

A Contextual Alignment Enhanced Cross Graph Attention Network for Cross-lingual Entity Alignment

Zhiwen Xie¹, Runjie Zhu², Kunsong Zhao¹, Jin Liu^{1*},
Guangyou Zhou³, Jimmy Xiangji Huang⁴

¹School of Computer Science, Wuhan University

²Lassonde School of Engineering, York University

²AI Singapore, Singapore

³School of Computer Science, Central China Normal University

⁴School of Information Technology, York University

{xiezhwen, kszhao, jinliu}@whu.edu.cn, sherryzh@cse.yorku.ca

sherryzhu@aisingapore.org

gyzhou@mail.ccnu.edu.cn, jhuang@yorku.ca

Abstract

Cross-lingual entity alignment, which aims to match equivalent entities in KGs with different languages, has attracted considerable focus in recent years. Recently, many graph neural network (GNN) based methods are proposed for entity alignment and obtain promising results. However, existing GNN-based methods consider the two KGs independently and learn embeddings for different KGs separately, which ignore the useful pre-aligned links between two KGs. In this paper, we propose a novel Contextual Alignment Enhanced Cross Graph Attention Network (CAECGAT) for the task of cross-lingual entity alignment, which is able to jointly learn the embeddings in different KGs by propagating cross-KG information through pre-aligned seed alignments. We conduct extensive experiments on three benchmark cross-lingual entity alignment datasets. The experimental results demonstrate that our proposed method obtains remarkable performance gains compared to state-of-the-art methods.

1 Introduction

Knowledge graphs (KGs) have recently demonstrated their potential in many natural language processing (NLP) tasks, such as language modelling (Zhang et al., 2019) and question answering (Wu et al., 2019a). With the rapid growth of multilingual KGs, such as DBpedia (Lehmann et al., 2015) and YAGO (Rebele et al., 2016; Suchanek et al., 2008), cross-lingual entity alignment has attracted considerable focus due to the lack of cross-lingual links. The task of cross-lingual entity alignment aims to automatically detect equivalent entities from different monolingual KGs to bridge the language gap.

Most recently, many graph neural networks (GNNs) based methods are proposed for entity alignment (Xu et al., 2019; Wang et al., 2018; Mao et al., 2020; Cao et al., 2019; Yang et al., 2019; Sun et al., 2019). Since GNNs are powerful to model graph-structured data by aggregating neighborhood information, the GNN-based methods have shown promising performance. However, existing GNN-based methods model different KGs separately, which ignore the useful pre-aligned links between two KGs. These GNN-based methods only use the pre-aligned seed alignments to optimize the objective function during training and fail to take full use of seed alignments, which provide useful contextual alignment information for entity alignment task. Intuitively, if two entities in different KGs share many pre-aligned neighbors, they are most likely to be equivalent entities. Taking the example shown in Figure 1(a), although the entity “哥威迅语(*Gothic language*)” in Chinese from G_1 has the same translated surface form with the entity “*Gothic language*” in English from G_2 , the correct alignment for “哥威迅语(*Gothic language*)” is the entity “*Gwich'in language*” in English. By considering the contextual alignments in different KGs, such as “美国(*United States*)” and “*United States*”, “加拿大(*Canada*)” and “*Canada*”, “纳—德内语系(*Na de nene*)” and “*Na-Dene languages*”, “德内语支(*German language branch*)” and

* Corresponding author.

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>.

“Athabaskan languages”, we can get more evidences that the embeddings of entity “哥威迅语(Gothic language)” and entity “Gwich’in language” are equivalent with each other. Thus, the mismatch issue can be easily corrected. The contextual alignments information is not explicitly considered in conventional GNN-based methods, which leads to sub-optimal results.

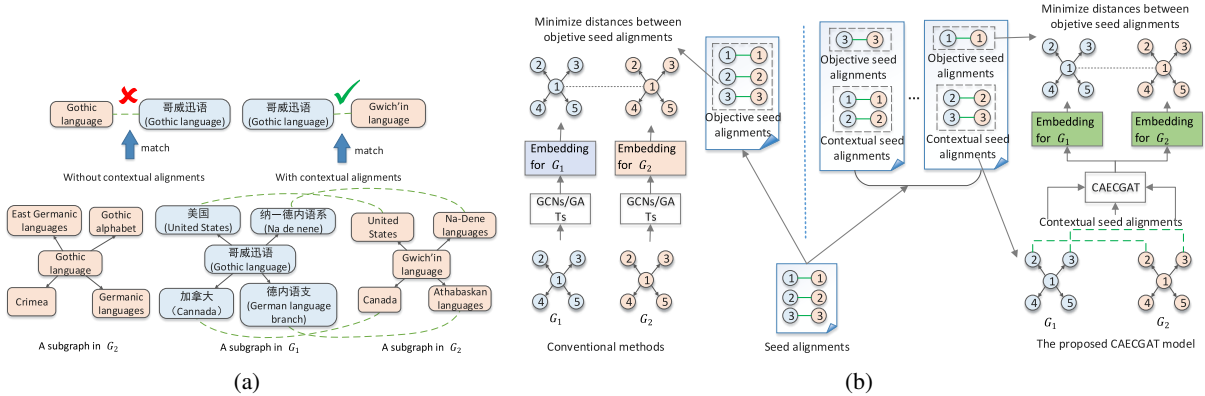


Figure 1: (a) An example of cross-lingual entity alignment in DBP15K_{ZH-EN} dataset. The translated surface strings for Chinese entities are provided in original DBP15K_{ZH-EN} dataset (Sun et al., 2017). The dashed lines represents the pre-aligned neighbors. (b) The comparison between conventional GNN-based methods and the proposed CAECGAT model. Our aim is to predict whether the $entity_1$ from G_1 and $entity_1$ from G_2 are two equivalent entities.

Therefore, it is beneficial to take full advantage of contextual alignments for entity alignment task. To this end, we propose a Contextual Alignment Enhanced Cross Graph Attention Network (CAECGAT) for the task of cross-lingual entity alignment. CAECGAT is able to jointly learn the embeddings in different KGs by propagating cross-KG information through pre-aligned seed alignments. Specifically, we develop a new cross graph attention (CGAT) layer to learn cross-KG information. The CGAT layer includes a cross-KG aggregation layer and an attention-based cross-KG propagation layer. We first use the cross-KG aggregation layer to transfer the entity information across two KGs through the pre-aligned seed entities. Thus, the embeddings of different KGs are mapped into the same semantic space by sharing cross-KG information. Then, the attention-based cross-KG propagation layer is applied to focus on the neighbors with important cross-KG information, so that the semantic gap between different languages can be alleviated by propagating the cross-KG information.

To train the model, we split the seed alignments into contextual seed alignments and objective seed alignments during training. The contextual seed alignments are fed into the model, which provide pre-aligned information and allow the entity information to propagate across different KGs. While the objective seed alignments are used to optimize parameters in the model. Figure 1(b) shows the comparison between conventional GNN-based methods and the proposed CAECGAT model. It is noted that there is a significant difference between the conventional GNN-based methods and our proposed CAECGAT model. The conventional GNN-based methods regard all pre-aligned entity pairs as objective seed alignments for training and ignore the contextual aligned information to propagate across different KGs, resulting in the sub-optimal results. On the contrary, our proposed CAECGAT model chooses one batch seed alignment as objective seed alignments (e.g., $entity_1$ in Figure 1(b)) and the rest as contextual seed alignments (e.g., $entity_2$ and $entity_3$ in Figure 1(b)) with an iterative manner. In this case, we can better learn the cross-KG embeddings by propagating contextual seed alignments across different KGs.

The main contributions of this study are summarised as follows:

- We propose a novel Contextual Alignment Enhanced Cross Graph Attention Network (CAECGAT) for cross-lingual entities alignment, which can jointly learn cross-KG embeddings by propagating information across different KGs.
- We propose a new training strategy by dividing the seed alignments into contextual and objective

Table 1: Notation List.

Notation	Description
G_1, G_2	The knowledge graphs in different languages.
E_1, E_2	The entities in G_1 and G_2 .
R_1, R_2	The relations in G_1 and G_2 .
T_1, T_2	The triples in G_1 and G_2 .
A	The seed alignments.
A_{ctx}, A_{obj}	The contextual seed alignments and objective seed alignments.
$\mathbf{E}_1, \mathbf{E}_2$	The initial entity embeddings for G_1 and G_2 .
$\bar{\mathbf{E}}_1, \bar{\mathbf{E}}_2$	The final entity embeddings for G_1 and G_2 .
$\mathbf{e}_1, \mathbf{e}_2$	The initial entity vectors for $e_1 \in G_1$ and $e_2 \in G_2$.
$\bar{\mathbf{e}}_1, \bar{\mathbf{e}}_2$	The final entity vectors for $e_1 \in G_1$ and $e_2 \in G_2$.
$gate$	The gate mechanism to combine cross-KG embeddings.
$crossAggr$	The function to aggregate cross-KG information.
$crossAtt$	The function to propagate cross-KG information in different KGs.
$\mathcal{L}(\phi; A_{obj})$	The loss function of the proposed CAECGAT model.
ϕ	The trainable parameters in the model.
$D(\bar{\mathbf{e}}_1, \bar{\mathbf{e}}_2)$	The L_1 distance between entity vectors $\bar{\mathbf{e}}_1$ and $\bar{\mathbf{e}}_2$.

seed alignments with an iterative manner, enabling our model to capture the cross-KG information.

- We conduct extensive experiments on three benchmark datasets. The experimental results demonstrate that our proposed method obtains remarkable performance gains compared to state-of-the-art methods.

2 Approach

2.1 Overview

Figure 2 illustrates the structure of the proposed CAECGAT model, which consists of multiple CGAT layers. A CGAT layer contains a cross-KG aggregation layer and an attention-based cross-KG propagation layer. As illustrated in Figure 2, given two different KGs G_1 and G_2 , and a collection of pre-aligned entity pairs, we first use the cross-KG aggregation layer to transfer the entity information across the two KGs through the pre-aligned entity pairs. This operation is useful to map the entity embeddings in different KGs into the same semantic space by sharing cross-KG entity embeddings. Then, we applied the attention-based cross-KG propagation layer to gather neighbors with important cross-KG information. Thus, the cross-KG information can be propagated in two KGs. By stacking multiple CGAT layers, the model is able to learn multi-hop cross-KG information. The main notations of this paper are summarized in Table 1.

2.2 Cross-KG Aggregation

The cross-KG aggregation is used to transfer graph information across different KGs through seed alignments. By taking full advantage of pre-aligned entity pairs, we can make full use of pre-aligned neighbors as contextual information to predict new alignments. This kind of cross-KG alignment information is very useful for entity alignment task but ignored in previous methods. In this study, we take full advantage of the pre-aligned entity pairs to aggregate cross-KG information and mitigate the semantic gap between different KGs.

Formally, give two different KGs $G_1 = (E_1, R_1, T_1)$ and $G_2 = (E_2, R_2, T_2)$, and a set of seed alignments $A = \{(e_1, e_2) | e_1 \in E_1, e_2 \in E_2\}$. In this paper, we represent the entities in the two KGs as k -dimensional embedding matrices \mathbf{E}_1 and \mathbf{E}_2 . During training, we split all the seed alignments into a set of contextual seed alignments A_{ctx} and a set of objective seed alignments A_{obj} . The contextual seed alignments in A_{ctx} are used as bridge to transfer information across different KGs. Thus, the cross-KG information provided by A_{ctx} can be used as contextual alignment information to predict the matching scores for the entity pairs in objective seed alignments in A_{obj} .

Concretely, for each pre-aligned entity pair $(e_1, e_2) \in A_{ctx}$, we use a gate mechanism to update their

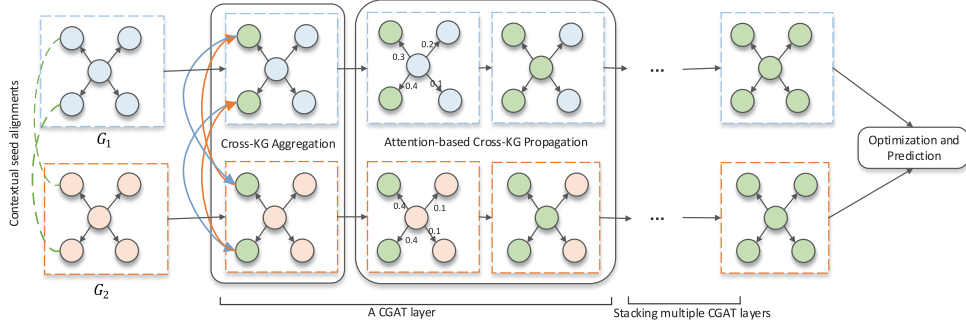


Figure 2: The architecture of the proposed CAECGAT model. The cross-KG aggregation layer is used to transfer cross-KG information across different KGs through contextual seed alignments. And the attention-based cross-KG propagation layer is used to propagate the cross-KG information in these two KGs. By stacking multiple CGAT layers, the cross-KG information can be propagated to multi-hop neighbors.

embeddings by combing the embeddings of themselves and their counterpart entities from the other KGs:

$$\begin{aligned} \mathbf{h}_{e_1}^l &= gate(\mathbf{e}_1^l, \mathbf{e}_2^l) = g_1^l \cdot \mathbf{e}_1^l + (1 - g_1^l) \cdot \mathbf{e}_2^l \\ \mathbf{h}_{e_2}^l &= gate(\mathbf{e}_2^l, \mathbf{e}_1^l) = g_2^l \cdot \mathbf{e}_2^l + (1 - g_2^l) \cdot \mathbf{e}_1^l \end{aligned} \quad (1)$$

where \mathbf{e}_1^l and \mathbf{e}_2^l are the vectors of entity e_1 and e_2 in the l -th layer, g_1^l and g_2^l are the gate mechanism to control how much information flow across KGs, which is computed as:

$$\begin{aligned} g_1^l &= \sigma(\mathbf{W}_1^l[\mathbf{e}_1 || \mathbf{e}_2] + \mathbf{b}_1^l) \\ g_2^l &= \sigma(\mathbf{W}_2^l[\mathbf{e}_1 || \mathbf{e}_2] + \mathbf{b}_2^l) \end{aligned} \quad (2)$$

where σ is the sigmoid activation function which can constrain the output values within the range of 0 to 1, $\mathbf{W}_{\{1,2\}}^l$ and $\mathbf{b}_{\{1,2\}}^l$ are the parameters, $||$ denotes the concatenation operation.

For the entities without pre-aligned counterpart entities in the other KGs, their embeddings do not change in the cross-KG aggregation layer. By applying this cross-KG aggregation method, we can obtain cross-KG embeddings \mathbf{H}_1^l and \mathbf{H}_2^l with sharing entity representations for equivalent entities. Formally, the cross-KG embeddings is computed as:

$$\begin{aligned} \mathbf{H}_1^l &= crossAggr(\mathbf{E}_1^l, \mathbf{E}_2^l, A_{ctx}) \\ \mathbf{H}_2^l &= crossAggr(\mathbf{E}_2^l, \mathbf{E}_1^l, A_{ctx}) \end{aligned} \quad (3)$$

where $crossAggr$ denotes the cross-KG aggregation layer, which can be formulated as

$$crossAggr(\mathbf{e}_1^l, \mathbf{E}_2^l, A_{ctx}) = \begin{cases} gate(\mathbf{e}_1^l, \mathbf{e}_2^l), & \text{if } (e_1, e_2) \in A_{ctx} \\ \mathbf{e}_1^l, & \text{otherwise} \end{cases} \quad (4)$$

and $crossAggr(\mathbf{e}_2^l, \mathbf{E}_1^l, A_{ctx})$ is computed in a similar form.

2.3 Attention-based Cross-KG Propagation

Recently, GNNs have been successfully applied for entity alignment. These preliminary studies use GNNs to propagate neighborhood information in a monolingual KG and fail to take full use of seed alignments to alleviate the semantic gap between different KGs. In this study, we attempt to jointly learn the entity embeddings and propagate cross-KG information in different KGs.

We have encoded cross-KG information in the entity embeddings by applying the cross-KG aggregation layer, we can further propagate the cross-KG information using GNNs rather than only mono-lingual KG information. Specifically, we use an attention-based cross-KG propagation layer, which is inspired

by graph attention network (GAT) (Velickovic et al., 2017), to aggregate neighborhood features. Our attention-based cross-KG propagation layer aims to select the most important common neighbors shared by different KGs to enhance the entity embeddings. Thus, the semantic gap can be alleviated by focusing on the common neighbors with cross-KG information and reducing noising neighbors.

Given the output entity embeddings of cross-KG aggregation layer, e.g., \mathbf{H}_1^l and \mathbf{H}_2^l for two KGs G_1 and G_2 , we use a graph attention mechanism to update the entity embeddings by gathering neighborhood information:

$$\begin{aligned}\mathbf{E}_1^{l+1} &= \text{crossAtt}(\mathbf{H}_1^l, G_1) \\ \mathbf{E}_2^{l+1} &= \text{crossAtt}(\mathbf{H}_2^l, G_2)\end{aligned}\quad (5)$$

where crossAtt is the attention-based cross-KG propagation function. For each entity $e_1 \in G_1$, the output feature of crossAtt function is generated by gathering the cross-KG neighborhood embeddings using a weighted sum function:

$$\begin{aligned}\mathbf{e}_1^{l+1} &= \text{crossAtt}(\text{crossAggr}(\mathbf{E}_1^l, \mathbf{E}_2^l, A_{ctx}), e_1) \\ &= \text{crossAtt}(\mathbf{H}_1^l, e_1) \\ &= \text{Relu}(\mathbf{h}_{e_1}^l + \sum_{e_k \in \mathcal{N}_{e_1}} \alpha_{1k} \mathbf{h}_{e_k}^l)\end{aligned}\quad (6)$$

where Relu (Glorot et al., 2011) is the activation function, \mathcal{N}_{e_1} is a set of neighbors of entity e_1 , and α_{1k} is the normalized attention weight for the neighborhood entity e_k , which is computed as:

$$\alpha_{1k} = \frac{\exp(s(\mathbf{h}_{e_1}^l, \mathbf{h}_{e_k}^l))}{\sum_{e'_k \in \mathcal{N}_{e_1}} \exp(s(\mathbf{h}_{e_1}^l, \mathbf{h}_{e'_k}^l))}\quad (7)$$

where s is the scoring function which is denoted as:

$$s(\mathbf{h}_{e_1}^l, \mathbf{h}_{e_k}^l) = \text{LeakyRelu}(\mathbf{v}^T [\mathbf{h}_{e_1}^l \parallel \mathbf{h}_{e_k}^l])\quad (8)$$

where \mathbf{v} is the attention vector, LeakyRelu (Maas et al., 2013) is the activation function which is widely used in graph attention mechanism. The crossAtt function for the entities in KG G_2 is computed in a similar form.

Formally, the update rule of the cross-KG aggregation and cross-KG propagation layers in a CGAT layer can be denoted as:

$$\begin{aligned}\mathbf{E}_1^{l+1} &= \text{crossAtt}(\text{crossAggr}(\mathbf{E}_1^l, \mathbf{E}_2^l, A_{ctx}), G_1) \\ \mathbf{E}_2^{l+1} &= \text{crossAtt}(\text{crossAggr}(\mathbf{E}_2^l, \mathbf{E}_1^l, A_{ctx}), G_2)\end{aligned}\quad (9)$$

By stacking L cross-KG aggregation and attention-based cross-KG propagation layers, we can obtain new entity embeddings $\bar{\mathbf{E}}_1 = \mathbf{E}_1^L$ and $\bar{\mathbf{E}}_2 = \mathbf{E}_2^L$, which contain both cross-KG and multi-hop neighborhood information.

2.4 Optimization and Prediction

The goal of entity alignment is to ensure that the learned embeddings of equivalent entities in different KGs are close to each other. Instead of just use the seed alignments to compute the loss function in conventional GNN-based methods, our model can take full use the seed alignments as contextual alignments during training and prediction.

During training, we use the objective seed alignments A_{obj} and margin-based ranking loss function to optimize the model, which is denoted as:

$$\begin{aligned}\mathcal{L}(\phi; A_{obj}) &= \sum_{(e_1, e_2) \in A_{obj}} [D(\bar{\mathbf{e}}_1, \bar{\mathbf{e}}_2) + \lambda - D(\bar{\mathbf{e}}_1, \bar{\mathbf{e}}_2^-)]_+ \\ &+ \sum_{(e_1, e_2) \in A_{obj}} [D(\bar{\mathbf{e}}_1, \bar{\mathbf{e}}_2) + \lambda - D(\bar{\mathbf{e}}_1^-, \bar{\mathbf{e}}_2)]_+\end{aligned}\quad (10)$$

Algorithm 1 CAECGAT Model

Input: Given KGs G_1 and G_2 , as well as the initial entity embeddings $\mathbf{E}_1, \mathbf{E}_2$ and pre-aligned seed alignments A .

Output: The final entity embeddings $\bar{\mathbf{E}}_1, \bar{\mathbf{E}}_2$ and parameters ϕ .

```
1: Let  $\mathbf{E}_1^0 = \mathbf{E}_1$  and  $\mathbf{E}_2^0 = \mathbf{E}_2$ .
2: repeat
3:   Select a batch seed alignments as objective seed alignments  $A_{obj}$  and the rest are used as context seed alignments  $A_{ctx}$ .
4:   for  $l = 1$  to  $L$  do
5:      $\mathbf{E}_1^{l+1} = crossAtt(crossAggr(\mathbf{E}_1^l, \mathbf{E}_2^l, A_{ctx}), G_1)$ ;
6:      $\mathbf{E}_2^{l+1} = crossAtt(crossAggr(\mathbf{E}_2^l, \mathbf{E}_1^l, A_{ctx}), G_2)$ ;
7:   end for
8:   Set  $\bar{\mathbf{E}}_1 = \mathbf{E}_1^L$  and  $\bar{\mathbf{E}}_2 = \mathbf{E}_2^L$ 
9:   Compute loss function  $\mathcal{L}(\phi; A_{obj})$ .
10:  Optimize parameters  $\phi$  in CAECGAT model.
11: until reach the maximum number of training epochs
12: Set  $A_{ctx} = A$ .
13: Use the learned CAECGAT model to predict the final entity embeddings  $\bar{\mathbf{E}}_1$  and  $\bar{\mathbf{E}}_2$ .
```

where $[x]_+ = \max\{0, x\}$, ϕ denotes the parameters in the model, (e_1, e_2^-) and (e_1^-, e_2) are the negative entity alignment pairs obtained by randomly replacing the positive entity e_1 or e_2 , $\bar{e}_1 \in \bar{\mathbf{E}}_1$ and $\bar{e}_2 \in \bar{\mathbf{E}}_2$ are the vectors of e_1 and e_2 , $D(\bar{e}_1, \bar{e}_2) = \|\bar{e}_1 - \bar{e}_2\|_1$ denotes the L_1 distance function, $\lambda > 0$ is the margin hyper-parameter.

To better understand the proposed model, we describe the details of CAECGAT model in Algorithm 1. Note that the contextual seed alignments A_{ctx} and objective seed alignments A_{obj} are not fixed during training. Actually, these two seed alignments sets will be dynamically changed to ensure all the seed alignments can be used to optimize the loss function. Specifically, all the seed alignments in A are split into several batch seed alignments. For each optimization step, we select one batch seed alignments as objective alignments A_{obj} , and all the rest seed alignments are used as contextual seed alignments A_{ctx} . We will repeat this operation until reach the maximum of training epochs, which is described from line 2 to line 11 in Algorithm 1.

During prediction, given a test entity e_1 (or e_2) in from KG G_1 (or G_2), we rank all entities in another KG G_2 (or G_1) according to the L_1 distance computed by using the entity vectors $\bar{\mathbf{E}}_1$ and $\bar{\mathbf{E}}_2$ learned by CAECGAT model. Note that, during prediction, we can use all the pre-aligned seed alignments in A as contextual seed alignments, namely $A_{ctx} = A$, which is described from line 12 to line 13 in Algorithm 1. Therefore, our proposed CAECGAT model provides a new way to take full advantage of the seed alignments in both training and prediction procedures.

3 Experiments

In this section, we conduct extensive experiments to evaluate our proposed method CAECGAT on cross-lingual entity alignment task. The detailed experiments on benchmark datasets and the results are described in the following subsections.

3.1 Datasets

To evaluate the ability of the proposed CAECGAT model, we conduct experiments on three widely used cross-lingual datasets from DBP15K (Sun et al., 2017). These datasets are extracted from multilingual DBpedia (Lehmann et al., 2015), including four KGs with different languages (English, Chinese, Japanese and French). Three cross-lingual datasets are built upon these KGs, including DBP15K_{ZH-EN} (Chinese-English), DBP15K_{JA-EN} (Japanese-English) and DBP15K_{FR-EN} (French-English). Each dataset contains 15,000 inter-lingual links connecting equivalent entity pairs in KGs with different languages. The statistics of the datasets are summarized in Table 2.

3.2 Implementation Details

In our experiments, as done in (Xu et al., 2019; Wu et al., 2019c), the entity embeddings are initialized by using the sum of word vectors of the entities' surface form. We choose the above initialization due to its good performance shown in various previous work (Xu et al., 2019; Wu et al., 2019c).

Table 2: Statistics of the datasets.

Dataset		Entities	Relations	Rel.Triples
DBP15K _{ZH-EN}	Chinese	66,469	2,830	153,929
	English	98,125	2,317	237,674
DBP15K _{JA-EN}	Japanese	65,744	2,043	164,373
	English	95,680	2,096	233,319
DBP15K _{FR-EN}	French	66,858	1,379	192,191
	English	105,889	2,209	278,590

We use the same data split as in various previous work (Sun et al., 2017; Yang et al., 2019), namely 30% for training and 70% for testing. We further sample 10% of the training data as the development set for parameter selection, and use the remaining 90% for training. We apply Adam (Kingma and Ba, 2015) to optimize the parameters in the model, and we train the model up to 5,000 epochs. The hyper-parameters are selected by using grid search according to the Hits@1 on development set. We select the number of negative entity pairs from $\{10, 20, 30, 50\}$, the margin parameter λ in Equation 10 from $\{1, 3, 5\}$, the number of CGAT layers from $\{1, 2, 3\}$, the dropout rate from $\{0.1, 0.2, 0.3\}$, and the batch size from 500, 1000, 1500, 2000, 3000, the learning rate from $\{0.002, 0.001, 0.0005\}$. Finally, we randomly sample 30 negative alignment entity pairs for each positive pair, and the margin parameter in Equation 10 is set to $\lambda = 3$. We stack two CGAT layers to propagate multi-hop cross-KG information. The learning rate is set to 0.002. The dropout rate is set to 0.2. The batch size is set to 2,000. We evaluate the performance of the model using the standard metrics Hits@1 (H@1), Hits@10 (H@10) and MRR (Mean Reciprocal Rank). All experiments reported in this study are conducted on NVIDIA GTX 1080Ti GPUs and the codes are implemented using TensorFlow.

3.3 Comparison Models

To investigate the power of the proposed CAECGAT, we compare the proposed method with various baselines. These baseline methods can be roughly classified into two groups: embedding-based models and GNN-based models.

- **Embedding-based Models.** The embedding-based models compared in our study include the following baselines: JE (Hao et al., 2016), JAPE (Sun et al., 2017), BootEA (Sun et al., 2018), MTransE (Chen et al., 2017), IPTransE (Zhu et al., 2017).
- **GNN-based Models.** We follow the line of GNN-based methods and compare our proposed CAECGAT with the following baselines: GCN-Align (Wang et al., 2018), KECG (Li et al., 2019), MuGNN (Cao et al., 2019), AliNet (Sun et al., 2019), HGCN-JE (Wu et al., 2019c), HMAN (Yang et al., 2019), MRAEA (Mao et al., 2020), NAEA (Zhu et al., 2019), RDGCN (Wu et al., 2019b) and GM-EHD-JEA (Xu et al., 2020).

We also note that some research studies, such as KDCoE (Chen et al., 2018), NTAM (Li et al., 2018), SEA (Pei et al., 2019) and OTEA (S. et al., 2019), and attribute enhanced methods (Trisedya et al., 2019), focus on the entity alignment on other datasets instead of DBP15K. We will leave these comparison in the future work due to the limited space.

3.4 Main Results

Table 3 presents the cross-lingual entity alignment performance of our proposed model, as well as the comparisons with various baselines. As shown in Table 3, our model CAECGAT achieves the best performance on DBP15K_{JA-EN} and DBP15K_{FR-EN} across all the metrics, and achieves the best result of H@10, the second best result of H@1 and MRR on DBP15K_{ZH-EN}. These results demonstrate the effectiveness of our proposed CAECGAT model. Compared to the most similar GNN-based models, such as GCN-Align (Wang et al., 2018), the proposed CAECGAT model is able to take the seed alignments as contextual information and propagate the cross-KG information in different KGs, making our CAECGAT model achieves the remarkable performance gains. For example, CAECGAT obtains a 34.30% improvements over GCN-Align by H@1 on DBP15K_{ZH-EN}. These results shows that the contextual

Table 3: Experimental results on entity alignment. The results of the baselines are directly taken from the original papers. The best results are in **bold**, and the second best results are in underline.

Methods	DBP15K _{ZH-EN}			DBP15K _{JA-EN}			DBP15K _{FR-EN}		
	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR
JE (Hao et al., 2016)	21.27	42.77	-	18.92	39.97	-	15.38	38.84	-
MTransE (Chen et al., 2017)	30.83	61.41	0.364	27.86	57.45	0.349	24.41	55.55	0.335
JAPE (Sun et al., 2017)	41.18	74.46	0.490	36.25	68.5	0.476	32.39	66.68	0.430
IPTransE (Zhu et al., 2017)	40.60	73.50	0.516	36.70	69.30	0.474	33.30	68.50	0.451
BootEA (Sun et al., 2018)	62.94	84.75	0.703	62.23	85.39	0.701	65.30	87.44	0.731
GCN-Align (Wang et al., 2018)	41.25	74.38	0.549	39.91	74.46	0.546	37.29	74.49	0.532
GM (Xu et al., 2019)	67.93	78.48	-	73.97	87.15	-	89.38	95.24	-
KECG (Li et al., 2019)	47.77	83.5	0.598	48.97	84.4	0.61	48.64	85.06	0.61
MuGNN (Cao et al., 2019)	49.40	84.40	0.611	50.10	85.70	0.621	49.50	87.00	0.621
AliNet (Sun et al., 2019)	53.90	82.60	0.628	54.90	83.10	0.654	55.20	85.20	0.657
RDGCN (Wu et al., 2019b)	70.75	84.55	-	76.74	89.54	-	88.64	95.72	-
HGCN-JE (Wu et al., 2019c)	72.03	85.7	-	76.62	89.73	-	89.16	<u>96.11</u>	-
HMAN (Yang et al., 2019)	56.20	85.10	-	56.70	86.90	-	54.00	87.10	-
NAEA (Zhu et al., 2019)	65.01	86.73	0.720	64.14	87.27	0.718	67.32	89.43	0.752
MRAEA (Mao et al., 2020)	75.70	<u>92.98</u>	0.827	75.78	<u>93.38</u>	<u>0.826</u>	78.04	94.81	<u>0.849</u>
GM-EHD-JEA (Xu et al., 2020)	73.58	-	-	<u>79.15</u>	-	-	<u>92.43</u>	-	-
CAECGAT (ours)	<u>75.55</u>	93.38	0.818	83.58	95.59	0.881	94.68	99.18	0.965

Table 4: Ablation Study.

Methods	DBP15K _{ZH-EN}			DBP15K _{JA-EN}			DBP15K _{FR-EN}		
	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR
BASELINE	56.09	71.55	0.614	67.1	78.25	0.712	85.48	91.92	0.878
GAT	63.86	91.92	0.709	72.35	94.64	0.815	86.24	98.70	0.895
CrossGCN	65.45	92.85	0.757	72.07	94.9	0.809	93.79	98.98	0.958
CAECGAT _{L=1}	72.25	92.03	0.796	81.9	93.68	0.862	93.55	98.5	0.954
CAECGAT _{L=2}	75.55	93.38	0.818	83.58	95.59	0.881	94.68	99.18	0.965
CAECGAT _{L=3}	74.95	93.38	0.818	82.46	96.21	0.877	93.78	98.77	0.956

seed alignments are important for entity alignment task and our model is effective to bridge the language gap between different KGs by taking these contextual seed alignments into account.

3.5 Ablation Study

In order to better understand how each component affects the performance of the proposed CAECGAT model, we conduct an ablation study shown in Table 4. There are three variants of our proposed CAECGAT model, “BASELINE” is a simple model which identifies the equivalent counterparts from different KGs with the embeddings learned using the sum of word vectors within the surface names of entities. “GAT” represents the conventional graph attention network, which is reduced the cross-KG aggregation layer from the proposed CAECGAT model. “CrossGCN” is the model obtained by replacing the attention-based propagation layer with multiple GCN layers. The subscripts $L = 1, 2, 3$ denote models with different numbers of CGAT layers.

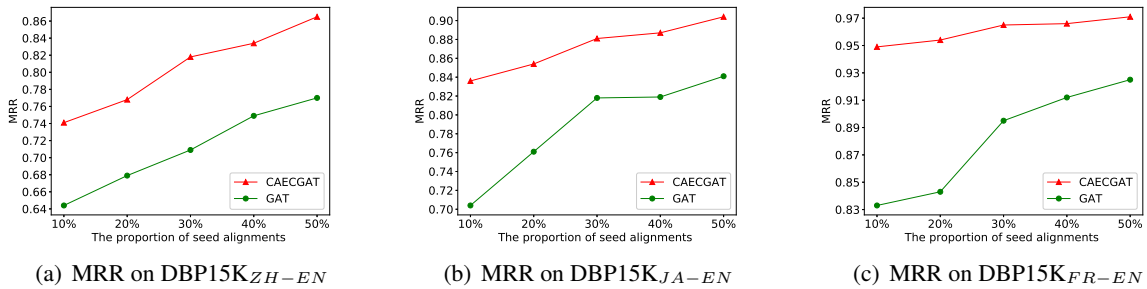


Figure 3: The MRR results on DBP15K using different proportions of seed alignments.

From Table 4, we can observe that the BASELINE model obtains good results by considering surface

names, which has been proved in previous works (Xu et al., 2019; Xu et al., 2020). By adding GNN layers, the performances obtain further improvements (e.g., GAT vs. BASELINE). And our proposed CAECGAT_{L=2} obtains large performance gains over the GAT model for all datasets, which indicates the cross-KG aggregation plays a very important role in our model. Compared with CrossGCN, our CAECGAT_{L=2} also performs much better. This is because the attention mechanism in cross-KG propagation layer is able to select the neighbors with important cross-KG information and mitigate the semantic gap between different KGs by propagating cross-KG information. These experimental results clearly demonstrate that all the components in the CAECGAT model can contribute a lot to the performance.

Compared the CAECGAT models with different layers, we can see that the CAECGAT_{L=2} with two CGAT layers and CAECGAT_{L=3} with three CGAT layers achieve better performance than CAECGAT_{L=1}. This is because CAECGAT_{L=1} with one CGAT layer can only capture neighborhood information from one-hop neighbors, while CAECGAT_{L=2} and CAECGAT_{L=3} are able to propagate multi-hop neighbors by stacking multiple CGAT layers. However, the performance of CAECGAT_{L=3} can not be further improved compared with CAECGAT_{L=2} and stacking more layers will increase the parameters in the model. Therefore, we set the number of CGAT layers to 2.

3.6 Performance vs. Different size of seed alignments

To investigate how the proportions of pre-aligned seed alignments affects the performance of our CAECGAT model, we further conduct experiments to evaluate the performance when the proportions of training sets vary from 10% to 50% with a step of 10%, and all the rest entity alignments are used for testing (e.g., from 10% for training and 90% for testing to 50% for training and 50% for testing). As depicted in Figure 3, we compare our proposed CAECGAT with the strong baseline GAT model which removes the cross-KG aggregation layer in our CAECGAT model. We can see that the performances for all datasets gradually improve when the proportions of training sets increase. Compared the proposed CAECGAT with the baseline GAT model, we can see that our CAECGAT consistently outperforms GAT model by large performance gains in all proportions, which reconfirms that our CAECGAT model is powerful to capture cross-KG features and reduces the semantic gap between different KGs.

4 Conclusions and Future Work

In this paper, we propose a novel CAECGAT model for cross-lingual KG entity alignment task, which take full use of seed alignments to alleviate the semantic gap between different KGs. We first use a cross-KG aggregation layer to transfer entity information across different KGs, which enables the embeddings of different KGs to share cross-KG information. Then, an attention-based cross-KG propagation layer is applied to gather neighbors with important cross-KG information. By staking multiple CGAT layers, the cross-KG information can be propagated in the two KGs. Experimental results on several benchmark datasets demonstrate the effectiveness of the proposed CAECGAT model.

In the future, we will consider more entity information, such as entity properties and descriptions. Besides, we may investigate the proposed model to other datasets (e.g., DBP100K (Sun et al., 2017) and WK31 (Chen et al., 2017)).

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grants 61972290 and 61972173, and also supported by the National Key R&D Program of China under Grant 2018YFC1604000. This research is also supported in part by the research grant from Natural Sciences and Engineering Research Council (NSERC) of Canada and York Research Chairs (YRC) program. This research is also supported by the National Research Foundation, Singapore under its AI Singapore programme. We thank anonymous reviewers for their thorough review comments on this paper.

References

Yixin Cao, Zhiyuan Liu, Chengjiang Li, Zhiyuan Liu, Juanzi Li, and Tat-Seng Chua. 2019. Multi-channel graph neural network for entity alignment. In *Proceedings of the ACL*.

- Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. 2017. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *Proceedings of IJCAI*, pages 1511–1517.
- Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena, and Carlo Zaniolo. 2018. Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. In *Proceedings of the IJCAI*.
- Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the AISTATS*.
- Yanchao Hao, Yuanzhe Zhang, Shizhu He, Kang Liu, and Jun Zhao. 2016. A joint embedding method for entity alignment of knowledge bases. In *Knowledge Graph and Semantic Computing: Semantic, Knowledge, and Linked Big Data - First China Conference, CCKS 2016, Beijing, China, September 19-22, 2016, Revised Selected Papers*, volume 650 of *Communications in Computer and Information Science*, pages 3–14. Springer.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the ICLR*.
- Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. Dbpedia – a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*.
- Shengnan Li, Xin Li, Rui Ye, Mingzhong Wang, Haiping Su, and Yingzi Ou. 2018. Non-translational alignment for multi-relational networks. In *Proceedings of the IJCAI*.
- Chengjiang Li, Yixin Cao, Lei Hou, Jiaxin Shi, Juanzi Li, and Tat-Seng Chua. 2019. Semi-supervised entity alignment via joint knowledge embedding model and cross-graph model. In *Proceedings of EMNLP-IJCNLP*, pages 2723–2732.
- Maas, Andrew L, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*.
- Xin Mao, Wenting Wang, Huimin Xu, Man Lan, and Yuanbin Wu. 2020. MRAEA: an efficient and robust entity alignment approach for cross-lingual knowledge graph. In *Proceedings of WSDM*.
- Shichao Pei, Lu Yu, Robert Hoehndorf, and Xiangliang Zhang. 2019. SEA: semi-supervised entity alignment via knowledge graph embedding with awareness of degree difference. In *Proceedings of WWW*.
- Thomas Rebele, Fabian M. Suchanek, Johannes Hoffart, Joanna Biega, Erdal Kuzey, and Gerhard Weikum. 2016. YAGO: A multilingual knowledge base from wikipedia, wordnet, and geonames. In *Proceedings of ISWC*.
- Pei S., Yu L., and Zhang X. 2019. Pei19: improving cross-lingual entity alignment via optimal transport. In *Proceedings of IJCAI*.
- Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO: A large ontology from wikipedia and wordnet. *J. Web Semant.*
- Zequn Sun, Wei Hu, and Chengkai Li. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *Proceedings of ISWC*.
- Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. 2018. Bootstrapping entity alignment with knowledge graph embedding. In *Proceedings of IJCAI*.
- Zequn Sun, Chengming Wang, Wei Hu, Muhao Chen, Jian Dai, Wei Zhang, and Yuzhong Qu. 2019. Knowledge graph alignment network with gated multi-hop neighborhood aggregation. *CoRR*, abs/1911.08936.
- Bayu Distiawan Trisedya, Jianzhong Qi, and Rui Zhang. 2019. Entity alignment between knowledge graphs using attribute embeddings. In *Proceedings of the AAAI*, pages 297–304.
- Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph attention networks. *ArXiv*, abs/1710.10903.
- Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. 2018. Cross-lingual knowledge graph alignment via graph convolutional networks. In *Proceedings of the EMNLP*.
- Peng Wu, Shujian Huang, Rongxiang Weng, Zaixiang Zheng, Jianbing Zhang, Xiaohui Yan, and Jiajun Chen. 2019a. Learning representation mapping for relation detection in knowledge base question answering. In *Proceedings of the ACL*.

- Yuting Wu, Xiao Liu, Yansong Feng, Zheng Wang, Rui Yan, and Dongyan Zhao. 2019b. Relation-aware entity alignment for heterogeneous knowledge graphs. In *Proceedings of IJCAI*, pages 5278–5284.
- Yuting Wu, Xiao Liu, Yansong Feng, Zheng Wang, and Dongyan Zhao. 2019c. Jointly learning entity and relation representations for entity alignment. In *Proceedings of EMNLP-IJCNLP*, pages 240–249.
- Kun Xu, Liwei Wang, Mo Yu, Yansong Feng, Yan Song, Zhiguo Wang, and Dong Yu. 2019. Cross-lingual knowledge graph alignment via graph matching neural network. In *Proceedings of the ACL*.
- Kun Xu, Linfeng Song, Yansong Feng, Yan Song, and Dong Yu. 2020. Coordinated reasoning for cross-lingual knowledge graph alignment. *CoRR*, abs/2001.08728.
- Hsiu-Wei Yang, Yanyan Zou, Peng Shi, Wei Lu, Jimmy Lin, and Xu Sun. 2019. Aligning cross-lingual entities with multi-aspect information. In *Proceedings of the EMNLP*.
- Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In *Proceedings of the ACL*.
- Hao Zhu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. 2017. Iterative entity alignment via joint knowledge embeddings. In *Proceedings of IJCAI*, pages 4258–4264.
- Qiannan Zhu, Xiaofei Zhou, Jia Wu, Jianlong Tan, and Li Guo. 2019. Neighborhood-aware attentional representation for multilingual knowledge graphs. In *Proceedings of IJCAI*.