.
14. Li, J.; Shomer, H.; Ding, J.; Wang, Y.; Ma, Y.; Shah, N.; Tang, J.; Yin, D. Are Message Passing Neural Networks Really Helpful for Knowledge Graph Completion? In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, Toronto, ON, Canada, 9–14 July 2023; pp. 10696–10711.
15. Geng, Y.; Chen, J.; Pan, J.Z.; Chen, M.; Jiang, S.; Zhang, W.; Chen, H. Relational Message Passing for Fully Inductive Knowledge Graph Completion. In Proceedings of the 39th IEEE International Conference on Data Engineering, Anaheim, CA, USA, 3–7 April 2023; pp. 1221–1233.
16. Lee, J.; Chung, C.; Whang, J.J. InGram: Inductive Knowledge Graph Embedding via Relation Graphs. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; Volume 202, pp. 18796–18809.
17. Zeng, W.; Zhao, X.; Tan, Z.; Tang, J.; Cheng, X. Matching Knowledge Graphs in Entity Embedding Spaces: An Experimental Study. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 12770–12784. [CrossRef]
18. Zhu, X.; Li, G.; Hu, W. Heterogeneous Federated Knowledge Graph Embedding Learning and Unlearning. In Proceedings of the ACM Web Conference (WWW 2023), Austin, TX, USA, 30 April–4 May 2023; pp. 2444–2454.
19. Liang, S. Knowledge Graph Embedding Based on Graph Neural Network. In Proceedings of the 39th IEEE International Conference on Data Engineering, Anaheim, CA, USA, 3–7 April 2023; pp. 3908–3912.
20. Shen, Y.; Ding, N.; Zheng, H.; Li, Y.; Yang, M. Modeling Relation Paths for Knowledge Graph Completion. *IEEE Trans. Knowl. Data Eng.* **2021**, *33*, 3607–3617. [CrossRef]
21. Niu, G.; Li, B. Logic and Commonsense-Guided Temporal Knowledge Graph Completion. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023; pp. 4569–4577.
22. Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the 27th Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; pp. 2787–2795.
23. Yang, B.; Yih, W.; He, X.; Gao, J.; Deng, L. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
24. Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; Bouchard, G. Complex Embeddings for Simple Link Prediction. In Proceedings of the 33nd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; pp. 2071–2080.
25. Lotito, Q.F.; Musciotto, F.; Montresor, A.; Battiston, F. Higher-order motif analysis in hypergraphs. *arXiv* **2021**, arXiv:2108.03192.
26. Yu, S.; Xia, F.; Xu, J.; Chen, Z.; Lee, I. OFFER: A Motif Dimensional Framework for Network Representation Learning. In Proceedings of the 29th ACM International Conference on Information and Knowledge Management, Virtual, 19–23 October 2020; pp. 3349–3352.
27. Xia, F.; Sun, K.; Yu, S.; Aziz, A.; Wan, L.; Pan, S.; Liu, H. Graph Learning: A Survey. *IEEE Trans. Artif. Intell.* **2021**, *2*, 109–127. [CrossRef]
28. Yu, S.; Xia, F.; Li, S.; Hou, M.; Sheng, Q.Z. Spatio-temporal Graph Learning for Epidemic Prediction. *ACM Trans. Intell. Syst. Technol.* **2023**, *14*, 1–25. [CrossRef]
29. Xia, F.; Chen, X.; Yu, S.; Hou, M.; Liu, M.; You, L. Coupled Attention Networks for Multivariate Time Series Anomaly Detection. *IEEE Trans. Emerg. Top. Comput.* **2023**, 1–14. [CrossRef]
30. Zhou, X.; Liang, W.; Li, W.; Yan, K.; Shimizu, S.; Wang, K.I. Hierarchical Adversarial Attacks Against Graph-Neural-Network-Based IoT Network Intrusion Detection System. *IEEE Internet Things J.* **2022**, *9*, 9310–9319. [CrossRef]
31. Yang, H.; Li, Z.; Qi, Y. Predicting traffic propagation flow in urban road network with multi-graph convolutional network. *Complex Intell. Syst.* **2023**, 1–13 [CrossRef]
32. Schlichtkrull, M.S.; Kipf, T.N.; Bloem, P.; van den Berg, R.; Titov, I.; Welling, M. Modeling Relational Data with Graph Convolutional Networks. In Proceedings of the European Semantic Web Conference, Heraklion, Greece, 3–7 June 2018; pp. 593–607.
33. Shang, C.; Tang, Y.; Huang, J.; Bi, J.; He, X.; Zhou, B. End-to-End Structure-Aware Convolutional Networks for Knowledge Base Completion. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 3060–3067.
34. Li, Z.; Liu, H.; Zhang, Z.; Liu, T.; Xiong, N.N. Learning Knowledge Graph Embedding with Heterogeneous Relation Attention Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 3961–3973. [CrossRef] [PubMed]
35. Dettmers, T.; Minervini, P.; Stenetorp, P.; Riedel, S. Convolutional 2D Knowledge Graph Embeddings. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 1811–1818.

36. Jiang, X.; Wang, Q.; Wang, B. Adaptive Convolution for Multi-Relational Learning. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 978–987.
37. Nguyen, D.Q.; Nguyen, T.D.; Nguyen, D.Q.; Phung, D.Q. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; pp. 327–333.
38. Vashishth, S.; Sanyal, S.; Nitin, V.; Talukdar, P.P. Composition-based Multi-Relational Graph Convolutional Networks. In Proceedings of the 8th International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
39. Zeb, A.; Haq, A.U.; Zhang, D.; Chen, J.; Gong, Z. KGEL: A novel end-to-end embedding learning framework for knowledge graph completion. *Expert Syst. Appl.* **2021**, *167*, 114164. [CrossRef]
40. Liang, S.; Shao, J.; Zhang, D.; Zhang, J.; Cui, B. DRGI: Deep Relational Graph Infomax for Knowledge Graph Completion. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 2486–2499. [CrossRef]
41. Xia, F.; Yu, S.; Liu, C.; Li, J.; Lee, I. Chief: Clustering with higher-order motifs in big networks. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 990–1005. [CrossRef]
42. Lin, X.V.; Socher, R.; Xiong, C. Multi-Hop Knowledge Graph Reasoning with Reward Shaping. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3243–3253.
43. Hoyt, C.T.; Berrendorf, M.; Galkin, M.; Tresp, V.; Gyori, B.M. A Unified Framework for Rank-based Evaluation Metrics for Link Prediction in Knowledge Graphs. *arXiv* **2022**, arXiv:2203.07544.
44. Sun, Z.; Deng, Z.H.; Nie, J.Y.; Tang, J. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In Proceedings of the 7th International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
45. Zhang, Z.; Wang, J.; Ye, J.; Wu, F. Rethinking Graph Convolutional Networks in Knowledge Graph Completion. In Proceedings of the ACM Web Conference 2022, Lyon, France, 25–29 April 2022; pp. 798–807.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.