Dangling-Aware Entity Alignment with Mixed High-Order Proximities

Juncheng Liu[1] **Zequn Sun**[2] **Bryan Hooi**[1] **Yiwei Wang**[1]
Dayiheng Liu[3] **Baosong Yang**[3] **Xiaokui Xiao**[1] **Muhao Chen**[4]
[1]National University of Singapore [2]Nanjing University
[3]Alibaba Group [4]University of Southern California
juncheng.liu@u.nus.edu, zqsun.nju@gmail.com
muhaoche@usc.edu

Abstract

We study dangling-aware entity alignment in knowledge graphs (KGs), which is an underexplored but important problem. As different KGs are naturally constructed by different sets of entities, a KG commonly contains some dangling entities that cannot find counterparts in other KGs. Therefore, dangling-aware entity alignment is more realistic than the conventional entity alignment where prior studies simply ignore dangling entities. We propose a framework using mixed high-order proximities on dangling-aware entity alignment. Our framework utilizes both the local high-order proximity in a nearest neighbor subgraph and the global high-order proximity in an embedding space for both dangling detection and entity alignment. Extensive experiments with two evaluation settings shows that our framework more precisely detects dangling entities, and better aligns matchable entities. Further investigations demonstrate that our framework can mitigate the hubness problem on dangling-aware entity alignment.

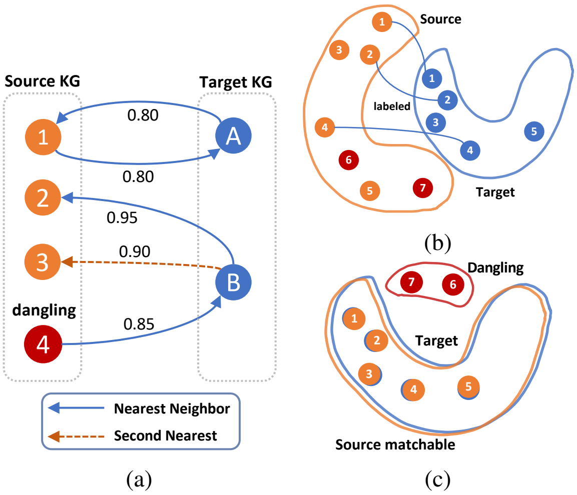


Figure 1: Toy examples for mixed high-order proximities. (a) Nearest neighbor (NN) subgraph where entities connect to NNs in the other KG using embedding similarities. 4 and its nearest neighbor B have 0.85 similarity. B prefers 2 and 3 with higher similarities. 1 and A are mutual nearest neighbors. (b) Labeled alignments and dangling entities. (c) Aligning matchable source and target distributions rather than only labeled alignments.

1 Introduction

Knowledge graphs (KGs) have become the backbone of many intelligent applications (Ji et al., 2021). In spite of their importance, many KGs are independently created without considering the interrelated and interchangeable nature of individually created knowledge (Chen et al., 2020). To allow complementary knowledge to be automatically combined and migrated across individual KGs, entity alignment seeks to identify equivalent entities in distinct KGs (Sun et al., 2020a). Recent literature has focused on learning embedding representations of multiple KGs where identical entities are aligned based on their embedding similarity (Chen et al., 2017; Cao et al., 2019; Fey et al., 2020; Sun et al., 2020a; Liu et al., 2021).

Aside from the surge of research effort on entity alignment (Zeng et al., 2021), an unresolved but important challenge that existing methods face is the *dangling entity* problem. Dangling entities are those unique entities in a KG that cannot find counterparts in another KG. Considering that individually created KGs are unlikely to share the same set of entities, identifying dangling entities is undoubtedly an indispensable step of any practical solution to entity alignment. However, nearly all prior studies have neglected dangling entities and assume there must be one-to-one entity mapping from the source KG to the target one (Sun et al., 2020c). This assumption prevents prior methods from practically supporting the alignment between KGs in real-world scenarios. To fill the gap, Sun et al. (2021) formally define a more practical problem setting where a model needs to both determine whether each given source entity is a matchable one, as well as retrieve counterparts for the predicted matchable entities.

Although some preliminary attempts have been made to implement dangling-aware entity alignment (Sun et al., 2021), the attempted methods still suffer from a major drawback, i.e., they only consider the first-order proximity (namely, pairwise cosine similarity) between source and target entities. However, to effectively discover dangling entities as outliers in the embedding representation, we argue that *high-order proximity measures* should also be involved. Fig. 1a shows the intuition of using the high-order proximity for dangling entity detection. Despite a fairly high cosine similarity, the source entity 4 is not the nearest neighbor of target entity B, indicating that 4 is likely to be dangling. In contrast, 1 and A are mutual nearest neighbors even with a relatively low similarity, indicating that 1 is more likely to be matchable. Hence, in addition to the first-order proximity from source to target, the *local high-order proximity* (e.g., the second-order proximity[1] in the nearest neighbor subgraph) is also useful for detecting dangling entities. In alignment learning, the previous works neglect global information since they merely optimize the entity-level alignment loss on labeled alignments without considering entity embedding distributions as shown in Fig. 1b. In Fig. 1c, we show that a desirable dangling-aware model should align the *global* distributions of matchable source and target entities (i.e., *global high-order proximity* in an embedding distribution space), such that dangling entities in both KGs could appear as dissimilar outliers in both distributions.

Motivated by the above intuition, we propose a dangling-aware entity alignment framework based on *m̲ixed h̲igh-order p̲roximities* (MHP). MHP considers both local and global high-order proximities to foster both dangling entity detection and matchable entity alignment. We introduce the optimization of global high-order proximity measure as finding the Optimal Transport between matchable source entities and target entities. Through this optimization process, to facilitate dangling detection, MHP also encourages a large distance between the dangling entity distribution and matchable entity distribution. Additionally, to leverage the local high-order proximity, we propose a dangling entity classifier which takes into account the second-order proximity in the nearest neighbor subgraph. Furthermore, with the similar principle of local high-order proximity, we adopt an NCA (Neighborhood Component Analysis) loss (Goldberger et al., 2004; Liu et al., 2021) for alignment learning to mitigate the hubness problem[2] (Radovanovic et al., 2010), which is severe in dangling-aware entity alignment as observed in our experiments.

Our main technical contributions to the studied problem are two-fold. First, the local high-order proximity (i.e., the second-order proximity in the nearest neighbor subgraph) is modeled to facilitate both dangling detection and alignment learning. Second, we design the use of the global high-order proximity to align the distributions of matchable entities, therefore precisely separating the representations of dangling entities and matchable ones. In addition, the techniques are model-agnostic and can be incorporated with various alignment methods (e.g., MTransE (Chen et al., 2017) or AliNet (Sun et al., 2020a)) and dangling detection methods (e.g., the marginal or background ranking (Sun et al., 2021)). Extensive experiments on DBP2.0 demonstrate its effectiveness and adaptiveness.

2 Preliminary

In this section, we provide the problem definition of dangling-aware entity alignment and briefly introduce previous methods for this problem.

2.1 Problem definition

A KG is defined as $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, where $\mathcal{E}$ denotes a set of entities; $\mathcal{R}$ denotes a set of relations, and $\mathcal{T} \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ is a set of triples. Following the convention (Chen et al., 2017), we consider entity alignment between a source KG $\mathcal{G}_s = (\mathcal{E}_s, \mathcal{R}_s, \mathcal{T}_s)$ and a target KG $\mathcal{G}_t = (\mathcal{E}_t, \mathcal{R}_t, \mathcal{T}_t)$. Our study focuses on a more practical and challenging setting with dangling entities (Sun et al., 2021). In this setting, the training data contain a set of seed entity alignment $\mathcal{A} = \{(x_s, x_t) \in \mathcal{E}_s \times \mathcal{E}_t \| x_s \equiv x_t\}$ and a set of source dangling entities $\mathcal{D} \subset \mathcal{E}_s$ that has no counterparts in target KG. After training, the model is required to first identify dangling entities and then find alignment for predicted matchable entities. This definition breaks the one-to-one assumption used in previous studies on the conventional setting (Sun et al., 2020c) and causes their methods to not be directly usable in our setting.

[1] The second-order proximity of a source entity s is defined as aggregated cosine similarities between the nearest targets of s and the nearest sources of these nearest targets.

[2] The hubness problem is where some target entities dominantly appear as the nearest neighbors of many source entities.

2.2 Dangling-aware entity alignment

To the best of our knowledge, there is only one previous work (Sun et al., 2021) which has been attempted for dangling-aware entity alignment along with the proposing of this important problem. This work also incorporates an embedding-based entity alignment technique (i.e., MTransE (Chen et al., 2017) and AliNet (Sun et al., 2020a)) as the backbone, which learns alignment of KG embeddings based on the seed entity pairs. Taking MTransE as an example, for each pair $(x_s, x_t) \in \mathcal{A}$, MTransE uses the learned embedding $\mathbf{x}_s$ and $\mathbf{x}_t$ to optimize a linear transformation matrix $\mathbf{M}$ by minimizing $||\mathbf{M}\mathbf{x}_s - \mathbf{x}_t||$. To detect dangling entities in the embedding space, a margin ranking (MR) loss and a background ranking (BR) loss are experimented with, both encouraging dangling entities to be isolated from others in the embedding space. MR sets a distance margin λ to separate the dangling entity x and its nearest neighbors by minimizing $\max(0, \lambda - \|\mathbf{M}\mathbf{x} - \mathbf{x}_{\text{nn}}\|)$. In like manner, BR treats dangling entities as the background of embedding space and learns to separate dangling entities and randomly-sampled other entities.

3 Methodology

In this section, we introduce the techniques in our framework which captures both local and global high-order proximities to collaboratively tackle dangling detection and entity alignment.

3.1 Global high-order proximity

To leverage the global high-order proximity in an embedding space, in MHP, we introduce a method based on Optimal Transport (OT) for globally aligning the distributions of matchable source and target entities. In addition, to facilitate dangling entity detection, the OT model encourages a large distribution distance between source- and target-KG dangling entities. Intuitively, this strategy treats dangling entities as dissimilar parts of two embedding distributions, therefore tending to put dangling entities as outliers in the embedding space.

Optimal transport. Let $\mathbf{s}$ and $\mathbf{t}$ be the distribution of transformed source-space embeddings $\mathbf{M}\mathbf{x}_s$ and target space embeddings $\mathbf{x}_t$, respectively.[3] Intuitively, $\mathbf{s}$ should be similar with $\mathbf{t}$ if they represent matchable entities. Meanwhile, to make dangling entities distinguishable, the distribution of transformed dangling entity vectors $\mathbf{M}\mathbf{x}_d$ should be different from $\mathbf{t}$. The discrepancy between $\mathbf{s}$ and $\mathbf{t}$ can be represented as a Wasserstein distance which is one type of OT distance (Peyré et al., 2019):

$$\mathcal{D}_c(\mathbf{s}, \mathbf{t}) = \inf_{\gamma \in \Pi(\mathbf{s}, \mathbf{t})} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma}[c(\mathbf{x}, \mathbf{y})], \quad (1)$$

where $\Pi(\mathbf{s}, \mathbf{t})$ is the set of all possible joint distributions $\gamma(\mathbf{s}, \mathbf{t})$ and $c(\mathbf{x}, \mathbf{y})$ denotes the cost function describing the distance between $\mathbf{x}$ and $\mathbf{y}$. Then the Wasserstein distance $\mathcal{D}_c(\mathbf{s}, \mathbf{t})$ denotes the cost of the optimal transport plan.

However, the infimum to calculate $\mathcal{D}_c(\mathbf{s}, \mathbf{t})$ is highly intractable (Arjovsky et al., 2017). To handle this, the Kantorovich-Rubinstein duality points out that Eq. (1) can be transformed to:

$$\mathcal{L}_{ot}(\mathbf{s}, \mathbf{t}) = \frac{1}{K} \sup_{\|f\|_L \leq K} \mathbb{E}_{x \sim \mathbf{t}}[f(x)] - \mathbb{E}_{x \sim \mathbf{s}}[f(x)], \quad (2)$$

where the supremum is over all possible K-Lipschitz functions f. As Arjovsky et al. (2017) point out that optimizing Wasserstein GAN (WGAN) can be used to solve this optimal transport problem, we utilize WGAN in our study. Specifically, we adopt a MLP to approximate the function f (called as critic D) since neural networks are universal approximators (Hornik et al., 1989). The objective of the critic is defined as follows:

$$\max_D \mathbb{E}_{y \sim \mathbf{t}} \left[f_D(y)\right] - \mathbb{E}_{x \sim \mathbf{s}} \left[f_D\left(\mathbf{M}x\right)\right]. \quad (3)$$

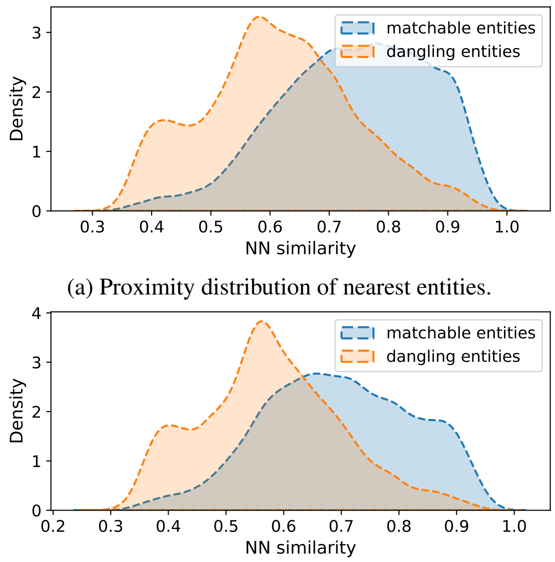
Thus, the critic D aims to distinguish transformed source embeddings from target embeddings. In contrast, the transformation matrix $\mathbf{M}$ tries to minimize the distance between the two sets of embeddings. The objective to optimize $\mathbf{M}$ is defined as:

$$\begin{aligned}&\min_{\mathbf{M}} \mathbb{E}_{y \sim \mathbf{t}} \left[f_D(y)\right] - \mathbb{E}_{x \sim \mathbf{s}} \left[f_D\left(\mathbf{M}x\right)\right] \\ &= \min_{\mathbf{M}} -\mathbb{E}_{x \sim \mathbf{s}} \left[f_D\left(\mathbf{M}x\right)\right]. \quad (4)\end{aligned}$$

Therefore, conducting entity alignment with the consideration of whole embedding distributions is converted to the problem of optimizing a WGAN.

So far, only the distribution of matchable source entity embeddings and that of target entity embeddings are considered, whereas the distribution of dangling entity embeddings is neglected. Therefore, to tailor the optimization problem for dangling entities, we adopt an additional objective for the

[3] Without loss of generality, we use a matrix M to transform embeddings from source KG to target KG.



(a) Proximity distribution of nearest entities.

(b) Proximity distribution of the second nearest entities.

Figure 2: Second-order proximity distributions.

transformation matrix $\mathbf{M}$:

$$\max_{\mathbf{M}} \mathbb{E}_{y\sim\mathbf{t}}\left[f_D(y)\right] - \mathbb{E}_{x\sim\mathbf{d}}\left[f_D\left(\mathbf{M}x\right)\right] = \min_{\mathbf{M}} \mathbb{E}_{x\sim\mathbf{d}}\left[f_D\left(\mathbf{M}x\right)\right], \qquad (5)$$

where $\mathbf{d}$ denotes the distribution of dangling entity embeddings. Hence, the transformation matrix $\mathbf{M}$ is enforced to maximize the difference between the distribution of transformed dangling embeddings and that of target entity embeddings, which can make dangling entities more distinguishable.

3.2 Local high-order proximity

In addition to the global proximity measure, MHP also captures local high-order proximity measures in the nearest neighbor subgraph. In contrast, the previous work (Sun et al., 2021) merely uses the first-order proximity between an individual source entity s and its nearest target entity to decide whether s should be dangling. However, apart from the first-order proximity, the second-order proximity measure could be informative as well for detecting dangling entities.

Furthermore, we verify the above hypothesis empirically using the previous work. From the nearest target entity t of a given source entity, we obtain the cosine similarities between t and its top 2 nearest source entities as the second-order proximity measures, and plot the proximity distributions in Fig. 2a and 2b for the first and second nearest entities, respectively. Fig. 2a shows that the second-order proximity between matchable entities and their nearest neighbors appear as a very different distribution in comparison to that of the proximity between dangling entities and their nearest neighbors. Fig. 2b demonstrates a similar observation for the proximity distributions of the second nearest entities. Therefore, the second-order proximities are informative and should be used for distinguishing dangling entities, meanwhile combining proximity measures that consider multiple neighboring entities is more useful than a single similarity measure on only the nearest entity.

To this end, we design a dangling entity classifier using both first-order and second-order proximity measures as the input. From the perspective of a given source entity s, we conduct nearest neighbor search to obtain top k nearest target entities $\{t_1, ..., t_k\}$ and their proximity score vector $\mathbf{d}_1 = [d_{st_1}, ..., d_{st_k}] \in \mathbb{R}^{1\times k}$. The proximity d_{st} is measured by the cosine similarity between transformed source embedding $\mathbf{M}\mathbf{x}_s$ and target entity embedding $\mathbf{x}_t$:

$$d_{st_1} = \left\langle \frac{\mathbf{M}\mathbf{x}_s}{\|\mathbf{M}\mathbf{x}_s\|_2}, \frac{\mathbf{x}_t}{\|\mathbf{x}_t\|_2} \right\rangle \qquad (6)$$

After getting the first-order proximity vector $\mathbf{d}_1$ between the source and target, through the reverse direction (target KG to source KG), we can further obtain the second-order proximity vector. Specifically, we retrieve top m nearest source entities $\{s_1, ..., s_m\}$ of each target entity t in $\{t_1, ..., t_k\}$. Accordingly, through the target entity t, the second-order proximity measures with regard to the m retrieved source entities are obtained as the vector $\mathbf{d}_t = [d_{s_1t}, ..., d_{s_mt}] \in \mathbb{R}^{1\times m}$. Subsequently, we can collect $\{\mathbf{d}_{t_1}...\mathbf{d}_{t_k}\}$ and concatenate them as the whole second-order proximity vector $\mathbf{d}_2 = [\mathbf{d}_{t_1}||...||\mathbf{d}_{t_k}] \in \mathbb{R}^{1\times km}$.

To utilize both second-order and first-order information, the whole proximity distribution vector is constructed as $\mathbf{d} = [\mathbf{d_1}||\mathbf{d_2}] \in \mathbb{R}^{1\times(k+1)m}$. In this way, we use the distribution as profile of the neighborhood of a source entity s, then we adopt a simple feed-forward neural network (FNN) binary classifier to determine whether s is dangling. The probability of s being a dangling entity can be calculated as $p(y = 1|s) = \text{sigmoid}(\text{FNN}(\mathbf{d}))$. Define $\mathcal{D}$ and $\mathcal{A}$ to be the training set of dangling entities and that of matchable entities, respectively.

We minimize the binary cross-entropy loss:

$$\mathcal{L}_s = -\frac{1}{|\mathcal{D} \cup \mathcal{A}|} \sum_{s \in \mathcal{D} \cup \mathcal{A}} (y_s \log(p(y=1|s)) + (1-y_s)\log(1-p(y=1|s))) \quad (7)$$

NCA loss. With the similar principle of local high-order proximity, MHP adopts an additional Neighbor Component Analysis (NCA) loss (Liu et al., 2021) to mitigate the hubness problem. The hubness problem can be more severe in dangling-aware entity alignment as dangling entities might be aligned to some certain hubs if they are not detected as dangling. The NCA loss measures importance of samples and punishes hard negative pairs based on the proximities. Given the set of seed entity alignments $\{(x_s, x_t) \in \mathcal{E}_s \times \mathcal{E}_t\}$, let $\mathbf{S}$ be the cosine similarity matrix between source and target entity embeddings $\mathbf{E}_1$ and $\mathbf{E}_2$. The NCA loss can be defined as follows:

$$\mathcal{L}_{NCA} = \frac{1}{N}\sum_{i=1}^{N}\Big(\frac{1}{\alpha}\log(1+\sum_{m \neq i} e^{\alpha \mathbf{S}_{im}}) + \frac{1}{\alpha}\log(1+\sum_{n \neq i} e^{\alpha \mathbf{S}_{ni}}) - \log(1+\beta e^{\mathbf{S}_{ii}})\Big), \quad (8)$$

where $\mathbf{S}_{ii}$ denotes the proximity of the i-th positive pair (i.e., the i-th source entity and the i-th target entity); α, β are temperature hyper-parameters; and N is the number of positive pairs in the mini-batch.

3.3 Learning and inference

Note that our techniques are used to improve existing first-order methods. MHP optimizes the entity alignment component and the dangling detection component alternately. For entity alignment, besides an entity-level loss (e.g., MTransE), we first train WGAN for optimal transport and then optimize the NCA loss $\mathcal{L}_{NCA}$. For dangling detection, besides a first-order objective (e.g., a marginal ranking loss) used in Sun et al. (2021), we train our dangling classifier for detection. In the inference phase, for each source entity, the dangling entity classifier provides a probability score and uses a probability threshold to decide whether an entity is dangling, where the threshold is set as the average probability. After this dangling detection process, the predicted dangling entities are excluded from being aligned. Then, in the alignment process, MHP conducts nearest neighbor search to find the alignment in the target KG embedding space for each of the rest matchable source entities.

4 Experiments

In this section, we report our experiments to show the effectiveness of MHP. We describe the evaluation settings in Sec. 4.1, and present the results in two alignment settings separately in Sec. 4.2 and 4.3. We conduct an ablation study and demonstrate that MHP can mitigate the hubness problem in Sec. 4.4, followed by a case study to show the importance of local high-order proximity in Sec. 4.5.

4.1 Experimental settings

We use two evaluation settings as suggested by Sun et al. (2021). The first one is *consolidated evaluation* which requires a model to first detect and remove dangling entities, and then conduct alignment search for the rest of entities. The performance of dangling entity detection is also evaluated in this setting. Besides, a simplified *relaxed evaluation setting* seeks to test the performance of alignment alone without involving dangling source entities in the test set. In this setting, the effect of dangling detection on entity alignment can be evaluated.

Evaluation protocol. For the *relaxed setting*, the counterpart list is selected by the Nearest Neighbor (NN) search in the embedding space for each source entity. To assess the ranking list, we use mean reciprocal rank (MRR), Hits@1 and Hits@10 (hereinafter H@1 and H@10) as metrics. Higher values indicate better performance.

For the *consolidated setting*, we evaluate the performance of both dangling entity detection and entity alignment using precision, recall, and F1 score, following Sun et al. (2021).[4] In this setting, only the source entities that are correctly predicted as matchable are sent to the NN search and the nearest counterpart is evaluated. Particularly, incorrect dangling detection (i.e., a matchable entity is wrongly predicated as dangling or a dangling entity is predicted as matchable) will propagate an error case to the alignment process. We refer to this practical entity alignment as *two-step entity alignment*.

Dataset. We use the cross-lingual dangling-aware entity alignment dataset DBP2.0 (Sun et al., 2021), which is constructed using multilingual DBpedia (Lehmann et al., 2015). There are three language pairs (ZH-EN, JA-EN, FR-EN) in DBP2.0 and two alignment directions are considered for each pair. We follow its data splits where 30% dangling

[4] Note that H@1, H@10 and MRR are not applicable to this entity alignment in the consolidated setting.

entities are for training, 20% for validation, and 50% for test. The dataset statistics are reported in Appx. A.

Baselines. To the best of our knowledge, the framework with a dangling detection module proposed in Sun et al. (2021) is the only study on dangling-aware entity alignment. It includes three dangling detection techniques: (i) NN classification, (ii) marginal ranking (MR), and (iii) background ranking (BR). As the NN classification performs much worse than others, we choose MR and BR as baselines. For a fair comparison with (Sun et al., 2021), we use the same base alignment model MTransE (Chen et al., 2017). The results using AliNet (Sun et al., 2020b) [5] as a base are presented in Appx. D. Note that our methods are model-agnostic and can be incorporated with any detection and alignment methods. The entity alignment models that consider side information are left for future work.

Model configuration. In MHP, aside from our proposed components as described in Sec. 3, we have a base alignment module (e.g., MTransE) and a base dangling entity loss (e.g., MR) as in Sun et al. (2021). For KG embeddings and model weights, we use Xavier initialization (Glorot and Bengio, 2010) and optimize them using Adam optimizer (Kingma and Ba, 2014). The number of hidden units in the dangling entity classifier is 128. The number of nearest targets k and nearest sources m are set as 5. The learning rate is set to 0.001 for all components except WGAN where the learning rate is 5e-5 for three objectives. To terminate training, early stopping is used based on the F1 score of two-step entity alignment on validation set. The computational environment and other configuration details are reported in Appx. B and C.

4.2 Consolidated evaluation

Dangling entity detection. According to the results in Tab. 1, no matter which base dangling detection loss we adopt, MHP consistently achieves better F1 scores compared with the corresponding baseline by Sun et al. (2021) without our proposed techniques. In terms of the recall, MHP also outperforms baselines with a large margin, which indicates that our framework has a better coverage to find more dangling entities. With better recalls, MHP has the same level or slightly worse precision compared with baselines. But our higher recall and F1 scores in dangling detection imply that more predicted matchable source entities would enter two-step entity alignment, which can improve the final alignment performance. Comparing MHP + MR and MHP + BR, we can see that the MR variant is generally better than the BR variant. This is because MR considers the similarity between a source and its nearest neighbor, which can benefit the learning of local high-order proximity in MHP. In summary, MHP demonstrates superior effectiveness for detecting dangling entities.

Two-step entity alignment. The results of two-step alignment are shown in Tab. 2. In general, MHP again consistently offers better F1 scores than baseline methods. The relative improvement ranges from 11% to 32%. The improvement can be partly attributed to the more accurate dangling entity detection performance, and thus less error is propagated to the alignment process. In contrast, baselines may try to align many dangling entities, which leads to lower performance on two-step alignment. As MHP with MR outperforms MHP + BR in dangling detection, MHP + MR also achieves better performance in two-step alignment. This indicates that dangling entity detection is of importance on the dangling-aware entity alignment problem since it has strong effects on the performance of two-step alignment.

4.3 Relaxed evaluation

Tab. 3 shows the results of relaxed evaluation. This setting only considers matchable source entities in the test phase to investigate how our framework affects the alignment learning of these entities.

Generally, MHP offers better performance than baselines on all language pairs in terms of all metrics. This indicates that dangling awareness captured by MHP further helps with a more precise alignment. The improvement can also be partly attributed to the alleviated hubness problem by the NCA loss which we investigate more in Sec. 4.4. Comparing two variants of MHP, we can see that MHP + MR usually outperforms the BR variants on most language pairs except for FR-EN. The reason could be that FR-EN has more entities and only with sufficient data BR can effectively separate dangling entities from randomly sampled target entities, while MR is not sensitive to data volume.

[5] AliNet performs worse than MTransE on dangling-aware entity alignment as found by Sun et al. (2021).

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
MR	.781	.702	.740	.866	.675	.759	.799	.708	.751	.864	.653	.744	.482	.575	.524	.639	.613	.625
BR	**.811**	.728	.767	**.892**	.700	.785	**.816**	.733	.772	**.888**	.731	.801	.539	.686	.604	.692	.735	.713
MHP + MR	.784	**.831**	**.807**	.858	**.861**	**.859**	.815	**.791**	**.803**	.865	**.852**	**.858**	**.580**	**.724**	**.644**	**.707**	**.749**	**.727**
MHP + BR	.758	.815	.785	.832	.847	.839	.783	.785	.784	.834	.848	.841	.569	.706	.635	.685	.747	.714

Table 1: Dangling entity detection results on DBP2.0. MR refers to marginal ranking and BR refers to the background ranking. The base alignment model is MTransE. More results based on AliNet are in Appx. Tab. 7.

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
MR	.302	.349	.324	.231	.362	.282	.313	.367	.338	.227	.366	.280	.260	.220	.238	.213	.224	.218
BR	.312	.362	.335	.241	**.376**	.294	.314	.363	.336	.251	.358	.295	.265	.208	.233	.231	.213	.222
MHP + MR	**.400**	**.363**	**.381**	**.375**	.372	**.373**	**.378**	**.394**	**.386**	**.371**	**.384**	**.377**	**.310**	**.249**	**.276**	.266	**.260**	**.263**
MHP + BR	.393	.347	.368	.347	.331	.339	.374	.372	.373	.359	.344	.352	.290	.235	.259	**.269**	.239	.253

Table 2: Two-step entity alignment results on DBP2.0. The base alignment model is MTransE.

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	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
MTransE	.358	.675	.463	.353	.670	.461	.348	.661	.453	.342	.670	.452	.245	.524	.338	.247	.531	.342
w/ MR	.378	.693	.487	.383	.699	.491	.373	.686	.476	.374	.707	.485	.259	.541	.348	.265	.553	.360
w/ BR	.360	.678	.468	.357	.675	.465	.344	.660	.451	.346	.675	.456	.251	.525	.342	.249	.531	.343
MHP + MR	**.418**	**.727**	**.523**	**.404**	**.724**	**.513**	**.408**	**.730**	**.517**	**.410**	**.747**	**.524**	.274	.568	.371	**.274**	.566	**.370**
MHP + BR	.412	.718	.517	.396	.714	.505	.400	.727	.511	.400	.728	.511	**.278**	**.574**	**.376**	.272	**.569**	**.370**

Table 3: Entity alignment results in the relaxed setting on DBP2.0.

Methods	Dangling detection		Two-step alignment	
	F1	Δ	F1	Δ
MHP	.807	0	.381	0
- Dangling cls.	.752	-.055	.339	-.042
- OT	.789	-.018	.369	-.012
- NCA	.803	-.004	.361	-.020

Table 4: Ablation study in the consolidated setting on ZH-EN. We remove each technique and report the performance decline Δ compared with the full MHP.

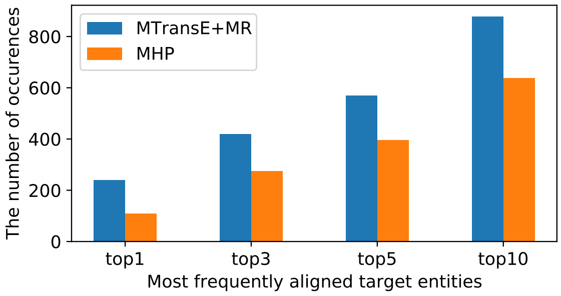


Figure 3: The number of total occurrences of most frequently aligned target entities on ZH-EN.

4.4 Ablation study

To investigate the effectiveness of each module in MHP, we conduct an ablation study on the consolidated setting and show the results in Tab. 4. Compared with the full version MHP, removing any component causes the degraded performance. Specifically, by removing the dangling classifier, the F1 score of dangling detection drops 0.055, which also leads to a large performance drop on two-step alignment. This indicates that the local high-order proximity is useful for dangling detection. Removing OT decreases the F1 scores on both detection and two-step alignment, showing the effectiveness of globally aligning distributions. Lastly, leaving the NCA loss out makes the F1 score of two-step alignment decrease 0.02 compared with MHP, because using the NCA loss reduces the extent of hubness, as discussed below.

Hubness problem. To examine whether the NCA loss reduces the hubness problem, we list a set of most frequently aligned (target) entities, and observe how frequently they appear as the nearest neighbor of other entities in the embedding space. We compare MHP with the MTransE + MR variant used by Sun et al. (2021). As shown in Fig. 3, the most frequently aligned target entity (i.e., top 1) appears over 200 times as the nearest neighbor using the baseline, whereas it only appears around 100 times using MHP. A similar phenomenon is also observed for the top 3, top 5, and top 10 frequently aligned target entities. This indicates that the hubness problem is mitigated by using NCA.

4.5 Case study

To further investigate the superiority of MHP, we provide a case study on ZH-EN comparing MHP with the previous method. Fig. 4 shows that, the previous method predicts some dangling source entities as matchable based on their high cosine similarities (i.e., > 0.7) to their nearest target entities. Each dangling entity and its corresponding nearest target entity are different but share similar meanings (e.g., are both war events in ancient China or locations). However, the nearest targets prefer other source entities with higher similarities. Using this second-order proximity information, MHP correctly detects these dangling entities with high probability scores (i.e., > 0.9).

Additionally, Tab. 5 demonstrates more dangling entities which are not correctly detected by the previous method (Sun et al., 2021). Most of the dangling entities are aligned to some similar counterparts sharing the same attribute. For example, the dangling entity and its nearest target entity are both colleges, theoretical physicists, or political parties. However, from the view of the nearest target entity, it prefers other nearest neighbors on source KG. Such second-order proximity information cannot be captured by the previous method, which causes those dangling entities not able to be detected. In contrast, MHP can successfully detect those dangling entities with high probabilities. This shows the effectiveness of MHP and the informativeness of the second-order proximity.

5 Related Work

Entity alignment. Embedding-based entity alignment methods seek to find identical entities between KGs in their embedding spaces. Such a method encodes each KG into an embedding space and capture entity alignment by learning a linear mapping between embedding spaces (Chen et al., 2017) or directly infer the embedding proximity in a shared space (Sun et al., 2017). Existing studies mainly fall into two lines of improving the embedding representations. The first line exploits better graph encoders to improve embedding learning (Sun et al., 2018; Wang et al., 2018; Cao et al., 2019; Sun et al., 2020a,b; Fey et al., 2020). The second group considers the side information of entities (Chen et al., 2018b; Trisedya et al., 2019; Zhang et al., 2019; Xu et al., 2019b; Wang et al., 2020; Tang et al., 2020; Wu et al., 2019; Yang et al., 2019; Liu et al., 2020, 2021). Interested readers can refer to the recent surveys (Sun et al., 2020c; Zeng et al., 2021). Note that prior methods nearly all assume one-to-one perfect match exists between two KGs, without considering dangling entities.

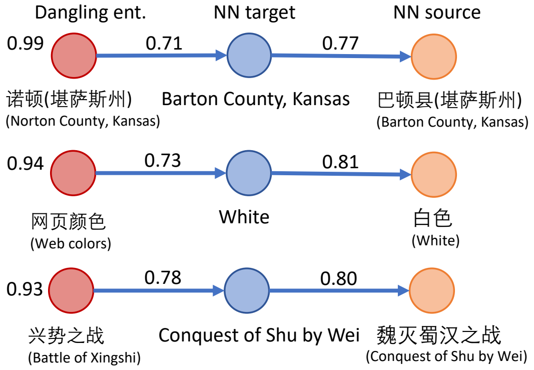


Figure 4: Case study on ZH-EN where some dangling entities wrongly predicted as matchable by the previous first-order method can be correctly predicted as dangling with high probabilities via MHP. Arrows point from an entity to its NN in the other KG. The scores above arrows denote cosine similarities and those beside dangling ent. are probabilities of dangling by MHP.

Recently, Sun et al. (2021) have proposed a new problem setting, i.e., danging-aware entity alignment, which is more practical as dangling entities naturally exist in real-world KGs. This problem setting requests a model to both detect dangling entities and align matchable ones. As the pioneering work, Sun et al. (2021) propose three baseline methods (i.e., marginal ranking, background ranking, and nearest neighbor classification) based on the nearest neighbor of source entities. Thus, these methods only rely on the first-order proximity, which is the major difference with MHP.

Optimal transport. Optimal transport (OT) aims to find the plan with minimal transportation cost for changing one distribution to another distribution, which naturally provides a way to align two distributions. Arjovsky et al. (2017) use the Wasserstein distances to recast the learning of generative adversarial network (GAN) as a transportation problem. OT has been widely used in other applications like text generation (Chen et al., 2018a) and graph

Dangling entity	Cls. Prob.	The nearest target	Cosine Sim.	The nearest source	Cosine Sim.
哥伦比亚国际学院(Columbia International College)	0.95	University of Ottawa	0.67	渥太华大学(University of Ottawa)	0.80
丁肇中(Samuel C. C. Ting)	0.91	George Uhlenbeck	0.65	乔治·乌伦贝克(George Uhlenbeck)	0.78
美国国会地铁(Congressional Subway)	0.99	United States Congress	0.67	美国国会(United States Congress)	0.75
王豫元(Larry Wang)	1.00	Wu Den-yih	0.65	吴敦义(Wu Den-yih)	0.91
新生党(Japan Renewal Party)	1.00	Democratic Party of Japan	0.64	民主党(日本) (Democratic Party of Japan)	0.85
意大利裔澳洲人(Italian Australians)	0.99	Chinese Australians	0.64	澳大利亚华人(Chinese Australians)	0.71
新加坡发展部(Ministry of Development (Singapore))	0.93	Ministry of Transport (Singapore)	0.72	林瑞生(Lim Swee Say)	0.75

Table 5: Some dangling source entities wrongly predicted as matchable by the previous method, while MHP predicts them as dangling with high probabilities. Cls. Prob. denotes the probabilities of dangling generated by MHP. The fourth column denotes the cosine similarity between the dangling entity and its nearest target. The nearest source is the nearest neighbor of the nearest target on the source KG. The last column denotes the cosine cosine similarity between the nearest target and its nearest source.

matching (Xu et al., 2019a). Pei et al. (2019) formalize entity alignment as OT in the conventional setting, which however only considers one-to-one alignment between matchable entities. We instead leverage OT to identify dissimilar parts of embedding distributions to detect dangling entities, meanwhile using OT only as one of the three high-order measures for alignment.

6 Conclusion

In this paper, we propose a framework, MHP, with mixed high-order proximities for dangling-aware entity alignment. MHP captures the local high-order proximity via a dangling classifier based on both the first- and second-order proximities. Additionally, we propose a Optimal Transport based method considering the global high-order proximity to facilitate both dangling detection and entity alignment. Comprehensive experiments on two alignment settings show the effectiveness of utilizing mixed high-order proximities. Furthermore, our extensive ablation study demonstrates the effectiveness of each technique.

Acknowledgments

We appreciate the anonymous reviewers for their insightful comments and suggestions. Juncheng Liu and Xiaokui Xiao are supported by the Ministry of Education, Singapore (Grant No. MOE2018-T2-2-091). Muhao Chen is supported by the National Science Foundation of United States Grant IIS 2105329, and by the DARPA MCS program under Contract No. N660011924033 with the United States Office Of Naval Research. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

References

Martín Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 214–223.

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 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1452–1461.

Liqun Chen, Shuyang Dai, Chenyang Tao, Haichao Zhang, Zhe Gan, Dinghan Shen, Yizhe Zhang, Guoyin Wang, Ruiyi Zhang, and Lawrence Carin. 2018a. Adversarial text generation via feature-mover's distance. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 4671–4682.

Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena, and Carlo Zaniolo. 2018b. Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3998–4004.

Muhao Chen, Yingtao Tian, Mohan Yang, and Carlo Zaniolo. 2017. Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1511–1517.

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, Online. Association for Computational Linguistics.

Matthias Fey, Jan Eric Lenssen, Christopher Morris, Jonathan Masci, and Nils M. Kriege. 2020. Deep graph matching consensus. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 249–256.

Jacob Goldberger, Sam T. Roweis, Geoffrey E. Hinton, and Ruslan Salakhutdinov. 2004. Neighbourhood components analysis. pages 513–520.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. 1989. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

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

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*, 6(2):167–195.

Fangyu Liu, Muhao Chen, Dan Roth, and Nigel Collier. 2021. Visual pivoting for (unsupervised) entity alignment. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4257–4266.

Zhiyuan Liu, Yixin Cao, Liangming Pan, Juanzi Li, Zhiyuan Liu, and Tat-Seng Chua. 2020. Exploring and evaluating attributes, values, and structures for entity alignment. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6355–6364.

Shichao Pei, Lu Yu, and Xiangliang Zhang. 2019. Improving cross-lingual entity alignment via optimal transport. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3231–3237.

Gabriel Peyré, Marco Cuturi, et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607.

Milos Radovanovic, Alexandros Nanopoulos, and Mirjana Ivanovic. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(sept):2487–2531.

Zequn Sun, Muhao Chen, and Wei Hu. 2021. Knowing the no-match: Entity alignment with dangling cases. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL/IJCNLP)*, pages 3582–3593.

Zequn Sun, Muhao Chen, Wei Hu, Chengming Wang, Jian Dai, and Wei Zhang. 2020a. Knowledge association with hyperbolic knowledge graph embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5704–5716.

Zequn Sun, Wei Hu, and Chengkai Li. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *Proceedings of the 16th International Semantic Web Conference (ISWC)*, pages 628–644.

Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. 2018. Bootstrapping entity alignment with knowledge graph embedding. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4396–4402.

Zequn Sun, Chengming Wang, Wei Hu, Muhao Chen, Jian Dai, Wei Zhang, and Yuzhong Qu. 2020b. Knowledge graph alignment network with gated multi-hop neighborhood aggregation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 222–229.

Zequn Sun, Qingheng Zhang, Wei Hu, Chengming Wang, Muhao Chen, Farahnaz Akrami, and Chengkai Li. 2020c. A benchmarking study of embedding-based entity alignment for knowledge graphs. *Proceedings of the VLDB Endowment*, 13(11):2326–2340.

Xiaobin Tang, Jing Zhang, Bo Chen, Yang Yang, Hong Chen, and Cuiping Li. 2020. BERT-INT: A bert-based interaction model for knowledge graph alignment. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3174–3180.

Bayu Distiawan Trisedya, Jianzhong Qi, and Rui Zhang. 2019. Entity alignment between knowledge graphs using attribute embeddings. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pages 297–304.

Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. 2018. Cross-lingual knowledge graph alignment via graph convolutional networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 349–357.

Zhichun Wang, Jinjian Yang, and Xiaoju Ye. 2020. Knowledge graph alignment with entity-pair embedding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1672–1680.

Yuting Wu, Xiao Liu, Yansong Feng, Zheng Wang, Rui Yan, and Dongyan Zhao. 2019. Relation-aware entity alignment for heterogeneous knowledge graphs. In

Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI), pages 5278–5284.

Yuting Wu, Xiao Liu, Yansong Feng, Zheng Wang, and Dongyan Zhao. 2020. Neighborhood matching network for entity alignment. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6477–6487.

Hongteng Xu, Dixin Luo, Hongyuan Zha, and Lawrence Carin. 2019a. Gromov-wasserstein learning for graph matching and node embedding. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 6932–6941.

Kun Xu, Liwei Wang, Mo Yu, Yansong Feng, Yan Song, Zhiguo Wang, and Dong Yu. 2019b. Cross-lingual knowledge graph alignment via graph matching neural network. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3156–3161.

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 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4431–4441.

Kaisheng Zeng, Chengjiang Li, Lei Hou, Juanzi Li, and Ling Feng. 2021. A comprehensive survey of entity alignment for knowledge graphs. *AI Open*, 2:1–13.

Qingheng Zhang, Zequn Sun, Wei Hu, Muhao Chen, Lingbing Guo, and Yuzhong Qu. 2019. Multi-view knowledge graph embedding for entity alignment. In *Proceedings of the 28th International Joint Conference (IJCAI)*, pages 5429–5435.

Appendices

A Dataset Statistics

We present the dataset statistics of DBP2.0 (Sun et al., 2021) in Tab. 6. DBP2.0 contains three cross-lingual settings for dangling-aware entity alignment, i.e., Chinese-English (ZH-EN), Japanese-English (JA-EN) and French-English (FR-EN). Please note that FR-EN is much larger than ZH-EN and JA-EN, and our methods are scalable to such a large dataset.

Datasets		# Entities	# Danglings	# Rel.	# Triples	# Align.
ZH-EN	ZH	84,996	51,813	3,706	286,067	33,183
	EN	118,996	85,813	3,402	586,868	
JA-EN	JA	100,860	61,090	3,243	347,204	39,770
	EN	139,304	99,534	3,396	668,341	
FR-EN	FR	221,327	97,375	2,841	802,678	123,952
	EN	278,411	154,459	4,598	1,287,231	

Table 6: Dataset statistics of DBP2.0

B Computational Environment

We run experiments on a Linux machine with a single GeForce RTX 2080 Ti GPU with 11 GB GPU memory and a Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz. The operating system of our machine is Ubuntu 18.04.2 LTS. The major software packages used are as follows: TensorFlow 1.12; CUDA 10.1; Python 3.6; NumPy 1.18.1; SciPy 1.4.1. Our source code is available in the attachment for reproducible experiments.

C Hyperparameter Settings

To ensure a fair comparison, we follow the hyperparameter settings of the base alignment model (i.e., MTransE and AliNet) and the base dangling detection loss (i.e., MR and BR) reported in the previous work (Sun et al., 2021). For our proposed methods, in WGAN, we use a two-layer FNN with 500 hidden units for the critic. As suggested by Arjovsky et al. (2017), we adopt weight clipping to ensure K-Lipschitz for WGAN and train the critic more than the generator (i.e., the transformation matrix). Besides the hyperparameter stated in Section 4.1, we tune other hyperparameters within a search space as follows:

- The number of nearest targets k: {5, 10, 15}
- The number of nearest sources m: {5, 10, 15}
- Batch size: {4096, 8192, 10240, 20480}

D More on Experiments

As shown in Sun et al. (2021), AliNet (Sun et al., 2020b) performs much worse than MTransE (Chen et al., 2017) in dangling-aware entity alignment. Dangling entity detection would also suffer as a result of the poor alignment performance. However, in this section, we still present the results of MHP with AliNet as the base alignment model to demonstrate that MHP is model-agnostic and has a good robustness.

Consolidated evaluation. Tab. 7 shows that, using AliNet as the base model, MHP still outperforms baselines in terms of F1 scores on dangling detection. We can see that baselines sometimes achieve better precision with the sacrifice of recall, which leads to unsatisfactory F1 scores. Comparing our two variants MHP + MR and MHP + BR, there is no one consistently achieving better performance than the other one. We report the performance of

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
MR	.752	.538	.627	.828	.505	.627	.779	.580	.665	**.854**	.543	.664	.552	.570	.561	**.686**	.549	.609
BR	**.762**	.556	.643	.829	.515	.635	**.783**	.591	.673	.846	.546	.663	.547	.556	.552	.674	.556	.609
MHP + MR	.750	.711	.730	.838	**.726**	**.778**	.743	**.702**	**.722**	.831	**.714**	**.768**	.541	**.601**	**.571**	.638	**.661**	**.649**
MHP + BR	.748	**.718**	**.733**	**.841**	.721	.776	.738	**.702**	.719	.833	.711	.767	**.556**	.568	.562	.681	.590	.632

Table 7: Dangling entity detection results on DBP2.0. MR refers to marginal ranking and BR refers to the background ranking (Sun et al., 2021). The base alignment model is AliNet (Sun et al., 2020b).

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
MR	.207	**.299**	.245	.159	**.320**	.213	.231	**.321**	.269	.178	**.340**	.234	.195	**.190**	.193	.160	.200	.178
BR	.203	.286	.238	.155	.308	.207	.223	.306	.258	.170	.321	.222	.183	.181	.182	.164	.200	.180
MHP + MR	**.259**	.280	**.269**	.222	.298	**.254**	**.266**	.288	**.276**	**.225**	.305	**.259**	**.204**	.186	**.195**	**.197**	.189	**.193**
MHP + BR	.258	.274	.265	**.223**	.305	.257	.261	.281	.271	.224	.306	.258	.183	.180	.182	.172	**.201**	.185

Table 8: Two-step entity alignment results on DBP2.0. The base alignment model is AliNet.

Methods	ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
	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
AliNet	.332	.594	.421	.359	.629	.451	.338	.596	.429	.363	.630	.455	.223	.473	.306	.246	.495	.329
w/ MR	.343	.606	.433	.364	.637	.459	.349	.608	.438	.377	.646	.469	**.230**	.477	**.312**	.252	.502	**.335**
w/ BR	.333	.599	.426	.357	.632	.451	.341	.608	.431	.369	.636	.461	.214	.468	.298	.238	.487	.321
MHP + MR	**.346**	**.613**	**.439**	**.375**	**.645**	**.469**	**.354**	**.617**	**.444**	**.379**	**.654**	**.473**	.228	**.477**	.311	**.253**	.496	**.335**
MHP + BR	.339	.611	.432	.373	.635	.464	.346	.614	.437	.367	.638	.460	.218	.473	.303	.244	**.504**	.331

Table 9: Entity alignment results in the relaxed setting on DBP2.0. The base alignment model is AliNet.

two-step entity alignment on Tab. 8. In general, MHP offers better performance on two-step alignment compared with baselines that do not consider high-order proximities. We observe that when we choose AliNet as the base model, the improvement over the baselines is less than the improvement when using MTransE as the base model. The reason could be that AliNet generally performs worse than MTransE, even only with MR or BR. For example, combining Tab. 1 and 7, MTransE+MR can achieve 0.740 F1 score, while AliNet+MR only obtains 0.627 F1 score. The observation is also pointed out by Sun et al. (2021). The inherent inferiority of AliNet in dangling-aware entity alignment can hinder our new proposed techniques. Therefore, we suggest to use MTransE as the base alignment model for dangling-aware entity alignment. Future work could investigate other advanced alignment models on this setting.

Relaxed evaluation. Tab. 9 demonstrates the results of entity alignment in the relaxed setting. We observe that AliNet without any dangling detection technique performs the worst. By applying dangling detection techniques, the alignment performance increases, indicating that learning to detect dangling entities can indirectly help alignment. MHP with two different base dangling losses (i.e., MR and BR) generally outperforms the corresponding baselines without our proposed techniques. For our two variants, MHP + MR slightly outperforms MHP + BR variants in most cases.

E Computational Cost

Note that, similar with the MR loss (Sun et al., 2021), MHP also relies on nearest neighbor search (NNS) for training. Therefore, MHP can reuse the results of NNS obtained by MR during the training phase, and cause negligible additional overhead. On ZH-EN, MHP averagely spends around 60 seconds training an epoch. When the efficiency is of importance in some real-time applications, we can adopt the large-scale efficient similarity search library faiss (Johnson et al., 2021) which uses GPUs for fast NNS. Additionally, we could also maintain a cache unit to store the results of NNS and only lazily update the results every ten or twenty epochs during training.

F Limitations

We notice that many prior studies on conventional entity alignment consider the side information of entities (e.g., names, descriptions and attributes) (Chen et al., 2018b; Trisedya et al., 2019; Zhang et al., 2019; Xu et al., 2019b; Wang et al., 2020; Wu et al., 2020). However, on dangling-aware entity alignment, the pioneer work (Sun et al., 2021) proposes a framework that only considers the structure information of entities since most KGs are built around relation triples. Thus, for a fair comparison, we follow their setting and do not utilize side information of entities. Future work could investigate how to effectively incorporate side information for dangling-aware entity alignment in the proper way and with a fair evaluation.