# Mixture-of-Graphs: Zero-shot Relational Learning for Knowledge Graph by Fusing Ontology and Textual Experts

**Anonymous ACL submission**

## Abstract

Knowledge Graph Embedding (KGE) has been proposed and successfully utilized to knowledge Graph Completion (KGC). But dominant KGE models often fail in zero-shot relational learning because they cannot learn effective representations for unseen relations. Previous studies mainly separately utilize the textual description of relation 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 and enhance each other. Therefore, to obtain a more accurate representation of relation in zero-shot learning, we propose the mixture-of-graphs (MoG) experts to improve the effectiveness of current KGE for unseen relations. We build multi-aspect associations between seen and unseen relations which will be used directly to guide previous KGE methods such as TransE and RotatE on zero-shot relational learning. The experiments on multiple public datasets verify the effectiveness of the proposed method, which improves the state-of-the-art zero-shot relational learning method by 12.84% 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, but the incompleteness of those KGs is an urgent issue for its widespread utilization. Recently, knowledge graph embedding (KGE) represented by translation-based methods (Bordes et al., 2013; Yang et al., 2014; Sun et al., 2019) have been proposed and successfully applied to knowledge graph completion (KGC), which attempts to embed a KG into a low-dimensional continuous space with the observed triplet facts. And the numerical representations (e.g., vectors) of entities and relations can be used to predict potential triplet facts.

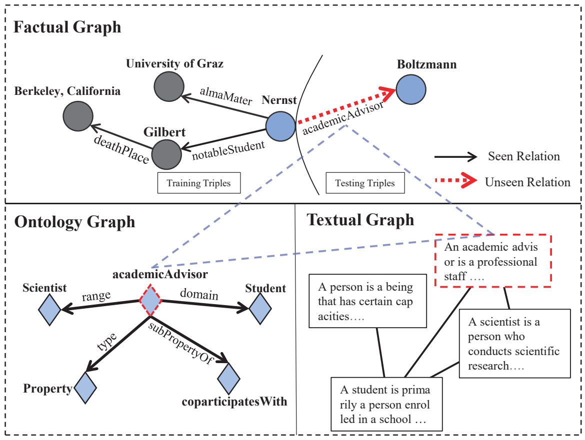



Figure 1: The semantics of a relation in a KG is expressed by three kinds of graphs: factual graph, ontology graph and textual graph. And the knowledge graph completion involving unseen relations in factual graph, which struggle in previous embedding methods, could be alleviated by utilizing their ontology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

However, as a kind of transductive learning paradigm (Zhang et al., 2021), translation-based and other dominant KGE methods are struggling in predicting facts involving unseen entities and relations during training. That is, KGE models can only deal with entities and relations that have been observed in the training set. And the representations of unseen entities and relations cannot be learned by previous methods. For example, as illustrated in Figure 1, the unseen relation "*academicAdvisor*", which do not appear in the observed facts (training set), is represented as a random initial tensor (e.g., vector) and cannot be used in KGC at all.

This paper focuses on zero-shot relational learning and deals with KGC on unseen relations, it receives less attention than zero-shot entity learning (Wang et al., 2019; Albooyeh et al., 2020; Teru et al., 2020). So far, the limited studies realize zero-shot generalization mainly by utilizing the textual description of the target relation and its neighboring relations (Qin et al., 2020; Geng et al., 2021; Zhang et al., 2020). 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 an appropriate seen relation to replace the unseen relation. Although those methods can deal with unseen relations to some extent, they still have the following weaknesses: 1) Unstructured textual description is 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 relation. Therefore, when making predictions about a fact that contains an unseen relation, the most challenge of zero-shot relational learning is how to obtain its rich and accurate semantic representations

In fact, as shown in Figure 1, the semantics of a relation can be expressed by three different forms: ①**factual graph** includes concrete relationships between entities, ②**ontology graph** describes high-level abstract relationships among concepts and relations, and ③**textual graph** contains textual descriptions of different relations. Those three graphs can complement and enhance each other. Firstly, the factual graph is wide and links large amounts entities by relations. The large-scale entity-relation networks enable the efficient propagation of representations. Secondly, the ontology is the precise specification of a KG, it contains a high-level definition of entities and relations. And the relations of a KG are constrained by concepts and properties through meta-relations such as "*domain*", "*range*" and "*subPropertyOf*". Finally, the textual descriptions are rich and hold massive semantic information by natural language. They can be built textual graph through word and sentence association. Therefore, we attempt to utilize the corresponding ontology and textual graphs of a KG to obtain rich and accurate semantic representations of unseen relations.

In this paper, we propose mixture-of-graph experts for zero-shot relational learning, named ZRL-MoG, which could represent the unseen relations of the factual graph by fusing ontology and textual graph. Specifically, we collect different ontology graphs by official or build them by official dump data. We generate textual graph by language association between each ontology node. To achieve the interactive information between seen and unseen relations, we leverage RGCN (Schlichtkrull et al., 2018) to encode textual graph and GAT (Veličković et al., 2017) to encode ontology graph. And to aggregate different roles of knowledge, we consider the mixture of experts approach to design different expert modules and mixture mechanism for them (Jordan and Jacobs, 1994). We adapt TransE, RotatE and other mainstream translation-based methods as base models, and the proposed method can be used in combination with most dominant KGE models.

We conducted extensive experiments on multiple benchmark datasets from public KGs such DBpedia and Wikidata. The experiments are conducted mainly on zero-shot relation learning. And we also verify the proposed model of KGC on general and sparse data. The experimental results demonstrate that the proposed method can learn a better knowledge graph embedding, and it significantly improves the KGC performance on unseen relations. Moreover, it also performs better on general and sparse data. The proposed zero-shot relational learning method improves the state-of-the-art method by 15.66% in MRR and 12.84% in Hits@10 on average. The experimental codes can be accessed by `https://github.com/AnonymousOne404/OntologyKGE`. In short, our main contributions are as follows:

- We consider that relations expressed by factual graph, ontology graph and textual graph, they can complement and enhance each other. Based on these observations, we propose mixture-of-graph (MoG) experts for zero-shot relational learning, which can represent unseen relations accurately and richly.

- We construct and generate ontology and textual graph, and leverage different GNN to encode them. We also utilize a mixture of experts to aggregate information from ontology and textual graph representations.

- We implement our method with some mainstream methods such as TransE and RotatE. And the experimental results show that our method significantly improves the KGC performance on the setting of zero-shot, general and sparse data.

## 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 translational 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 include 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 involve unseen relations.

### 2.2 Zero-shot Learning for KGC

Zero-shot learning describes tasks that give 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 entity. 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. The limited 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, proposes a method that incorporates ontology graph and textual description to leaning the representations of unseen relations.

### 2.3 Ontology and Textual Information in 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 Translation-based 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 translation-based models usually 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 should be close to the corresponding tail entity with the scoring function, minimizes the score of a triple as follows:

$$s(h, r, t) = \| \mathbf{h} + \mathbf{r} - \mathbf{t} \|_2^2 \tag{1}$$

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

Translation-based models usually use the hinge loss function to effectively minimize the score. The $\mathcal{L}(\theta)$ for a single batch of labeled triples are defined as follows:

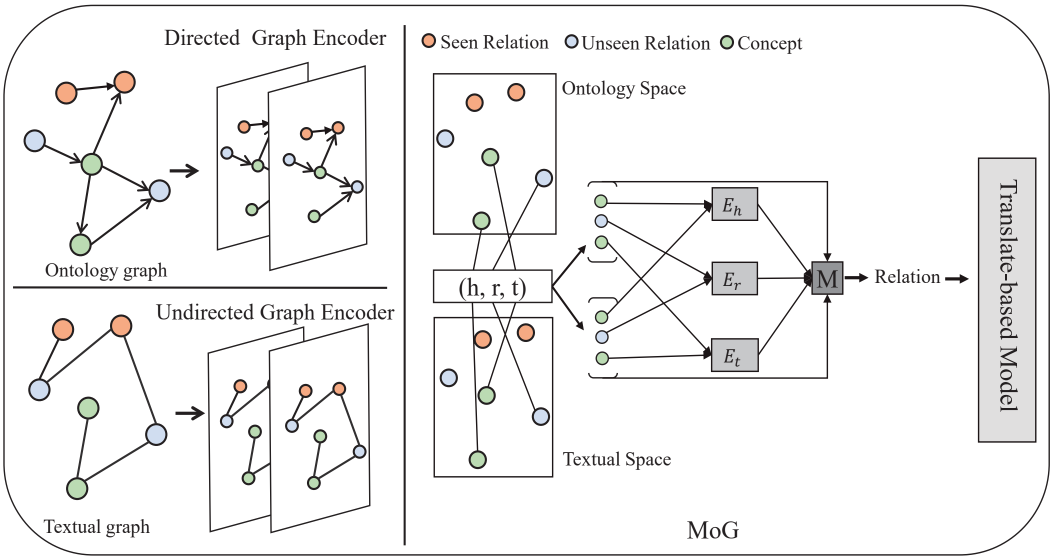



Figure 2: Our method leverages different GNN to capture ontology graph and textual graph nodes information and aggregate them by knowledge mixture of experts. By fusing ontology and textual features, MoG pools the representation of the relation in predicting a triplet fact.

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

where $\gamma$ is a fixed margin, $(h', r, t')$ is the negative fact which is commonly constructed by randomly replacing the head or tail entities of 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 $(h, r, ?)$ and search the entity $t$ who gets minimum score with scoring function.

However, the embeddings (e.g., vectors) of all entities and relations must be initialized at the beginning for previous translation-based 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 missing relations and conduct zero-shot relational learning, we consider to leverage multi-aspects information.

## 4 Mixture-of-Graphs Experts

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

### 4.1 Framework

Our method mainly utilizes 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 for guiding the knowledge graph construction and completion. Ontology describe as directed graph $\mathcal{G}_o = (\mathcal{R}_o, \mathcal{E}_o)$, which uses meta-relations to associate between 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 meta information in ontology. And relations have a directly mapping between the edges of factual graph and the nodes of ontology graph. Textual graph is undirected graph $\mathcal{G}_t = (\mathcal{R}_t)$, the nodes is textual descriptions of concept and property and the edge is semantic 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 and property 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 bellow:

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

$A_t$ is adjacency matrix of textual graph $\mathcal{G}_t$, $d(\cdot,\cdot)$ describes the cosine similarity function, $\varepsilon$ is a threshold for connection nodes. $x_i$ is the word embedding of each node. Following previous work (Qin et al., 2020), the Glove (Pennington et al., 2014) has higher performance than the pre-trained language model, and we use Glove to initialize word embedding. 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 multiple relations can be represented as an undirected graph. Our goal is to obtain the representation of unseen relations based on other seen nodes (entities, concepts, textual descriptions and relations) in different graphs. Therefore, we encode ontology and textual graph 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:

$$\mathbf{h}_{i,s} = \sigma(\sum_{j\in\mathcal{N}_i} w_{ij}\mathbf{W}_s\mathbf{h}_{j,s}) \tag{3}$$

The $\sigma$ is sigmoid activation function. $\mathbf{W}_s$ is GAT weight. $\mathcal{N}_i$ denotes neighbor nodes of $i$.

Similarly, ontology graph is undirected graph, and each edge has its own type. 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:

$$\begin{aligned}\mathbf{h}_{o,i}^{(l+1)} = ReLU(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{C_{i,r}}\mathbf{W}_r^{(l)}\mathbf{h}_{o,j}^{(l)} \\ +\mathbf{W}_0^{(l)}\mathbf{h}_{o,i}^{(l)})\end{aligned} \tag{4}$$

$$\mathbf{h}_{o,i}^{(l+1)} = Norm\_Layer(\mathbf{h}_{o,i}^{(l+1)}) \tag{5}$$

where $\mathbf{h}_{o,i}^{(l)} \in \mathbb{R}^d$ is hidden state of ontology node $h_{o,i}$ in the $l$-th layer of RGCN, 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 activate function. We also use layer normalization to speed the training.

### 4.4 Mixture of Graph Experts

The core issue of this paper is how to obtain the effective representation of the relation $r$ from ontology space and textual space, especially for the triple involving unseen 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. Based on the previous node 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 $M$, process as follows:

$$p_i = M(\mathbf{x}) \tag{6}$$

$$\mathbf{r} = \sum_{i\in(h,r,t)} p_i E_i(\mathbf{x}) \tag{7}$$

where $\mathbf{x} \in \{[\mathbf{h_o} : \mathbf{h_t}], [\mathbf{r_o} : \mathbf{r_t}], [\mathbf{t_o} : \mathbf{t_t}]\}$ each input $\mathbf{x}$ concats corresponding representation of ontology and textual description. The expert networks and gating network are single-layer MLPs, and same dimension between input and output for expert networks. Experts analysis tree roles individually and voting for final results.

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

| 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 | 5.53 | 10.44 | 10.13 | 22.97 |
| TransE♣ | 9.70 | 20.30 | 5.30 | 11.90 | 2.64 | 9.18 | 17.55 | 43.34 |
| $ZSGAN_{KG}$(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 | - | - | - | - |
| GRL(TransE) (Zhang et al., 2020) | - | - | - | - | 4.42 | 12.72 | 16.72 | 40.48 |
| MoG(DistMult) | 25.72 | 39.65 | 21.42 | 29.56 | 11.33 | 27.91 | 14.90 | 36.08 |
| MoG(TransE) | **31.29** | **49.91** | **27.67** | **38.42** | **27.08** | **45.60** | **28.49** | **53.50** |

Table 1: Zero-shot relational learning results on NELL-ZS, Wiki-ZS and DB100K-ZS. Seen relations is that relation of triples exist in training. The results of ♣ consider candidate sets to filter, which are constructed by the entity type (Qin et al., 2020; Toutanova et al., 2015). The other results search the whole entity set in testing. **Bold** numbers denote the best results.

| Meta Relation | Count |
|---|---|
| dense#sameDomain | 11,668 |
| 22-rdf-syntax-ns#type | 5,004 |
| dense#sameRange | 3,136 |
| rdf-schema#range | 2,588 |
| rdf-schema#domain | 2,421 |
| rdf-schema#subPropertyOf | 975 |
| rdf-schema#subClassOf | 769 |
| owl#disjointWith | 25 |
| ALL | 26,586 |

Table 2: The DBpedia ontology contains 8 meta-relations and 3,937 nodes. To increase the density of the ontology graph, we add two new meta-relation *dense#sameRange* and *dense#sameDomain*, which connect two properties (relations) who has same range or domain.

optimizing ranking loss :

$$\begin{aligned}\mathcal{L} = &- log\sigma(\gamma - f(h\mathbf{W}^E, \mathbf{r}, t\mathbf{W}^E)) \\ &- \sum_{i=1}^{n} \frac{1}{k} log\sigma(f(h_i'\mathbf{W}^E, \mathbf{r}, t_i'\mathbf{W}^E) - \gamma)\end{aligned} \tag{8}$$

where $\mathbf{W}^E$ indicates entity embedding, $\gamma$ is a fixed margin value, $\sigma$ 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 once.

## 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 effectiveness of the proposed method for KGC on general and sparse data.

### 5.1 Dataset

We select datasets from four public knowledge graphs, DBpedia, NELL, YAGO, and Wikidata, to evaluate the effectiveness of KGC on zero-shot relational learning. The current benchmark datasets contain only factual graph and not ontology graph. Therefore, we extract ontology from their origin websites[1,2,3]. Generally, we collect series ontology: **DBpedia** have human-created high-quality ontology, their meta-relations and statistics are shown in Table 2. The ontology of **NELL** has 1,494 nodes, 6,907 triples and 14 meta-relations (e.g. *antisymmetric*, *mutexpredicates*). And ontology of **YAGO** has 654 nodes, 2,452 triples and 28 meta-relations (e.g. *causes*, *synonym*). It should be noted that **Wikidata** has no publish ontology by official, 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 [4].

Current zero-shot relational benchmarks are totally inference for unseen relations. However, the seen and unseen relations should be be considered together. It requires that the model must be effective for unseen relations and maintain seen relation performance. Therefore, we propose DB100K-ZS from DB100K, which contains 383 seen relations and 77 unseen relations. We move 77 relations

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

[2] http://resources.mpi-inf.mpg.de/yago-naga/yago3.1/yagoSchema.tsv.7z

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

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

| | DB100K | | | YAGO26K | | | DB111K | | |
|---|---|---|---|---|---|---|---|---|---|
| | MRR | MR | Hits@10 | MRR | MR | Hits@10 | MRR | MR | Hits@10 |
| DistMult | 13.16 | 9915 | 28.84 | 6.12 | 1142 | 30.74 | 13.79 | 11870 | 29.57 |
| Complex | 14.46 | 11443 | 30.32 | 7.32 | 1428 | 29.55 | 15.22 | 9231 | 36.21 |
| TransE | 17.23 | 1611 | 41.75 | 12.37 | 421 | 31.90 | 16.6 | 2217 | 40.80 |
| RotatE | 21.40 | 1052 | 47.50 | 13.28 | 409 | 36.12 | 21.35 | 1600 | 48.82 |
| MoG(RotatE) | **29.64** | **445** | **54.99** | **16.94** | **401** | **42.01** | **30.00** | **981** | **56.28** |

Table 3: The KGC results on DB100K (Ding et al., 2018), YAGO26K and DB111K (Hao et al., 2019). The effect of previous KGE models is shown at the top of the table. MoG represent relations for RotatE. **Bold** numbers denote the best results.

from training set to validation set and testing set base on DB100K. We select relations by frequency of appearing $k$, $k$> 60 and $k$< 300. Considering that the ontology is a fix schema, we reserve the entire ontology graph in training.

### 5.2 Evaluation Metrics

Triples in training data are utilized to learning KGE model, while those of validation and test dataset are respectively used to tune (hyper-parameters selection) and evaluate 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 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:

$$\frac{1}{2N_{Test}} \sum_{(h,r,t)\in Test} \left(\frac{1}{MR_{(h,r,?)}} + \frac{1}{MR_{(?,r,t)}}\right) \quad (9)$$

### 5.3 Implementation Details

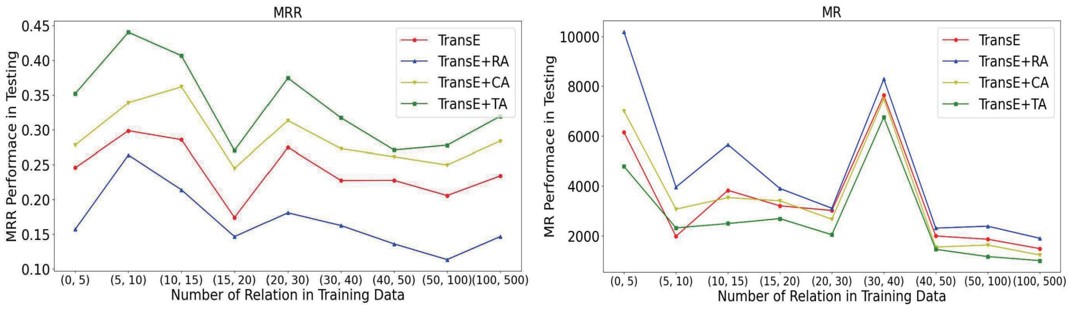
In our experiments, we adopt the following translation-based methods because of their efficiency and effectiveness on link predictions: DistMult, Complex, TransE, and RotatE. Our codes are based on (Sun et al., 2019) and adopt the PyTorch (Paszke et al., 2017) framework. For graph encoder, we used the implementation in the deep graph library (DGL). The initial word embedding is come from GloVe (Pennington et al., 2014) and we set a similar threshold $\varepsilon$ to 0.85. The entity embedding size is set to 100 for all translation-based methods. 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 in the test set but do not appear in the training set. Previous translation-based 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. Comparatively speaking, the newly constructed DB100K-ZS is more suitable for real-world applications.

To verify our method for zero-shot relational 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 and only unseen relation in the test dataset. ZSGAN (Qin et al., 2020) and OntoZSL (Geng et al., 2021) always generate a representation for relation, therefore it is hard to keep traditional translation-based method performance in the seen dataset, and they do not work in DB100K-ZS which match has no candidate sets.

From Table 1, we can find that 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 15.66% and 12.84% 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. On DB100K, MoG not only improves the performance of unseen relations but beyond the base model on seen relations. We believe that the proposed model more suitable in real-world scenes. In fact, Graph encoder effectively represents nodes from ontology graph and textual graph. And MoG fully mixes different roles to extract the representation of unseen relations. The above two reasons are the key factors for our approach to achieve better results.




(a) (b)

Figure 3: The four figures provide results of DB100K sparse relations. They show the number of appearing in training (on the x-axis). The results in testing (on the y-axis) for MMR and MR. “RA” (Relation Aggregating) considers properties of relation in ontology nodes. “CA” (Concept Aggregating) considers entity in ontology nodes. “TA” (Triple Aggregating) considers relation node and entity in ontology nodes.

| **Model** | **DB100K-ZS** | | | |
|---|---|---|---|---|
| | **UNSEEN** | | **SEEN** | |
| | MRR | H@10 | MRR | H@10 |
| TransE | 2.64 | 9.18 | 17.55 | 43.34 |
| MoG-T(TransE) | 16.23 | 30.21 | 21.57 | 45.56 |
| MoG-O(TransE) | 23.12 | 40.43 | 27.96 | 52.09 |
| MoG(TransE) | **27.08** | **45.60** | **28.49** | **53.50** |

Table 4: The table shows the ablation experiment for using different information. “-T” denote only textual graph. “-O” denote only ontology graph.

### 5.5 Ablation Experiment

In order to further evaluate the effect of each module of the model, we design an ablation experiment for different graphs. The experimental results are shown in Table 4, from which we can see that both ontology and textual graphs are helpful to KGC. MoG fusing ontology and textual graph into relations to enhance representation quality of relations and improves 4.3 and 1.93 MMR for single information. Further analysis showed that ontology graph is better than textual graph, formal language describe knowledge more accurate than natural language.

### 5.6 KGC on General and Sparse Data

In order to verify whether the incorporation of ontology graph has conducted to normal and sparse data (relations). We conducted KGC experiments on several benchmark datasets. We calculate sparsity by the number of occurrences of entities and relationships in facts. We retrain all base models in our environment.

According to the results on datasets reported in Table 3, we can observe that all models were significantly improved by strengthening our proposed method. Our method consistently improves four base models (DistMult, Complex, TransE, RotatE) on MMR, MR and Hits@10. Our method improves the previous state-of-the-art zero-shot method by 36.67% in MMR, 15.73% in Hist@10 and 40.13% MR. And we leverage different parts of the triple base on TransE to represent relations. We believe the most reason is the ontology and textural graph are naturally fit for our graph encoder to obtain relations’ representation.

The sparse learning main focus on the sparse relations in training. Figure 3 show the sparse information result on DB100K. Our method improves the base model by average 12.0, 28.8 and 19.5 Hits@10 scores for sparse relations. Our method obtains high performance in sparse relations. For sparse data, as the information of ontology nodes increases, our method becomes more effective for sparse data. We believe the most reason is more nodes serve as the representation of the relation can carry more information from other properties and concepts.

## 6 Conclusion

This paper focuses on zero-shot relational learning for knowledge graph. We propose to utilize three different kinds of graphs (factual graph, ontology graph and textual graph) to obtain a more accurate representation of relation in zero-shot learning. By a mixture-of-graphs (MoG) experts, the proposed method will be used directly to guide previous KGE methods such as TransE on zero-shot relational learning. Experimental results demonstrate that our method significantly outperforms the existing state-of-the-art method on unseen relation learning.

## 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.

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.

Boyang Ding, Quan Wang, Bin Wang, and Li Guo. 2018. Improving knowledge graph embedding using simple constraints. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 110–121.

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.

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.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1499–1509.

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.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*.

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.