Decoupling Mixture-of-Graphs: Unseen Relational Learning for Knowledge Graph Completion by Fusing Ontology and Textual Experts

Ran Song[1,2], **Shizhu He**[3,4], **Suncong Zheng**[6], **Shengxiang Gao**[1,2] *;
Kang Liu[3,4,5], **Zhengtao Yu**[1,2], and **Jun Zhao**[3,4]
[1] Faculty of Information Engineering and Automation, Kunming University of Science and Technology
[2] Yunnan Key Laboratory of Artificial Intelligence, Kunming, China
[3] Institute of Automation, Chinese Academy of Sciences, Beijing, China
[4] School of Artificial Intelligence, University of Chinese Academy of Science, Beijing, China
[5]Beijing Academy of Artificial Intelligence, Beijing, China [6] Tencent AI Lab
{song_ransr, junzhao123}@163.com, {shizhu.he, kliu}@nlpr.ia.ac.cn,
congzheng@tencent.com, {gaoshengxiang.yn, ztyu}@hotmail.com

Abstract

Knowledge Graph Embedding (KGE) has been proposed and successfully utilized for knowledge Graph Completion (KGC). But classic KGE paradigm often fail in unseen relation representations. Previous studies mainly utilize the textual descriptions of relations and its neighbor relations to represent unseen relations. In fact, the semantics of a relation can be expressed by three kinds of graphs: factual graph, ontology graph, textual description graph, and they can complement each other. A more common scenario in the real world is that seen and unseen relations appear at the same time. In this setting, the training set (only seen relations) and testing set (both seen and unseen relations) own different distributions. And the train-test inconsistency problem will make KGE methods easily overfit on seen relations and underperformance on unseen relations. In this paper, we propose decoupling mixture-of-graph experts (DMoG) for unseen relations learning, which could represent the unseen relations in the factual graph by fusing ontology and textual graphs, and decouple fusing space and reasoning space to alleviate overfitting for seen relations. The experiments on two unseen-only public datasets and a mixture dataset verify the effectiveness of the proposed method, which improves the state-of-the-art methods by 6.84% in Hits@10 on average.

1 Introduction

Knowledge Graphs (KGs) such as Freebase (Bollacker et al., 2008), DBpedia (Lehmann et al., 2015) and YAGO (Mahdisoltani et al., 2014) contain large amounts of entities, relations and facts, which can be used to support many NLP tasks. Knowledge-dependent tasks rely heavily on the coverage of KGs. And the incompleteness of those KGs is an urgent issue for its widespread utilization (Hogan et al., 2021). Therefore, knowledge graph embedding (KGE) (Arora, 2020; Ji et al., 2021) methods have been proposed and successfully applied to knowledge graph completion (KGC), which seek out potential facts inside KGs.

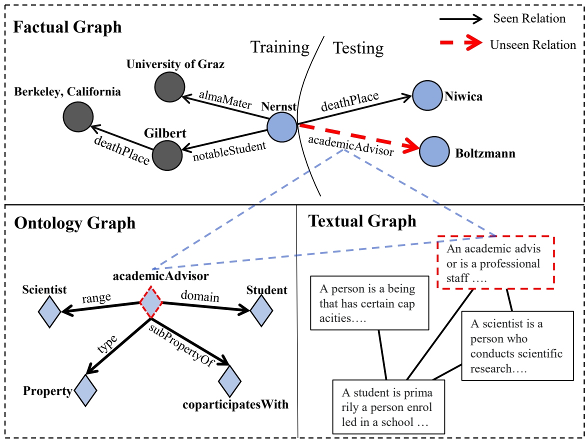


Figure 1: The semantics of a relation in a KG is expressed by three kinds of graphs: factual graph, ontology graph and textual graph. And the knowledge graph completion involving unseen relations in factual graph, which struggle in previous KGE methods, could be alleviated by utilizing their ontology and textual graphs.

In fact, most knowledge involves constantly emerging new entities and relations. And traditional KGE paradigm makes hard to deal with unseen entities and relations. For example, as illustrated in Figure 1 top, the model was trained on seen triples dataset, but required to answer the open query "*(Nernst, academicAdvisor, ?)*". Little research try to learning unseen relations representation mainly by utilizing textual description and neighboring seen relations (Qin et al., 2020; Geng et al., 2021; Zhang et al., 2020). And some datasets have been proposed to evaluate models generalizability for unseen relations. (Qin et al., 2020).

* Corresponding author

Proceedings of the 29th International Conference on Computational Linguistics, pages 2237–2246
October 12–17, 2022.

ZSGAN-KG (Qin et al., 2020) leverages a generative adversarial network to generate representations of unseen relations based on their textual descriptions. And OntoZSL (Geng et al., 2021) designs several functions to learn and fuse textual features, and then adapt a text-aware encoder to represent zero-shot entities and relations. GRL (Zhang et al., 2020) designs a classifier to select a neighboring seen relation to replace the unseen relation.

Although the above-mentioned methods can deal with unseen relations to some extent, they still have the following weaknesses: 1) Unstructured textual descriptions are incomplete and can only cover part of the semantics of relations. 2) It is often inaccurate and even noisy to use neighbor relations to represent an unseen relations. 3) Previous models are evaluated for unseen data only, but do not consider mixture dataset of seen and unseen relations. Therefore in this paper, we propose three questions for currently unseen relation methods: 1) What resources are accurate and abundant for unseen relations? 2) How to efficiently use resources to improve representations of unseen relations? 3) What challenges arise with models evaluated in mixture dataset of seen and unseen relations?

To answer above questions 1) and 2), we found that ontology, as an accurate resource, is worth considering. In fact, as shown in Figure 1, the semantics of a relation can be expressed by three different forms: 1 **factual graph** includes concrete relations between entities, 2 **ontology graph** describes high-level definitions for relations, and 3 **textual graph** contains textual descriptions of different relations. The factual graph is wide and links large amounts of entities by relations. Most KGE methods can represent KG effectively, but can not represent unseen relations. The ontology contains a high-level definition of entities and relations. In Figure 1, for relation *academicAdvisor*, the ontology graph means that *scientist have academic advisors who are students*. The textual descriptions are rich and contain different semantic information in natural language. In Figure 1, they can be assembled into a textual graph through words and sentences association. The above three graphs complete each other. We leverage ontology graph and textual graph to support factual graph to find unseen relations information.

To answer above the question 3), we found that seen and unseen relations appear at the same time in the real world. In this setting, the training set (only seen relations) and testing set (both seen and unseen relations) own different distributions. And the train-test inconsistency problem will make KGE methods easily overfit on seen relations and underperforman on unseen relations. Empirically, as the model converges in the training process the performance of seen relations gets higher, but the performance of unseen relations decreases. To overcome this issue, we are committed to making the learning of relation representations and factual reasoning in different spaces. That is, the relation representations by mixture-of-graphs is implemented on the fusion space, and fact prediction with the learned entities and relations is conducted on the reasoning space.

In this paper, we propose decoupling mixture-of-graph experts (DMoG) for unseen relations learning, which could represent the unseen relations of the factual graph by fusing ontology and textual graphs. And our method decouples fusion space and reasoning space to alleviate the overfitting on seen relations. Specifically, we collect different ontology graphs from official graph-based data, or we derive them from official dump data in other formats. To achieve the interactive information between seen and unseen relations, we leverage different GNNs to encode ontology and textual graphs. And we design different expert modules and mixture mechanism to fuse different graph information. Morevoer, we propose a transpose linear mapping to separate fusion space and reasoning space and alleviate overfitting.

We conducted extensive experiments on multiple benchmarks from public KGs such DBpedia and Wikidata. The proposed unseen relations learning method improves the state-of-the-art method by 3.68% in MRR and 6.15% in Hits@10 on average.

In short, our main contributions are as follows:

- We found that relations are expressed by factual graph, ontology graph and textual graph. Based on these observations, we propose mixture-of-graph (MoG) experts for unseen relations learning, which can represent unseen relations accurately and richly.
- We propose a decoupling strategy that alleviates the overfitting on seen relations during training. Trained KGE models effectively represent seen relations and maintain unseen relations performance.
- We implement our method with some main-

stream KGE methods. And the experimental results show that our method significantly improves the performance on the seen and unseen relations.

2 Related Work

2.1 Knowledge Graph Embedding

Recently, massive work focused on translation-based methods for knowledge graph completion (Zhang et al., 2021). The key issue of knowledge graph embedding is to learn low dimensional distributed embedding of entities and relations (Ji et al., 2021). The current KGE models can generally be categorized into translation-based models and similarity-based models. For KGE models: the pioneering model TransE (Bordes et al., 2013) embeds entities and relations as d-dimension vectors in same space, and makes vectors follow the translational principle $\mathbf{h} + \mathbf{r} = \mathbf{t}$. The subsequent work of TransE usually modifies the translational principle in different forms of relationship-specific spaces. And others translation-based models including TransR (Lin et al., 2015), TransD (Ji et al., 2015), TransAt (Qian et al., 2018) and RotatE (Sun et al., 2019) have been improved from the perspective of how entities can be better represented and translated. As for the similarity-based models, ComplEx (Trouillon et al., 2016) migrates DistMult in a complex space and offers comparable performance. However, previous embedding methods struggle in knowledge completion involving unseen relations.

2.2 Zero-shot Learning for KGC

Zero-shot learning describes tasks that given the prior knowledge (seen classes) and then transfer features from seen classes to unseen classes. Most works focus on computer vision such as image classification. In the area of knowledge graph completion, more studies focus on zero-shot entity learning which is devoted to deal with unseen entities. Some works leverage text and other auxiliary features to learn the entity representation(Xie et al., 2016; Shah et al., 2019). Some works design different models or strategies to aggregate neighbor seen entities for unseen entities (Wang et al., 2019; Albooyeh et al., 2020). Currently, inductive reasoning(Teru et al., 2020) completely disregards the symbol of entities and it means that all entities can be unseen entities. While few works consider zero-shot relation learning and model unseen relations. Few works take text-embedding spaces as semantic spaces of relation to represent unseen relations (Qin et al., 2020; Geng et al., 2021). And (Zhang et al., 2020) design a classifier-based method, which select an appropriate seen relation to replace the unseen relation. Our work focuses on unseen relations in knowledge graph completion and proposes a method that incorporates ontology graph and textual description to leaning the representations of unseen relations.

2.3 Ontology and Textual Information for KGE

The ontology is the definition and meta-information of KG, it is a core part of KG construction (Stevens et al., 2000). The massive KG relation facts are subject to frequent conflicts in the absence of ontological boundaries (Pasternack and Roth, 2013). A few studies focus on embedding techniques of cross-domain ontology and encode ontology from different perspectives (Chen et al., 2018; Gutiérrez-Basulto and Schockaert, 2018). Currently, some studies try to adapt ontology to enhance the representation of knowledge base. JOIE (Hao et al., 2019) employs both cross-view and intra-view modeling that learn on multiple facets of the knowledge base. For textual information, (Yao et al., 2019) propose to use pre-trained language models for knowledge graph completion. However, there are significant differences in the ontology of the knowledge base and knowledge graph. And some popular knowledge graphs do not distinguish between KB and KG (Ehrlinger and Wöß, 2016). Our work focuses on learning ontology representation for KGE involving unseen relations.

3 Knowledge Graph Embedding Models for KGC

KGC aims at scoring a triple (h, r, t) from KG $\mathcal{G} = (\mathcal{R}, \mathcal{E})$, where $r \in \mathcal{R}$ is relation and $h, t \in \mathcal{E}$ are entities. Traditional KGE models learn embedding matrix to translate head entity h to tail entity t through relation r. And different models have been proposed by mainly changing translating strategies. For example, TransE focuses on adding head entity and relation, which should be close to the corresponding tail entity with the scoring function by minimizes the score of a triple as follows:

$$s(h, r, t) = \parallel \mathbf{h} + \mathbf{r} - \mathbf{t} \parallel_2^2 \tag{1}$$

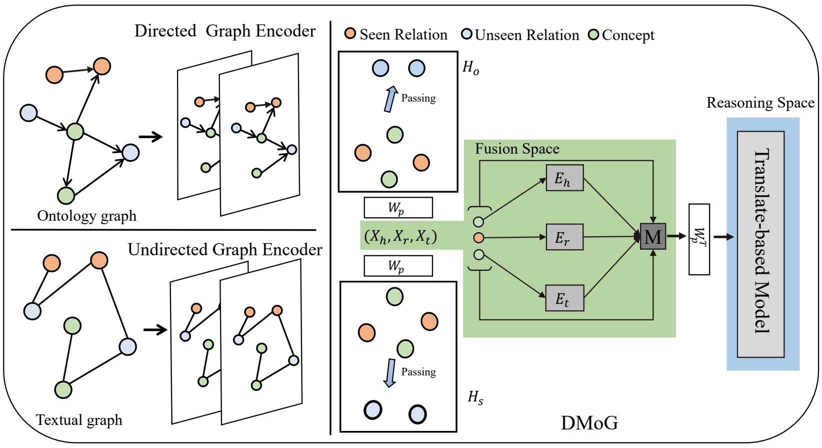


Figure 2: Our method leverages different GNN to capture ontology graph and textual graph nodes information and aggregate them by knowledge mixture of experts. By fusion ontology and textual features in fusion space (green), DMoG pools the representation of the relation in predicting a triplet fact in reasoning space (blue).

where $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{R}^d$, and d is dimension of embedding.

KGE models use the hinge loss function to effectively minimize the score. The loss function for a minibatch of labeled triples is defined as follows:

$$\mathcal{L}(\theta) = \sum_{(h,r,t) \in \mathcal{G}_b} [\gamma + f(h', r, t') - f(h, r, t)]_+ \tag{2}$$

where γ is a fixed margin, (h', r, t') is the negative fact that is commonly constructed by randomly replacing the head or tail entities from the true fact (h, r, t).

For evaluation, KGC is a link prediction task that aims to predict the missing h or t for a triple (h, r, t). Given the query $(h, r, ?)$, search the entity t that gets the minimum score with scoring function.

However, the embeddings (e.g., vectors) of all entities and relations must be initialized at the beginning for previous KGE models. If some relations r miss in training but appear in testing, they cannot be learned at all by the model. Therefore, in order to represent the unseen relations and conduct zero-shot relational learning, we consider to leverage multi-aspects information.

4 Decoupling Mixture-of-Graphs Experts

This section describes in detail our proposed approach. The framework is shown in Figure 2. Our method directly improves the effectiveness of previous KGE models for unseen relations by making rich and accurate their representations.

4.1 Framework

Our method mainly deals with three types of graphs: factual graph, ontology graph and textual graph. Factual graph is knowledge graph, following as the previous definition. Ontology is the backbone of KGs, which provide meta-descriptions to guide the knowledge graph construction and completion. Ontology is describe as directed graph $\mathcal{G}_o = (\mathcal{R}_o, \mathcal{E}_o)$, which uses meta-relations to associate ontology nodes (concepts and properties) (h_o, r_o, t_o). And the relations $\mathcal{R}$ and entities $\mathcal{E}$ of factual graph all find their own type in ontology. And relations have a unique mapping between the edges of factual graph and the nodes of ontology graph. Textual graph is undirected graph $\mathcal{G}_t = (\mathcal{R}_t, \mathcal{E}_t)$, the nodes are textual descriptions of concept and property and the edge is word embedding similarity between two nodes (h_t, r_t, t_t), and $0 < r_t < 1$.

4.2 Graph Construction

Ontology is stored in triples *(head, relation, tail)*, we take head and tail as node (indicates concept (type of entity) and property (type of relation) of factual graph) and relation (indicate meta-relations among concepts and properties) as edge. Based on the official released ontology file or the dump data, we can directly construct or build ontology graph by simple data filtering and format conversion.

For textual descriptions, we generate textual graph from textual descriptions or full names of concepts and properties. We want to find them associated as below:

$$A_t = \begin{cases} d(\mathbf{x}_i, \mathbf{x}_j), & if \quad d(\mathbf{x}_i, \mathbf{x}_j) > \varepsilon \\ 0, & otherwise \end{cases}$$

A_t is adjacency matrix of textual graph $\mathcal{G}_t$, $d(\cdot, \cdot)$ describes the cosine similarity function, ε is a threshold for connection between nodes. x_i is the word embedding of each node. Following previous work (Qin et al., 2020), Glove (Pennington et al., 2014) has higher performance than the pre-trained language model, and we use Glove to initialize word embeddings. The representation of a sentence is obtained by averaging its word embeddings.

4.3 Graph Encoder

The ontology can be represented as a directed attribute graph. Identically, the text descriptions of relations can be represented as an undirected graph. Our goal is to obtain the representation of unseen relations based on other seen nodes (concepts, properties and textual descriptions) in different graphs. Therefore, we encode ontology and textual graphs by graph neural network (GNN).

In the textual graph $\mathcal{G}_t$, the weight w_{ij} of each edge is the similarity between nodes. We consider the commonly used graph attention network (Veličković et al., 2017), but the attention value is replaced by edge weight w. The process as following:

$$\mathbf{h}_{s,i}^{(l+1)} = \sigma(\sum_{j \in \mathcal{N}_i} w_{ij} \mathbf{W}_s^l \mathbf{h}_{s,j}^l + \mathbf{W}_0^l \mathbf{h}_{s,i}) \qquad (3)$$

where $\mathbf{h}_{s,i}^{(l+1)} \in \mathbb{R}^d$. The σ is sigmoid activation function. $\mathbf{W}_s^l$ is GAT weight. $\mathcal{N}_i$ denotes neighbor nodes of i. And the $\mathbf{h}_s^0$ for each node come from pretrained word embedding. To overcome the over-smooth problem of node representations, we add self-loop encoding for nodes. $\mathbf{W}_0^l$ is self-loop weight.

Similarly, ontology graph is directed graph, and each edge has its own type. We are inspired by RGCN (Schlichtkrull et al., 2018), a GNN model for relational (directed and labeled) multi-graph. To obtain the representations of concepts and properties, we use RGCN to get the representation of ontology nodes by aggregating neighborhoods nodes through different meta-relations, as follow:

$$\mathbf{h}_{o,i}^0 = e_o \mathbf{E}_o \qquad (4)$$

$$\begin{aligned}\mathbf{h}_{o,i}^{(l+1)} = ReLU(&\sum_{r \in \mathcal{R}_l} \sum_{j \in \mathcal{N}_i^r} \frac{1}{c_{i,r}} \mathbf{W}_r^{(l)} \mathbf{h}_{o,j}^{(l)} \\ &+ \mathbf{W}_0^{(l)} \mathbf{h}_{o,i}^{(l)})\end{aligned} \qquad (5)$$

$$\mathbf{h}_{o,i}^{(l+1)} = Norm\_Layer(\mathbf{h}_{o,i}^{(l+1)}) \qquad (6)$$

where $e_o \in \mathcal{E}_o$ is node in ontology graph. $\mathbf{h}_{o,i}^{(l+1)} \in \mathbb{R}^d$ is hidden state of ontology node $h_{o,i}$ in the l-th layer, and d is dimension of layer's representation. $\mathcal{N}_i^r$ denotes the set of neighbor indices of node i under meta-relation $r_o \in \mathcal{R}_o$. $\mathbf{W}_r^{(l)}$ is relation parameters of meta-relation r which weight for node i neighboring node in l-th layer. $\mathbf{W}_0^{(l)}$ is self-loop weight for encoding self-node features. $c_{i,r}$ is a normalization constant that can either be learned or chosen in advance. ReLU is the activation function. We also use layer normalization to speed the training.

4.4 Decoupling Mixture of Graph Experts

We obtain the effective representation of the relation r from ontology space and textual space for the triple involving unseen relations and alleviate the overfitting on seen relations. For each factual triple (h, r, t), we can find ontology representation (h_o, r_o, t_o) and textual representation (h_t, r_t, t_t) from their space. We leverage adding operation to fuse two graphs information in another space, as show in:

$$\mathbf{x}_h = \mathbf{h}_o^L \mathbf{W}_p + \mathbf{h}_s^L \mathbf{W}_p \qquad (7)$$

$$\mathbf{x}_r = \mathbf{r}_o^L \mathbf{W}_p + \mathbf{r}_s^L \mathbf{W}_p \qquad (8)$$

$$\mathbf{x}_t = \mathbf{t}_o^L \mathbf{W}_p + \mathbf{t}_s^L \mathbf{W}_p \qquad (9)$$

where $\mathbf{W}_p$ is transformation matrix, it transforms the multi representations into the fusion-aware space.

Based on the previous representations, we design aggregating strategies with mixture-of-graph experts to represent relations. Recently, the mixture of experts (Jordan and Jacobs, 1994; Shazeer et al., 2017; Fedus et al., 2021) has been widely used to capture features by different experts' views, and it can efficiently merge different features. For different knowledge roles (head, relation, tail), MoG can capture each role representation through ontology and textual space. We define different expert networks E_h, E_r, E_t for the head, relation, tail, and a gating network M, proceeding as follows:

$$p_i = M(\mathbf{x_i}) \tag{10}$$

$$\mathbf{r} = \sum_{i \in (h,r,t)} p_i E_i(\mathbf{x}_i) \tag{11}$$

The expert networks and gating network are single-layer MLPs, and same dimension between input and output for expert networks. analyze the three roles individually and then vote to obtain the overall result.

After those processes, the representation r contain multiple information, but it still need to be put in fusion space V_f. We operate a inverse transformation to pull r back to reasoning space V_r from fusion space V_f, as follows:

$$\mathbf{r} = \mathbf{r}\mathbf{W}_p^T \tag{12}$$

where $\mathbf{W}_p^T$ transpose of a linear map, and we define $\mathbf{W}_p^T$ as square matrix to simplify calculations. It can be learned to satisfy two space bilinear forms, as follows:

$$\mathbf{W}_p = \min_{\substack{\mathbf{W}_p: V_r \to V_f \\ \mathbf{W}_p^T: V_f \to V_r}} \sum_i^N \mathcal{L}(h_i, r_i, t_i) \tag{13}$$

where N is seen dataset.

Following previous KGE models, we train our model with the margin-based ranking loss, and use a negative sampling loss function for effectively optimizing ranking loss :

$$\begin{aligned} \mathcal{L} = & -log\sigma(\gamma - f(h\mathbf{W}^E, \mathbf{r}, t\mathbf{W}^E)) \\ & - \sum_{i=1}^{n} \frac{1}{k} log\sigma(f(h_i'\mathbf{W}^E, \mathbf{r}, t_i'\mathbf{W}^E) - \gamma) \end{aligned} \tag{14}$$

where $\mathbf{W}^E$ indicates entity embedding, γ is a fixed margin value, σ is the sigmoid function, and (h_i', r, t_i') is the corresponding negative triple. The loss function can sample multiple negative triples for each positive triple at one minibatch.

5 Experiments

We conduct extensive experiments with KGC task on several public datasets, and mainly evaluate the performance of the proposed framework on zero-shot relational learning. We also verify the proposed decoupling strategy to prevent overfitting on seen relations. To directly demonstrate the effectiveness of our method, we show a visualization of seen and unseen relations.

5.1 Dataset

We select datasets from four public knowledge graphs, DBpedia, NELL, and Wikidata, to evaluate models on unseen relation learning. The current benchmark datasets contain only factual graph and not ontology graph. Therefore, we extract ontology from their origin websites[12]. Generally, we collect series ontology: **DBpedia** have human-created high-quality ontology, their have 17,663 triples, 7,966 nodes and 8 meta-relations. The ontology of **NELL** has 1,494 nodes, 6,907 triples and 14 meta-relations (e.g. *antisymmetric*, *mutexpredicates*). It should be noted that **Wikidata** has no official ontology, we collect 20,899 triples including 8,907 nodes and 604 meta-relations (e.g. *instance of* (P31), *see also* (P1659)) as their ontology from the released dump data [3].

Current zero-shot relational benchmarks focus entirely on inference on unseen relations. However, the seen and unseen relations should be be considered together. It requires that the model must be effective for seen relations and maintain unseen relation performance. Therefore, we propose DB100K-ZS from DB100K, which contains 383 seen relations and 77 unseen relations. We move 77 relations from training set to testing set based on DB100K. We select relations by frequency of appearing k, $k > 60$ and $k < 300$. Finally, we get training triples 540,570, seen validation triples 45,357, and seen testing triples 45,282 and unseen testing triples 13,420.

5.2 Evaluation Metrics

Triples in training data are utilized to learn KGE model, while those of validation and test dataset are respectively used to tune (hyper-parameters selection) and evaluate the model. The most typical KGC task is link prediction which aims to predict the missing h or t for a triple (h, r, t). We follow

[1] https://www.dbpedia.org/resources/ontology/

[2] http://rtw.ml.cmu.edu/resources/results/08m/NELL.08m.1115.ontology.csv.gz

[3] https://www.wikidata.org/wiki/Wikidata:Database_download

Model	NELL-ZS		Wiki-ZS		DB100K-ZS			
	UNSEEN		UNSEEN		UNSEEN		SEEN	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
DistMult	23.50	32.60	18.90	23.60	4.61	9.12	9.23	20.17
TransE	9.70	20.30	5.30	11.90	2.24	7.41	14.87	40.14
GRL(TransE) (Zhang et al., 2020)	-	-	-	-	5.15	13.12	15.12	41.51
$ZSGAN_{KG}$(DistMult) (Qin et al., 2020)	25.30	37.10	20.80	29.40	-	-	-	-
$ZSGAN_{KG}$(TransE) (Qin et al., 2020)	24.00	37.60	18.50	26.10	-	-	-	-
OntoZSL(DistMult) (Geng et al., 2021)	25.60	38.50	21.10	28.90	-	-	-	-
OntoZSL(TransE) (Geng et al., 2021)	25.00	39.90	18.40	26.50	-	-	-	-
DMoG(DistMult)	25.81	38.41	19.12	28.86	10.33	23.91	14.51	35.18
DMoG(TransE)	**30.49**	**49.11**	**23.18**	**31.13**	**23.31**	**40.79**	**27.37**	**52.07**

Table 1: Zero-shot relational learning results on NELL-ZS, Wiki-ZS and DB100K-ZS. SEEN is that relation of triples exist in training. UNSEEN is that relation of triples only in testing. **Bold** numbers denote the best results.

the setting (Sun et al., 2019) and create the query $(h, r, ?)$, and then find the ranking entities assigned by our proposed method and other KGE methods. We also apply bi-direction prediction that evaluate query $(h, r, ?)$ and $(?, r, t)$ for a test triple. The mean reciprocal rank (MRR) is computed as:

$$\frac{1}{2N_{Test}} \sum_{(h,r,t)\in Test} \left(\frac{1}{MR_{(h,r,?)}} + \frac{1}{MR_{(?,r,t)}}\right) \tag{15}$$

5.3 Implementation Details

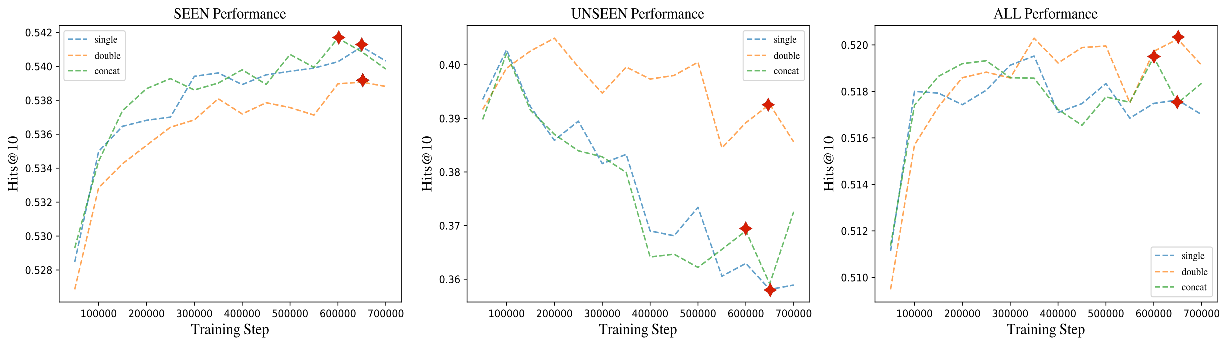
In our experiments, we adopt the following KGE methods because of their effectiveness on link predictions. Our codes are based on (Sun et al., 2019) and adopt the PyTorch (Paszke et al., 2017) framework. For graph encoder, we used the implementation in the deep graph library (DGL). The initial word embedding is from GloVe (Pennington et al., 2014) and we set a similar threshold ε to 0.85. The entity embedding size is set to 100 for all KGE models. The GNN hidden size is set to 100, the number of layers is set to 2, and use self-loop for each node. We selected the hyperparameters corresponding to learning rate and batch size from $\{0.0001, 0.0005, 0.001\}$ and $\{128, 256, 512, 1024\}$. And we use Adam to optimize all the parameters.

5.4 Results

The unseen relations denote that relation of the triples are in the test set, but they do not appear in the training set. Previous KGE models are transductive inference methods, and cannot deal with those relations. Table 1 shows the experimental results on NELL-ZS, WiKi-ZS and DB100K-ZS. The testing set of NELL-ZS and WiKi-ZS are all unseen relations (Qin et al., 2020), DB100K-ZS mix seen and unseen relations. Apparently, the newly constructed DB100K-ZS is more suitable for real-world applications.

To verify our method for unseen relation learning, we chose the latest proposed models for comparison. The GRL (Zhang et al., 2020) is the classifier-based method and hard to solve massive unseen relation. ZSGAN (Qin et al., 2020) and OntoZSL (Geng et al., 2021) always generate a representation for relation, therefore it cannot to keep traditional KGE methods performance in the seen dataset, and they do not work in DB100K-ZS.

From Table 1, our method performs better than other comparative methods in all evaluation metrics and on all three datasets. Our method increases MMR and Hits@10 by 3.86% and 6.15% for the previous state-of-the-art zero-shot method on NELL-ZS and Wiki-ZS. And our method can deal with seen and unseen relations at same time. For DB100K-ZS, DMoG not only improves the performance of unseen relations but goes beyond the base model on seen relations. We believe that the proposed model is more suitable to real world applications. In fact, Graph encoder effectively represents nodes from ontology graph and textual graph. And DMoG fully mixes different roles to extract the representation of unseen relations. The decoupling alleviates overfitting in training. The above three reasons are the key factors for our approach to achieve better results. In addition, our method could apply to any conteined in the ontology.

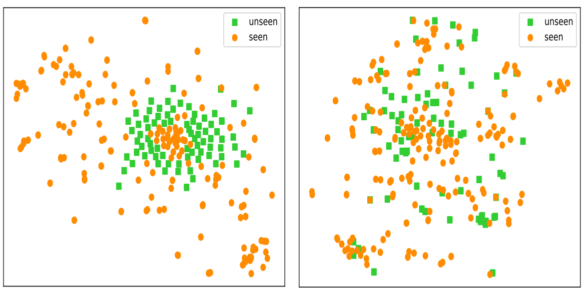


(a) (b) (c)

Figure 3: Hist@10 for seen (a), unseen (b) and all (c) relations performance in different training step. The red start denote the best performance of model in seen relations, and it still marks the same step in seen and all relations performance. In each figure, the single denote directly adding two representations. The double denote proposed decoupling strategy methods to separate the representations in different spaces. The concat denote concatenated two representations.

5.5 Alleviate Overfitting Experiment

In our setting, model trained by seen relations triples and can not get any information of unseen relations during training. And, we take early stopping through seen relations triples performance. The red star marks best checkpoint in seen performance but not all performance. As seen relations performance increases the unseen relations performance become lower, as show in Figure 3 (a, b). The reason is that model fit seen relations data and far away unseen relations latent representation. Therefore, we propose methods, which decouple fuse space and reasoning space, to alleviate the overfitting on seen relations. As show in Figure 3 (b), our method could make unseen performance decline more slowly compared to single space methods. While, our method harms seen performance little, due to its excellent unseen performance, it still has the best performance on whole seen+unseen performance, as show in Figure 3 (c).



(a) TransE (b) DMoG(TransE)

Figure 4: Visualization for relation representations of DB100K-ZS testing set via t-SNE.

Model	DB100K-ZS			
	UNSEEN		SEEN	
	MRR	H@10	MRR	H@10
TransE	2.24	7.41	14.87	40.14
DMoG-T(TransE)	15.92	30.10	20.17	44.16
DMoG-O(TransE)	21.12	37.13	25.96	51.10
DMoG(TransE)	**23.31**	**40.79**	**27.37**	**52.07**

Table 2: The table shows the ablation experiment for using different information. "-T" denote only textual graph. "-O" denote only ontology graph.

5.6 Visualization of Relation Representations

In Figure 4, we show the visualization of relation representations via t-SNE. As show in Figure 4 (a), TransE can not represent unseen relations effectively, the unseen relation embeddings crowded in a cluster, which separate away seen relations space. However, our method can fully represent seen and unseen relations, the relation representations uniformly distributed in the same space, as show in Figure 4 (b).

5.7 Ablation Experiment

In order to further evaluate the effect of each module of the model, we design an ablation experiment for different graphs. As shown in Table 2, we can see that both ontology and textual graphs are helpful to KGC. DMoG enhances relations representation quality by fusing ontology and textual graph compared to single information. Further analysis showed that ontology graph is better than textual graph, formal language describe knowledge more accurately than natural language.

6 Conclusion

Our paper focuses on unseen relation representations of knowledge graph. We propose to utilize three different kinds of graphs to obtain representations of relation. And decoupling strategy alleviates the overfitting in training process. Experimental results demonstrate that our method significantly outperforms the existing state-of-the-art method on unseen relation learning.

7 Acknowledgements

The work is supported by the National Natural Science Foundation of China (U21B2027, 61972186, 61732005), the independent research project of National Laboratory of Pattern Recognition, the Youth Innovation Promotion Association CAS, the Program For Applied Basic Research of Yunnan Province in Key Areas (202002AD080001), Yunnan Provincial Major Science and Technology Special Plan Projects (No.202103AA080015) and CCF-DiDi GAIA Collaborative Research Funds for Young Scholars.

References

Marjan Albooyeh, Rishab Goel, and Seyed Mehran Kazemi. 2020. Out-of-sample representation learning for knowledge graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2657–2666.

Siddhant Arora. 2020. A survey on graph neural networks for knowledge graph completion. *arXiv preprint arXiv:2007.12374*.

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

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems (NIPS)*, pages 1–9.

Muhao Chen, Yingtao Tian, Xuelu Chen, Zijun Xue, and Carlo Zaniolo. 2018. On2vec: Embedding-based relation prediction for ontology population. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 315–323. SIAM.

Lisa Ehrlinger and Wolfram Wöß. 2016. Towards a definition of knowledge graphs. *SEMANTiCS (Posters, Demos, SuCCESS)*, 48:1–4.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*.

Yuxia Geng, Jiaoyan Chen, Zhuo Chen, Jeff Z Pan, Zhiquan Ye, Zonggang Yuan, Yantao Jia, and Huajun Chen. 2021. Ontozsl: Ontology-enhanced zero-shot learning. In *Proceedings of the Web Conference 2021*, pages 3325–3336.

Víctor Gutiérrez-Basulto and Steven Schockaert. 2018. From knowledge graph embedding to ontology embedding? an analysis of the compatibility between vector space representations and rules. *arXiv preprint arXiv:1805.10461*.

Junheng Hao, Muhao Chen, Wenchao Yu, Yizhou Sun, and Wei Wang. 2019. 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*, pages 1709–1719.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. 2021. Knowledge graphs. *Synthesis Lectures on Data, Semantics, and Knowledge*, 12(2):1–257.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. 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 (volume 1: Long papers)*, pages 687–696.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

Michael I Jordan and Robert A Jacobs. 1994. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2):181–214.

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

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian Suchanek. 2014. Yago3: A knowledge base from multilingual wikipedias. In *7th biennial conference* *on innovative data systems research*. CIDR Conference.

Jeff Pasternack and Dan Roth. 2013. Latent credibility analysis. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1009–1020.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Wei Qian, Cong Fu, Yu Zhu, Deng Cai, and Xiaofei He. 2018. Translating embeddings for knowledge graph completion with relation attention mechanism. In *IJCAI*, pages 4286–4292.

Pengda Qin, Xin Wang, Wenhu Chen, Chunyun Zhang, Weiran Xu, and William Yang Wang. 2020. Generative adversarial zero-shot relational learning for knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8673–8680.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Haseeb Shah, Johannes Villmow, Adrian Ulges, Ulrich Schwanecke, and Faisal Shafait. 2019. An open-world extension to knowledge graph completion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3044–3051.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Robert Stevens, Carole A Goble, and Sean Bechhofer. 2000. Ontology-based knowledge representation for bioinformatics. *Briefings in bioinformatics*, 1(4):398–414.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

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

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pages 2071–2080. PMLR.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Peifeng Wang, Jialong Han, Chenliang Li, and Rong Pan. 2019. Logic attention based neighborhood aggregation for inductive knowledge graph embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7152–7159.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193*.

Jing Zhang, Bo Chen, Lingxi Zhang, Xirui Ke, and Haipeng Ding. 2021. Neural, symbolic and neural-symbolic reasoning on knowledge graphs. *AI Open*, 2:14–35.

Yao Zhang, Xu Zhang, Jun Wang, Hongru Liang, Wenqiang Lei, Zhe Sun, Adam Jatowt, and Zhenglu Yang. 2020. Generalized relation learning with semantic correlation awareness for link prediction. *CoRR*, abs/2012.11957.