FOG: A Generic Framework for Knowledge Graph Embedding with Ontology Guided Relational Constrains

Tengwei Song [a], Jie Luo [a,*] and Xiangyu Chen [a]
[a] *State Key Laboratory of Software Development Environment, Beihang University, Beijing, China*
E-mail: {songtengwei, luojie, xiangyu_chen}@buaa.edu.cn

Abstract. Many knowledge graph (KG) embedding models have been proposed for knowledge acquisition tasks and have achieved high performance on common evaluation metrics. However, many current KG embedding models have only limited capability for complex implicit information reasoning and may derive results that contradict the ontology of the KG. To tackle this problem, we propose an ontology-guided joint embedding framework to incorporate the constraints specified in the ontology into the representation learned by KG embedding models through a joint loss function, which is defined on positive and negative instances derived from two sets of ontology axioms. Furthermore, we propose two additional reasoning capability evaluation metrics for measuring the capability of models to correctly predict relations or links deduced from the KG and ontology, and avoid miss-predictions. The experimental results demonstrated that models with our framework performed better in most cases across tasks and datasets, and performed significantly better for reasoning capability evaluation metrics in many cases.

Keywords: knowledge graph, ontology, relation property, embedding, representation learning

1. Introduction

Knowledge graph embedding aims to mapping the entities and relations in KGs to low-dimensional vectors and capture their semantics. This reduces the dependence of machine learning algorithms on feature engineering [1] in downstream tasks, such as question answering [2–4], information extraction [5, 6], and item recommendation [7, 8].

Because relations in KGs exhibit multiple patterns, the first strand of existing KG embedding models focuses on finding embeddings that could preserve relational patterns. For instance, RotatE [9] models a relation as rotation from the source to the target for inferring, for example, symmetry and asymmetry, and Rot-Pro [10] models a relation as an additional projection on both entities simultaneously to model transitivity. However, in addition to relational patterns, there also exist relational constraints in relation properties. For example, for a functional relation property, there is an implicit constraint that for a certain entity with a functional relation, there is only one positive fact. Although these models have the capability to express these patterns, these models still derive results that violate such relational constraints. The neglect of relational constraints may cause miss-predictions, particularly for KGs with large-scale instances and complex schema.

Relational constraints could be naturally expressed in logic rule form, which is defined in the ontology layer of the KG. As a result of the build and refinement of the Web Ontology Language (OWL) [11], many open knowledge bases, such as Freebase [12], YAGO [13], and DBpedia [14], have comprehensive ontologies that contain rich relation properties and class hierarchies [15]. Ontology information has been widely used as the second strand of KG embedding methods because the instance layer alone cannot make sufficient use of all the structural information of KGs. These

*Corresponding author. E-mail: luojie@buaa.edu.cn.

methods focus on learning KG representations by incorporating ontology information into the embedding and use the joint loss function technique for embedding such information with KG instances simultaneously.

Different from the first strand methods, we do not force KG embedding models to find a projection that fit relational patterns. Similar to the second strand, we utilize ontology information, but transform the description of ontology in OWL to logical rules to generate instances that feed the KG embedding models to generically learn both relational patterns and relational constrains.

In detail, we introduce FOG, which is a generic **o**ntology-**g**uided semantic reasoning approach for learning relational patterns and constraints. Specifically, FOG derives positive and negative examples semantically by dividing axioms specified in the ontology into two sets, that is, C-set for deriving positive examples and V-set for deriving negative examples, which enable us to handle axioms in the ontology in a uniform manner other than item by item. We further propose a novel loss function with semantic constraints based on the positive and negative examples derived by C-set and V-set, which fits the semantics of the KG under the open-world assumption (OWA) [16] better. By combining the reasoning approach with the loss function, we present the ontology-guided joint embedding framework for KG representation learning.

To evaluate the effectiveness of the ontology-guided joint embedding framework, we propose evaluation metrics to measure the semantic reasoning capability of the model based on whether the inference results of the model obey or violate the axioms in the ontology. The experimental results for five representative KG embedding models on three widely used benchmark datasets demonstrated that our framework not only improved the performance of these models on standard metrics, but also largely improved their performance on the proposed reasoning capability evaluation metrics. This helps us to reveal the advantages and weaknesses of models.

To summarize, the main contributions of this paper are as follows:

- We introduce an approach for dividing axioms in the ontology into a conformance set and violation set, which we use to semantically derive true positive and true negative instances of the KG, respectively.
- We propose a generic ontology-guided joint embedding framework for KG representation learning through a loss function with semantic constraints defined on top of instances derived through the C-set and V-set axioms.
- We define two reasoning capability evaluation metrics for measuring the capability of KG embedding models for predicting triples that are derived from KG through complex reasoning and avoiding miss-predictions for triples that contradict the KG.
- Experiments on five representative models demonstrated that models with our framework obtained better performance, not only for three common evaluation metrics but also for the two proposed reasoning capability evaluation metrics across two KG completion tasks and three datasets.

The organization of this paper is as follows: We discuss related work in Section 2. Then, we illustrate the necessary preliminaries of the KG embedding techniques and present the ontology analysis in Section 3. In Section 4, we introduce the proposed framework, FOG, from the perspective of majority modules based on OWA. Meanwhile, we also propose generic evaluation metrics for measuring the model's capability to infer relational patterns as well as learn relational constraints. Finally, we discuss the experimental study and the corresponding analysis in Section 5, followed by the conclusions in Section 6.

2. Related work

In recent years, many embedding models that encode entities and relations of KGs through specific architectures have been proposed [17]. In this section, we summarize related work on KG embedding from the following two perspectives:

Embedding models focus on instances. The first research direction for this type of KG embedding model is to build distance-based models, whose scoring functions assume that the embedding of subject entity s should be close to the embedding of object entity o after the corresponding transformation of relation r. TransE [18], a distance-based scoring function assumes the added embedding of subject entity s and relation r should be close to the embedding of object entity o. Following TransE, many variants and extensions were proposed. TransH [19] projects entities and relations into a relation-specific hyperplane, which enables different projections of an entity for different relations. TransR [20] introduces relation-specific spaces, which builds entity and relation embeddings in different spaces separately. Recently, Tan et al. proposed a generic model Gtrans [21] to ential all the existing translation models using multi-state entities dynamic relation spaces. RotatE [9] regards relations as rotations from source entities to target entities, and hence can also capture additional inversion and composition patterns by introducing rotational Hadmard product. To better handle 1-To-N relations, PairRe [22] proposed a method to model each relation using paired vectors to project the corresponding head and tail entities. To remedy the drawback that previous models cannot model the transitive relation pattern, Rot-Pro [10] imposes a projection on both source and target entities to express transitivity, and uses a rotation operation as RotatE to support other relation patterns.

Semantic matching models are another research direction for defining scoring functions. DistMult [23] is a bilinear diagonal model for multi-relational representation learning. HolE [24] is a holographic embedding model which introduces circular correlations. Dihedral [25] is a recent model that models relations in KGs using the representation of a dihedral group with properties to support the relations as symmetry. To expand Euclidean space, ComplEx [26] first introduces a complex vector space that can capture both symmetric and asymmetric relations. QuatE [27] uses a quaternion inner product, that is, the Hamilton product, to capture the latent dependency within four-dimensional space of relations and entities, and gains more expressive semantic learning capability than RotatE.

Deep neural networks represent another branch for encoding and have yielded remarkable performance in recent studies. SME [28], MLP [29], and NTN [30] attempt to encode linear/bilinear projecting blocks to neural networks. Recently, models that applying convolutional neural networks (CNNs) as ConvE [31] and ConvKB [32] have attracted more attention to learning deep expressive features.

With the popularity of graph neural networks (GNNs), GNN-based models have been introduced to learn the graph data structure under an encoder-decoder framework [17]. KBGAT [33] is an attention-based embedding model that captures both the entity and relation features of the neighborhoods of any given entity. R-GCN [34] is another GNN-based model and applies graph convolutional networks (GCNs) [35] to act as a graph encoder on relational knowledge bases. The latest model GAATs [36], integrates an attenuated attention mechanism to assign different weights in different relation paths and acquires information from the neighbors.

Joint embedding models with ontology. To facilitate more effective KG embedding models, the ontology information often used to incorporate can be roughly divided into three categories: (i) ontology rule information; (ii) ontology typing information; and (iii) ontology hierarchy information.

Wang et al. [37] used rules to improve the embedding model and expressed the KG completion task as an integer linear programming problem, where the objective function is generated by the embedding model and the constraints are generated by rules. KALE [38] is a unified framework for jointly embedding facts and logical rules in the KG. The confidence of each rule is calculated using modal logic and filtered manually. UOKGE [39] learns the embeddings of entities, classes, and properties on uncertain ontology-aware KGs according to confidence scores, where the confidence scores of unobserved facts are inferred during the training process through probabilistic soft logic. IterE [40] is an iterative KG representation learning framework, which mainly focuses on the sparsity problem. It applies modal logic that is similar to instantiating rules, and considers that the confidence of a propositional logic expression is composed of the confidence of the instances connected by specific logical conjunctions.

Li et al. [41] proposed a framework that embeds entities and categories into a semantic space by integrating structured knowledge and the taxonomy hierarchy from large knowledge bases to fit concept categorization and dataless hierarchical classification tasks. JOIE [42] designed a mechanism to categorize KGs as an instance view and ontology view, and designed a specified mapping method by assuming that any instances are close to their corresponding concept in the same space, or in the transformed space. UOKGE [39] is a novel embedding model that learns the embeddings of entities, classes, and properties on uncertain ontology-aware KGs according to confidence scores.

The semantic hierarchy is a ubiquitous property in KGs and was widely used in previous studies. TKRL [43] embeds hierarchy type information into KG embeddings, and considers hierarchical types as projection matrices for entities so that the entities have multiple representations in different types. hTransM [44] is another hierarchy-constrained link prediction method that is based on translation-based KG embedding methods. A recently proposed hierarchy-aware model HAKE [45] makes improvements by mapping entities into the polar coordinate system, and can automatically learn the semantic hierarchy in KGs without using clustering algorithms. Nickel et al. [46] proposed an embedding method based on Poincaré ball for learning hierarchical representations. The recent ATTH [47] is a low-dimensional hyperbolic KG embedding method that captures tree-like structures, and hence modeling hierarchy data.

3. Preliminaries

Before providing the details of the ontology-guided joint embedding framework, we first introduce a few concepts that are used in the following sections.

3.1. Knowledge Graph

A KG typically consists of a schema and a set of instances, where the schema is an ontology specified in description languages, such as RDFS and OWL2 RL. Hence, KG $\mathcal{G}$ is defined as a pair $(\mathcal{O}, \mathcal{I})$, where O denotes the ontology and I denotes the set of instances [48]. For a typical KG, the size of the instances is far larger than that of the ontology, that is, $|\mathcal{I}| \gg |\mathcal{O}|$. Each instance in $\mathcal{I}$ is a triple of the form (s, r, o), where s is the subject entity, r is a relation (or predicate), and o is the object entity.

The instances of KGs are typically used to model factual knowledge, that is, relations between entities, whereas the ontology is used to model the semantic constraints that entities and relations must obey. Thus, a KG can be regarded as a labeled directed graph, in which a relation can be regarded as a link from the subject vertex to the object vertex.

3.2. Ontology

An ontology consists of a set of axioms that specify the semantic constraints that certain knowledge must satisfy. A typical ontology written in OWL 2 Web Ontology Language (OWL2) [11] mainly contains three types of axioms: axioms about classes, axioms about relations and data properties, and axioms about roles [49].

In this study, we focus on axioms about relations and attempt to make the KG embedding reflect the semantic constraints defined by these axioms. For the ontology of KGs, two ontology description languages are commonly used: RDFS for specifying a simple ontology and an OWL 2 RL language profile [50] for specifying an ontology that requires more powerful expressiveness. To gain and use a stronger reasoning capability, we write the ontology in this study in the OWL 2 RL language.

Axioms about relations in OWL 2 RL are typically specified by defining semantic properties about relations. These semantic properties include irreflexive, symmetric, asymmetric, transitive, functional, inverse functional, inverse to some other relations, subproperty of some other relations, and equivalent to some other relations. The exact semantics are defined as follows:

(1) For a *reflective* relation r, (e, r, e) is an instance of r for any entity e.
(2) For an *irreflective* relation r, (e, r, e) is not an instance of r for any entity e.
(3) For a *symmetric* relation r, if (s, r, o) is a relation instance, then its symmetric form (o, r, s) is also an instance of r.
(4) For an *asymmetric* relation r, if (s, r, o) is a relation instance, then its symmetric form (o, r, s) is not an instance of r.
(5) For a *transitive* relation r, if (a, r, b) and (b, r, c) are both instances of r, then (a, r, c) is also an instance of r.
(6) For a *functional* relation r, if (a, r, b) is an instance of r, then for any $c \neq b$, (a, r, c) is not an instance of r.
(7) For an *inverse functional* relation r, if (a, r, b) is an instance of r, then for any $c \neq a$, (c, r, b) is not an instance of r.
(8) For a relation r that is *inverse to* a relation p, if (s, r, o) is an instance of r, then (o, p, s) is an instance of p.
(9) For a relation r_1 that is a *subproperty of* relation r_2, if (s, r_1, o) is an instance of r_1, then (s, r_2, o) is an instance of r_2.
(10) For a relation r_1 that is *equivalent to* relation r_2, (s, r_1, o) is an instance of r_1 if and only if (iff) (s, r_2, o) is an instance of r_2.

3.3. Knowledge graph embedding

To enable the semantics of structured data in the KG to be modeled and learned by a machine, KG embedding methods use various machine learning strategies. The overall procedure can be summarized from the following aspects.

Representation Space. Current KG embedding models mainly use Euclidean space for the representation of a vector, matrix, and tensor. Additionally, motivated by Euler's identity, models such as RotatE represent entities and relations in a complex space to support taking relation as a rotation. Hyperbolic space is also used for KG embedding since its geometric feature benefits the modeling of hierarchical structure among entities [47].

Scoring function. The scoring function is used to measure the plausibility of facts [17]. There are two typical types of scoring functions. Distance-based scoring functions represented by TransX aim to calculate the Euclidean distance between the relational projection of entities. Semantic matching scoring functions calculate the semantic similarity, which typically adopts a multiplicative formulation. Different scoring functions are designed by different models as their measurement criteria.

Negative Sampling. Given a positive instance $\tau^+ = (s, r, o) \in \mathcal{I}$, where $\mathcal{I}$ stores positive instances in the KG, negative sampling generates negative instances by replacing either the subject s or object o with a random entity sampled uniformly from entity set $\mathcal{E}$ [18]. $\mathcal{I}^-$ denotes the set of negative instances. Note that under the OWA, instances in $\mathcal{I}^-$ are less likely to be true, but not definitely wrong.

Loss Functions. Two types of loss function are commonly used by KG embedding models. Both of them fit the OWA and are listed below.

Margin-based loss: Translation-based models [18–20, 51] generally apply margin-based loss (also called pairwise ranking loss) [52] to maximize the discriminative margin γ between a positive instance $\tau^+ = (s, r, o)$ and negative instance $\tau^- = (s', r, o')$. Given a positive set $\mathcal{I}$, a negative set $\mathcal{I}^-$ is constructed using negative sampling which replaces either the subject s or object o with a random entity sampled uniformly from entity set $\mathcal{E}$ [18]. Entity and relation representations can be learned by minimizing the pairwise ranking loss

$$\mathcal{L}_{marg} = \sum_{\substack{\tau^+ \in \mathcal{I} \\ \tau^- \in \mathcal{I}^-}} \max\left(0, \gamma - f_r(s, o) + f_r\left(s', o'\right)\right) \tag{3.1}$$

to make the scores of positive instances higher than those of negative instances. The models do not assume that negative instances obtained by sampling are necessarily false, just that they are more invalid than the positive instances [53].

Logistic loss: According to Trouillon et al. [26], semantic matching models such as DistMult [23] and ComplEx [26] have yielded better results on logistic loss generally. Recently, the logistic loss has also been applied to CNN-based models, such as ConvKB [32]. The formulation of logistic loss is as:

$$\mathcal{L}_{logi} = \sum_{\substack{\tau^+ \in \mathcal{I} \\ \tau^- \in \mathcal{I}^-}} \log\left(1 + \exp\left(-l_{(s,r,o)} \cdot f_r(s, o)\right)\right) \tag{3.2}$$

, where $\tau = (s, r, o)$ is a training instance in $\mathcal{I}$ or $\mathcal{I}^-$, and $l_{(s,r,o)}$ is the label of τ. If $\tau \in \mathcal{I}$, $l_{(s,r,o)}$ is 1; otherwise it is -1. It has been shown that minimizing the logistic loss can help to find compact representations for some complex semantic properties, such as transitivity [54].

4. Ontology guided joint embedding framework

In this section, we introduce FOG, which is an ontology-guided joint embedding framework for learning both relational patterns and relational constraints. The main architecture of the FOG framework is shown in Figure 1. We then describe the majority modules of FOG. In Section 4.1, we describe the batching process that generates instances that reflect relational patterns and constraints based on ontology axioms. In Section 4.2, we illustrate the loss constructing module, which utilizes various loss functions according to instances from different batches, and the joint loss. Finally, in Section 4.3, we propose new generic evaluation metrics, which measure the reasoning capability of KG embedding models over relation properties.

4.1. Batching module

In the KG embedding scenario, only relation instances between different entities are considered; hence, the relations under consideration are already assumed to be irreflective. Additionally, a semantic property, such as equivalent to, can be handled by mapping two equivalent relations to the same embedding. Hence, axioms about reflective, irreflective, and equivalent to semantic properties do not require special consideration during KG embedding.

The other seven types of axioms about relation semantic properties can be divided into two sets based

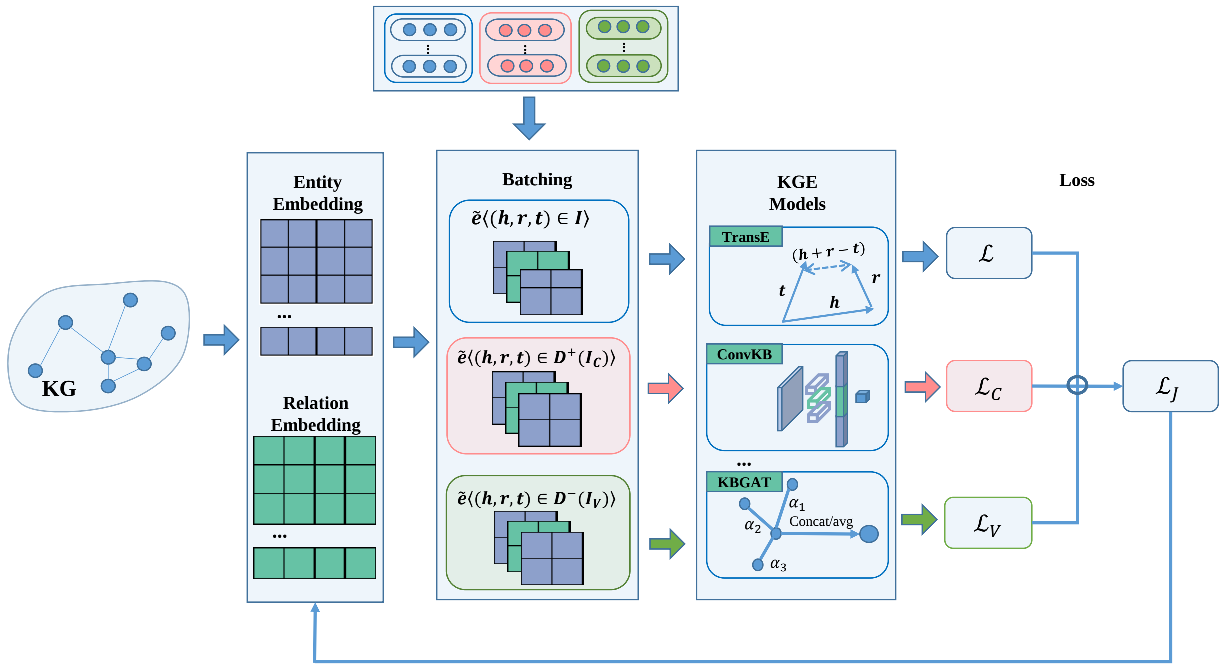


Fig. 1. Overall architecture of the ontology-guided KG embedding framework. Different from general KG embedding models, additional true positive and true negative instances derived beyond KG ($\mathcal{D}^{+}(\mathcal{I}_C)$ and $\mathcal{D}^{-}(\mathcal{I}_V)$) are introduced in the batching step. After that, the KG embedding methods generate a loss respectively according to the definition in Section 4.2. Finally a joint loss $\mathcal{L}_J$ is derived after the conjunction of three losses and back propagate for the optimization of the KG embedding.

Table 1
Notations and descriptions.

Notation	Description
$\mathcal{I}$	Knowledge Graph instances
$\mathcal{I}^{-}$	Instances not in KG
$\mathcal{I}_C$	Instances in KG related to the C-set
$\mathcal{I}_V$	Instances in KG related to the V-set
$\mathcal{D}^{+}(\mathcal{I}_C)$	True-positive instances derived from $\mathcal{I}_C$
$\mathcal{D}^{-}(\mathcal{I}_V)$	True-negative instances derived from $\mathcal{G}_V$

on the characteristics of the relation instances derived from those axioms. The *conformance set* (C-set) of axioms specify that some other instances should be treated as positive instances during training and prediction because they are semantic consequences of known triples (relation instances) in $\mathcal{I}$.

(1) According to the semantics of symmetry, both (s,r,o) and (o,r,s) should be annotated and predicted to reflect the symmetry property of relation r. For example, when a KG embedding model is fed an observed fact (Amy is_Married_To John), the model is expected to be equipped with the reasoning capability so that it will know that (John is_Married_To Amy) is also a true fact.

(2) According to the semantics of transitivity, transitive closure can be deduced for relation r based on the above rule, which is an instance set of r that is closed under transitivity. Similar to human common sense, the stronger the capability of the KG embedding model for reasoning transitive relations, the more multi-hop triples in the transitive closure can be inferred.

(3) According to the semantics of the inverse to property, relations that are inverse to each other can both generate positive instances. For example, the relations “nominations" and "nominees" are inverse to each other. Hence the model should be able to infer relation instances using this property by switching the subject and object entities.

(4) According to the semantics of subproperty of, two entities are predicted or known to have a relation that indicates that they also have the corresponding super-relations. For example, when a model predicts $(A, \text{isLocatedIn}, B)$, it should also infer $(A, \text{placedIn}, B)$ because *isLocatedIn* is a subproperty of *placedIn*.

Table 2

Semantic reasoning rules for relation property axioms.

	Relation Property	Semantic Reasoning Rule
Axioms in *C*-set	symmetry	if $(s, r, o) \in \mathcal{I}_C$, then $(o, r, s) \in \mathcal{D}^+(\mathcal{I}_C)$
	transitivity	if $(a, r, b), (b, r, c) \in \mathcal{I}_C$, then $(a, r, c) \in \mathcal{D}^+(\mathcal{I}_C)$
	inverse to	if $r = p^{-1}$ and $(s, r, o) \in \mathcal{I}_C$, then $(o, p, s) \in \mathcal{D}^+(\mathcal{I}_C)$
	subproperty of	if $r_1 \subseteq r_2$ and $(s, r_1, o) \in \mathcal{I}_C$, then $(s, r_2, o) \in \mathcal{D}^+(\mathcal{I}_C)$
Axioms in *V*-set	asymmetry	if $(s, r, o) \in \mathcal{I}_V$, then $(o, r, s) \in \mathcal{D}^-(\mathcal{I}_V)$
	functionality	if $b \neq c$ and $(a, r, b) \in \mathcal{I}_V$, then $(a, r, c) \in \mathcal{D}^-(\mathcal{I}_V)$
	inverse functionality	if $a \neq c$ and $(a, r, b) \in \mathcal{I}_V$, then $(c, r, b) \in \mathcal{D}^-(\mathcal{I}_V)$

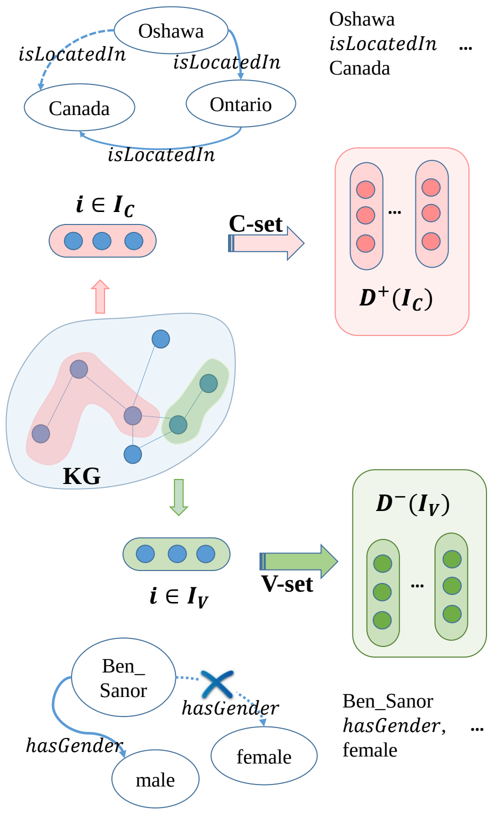


Fig. 2. The process of deriving $\mathcal{D}^+(\mathcal{I}_C)$ and $\mathcal{D}^-(\mathcal{I}_V)$ semantically based on the C-set and V-set axioms.

Hence, axioms related to semantic properties {*symmetry*, *transitivity*, *inverse to*, *subproperty of*} belong to C-set. The instances in $\mathcal{I}$ that are related to C-set are denoted by $\mathcal{I}_C$ and the positive instances derived from $\mathcal{I}_C$ based on axioms in C-set are denoted by $\mathcal{D}^+(\mathcal{I}_C)$.

The *violation set* (V-set) of axioms that specify some other instances should be treated as negative instances during training and prediction because they contradict some known triples in $\mathcal{I}$.

(1) According to the semantics of asymmetry, only one of the two relation instances of the symmetric form should be annotated and predicted to reflect the asymmetry property of relation r. For example, (man, created, account) is a ground-truth annotation in the dataset. Because “created” is an asymmetric relation, the symmetric form (account, created, man) cannot be an instance of the relation “created." Therefore, we expect that the model could avoid making the mistake of matching the symmetric forms of the ground-truth annotations for asymmetric relations when making predictions. The more predictions match the symmetric forms, the weaker the capability of the KG embedding model to capture the asymmetry property.
(2) According to the semantics of functionality, the functional property requires that for any given subject entity, there is only one object entity that constitutes a relation instance with the subject entity. It can be viewed as a constraint for the embedding of many-to-one relations.
(3) The inverse functionality property is similar to the functional property, except it requires that for any given object entity, there is only one subject entity that constitutes a relation instance with the subject entity. It can be viewed as a constraint for the embedding of one-to-many relations.

Hence, axioms related to semantic properties {*asymmetry*, *functionality*, *inverse functionality*} belong to V-set. The instances in $\mathcal{I}$ that are related to the V-set are denoted by $\mathcal{I}_V$ and the negative instances derived from $\mathcal{I}_V$ based on axioms in V-set are denoted by $\mathcal{D}^-(\mathcal{I}_V)$. All notations and descriptions are listed in Table 1.

Based on the above analysis, the semantic reasoning rules for deriving positive and negative relation instances based on all the relation semantic property axioms are defined in Table 2. For example, an axiom that specifies a relation *r* is symmetric belongs to C-set because, according to the semantics of symmetry, $(a, r, b) \in \mathcal{I}$ implies that its symmetric form (b, r, a) is a positive instance. An axiom that specifies that relation r is functional belongs to the V-set because, according to the semantics of functionality, $(a, r, b) \in \mathcal{I}$ implies that for any $c \neq b$, (a, r, c) is a negative instance.

Following the semantic reasoning rules for relation property axioms,new ground-truth triples can be derived from known ground-truth triples in the KG that match the C-set or V-set axioms [1]. The semantic reasoning process for constructing $\mathcal{D}^+(\mathcal{I}_C)$ and $\mathcal{D}^-(\mathcal{I}_V)$ from the original KG is shown in Figure 2. In fact, the semantic reasoning rules can be defined not only for the relation property axioms but also for other axioms defined in the ontology, such as the broadly used class assertion axioms and class hierarchy axioms [15, 45]. We will leave this for future works.

4.2. Loss constructing module

The basic idea used to incorporate semantic constraints specified in the ontology into the KG embedding models is to use the two sets of derived instances, $\mathcal{D}^+(\mathcal{I}_C)$ and $\mathcal{D}^-(\mathcal{I}_V)$, because the semantics of relation property axioms in the ontology can be reflected by the relation instances in the KG and the derived instances together.

To follow more closely the OWA used in KG construction, unobserved instances are not treated as negative instances, but just assumed to be not necessarily wrong. As a result, the relation instances used for training can be divided into three types: instances in both $\mathcal{I}$ and $\mathcal{D}^+(\mathcal{I}_C)$ as positive examples, instances in $\mathcal{D}^-(\mathcal{I}_V)$ as true negative examples, and instances in I- that are generated by negative sampling as possibly negative examples.

Because the original loss function already reflects the fitness of the model with $\mathcal{I}$ and $\mathcal{I}^-$, two loss functions are introduced, *conformance loss* $\mathcal{L}_C$ and *violation loss* $\mathcal{L}_V$, for measuring the fitness of the model with $\mathcal{D}^+(\mathcal{I}_C)$ and $\mathcal{D}^-(\mathcal{I}_V)$ respectively.

Conformance loss $\mathcal{L}_C$**.** According to the semantics of the axioms in the OWL2 RL [11] ontology, the semantic reasoning rules are always sound; that is, the derived instance set $\mathcal{D}^+(\mathcal{I}_C)$ contains only true positive instances with 100% confidence. Thus, similar to $\mathcal{L}_{marg}$, conformance loss $\mathcal{L}_C$ is defined by replacing $\mathcal{I}$ with $\mathcal{D}^+(\mathcal{I}_C)$ and $\mathcal{I}^-$ with $\emptyset$ in the pairwise ranking loss $\mathcal{L}_{marg}$. Given a positive instance $\tau^+ = (s, r, o) \in \mathcal{D}^+(\mathcal{I}_C)$ and negative instance $\tau^- = (s', r, o') \in \mathcal{I}^-$, the construction of $\mathcal{L}_C$ is as follows:

$$\mathcal{L}_C = \sum_{\substack{\tau^+ \in \mathcal{D}^+(\mathcal{I}_C) \\ \tau^- \in \mathcal{I}^-}} \max\left(0, \gamma - f_r(s, o) + f_r\left(s', o'\right)\right) \tag{4.1}$$

In the case of the logistic loss, the definition of $\mathcal{L}_C$ is similar to that of the pairwise ranking loss. We omit the details here.

Violation loss $\mathcal{L}_V$**.** Following the basic idea that the instances in $\mathcal{D}^-(\mathcal{I}_V)$ are "more negative" than the instances that are generated by negative sampling, the form of the loss functions for the pairwise ranking loss and logistic loss also need to be corrected slightly. In the case of the pairwise ranking loss, the violation loss $\mathcal{L}_V$ is defined as follows:

$$\mathcal{L}_V = \sum_{\substack{\tau^+ \in \mathcal{I}_V \\ \tau^- \in \mathcal{D}^-(\mathcal{I}_V)}} \max\left(0, \gamma' - f_r(s, o) + f_r\left(s', o'\right)\right) \tag{4.2}$$

$\mathcal{L}_V$ is defined by replacing the $\mathcal{I}$ with $\emptyset$ and $\mathcal{I}^-$ with $\mathcal{D}^-(\mathcal{I}_V)$ in $\mathcal{L}_{marg}$. Specifically, an additional hyperparameter γ' is introduced to replace the discriminative margin γ in $\mathcal{L}_\mathcal{V}$. γ' is set to be larger than γ to ensure that $\mathcal{D}^-(\mathcal{I}_V)$ is "more negative" by making the margin between $\mathcal{I}$ and $\mathcal{I}^-$ smaller than that between $\mathcal{I}$ and $\mathcal{D}^-(\mathcal{I}_V)$.

In the case of the logistic loss, another hyperparameter θ ($\theta > 1$) is introduced as the label of instances in $\mathcal{D}^-(\mathcal{I}_V)$ to make it "more negative" and the label function is corrected accordingly:

$$l_{(s,r,o)} = \begin{cases} 1, & \text{if } (s, r, o) \in \mathcal{I} \cup \mathcal{D}^+(\mathcal{I}_C) \\ -1, & \text{if } (s, r, o) \in \mathcal{I}^- \\ -\theta, & \text{if } (s, r, o) \in \mathcal{D}^-(\mathcal{I}_V) \end{cases} \tag{4.3}$$

Ontology Guided Joint Loss. The original loss is combined with the conformance loss and violation loss in the *ontology-guided joint loss function*, which is defined as follows:

$$\mathcal{L}_J = \mathcal{L} + \alpha_1 \cdot \mathcal{L}_C + \alpha_2 \cdot \mathcal{L}_V \tag{4.4}$$

[1] In practice, to avoid the problem of label leaking, $\mathcal{D}^+(\mathcal{I}_C)$ is a set which filtered out the testing triples.

where α_1 and α_2 are hyperparameters for tuning the weights between the three losses.

The above correction to the loss function makes the FOG framework conform more to the OWA by differentiating positive instances, negative instances, and potentially negative instances.

4.3. Generalization capability evaluation metrics.

We expect an effective KG embedding model to be capable of capturing relational properties as defined in Section 3.2. Recent work has proposed analyzing methods from the perspective of evaluating the model performance on various relation properties [55]. On the other hand, to evaluate the reasoning capability to infer implicit positive instances of the KG and avoid the derivation of true negative instances based on the relation property axioms, we propose two new evaluation metrics. We denote the instances in the test set that are related to the C-set and V-set axioms as T_C and T_V respectively. The true positive instances and true negative instances derived from them are denoted by $\mathcal{D}^+(T_C)$ and $\mathcal{D}^-(T_C)$ respectively.

Hit$_{\mathcal{D}^+(T_C)}$@k**:** We define the top-k Hit Ratio on $\mathcal{D}^+(T_C)$ as

$$\mathrm{Hit}_{\mathcal{D}^+(T_C)}@k = \frac{|\mathcal{D}^+(T_C)@k|}{|\mathcal{D}^+(T_C)|}, \tag{4.5}$$

where $|\mathcal{D}^+(T_C)|$ is the number of instances in $\mathcal{D}^+(T_C)$ and $|\mathcal{D}^+(T_C)@k|$ is the number of instances in $\mathcal{D}^+(T_C)$ that appear in the top-k rank list during prediction.

Hit$_{\mathcal{D}^-(T_V)}$@k**:** We define the top-k Hit Ratio on $\mathcal{D}^-(T_V)$ as

$$\mathrm{Hit}_{\mathcal{D}^-(T_V)}@k = \frac{|\mathcal{D}^-(T_V)@k|}{|\mathcal{D}^-(T_V)|}, \tag{4.6}$$

where $|\mathcal{D}^-(T_V)|$ is the number of instances in $\mathcal{D}^-(T_V)$ and $|\mathcal{D}^-(T_V)@k|$ is the number of instances in $\mathcal{D}^-(T_V)$ that appear in the top-k rank list during prediction.

According to the definition, $\mathrm{Hit}_{\mathcal{D}^+(T_C)}@k$ measures how capable the model is of deriving positive triples that are implicitly contained in the KG, and $\mathrm{Hit}_{\mathcal{D}^-(T_V)}@k$ measures how bad the model is at avoiding the derivation of instances that contradict the KG.

5. Experiments

We evaluated the FOG framework on two KG completion tasks: relation prediction and link prediction. In this section, we discuss the performance of the FOG enhanced models from an ontology-specific view. Additionally, we provide an axiom study and case study in Section 5.7 and Section 5.8, respectively, for relations with semantic properties.

5.1. Datasets

We evaluated the effectiveness of our FOG framework on the following three widely adopted benchmark datasets.

- FB15k-237 [56] contains 14,541 entities, 237 relations, and 272,115 training triples. . It is a modified version of FB15k that excludes inverse relations to resolve a flaw with FB15k [31].
- NELL-995 [57] contains 75,492 entities, 200 relations, and 149,678 training triples. The number of entities in NELL-995 is approximately five times that in FB15k-237, whereas the number of training triples in NELL-995 is only about half of that in FB15k-237, which means that NELL-995 is sparser.
- YAGO3-10 [58] is a subset of the YAGO3 dataset. It contains 123,182 entities, 37 relations, and 1,079,040 training triples. Each entity has a minimum of 10 relations. Test results on YAGO3-10 demonstrate the performance of models on large-scale datasets, to a certain extent.

The number of entities and relations, and the statistics of the train/validation/test set splits for the FB15k-237, NELL-995, and YAGO3-10 datasets are listed in Table 3.

Because not all relation semantic property axioms supported by OWL2 RL appeared in all three datasets, we analyzed the related relation semantic property axioms for each dataset. The relation semantic properties for the YAGO3-10 dataset include *{symmetry, transitivity, subproperty of, asymmetry, functionality}*, which we used for the training and testing of models with our framework. Because no ontology is defined for the FB15k-237 and NELL-995 datasets, we aligned the relations on FB15k-237 and NELL-995 with the ontology of YAGO3 manually. We also used the corresponding axioms in the YAGO3 ontology to construct an ontology for these two datasets. Note that WN18RR [31] is a dataset as commonly used as FB15k-237, but we did not conduct an experiment on it because WN18RR has no relation that can be aligned with YAGO3-10.

Table 3
Statistics of datasets. $\mathcal{I}_C$ and $\mathcal{I}_V$ denote the number of relation instances related to the C-set and V-set axioms respectively in the train set. $\mathcal{T}_C$ and $\mathcal{T}_V$ denote the number of relation instances related to the C-set and V-set axioms respectively in the test set.

			Triples						
	entities	relations	train	valid	test	$\mathcal{I}_C$	$\mathcal{I}_V$	$\mathcal{T}_C$	$\mathcal{T}_V$
FB15k-237	14,541	237	272,115	17,535	20,466	7,810	10,698	396	933
NELL-995	75,492	200	149,678	543	3,992	3,263	10,771	0	3,992
YAGO3-10	123,182	37	1,079,040	5,000	5,000	124,484	122,955	578	569

5.2. Evaluation protocol

We evaluated the effectiveness of the proposed ontology-guided joint embedding framework on the relation and link prediction tasks. The purpose of relation prediction is to predict the relation between two given entities, that is, infer r given (s, o), and link prediction aims to predict a missing entity given a relation and an entity, that is, infer s given (r, o) or infer o given (s, r). We evaluated all the models using the following two types of metrics.

Apart from the generalization capability evaluation metrics proposed in Section 4.3, we evaluated the KG embedding models using three common evaluation metrics: mean rank (MR), mean reciprocal rank (MRR), and Hit@k. Let N be the number of valid instances in the test set.

$$\mathrm{MR} = \frac{\sum_{i=1}^{N} r_i}{N} \tag{5.1}$$

$$\mathrm{MRR} = \frac{\sum_{i=1}^{N} (1/r_i)}{N} \tag{5.2}$$

$$\mathrm{Hit@}k = \frac{N@k}{N} \tag{5.3}$$

where r_i is the predicted rank of the ith instance in test set, and $N@k$ is the number of instances in test set that appear in the top-k rank list.

For each valid triples (s, r, o) in the test set, we replace r with every other relation in the dataset to create corrupted triples in the relation prediction task, and replace either s or o with every other entities in the dataset to create corrupted triples in the link prediction task. Following previous works [18, 27, 31–33], we evaluated all the models in a *filtered* setting, that is, we removed corrupt triples that appeared in the training, validation, or test sets during ranking. We ranked the valid triple and filtered corrupted triples in ascending order of their prediction scores.

It was recently proposed [59] that the position at which the correct triplet is inserted can substantially affect the prediction results since certain KG embedding models derive the same score for different triples. Therefore, we use the robust evaluation protocol proposed in [59] to minimize the evaluation bias.

We report Hit@1 for relation prediction and Hit@10 for link prediction because the number of relations in datasets is relatively smaller than that of entities. The lower the MR, higher the MRR, or higher the Hit@k, the better the performance.

5.3. Experimental Setup

We implemented all the baseline models and models enhanced with the ontology-guided joint embedding framework (i.e., FOG) in Python using TensorFlow 1.13.1 or PyTorch 1.1.0. We performed most of the experiments on a workstation with an Intel Xeon Gold 5118 2.30 GHz CPU and NVIDIA Tesla V100-SXM2 16 GB GPU. The exception was experiments for KBGAT on YAGO3-10, which we performed on a different workstation, with an NVIDIA Titan RTX 24 GB GPU.

To ensure that the implemented environment consisted of the FOG enhanced models, we reproduced the baseline models using publicly available source code. The overall deviation of the reproduction results was within $\pm 4\%$of the reported results in [31, 33]. We list all the hyperparameter settings for the baseline models in Appendix A.

For the newly introduced hyperparameters, we trained the FOG enhanced models using a grid search of the hyperparameters: weight tuning hyperparameters for $\mathcal{L}_C, \mathcal{L}_V, \alpha_1, \alpha_2 \in \{0.3, 0.5, 0.6, 1.0\}$, discriminative margin for true negative instances in pairwise ranking loss $\gamma' \in \{1, 1.5, 2.0\}$, label of true negative instances in logistic loss $\theta \in \{1.1, 1.2, 1.3, 1.5, 2.0\}$,

Table 4

Relation prediction results on FB15k-237, NELL-995 and YAGO3-10 ($k = 1$). We omit the $\mathcal{D}^+(T_C)$ column for the NELL-995 dataset, because $\mathcal{D}^+(T_C)$ is empty for NELL-995. The best score for each metric of each KG embedding model is listed in **bold**.

	FB15k-237			NELL-995		YAGO3-10		
	Hit@1	Hit$_{\mathcal{D}^+(T_C)}$@1	Hit$_{\mathcal{D}^-(T_V)}$@1	Hit@1	Hit$_{\mathcal{D}^-(T_V)}$@1	Hit@1	Hit$_{\mathcal{D}^+(T_C)}$@1	Hit$_{\mathcal{D}^-(T_V)}$@1
TransE (base)	80.97	50.21	1.18	48.07	0.00	84.62	31.84	1.58
TransE (aug)	82.26	68.13	0.43	46.42	0.00	**86.14**	**58.75**	2.64
OG-TransE	**82.36**	**71.43**	**0.21**	**50.13**	**0.00**	84.50	50.27	**1.41**
ConvKB (base)	88.66	52.66	**0.11**	67.79	0.00	62.98	20.88	3.52
ConvKB (aug)	88.98	72.70	0.32	67.51	0.00	59.26	86.90	**1.93**
OG-ConvKB	**91.70**	**83.77**	0.21	**71.39**	**0.00**	**68.72**	**97.28**	6.68
QuatE (base)	**91.67**	36.77	**0.11**	46.29	0.25	90.46	86.63	2.81
QuatE (aug)	91.66	**63.06**	0.32	46.79	0.33	90.10	98.66	1.58
OG-QuatE	87.68	54.35	0.43	**46.79**	**0.15**	**91.67**	**98.67**	**0.18**
KBGAT (base)	80.40	69.06	0.54	33.24	0.02	45.30	6.44	2.46
KBGAT (aug)	70.82	78.44	7.82	30.98	29.38	50.52	**47.95**	2.11
OG-KBGAT	**88.49**	**84.36**	**0.21**	**40.38**	**0.00**	**53.98**	13.90	**1.23**
On2Vec (base)	97.81	**98.31**	32.94	75.78	66.14	98.90	**99.78**	37.88
On2Vec (aug)	97.35	97.03	26.37	74.03	68.83	99.10	99.63	42.58
OG-On2Vec	**98.31**	95.34	**1.61**	**76.39**	**0.28**	**99.32**	99.65	**35.75**

Table 5

Link prediction results on FB15k-237, NELL-995, and YAGO3-10 on three common evaluation metrics. The results of ConvKB (base) and KBGAT (base) are taken from the revised version from [59] and others are the reproduced results with suggested hyper-parameters.

	FB15k-237			NELL-995			YAGO3-10		
	MR	MRR	Hit@10	MR	MRR	Hit@10	MR	MRR	Hit@10
TransE (base)	329.9	0.282	42.56	**6773.6**	0.224	38.06	1335.3	0.150	40.62
TransE (aug)	264.8	0.292	46.61	7235.3	0.222	38.70	**1264.7**	0.172	42.32
OG-TransE	**266.0**	**0.297**	**47.14**	6774.8	**0.226**	**38.77**	1366.8	**0.179**	**45.64**
ConvKB (base)	372.1	0.213	39.60	4788.7	0.347	48.18	6635.4	0.451	57.80
ConvKB (aug)	369.9	0.216	39.75	4944.9	0.344	48.06	**4574.8**	0.424	56.24
OG-ConvKB	**243.6**	**0.422**	**54.76**	**4052.7**	**0.370**	**48.68**	6238.1	**0.459**	**63.30**
QuatE (base)	95.6	0.350	53.92	166.8	0.468	58.72	381.7	0.450	65.78
QuatE (aug)	94.9	0.353	54.26	167.1	0.465	58.38	404.7	0.440	64.94
OG-QuatE	**75.3**	**0.408**	**59.80**	**163.1**	**0.493**	**62.65**	**264.3**	**0.542**	**74.70**
KBGAT (base)	360.1	0.167	31.89	3032.4	0.310	46.96	3815.9	0.075	13.88
KBGAT (aug)	378.7	0.157	30.43	3016.0	0.338	47.48	5829.8	0.073	13.04
OG-KBGAT	**266.2**	**0.189**	**36.90**	**1194.2**	**0.421**	**60.61**	**1018.1**	**0.253**	**46.83**

ratio of used derived true positive instances {0.1, 0.2, 0.5, 0.8}, and ratio of generating true negative instances {8, 10, 20, 40}.

We selected the model with the highest Hit@k value on the validation set to be evaluated on the test set. We list the best hyperparameter settings, in addition to the total number of epochs for the training process for the FOG enhanced models, in Appendix A.

5.4. Baseline models

We chose TransE, ConvKB, QuatE, and KBGAT as baseline models and applied FOG to them. We discovered problems with these baseline models, which may represent a common problem for current KG embedding models.

Inconsistent performance for link and relation prediction across datasets. The experimental results also demonstrated that some models performed differently on different datasets and prediction tasks, which

Table 6

Link prediction results on two reasoning capability evaluation metrics.

	FB15k-237		NELL-995	YAGO3-10	
	$\text{Hit}_{\mathcal{D}^+(T_C)}$@10	$\text{Hit}_{\mathcal{D}^-(T_V)}$@10	$\text{Hit}_{\mathcal{D}^-(T_V)}$@10	$\text{Hit}_{\mathcal{D}^+(T_C)}$@10	$\text{Hit}_{\mathcal{D}^-(T_V)}$@10
TransE (base)	30.09	**0.00**	4.94	16.38	0.00
TransE (aug)	**57.69**	0.05	5.30	**71.93**	0.00
OG-TransE	57.10	0.05	**4.17**	65.15	**0.00**
ConvKB (base)	24.85	0.16	1.04	18.47	60.28
ConvKB (aug)	56.81	0.16	1.05	43.65	60.19
OG-ConvKB	**57.44**	**0.16**	**0.36**	**62.28**	**55.36**
QuatE (base)	23.71	1.39	35.88	19.15	**0.53**
QuatE (aug)	**52.07**	1.66	36.76	77.11	0.97
OG-QuatE	49.45	**0.64**	**15.98**	**78.13**	0.70
KBGAT (base)	22.61	0.05	16.87	10.09	0.00
KBGAT (aug)	40.88	0.02	43.33	**73.44**	0.00
OG-KBGAT	**36.73**	**0.05**	**13.98**	16.42	**0.00**

illustrates that a lack of generalizability across tasks and datasets is still a problem for current KG embedding models. For instance, the performance of KBGAT was significantly different on the relation and link prediction tasks; that is, it outperformed other models for link prediction, whereas its performance for the relation prediction task was relatively low in comparison to other models. The construction of KG embedding models that have consistently good performance on both tasks on different datasets is still a challenge.

Unstable performance for certain types of models. By conducting training and testing repeatedly on the relation and link prediction tasks, we found that CNN-based models, such as ConvKB and KB-GAT (which uses ConvKB as the decoder), demonstrated unstable performance each time. A possible reason is that CNN-based models are more sensitive to the initial value of the embedding vectors, which may cause the models to find a local optimal solution instead of a global optimal solution. For example, the reported link prediction result for KBGAT on FB15k-237 was 62.6% for Hit@10.After reproducing ten times on FB15k-237 with the suggested best parameters [33], we obtained an average Hit@10 result of 59.13% with a standard deviation of 1.5%.

5.5. Result and Discussion

In this section, we verify the effectiveness of our ontology-guided joint embedding framework by evaluating the performance of the baseline models without/with the FOG framework on both the relation and link prediction tasks.

We considered four representative models as our baseline models: TransE [18], ConvKB [32], QuatE [27], KBGAT [33], and On2Vec[2] [15].We represent the baseline models as [name](base). We also deployed variants of baselines for the ablation study: for data augmentation, we also used triples inferred from the training set based on axioms in C-set for training. We represent the augmented baselines as [name](aug). We represent the models enhanced by the FOG framework as FOG-[name]. The experimental results for relation prediction are presented in Table 4. The results for link prediction are presented in Table 5 and Table 6.

Improvement on the common evaluation metrics. . The common evaluation metrics (MR, MRR, Hit@k) measure the general prediction capability of models. By analyzing results for both the relation and link prediction tasks on three benchmark datasets, we found that models with our ontology-guided joint embedding framework performed better in most cases for the common evaluation metrics. For instance, for the Hit@k metrics, models with our framework gained a 0.5% to 15% improvement on most datasets, with only a few exceptions, which indicates that the ontology-guided joint embedding framework did not damage the original advantages of the baseline models, and in most cases could enhance the performance of different KG embedding models.

[2]We will not show the link prediction results of On2Vec since it is specially designed for relation prediction and has a very low performance on link prediction.

Improvement on the reasoning capability evaluation metrics. The proposed reasoning capability evaluation metrics measure the effectiveness of the ontology-guided joint embedding framework for inferring hidden relation instances in the KG and avoiding deriving relation instances that contradict the KG. The experimental results on both the relation and link prediction tasks demonstrated that there were significant improvements to the reasoning capability evaluation metrics for the models with the FOG framework. According to the definitions of the reasoning capability evaluation metrics in Section 4.3, the higher the value of $\text{Hit}_{\mathcal{D}^+(T_C)}@k$ and the lower the value of $\text{Hit}_{\mathcal{D}^-(T_V)}@k$, the better the reasoning performance. From Table 4 and 6 the best $\text{Hit}_{\mathcal{D}^+(T_C)}@k$ and $\text{Hit}_{\mathcal{D}^-(T_V)}@k$ achieved a 295% and 100% improvement, respectively. This indicates that our framework improved the reasoning capability of the KG embedding models.

The experimental results also demonstrated that the reasoning capability of the models exhibited different patterns on different datasets. For instance, all the models, except On2Vec, had a very low hit ratio for miss-predictions for true negative instances ($\text{Hit}_{\mathcal{D}^-(T_V)}@k$) on the FB15k-237 dataset, whereas many models had a high $\text{Hit}_{\mathcal{D}^-(T_V)}@k$ on the NELL-995 dataset. The models with our framework performed particularly better on datasets such as NELL-995. The $\text{Hit}_{\mathcal{D}^-(T_V)}@k$ values for On2Vec and KBGAT improved significantly with our framework. The hit ratio of miss-predictions reduced by 100% and 99.5%, respectively, for KBGAT and On2Vec for relation prediction, and 88.6% for KBGAT for link prediction. A possible reason is that sparse graphs provide fewer facts and thus rely more on the reasoning capability of models.

Generally, the experimental results demonstrated that the FOG framework improved the reasoning capability of the KG embedding models, while achieving performance on the common evaluation metrics that was the same as or better than that of the original model. To further explore the effectiveness of the FOG framework, we conducted an ablation study, axiom study, and case study.

5.6. Ablation study

We conducted an ablation study using the same $\mathcal{D}^+(\mathcal{I}_C)$ derived in the FOG enhanced models as augmentation data for training. The results demonstrated that the augmentation results were higher than those of baseline models overall, but lower than those of the models with the FOG framework. Even though some models without the FOG framework had a higher $\text{Hit}_{\mathcal{D}^+(T_C)}@10$ value in some cases, they typically also had a higher $\text{Hit}_{\mathcal{D}^-(T_V)}@10$, which indicates that they had a higher miss-prediction ratio; that is, more true negative instances were predicted as positive instances by mistake. This indicates that simply augmenting the models with triples derived from the KG cannot equip the KG embedding models with sufficient capability to fully learn the axiom constraints from the ontology. The ablation study demonstrated that the proposed KG framework is a generic approach for enhancing the KG embedding model with better performance and reasoning capability.

5.7. Axiom Study

We further discuss the interpretability of axioms, that is, whether axioms can be used to explain KG embedding.

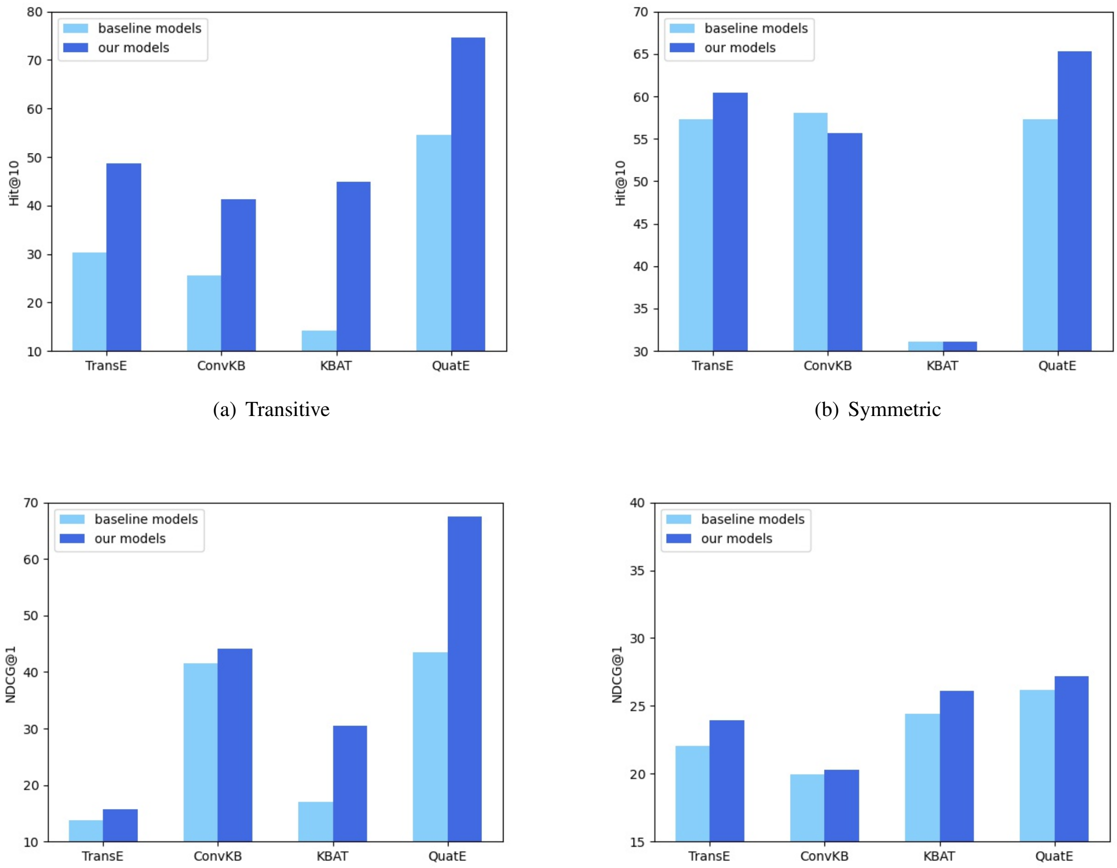
We split the triples in T_C according to the specific semantic property axiom and conducted an axiom study on each set. Specifically, we used Hit@10 as the evaluation metric because, for triples containing relations related to a semantic property axiom in C-set, the KG embedding model should be able to predict both the original triples and the triples derived from them according to the C-set axioms if the constraint specified by the axiom is learned by the model.

For triples in T_V, we also conducted an axiom study on each set that was split using relation properties. We used the common evaluation metric NDCG. NDCG measures the discounted cumulative gain of the actual returned result. For relations in V-set, we expect that the correct entity should not only be hit but also separated from the false entities as far as possible; that is, we care about the specific ranking of the missing entities. We intuitively consider that the correlation of the correct entity is 1 and that of the wrong entities is 0, and hence, the NDCG indicator can be written as

$$\text{NDCG} = \frac{1}{\log_2(i+1)},$$

where i denotes the rank of the missing entity.

The axiom study result is shown in n Figure 3. The histograms directly show that the improvement for a specific axiom was rather significant for some models, such as TransE and ConvKB for the transitive property and QuatE for the transitive, symmetric, and func-



(a) Transitive

(b) Symmetric

(c) Functional

(d) Asymmetric

Fig. 3. Histograms of axiom study results. We choose the results of FB15k-237 for symmetric and asymmetric relations. Since there is no transitive and functional relations in FB15k-237, we use the results of YAGO3-10 instead for and transitive and functional relations. For each pair of results on a model, the left one indicates the Hit@10 (NDCG@1) value of the baseline model, while the right one indicates the value of the model with the FOG framework.

tional properties. The axiom study results indicate that models with the FOG framework learned the relation semantic property axioms better.

5.8. Case Study

In this section, we provide a case study for the link prediction task to demonstrate the results of the FOG enhanced models.

Table 7 shows the top-5 prediction results for the FOG enhanced models. Among the queries, *isMarriedTo* and *isLocatedIn* were relations specified by an axiom in C-set. The first two queries were head and tail predictions for the same triple with symmetric relation *isMarriedTo*. The results demonstrated that the top-1 predictions were exactly entities of the marriage couples.

To simply demonstrate the performance of the model for the transitive relation, we performed leave-one-out filtering only on the training/validation/test sets to determine whether the top predictions made sense. As the results demonstrated, the top-5 predictions were all true positive triples derived from FOG, which demonstrated that the model did allow for some generalization and inference over the transitive relations. Meanwhile, relation *diedIn* and *hasCapital* were functional and inverse functional relations specified by axioms in V-set. The results demonstrated that the top-1 prediction not only accurately hit the unique ground truth but also had a significantly higher predic-

Table 7
Case study on link prediction task for FOG enhanced models. For each query, we list the top-5 results ranked by prediction scores. Valid predictions are marked in **bold**.

Query	Top 5 triples with highest score
(Henry_VII_of_England, isMarriedTo, ?)	**Elizabeth_of_York**(1.71)
	Mary_Tudor,_Queen_of_France(1.56)
	Catherine_of_Valois(0.99)
	Margaret_Tudor(0.83)
	Beatrice_of_England(0.72)
(?, isMarriedTo, Elizabeth_of_York)	**Henry_VII_of_England**(1.71)
	Elizabeth_Woodville (0.97)
	Edward_IV_of_England (0.94)
	Henry_VIII_of_England(0.80)
	Margaret_Tudor(0.63)
(?, isLocatedIn, Pilsen)	**Plzeň-South_District**(1.78)
	Klatovy_District(1.59)
	Domažlice_District(1.45)
	Rokycany(1.29)
	Plzeň-City_District(1.24)
(Aleksei_Yuryevich_German, diedIn, ?)	**Saint_Petersburg**(0.61)
	Moscow (-1.30)
	Minsk (-1.78)
	Riga (-2.06)
	Samara,_Russia (-2.24)
(?, hasCapital, Kandy)	**Central_Province_Sri_Lanka**(1.52)
	Northern_Province,_Sri_Lanka (0.47)
	Central_Province,_Sri_Lanka(0.45)
	North_Central_Province,_Sri_Lanka (0.24)
	Kandy_District (-0.24)

tion score than the left, which means that the FOG enhanced model was capable of capturing the semantic property of functionality and inverse functionality.

6. Conclusion

In this paper, we proposed an ontology-guided joint embedding framework that can be easily used with existing KG embedding models. The basic idea of this framework is to incorporate the semantic rules and constraints specified in the ontology through a joint loss function with two losses defined on the derived positive and negative instance sets under the OWA. We also proposed two reasoning capability evaluation metrics to evaluate the capability of the model to predict positive triples whose derivation requires complex reasoning and avoid mistaken predictions of true negative triples. The experimental results demonstrated that models with our ontology-guided joint embedding framework generally performed better than their corresponding baseline models on most evaluation metrics for different tasks and datasets, which indicated the effectiveness of the framework for models with different structures and designed for different tasks.

For future work, we will consider more complete types of ontology axioms to enhance our framework. Meanwhile, we will also build an ontology-guided KG embedding framework that includes typical KG embedding methods to equip them with better reasoning capability.

Acknowledgement

This work was supported by National Natural Science Foundation of China (Grand No. 61502022).

Table A.1

Hyper-parameter settings of baseline models. The dimension of embeddings k is set to be 100 except training TransE and QuatE with YAGO3-10. k is set to be 500 under these circumstances. L^2-norm regularization is applied for all baseline models. lr is the learning rate, γ is the discriminative margin in pairwise ranking loss, and the three numbers in batch_size separated by a slash are the batch_size for FB15k-237, NELL-995, and YAGO3-10 respectively. Since KBGAT uses a Graph Attention Network (GAT) as encoder and a CNN as decoder, we denote the two networks as KBGAT-GAT and KBGAT-conv respectively.

Model	batch_size	lr	γ	others
TransE	512/512/1,024	0.001	1.0	-
ConvKB	128/128/128	0.000005	-	filter_sizes = 1, num_filters = 50, dropout_keep_prob = 1, useConstantInit = True (useConstantInit = False for NELL-995)
QuatE	27,211/14,967/1,024	0.1	1.0	λ_1 = 0.3, λ_2 = 0.3, neg = 10, (λ_1 = 0.1, λ_2 = 0.1 for NELL-995)
On2Vec	500/500/500	0.001	1.0	$\alpha_1 = 0$, $\alpha_2 = 0.5$
KBGAT-GAT	272,115/149,678/107,904	0.001	1.0	neg_s_gat = 2, drop$_{GAT}$ = 0.3, α = 0.2, out_dim = [100, 200] weight_decay_gat = 0.00001, partial_2hop = True, nheads = [2, 2]
KBGAT-conv	128/128/128	0.001	-	α_{conv} = 0.2, neg_s_conv = 40, out_channels = 50, drop$_{\text{conv}}$ = 0.3

Table A.2

Hyper-parameters settings for models with our FOG mechanism in FB15k-237.

	FOG-TranE	FOG-ConvKB	FOG-QuatE	FOG-On2Vec	FOG-KBGAT(GAT)	FOG-KBGAT(conv)
α_1	1	0.5	8	0.6	1	1
α_2	1	0.5	8	0.6	1	1
γ'	1.5	-	-	1.5	1.5	-
θ	-	1.2	1.2	-	-	1.2

Table A.3

Hyper-parameters settings for models with our FOG mechanism in NELL-995.

	FOG-TranE	FOG-ConvKB	FOG-QuatE	FOG-On2Vec	FOG-KBGAT(GAT)	FOG-KBGAT(conv)
α_1	1	1	1	0.6	1	1
α_2	1	5	1	0.6	1	1
γ'	1.5	-	-	1.5	1.5	-
θ	-	1.2	1.2	-	-	1.2

Appendix A. Hyper-parameter settings

Hyper-parameters of the baseline models. The hyper-parameter settings of baseline models are listed in Table A.1.

During the testing of baseline models, we found the results of KBGAT are not stable on both relation prediction and link prediction tasks, and the results of ConKB are not stable on relation prediction task. For these unstable models, we repeated the training and testing for 10 times and reported the average values for each metrics. For other models, we trained for 3 times and reported the latest result.

Hyper-parameters of the FOG enhanced models. The hyper-parameter settings as well as the number of epoch implemented on our FOG enhanced models are listed in Table A.2, A.3, A.4, and A.5.

Table A.4
Hyper-parameters settings for models with our FOG mechanism in YAGO3-10.

	FOG-TranE	FOG-ConvKB	FOG-QuatE	FOG-On2Vec	FOG-KBGAT(GAT)	FOG-KBGAT(conv)
α_1	1	1	1	0.6	1	1
α_2	1	1	1	0.6	1	1
γ'	1.5	-	-	1.5	1.5	-
θ	-	1.2	1.2	-	-	1.2
$ratio_{rp}$	57%	3.4%	3.4%	10%	23%	23%
$ratio_{lp}$	57%	3.4%	3.4%	-	3.4%	3.4%

Table A.5
Number of epoch for the training of the FOG enhanced models.

	FOG-TranE	FOG-ConvKB	FOG-QuatE	FOG-On2Vec	FOG-KBGAT(GAT/conv)
FB15k-237	800	200	5000	1000	3000/150
NELL-995	800	200	5000	1200	3000/150
YAGO3-10	4000	4000	5000	3000	3000/500

Appendix B. Experimental results on efficiency

Time cost of training and testing We list the average training time per epoch and the average prediction time per triple on FB15k-237 for all baseline models and enhanced models in Table A.6.

Table A.6
Average training/predicting time per epoch/triple on FB15k-237 for baseline models and FOG enhanced models.

	T(train)	T(test$_{lp}$)	T(test$_{rp}$)
TransE	1.35s	3.13ms	0.62ms
FOG-TransE	1.31s	3.13ms	0.60ms
ConvKB	3.34s	0.10s	0.84ms
FOG-ConvKB	4.78s	0.12s	0.90ms
QuatE	1.54s	3.67ms	1.12ms
FOG-QuatE	1.81s	3.68ms	1.13ms
On2Vec	7.55s	-	0.37ms
FOG-On2vec	10.92s	-	0.39ms
KBGAT	0.89s/43.12s	15.26ms	1.85ms
FOG-KBGAT	1.46s/45.77s	15.33ms	2.16ms

T(test$_{lp}$) and T(test$_{rp}$) denote the prediction time of link prediction and relation prediction respectively. The two results of T(train) for KBGAT (FOG-KBGAT) separated by a slash are the training time of the encoder and decoder respectively.

Appendix. References

[1] R.S. M. Mintz S. Bills and D. Jurafsky, Distant supervision for relation extraction without labeled data, in: *ACL and IJCNLP of the AFNLP*, 2009, pp. 1003–1011. https://www.aclweb.org/anthology/P09-1113.

[2] N. Ghasemi, R. Fatourechi and S. Momtazi, User Embedding for Expert Finding in Community Question Answering, *ACM Trans. Knowl. Discov. Data* **15**(4) (2021). doi:10.1145/3441302.

[3] K.L. S. He C. Liu and J. Zhao, Generating natural answers by incorporating copying and retrieving mechanisms in sequence-tosequence learning, in: *ACL*, 2017, pp. 199–208. https://www.aclweb.org/anthology/P17-1019.

[4] K.S. Dennis Diefenbach and P. Maret, Wdaqua-core1: a question answering service for rdf knowledge bases, in: *Companion of the The Web Conference 2018 on The Web Conference (WWW)*, International World Wide Web Conferences Steering Committee, 2018, pp. 1087–1091. doi:10.1145/3184558.3191541.

[5] Y.L.M.L. R. Koncel-Kedziorski D. Bekal and H. Hajishirzi, Text generation from knowledge graphs with graph transformers, in: *NAACL*, 2019, pp. 2284–2293. https://www.aclweb.org/anthology/N19-1238.

[6] Z. Wang, C. Chen and W. Li, Information Diffusion Prediction with Network Regularized Role-Based User Representation Learning, *ACM Trans. Knowl. Discov. Data* **13**(3) (2019). doi:10.1145/3314106.

[7] Z. Zhang, L. Zhang, D. Yang and L. Yang, KRAN: Knowledge Refining Attention Network for Recommendation, *ACM Trans. Knowl. Discov. Data* **16**(2) (2021). doi:10.1145/3470783.

[8] Y. Liu, B. Guo, D. Zhang, D. Zeghlache, J. Chen, K. Hu, S. Zhang, D. Zhou and Z. Yu, Knowledge Transfer with Weighted Adversarial Network for Cold-Start Store Site Recommendation, *ACM Trans. Knowl. Discov. Data* **15**(3) (2021). doi:10.1145/3442203.

[9] Z. Sun, Z.-H. Deng, J.-Y. Nie and J. Tang, RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space, in: *International Conference on Learning Representations*, 2019, pp. 1–18.

[10] T. Song, J. Luo and L. Huang, Rot-Pro: Modeling Transitivity by Projection in Knowledge Graph Embedding, in: *Proceedings of the Thirty-Fifth Annual Conference on Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[11] I.H.Z.W.A.F. B. Motik B. C. Grau and C. Lutz, OWL 2 Web Ontology Language Profiles (Second Edition), *W3C Recommendation 11 December 2012* (2012).

[12] P.P.T.S. K. Bollacker C. Evans and J. Taylor, Freebase: a collaboratively created graph database for structuring human knowledge, in: *SIGMOD*, 2008, pp. 1247–1250.

[13] G.K. F. M. Suchanek and G. Weikum, Yago: a core of semantic knowledge, in: *WWW*, 2007, pp. 697–706.

[14] G.K.J.L.R.C. S. Auer C. Bizer and Z. Ives, Dbpedia: A nucleus for a web of open data, in: *The semantic web*, 2007, pp. 722–735.

[15] M. Chen, Y. Tian, X. Chen, Z. Xue and C. Zaniolo, On2Vec: Embedding-based Relation Prediction for Ontology Population, *Proceedings of the 2018 SIAM International Conference on Data Mining* (2018), 315–323.

[16] R. Reiter, Deductive Question-Answering on Relational Data Bases, in: *Logic and Data Bases*, H. Gallaire and J. Minker, eds, Springer US, Boston, MA, 1978, pp. 149–177. doi:10.1007/978-1-4684-3384-5_6.

[17] S. Ji, S. Pan, E. Cambria, P. Marttinen and P.S. Yu, A Survey on Knowledge Graphs: Representation, Acquisition and Applications, *ArXiv* **abs/2002.00388** (2020).

[18] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko, Translating Embeddings for Modeling Multi-relational Data, in: *Advances in Neural Information Processing Systems 26*, 2013, pp. 2787–2795.

[19] Z. Wang, J. Zhang, J. Feng and Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: *AAAI Conference on Artificial Intelligence*, 2014, pp. 1112–1119.

[20] M.S.Y.L. Yankai Lin Zhiyuan Liu and X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 2015, pp. 2181–2187.

[21] Z. Tan, X. Zhao, Y. Fang and W. Xiao, GTrans: Generic Knowledge Graph Embedding via Multi-State Entities and Dynamic Relation Spaces, *IEEE Access* **6** (2018), 8232–8244. doi:10.1109/ACCESS.2018.2797876.

[22] L. Chao, J. He, T. Wang and W. Chu, PairRE: Knowledge Graph Embeddings via Paired Relation Vectors, in: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, Online, 2021, pp. 4360–4369. https://aclanthology.org/2021.acl-long.336.

[23] X.H.J.G. B. Yang W.-t. Yih and L. Deng, Embedding entities and relations for learning and inference in knowledge bases, in: *ICLR*, 2015, pp. 1–13.

[24] L.R. M. Nickel and T. Poggio, Holographic embeddings of knowledge graphs, in: *Proceedings of AAAI conference on artificial intelligence*, 2016, pp. 1955—1961.

[25] C. Xu and R. Li, Relation embedding with dihedral group in knowledge graph, in: *Proceedings of ACL*, Association for Computational Linguistics, 2019, pp. 263–272. doi:10.18653/v1/P19-1026. https://www.aclweb.org/anthology/P19-1026.

[26] S.R.E.G. T. Trouillon J. Welbl and G. Bouchard, Complex embeddings for simple link prediction, in: *Proceedings of 33rd Int. Conf. Mach. Learn*, 2016, pp. 2071–2080.

[27] S. Zhang, Y. Tay, L. Yao and Q. Liu, Quaternion Knowledge Graph Embedding, *Advances in Neural Information Processing Systems* **32** (2019), 2731—2741. https://proceedings.neurips.cc/paper/2019/file/d961e9f236177d65d21100592edb0769-Paper.pdf.

[28] J.W. A. Bordes X. Glorot and Y. Bengio, A semantic matching energy function for learning with multi-relational data, in: *Machine Learning*, Vol. 94, 2014, pp. 233—259. doi:10.1007/s10994-013-5363-6.

[29] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun and W. Zhang, Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion, in: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 601—610.

[30] R. Socher, D. Chen, C.D. Manning and A.Y. Ng, Reasoning with Neural Tensor Networks for Knowledge Base Completion, in: *Proceedings of the 26th International Conference on Neural Information Processing Systems*, Vol. 1, 2013, pp. 926—934.

[31] P.S. Tim Dettmers Pasquale Minervini and S. Riedel, Convolutional 2D Knowledge Graph Embeddings, *Proceedings of the 32nd AAAI Conference on Artificial Intelligence* (2018).

[32] D.Q.N. Dai Quoc Nguyen Tu Dinh Nguyen and D. Phung, 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 (NAACL)*, Vol. 2, 2018, pp. 327–333.

[33] D. Nathani, J. Chauhan, C. Sharma and M. Kaul, Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs, in: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2019, pp. 4710–4723. doi:10.18653/v1/P19-1466. https://www.aclweb.org/anthology/P19-1466.

[34] P.B.R.V.D.B.I.T. M. Schlichtkrull T. N. Kipf and M. Welling, Modeling relational data with graph convolutional networks, in: *ESWC*, 2018, pp. 593–607–.

[35] T. Kipf and M. Welling, Semi-Supervised Classification with Graph Convolutional Networks, in: *Proceedings of ICLR*, 2017, pp. 1–14.

[36] R. Wang, B. Li, S. Hu, W. Du and M. Zhang, Knowledge Graph Embedding via Graph Attenuated Attention Networks, *IEEE Access* **8** (2020), 5212–5224. doi:10.1109/ACCESS.2019.2963367.

[37] Q. Wang, B. Wang and L. Guo, Knowledge Base Completion Using Embeddings and Rules, in: *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, AAAI Press, 2015, pp. 1859–1865–. ISBN 9781577357384.

[38] S. Guo, Q. Wang, L. Wang, B. Wang and L. Guo, Jointly Embedding Knowledge Graphs and Logical Rules, in: *Proceedings of the 2016 Conference on Empiri-*

cal Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 192–202. doi:10.18653/v1/D16-1019. https://aclanthology.org/D16-1019.

[39] Z.J.Q.G.G.H. Boutouhami Khaoula, Uncertain Ontology-Aware Knowledge Graph Embeddings, in: *Semantic Technology*, 2020, pp. 129–136.

[40] W. Zhang, B. Paudel, L. Wang, J. Chen, H. Zhu, W. Zhang, A. Bernstein and H. Chen, Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning, 2019.

[41] R..T.T..H.Z..I.R.. Li Y.; Zheng and K. Sycara, Joint Embedding of Hierarchical Categories and Entities for Concept Categorization and Dataless Classification, in: *Proceedings of COLING*, 2016, pp. 2678–2688. https://www.aclweb.org/anthology/C16-1252.

[42] J. Hao, M. Chen, W. Yu, Y. Sun and W. Wang, Universal Representation Learning of Knowledge Bases by Jointly Embedding Instances and Ontological Concepts, in: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, 2019, pp. 1709–1719–.

[43] R. Xie, Z. Liu and M. Sun, Representation Learning of Knowledge Graphs with Hierarchical Types, in: *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 2016, pp. 2965—2971.

[44] M. Li, D. Zhang, Y. Jia, Y. Wang and X. Cheng, Link Prediction in Knowledge Graphs: A Hierarchy-Constrained Approach, *IEEE Transactions on Big Data* (2018).

[45] Z. Zhang, J. Cai, Y. Zhang and J. Wang, Learning Hierarchy-Aware Knowledge Graph Embeddings for Link Prediction, 2019.

[46] M. Nickel and D. Kiela, Poincaré Embeddings for Learning Hierarchical Representations, in: *Advances in Neural Information Processing Systems*, Vol. 30, I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan and R. Garnett, eds, Curran Associates, Inc., 2017. https://proceedings.neurips.cc/paper/2017/file/59dfa2df42d9e3d41f5b02bfc32229dd-Paper.pdf.

[47] I. Chami, A. Wolf, D.-C. Juan, F. Sala, S. Ravi and C. Ré, Low-Dimensional Hyperbolic Knowledge Graph Embeddings, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, 2020, pp. 6901–6914.

[48] J. Luo, Y. Wang and D. Jiang, Rule-based hidden relation recognition for large scale knowledge graphs, *Pattern Recognition Letters* **125** (2019), 13–20.

[49] P.F.P.-S. B. Motik and B. Parsia, OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), *W3C Recommendation 11 December 2012* (2012).

[50] I.H.Z.W.A.F.C.L. Boris Motik Bernardo Cuenca Grau, OWL 2 Web Ontology Language Profiles (Second Edition), *W3C Recommendation 11 December 2012* (2012).

[51] L.X.K.L. Guoliang Ji Shizhu He and J. Zhao, Knowledge graph embedding via dynamic mapping matrix, in: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, 2015, pp. 687–696. doi:10.3115/v1/P15-1067. https://www.aclweb.org/anthology/P15-1067.

[52] L. Cai and W.Y. Wang, KBGAN: Adversarial learning for knowledge graph embeddings, in: *NAACL*, 2018, pp. 1470–1480. doi:10.18653/v1/N18-1133.

[53] V.T. M. Nickel K. Murphy and E. Gabrilovich, A review of relational machine learning for knowledge graphs, in: *Proceedings of IEEE*, Vol. 104, 2016, pp. 11–33.

[54] S.S. G. Bouchard and T. Trouillon, On approximate rea- soning capabilities of low-rank vector spacess, in: *Proceedings of AAAI Spring Symp. Knowl. Representation Reasoning*, 2015, pp. 1–4. doi:10.1561/2400000009.

[55] A. Rossi, D. Barbosa, D. Firmani, A. Matinata and P. Merialdo, Knowledge Graph Embedding for Link Prediction: A Comparative Analysis, *ACM Trans. Knowl. Discov. Data* **15**(2) (2021). doi:10.1145/3424672.

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

[57] T.H. Wenhan Xiong and W.Y. Wang, Deeppath: A reinforcement learning method for knowledge graph reasoning, *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2017). https://www.aclweb.org/anthology/D17-1060.

[58] B.J. Mahdisoltani F. and F.M. Suchanek, YAGO3: A Knowledge Base from Multilingual Wikipedias, *Proceedings of CIDR 2015* (2015).

[59] Z. Sun, S. Vashishth, S. Sanyal, P. Talukdar and Y. Yang, A Re-evaluation of Knowledge Graph Completion Methods, in: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, 2020, pp. 5516–5522. doi:10.18653/v1/2020.acl-main.489. https://aclanthology.org/2020.acl-main.489.

[60] F.M. Suchanek, G. Kasneci and G. Weikum, Yago: A Core of Semantic Knowledge, in: *Proceedings of the 16th International Conference on World Wide Web*, 2007, pp. 697–706.

[61] S. Kok and P. Domingos, Statistical predicate invention, *roceedings of the 24th International Conference on Machine Learning, (ICML)* (2007).

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

[63] A. Maedche and S. Staab, Learning Ontologies for the Semantic Web (2001).

[64] N. Zhang, S. Deng, Z. Sun, G. Wang, X. Chen, W. Zhang and H. Chen, Long-tail Relation Extraction via Knowledge Graph Embeddings and Graph Convolution Networks, *NAACL* (2019), 3016–3025.

[65] S.R. L. Drumond and L. Schmidt-Thieme, Predicting RDF triples in incomplete knowledge bases with tensor factorization, in: *Proc. 27th Annu. ACM Symp. Appl. Comput.*, Association for Computing Machinery, New York, NY, USA, 2012, pp. 326–331. doi:10.1145/2245276.2245341.

[66] V.T. M. Nickel K. Murphy and E. Gabrilovich, A review of relational machine learning for knowledge graphs, in: *Proceedings of IEEE*, Vol. 104, 2015, pp. 11–33.

[67] B.K.B.S.E.R.H.J. A. Carlson J. Betteridge and T.M. Mitchell, Toward an Architecture for Never-Ending Language Learning, in: *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence*, 2010, pp. 1306–1313.

[68] L. Chen, G. Zeng, Q. Zhang, X. Chen and D. Wu, Question answering over knowledgebase with attention-based LSTM networks and knowledge embeddings, in: *2017 IEEE 16th International Conference on Cognitive Informatics and Cognitive Computing (ICCI*CC)*, 2017, pp. 243–246.

[69] M.Z.W.L.X.X. H. Wang F. Zhang and M. Guo, Multi-task feature learning for knowledge graph enhanced recommendation, in: *WWW*, 2019, pp. 2000–2010.

[70] Y.C.M.L. X. Wang X. He and T.-S. Chua, KGAT: Knowledge graph attention network for recommendation, in: *SIGKDD*, 2019, pp. 950–958. https://doi.org/10.1145/3292500.3330989.

[71] J.F. Z. Wang J. Zhang and Z. Chen, Knowledge graph and text jointly embedding, in: *EMNLP*, 2014, pp. 1591–1601–.

[72] H.X. Y. Wei J. Luo, KGRL: an OWL2 RL reasoning system for large scale knowledge graph, in: *12th International Conference on Semantics, Knowledge and Grids*, 2016, pp. 83–89.

[73] V.T. Maximilian Nickel and H.-P. Kriegel, A three-way model for collective learning on multi-relational data, in: *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, Omnipress, 2011, pp. 809–816–.

[74] B. Wen, J. Luo, X. Liu and L. Huang, Unbiased Scene Graph Generation via Rich and Fair Semantic Extraction, *IEEE Transactions On Image Processing* (2020).

[75] Y.H. H. Xiao M. Huang and X. Zhu, TransA: An adaptive approach for knowledge graph embedding, in: *Proceedings of AAAI Conference on Artificial Intelligence*, 2015, pp. 1–7.

[76] H.Z.J.Y.H.H. S. Yang J. Tian and Y. Jin, TransMS: knowledge graph embedding for complex relations by multidirectional semantics, in: *Proceedings of IJCAI*, 2019, pp. 1935–1942. doi:10.24963/ijcai.2019/268.

[77] H. Xiao, M. Huang, L. Meng and X. Zhu, SSP: Semantic Space Projection for Knowledge Graph Embedding with Text Descriptions, 2017.

[78] V. Gutiérrez-Basulto and S. Schockaert, From Knowledge Graph Embedding to Ontology Embedding? An Analysis of the Compatibility between Vector Space Representations and Rules, *arXiv e-prints* (2018), arXiv:1805.10461–.

[79] G. Diaz, A. Fokoue and M. Sadoghi, EmbedS: Scalable, Ontology-aware Graph Embeddings, in: *Proceedings of the 21st International Conference on Extending Database Technology*, 2018, pp. 433–436. doi:10.5441/002/edbt.2018.40.

[80] B. Ding, Q. Wang, B. Wang and L. Guo, Improving Knowledge Graph Embedding Using Simple Constraints, in: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 110–121.

[81] R. Xie, Z. Liu, J. Jia, H. Luan and M. Sun, Representation Learning of Knowledge Graphs with Entity Descriptions, in: *Proceedings of AAAI Conference on Artificial Intelligence*, AAAI'16, AAAI Press, 2016, pp. 2659—2665.

[82] M. Ou, P. Cui, J. Pei, Z. Zhang and W. Zhu, Asymmetric Transitivity Preserving Graph Embedding, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 1105––1114.

[83] H.-G. Yoon, H.-J. Song, S.-B. Park and S.-Y. Park, A Translation-Based Knowledge Graph Embedding Preserving Logical Property of Relations, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, 2016, pp. 907–916. doi:10.18653/v1/N16-1105. https://www.aclweb.org/anthology/N16-1105.

[84] S.Y. Min, P. Raghavan and P. Szolovits, Trans{INT}: Embedding Implication Rules in Knowledge Graphs with Isomorphic Intersections of Linear Subspaces, in: *Automated Knowledge Base Construction*, 2020, pp. 1–18. https://openreview.net/forum?id=shkmWLRBXH.

[85] J. Hao, M. Chen, W. Yu, Y. Sun and W. Wang, Universal Representation Learning of Knowledge Bases by Jointly Embedding Instances and Ontological Concepts, in: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1709—1719.

[86] Q. Wang, B. Wang and L. Guo, Knowledge Base Completion Using Embeddings and Rules, in: *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, AAAI Press, 2015, pp. 1859—1865.

[87] P. Minervini, T. Demeester, T. Rocktaschel and S. Riedel, Adversarial sets for regularising neural link predictors, in: *Uncertainty in Artificial Intelligence - Proceedings of the 33rd Conference, UAI 2017*, Sydney, NSW, Australia, 2017, pp. 1–10.

[88] M. Krötzsch, S. Rudolph and P. Hitzler, Complexities of Horn Description Logics, *ACM Trans. Comput. Logic* **14**(1) (2013). doi:10.1145/2422085.2422087.

[89] H.I.e.a. Motik B Grau B C, OWL 2 web ontology language profiles, *W3C Recommendation 2009* (2009).

[90] V.H.F. McGuinness D L, OWL web ontology language overview, *W3C Recommendation 2004* (2004).

[91] H.P.L. Dave Reynolds, OWL 2 RL in RIF (Second Edition), *W3C Working Group Note 5 February 2013* (2013).

[92] W.O.W. Group, OWL 2 Web Ontology Language Document Overview (Second Edition), *W3C Recommendation 11 December 2012* (2012).

[93] J. Hao, M. Chen, W. Yu, Y. Sun and W. Wang, Universal Representation Learning of Knowledge Bases by Jointly Embedding Instances and Ontological Concepts, in: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, KDD '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1709–1719–. ISBN 9781450362016. doi:10.1145/3292500.3330838.

[94] P.W. Battaglia, J.B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro and R. Faulkner, Relational inductive biases, deep learning, and graph networks, *CoRR* (2018).

[95] Y. Bengio, Y. Lecun and G. Hinton, Deep Learning for AI, *Commun. ACM* **64**(7) (2021), 58–65–. doi:10.1145/3448250.

[96] Y. Bengio, The Consciousness Prior, 2019.

[97] S.M. Kazemi, R. Goel, K. Jain, I. Kobyzev, A. Sethi, P. Forsyth and P. Poupart, Representation Learning for Dynamic Graphs: A Survey, 2020.

[98] T. Ebisu and R. Ichise, TorusE: Knowledge Graph Embedding on a Lie Group, in: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2018.

[99] T. Lacroix, N. Usunier and G. Obozinski, Canonical Tensor Decomposition for Knowledge Base Completion, 2018.

[100] R. Abboud, İsmail İlkan Ceylan, T. Lukasiewicz and T. Salvatori, BoxE: A Box Embedding Model for Knowledge Base Completion, 2020.