Improving Cross-lingual Entity Alignment via Optimal Transport

Shichao Pei, Lu Yu and **Xiangliang Zhang**
The Computer, Electrical and Mathematical Sciences and Engineering Division
King Abdullah University of Science and Technology (KAUST), Thuwal, 23955, SA
{shichao.pei, lu.yu, xiangliang.zhang}@kaust.edu.sa

Abstract

Cross-lingual entity alignment identifies entity pairs that share the same meanings but locate in different language knowledge graphs (KGs). The study in this paper is to address two limitations that widely exist in current solutions: 1) the alignment loss functions defined at the entity level serve well the purpose of aligning labeled entities but fail to match the whole picture of labeled and unlabeled entities in different KGs; 2) the translation from one domain to the other has been considered (e.g., X to Y by M1 or Y to X by M2). However, the important duality of alignment between different KGs (X to Y by M1 and Y to X by M2) is ignored. We propose a novel entity alignment framework (OTEA), which dually optimizes the entity-level loss and group-level loss via optimal transport theory. We also impose a regularizer on the dual translation matrices to mitigate the effect of noise during transformation. Extensive experimental results show that our model consistently outperforms the state-of-the-arts with significant improvements on alignment accuracy.

1 Introduction

Along with the fast development of knowledge graphs (KGs) in different languages, cross-lingual entity alignment has become increasingly important due to its substantial assistance to many NLP applications. The mission is to align two entities in different KGs if they share the same semantic meaning. Given a number of labeled entity pairs, the alignment problem can be addressed in supervised ways with entity vectors described by human-designed features [Mahdisoltani *et al.*, 2013; Nguyen *et al.*, 2011] or with entity embeddings learned from KG embedding models [Chen *et al.*, 2016; Sun *et al.*, 2017; Wang *et al.*, 2018], as shown in Figure 1(a). However, labeled entity pairs are often difficult to obtain, and thus much less than unlabeled entities. Following ideas in semi-supervised learning, representative works like [Zhu *et al.*, 2017; Chen *et al.*, 2018b; Sun *et al.*, 2018] employ self-training to iteratively retrieval potentially aligned entities from unlabeled samples, then feed them back to update the alignment models. In spite of their success, current solutions minimize the alignment loss function defined at the entity level (focusing on the distance of given or augmented aligned entities), and thus suffer from the following shortcomings:

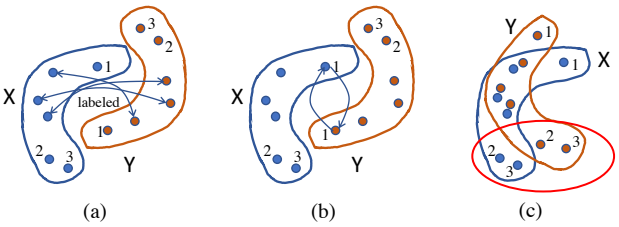


Figure 1: Toy examples of cross-lingual entity alignment in two knowledge graphs X and Y. (a) With a set of labeled entity pairs, the alignment is to match more entities in X with those in Y. (b) Duality in entity alignment, where the distribution of entities in X overlaps that of Y after taking a rotation on X or Y. (c) Labeled entities are well aligned, while others (like 1, 2, and 3) are not.

- **Limited gain due to the shortage of labeled entity pairs**: labeled entities usually take only a small portion of the entity set. Even though we can iteratively do data augmentation with learned alignment function, the mapping error will be accumulated along with the depth of augmentation. There will be no benefits to gain from data augmentation after certain iterations.
- **Ignorance of duality**: alignment models through entity embedding map entities of different KGs to the shared concept space. Hence, the distribution of entity embedding should be similar in cross-lingual KGs. From the toy example shown in Figure 1(b), we can see that learned representations of entities in X and Y appear with similar shapes. One overlaps the other by taking a rotation. Such similarity between two distributions has been only explored by learning a translation function from $X \xrightarrow{M^1} Y$ or $Y \xrightarrow{M^2} X$, without investigating the dual alignment ($X \underset{M^2}{\overset{M^1}{\rightleftarrows}} Y$) of entities from different KGs.
- **Failure on matching the whole distribution**: alignment loss functions defined at the *entity level* serve well the

purpose of aligning labeled entities but fail to match the whole picture of labeled and unlabeled entities in different KGs. In Figure 1(c), the learned mapping function works well on labeled entities, but unlabeled samples in the red circle will impose a large loss since no objective functions of entity alignment have incorporated the *group-level* loss between embeddings of different KGs.

The aforementioned problems motivate us to design a novel Optimal Transport-based Entity Alignment (OTEA) model that learns the translation matrix by *dually* minimizing both *entity-level* and *group-level* loss. The group-level loss describes the discrepancy between two distributions of different embeddings. However, unlike entity-level loss, the group-level loss is difficult to measure using a statistical distance between two probability distributions, because the marginal distributions of two embedding sets are not available. Recently, adversarial training has emerged as a powerful paradigm to address this issue. Generative Adversarial Network (GAN) [Goodfellow *et al.*, 2014] is the representative work for distribution matching. However, GAN still suffers from an unstably weak learning signal due to the problem of JS divergence and the gradient vanishing effect. Inspired by the progress of Optimal Transport (OT), we set group-level loss by the minimum cost of transporting mass in converting the distribution of embedding in X to the distribution of embedding in Y. By jointly minimizing the entity-level and group-level loss, the entity alignment model can improve its generalization ability, and thus the accuracy of entity alignment.

We also propose to impose $L_{2,1}$ norm of the translation matrix as a regularizer in the alignment loss function, in order to force the translation matrix to be orthogonal. Orthogonal translation matrix is desirable when transforming one isomorphic embedding to another, as theoretically proved in [Smith *et al.*, 2017]. However, it is difficult to enforce the orthogonality of the translation matrix. $L_{2,1}$ norm has been widely used in machine learning community [Nie *et al.*, 2010] and compressed sensing theory [Yin *et al.*, 2015]. It plays here a new role on regularizing the translation matrix to make orthogonal transformation between two KG embeddings.

Our contributions in this work are summarized as follows:

- We propose to solve entity alignment by dually minimizing both the entity-level loss and the group-level loss via optimal transport theory.
- We impose $L_{2,1}$ norm on the dual translation matrices, which can enforce the translation matrix to be close to orthogonal.
- We conduct extensive experiments on six real-world datasets and show the superior performance of our proposed model over the state-of-the-art methods, with significant improvement on entity alignment accuracy.

2 Related Work

2.1 KG Embedding

Knowledge graph (KG) embedding has been developed as a fundamental tool to analyze and model the structure and semantic information in KG. Researchers explored semantic matching models which employ the similarity-based scoring functions, like DistMult [Yang *et al.*, 2014]. Many translational distance based models have been developed recently. The most representative translational distance model is TransE [Bordes *et al.*, 2013]. Several approaches improved TransE by introducing relation-specific hyperplanes [Wang *et al.*, 2014], relation-specific spaces [Lin *et al.*, 2015], decomposing the projection matrix into a product of two vectors in TransD [Ji *et al.*, 2015].

2.2 Entity Alignment

Pioneering work proposed to address entity alignment by hand-crafted features [Mahdisoltani *et al.*, 2013]. Crowdsourcing [Vrandečić and Krötzsch, 2014] was also employed. These methods suffer from the requirement of heavy human efforts. Alignment by leveraging extra resources was studied in OWL properties [Hu *et al.*, 2011], and entity descriptions [Yang *et al.*, 2015]. However, it is hard to obtain the extra resources for all KGs and modeling different extra resources is a complex process.

Recently, finding alignment using KG embeddings becomes the most popular solution. MTransE [Chen *et al.*, 2016] is the earliest work which encodes entities and relations of languages in a separated embedding space. Encoding entities and relations of different KGs into a unified low-dimensional space jointly [Zhu *et al.*, 2017] is the other way to do alignment. BootEA [Sun *et al.*, 2018] mitigated the problem of lacking labeled data by bootstrapping strategy and achieved a significant performance improvement. Further, JAPE [Sun *et al.*, 2017], KDCoE [Chen *et al.*, 2018b], GCN-based approach [Wang *et al.*, 2018] jointly modeled the structure and attributes information of KGs. However, all these models suffer from the three limitations discussed in the Introduction section.

It is worth noting that entity alignment differs from entity co-reference resolution [Ng and Cardie, 2002] and link discovery [Nentwig *et al.*, 2017] problem, because entity alignment focuses on the alignment of the entity in KGs, rather than text and knowledge bases.

2.3 Optimal Transport

Optimal transport (OT) is the natural geometry for probability measures supported on a geometric space [Peyré and Cuturi, 2018]. [Cuturi, 2013] proposed sinkhorn distance to improve the computational problem of OT, and [Arjovsky *et al.*, 2017] views the learning of generative adversarial networks (GANs) as a transportation problem by introducing the Wasserstein distances. OT has been widely used in many tasks which include, but not limited to, image segmentation [Peyré *et al.*, 2012], word embedding [Xu *et al.*, 2018] and text generation [Chen *et al.*, 2018a]. Yet, to the best of our knowledge, we are the first to adapt the theory of optimal transport for cross-lingual entity alignment between KGs.

3 Methodology

A knowledge graph can be denoted as $G = (E, R, T)$, where E is the set of entities, R is the set of relations, and T is the set of triples, each of which is a triple (h, r, t), including the

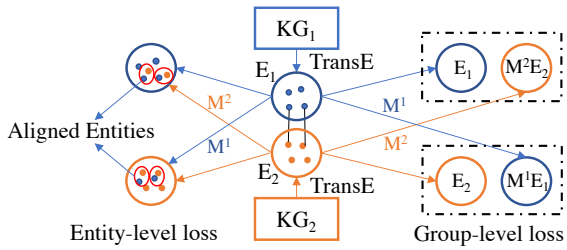


Figure 2: Framework of the proposed model for cross-lingual entity alignment. In detail, E_1 and E_2 are the embedding of KG_1 and KG_2. Entity-level loss measures the distance between aligned entity pairs, while group-level loss measures the distance between E_1 and M^2E_2, E_2 and M^1E_1. Both distances are modeled dually.

head entity h, the relation r and the tail entity t. By using KG embedding, each triple can be presented as $(\mathbf{h}, \mathbf{r}, \mathbf{t})$, in which boldfaced $\mathbf{h}$, $\mathbf{r}$, and $\mathbf{t}$ represent the embedding vectors of head h, relation r, and tail t, respectively. Cross-lingual knowledge graphs are a set of KGs with the language set LA denoted as $G_{LA} = \{G_1, G_2, ..., G_i\}$, where G_i denotes the KG with language $i \in LA$. In our work, we only consider the 1-to-1 entity alignment between two cross-lingual KGs.

Let $G_i = (E_i, R_i, T_i)$ and $G_j = (E_j, R_j, T_j)$ be two KGs in different language i and j. $AS = \{(e_i, e_j)|e_i \in E_i^L, e_j \in E_j^L\}$ is a set of labeled entity pairs that have same meaning, e.g., e_i in G_i shares same meaning with its counterpart e_j in G_j. Entity alignment is a task to find and align the remaining entities $\{e_i \in E_i^U\}$ and $\{e_j \in E_j^U\}$ which share same meaning, where $E_i^U = E_i \backslash E_i^L$ and $E_j^U = E_j \setminus E_j^L$.

Fig.2 shows the overall framework of our approach. The whole process involves minimizing the loss for knowledge graph embeddings ($\mathcal{L}_k$), the entity-level alignment loss ($\mathcal{L}_e$), the group-level alignment loss ($\mathcal{L}_g$) and a regularizer ($\mathcal{L}_r$).

3.1 Knowledge Graph Embedding

We build our model based on the basic TransE [Bordes *et al.*, 2013], like the previous works in [Chen *et al.*, 2016; Zhu *et al.*, 2017; Chen *et al.*, 2018b], as we focus on the alignment problem, rather than KG embedding that has a number of candidate solutions. When employing TransE on both knowledge graphs G_i and G_j, entities and relations are projected into the same low-dimensional space by encoding the triples (h, r, t), and making $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ when (h, r, t) holds. Specifically, the embeddings of relations can translate the embeddings of head entities to tail entities. The margin-based ranking objective function minimized by TransE over a knowledge graph G_i is defined as:

$$L_{G_i}(G_i; \theta_e^i) = \sum_{(h,r,t) \in T_i} L_t(h, r, t) \tag{1}$$

where θ_e^i presents the learned embedding from G_i, and $L_t(h, r, t)$ is the objective function defined for a triple (h, r, t):

$$L_t(h, r, t) = \sum_{(h', r, t') \in T'_{(h,r,t)}} [\gamma + E(h, r, t) - E(h', r, t')]_+ \tag{2}$$

where $[x]_+ = \max\{0, x\}$ denotes the positive part of x, γ is a margin hyper-parameter which is greater than 0, and $E(h, r, t)$ indicates the energy function:

$$E(h, r, t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_2 \tag{3}$$

and T' denotes the negative sample set for the triple (h, r, t):

$$T'_{(h,r,t)} = \{(h', r, t)|h' \in E\} \cup \{(h, r, t')|t' \in E\} \tag{4}$$

where (h', r, t) and (h, r, t') are the Bernoulli negative-sampled triples by replacing h or t in (h, r, t). The loss function of KG embedding for G_i and G_j together is:

$$\mathcal{L}_k = L_{G_i} + L_{G_j} \tag{5}$$

3.2 Entity-level Loss

We first define the entity-level loss. After obtaining entity embeddings of graph G_i and G_j from TransE, we make labeled entities aligned by dually minimizing

$$\mathcal{L}_e = \alpha_1 \sum_{(e_i, e_j) \in AS} \left\| \mathbf{M^1}\theta_{e_i}^i - \theta_{e_j}^j \right\|_2 + \left\| \mathbf{M^2}\theta_{e_j}^j - \theta_{e_i}^i \right\|_2 \tag{6}$$

where $\mathbf{M^1}$ and $\mathbf{M^2}$ are the $d \times d$ translation matrices, d is the dimension of entity embedding, and α_1 is a trade-off parameter. Note that our model dually learns the two translations of two embedding spaces in both directions. That is to say, $\mathbf{M^1}$ is learned to transfer the embeddings of G_i into the embedding space of G_j, and $\mathbf{M^2}$ is to transfer the embeddings of G_j into the embedding space of G_i.

3.3 Group-level Loss

Then, we define the group-level loss. Setting G_i as the source KG, and G_j is the target KG, as an example. After translating, we can measure how $\mathbf{M^1}\theta_e^i$ and θ_e^j are close. At the group-level, the embedding distribution of θ_e^j should be similar to the distribution of $\mathbf{M^1}\theta_e^i$. Let $\mathbf{p}$ be the distribution of $\mathbf{M^1}\theta_e^i$, and $\mathbf{q}$ be the distribution of θ_e^j. We define the group-level loss by measuring the difference between $\mathbf{p}$ and $\mathbf{q}$ with optimal transport distance, which is [Peyré and Cuturi, 2018]:

$$D_c(\mathbf{p}, \mathbf{q}) = \inf_{\gamma \in \prod(\mathbf{p}, \mathbf{q})} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma}[c(\mathbf{x}, \mathbf{y})] \tag{7}$$

where $\prod(\mathbf{p}, \mathbf{q})$ denotes the set of all joint distributions $\gamma(\mathbf{p}, \mathbf{q})$ with marginals $\mathbf{p}(\mathbf{x})$ and $\mathbf{q}(\mathbf{y})$; $c(\mathbf{x}, \mathbf{y}) : \mathbb{G} \times \mathbb{G} \to \mathbb{R}$ indicates the transportation cost function for moving $\mathbf{x}$ to $\mathbf{y}$.

Since $\mathbf{p}$ and $\mathbf{q}$ are discrete distributions, they can be represented by a sum of Dirac delta functions. That is, $\mathbf{p} = \sum_{i=1}^n u_i \delta_{x_i}$ and $\mathbf{q} = \sum_{i=1}^m v_i \delta_{y_i}$ with the Dirac function δ. The weight vectors $\mathbf{u} = \{u_i\}_{i=1}^n \in \Delta_n$ and $\mathbf{v} = \{v_i\}_{i=1}^m \in \Delta_m$ belong to the n and m-dimensional simplex. Then, the distance measure defined in Eq. (7) is equivalent to solving the following network-flow problem [Luise *et al.*, 2018]:

$$\mathcal{L}_{ot}(\mathbf{p}, \mathbf{q}) = \min_{\mathbf{T} \in \prod(\mathbf{p}, \mathbf{q})} \sum_{i=1}^n \sum_{j=1}^m \mathbf{T}_{ij} \cdot c(\mathbf{x}_i, \mathbf{y}_j) \tag{8}$$

where $\mathbf{T}$ denotes the transport matrix, $\prod(\mathbf{p}, \mathbf{q})$ is the transport polytope, defined as: $\prod(\mathbf{p}, \mathbf{q}) = \{\mathbf{T} \in \mathbb{R}_+^{n \times m} | \mathbf{T}\mathbf{1}_m = \mathbf{u}, \mathbf{T}^T\mathbf{1}_n = \mathbf{v}\}$, where $\mathbf{1}_n$ indicates an n-dimensional all-one vector. The transport matrix

$\mathbf{T}$ stores information of the transport plan, a non-zero T_{ij} denotes the amount of probability mass transported from $\mathbf{x}_i$ to $\mathbf{y}_j$. When $c(\mathbf{x}, \mathbf{y})$ is a metric on $\mathbb{G}$, $D_c(\mathbf{p}, \mathbf{q})$ induces a proper metric on the space of probability distribution supported on $\mathbb{G}$, commonly known as the Wasserstein distance [Villani, 2008].

The group-level loss measured by Eq. (8) is hard to calculate directly due to its high computational complexity. According to Kantorovich-Robinstein duality and Farkas Theorem [Villani, 2008], Eq. (8) is equivalent to as follow:

$$\mathcal{L}_{ot}(\mathbf{p}^{\mathbf{M^1}\theta_e^i}, \mathbf{q}^{\theta_e^j}) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{y\sim\mathbf{q}^{\theta_e^j}}[f(y)] - \mathbb{E}_{y\sim\mathbf{p}^{\mathbf{M^1}\theta_e^i}}[f(y)] \tag{9}$$

where the supremum is over all K-Lipschitz functions f. Hence, solving the optimal transport problem has been tranformed to optimize Wasserstein GAN [Arjovsky *et al.*, 2017]. We adopt a neural network to approximate the function f, since the neural networks are universal function approximators. A simple MLP can be used as the approximator, also called critic D_1. Similar to [Arjovsky *et al.*, 2017], we employ weight clipping to ensure the function family is K-Lipschitz. The loss function of the critic is defined as:

$$\max_{D_1} \mathbb{E}_{y\sim\mathbf{q}^{\theta_e^j}}[f_{D_1}(y)] - \mathbb{E}_{x\sim\mathbf{p}^{\theta_e^i}}[f_{D_1}(\mathbf{M^1}x)] \tag{10}$$

It means that the critic D_1 tries to distinguish the target embeddings and the transferred source embeddings. The D_1 denotes the distance between two sets of embeddings. The loss function of the translation matrix is defined as:

$$\min_{\mathbf{M^1}\in\mathbb{R}^{d\times d}} -\mathbb{E}_{x\sim\mathbf{p}^{\theta_e^i}}[f_{D_1}(\mathbf{M^1}x)] \tag{11}$$

where the $\mathbf{M^1}$ aims to minimize the approximate distance to fool the critic, such that the critic cannot distinguish the target embeddings and the transferred source embeddings.

Finally, the minimax loss function for group-level loss is defined as follow:

$$\mathcal{L}_{g_1} = \min_{\mathbf{M^1}} \max_{D_1} \mathbb{E}_{y\sim\mathbf{q}^{\theta_e^j}}[f_{D_1}(y)] - \mathbb{E}_{x\sim\mathbf{p}^{\theta_e^i}}[f_{D_1}(\mathbf{M^1}x)] \tag{12}$$

The similar idea applies for the other direction, transferring from target KG to the source KG. We define the similar minimax loss function:

$$\mathcal{L}_{g_2} = \min_{\mathbf{M^2}} \max_{D_2} \mathbb{E}_{y\sim\mathbf{q}^{\theta_e^i}}[f_{D_2}(y)] - \mathbb{E}_{x\sim\mathbf{p}^{\theta_e^j}}[f_{D_2}(\mathbf{M^2}x)] \tag{13}$$

Therefore, we convert the calculation and optimization of group-level loss to the problem of directly optimizing the Wasserstein GAN. By skipping the calculation step, the high computational of OT is avoidable.

3.4 Regularizer

As the discussion in Section 1, the translation matrix is desired to be orthogonal by enforcing the translation matrix to be a sparse matrix, or controlling the trend to be a dense matrix. In our work, we employ $L_{2,1}$ norm as the regularizer to prevent the translation matrix to be dense.

We define the regularizer (for two translation matrices) as:

$$\mathcal{L}_r = \alpha \left\|\mathbf{M^1}\right\|_{2,1} + \alpha \left\|\mathbf{M^2}\right\|_{2,1} = \alpha(\sum_{i=1}^{n} \left\|m^{1i}\right\|_2 + \sum_{i=1}^{n} \left\|m^{2i}\right\|_2) \tag{14}$$

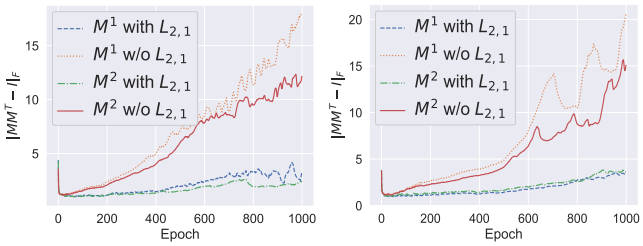


Figure 3: The impact of $L_{2,1}$ norm. The $\left\|\mathbf{MM^T} - \mathbf{I}\right\|_F$ curves of different datasets (left: En-Fr 15K, right: En-De 15K) with training epochs. $\mathbf{M^1}$ and $\mathbf{M^2}$ are the translation matrices.

Algorithm 1: OTEA

Input: KG G_i and G_j, the set of aligned entity pairs AS.
Output: Mappings $\mathbf{M}^1$, $\mathbf{M}^2$.

Initialize parameters of critics D_1 and D_2, embeddings θ_e^i, θ_e^j, and mappings $\mathbf{M}^1$, $\mathbf{M}^2$;
for *iteration = 1, ... MaxIter* **do**
  for *batch = 1, ... NumBatch_K* **do**
    Sample a batch $\hat{T}_i$ of triples (h, r, t) in G_i and $\hat{T}_j$ in G_j;
    Update θ_e^i and θ_e^j, according to $\mathcal{L}_k$ with the batch $\hat{T}_i$ and $\hat{T}_j$;
  end
  for *batch = 1, ... NumBatch_AS* **do**
    Sample a batch $\hat{AS}$ from AS;
    Update $\mathbf{M}^1$, $\mathbf{M}^2$, θ_e^i, θ_e^j according to $\{\mathcal{L}_r + \mathcal{L}_e\}$ with the batch $\hat{AS}$.
  end
  for *batch = 1, ... NumBatch_G* **do**
    Sample a batch $\hat{E}_i$ from E_i, and $\hat{E}_j$ from E_j;
    Update D_1, D_2 by fixing $\mathbf{M^1}$, θ_e^i, $\mathbf{M^2}$, θ_e^j, according to $\mathcal{L}_{g1}$ and $\mathcal{L}_{g2}$, with weight clipping c;
    Update $\mathbf{M^1}$, $\mathbf{M^2}$, θ_e^i, θ_e^j by fixing D_1, D_2, according to $\mathcal{L}_{g1}$ and $\mathcal{L}_{g2}$;
  end
end

where α is the tradeoff parameter.

To demonstrate the effectiveness of $L_{2,1}$ norm, we show $v = \left\|\mathbf{MM^T} - \mathbf{I}\right\|_F$ which measures how matrix $\mathbf{M}$ is far from an orthogonal matrix, in the training process on two datasets in Figure 3. We can observe that $\mathbf{M^1}$ and $\mathbf{M^2}$ regularized by $L_{2,1}$ norm are closer to orthogonal than those without regularization. The results demonstrate that the $L_{2,1}$ norm as a regularizer can effectively prevent the matrix to be dense, and mitigating the error induced by dense matrix.

The overall optimization process of our model is given in Algorithm 1. The embeddings of KGs and matrices are initialized by drawing from a Gaussian and orthogonal initialization, respectively. We use SGD as our optimizers, and normalize all embeddings by L_2 norm. The tradeoff parameters are set by grid search. In the testing stage, an entity e in G_i can be aligned by first transferring to G_j as $\mathbf{M^1}\theta_e$ and then selecting the most similar entity in G_j. Similarly, an entity e in G_j can be aligned by first transferring to G_i as $\mathbf{M^2}\theta_e$ and then selecting the most similar entity in G_i.

dataset	#Triple	#Entity	#Relation	#Aligned Entitiy
WK31-15K En-Fr	En: 203,502 Fr: 170,605	En: 15,170 Fr: 15,393	En: 2,228 Fr: 2,422	En-Fr: 10,108 Fr-En: 10,164
WK31-15K En-De	En: 203,502 De: 145,616	En: 15,127 De: 14,603	En: 1,841 De: 596	En-De: 11,594 De-En: 11,445
WK31-60K En-Fr	En: 569,393 Fr: 258,337	En: 64,539 Fr: 45,255	En: 458 Fr: 277	En-Fr: 48,851 Fr-En: 48,851
WK31-60K En-De	En: 569,393 De: 244,647	En: 64,539 De: 43,503	En: 458 De: 172	En-De: 46,195 De-En: 46,195
WK31-120K En-Fr	En: 1,376,011 Fr: 767,750	En: 119,749 Fr: 118,591	En:3,109 Fr:2,336	En-Fr: 117,947 Fr-En: 117,212
WK31-120K En-De	En: 1,376,011 De: 391,108	En: 67,650 De: 61,942	En: 2,393 De: 861	En-De: 55,640 De-En: 54,287

Table 1: Statistics of the WK31 dataset

4 Experiments

In this section, we conduct experiments on several real-world datasets with different sizes, and evaluate our proposed method for entity alignment. Specifically, we attempt to answer the following research questions:
(**RQ**1) Can our approach OTEA (OT-based Entity Alignment) outperform the state-of-the-art approaches?
(**RQ**2) How important the dual alignment is? comparing with the single translation.
(**RQ**3) How important the $L_{2,1}$ regularizer is?
(**RQ**4) Sensitivity of the parameter settings and complexity.

4.1 Experimental Design

Datasets. We used three trilingual knowledge graph datasets from WK31 provided in [Chen *et al.*, 2016; Chen *et al.*, 2018b]. English(En), French(Fr), and German(De) knowledge graphs are included in WK31 datasets, and the KGs are extracted from Person domain of DBpedia with known aligned entities as the ground truth. WK31 includes three datasets with different sizes, as shown in Table 1.

Baselines. To comprehensively evaluate the effectiveness of our proposed method, we include the following methods for performance comparison, including: **MTransE** [Chen *et al.*, 2016] and **ITransE** [Zhu *et al.*, 2017] encode the KGs in separated embedding space or unified embedding space, respectively. **JAPE** [Sun *et al.*, 2017] and **GCN-based method** jointly model the KGs and attributes, we only use the structure part of their models, and **BootEA** [Sun *et al.*, 2018] iteratively enlarges the labeled entity pairs based on the bootstrapping strategy.

Experimental settings. In this work, we adopt popular metrics, $Hits@k$ and MRR for evaluating entity alignment results. We find the optimal parameters or follow the settings in original papers of baselines. For our OTEA method, the best configuration is $\gamma = 0.5$, $\alpha = 0.025$, $\alpha_1 = 2.5$, weight clippling $c = 0.01$. Critics are set as two-layers MLPs with 500 hidden units. We use Adam [Kingma and Ba, 2014] to optimize the $\mathcal{L}_k$, $\mathcal{L}_e + \mathcal{L}_r$ with $lr = 0.001$, and use RMSProp [Hinton *et al.*, 2012] to optimize the $\mathcal{L}_g$ with $lr = 5e - 5$. Meanwhile, we use L_2 norm to avoid potential over-fitting. We randomly sample 30% of the aligned entities as the training set, and the rest aligned entities for testing. Each evaluation is repeated 5 times and we report the averaged $Hits@k$ and MRR.

4.2 Performance Evaluation Results (RQ1)

Table 2 shows the experimental results of baselines and our method. We can find that our proposed method consistently outperforms all baselines methods on all datasets under different evaluation metrics. Especially, we have significant improvement (10%∼50%) of $Hits@1$ value on almost all datasets, it means that our method achieves the better performance on directly successful aligning entities.

In the largest dataset WK31-120K, our method improves the best baseline with 33%∼59% under different metrics, indicating the significant advantage of our method works in the large KGs scenario. BootEA is the best baseline method in the results, since it improves the KG embedding method and also employs improved bootstrapping strategy with an alignment editing method to reduce the error accumulation. The results show that the improved bootstrapping strategy has better performance than the original bootstrapping method (ITransE). However, the error is not avoidable even though the improved method is adopted. On the largest dataset WK31-120k, it sometimes performs worse than MTransE. All the results demonstrate the advantage of our OTEA method.

4.3 Components Analysis (RQ2, RQ3)

To answer RQ2 and RQ3, we compare OTEA with its variant without dual alignment (only single translation), noted at "OTEA w/o dual", and another variant without the $L_{2,1}$ regularizer, noted at "OTEA w/o reg". The performance of "OTEA w/o dual" and "OTEA w/o reg" in Table 2 shows that these components are important for OTEA to achieve superior results. The "OTEA w/o reg" results in dense translation matrices, which introduce increased noise into the translation (demonstrated also in Figure 3). The "OTEA w/o dual" is harder than OTEA to reach the optimal and convergence, because it needs to search in a broader parameter space.

4.4 Parameter Sensitivity and Complexity (RQ4)

Sensitivity to the Proportion of Prior Aligned Entities

We randomly sample 10%, 30%, 50% and 70% of the aligned entities from WK31-15K(En-Fr) and WK31-60K(En-Fr) datasets as the training samples, and compare the performance of our model to that of other baseline models. Figure 4 shows $Hits@10$ of different methods when varying the proportion of prior aligned entities. First, as expected, all methods have better performance with the growth of the proportion of aligned entities, because more information has been provided to align the entities. Second, OTEA and BootEA have much better performance than other baselines, due to the employment of unlabeled data and the selection of labeled data, respectively. Last, OTEA is persistently better than all other baselines, including BootEA, when varying the proportion of aligned entities on two datasets.

Sensitivity to the Dimension of KG Embeddings

Figure 5(a) shows how the dimensionality of embeddings influences the performance of different entity alignment methods on WK31-15K(En-Fr) dataset. We can see that our OTEA method is consistently better than all other baselines. In addition, its performance is quite stable when varying d.

WK31-15K dataset

Language	En-Fr			Fr-En			En-De			De-En		
Metric	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR
MTransE	16.77	21.64	0.198	19.85	31.27	0.261	6.17	8.48	0.078	4.69	6.61	0.059
ITransE	18.21	24.34	0.214	18.61	33.64	0.248	15.98	28.63	0.218	13.42	25.63	0.205
JAPE	15.68	23.45	0.208	16.22	28.93	0.219	16.85	27.32	0.226	13.92	22.15	0.189
GCN	17.24	27.29	0.220	17.58	30.82	0.237	18.25	31.30	0.248	15.70	27.53	0.217
OTEA w/o reg	34.06	53.74	0.432	36.96	57.09	0.457	32.56	53.68	0.421	31.86	47.61	0.394
OTEA w/o dual	32.49	52.43	0.411	34.84	55.21	0.443	31.21	52.78	0.409	31.54	45.86	0.378
BootEA	29.72	52.92	0.395	30.77	55.44	0.428	33.13	54.13	0.435	30.47	45.33	0.381
OTEA	**37.53**	**57.74**	**0.472**	**40.47**	**60.90**	**0.502**	**37.41**	**57.19**	**0.470**	**36.80**	**56.79**	**0.465**
Improv. %	**26.28**	**9.11**	**19.50**	**31.52**	**9.85**	**17.30**	**12.92**	**5.65**	**8.04**	**20.77**	**25.28**	**22.05**

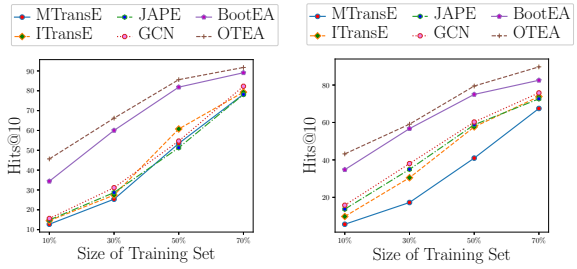
WK31-60K dataset

Language	En-Fr			Fr-En			En-De			De-En		
Metric	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR
MTransE	12.05	14.94	0.141	13.95	20.25	0.177	0.86	2.15	0.017	3.37	10.07	0.072
ITransE	17.27	25.31	0.192	18.72	32.96	0.263	15.53	24.61	0.204	16.74	24.96	0.215
JAPE	15.32	27.63	0.226	16.85	35.41	0.271	12.52	22.74	0.171	14.71	23.86	0.192
GCN	18.35	32.35	0.250	21.47	37.81	0.293	14.48	23.54	0.189	13.8	24.55	0.190
OTEA w/o reg	31.41	48.23	0.412	34.18	49.48	0.428	23.59	35.78	0.286	22.46	38.94	0.306
OTEA w/o dual	31.35	47.47	0.398	32.98	48.51	0.413	22.90	35.27	0.282	21.07	36.99	0.287
BootEA	30.82	49.42	0.406	33.31	51.14	0.425	24.45	37.63	0.308	23.28	39.29	0.316
OTEA	**34.47**	**51.51**	**0.428**	**36.07**	**54.08**	**0.447**	**27.05**	**42.12**	**0.345**	**26.97**	**43.97**	**0.352**
Improv. %	**11.84**	**4.23**	**5.42**	**8.28**	**5.75**	**5.17**	**10.63**	**11.93**	**12.01**	**15.85**	**11.91**	**11.39**

WK31-120K dataset

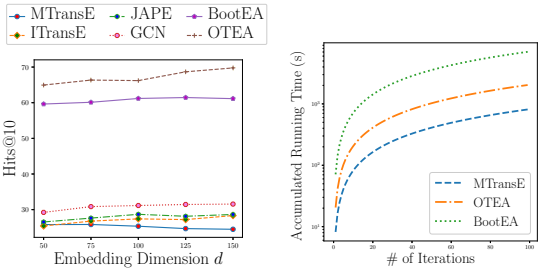
Language	En-Fr			Fr-En			En-De			De-En		
Metric	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR
MTransE	21.01	22.24	0.217	21.11	23.63	0.227	5.38	6.53	0.062	4.97	7.39	0.066
ITransE	11.54	20.41	0.176	13.35	21.20	0.197	7.62	15.54	0.112	6.41	12.82	0.085
JAPE	6.98	16.10	0.127	8.64	17.85	0.134	4.37	12.91	0.076	5.23	10.46	0.071
GCN	9.32	18.62	0.146	10.81	18.22	0.153	6.32	15.14	0.109	5.91	13.85	0.092
OTEA w/o reg	24.71	33.84	0.302	25.13	33.32	0.291	14.81	26.60	0.217	14.06	26.43	0.214
OTEA w/o dual	23.06	33.17	0.287	23.58	33.50	0.289	14.40	26.15	0.208	13.14	25.42	0.395
BootEA	17.56	27.41	0.235	18.46	28.65	0.241	11.57	22.08	0.179	10.32	22.11	0.169
OTEA	**27.92**	**37.33**	**0.328**	**28.07**	**37.41**	**0.332**	**17.98**	**30.41**	**0.244**	**17.00**	**29.46**	**0.235**
Improv. %	**59.00**	**36.20**	**39.57**	**52.06**	**30.58**	**37.76**	**55.40**	**37.73**	**36.31**	**64.73**	**33.24**	**39.05**

Table 2: Entity alignment results of different methods. The best results are in bold, along with the percentage of improvement when comparing OTEA with the best baseline method.



(a) WK31-15K(En-Fr) (b) WK31-60K(En-Fr)

Figure 4: $Hits@10$ of different methods when varying the proportion of prior aligned entities.



(a) (b)

Figure 5: (a) $Hits@10$ of different methods on WK31-15K(En-Fr) when varying dimension d. The % of prior aligned entities is 30%. (b) Accumulated running times of three methods on WK31-15K(En-Fr) with # of iterations.

Time Complexity Comparison

Figure 5(b) shows the running time comparison of the best baseline (BootEA), simplest method (MTransE) and our method (OTEA). We set same batch size for all methods and run them on a same GPU device, then record the running time of each iteration. The results show that OTEA is (3 times) faster than BootEA, because the bootstrapping based method need to propose the new aligned entities by calculating the similarity with all unaligned entities. Our method need more time than MTransE, but it is worthwhile to spend the time to achieve significant improvement on the task.

5 Conclusion

We introduced a novel framework for cross-lingual entity alignment in knowledge graphs. We proposed to solve the entity alignment by dually minimizing both the entity-level loss and the group-level loss via optimal transport theory, in order to model the whole picture of labeled and unlabeled entities in different language KGs. We also impose $L_{2,1}$ regularizer on the dual translation matrices to mitigate the effect of noise during transformation. Our experiments on real-world datasets demonstrated that our approach achieved superior results comparing with other state-of-the-art methods on alignment accuracy.

Acknowledgments

The research reported in this publication was supported by funding from King Abdullah University of Science and Technology (KAUST), under award number FCC/1/1976-19-01, and NSFC No. 61828302.

References

[Arjovsky *et al.*, 2017] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017.

[Bordes *et al.*, 2013] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, 2013.

[Chen *et al.*, 2016] Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. *arXiv preprint arXiv:1611.03954*, 2016.

[Chen *et al.*, 2018a] Liqun Chen, Shuyang Dai, Chenyang Tao, Haichao Zhang, Zhe Gan, Dinghan Shen, Yizhe Zhang, Guoyin Wang, Ruiyi Zhang, and Lawrence Carin. Adversarial text generation via feature-mover's distance. In *NIPS*, 2018.

[Chen *et al.*, 2018b] Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena, and Carlo Zaniolo. Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. *IJCAI*, 2018.

[Cuturi, 2013] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NIPS*, 2013.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[Hinton *et al.*, 2012] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14, 2012.

[Hu *et al.*, 2011] Wei Hu, Jianfeng Chen, and Yuzhong Qu. A self-training approach for resolving object coreference on the semantic web. In *WWW*, 2011.

[Ji *et al.*, 2015] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *ACL*, 2015.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Lin *et al.*, 2015] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, 2015.

[Luise *et al.*, 2018] Giulia Luise, Alessandro Rudi, Massimiliano Pontil, and Carlo Ciliberto. Differential properties of sinkhorn approximation for learning with wasserstein distance. In *NIPS*, 2018.

[Mahdisoltani *et al.*, 2013] Farzaneh Mahdisoltani, Joanna Biega, and Fabian M Suchanek. Yago3: A knowledge base from multilingual wikipedias. In *CIDR*, 2013.

[Nentwig *et al.*, 2017] Markus Nentwig, Michael Hartung, Axel-Cyrille Ngonga Ngomo, and Erhard Rahm. A survey of current link discovery frameworks. *Semantic Web*, 8(3):419–436, 2017.

[Ng and Cardie, 2002] Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution. In *ACL*, 2002.

[Nguyen *et al.*, 2011] Thanh Nguyen, Viviane Moreira, Huong Nguyen, Hoa Nguyen, and Juliana Freire. Multilingual schema matching for wikipedia infoboxes. *Proceedings of the VLDB Endowment*, 5(2):133–144, 2011.

[Nie *et al.*, 2010] Feiping Nie, Heng Huang, Xiao Cai, and Chris H Ding. Efficient and robust feature selection via joint l2, 1-norms minimization. In *NIPS*, 2010.

[Peyré and Cuturi, 2018] Gabriel Peyré and Marco Cuturi. Computational optimal transport. *arXiv preprint arXiv:1803.00567*, 2018.

[Peyré *et al.*, 2012] Gabriel Peyré, Jalal Fadili, and Julien Rabin. Wasserstein active contours. In *ICIP*, 2012.

[Smith *et al.*, 2017] Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *ICLR*, 2017.

[Sun *et al.*, 2017] Zequn Sun, Wei Hu, and Chengkai Li. Cross-lingual entity alignment via joint attribute-preserving embedding. In *ISWC*, 2017.

[Sun *et al.*, 2018] Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. Bootstrapping entity alignment with knowledge graph embedding. In *IJCAI*, 2018.

[Villani, 2008] Cédric Villani. *Optimal transport: old and new*. Springer Science & Business Media, 2008.

[Vrandečić and Krötzsch, 2014] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 2014.

[Wang *et al.*, 2014] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *AAAI*, 2014.

[Wang *et al.*, 2018] Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. Cross-lingual knowledge graph alignment via graph convolutional networks. In *EMNLP*, 2018.

[Xu *et al.*, 2018] Hongteng Xu, Wenlin Wang, Wei Liu, and Lawrence Carin. Distilled wasserstein learning for word embedding and topic modeling. In *NIPS*, 2018.

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

[Yang *et al.*, 2015] Yang Yang, Yizhou Sun, Jie Tang, Bo Ma, and Juanzi Li. Entity matching across heterogeneous sources. In *KDD*, 2015.

[Yin *et al.*, 2015] Penghang Yin, Yifei Lou, Qi He, and Jack Xin. Minimization of 1-2 for compressed sensing. *SIAM Journal on Scientific Computing*, 2015.

[Zhu *et al.*, 2017] Hao Zhu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun. Iterative entity alignment via joint knowledge embeddings. In *IJCAI*, 2017.