Multilingual Knowledge Graph Completion with Language-Sensitive Multi-Graph Attention

Rongchuan Tang[1,2], **Yang Zhao**[1,2], **Chengqing Zong**[1,2], **Yu Zhou**[1,3]*
[1] State Key Laboratory of Multimodal Artificial Intelligence Systems,
Institute of Automation, CAS, Beijing, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3] Fanyu AI Laboratory, Zhongke Fanyu Technology Co., Ltd, Beijing, China
{tangrongchuan2021@,zhaoyang2015@,cqzong@nlpr.,yzhou@nlpr.}ia.ac.cn

Abstract

Multilingual Knowledge Graph Completion (KGC) aims to predict missing links with multilingual knowledge graphs. However, existing approaches suffer from two main drawbacks: (a) *alignment dependency*: the multilingual KGC is always realized with joint entity or relation alignment, which introduces additional alignment models and increases the complexity of the whole framework; (b) *training inefficiency*: the trained model will only be used for the completion of one target KG, although the data from all KGs are used simultaneously. To address these drawbacks, we propose a novel multilingual KGC framework with language-sensitive multi-graph attention such that the missing links on all given KGs can be inferred by a universal knowledge completion model. Specifically, we first build a relational graph neural network by sharing the embeddings of aligned nodes to transfer language-independent knowledge. Meanwhile, a language-sensitive multi-graph attention (LSMGA) is proposed to deal with the information inconsistency among different KGs. Experimental results show that our model achieves significant improvements on the DBP-5L and E-PKG datasets.[1]

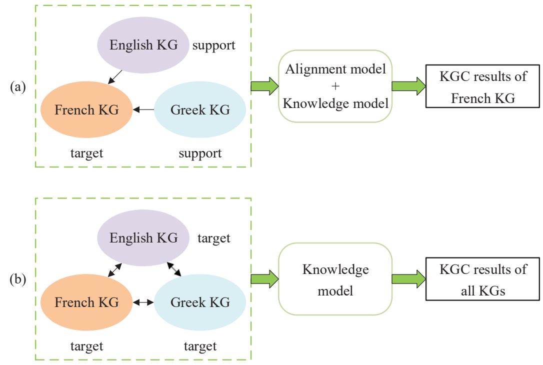

Figure 1: An example of multilingual KGC. (a) Existing methods try to improve the KGC task on a French KG by utilizing two support KGs and an external alignment model is included. (b) Our goal is to simultaneously improve the KGC task on all given KGs by only one knowledge model.

1 Introduction

Knowledge graphs (KGs) with plentiful structured semantic information have been widely used in various NLP applications such as question answering (Saxena et al., 2020; Ren et al., 2021), recommender systems (Wang et al., 2021a, 2022b) and information extraction (Hu et al., 2021; Zong et al., 2021). Due to the well-known incompleteness of KG, the task of KG completion seeks to facilitate the automatic construction of knowledge graphs by predicting missing links (Bordes et al., 2013; Balažević et al., 2019; Zhu et al., 2021b; Wang et al., 2022a). Recently, there has been a lot of interest in improving KG completion by leveraging KGs from different languages. Known as multilingual knowledge graph completion (KGC), various attempts have been made to transfer knowledge from one KG to another, such as KEnS (Chen et al., 2020), AlignKGC (Singh et al., 2021), and SS-AGA (Huang et al., 2022). And these studies have demonstrated that it is viable to improve the single KGC task by utilizing information from other language-specific KGs.

However, the methods for multilingual KGC mainly involve two shortcomings. The first one is referred to be *alignment dependency*, indicating that in previous frameworks, the multilingual KGC task has to be carried out in conjunction with entity or relation alignment tasks. This leads to a cumbersome framework that always consists of two separate models, one for alignment and the other for completion, and the multilingual KGC task can not be trained alone. In addition, the alignment model will increase extra computational and memory costs, which are usually comparable to the knowledge model.

*Corresponding author.
[1]Our codes are available at https://github.com/RongchuanTang/LSMGA-MKGC

The second shortcoming is called *training inefficiency*: although multiple KGs are given, the knowledge is only propagated from support KGs (the KGs to provide knowledge of other languages) to one target KG (the KG to be completed). In this way, if we wish to perform the completion task on another KG, we have to re-designate that KG as the target and retrain entirely, thus making the training process far from efficient. As the example shown in Figure 1(a), the French KG is the target KG and two support KGs in English and Greek are available. Prior methods output the KGC results only for the French KG through a framework including an alignment model and a knowledge model.

According to the multilingual machine translation (MT) task (Johnson et al., 2017; Zhang and Zong, 2020), it is capable of building a unified MT model to translate multiple languages. Since multilingual KGC and multilingual MT are both related to knowledge transfer across languages, we attempt to investigate the multilingual KGC in a compact manner without a redundant alignment model while using a universal model to infer the missing facts in all given KGs. Like depicted in Figure 1(b), the completion results for all KGs are expected to be obtained by only one knowledge model.

Motivated by the discussions above, we propose a graph neural network with language-sensitive multi-graph attention (LSMGA) for multilingual KGC. First, several separate KGs are connected into a single graph by sharing the embedding of aligned entities. Second, we design a language-specific multi-graph attention to better capture different patterns stored in different graphs. At last, a language-sensitive aggregation module is utilized to integrate the information from multiple sources. Experimental results show that our approach achieves better results than previous methods on the multilingual KGC task. It indicates that our framework can take full advantage of all given KGs by using a universal knowledge model.

Our main contributions are summarized as follows:

- A novel framework for the multilingual KGC task to predict the KGC results of all given KGs by only using one knowledge model is proposed by forcing the aligned entities to share the same embedding and treating each source KG equally.
- A graph neural network based on language-sensitive multi-graph attention is put forward to capture the different knowledge patterns of the KGs from different languages and make the knowledge transfer among different KGs in distinct ways.
- Experiments have been conducted to validate the effectiveness of our multilingual KGC framework, and the results show that our LSMGA-based graph neural network achieves significant improvements over existing approaches on the multilingual KGC task.

2 Related Work

Knowledge Graph Completion involves predicting missing links based on the existing facts, which are usually from a single KG. Translation-based methods (Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015; Ji et al., 2015) establishes different geometric relationships for the triples, and then design a score function to evaluate the plausibility of the triples. In order to capture the deeper information of the facts, DKRL (Xie et al., 2016) and ConMASK (Shi and Weninger, 2018) adopt convolutional neural networks to extract features from the text descriptions of entities and realize open-world knowledge graph completion. Another type of work, such as KE-BERT (Yao et al., 2019), KEPLER (Wang et al., 2021b), and SimKGC (Wang et al., 2022a), has attempted to combine knowledge graph embeddings with pre-trained language models and achieved some promising results on the KGC task. Recently, the methods based on graph neural networks (GNNs) (Schlichtkrull et al., 2018; Zhu et al., 2021b; Zhang and Yao, 2022) have shown great potential in knowledge graph completion, due to GNN's powerful ability to model graph structures (Wu et al., 2021; Cai et al., 2021).

Multilingual Knowledge Graph Completion aims to boost KGC with multiple KGs that are in different languages. For the first time, the MTransE (Chen et al., 2017) model extended knowledge graph embeddings from a monolingual scenario to a multilingual scenario, where the information of multiple KGs can be transferred to each other. After then, a lot of work focused on entity alignment task between different KGs (Zhang et al., 2019; Sun et al., 2020; Zhu et al., 2021a; Guo et al., 2022). On the other hand, (Chen et al., 2020) proposes a new framework KEnS, which improves monolin-

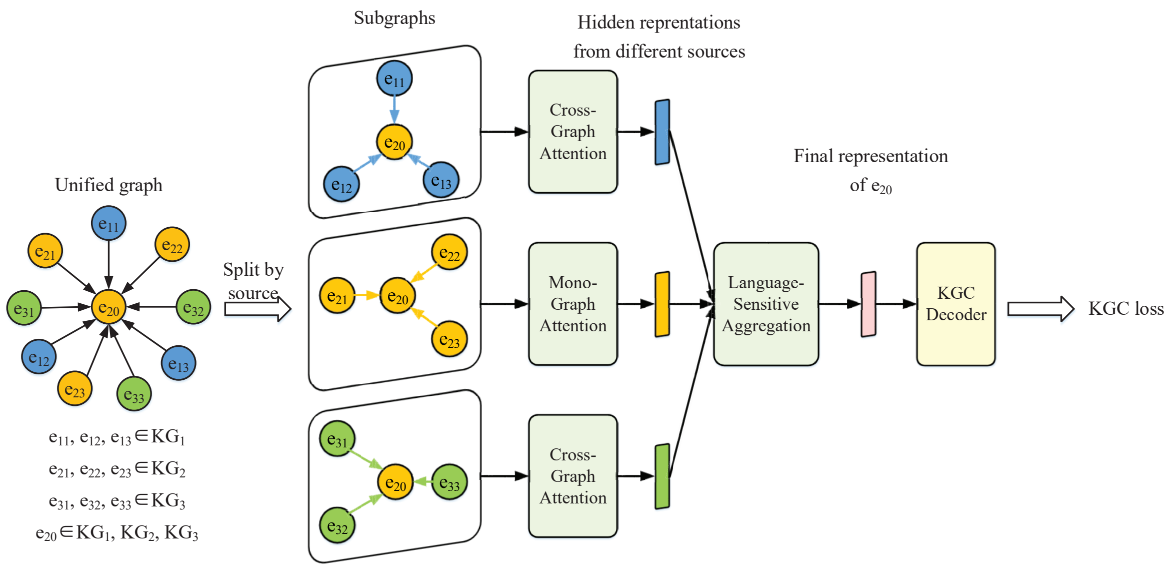

Figure 2: The overall architecture of our language-sensitive multi-graph attention which contains three KGs. The four main components are: (i) constructing the unified graph; (ii) using multi-graph attention to capture different information from different sources; (iii) using language-sensitive aggregation to integrate the information from (ii); (iv) computing the scores of triples by a KGC decoder.

gual KGC by effectively leveraging complementary knowledge of multiple language-specific KGs. AlignKGC (Singh et al., 2021) performs KGC together with entity alignment and relation alignment on multilingual KGs and improves KGC accuracy, as well as alignment scores. SS-AGA (Huang et al., 2022) improves the multilingual KGC task by using a relation-aware graph neural network and dynamically generating more potential alignment pairs. However, entity alignment is still the primary focus of the aforementioned methods for multilingual KGC. Additionally, the knowledge models in above works are almost built upon the methods for monolingual KGC, while few works address multilingual knowledge transfer directly. In this paper, we are going to address both *alignment dependency* and *training inefficiency*.

3 Notations and preliminaries

A knowledge graph is denoted as $G = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, where $\mathcal{E}$ is the set of entities, $\mathcal{R}$ is the set of relations and $\mathcal{T}$ is the set of triples. A fact is in the form of a triple (h, r, t) consisting of a head entity h, a relation r and a tail entity t, where $h, t \in \mathcal{E}$ and $r \in \mathcal{R}$.

Knowledge graph completion is the task of predicting new facts based on the existing facts in a single KG. Usually, the system needs to answer a query like $(h, r, ?)$ or $(?, r, t)$ by inferring the missing tail entity or head entity.

Multilingual knowledge graph completion is the task of predicting new facts based on the existing facts in multiple KGs with different languages. The concrete situation is that there are N KGs with N different languages as $G_1, G_2, \cdots, G_N$ and between any two KGs $G_i = (\mathcal{E}_i, \mathcal{R}_i, \mathcal{T}_i)$ and $G_j = (\mathcal{E}_j, \mathcal{R}_j, \mathcal{T}_j)$, a limited number of aligned entity pairs as $\{(e_i, e_j) : e_i \in \mathcal{E}_i, e_j \in \mathcal{E}_j\}$ (e denotes an entity) are known in advance. Besides, all the relations are represented within a unified schema $\mathcal{R}$, i.e. $\mathcal{R}_i \in \mathcal{R}$ for $i = 1, 2, \cdots, N$.

4 Methodology

The overall architecture is illustrated in Figure 2. Four components are included in the whole framework:

Creating the unified graph. Assuming N source KGs: $G_1, G_2, \cdots, G_N$, we force the aligned entity pairs e_i and e_j to be representated by the same embedding vector. In this way, N separate KGs are linked into a unified graph G_u and the duplicate aligned entities are removed in the unified entity vector set $\mathcal{E}_u$. Besides, we maintain a unified relation vector set $\mathcal{R}$. By sharing the aligned entities, there is no need to introduce an alignment model in our framework. And since the given KGs are treated equally on the unified graph, we are able to train a model in one go that can be used for the inference on all given KGs. Therefore, creating the unified graph is a simple but crucial step to deal with both *alignment dependency* and *training inefficiency*.

GNN encoder with multi-graph attention. We split the neighbor nodes of the target node into N subgraphs by different sources and encode the target node into N hidden representations by a novel GNN with multi-graph attention.

Language-sensitive aggregation. With multiple outputs from the GNN encoder, the final representation of the target node is computed by a language-sensitive aggregation module.

KGC decoder. Given the embeddings of entities and relations, the scores of triplets are computed by a KGC decoder and then the KGC loss to be optimized can be obtained.

Remark 1. By creating the unified graph, our method could complete all the KGs simultaneously through a shared knowledge model (sharing both aligned entities and relations). To complete the N KGs, the number of entity and relation embeddings in our framework is $|\mathcal{E}_u| + |\mathcal{R}|$, while previous frameworks like SS-AGA (Huang et al., 2022) is $2 * N * \left(\sum_{i}^{N} |\mathcal{E}_i| + N * |\mathcal{R}|\right)$ (including the knowledge model and the alignment model). By utilizing a smaller model size, our approach alleviates the scalability challenges posed by massive KGs, making it more feasible and efficient for real-world applications.

4.1 GNN with Multi-Graph Attention

Considering the powerful representation ability of GNN networks for graph structure, we try to build a GNN model after creating the unified graph. A staright-forward way is to learn directly on the unified graph with a commonly used GNN encoder. However, there is inevitably lots of repetitive knowledge among different KGs since knowledge is language-independent, and simple aggregation will cause the model to reduce the weight of those different parts that can reflect the characteristics of KGs. On the other hand, as each KG has its own unique knowledge pattern, there should be different approaches taken when transferring knowledge across them. In fact, similar multilingual attention mechanism has been practiced in the relation extraction task (Lin et al., 2017).

Explicitly, our model adopts two kinds of graph attention mechanisms for multilingual KGC, including (a) mono-graph attention to select the neighbor nodes within one language and (b) cross-graph attention to select the neighbor nodes among different languages.

4.1.1 Mono-Graph Attention

At first, we follow the idea of multi-layer relation-aware message passing architecture proposed in (Huang et al., 2022):

$$h_i^{l+1} = h_i^l + \sigma\Big(\sum_{e_j \in \mathrm{N}_{n_i}(e_i)} \mathrm{Att}(h_i^l, h_{j(r)}^l) h_{j(r)}^l\Big), \tag{1}$$

where h_i^l indicates the hidden representation of entity e_i at the l-th layer, $\sigma(\cdot)$ is a non-linear activation function, $\mathrm{N}_{n_i}(e_i)$ indicates the neighbor node set from the n_i-th source KG of e_i, $h_{j(r)}^l$ indicates the relation-aware message conveyed by entity e_j in a relational triple (e_i, r, e_j) and $\mathrm{Att}(h_i^l, h_{j(r)}^l)$ is the attention score of each message from neighbor nodes. It should be noted that h_i^l and h_j^l in this subsection are abbreviations of $h_{in_i}^l$ and $h_{jn_i}^l$ in brief.

Since each KG has its own characteristics, it is intuitive that we adopt different mono-graph attentions to weight the neighbor information within each language. Specifically, when the target node e_i and its neighbor nodes are from the same n_i-th source KG like in the second subgraph in Figure 2, the neighbor message $h_{j(r)}^l$ is calculated as:

$$h_{j(r)}^l = W_{vn_i}^l \mathrm{Concat}(h_j^l, r), \tag{2}$$

where $W_{vn_i}^l \in \mathbb{R}^{d \times 2d}$ is a transformation matrix of the n_i-th KG (d is the dimension of the embeddings of entities and relations), $\mathrm{Concat}(\cdot)$ is a vector concatenation function. Then the attention score is defined as:

$$\mathrm{Att}(h_i^l, h_{j(r)}^l) = \frac{\exp(\alpha_{ij}^r)}{\sum_{e_{j'} \in \mathrm{N}_{n_i}(e_i)} \exp(\alpha_{ij'}^r)}, \tag{3}$$

where α_{ij}^r is referred as a function which scores the significance of neighbor nodes to the target node. Here α_{ij}^r is computed as:

$$\alpha_{ij}^r = \frac{\beta_r}{\sqrt{d}} (W_{qn_i}^l h_i^l)^T (W_{kn_i}^l h_{j(r)}^l), \tag{4}$$

where β_r is a learnable relation variable to weigh the importance of relation r (Huang et al., 2022), and $W_{qn_i}^l \in \mathbb{R}^{d \times d}$, $W_{kn_i}^l \in \mathbb{R}^{d \times d}$ are two transformation matrices of the n_i-th KG.

4.1.2 Cross-Graph Attention

Besides mono-graph attention, we propose cross-graph attention for multilingual KGC in order to better make use of multi-lingual KGs. The key idea of cross-graph attention is that knowledge transfer between different knowledge graphs should be performed in different ways. Hence, cross-graph attention is proposed to aggregate the information from other KGs in different languages.

Cross-graph attention works similarly to mono-graph attention. Assume that the target node e_i is from the n_i-th KG and its neighbor nodes are from the n_j-th KG ($j \neq i$) (e.g., the first subgraph in Figure 2). Formally, the cross-graph representation is updated as:

$$h_i^{l+1} = h_i^l + \sigma(\sum_{e_j \in \mathrm{N}_{n_j}(e_i)} \mathrm{Att}(h_i^l, h_{j(r)}^l) h_{j(r)}^l), \tag{5}$$

where $\mathrm{N}_{n_j}(e_i)$ indicates the neighbor node set from the n_j-th KG of e_i and h_i^l and h_j^l in this subsection are abbreviations of $h_{in_j}^l$ and $h_{jn_j}^l$ It should be noted that h_i^l and h_j^l in this subsection are abbreviations of $h_{in_i}^l$ and $h_{jn_i}^l$ in brief. The neighbor message $h_{j(r)}^l$ is calculated as:

$$h_{j(r)}^l = W_{vn_j}^l \mathrm{Concat}(h_j^l, r), \tag{6}$$

where $W_{vn_j}^l \in \mathbb{R}^{d \times 2d}$ is a transformation matrix of the n_j-th KG. Then the attention score is defined as follows:

$$\mathrm{Att}(h_i^l, h_{j(r)}^l) = \frac{\exp(\alpha_{ij}^r)}{\sum_{e_{j'} \in \mathrm{N}_{n_j}(e_i)} \exp(\alpha_{ij'}^r)}. \tag{7}$$

Similar to the mono-graph attention, we calculate α_{ij}^r as:

$$\alpha_{ij}^r = \frac{\beta_r}{\sqrt{d}} (W_{qn_i}^l h_i^l)^T (W_{kn_j}^l h_{j(r)}^l), \tag{8}$$

where $W_{qn_i}^l \in \mathbb{R}^{d \times d}$, $W_{kn_j}^l \in \mathbb{R}^{d \times d}$ are the corresponding transformation matrices of the n_i-th KG and n_j-th KG, respectively.

4.2 Language-Sensitive Aggregation

After getting the hidden representations from different source KGs, we need an efficient aggregation module to integrate the information from multiple sources. In order to ensure the model explicitly knows which language-specific KG the embedding belongs to, we propose a method similar to that used in multilingual machine translation (Firat et al., 2016; Zhang et al., 2020), that is, adding a language tag to each language-specific information. As a result, we design a language-sensitive aggregation module to get the final representation of the target node. To begin, we denote the multiple vectors output by multi-graph attention as $h_1, h_2, \cdots, h_N$ and the output vector of the mono-graph attention is also denoted as h_t. Second, we maintain N language vectors $h_{k1}, h_{k2}, \cdots, h_{kN}$, as the indicators of each language. Then, the i-th representation with the corresponding language indicator is defined as:

$$h_{vi} = \mathrm{Concat}(h_i, h_{ki}). \tag{9}$$

The vectors in Eq. (9) are used to calculate the weights of different languages to be aggregated into the target one. Finally, the final representation of the target node is calculated as follows:

$$h_{tf} = h_t + \sigma(\sum_{i \in 1,2,\cdots,N} \mathrm{Att}(h_t, h_i) W_v h_i), \tag{10}$$

$$\mathrm{Att}(h_t, h_i) = \frac{\exp(\alpha_{ti})}{\sum_{j \in 1,2,\cdots,N} \exp(\alpha_{tj})}, \tag{11}$$

$$\alpha_{ti} = \frac{1}{\sqrt{d}} (W_q h_{vt})^T (W_k h_{vi}), \tag{12}$$

where W_q, W_k, W_v are three transformation matrices.

4.3 KGC Decoder

Given the embeddings of entities and relations, the score of a candidate triple could be calculated by a KGC decoder. In this paper, we adopt the score function proposed in TransE (Bordes et al., 2013) as:

$$\phi(h, r, t) = -\|h + r - t\|_2. \tag{13}$$

In order to increase the score of the correct triple (h, r, t) while decrease the score of the false triple (h, r, t'), we minimize the following margin-based ranking loss:

$$\mathcal{L} = \sum_{\substack{(h,r,t) \in \mathcal{T}, \\ (h,r,t') \notin \mathcal{T}}} \left[\lambda - \phi(h, r, t) + \phi(h, r, t')\right]_+, \tag{14}$$

where $\mathcal{T}$ is the triple set consisting of all triples from the given N KGs, $\lambda > 0$ is a margin hyperparameter and $[x]_+ = \max(x, 0)$.

5 Experiments and Analysis

We have conducted a series of experiments on the multilingual KGC task with our model and the results have been carefully analyzed.

5.1 Datasets and Evaluation Metrics

Datasets. We use two datasets for evaluation: DBP-5L (Chen et al., 2020) and E-PKG (Huang et al., 2022). The statistics are shown in Table 1. The DBP-5L dataset consists of five language-specific KGs extracted from DBpedia, including Greek (EL), English (EN), Spanish (ES), French (FR) and Japanese (JA). The E-PKG dataset is a multilingual E-commerce KG dataset about phone-related product information in six languages, including German (DE), English (EN), Spanish (ES), French (FR), Italian (IT) and Japanese (JA). The relations in both datasets are in English and shared across different language-specific KGs. The English KGs in both datasets are the largest, and the smallest are the Greek KG and the Japanese KG, respectively.

Evaluation Metrics. Following previous work, we evaluate our LSMGA-based model with the task of tail entity prediction. Concretely, we rank all entities in the candidate set to predict t given h and r for each triple (h, r, t) in the test set. Then three common evaluation metrics are reported, i.e. Hits@1, Hits@10 and mean reciprocal ranks (MRR). Hits@k computes the fraction of correct entities ranked within top-k, and MRR is the average reciprocal rank of all test instances. Besides, we adopt the *filtered setting* in (Bordes et al., 2013) which removes the scores of all known triples in the training, validation and test sets.

Lang.	#Entity	#Relation	#Train	#Valid	#Test
DBP-5L					
EL	5,231	111	8,670	4,152	1,017
EN	13,996	831	48,652	24,051	7,464
ES	12,382	144	33,036	16,220	4,810
FR	13,176	178	30,139	14,705	4,171
JA	11,805	128	17,979	8,633	2,162
E-PKG					
DE	17,223	21	45,515	22,753	7,602
EN	16,544	21	60,310	39,150	10,071
ES	9,595	21	18,090	9,039	3,034
FR	17,068	21	47,999	23,994	8022
IT	15,670	21	42,767	21,377	7,148
JA	2,642	21	10,013	5,002	1,688

Table 1: The statistics of DBP-5L and E-PKG datasets.

5.2 Details of Implementation

During the training stage, we combine the instances in all training sets for training. There are two ways to select the optimal model, one is to select an optimal model via the average MRR on all validation sets and the other is to use the validation set of each KG separately to save the optimal model corresponding to that KG. In this paper, the experiments are carried out in the first way as depicted in Figure 3, which is consistent with the goal of this paper to implement KGC on all given KGs by using only one model.

Most hyperparameters are shared between both datasets. We use Adam (Kingma and Ba, 2015) optimizer to train our model. The embeddings of entities and relations are initialized randomly and their dimensions are 256, as is the hidden dimension of the GNN encoder. Besides, the layer of GNN is set as 2. Grid search is used to select the learning rate lr and the margin λ with ranges $lr = \{1\times10^{-4}, 5\times10^{-4}, 1\times10^{-3}, 5\times10^{-3}, 1\times10^{-2}\}$, $\lambda = \{0.2, 0.3, 0.5, 0.8\}$. The best learning rate is 5×10^{-3} for DBP-5L dataset and 1×10^{-3} for E-PKG dataset, while the best margin is 0.5 for DBP-5L and 0.3 for E-PKG. Models are trained with a batch size of 200 on one GeForce GTX 1080Ti GPU.

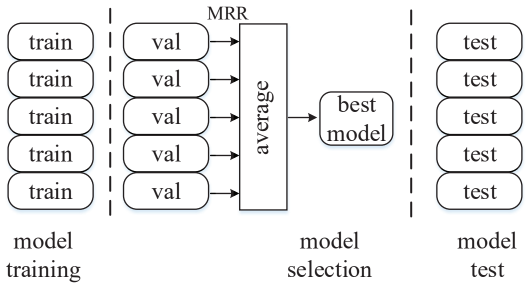

Figure 3: The training process. The best model is selected via the average MRR on all validation sets.

5.3 Main Results

In order to prove the effectiveness of our proposed method, we empirically compare different methods. For monolingual KGC methods, we choose the classic models TransE (Bordes et al., 2013), RotatE (Sun et al., 2019) and DisMult (Yang et al., 2015). In addition, we also include KG-BERT (Yao et al., 2019) which adopts pre-trained language models for KGC tasks. For multilingual KGC methods, we compare three recent related works: KEnS (Chen et al., 2020) ensembles knowledge transfer across

Method	EL			EN			ES			FR			JA		
	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR
Monolingual KGC methods															
TransE	13.1	43.7	24.3	7.3	29.3	16.9	13.5	45.0	24.4	17.5	48.8	27.6	21.1	48.5	25.3
RotatE	14.5	36.2	26.2	12.3	30.4	20.7	21.2	53.9	33.8	23.2	55.5	35.1	26.4	60.2	39.8
DistMult	8.9	11.3	9.8	8.8	30.0	18.3	7.4	22.4	13.2	6.1	23.8	14.5	9.3	27.5	15.8
KG-BERT	17.3	40.1	27.3	12.9	31.9	21.0	21.9	54.1	34.0	23.5	55.9	35.4	26.9	59.8	38.7
Multilingual KGC methods															
KEnS	28.1	56.9	-	15.1	39.8	-	23.6	60.1	-	25.5	62.9	-	32.1	65.3	-
AlignKGC	27.6	56.3	33.8	15.5	39.2	22.3	24.2	60.9	35.1	24.1	62.3	37.4	31.6	64.3	41.6
SS-AGA	30.8	58.6	35.3	16.3	41.3	23.1	25.5	61.9	36.6	27.1	65.5	38.3	**34.6**	66.9	42.9
LSMGA	**33.1**	**89.9**	**54.5**	**16.8**	**61.7**	**32.4**	**25.6**	**74.8**	**42.8**	**31.2**	**81.3**	**48.6**	33.5	**79.1**	**49.8**

Table 2: Main results for the DBP-5L dataset. H@k is a shorthand of Hits@k.

Method	DE			EN			ES			FR			IT			JA		
	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR
Monolingual KGC methods																		
TransE	21.2	65.5	37.4	23.2	67.5	39.4	17.2	58.4	33.0	20.8	66.9	37.5	22.0	63.8	37.8	25.1	72.7	43.6
RotatE	22.3	64.3	38.2	24.2	66.8	40.0	18.3	58.9	33.7	22.1	64.3	38.2	22.5	64.0	38.1	26.3	71.9	41.8
DistMult	21.4	54.5	35.4	23.8	60.1	37.2	17.9	46.2	30.9	20.7	53.5	35.1	22.8	51.8	34.8	25.9	62.6	38.0
KG-BERT	21.8	64.7	38.4	24.3	66.4	39.6	18.7	58.8	33.2	22.3	67.2	38.3	22.9	63.7	37.2	26.9	72.4	44.1
Multilingual KGC methods																		
KEnS	24.3	65.8	-	26.2	69.5	-	21.3	59.5	-	25.4	68.2	-	25.1	**64.6**	-	33.5	73.6	-
AlignKGC	22.1	65.1	38.5	25.6	68.3	40.5	19.4	59.1	34.2	22.8	67.2	38.8	24.2	63.4	37.3	31.2	72.3	46.2
SS-AGA	24.6	66.3	39.4	26.7	69.8	41.5	21.0	60.1	36.3	**25.9**	**68.7**	**40.2**	24.9	63.8	38.4	33.9	74.1	48.3
LSMGA	**30.7**	**68.5**	**44.8**	**31.9**	**70.2**	**45.9**	**23.1**	**61.1**	**36.5**	23.7	63.5	38.2	**26.8**	64.5	**41.0**	**43.7**	**78.4**	**57.1**

Table 3: Main results for the E-PKG dataset. H@k is a shorthand of Hits@k.

multiple language-specific KGs; AlignKGC (Singh et al., 2021) performs KGC together with entity alignment and relation alignment on multilingual KGs; SS-AGA (Huang et al., 2022) improves the multilingual KGC task by dynamically generating more potential alignment pairs.

The results are displayed in Table 2 and Table 3, and the figures of the methods for comparison are derived from (Huang et al., 2022). It should be noted that those methods employ mBERT (Devlin et al., 2018) to initialize the entity and relation embeddings from their textual descriptions. For the DBP-5L dataset, the performance of our LSMGA is much better than the baseline methods in most situations except Hits@1 metric on the Japanese KG, which strongly verifies the effectiveness of our proposed framework which only uses a single knowledge model. As for the E-PKG dataset, LSMGA achieves the best results in 5 out of 6 KGs and is comparable to the baselines on the French KG. Considering that we do not use the textual descriptions of the entities, our results are more competitive.

The Greek KG in DBP-5L and the Japanese KG in E-PKG are much smaller than other KGs in both datasets, with fewer entities and facts. And we can see from the results that the performance gains on the Greek KG and the Japanese KG are the two biggest compared to other KGs, with MRR rising from 35.3% to 54.5% and from 48.3% to 57.1%. This phenomenon demonstrates that the proposed multi-graph attention can very effectively take advantage of other language-specific KGs to improve the KGC performance of the KG in low-resource languages.

Table 4 shows the results of multilingual KGC on the remaining KGs after each KG has been removed from the DBP-5L dataset. Overall, we can see that removing any KG will reduce the performance of others. This further validates the complementarity of different language-specific KGs on the KGC task and also proves that our model can well realize knowledge transfer among multiple KGs. Moreover, it can be discovered that removing the English KG has the biggest impact on the overall performance, since the English KG provides the most information in the DBP-5L dataset.

5.4 Ablation Study

We conduct ablation studies to gain a deeper understanding of our model design. The models used for comparison are the following: (a) W/O MGA is the model to learn directly on the unified graph and the information from all given KGs are aggregated by

Method	EL			EN			ES			FR			JA		
	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR
Use all	**33.1**	**89.9**	**54.5**	**16.8**	**61.7**	**32.4**	**25.6**	**74.8**	**42.8**	**31.2**	**81.3**	**48.6**	**33.5**	**79.1**	**49.8**
-EL	-	-	-	15.6	62.6	31.9	21.7	73.4	39.7	28.8	79.8	46.6	31.6	77.8	47.8
-EN	21.0	86.0	45.0	-	-	-	18.8	68.0	35.9	22.3	75.6	41.2	22.8	76.4	42.3
-ES	23.9	82.6	45.7	13.0	55.9	27.8	-	-	-	25.0	73.8	41.7	29.1	76.8	46.3
-FR	27.5	87.4	49.7	13.5	58.5	28.7	19.8	69.0	36.9	-	-	-	26.4	74.4	42.9
-JA	27.1	83.6	48.1	15.0	60.1	30.5	23.0	72.5	40.2	24.2	75.0	42.0	-	-	-

Table 4: The complementarity among different KGs in DBP-5L dataset.

mono-graph attention rather than multi-graph attention; (b) W/O LSA is the model using multi-graph attention without language-sensitive aggregation, which means the multiple vectors output by multi-graph attention are aggregated without language-indicator; (c) Add-LSMGA uses addition instead of concatenation in Eq. (9); (d) Concat-LSMGA is our proposed method.

Method	Avg H@1	Avg H@10	Avg MRR
KEnS	24.9	57	-
AlignKGC	24.6	56.6	34.0
SS-AGA	26.9	58.8	35.2
W/O MGA	24.3	75.6	43.2
W/O LSA	23.0	77.0	42.3
Add-LSMGA	25.3	77.0	43.7
Concat-LSMGA	**28.1**	**77.4**	**45.6**

Table 5: Ablation on the DBP-5L dataset.

In order to have a more comprehensive comparison, three current methods are also included and the results are shown in Table 5. The DBP-5L dataset is used and the metrics are the average corresponding metrics of the five KGs. First, we can see that W/O MGA has made significant improvements on Hits@10 and MRR, which proves the effectiveness of sharing the aligned entities. Second, the results of W/O LSA decreases slightly on Hits@1 and MRR, indicating that only using multi-graph attention without language-indicators can not aggregate information from different KGs effectively. After combining the language-sensitive aggregation module, the performance outperforms the method without using multi-graph attention. Furthermore, our proposed LSMGA based on concatenation achieves the state-of-the-art.

6 Conclusion

In this paper, a multilingual knowledge graph completion model using a graph neural network with language-sensitive multi-graph attention has been proposed. We emphasize that the multilingual KGC could be implemented well without an alignment model. In addition, language-sensitive multi-graph attention allows knowledge transfer among multiple KGs to be carried out in different ways. Finally, the experiments on the DBP-5L and E-PKG datasets show that our framework achieves considerable improvement over existing methods.

Limitations

Since the unified graph is very large, it will take more time to construct the subgraphs before the first training. But after saving these subgraphs, there is no need to rebuild the subgraphs in the subsequent training process. On the other hand, the aligned entities among different KGs is a necessary condition for our proposed framework and otherwise, our model can not conduct knowledge transfer among the given KGs without an alignment model or other techniques.

Acknowledgements

We thank anonymous reviewers for helpful suggestions. This work was supported by the Natural Science Foundation of China under Grant No. 62006224.

References

Ivana Balažević, Carl Allen, and Timothy Hospedales. 2019. Tucker: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5185–5194.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Lei Cai, Jundong Li, Jie Wang, and Shuiwang Ji. 2021. Line graph neural networks for link prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. 2017. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *IJCAI*.

Xuelu Chen, Muhao Chen, Changjun Fan, Ankith Uppunda, Yizhou Sun, and Carlo Zaniolo. 2020. Multilingual knowledge graph completion via ensemble knowledge transfer. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3227–3238.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875.

Lingbing Guo, Qiang Zhang, Zequn Sun, Mingyang Chen, Wei Hu, and Huajun Chen. 2022. Understanding and improving knowledge graph embedding for entity alignment. In *International Conference on Machine Learning*, pages 8145–8156. PMLR.

Zikun Hu, Yixin Cao, Lifu Huang, and Tat-Seng Chua. 2021. How knowledge graph and attention help? a qualitative analysis into bag-level relation extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4662–4671.

Zijie Huang, Zheng Li, Haoming Jiang, Tianyu Cao, Hanqing Lu, Bing Yin, Karthik Subbian, Yizhou Sun, and Wei Wang. 2022. Multilingual knowledge graph completion with self-supervised adaptive graph alignment. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 474–485.

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.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, pages 1–13.

Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2017. Neural relation extraction with multi-lingual attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 34–43.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

Hongyu Ren, Hanjun Dai, Bo Dai, Xinyun Chen, Michihiro Yasunaga, Haitian Sun, Dale Schuurmans, Jure Leskovec, and Denny Zhou. 2021. Lego: Latent execution-guided reasoning for multi-hop question answering on knowledge graphs. In *International Conference on Machine Learning*, pages 8959–8970. PMLR.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4498–4507.

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.

Baoxu Shi and Tim Weninger. 2018. Open-world knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*.

Harkanwar Singh, Soumen Chakrabarti, PRACHI JAIN, Sharod Roy Choudhury, and Mausam. 2021. Multilingual knowledge graph completion with joint relation and entity alignment. In *3rd Conference on Automated Knowledge Base Construction*.

Zequn Sun, Chengming Wang, Wei Hu, Muhao Chen, Jian Dai, Wei Zhang, and Yuzhong Qu. 2020. Knowledge graph alignment network with gated multi-hop neighborhood aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 222–229.

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*.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022a. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4281–4294.

Xiang Wang, Tinglin Huang, Dingxian Wang, Yancheng Yuan, Zhenguang Liu, Xiangnan He, and Tat-Seng Chua. 2021a. Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the Web Conference 2021*, pages 878–887.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021b. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Xiting Wang, Kunpeng Liu, Dongjie Wang, Le Wu, Yanjie Fu, and Xing Xie. 2022b. Multi-level recommendation reasoning over knowledge graphs with reinforcement learning. In *Proceedings of the ACM Web Conference 2022*, pages 2098–2108.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pages 1112–1119.

Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, and Bo Long. 2021. Graph neural networks for natural language processing: A survey. *arXiv preprint arXiv:2106.06090*.

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*.

Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*.

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

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639.

Jiajun Zhang and Chengqing Zong. 2020. Neural machine translation: Challenges, progress and future. *Science China Technological Sciences*, 63(10):2028–2050.

Qingheng Zhang, Zequn Sun, Wei Hu, Muhao Chen, Lingbing Guo, and Yuzhong Qu. 2019. Multi-view knowledge graph embedding for entity alignment. *arXiv preprint arXiv:1906.02390*.

Yongqi Zhang and Quanming Yao. 2022. Knowledge graph reasoning with relational digraph. In *Proceedings of the ACM Web Conference 2022*, pages 912–924.

Yao Zhu, Hongzhi Liu, Zhonghai Wu, and Yingpeng Du. 2021a. Relation-aware neighborhood matching model for entity alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4749–4756.

Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal Xhonneux, and Jian Tang. 2021b. Neural bellman-ford networks: A general graph neural network framework for link prediction. *Advances in Neural Information Processing Systems*, 34:29476–29490.

Chengqing Zong, Rui Xia, and Jiajun Zhang. 2021. *Text Data Mining*, volume 711. Springer.