

Article

# Rule-Guided Compositional Representation Learning on Knowledge Graphs with Hierarchical Types

Yanying Mao  and Honghui Chen \*

Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China; maoyanying@nudt.edu.cn

\* Correspondence: chenhonghui@nudt.edu.cn; Tel.: +86-133-0848-0588

**Abstract:** The representation learning of the knowledge graph projects the entities and relationships in the triples into a low-dimensional continuous vector space. Early representation learning mostly focused on the information contained in the triplet itself but ignored other useful information. Since entities have different types of representations in different scenarios, the rich information in the types of entity levels is helpful for obtaining a more complete knowledge representation. In this paper, a new knowledge representation frame (TRKRL) combining rule path information and entity hierarchical type information is proposed to exploit interpretability of logical rules and the advantages of entity hierarchical types. Specifically, for entity hierarchical type information, we consider that entities have multiple representations of different types, as well as treat it as the projection matrix of entities, using the type encoder to model entity hierarchical types. For rule path information, we mine Horn rules from the knowledge graph to guide the synthesis of relations in paths. Experimental results show that TRKRL outperforms baselines on the knowledge graph completion task, which indicates that our model is capable of using entity hierarchical type information, relation paths information, and logic rules information for representation learning.



**Citation:** Mao, Y.; Chen, H. Rule-Guided Compositional Representation Learning on Knowledge Graphs with Hierarchical Types. *Mathematics* **2021**, *9*, 1978. <https://doi.org/10.3390/math9161978>

Academic Editor: Frank Werner

Received: 19 July 2021

Accepted: 16 August 2021

Published: 18 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** knowledge graph; representation learning; hierarchical types; logic rules

## 1. Introduction

### 1.1. Research Motivation

Knowledge graphs (KGs), such as Freebase [1], DBpedia [2], and NELL [3], are used to describe the relationship between things in the real world. KGs provide effective structured information and have been widely used in many fields, such as information retrieval [4,5] and question answering [6,7]. A typical knowledge graph usually stores facts in the form of triples (*head, relationship, tail*), denoted  $(h, r, t)$ .

Even though many large KGs often contain billions of triples, they are still incomplete. Specifically, in DBpedia, 60% of individual entities do not indicate their place of birth [8]. Owing to the incompleteness of KGs, it is difficult for people to further apply them to certain scenarios, for example, in a question answering system in which incomplete questions will cause errors in the answers obtained. Therefore, the task of supplementing the missing parts of the KGs has become a top priority.

At present, most KGs completion methods are based on knowledge representation learning [9], which projects the entities and relationships in the triples into a low-dimensional continuous space. TransE [10] is one of the most classic KGs completion models and embeds entities and relationships into the same latent space. To better handle complex relationships, such as 1-to-N, N-to-1, and N-to-N, TransH [11] and TransR [12] use relation-specific hyper-planes and relation-specific spaces, respectively, to separate triples according to their correspondence. However, these models only focus on the triples themselves, ignoring the rich information located in the entity hierarchy types that can also be useful for obtaining a more complete entity representation. Entities have different

types of representations in different scenarios. For example, a man can be the manager of a company or the father of a child, so entities with multiple types should be represented differently in different scenarios. In addition, relation paths in KGs can provide additional relationships for entity pairs. For instance, PTransE [13] successfully uses the relation paths information to obtain embedding of entities and relationships. However, in the present work, the embedding of relationships is randomly initialized, while the representation of paths is obtained by summing or multiplying relations in paths [14]. Since the representation of the path is obtained purely through numerical calculations in the latent space, errors will be propagated, thereby affecting the entire knowledge representation learning. To address this problem, we introduce logic rules with the expectation that the accuracy of logic rules can be used to improve the accuracy of relational path inference. At the same time, the interpretability of logical rules can also enhance the interpretability of relational path inference.

Specifically, we propose a knowledge representation learning framework that combines entity hierarchical type information and rule path information (TRKRL). Moreover, we introduce these bits of information into the embedding level. We regard the entity hierarchical type information as the entity's projection matrix and use a type encoder to model it for addressing the problem that entities have different types of representations in different scenarios. For relation paths information and logic rules information, we use Horn rules mined from KGs to guide the synthesis of relations in the path and improve the accuracy of relational path reasoning, while the interpretability of logic rules can also enhance the interpretability of a model's representation learning. We evaluate the TRKRL model on a benchmark dataset in Freebase, and experimental results show that compared with all baselines TRKRL exhibits a significant and consistent improvement. The main contributions of the present work can be summarized as follows. (1) We introduce logic rules information and use the accuracy of logical rules to improve the accuracy of relational path reasoning. At the same time, the interpretability of logical rules can also improve the interpretability of representation learning. (2) Entity hierarchical type information is introduced to obtain a more comprehensive representation of entities in order to cope with different scenarios in which the same entity has different types. (3) We propose a novel knowledge representation learning model that combines relation paths information, logic rules information, and entity hierarchical type information, while experiments show that our model outperforms all baseline approaches.

## 1.2. Related Work

### 1.2.1. Translation-Based Models

In recent years, great progress has been made in the representation learning of KGs [15–18], and many models are based on translation operations. TransE [10] is the most classic and representative translation-based model. TransE first projects both entities and relationships into the same continuous low-dimensional vector space as  $h, r, t \in R^s$ . The key operation of TransE is then to translate the semantics from head entities to tail entities by relationships. TransE believes that the tail  $t$  should be in the neighborhood of  $h + r$ ; that is,  $h + r \approx t$  when triple  $(h, r, t)$  holds. Hence, the energy function is  $E(h, r, t) = \|h + r - t\|$ . TransE is effective and simple for 1-to-1 relationships but has issues for modeling 1-to-N, N-to-1, and N-to-N relationships.

Some researchers have made efforts to solve the problem of the representation of complex relationships. TransH [11] interprets relations as translating operations on relation-specific hyper-planes, and projects  $h$  and  $r$  to the relation-specific hyper-plane. In this way, different embedded representations of entities are realized when the entities correspond to different relationships. TransR [12] first models entities and relationships in independent entity space and relationship space, and then maps entities from entity space to relationship-specific space. STransE [19] puts the head and tail entities in different spaces on the basis of TransR. TransRHS [20] considers the inherent generalization relationships among relations.

However, these models only focus on the relationship between triples and ignore the rich information carried in the triples, which will be applied in our model TRKRL.

### 1.2.2. Multi-Source Information Learning Models

Multi-source information refers to textual information, type information, and logical information that can complete the triple structure. In terms of text information, Socher et al. [14] proposed representing the entity as the average value of its word embeddings in the entity name, so as to share the textual information of similar entities. Based on the entity name and Wikipedia anchor, Wang et al. [11] and Zhong et al. [21] encode entities and words into the joint vector space. DKRL [22] explores two encoders to represent the semantics of entity descriptions, and considers the zero-shot scenario, in which some entities are novel compared to existing KGs with only descriptions.

Hierarchical entity types information and logic rules information are also significant for KGs. Krompaß et al. [8] propose that the entity types comprise a hard constraint in the KG latent variable model. In order to realize the explicit coding of type information, Xie et al. [23] proposed the TKRL. TKRL considers the hierarchical structure of entity types and solves the problems of noise and incomplete types in hard constraints. The interpretability of logic rules enhances the interpretability of representation learning. For instance, Minervini et al. [24] simply impose equivalence and reversal constraints on relational embedding; Ruge [25] converts triples into complex formulas formed by atoms with logical connectives; Niu et al. [26] explicitly use Horn rules to derive path embeddings and create semantic associations between relationships. However, none of these approaches can simultaneously apply structured information, hierarchical types information, and logic rules information in the representation learning of the SG. The model TRKRL proposed in this paper performs well in fusing multi-source information and improves the interpretability and generalization of representation learning on the KG.

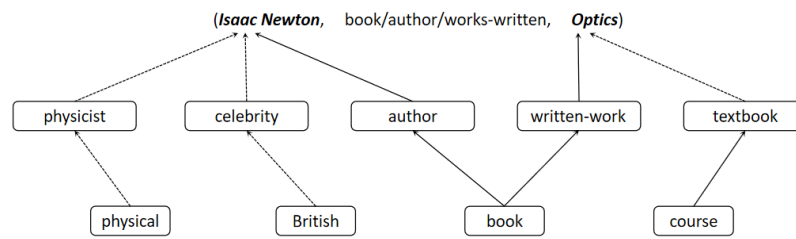
## 2. Methodology

### 2.1. Extraction of Hierarchical Type Information

The fact that the same entity has different meanings at different levels of a scenario is important for the learning of representations in the KG. However, most previous research pays less attention to the rich information located in hierarchical types of entities. Figure 1 shows a triple instance; *Isaac Newton* has a variety of types (e.g., book/author, physical/physicist, and British/celebrity). It is, therefore, reasonable to believe that each entity should be represented differently in different scenarios, as a reflection of itself from different perspectives. Take the example of a hierarchy type  $g$  with  $m$  layers, where  $g^{(i)}$  is the  $i$ th sub-type of  $g$ . Each sub-type  $g^{(i)}$  has only one parent sub-type  $g^{(i+1)}$ , while the most precise sub-type is the first layer, and the most general sub-type is the last. Going through the hierarchy from the bottom-up, we can obtain a representation of the hierarchical type as  $g = \{g^{(1)}, g^{(2)}, \dots, g^{(i)}\}$ . We assign the type-specific projection matrix  $W_g$  to each type  $g$ , and the head  $h$  and tail  $t$  of this relation are then represented in the projection under the particular types as  $g_{rh}$  and  $g_{rt}$ , respectively. The energy function is defined as follows:

$$E(h, r, t) = \|W_{rh}h + r - W_{rt}t\|, \quad (1)$$

in which  $W_{rh}$  and  $W_{rt}$  are different projection matrices for  $h$  and  $t$ , respectively.



**Figure 1.** Example of entity hierarchical types in Freebase.

### 2.1.1. Type Encoder

We use a general form of type encoder to encode hierarchical type information into the representation learning. In the general form of a KG, most entities have more than one type, so the projection matrix  $W_e$  for entity  $e$  is a weighted sum of all type matrices:

$$W_e = a_1 W_{g_1} + a_2 W_{g_2} + \dots + a_n W_{g_n}, \quad (2)$$

where  $n$  is the number of types entity that  $e$  has,  $a_i$  is the weight for  $g_i$ ,  $g_i$  is the  $i$ th type that  $e$  belongs to, and  $W_{g_i}$  represents the projection matrix of  $g_i$ . The weights can be set according to statistical characteristics, such as type frequency. With this operation, all projection matrices of entity  $e$  are the same in different scenes.

In practice, however, the importance of entity attributes varies in different scenarios. Therefore, we have improved the type encoder, and the projection matrix  $W_{rh}$  in a specific triple will be:

$$W_{rh} = \frac{\sum_{i=1}^n a_i W_{g_i}}{\sum_{i=1}^n a_i}, \quad (3)$$

where  $1 \geq a_i > 0$ . Similarly, the projection matrix  $W_{rt}$  of the entity at the tail position can be obtained.

### 2.1.2. Hierarchical Encoder

As mentioned earlier, we consider the type information of entities to be hierarchical, so a recursive hierarchical encoder is used. During the projection process, entities (e.g., Isaac Newton) will be first mapped to the more general sub-type space (e.g., physical) and then be sequentially mapped to the more precise sub-type space (e.g., physical/physicist). The mathematical formula is:

$$W_g = \prod_{i=1}^m W_{g^{(i)}} = W_{g^{(1)}} W_{g^{(2)}} \dots W_{g^{(m)}}, \quad (4)$$

where  $W_g$  is the projection matrix for type  $g$ ,  $W_{g^{(i)}}$  is the projection matrix of the  $i$ th sub-type  $g^{(i)}$ , and  $m$  is the number of layers for type  $g$ .

## 2.2. Extraction of Logic Rules Information

To enable our model to provide more semantic information, we have further fused paths and logic rules information. First, we mine the rules with their confidence levels  $\mu \in [0, 1]$  from the KG. The higher the confidence level of the rule, the higher the probability that it holds. Second, we restrict the maximum length of rules to 2. Thus, rules are classified into two categories according to their length, as follows. (1)  $R_1$ : A rule set of length 1 is called  $R_1$ , which associates two relations in the rule body and rule head. (2)  $R_2$ : A rule set of length 2 is denoted  $R_2$  and it can be used to compose paths. Figure 2 provides specific examples.

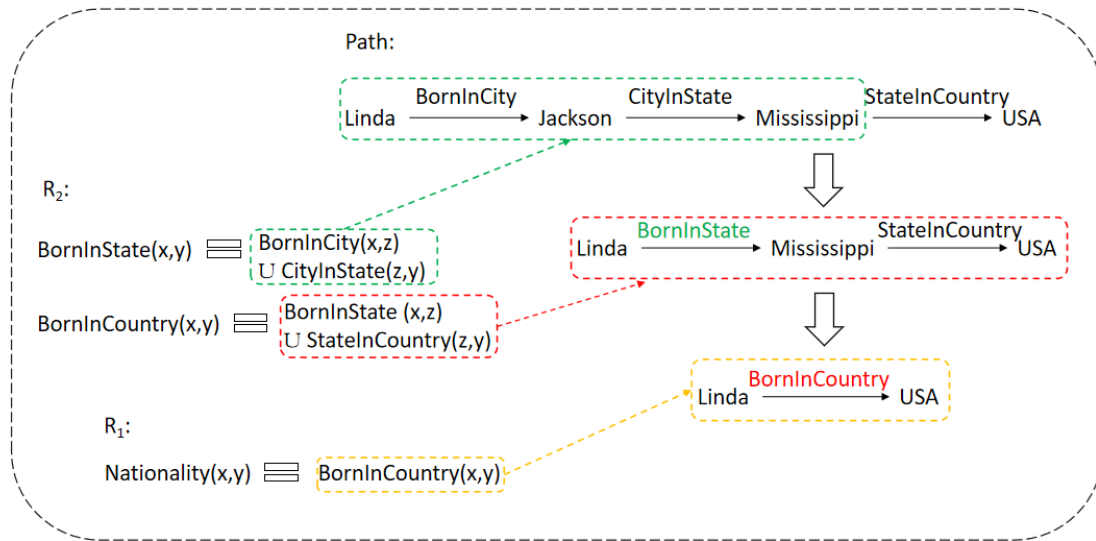


Figure 2. Example of rule  $R_1$  and  $R_2$ .

We use PTransE to implement the path extraction process, where the reliability of each path  $p$  is denoted as  $R(p|h, t)$  between pairs of entities  $(h, t)$ . Table 1 lists the modes for  $R_2$ . Obviously, it is crucial that, in the rule  $R_2$ , which constitutes the path, sequential paths are formed by the atoms of each rule body. Therefore, we encode the eight rules to facilitate the formation of a valid path set  $P(h, t)$  for the entity pair  $(h, t)$ . Taking the original rule  $r_3(a, b) = r_1(b, e) \cup r_2(a, e)$ , for instance, we first convert the atom  $r_1(b, e)$  into  $r_1^{-1}(e, b)$ , and then exchange two atoms in the rule body to obtain a chain rule  $r_3(a, b) = r_2(a, e) \cup r_1^{-1}(e, b)$ , which could be further abbreviated as  $r_3 = r_2 \cup r_1^{-1}$ .

Table 1. List of conversion modes for rule  $R_2$ .

Original Rules	Encoded Rules
$r_3(a, b) = r_1(a, e) \cup r_2(e, b)$	$r_3 = r_1 \cup r_2$
$r_3(a, b) = r_1(e, b) \cup r_2(a, e)$	$r_3 = r_2 \cup r_1$
$r_3(a, b) = r_1(e, b) \cup r_2(e, a)$	$r_3 = r_2^{-1} \cup r_1$
$r_3(a, b) = r_1(e, a) \cup r_2(e, b)$	$r_3 = r_1^{-1} \cup r_2$
$r_3(a, b) = r_1(a, e) \cup r_2(b, e)$	$r_3 = r_1 \cup r_2^{-1}$
$r_3(a, b) = r_1(b, e) \cup r_2(a, e)$	$r_3 = r_2 \cup r_1^{-1}$
$r_3(a, b) = r_1(e, a) \cup r_2(b, e)$	$r_3 = r_1^{-1} \cup r_2^{-1}$
$r_3(a, b) = r_1(b, e) \cup r_2(e, a)$	$r_3 = r_2^{-1} \cup r_1^{-1}$

In order to make full use of the encoded rules, we should traverse the paths and iteratively perform the composition operation at the semantic level until the rules cannot combine any relations. In the actual path synthesis process, consider the optimal case in which all relations in the path can be synthesized iteratively by the rule  $R_2$  and eventually joined together as a single relation between pairs of entities. In addition, when the path can match multiple rules at the same time, we choose the rule with a high confidence level to form the path. This leads to the path embedding  $H(p)$  of the path  $p$ .

When rule  $R_1$  holds, relation  $r_1$  may have more semantic similarity to its directly implicated relation  $r_2$ . We, therefore, encode rules of the form of representation learning,  $(a, r_2, b) = (b, r_1, a)$  as  $(a, r_2, b) = (a, r_1^{-1}, b)$ . During training, embedding representing pairs of relations that appear simultaneously in rule  $R_1$  are considered closer than embedding of two relations that do not match any rule.

### 2.3. Integration of Information

For each triple  $(h, r, t)$ , we define three energy functions that model correlations for direct triples and hierarchical type methods, path pairs using rule  $R_2$ , and relationship pairs using rule  $R_1$ :

$$E_1(h, r, t) = \|W_{rh}h + r - W_{rt}t\|, \tag{5}$$

$$E_2(p, r) = R(p|h, t) \left( \prod_{\mu_i \in U(p)} \mu_i \right) \|H(p) - r\|, \tag{6}$$

$$E_3(r, r_e) = \|r - r_e\|, \tag{7}$$

where  $E_1(h, r, t)$  measures the effectiveness of type information.  $E_2(p, r)$  denotes the energy function evaluating the similarity between path  $p$  and relation  $r$ , and  $U(p) = \mu_1, \dots, \mu_n$  denotes the set of confidence levels corresponding to all rules in rule  $R_2$  employed in the composition of path  $p$ .  $E_3(r, r_e)$  is an energy function that represents the similarity between a relation  $r$  and another relation  $r_e$ . If the relations contained in the relation  $r_e$  are re-defined using rule  $R_1$ , it should be assigned a smaller fraction.

### 2.4. Loss Function and Optimization

We formalize the loss function as a margin-based score function targeting negative sample sampling:

$$L = \sum_{(h,r,t) \in T} \sum_{(h',r',t') \in T'} (L_1(h, r, t) + \alpha_1 \sum_{p \in P(h,t)} L_2(p, r) + \alpha_2 \sum_{r_e \in R(r)} L_3(r, r_e)), \tag{8}$$

where  $T$  represents a set that contains all the positive triples observed in KG.  $T'$  is the negative sampling set of  $T$ ,  $R_r$  is the set of all relations deduced from  $r$  on the basis of rule  $R_1$ , and  $r_e$  is any one of the relations in  $R_r$ .  $P(h, t)$  is the set of all paths connecting entity pair  $(h, t)$ , of which  $p$  is a path.  $L_1, L_2$ , and  $L_3$  correspond to marginal-based loss functions for the triple  $(h, r, t)$  of entity hierarchical types, path pairs  $(p, r)$ , and relationship pairs  $(r, r_e)$ :

$$L_1(h, r, t) = \max(0, \gamma_1 + E(h, r, t) - E(h', r', t')), \tag{9}$$

$$L_2(p, r) = \max(0, \gamma_2 + E_2(p, r) - E_2(p', r')), \tag{10}$$

$$L_3(r, r_e) = \max(0, \gamma_3 + \beta E_3(r, r_e) - E(r, r')), \tag{11}$$

where  $\gamma_1, \gamma_2$ , and  $\gamma_3 > 0$  are hyper-parameters;  $\beta$  denotes the confidence level of associations  $r$  and  $r_e$ .

Since there are no explicit negative triples in KGs, the entities or relationships in the training triples are randomly replaced by any other entity in  $E$ . Moreover, the new triples after replacements will not be considered as negative samples if they are already in  $T$ . In addition, the negative triples sampling rule is expressed as follows:

$$T' = (h', r, t) \cup (h, r', t) \cup (h, r, t'). \tag{12}$$

For optimization, mini-batch stochastic gradient descent (SGD) is used to minimize the loss function. The projection matrix set  $W$  could be initialized randomly or by identity matrix. In addition, the embeddings of entities and relations could be either initialized randomly or be pre-trained by existing translation-based models, such as TransE.

## 3. Experiments

### 3.1. Experiment Settings

#### 3.1.1. Datasets

We evaluate our model on two typical KGs, i.e., FB15K and FB15K-237, which are extracted from the large-scale Freebase [1]. FB15K contains 14,951 entities, 1345 relations, and 592,213 triples in total, and we split the triples into training, validation, and testing sets.

We collected a total of 571 entity types in FB15K, with the average number of types for all entities being approximately eight and having at least one hierarchical type. However, in order to verify the validity of the logic rule information, the FB15K-237 dataset is also used in the experiment. Note that FB15K-237 contains no inverse relation; hence, it is difficult to learn embeddings by these mutually independent relations, so it is different than the FB15K dataset. The statistics of all datasets are listed in Table 2.

**Table 2.** Dataset statistics.

Dataset	Relationships	Entities	Training	Validation	Testing
FB15K	1345	14,951	483,142	50,000	59,071
FB15K-237	237	14,541	272,115	17,535	20,466

We collect all type instances of *type/instance* fields in FB15K, as well as the relationship-specific type information distributed in the *rdf-schema#domain* and *rdf-schema#range* fields. Regarding the logic rules information, we choose AMIE+ as the rule mining tool for its convenience and fast speed to mine rich information. We set the confidence threshold to be chosen in the range [0, 1] in steps of 0.1 to search for the best performance of the rule on the dataset.

### 3.1.2. Settings

TransE and TransR are the comparison objects of the proposed models. Considering the differences in application scenarios, we make changes in their original settings. We first use the  $L_1$ -norm to improve the dissimilarity measure of TransE. Then, in the negative sampling process, we replace the relationship and the entity, and use “bern” to represent the head or tail of different probabilities %. Similarly, we perform relationship replacement during the negative sampling process of TransR and train with the best parameters marked in the paper %.

We use mini-batch SGD to help train the TRKRL model. In this paper, the best configuration of parameters is size  $S = 4800$ , margin  $\gamma = 1.0$ , descending weight  $\eta = 0.1$ , and learning rate  $\lambda$  designed by a linear-declined function. The training dimension for all models is 100. In the course of our experiments, we used several models for comparison. Among them, TransE and TransR are trained with the best parameters reported in their respective papers. For other baselines including RESCAL, SE, SME, LFM, and TKRL, we use the results reported directly.

## 3.2. KG Completion

### 3.2.1. Evaluation Protocol

The complementary task of the KG refers to completing any of the missing elements in the triple. Taking entity prediction as an example, the comparison process of relationship prediction is similar. Three principle assessment metrics are focused on, i.e., (1) the mean rank of correct entities (MR), (2) mean reciprocal rank of correct entities (MRR), and (3) proportion of correct answers ranked in top n (Hits@n). The evaluation settings are named “Raw” and “Filter”.

The KG completion task requires entity and relationship information, so we divide this task into entity and relationship prediction sub-tasks. We use FB15K for evaluation and the same evaluation conditions for all models to ensure the reliability of the results.

### 3.2.2. Entity Prediction

Table 3 shows the entity prediction results, from which we can observe the following. (1) Our method (TRKRL) surpasses other baselines in all indicators. This illustrates that the fusion of logical rule information and hierarchical type information can improve the representation learning of the KG. (2) The results of TRKRL and TKRL on the MR and the number of Hits@10 are better than those of all baselines. The results show that the embedding of the hierarchical type information of entities and relationships can improve

the representation learning of the KG. (3) In particular, TRKRL is superior to TKRL in every metric, which shows the advantages of introducing logical rules in providing higher path synthesis accuracy and learning better path embedding.

**Table 3.** Entity prediction results on FB15K. Best scores are in bold; second-best scores are underlined.

Metric	Mean Rank		Hits@10 (%)	
	Raw	Filtered	Raw	Filtered
RESCAL	828	683	28.4	44.1
SE	273	162	28.8	39.8
SME	274	154	30.7	40.8
LFM	283	164	26.0	33.1
TransE	236	142	46.9	62.3
TransR	198	75	47.3	67.3
TKRL	<u>184</u>	<u>68</u>	<u>49.2</u>	<u>69.4</u>
TRKRL	<b>182</b>	<b>65</b>	<b>50.5</b>	<b>73.6</b>

### 3.2.3. Relationship Prediction

Table 4 displays the results on the FB15K dataset for all compared methods. We adopt two classic models, TransE and TransR, as comparison objects. Consistent with our conjecture, the results obtained after data filtering have lower mean ranks and higher hits@10 than the results of without filtering. More specifically, we observe the following. (1) Our method, TRKRL, outperforms all baselines on all metrics. In particular, it achieves a superior absolute performance score of 94.1% on the hits@10 index. This indicates that the logic rules information added in TRKRL is not only conducive to entity prediction, but also conducive to relationship prediction. (2) The mean rank results of TKRL and TRKRL before filtering are significantly improved compared to other baselines, which illustrates the positive impact of hierarchical type information as a constraint.

**Table 4.** Relationship prediction results on FB15K. Best score in bold.

Metric	Mean Rank		Hits@10 (%)	
	Raw	Filtered	Raw	Filtered
TransE	2.81	2.51	67.5	88.3
TransR	2.63	2.22	71.4	90.7
TKRL	2.12	1.73	71.1	92.8
TRKRL	<b>2.08</b>	<b>1.69</b>	<b>72.6</b>	<b>94.1</b>

### 3.2.4. Ablation Study

In order to fully prove the universality and reliability of our proposed method, we also conducted test experiments on the FB15K-237 dataset. Compared with the classic datasets (i.e., FB15K, WN18, etc.), the FB15K-237 dataset has been constructed only recently. At present, relatively little work has been done on this dataset showing test results, so we can use it as a baseline. Table 5 shows the experimental results, in which TRKRL obtains the best performance with approximately 25.47% improvement compared to best baseline TransR on Mean Reciprocal Rank. According to the results of Mean Reciprocal Rank and Hits@10, it is found that TransR is more suitable for the FB15K-237 dataset than most of the baselines. This may be attributed to the fact that TransR clusters entities with the same relationship. Although FB15K-237 eliminates the reverse relation, we can use Horn rules to help establish semantic associations.

To verify that the components of TRKRL are meaningful, we performed entity prediction ablation experiments on FB15K, and removed the paths, hierarchical types, and logic rules from TRKRL. As shown in Table 6, TRKRL-P, TRKRL-HT, and TRKRL-LR represent the model TRKRL without paths, hierarchical types, and logic rules, respectively. Obviously, deleting any one component will cause the performance degradation of the model.



This illustrates that the multi-information fusion theory proposed by us is completely beneficial to knowledge representation learning.

**Table 5.** Entity prediction results on FB15K-237. Best score is in bold; second-best score is underlined.

Metric	Mean Rank	Mean Reciprocal Rank	Hits@10 (%)
SME	483	0.255	30.1
LFM	462	0.271	33.8
TransE	345	0.282	50.3
TransR	298	<u>0.369</u>	<u>59.7</u>
TKRL	<u>204</u>	0.327	54.5
TRKRL	<b>192</b>	<b>0.463</b>	<b>63.4</b>

**Table 6.** Ablation study by removing paths, hierarchical types, and logic rules.

Metric	Mean Rank		Hits@10 (%)	
	Raw	Filtered	Raw	Filtered
TRKRL	<b>182</b>	<b>65</b>	<b>50.5</b>	<b>73.6</b>
TRKRL-P	208	96	40.9	52.3
TRKRL-HT	194	75	47.1	64.4
TRKRL-LR	189	68.4	49.3	70.5

### 3.3. Triple Classification

#### 3.3.1. Evaluation Protocol

The purpose of this task is to confirm whether the triple  $(h, r, t)$  is correct or not. This task has been considered as one of the indicators for evaluating the performance of the learning model. To accomplish this task, we constructed negative examples for the FB15K dataset according to the method of Socher et al. [14]. The specific method is to determine different thresholds  $\zeta$  for different relationships. When the dissimilarity score  $E(h, r, t)$  of the triple is higher than the threshold  $\zeta$ , it is considered negative; otherwise, it is positive.

#### 3.3.2. Results

Table 7 shows the result of triple classification, from which we can observe that TRKRL has the best performance, which shows the advantages of TRKRL over baselines in the triple classification, and further proves the superiority of the fusion of logical rules information and hierarchical type information.

**Table 7.** Evaluation results on triple classification. Best score is in bold; second-best score is underlined.

Methods	Accuracy (%)
TransE	85.6
TransR	86.5
TKRL	<u>88.5</u>
TRKRL	<b>88.7</b>

## 4. Conclusions

In this paper, we propose the knowledge graph representation learning framework TRKRL, which combines rule path information and entity hierarchical type information. By integrating entity hierarchical type information, Horn rules, and relationship path information in a triple embedding framework, TRKRL improves the accuracy of representation learning and obtains better knowledge representation. For entity hierarchical type information, we use a type encoder to model the hierarchical type information and then treat it as a projection matrix of entities to cope with different scenarios in which entities have different type representations. For Horn rules and relational path information, we use logical rules to guide the synthesis of relations in paths to improve the accuracy of

relational path reasoning. In addition, logical rules can also enhance the interpretability of representation learning. Experimental results show that TRKRL outperforms all other baselines, which illustrates the importance of entity hierarchical type information and logical rules information in guiding the synthesis of relationships in paths.

In planned future work, we will explore the following directions: (1) exploring new entity hierarchical type encoders to better model entity hierarchical type information; (2) exploring potential rules to guide the synthesis of relationships in the path to better combining rules and paths; and (3) introducing other auxiliary information, such as textual information and visual information, in order to learn a more complete representation.

**Author Contributions:** Propose the project, Y.M.; mathematical methods, Y.M.; experimental proof, Y.M.; writing—original, Y.M.; writing—review and editing, H.C. All authors have read and agreed to submitting the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Acknowledgments:** The authors are sincerely thankful to the editors and anonymous referees.

**Conflicts of Interest:** The authors declare no conflicts of interest.

### Abbreviations

The following abbreviations are used in this manuscript:

KGs	Knowledge Graphs
TRKRL	Rule-Guided Compositional Representation Learning on Knowledge Graphs with Hierarchical Types

### References

- Bollacker, K.D.; Evans, C.; Paritosh, P.; Sturge, T.; Taylor, J. Freebase: A collaboratively created graph database for structuring human knowledge. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Vancouver, BC, Canada, 10–12 June 2008; pp. 1247–1250.
- Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P.N.; Hellmann, S.; Morse, M.; van Kleef, P.; Auer, S.; et al. DBpedia—A large-scale, multilingual knowledge base extracted from Wikipedia. *Semant. Web* **2015**, *6*, 167–195. [\[CrossRef\]](#)
- Mitchell, T.; Cohen, W.; Hruschka, E.; Talukdar, P.; Yang, B.; Betteridge, J.; Carlson, A.; Dalvi, B.; Gardner, M.; Kisiel, B.; et al. Never-ending learning. *Commun. ACM* **2018**, *61*, 103–115. [\[CrossRef\]](#)
- Zhang, F.; Yuan, N.J.; Lian, D.; Xie, X.; Ma, W. Collaborative Knowledge Base Embedding for Recommender Systems. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 353–362. [\[CrossRef\]](#)
- Ke, P.; Ji, H.; Ran, Y.; Cui, X.; Wang, L.; Song, L.; Zhu, X.; Huang, M. JointGT: Graph-Text Joint Representation Learning for Text Generation from Knowledge Graphs. *arXiv* **2021**, arXiv:2106.10502. [\[CrossRef\]](#)
- Hao, Y.; Liu, H.; He, S.; Liu, K.; Zhao, J. Patternrevising enhanced simple question answering over knowledge bases. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; pp. 3272–3282.
- Li, C.; Xian, X.; Ai, X.; Cui, Z. Representation Learning of Knowledge Graphs with Embedding Subspaces. *Sci. Program.* **2020**, *2020*, 4741963:1–4741963:10. [\[CrossRef\]](#)
- Krompaß, D.; Baier, S.; Tresp, V. Type-Constrained Representation Learning in Knowledge Graphs. In Proceedings of the Semantic Web—ISWC 2015—14th International Semantic Web Conference, Bethlehem, PA, USA, 11–15 October 2015; pp. 640–655. [\[CrossRef\]](#)
- Zhang, Z.; Cao, L.; Chen, X.; Tang, W.; Xu, Z.; Meng, Y. Representation Learning of Knowledge Graphs with Entity Attributes. *IEEE Access* **2020**, 7435–7441. [\[CrossRef\]](#)
- Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; pp. 2787–2795.
- Wang, Z.; Zhang, J.; Feng, J.; Chen, Z. Knowledge Graph Embedding by Translating on Hyperplanes. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, QC, Canada, 27–31 July 2014; pp. 1112–1119.
- Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X. Learning Entity and Relation Embeddings for Knowledge Graph Completion. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; pp. 2181–2187.

13. Lin, Y.; Liu, Z.; Luan, H.; Sun, M.; Rao, S.; Liu, S. Modeling Relation Paths for Representation Learning of Knowledge Bases. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 705–714. [[CrossRef](#)]
14. Socher, R.; Chen, D.; Manning, C.D.; Ng, A.Y. Reasoning With Neural Tensor Networks for Knowledge Base Completion. In Proceedings of the Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–8 December 2013; pp. 926–934.
15. Ait-Mlouk, A.; Vu, X.S.; Jiang, L. WINFRA: A Web-Based Platform for Semantic Data Retrieval and Data Analytics. *Mathematics* **2020**, *8*, 2090. [[CrossRef](#)]
16. Miao, H.; Zhang, Y.; Wang, D.; Feng, S. Multi-Output Learning Based on Multimodal GCN and Co-Attention for Image Aesthetics and Emotion Analysis. *Mathematics* **2021**, *9*, 1437. [[CrossRef](#)]
17. Oehlers, M.; Fabian, B. Graph Metrics for Network Robustness—A Survey. *Mathematics* **2021**, *9*, 895. [[CrossRef](#)]
18. Hur, Y.; Jo, J. Development of Intelligent Information System for Digital Cultural Contents. *Mathematics* **2021**, *9*, 238. [[CrossRef](#)]
19. Nguyen, D.Q.; Sirts, K.; Qu, L.; Johnson, M. STransE: A novel embedding model of entities and relationships in knowledge bases. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 460–466. [[CrossRef](#)]
20. Zhang, F.; Wang, X.; Li, Z.; Li, J. TransRHS: A Representation Learning Method for Knowledge Graphs with Relation Hierarchical Structure. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI, Yokohama, Japan, 7–15 January 2021; pp. 2987–2993. [[CrossRef](#)]
21. Zhong, H.; Zhang, J.; Wang, Z.; Wan, H.; Chen, Z. Aligning Knowledge and Text Embeddings by Entity Descriptions. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 267–272. [[CrossRef](#)]
22. Xie, R.; Liu, Z.; Jia, J.; Luan, H.; Sun, M. Representation Learning of Knowledge Graphs with Entity Descriptions. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 2659–2665.
23. Xie, R.; Liu, Z.; Sun, M. Representation Learning of Knowledge Graphs with Hierarchical Types. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 2965–2971.
24. Minervini, P.; Costabello, L.; Muñoz, E.; Nováček, V.; Vandembussche, P. Regularizing Knowledge Graph Embeddings via Equivalence and Inversion Axioms. In Proceedings of the Machine Learning and Knowledge Discovery in Databases—European Conference, Skopje, Macedonia, 18–22 September 2017; pp. 668–683. [[CrossRef](#)]
25. Guo, S.; Wang, Q.; Wang, L.; Wang, B.; Guo, L. Knowledge Graph Embedding With Iterative Guidance From Soft Rules. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 4816–4823.
26. Niu, G.; Zhang, Y.; Li, B.; Cui, P.; Liu, S.; Li, J.; Zhang, X. Rule-Guided Compositional Representation Learning on Knowledge Graphs. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; pp. 2950–2958.