A Novel Embedding Model for Knowledge Graph Completion Based on Multi-Task Learning

Jiaheng Dou, Bing Tian, Yong Zhang(✉), and Chunxiao Xing

BNRist, Department of Computer Science and Technology, RIIT,
Institute of Internet Industry, Tsinghua University, Beijing, China
{djh19,tb17}@mails.tsinghua.edu.cn, {zhangyong05,xingcx}@tsinghua.edu.cn

Abstract. Knowledge graph completion is the task of predicting missing relationships between entities in knowledge graphs. State-of-the-art knowledge graph completion methods are known to be primarily knowledge embedding based models, which are broadly classified as translational models and neural network models. However, both kinds of models are single-task based models and hence fail to capture the underlying inter-structural relationships that are inherently presented in different knowledge graphs. To this end, in this paper we combine the translational and neural network methods and propose a novel multi-task learning embedding framework (TransMTL) that can jointly learn multiple knowledge graph embeddings simultaneously. Specifically, in order to transfer structural knowledge between different KGs, we devise a global relational graph attention network which is shared by all knowledge graphs to obtain the global representation of each triple element. Such global representations are then integrated into task-specific translational embedding models of each knowledge graph to preserve its transition property. We conduct an extensive empirical evaluation of multi-version TransMTL based on different translational models on two benchmark datasets WN18RR and FB15k-237. Experiments show that TransMTL outperforms the corresponding single-task based models by an obvious margin and obtains the comparable performance to state-of-the-art embedding models.

1 Introduction

Knowledge Graphs (KGs) such as WordNet [16] and Freebase [1] are graph-structured knowledge bases whose facts are represented in the form of relations (edges) between entities (nodes). This can be represented as a collection of triples ($head\,entity$, $relation$, $tail\,entity$) denoted as (h, r, t), for example ($Beijing$, $CapitalOf$, $China$) is represented as two entities: $Beijing$ and $China$ along with a relation $CapitalOf$ linking them. KGs are important sources in many applications such as question answering [2], dialogue generation [10] and

J. Dou and B. Tian – contribute equally to this work.

C. S. Jensen et al. (Eds.): DASFAA 2021, LNCS 12681, pp. 240–255, 2021.
https://doi.org/10.1007/978-3-030-73194-6_17

recommender systems [34]. Containing billions of triples though, KGs still suffer from incompleteness, that is, missing a lot of valid triples [24,31]. Therefore, many research efforts have concentrated on the Knowledge Graph Completion (KGC) or link prediction task which entails predicting whether a given triple is valid or not [4,24]. Recent state-of-the-art KGC methods are known to be primarily knowledge embedding based models, which are broadly classified as translational models [3,21,32] and neural network models [8,20,23]. Translational models aim to learn embeddings by representing relations as translations from head to tail entities. For example, the pioneering work TransE [3] assumes that if (h, r, t) is a valid fact, the embedding of head entity h plus the embedding of relation r should be close to the embedding of tail entity t, i.e. $v_h + v_r \approx v_t$ (here, v_h, v_r and v_t are embeddings of h, r and t respectively). In order to learn more deep expressive features, recent embedding models have raised interests in applying deep neural networks for KGC such as Convolutional Neural Network (CNN) [8] and capsule network [20]. Recently, some studies explored a new research direction of adopting Graph Neural Network (GNN) [23] for knowledge graph completion, which demonstrates superior effectiveness and advantages than traditional translational methods since it takes the relationship of different triples into consideration. Among the GNN models, Graph Attention Network (GAT) [29] is an effective and widely used model which utilizes attentive nodes aggregation to learn neighborhood information. Although the effectiveness of these models, they are all single-task based models and ignore the inter-structural relations that are inherently presented in different knowledge graphs. To that end, such methods need to train different models for each knowledge graph, which involves substantial extra efforts and resources.

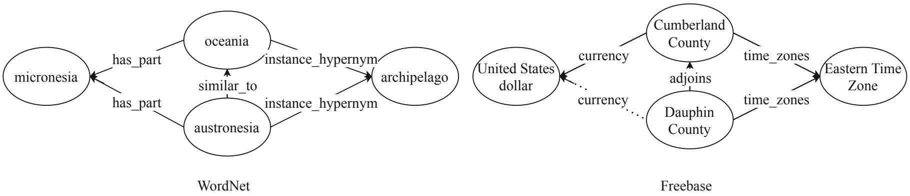

Fig. 1. An example of shared structure pattern

Nevertheless, we find that different knowledge graphs are structurally interrelated and one knowledge graph can benefit from others. On the one hand, since different knowledge graphs have different data characteristics, they can complement each other by simultaneously learning the representations. For example, WordNet provides semantic knowledge of words. It contains few types of relations but each relation corresponds to a large number of triples. In comparsion, Freebase provides more specific facts of the world and contains a lagre number of relations with fewer entities. Therefore, knowledge representation model based on WordNet would be good at modeling and inferring the patterns of (or between) each relation such as symmetry/antisymmetry, inversion and composition [25] whereas the model based on Freebase enables to model more complex relations. As such, simultaneously learning the representations of these knowledge graphs can definitely promote and benefit each other. On the other hand, we observe that one knowledge graph may contain some common structural patterns that are beneficial for other knowledge graphs. An example is shown in Fig. 1 where the dotted line is the link needs to be predicted. For the missing link (*Dauphin Country*, ?, *United Stated dollar*) in Freebase, it is essential to understand the structural pattern that two entities connected by a symmetric relation usually exist in some triples linked by the same relations. However, it is hard for Freebase based embedding model to capture this kind of pattern since it is rare in this knowledge graph. As this kind of struture pattern is very common in WordNet which is shown in the left of the figure, the knowledge graph completion task based on Freebase can definitely benefit from them.

Motivated by such observations, in this paper we propose a novel embedding model for knowledge graph completion based on multi-task learning (TransMTL) where multiple knowledge graphs can be trained and represented simultaneously and benefit from each other. Specifically, in order to preserve the transition property of knowledge graphs, we first adopt the widely used translational models such as TransE, TransH and TransR to represent the entities and relations of each single knowledge graph. And then, we devise a global Translation preserved Relational-Graph Attention Network (TR-GAT) which is shared by all knowledge graphs to capture the inter-structural information between different knowledge graphs and obtain the global representation of each triple element. Such global representations are then integrated intotask-specific translational embedding models of each knowledge graph to enhance its transition property. In this way, each single knowledge graph can benefit from the common inter-structural information from other knowledge graphs through the global shared layer. Recall the example in Fig. 1, with the help of MTL, the information learned from WordNet can be transferred to Freebase representation task by means of the global sharing mechanism. Specifically, in WordNet, there exists a triple (*austronesia*, *similar_to*, *oceania*) containing the symmetry realtion *similar_to*. Then the head entity *austronesia* and tail entity *oceania* would exist in some triples linked by the same relation such as *instance_hypernym* and *has_part*. Once recognizing this kind of pattern in WordNet, the multi-task learning model could take advantage of such knowledge for link prediction in Freebase dataset. As there exists a triple (*Dauphin Country*, *adjoins*, *Cumberland Country*) with the symmetric relation *adjoins* and the head entity *Dauphin Country* and tail entity *Cumberland Country* exist in triples linked by the same relation *time_zones*, we can assume that the entity *Dauphin Country* may also linked by the relation *currency* since the *Cumberland Country* and *United Stated dollar* are linked by *currency*. We conduct an extensive empirical evaluation TransMTL based on different translational models on two benchmark datasets WN18RR and FB15k-237. Experiments show that our TransMTL outperforms the corresponding single-task based models by an obvious margin and obtains the comparable performance to state-of-the-art embedding models.

Contributions of this paper are summarized as follows:

- We propose a novel embedding model for knowledge graph completion based on multi-task learning (TransMTL) that can learn embeddings of multiple knowledge graphs simultaneously. To the best of our knowledge, this is the first attempt of multi-task learning in the field of knowledge representation for knowledge graph completion.
- We devise a translation preserved relational-graph attention network (TR-GAT) to utilize the shared information from multiple knowledge graphs, capturing inter-structural information in different knowledge graphs.
- We conduct extensive experiments on WN18RR and FB15k-237. Experimental results show the effectiveness of our model TransMTL.

2 Related Work

2.1 Knowledge Graph Completion (KGC)

Representation learning has been widely adopted in a variety of applications [15,35,36]. Recently, several variants of KG embeddings have been proposed following the paradigm of representation learning. These methods can be broadly classified as: semantic matching, translational and neural network based models. Firstly, semantic matching models such as DistMult [32], ComplEx [28] and Holographic Embeddings model (HolE) [22] use similarity-based functions to infer relation facts. Differently, translational models aim to learn embeddings by representing relations as translations from head entities to tail entities. For example, Bordes et al. [3] proposed TransE by assuming that the added embedding of $h + r$ should be close to the embedding of t with the scoring function defined under $L1$ or $L2$ constraints. Starting with it, many variants and extensions of TransE have been proposed to additionally use projection vectors or matrices to translate embeddings into the vector space, such as TransH [30], TransR [13] and TransD [11]. In recent studies, neural network models that exploit deep learning techniques have yielded remarkable predictive performance for KG embeddings. Dettmers et al. [8] introduced ConvE that used 2D convolution over embeddings and multiple layers of non-linear features to model knowledge graphs. To preserve the transitional characteristics, Nguyen et al. [19] proposed ConvKB that applied the convolutional neural network to explore the global relationships among same dimensional entries of the entity and relation embeddings. To capture long-term relational dependency in knowledge graphs, recurrent networks are utilized. Gardner et al. [9] and Neelakantan et al. [18] proposed Recurrent Neural Network (RNN)-based models over relation path to learn vector representation without and with entity information, respectively. To cover the complex and hidden information that is inherently implicit in the local neighborhood surrounding a triple, some studies used Graph Neural Networks (GNNs) for knowledge embeddings such as R-GCN [23] and KBGAT [17] etc.

Though the effectiveness of these models, they are all single-task based models and hence fail to capture the underlying inter-structural relationships that are inherently present in different knowledge graphs.

2.2 Multi-Task Learning

Multi-Task Learning (MTL) [5] is a learning paradigm in machine learning aiming at leveraging potential correlations and common features contained in multiple related tasks to help improve the generalization performance of all the tasks. It has been widely adopted in many machine learning applications from various areas including web applications, computer vision, bioinformatics, health informatics, natural language processing and so on. For example, Chapelle et al. [6] introduced a multi-task learning algorithm based on gradient boosted decision trees that is specifically designed with web search ranking in mind. Yim et al. [33] proposed a multi-task deep model to rotate facial images to a target pose and the auxiliary task aimed to use the generated image to reconstruct the original image. Chowdhury et al. [7] provided an end-to-end multi-task encoder-decoder framework for three adverse drug reactions detection and extraction tasks by leveraging the interactions between different tasks. Tian et al. [26] devised a multi-task hierarchical inter-attention network model to improve the task-specific document representation in nature language processing for document classification. In this paper, we utilize the idea of multi-task learning to transfer structural knowledge between different KGs by jointly learning multiple knowledge graph embeddings simultaneously.

3 Method

We begin this section by introducing the notations and definitions used in the rest of the paper, followed by a brief background on GAT [29]. Immediately afterwards, we introduce the details of our TransMTL framework as displayed in Fig. 2. It consists of two components: the task-specific knowledge embedding layer and the global shared layer. The task specific knowledge embedding layer is a translational model in which each KG learns low-dimensional embeddings of entities and relations. The global shared layer enables multi-task learning: we devise a Translation preserved Relational-Graph Attention Network (TR-GAT) to acquire the entity and relation embeddings simultaneously consisting of common structural information from all the knowledge graphs. Then such information is shared and integrated into the task-specific knowledge embedding layer to further enhance the entity and relation representations.

3.1 Background and Definition

A knowledge graph G is donated by $\mathscr{G} = (E, R, T)$ where E, R and T represent the set of entities (nodes), relations (edges) and triplets, respectively. It contains a collection of valid factual triples in the form of (head entity, relation, tail entity) denoted as (h, r, t) such that $h, t \in E$ and $r \in R$, representing the specific relation r linking from the head entity h to tail entity t. Knowledge embedding models aim to learn an effective representation of entities, relations, and a scoring function f which gives an implausibility score for each triple (h, r, t) such that

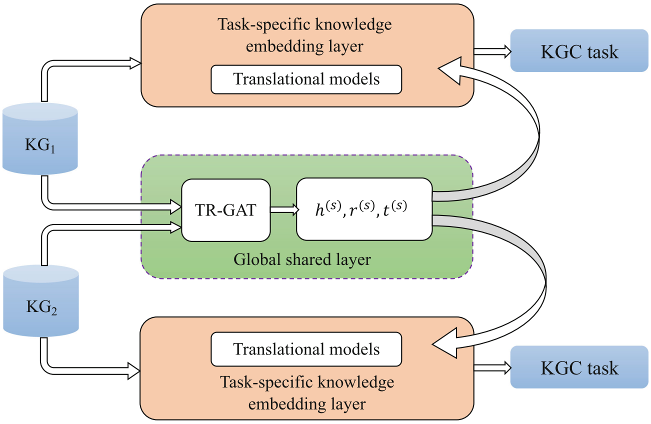
Task-specific knowledge embedding layer
Translational models
KGC task
KG$_1$
TR-GAT
$h^{(s)}, r^{(s)}, t^{(s)}$
Global shared layer
KG$_2$
Translational models
Task-specific knowledge embedding layer
KGC task

Fig. 2. The overall architecture of TransMTL

valid triples receive lower scores than invalid triples. With the learned entity and relation embeddings, the knowledge graph completion is to predict the missing head entity h given query $(?, r, t)$ or tail entity t given query $(h, r, ?)$.

3.2 Graph Attention Networks (GAT)

The concept of Graph Convolutional Networks (GCN) was first proposed in [12], which extended existing neural networks for processing the graph structured data. It gathers information from the entity's neighborhood and all neighbors contribute equally in the information passing. To resolve the shortcomings the GCNs, Velickovic et al. [29] proposed Graph Attention Networks (GAT). The advantage of GAT lies in the aspect that it leverages attention mechanism to assign varying levels of importance to nodes in every node's neighborhood, which enables the model to filter out noises and concentrate on important adjacent nodes. Specifically, the convolution layer attentionally aggregates features of each node in the graph as well as its one-hop neighbors as new features. The convolution process on the t^{th} layer for node v is formalized as Eq. (1)–(2).

$$h_v^{(t)} = \sigma\left(\sum_{u \in \mathcal{N}(v) \cup v} \alpha_{vu} W^{(t)} h_u^{(t-1)}\right) \tag{1}$$

$$\begin{aligned} \alpha_{vu} &= softmax(f(a_t^T [W^{(t)} h_v^{(t-1)} || W^{(t)} h_u^{(t-1)}])) \\ &= \frac{exp(f(a_t^T [W^{(t)} h_v^{(t-1)} || W^{(t)} h_u^{(t-1)}]))}{\sum_{j \in \mathcal{N}(v) \cup v} exp(f(a_t^T [W^{(t)} h_v^{(t-1)} || W^{(t)} h_j^{(t-1)}]))} \end{aligned} \tag{2}$$

where $W^{(t)}$ is the weight matrix, α_{vu} is the attention coefficient of node u to v, $\mathcal{N}(v)$ presents the neighborhoods of node v in the graph, f donates the $LeakyReLU$ function and a_t is the weight vector.

3.3 Task-Specific Knowledge Embedding Layer

In order to integrate the strength of translational property in knowledge graphs, we adopt the widely-used translation-based methods for each involved KG in task specific knowledge embedding layer, which benefit the multi-task learning tasks by representing embeddings uniformly in different contexts of relations. Here, we take the basic translational model TransE as an example to describe the embedding model. TransE [3] projects both relations and entities into the same continuous low-dimension vector space, in which the relations are considered as translating vectors from head entities to tail entities. Following the energy-based framework in TransE, the energy of a triplet is equal to $d(h+r,t)$ for some dissimilarity measure d. Specifically, the energy function is defined as:

$$E(h,r,t) = \|h + r - t\| \tag{3}$$

To learn such embeddings, we minimize the margin-based objective function over the training set, defined as:

$$K = \Sigma_{(h,r,t)\in T} L(h,r,t), \tag{4}$$

where L(h, r, t) is a margin-based loss function with respect to the triple (h, r, t):

$$L(h,r,t) = \Sigma_{(h',r',t')\in T'}[\gamma + E(h,r,t) - E(h',r',t')]_+, \tag{5}$$

where $[x]_+ = max(0,x)$ represents the maximum between 0 and x. T' stands for the negative sample set of T, donated as follows:

$$\begin{aligned} T' = \{(h',r,t)|h' \in E\} \cup \{(h,r,t')|t' \in E\} \\ \cup \{(h,r',t)|r' \in R\}, (h,r,t) \in T. \end{aligned} \tag{6}$$

The set of corrupted triplets, constructed according to Eq. (6), is composed of training triplets with either the head or tail replaced by a random entity (but not both at the same time). The objective function is optimized by stochastic gradient descent (SGD) with mini-batch strategy.

Note that in this paper, we aim at providing a general multi-task leaning solution to take advantage of the inter-structural knowledge between different KGs and not limited to any knowledge representation learning method. In other words, this task specific knowledge embedding layer can also be implemented through any other knowledge representation learning methods, including translational models and neural network models. In order to illustrate the effectiveness of multi-task learning in knowledge graph representation and KGC task, we implemented our TransMTL model based on TransE, TransH [30] and TransR [13] in this paper. The energy functions of TransH and TransR are defined as in Eq. (7) and Eq. (8) respectively.

$$\begin{aligned} E(h_\perp, r, t_\perp) &= \|h + d_r - t\| \\ &= \left\|(h - w_r^\top h w_r) + d_r - (t - w_r^\top t w_r)\right\|, \|w_r\|_2 = 1 \end{aligned} \tag{7}$$

$$E(h_r, r, t_r) = \|h_r + r - t_r\| = \|hM_r + r - tM_r\| \tag{8}$$

3.4 Global Shared Layer for Multi-task Learning

On the basis of task-specific model, we then utilize MTL techniques to improve the entity and relation representations. The intuition is that different knowledge graphs share some common structural knowledge, which can help improve the entity and relation representations of each knowledge graph and contribute to a better knowledge graph completion performance. The key factor of multi-task learning is the sharing scheme among different tasks. Considering the observations that existing KG embedding models treat triples independently and thus fail to cover the complex and hidden information that is inherently implicit in the local neighborhood surrounding a triple, we propose TR-GAT in Fig. 3 to acquire the entity and relation embeddings simultaneously by capturing both entity and relation features in any given entity's neighborhood.

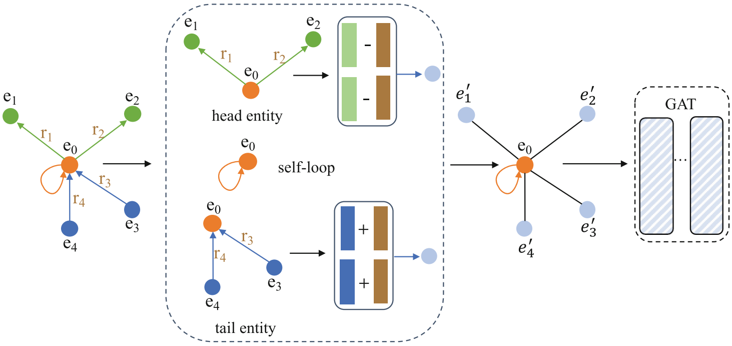

Fig. 3. Embedding processing in TR-GAT. Orange represents the center entity, brown represents relations connected with it, and green and blue represent its neighboring entities. If the entity has the head role, accumulating its neighboring tail nodes and relations with $t - r$. If it has the tail role, accumulating its neighboring head nodes and relations with $h + r$. Then the role discrimination representations passed through a GAT network during which the embeddings of entities and relations are updated.(Color figure online)

TR-GAT integrates the strength of GAT and the translational property in knowledge graphs ($h + r \approx t$) to design the new propagation model. As such, we modify the update rule in GAT for the entity and relation embeddings and the convolution process on the t^{th} layer for node v is formalized as Eq. 9:

$$h_v^l = \sigma(W^l(\sum_{r \in N_r} \sum_{t \in N_t^r} \alpha_{vt} c(h_t^{l-1}, h_r^{l-1}) + \sum_{r \in N_r} \sum_{h \in N_h^r} \alpha_{vh} \tilde{c}(h_h^{l-1}, h_r^{l-1}) + \alpha_v h_v^{l-1})) \tag{9}$$

where N_r denotes the set of relations connecting the entity i, N_t^r represents the set of tail entities connected with the entity i by the relation r, N_h^r is the set of head entities connected with the entity i by the relation r. $h_h^l \in R^{d(l)}$, $h_r^l \in R^{d(l)}$ and $h_t^l \in R^{d(l)}$ denote the l^{th} layer embedding of the head entity, relation and tail entity respectively in the neural network and $d(l)$ is the dimension of this layer. σ is the activation function. $c(\cdot, \cdot)$ is the function to describe the relationship between h_t^l and h_r^l, and $\tilde{c}(\cdot, \cdot)$ describes the relationship between h_h^l and h_r^l. W^l is the weight matrix of the l^{th} layer.

Equation 9 features the role discrimination criterion to identify if entity v in the knowledge graph takes the role of head or tail entity regarding a specific relation r. It performs different convolution operations for them: if v has the head entity role, its embedding is calculated by combining the related tail entity $h_t^{(l-1)}$ and relation $h_r^{(l-1)}$. Otherwise, its embedding is calculated with the related head entity $h_h^{(l-1)}$ and relation $h_r^{(l-1)}$. Thereafter, all occurrences of head roles and tail roles of v are added, together with a single self-connection representation h_v^{l-1}, to infer the l representation of the entity v.

The design of function c and $\tilde{c}$ features the translation adoption criterion which is $h+r \approx t$ for a triplet (h, r, t) in the graph. Alternatively, the translational property can be transformed into $h \approx t - r$ and $t \approx h + r$. Therefore,

$$c(h_t^l, h_r^l) = h_t^l - h_r^l \tag{10}$$

$$\tilde{c}(h_h^l, h_r^l) = h_h^l + h_r^l \tag{11}$$

Based on the TR-GAT, the overall embedding process of our TransMTL is as follows: for each entity and relation in a knowledge graph k ($G^{[k]}$), we first compute the global output based on the global shared TR-GAT with Eq. (12) to utilize the global shared inter-structural information of all KGs.

$$(\boldsymbol{h}^{(s)}, \boldsymbol{r}^{(s)}, \boldsymbol{t}^{(s)}) = \mathsf{TR\text{-}GAT}((E, R, T), \Theta^{(s)}) \tag{12}$$

Here, we use a TR-GAT$(\cdot, \cdot)$ as a shorthand for convolution process (Eq. (9)–(11)) in the global shared layer and Θ represents the parameters of global shared layer which are shared by all KGs. Then such global information is integrated into each task to enhance the embeddings of each KG. Formally, for a specific task k, the energy function in Eq. (3) is modified as follows:

$$(h^{(k)} + M_h h^{(s)}) + (r^{(k)} + M_r r^{(s)}) \approx (t^{(k)} + M_t t^{(s)}) \tag{13}$$

$$E(h, r, t)^{[k]} = \left\| (h^{(k)} + M_h h^{(s)}) + (r^{(k)} + M_r r^{(s)}) - (t^{(k)} + M_t t^{(s)}) \right\| \tag{14}$$

where $h^{(k)}, r^{(k)}, t^{(k)}$ are the task specific embeddings of knowledge graph k, $h^{(s)}, h^{(s)}, h^{(s)}$ are global embeddings obtained through the shared TR-GAT layer, and M_h, M_r, M_t are transform matrix to guarantee the consistency of the vector spaces between the embeddings of task-specific and global shared layers. In this way, every entity and relation of each single knowledge graph can benefit from common knowledge and extra information from all other knowledge graphs.

3.5 Training

Following the translational models, to learn such embeddings, we adopt a margin-based loss function respect to the triples for all tasks:

$$L = \sum_{k \in K} \sum_{(h,r,t) \in S} \sum_{(h',r',t') \in S} [\gamma + d(h + r, t) - d(h' + r, t')]_+ \tag{15}$$

$$h = h^{(k)} + M_h h^{(s)}; r = r^{(k)} + M_r r^{(s)}; t = t^{(k)} + M_t t^{(s)} \tag{16}$$

where K is the number of knowledge graphs.

4 Experiments

In this section, we evaluate the performance of our framework. We conduct an extensive empirical evaluation of multi-version TransMTL based on TransE, TransH and TransR respectively to verify the effectiveness of the proposed model. We further vary the training set size to illustrate that our proposed multi-task learning framework can still perform well on low resource settings.

4.1 Experiment Setup

Data Sets. Our model is evaluated on two widely used knowledge graphs: WordNet [16] and Freebase [1]. WordNet provides semantic knowledge of words. Entities in WordNet are synonyms which express distinct concepts. Relations in WordNet are conceptual-semantic and lexical relations. In this paper, we employ the dataset WN18RR [8] from WordNet. Freebase provides general facts of the world. In this paper, we employ data set FB15k-237 [27] from Freebase.

Notably, WN18RR and FB15k-237 are correspondingly subsets of two common data sets WN18 and FB15k. It is firstly discussed by [27] that WN18 and FB15k suffer from test leakage through inverse relations, i.e. many test triplets can be obtained simply by inverting triplets in the training set. To address this issue, Toutanova and Chen et al. [27] generated FB15k-237 by removing redundant relations in FB15k and greatly reducing the number of relations. Likewise, Dettmers et al. [8] removed reversing relations in WN18. As a consequence, the difficulty of reasoning on these two data sets is increased dramatically. The statistics of the two datasets are described in Table 1.

Table 1. Statistics of the datasets

Dataset	#Relation	#Entity	#Train	#Valid	#Test
WN18RR	11	40,943	86,835	3,034	3,134
FB15k-237	237	14,541	272,115	17,535	20,466

Baselines. We first compared our TransMTL with the corresponding single-task models, namely TransE [3], TransH [30] and TransR [13] respectively. To further illustrate the effectiveness of multi-task learning, we then compared our model with recent knowledge embedding models, including both non-neural models and neural models. Specifically, we compared our TransMTL with DistMult [32], ConvE [8], ComplEx [28], KBGAT [17], ConvKB [19], RotatE [25] and DensE [14] for comparison.

Evaluation Protocol. Link prediction aims to predict the missing h or t for a triplet (h, r, t). In this task, the model is asked to rank a set of candidate entities from the KG, instead of giving one best result. For each test triplet (h, r, t), we replace the head/tail entity by all possible candidates in the KG, and rank these entities in ascending order of scores calculated by score function showed in Eq. 14. We follow the evaluation protocol in [3] to report filtered results. Because a corrupted triplet, generated in the aforementioned process of removal and replacement, may also exist in KG, and should be considered as correct. In other words, while evaluating on test triples, we filter out all the valid triples from the candidate set, which is generated by either corrupting the head or tail entity of a triple. We report three common measures as our evaluation metrics: the average rank of all correct entities (Mean Rank), the mean reciprocal rank of all correct entities (MRR), and the proportion of correct entities ranked in top 10 (Hits@10). We report average results across 5 runs. We note that the variance is substantially low on all the metrics and hence omit it. A good link predictor should achieve lower Mean Rank, higher MRR,

Training Protocol. We use the common Bernoulli strategy [13,30] when sampling invalid triples. We select 500 as the batch size, which is not too big or too small for both the two datasets. There are two learning rates in our multi-task model: one for the global shared layer and the other for the task specific knowledge embedding layer. We use grid search method to find the appropriate learning rate for the two parts. And finally in our experiments, we use learning rate 0.5 for task-specific knowledge embedding layer and 0.01 for the global shared TR-GAT model. We use the Stochastic Gradient Descent (SGD) optimizer for training. In our model, the embedding size of entities and relations from the two knowledge graphs should be equal and we set it to 200. We use a two-layer GAT for the global shared TransMTL model that allows message passing among nodes that are two hops away from each other. As a result, although for some entity pairs, there are no direct edges in the knowledge graph, the two-layer

GAT is still capable to learn the inter-entity relations and enables the information exchange between pairs of entities. In our preliminary experiment, we found that a two-layer GAT performs better than a one-layer GAT, while more layers do not improve the performances. We set the dropout rate as 0.1 in order to release overfitting.

For multi-task learning, the training data come from completely different datasets, so our training process is conducted by looping over the tasks as follow:

1. Select a random task.
2. Select a mini-batch of examples from this task.
3. Backward the model and update the parameters of both task-specific layer and global shared layer with respect to this mini-batch.
4. Go to 1.

4.2 Results and Analysis

Table 2. Link prediction results of WN18RR and FB15k-237 compared with translational models. [*]: Results are taken from [19]. Best scores are highlighted in bold.

Models	WN18RR			FB15k-237		
	MR	MRR	Hit@10 (%)	MR	MRR	Hit@10 (%)
TransE[*]	3384	0.226	50.1	347	0.294	46.5
TransH	3048	0.286	50.3	348	0.284	48.8
TransR	3348	0.303	51.3	310	0.310	50.6
TransMTL-E	3065	0.363	54.1	116	0.336	52.6
TransMTL-H	**2521**	**0.498**	**57.0**	**111**	**0.349**	**53.7**
TransMTL-R	3154	0.465	54.6	133	0.333	52.2

Table 2 compares the experimental results of our TransMTL with different task specific knowledge embedding models to the corresponding single-task based models, using the same evaluation protocol. Here, TransMTL-E, TransMTL-H and TransMTL-R are models with TransE, TransH and TransR as their task specific knowledge embedding models respectively. From the table, we can see that our multi-task learning models outperform the corresponding single-task based models by an obvious margin. Specifically, TransMTL-E shows an improvement of Hit@10 4%, 6.1% to TransE on WN18RR and FB15k-237 respectively. TransMTL-H shows an improvement of Hit@10 6.7%, 4.9% to TransH and such numbers are 3.3% and 1.6% for the pair of TransMTL-R and TransR on dataset WN18RR and FB15k-237. Moreover, our TransMTL also obtains better MR and MRR scores than single-task models on both datasets. We argue that it is because with the global shared TR-GAT layer, the entities and relations of each single task can benefit from extra information from other tasks for better representations.

To further illustrate the effectiveness of multi-task learning, we then compared our model with recent knowledge embedding models, including both non-neural models and neural models. The experimental results are shown in Table 3. Since the datasets are same, we directly copy the experiment results of several baselines from [14,17]. From the table, we can see that even with the basic translational models, our TransMTL can obtain comparable performance to these recent models that integrate much additional information and new technologies into their models. Moreover, our TransMTL performs better on FB15k-237 than on WN18RR. The reason may be that there are rich conceptual-semantic and lexical relations in WN18RR and the entities and relations in FB15k-237 can benefit from these information through multi-task learning.

Table 3. Link prediction results for WN18RR and FB15k-237. Best scores are highlighted in bold.

Models	WN18RR			FB15k-237		
	MR	MRR	Hit@10 (%)	MR	MRR	Hit@10 (%)
DistMult	5110	0.430	49.0	512	0.281	44.6
ConvE	4187	0.43	52.0	244	0.325	50.1
ComplEx	7882	0.449	53.0	546	0.278	45.0
KBGAT	1921	0.412	55.4	270	0.157	33.1
ConvKB	**1295**	0.265	55.8	216	0.289	47.1
RotatE	3340	0.476	57.1	177	0.338	53.3
DensE	3052	0.491	**57.9**	169	0.349	53.5
TransMTL-E	3065	0.363	54.1	116	0.336	52.6
TransMTL-H	2521	**0.498**	57.0	**111**	**0.349**	**53.7**
TransMTL-R	3154	0.465	54.6	133	0.333	52.2

Varying the Data Size. In order to illustrate the robustness of our proposed multi-task learning framework, we vary the data sizes by randomly sampling different ratios of the training data for training and test them on the whole test sets of two datasets. Figure 4 shows the experimental results of Hit@10 scores of our TransMTL and the corresponding single task translational models: TransE, TransH, and TransR on two datasets respectively. Here, for the performance of single task translational models in different training data sizes, we use the same settings to the task-specific models of our TransMTL. From the figure, we can readily see that TransMTL consistently outperforms all single-task models across all datasets. Besides, we can see that with the decrease of training triples, the Hit@10 metrics decrease with different degrees. More specifically, the performance gap between our TransMTL models and the baselines are larger in small dataset settings than in big dataset settings. For example, with only the 60% of the training data, the performance of our TransMTL is still competitive, which shows an improvement of Hit@10 8.5% and 8% to TransE on WN18RR and FB15k-237 respectively. We argue that this is because our multi-task learning framework can exploit the underlying inter-structural relationships that are inherently presented in different knowledge graphs, thus it can alleviate the data insufficiency problem and achieve good results with less data.

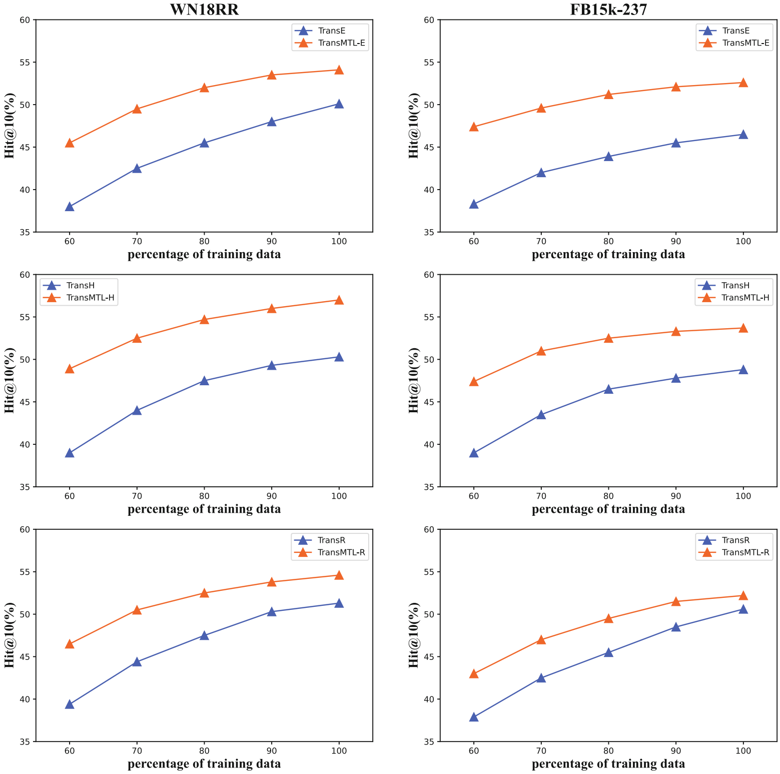

Fig. 4. The Hit@10 with different percentage of training data

5 Conclusions

In this paper, we propose a novel embedding model based on multi-task learning that can jointly learn multiple knowledge graph embeddings simultaneously for knowledge graph completion. We devise a global translation preserved relational graph attention network which is shared by all knowledge graphs to capture and transfer structural knowledge between different KGs. To preserve the transition property of each KG, we then integrate the global information learned by the global shared layer into the translational models for each KG. Experimental results on two benchmark datasets WN18RR and FB15k-237 show that our proposed model outperforms the corresponding single-task based models by an obvious margin and obtains the comparable performance to state-of-the-art embedding models, indicating the effectiveness of multi-task learning on knowledge graph representations.

Acknowledgements. This work was supported by NSFC (91646202), National Key R&D Program of China (2018YFB1404401, 2018YFB1402701).

References

1. Bollacker, K.D., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: SIGMOD, pp. 1247–1250 (2008)
2. Bordes, A., Chopra, S., Weston, J.: Question answering with subgraph embeddings. In: EMNLP, pp. 615–620 (2014)
3. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: NIPS, pp. 2787–2795 (2013)
4. Bordes, A., Weston, J., Collobert, R., Bengio, Y.: Learning structured embeddings of knowledge bases. In: AAAI (2011)
5. Caruana, R.: Multitask learning. Mach. Learn. **28**(1), 41–75 (1997)
6. Chapelle, O., Shivaswamy, P.K., Vadrevu, S., Weinberger, K.Q., Zhang, Y., Tseng, B.L.: Multi-task learning for boosting with application to web search ranking. In: ACM SIGKDD, pp. 1189–1198 (2010)
7. Chowdhury, S., Zhang, C., Yu, P.S.: Multi-task pharmacovigilance mining from social media posts. In: WWW, pp. 117–126 (2018)
8. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2D knowledge graph embeddings. In: AAAI, pp. 1811–1818 (2018)
9. Gardner, M., Talukdar, P.P., Krishnamurthy, J., Mitchell, T.M.: Incorporating vector space similarity in random walk inference over knowledge bases. In: EMNLP, pp. 397–406 (2014)
10. He, H., Balakrishnan, A., Eric, M., Liang, P.: Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In: ACL, pp. 1766–1776 (2017)
11. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic mapping matrix. In: ACL, pp. 687–696 (2015)
12. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2017)
13. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Bonet, B., Koenig, S. (eds.) AAAI, pp. 2181–2187. AAAI Press (2015)
14. Lu, H., Hu, H.: Dense: An enhanced Non-Abelian group representation for knowledge graph embedding. CoRR abs/2008.04548 (2020)
15. Luo, L., et al.: Beyond polarity: interpretable financial sentiment analysis with hierarchical query-driven attention. In: IJCAI, pp. 4244–4250 (2018)
16. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)
17. Nathani, D., Chauhan, J., Sharma, C., Kaul, M.: Learning attention-based embeddings for relation prediction in knowledge graphs. In: ACL, pp. 4710–4723 (2019)
18. Neelakantan, A., Roth, B., McCallum, A.: Compositional vector space models for knowledge base completion. In: ACL, pp. 156–166 (2015)

19. Nguyen, D.Q., Nguyen, T.D., Nguyen, D.Q., Phung, D.Q.: A novel embedding model for knowledge base completion based on convolutional neural network. In: NAACL-HLT, pp. 327–333 (2018)
20. Nguyen, D.Q., Vu, T., Nguyen, T.D., Nguyen, D.Q., Phung, D.Q.: A capsule network-based embedding model for knowledge graph completion and search personalization. In: NAACL-HLT, pp. 2180–2189 (2019)
21. Nguyen, D.Q., Sirts, K., Qu, L., Johnson, M.: Neighborhood mixture model for knowledge base completion. In: CoNLL, pp. 40–50 (2016)
22. Nickel, M., Rosasco, L., Poggio, T.A.: Holographic embeddings of knowledge graphs. In: AAAI, pp. 1955–1961 (2016)
23. Schlichtkrull, M.S., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: ESWC, pp. 593–607 (2018)
24. Socher, R., Chen, D., Manning, C.D., Ng, A.Y.: Reasoning with neural tensor networks for knowledge base completion. In: NIPS, pp. 926–934 (2013)
25. Sun, Z., Deng, Z., Nie, J., Tang, J.: Rotate: Knowledge graph embedding by relational rotation in complex space. In: ICLR (2019)
26. Tian, B., Zhang, Y., Wang, J., Xing, C.: Hierarchical inter-attention network for document classification with multi-task learning. In: IJCAI, pp. 3569–3575 (2019)
27. Toutanova, K., Chen, D.: Observed versus latent features for knowledge base and text inference. In: CVSM (2015)
28. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. In: ICML, pp. 2071–2080 (2016)
29. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: ICLR (2018)
30. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: AAAI, pp. 1112–1119 (2014)
31. West, R., Gabrilovich, E., Murphy, K., Sun, S., Gupta, R., Lin, D.: Knowledge base completion via search-based question answering. In: WWW, pp. 515–526 (2014)
32. Yang, B., Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: ICLR (2015)
33. Yim, J., Jung, H., Yoo, B., Choi, C., Park, D., Kim, J.: Rotating your face using multi-task deep neural network. In: IEEE CVPR, pp. 676–684 (2015)
34. Zhang, F., Yuan, N.J., Lian, D., Xie, X., Ma, W.: Collaborative knowledge base embedding for recommender systems. In: ACM SIGKDD, pp. 353–362 (2016)
35. Zhao, K., et al.: Modeling patient visit using electronic medical records for cost profile estimation. In: DASFAA, pp. 20–36 (2018)
36. Zhao, K., et al.: Discovering subsequence patterns for next POI recommendation. In: IJCAI, pp. 3216–3222 (2020)