Data Science and Engineering (2022) 7:16–29
https://doi.org/10.1007/s41019-022-00178-4

RESEARCH PAPERS

Toward Entity Alignment in the Open World: An Unsupervised Approach with Confidence Modeling

Xiang Zhao[1] · Weixin Zeng[1] · Jiuyang Tang[1] · Xinyi Li[1] · Minnan Luo[2] · Qinghua Zheng[2]

Received: 8 June 2021 / Revised: 18 October 2021 / Accepted: 13 January 2022 / Published online: 29 January 2022

Abstract
Entity alignment (EA) aims to discover the equivalent entities in different knowledge graphs (KGs). It is a pivotal step for integrating KGs to increase knowledge coverage and quality. Recent years have witnessed a rapid increase of EA frameworks. However, state-of-the-art solutions tend to rely on labeled data for model training. Additionally, they work under the closed-domain setting and cannot deal with entities that are unmatchable. To address these deficiencies, we offer an unsupervised framework UEA that performs entity alignment in the open world. Specifically, we first mine useful features from the side information of KGs. Then, we devise an unmatchable entity prediction module to filter out unmatchable entities and produce preliminary alignment results. These preliminary results are regarded as the pseudo-labeled data and forwarded to the progressive learning framework to generate structural representations, which are integrated with the side information to provide a more comprehensive view for alignment. Finally, the progressive learning framework gradually improves the quality of structural embeddings and enhances the alignment performance. Furthermore, noticing that the pseudo-labeled data are of various qualities, we introduce the concept of confidence to measure the probability of an entity pair of being true and develop a confidence-based unsupervised EA framework CUEA. Our solutions do not require labeled data and can effectively filter out unmatchable entities. Comprehensive experimental evaluations validate the superiority of our proposals .

Keywords Entity alignment · Unsupervised learning · Knowledge graph

1 Introduction

Knowledge graphs (KGs) have been applied to various fields such as natural language processing and information retrieval. To improve the quality of KGs, many efforts have been dedicated to the alignment of KGs, since different KGs usually contain complementary information. Particularly, entity alignment (EA), which aims to identify equivalent entities in different KGs, is a crucial step of KG alignment and has been intensively studied over the last few years [1–8]. We use Example 1 to illustrate this task.

Example 1 In Figure 1 are a partial English KG and a partial Spanish KG concerning the director `Hirokazu Koreeda`, where the dashed lines indicate known alignments (i.e., seeds). The task of EA aims to identify equivalent entity pairs between two KGs, e.g., (`Shoplifters`, `Manbiki Kazoku`).

State-of-the-art EA solutions [9–12] assume that equivalent entities usually possess similar neighboring information.

✉ Xiang Zhao
xiangzhao@nudt.edu.cn

Weixin Zeng
zengweixin13@nudt.edu.cn

Jiuyang Tang
jiuyang_tang@nudt.edu.cn

Xinyi Li
xinyili@nudt.edu.cn

Minnan Luo
minnluo@mail.xjtu.edu.cn

Qinghua Zheng
qhzheng@mail.xjtu.edu.cn

[1] Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha, China

[2] Department of Computer Science, Xi'an Jiaotong University, Xi'an, China

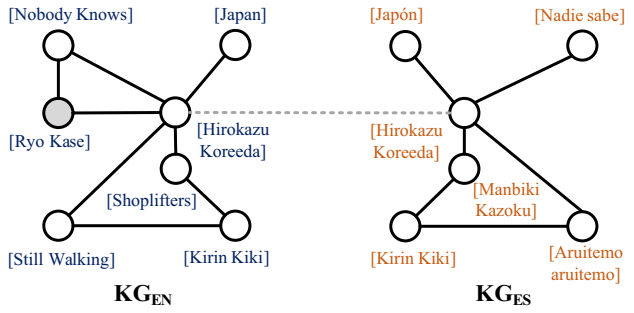

Fig. 1 An example of EA

Consequently, they utilize KG embedding models, e.g., TransE [13], or graph neural network (GNN) models, e.g., GCN [14], to generate structural embeddings of entities in individual KGs. Then, these separated embeddings are projected into a unified embedding space by using the seed entity pairs as connections, so that the entities from different KGs are directly comparable. Finally, to determine the alignment results, the majority of current works [3, 15–17] formalize the alignment process as a ranking problem; that is, for each entity in the source KG, they rank all the entities in the target KG according to some distance metric, and the closest entity is considered as the equivalent target entity.

Nevertheless, we still observe several issues from current EA works:

- *Reliance on labeled data* Most of the approaches rely on pre-aligned seed entity pairs to connect two KGs and use the unified KG structural embeddings to align entities. These labeled data, however, might not exist in real-life settings. For instance, in Example 1, the equivalence between `Hirokazu Koreeda` in KG_{EN} and `Hirokazu Koreeda` in KG_{ES} might not be known in advance. In this case, state-of-the-art methods that solely rely on the structural information would fall short, as there are no seeds to connect these individual KGs.
- *Closed-domain setting* All of current EA solutions work under the closed-domain setting [18]; that is, they assume every entity in the source KG has an equivalent entity in the target KG. Nevertheless, in practical settings, there always exist unmatchable entities. For instance, in Example 1, for the source entity `Ryo Kase`, there is no equivalent entity in the target KG. Therefore, an ideal EA system should be capable of predicting the unmatchable entities.

In response to these issues, we put forward an unsupervised EA solution UEA that is capable of addressing the unmatchable problem. Specifically, to mitigate the reliance on labeled data, we mine useful features from the KG side information and use them to produce preliminary pseudo-labeled data. These preliminary seeds are forwarded to our devised progressive learning framework to generate unified KG structural representations, which are integrated with the side information to provide a more comprehensive view for alignment. This framework also progressively augments the training data and improves the alignment results in a self-training fashion. Besides, to tackle the unmatchable issue, we design an unmatchable entity prediction module, which leverages thresholded bi-directional nearest neighbor search (TBNNS) to filter out the unmatchable entities and exclude them from the alignment results. We embed the unmatchable entity prediction module into the progressive learning framework to control the pace of progressive learning by dynamically adjusting the thresholds in TBNNS. Nevertheless, we discover that there is still a notable issue with UEA:

- *Ignorance of the quality of pseudo-labeled data.* UEA treats the pseudo-labeled data generated in the progressive learning process equally. Nevertheless, these pseudo-labeled data are generated with different degrees of confidence. That is, the framework could have a higher degree of *confidence* for believing some pseudo-labeled entity pairs to be correct, while for the others such *confidence* could be relatively low.

As thus, we introduce the concept of confidence to measure the probability of an entity pair of being correct. We further incorporate such confidence scores into KG representation learning with the aim of producing more accurate structural embeddings. Through empirical studies, we demonstrate that the confidence-based framework, CUEA, has a more stable performance than UEA regardless of the quality of input side information, and is particularly useful when the side information is low-grade.

This article is an extended version of our previous work [19]. In this extension, we make the following improvement:

- We extend UEA to a confidence-based framework CUEA, where we put forward C-TBNNS to assign confidence scores to aligned entity pairs and incorporate such probabilities into the KG representation learning process, so as to improve the quality of learned entity representations and also the alignment performance.
- We add more datasets for evaluation and conduct a more comprehensive analysis, which empirically validate that, compared with UEA, CUEA has a more consistent performance and is more effective given low-quality side information.

Organization In Section 2, we formally define the task of EA and introduce related work. In Section 3, we introduce

the preliminaries. In Section 4, we detail unmatchable entity prediction and the confidence-based extension. In Section 5, we introduce the progressive learning framework. In Section 6, we report the experimental results and conduct detailed analysis. In Section 7, we conclude this article.

2 Task Definition and Related Work

In this section, we formally define the task of EA, and then introduce the related work.

2.1 Task Definition

The inputs to EA are a source KG $\mathcal{G}_1$ and a target KG $\mathcal{G}_2$. The task of EA is defined as finding the equivalent entities between the KGs, i.e., $\Psi = \{(u, v)|u \in \mathcal{E}_1, v \in \mathcal{E}_2, u \leftrightarrow v\}$, where $\mathcal{E}_1$ and $\mathcal{E}_2$ refer to the entity sets in $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively, $u \leftrightarrow v$ represents the source entity u and the target entity v are *equivalent*, i.e., u and v refer to the same real-world object.

Most of current EA solutions assume that there exist a set of seed entity pairs $\Psi_s = \{(u_s, v_s)|u_s \in \mathcal{E}_1, v_s \in \mathcal{E}_2, u_s \leftrightarrow v_s\}$. Nevertheless, in this work, we focus on unsupervised EA and do not assume the availability of such labeled data.

2.2 Related Work

Entity alignment The majority of state-of-the-art methods are supervised or semi-supervised, which can be roughly divided into three categories, i.e., methods merely using the structural information, methods that utilize the iterative training strategy, and methods using information in addition to the structural information [20].

The approaches in the first category aim to mine useful structural signals for alignment, and devise structure learning models such as recurrent skipping networks [21] and multi-channel GNN [17], or exploit existing models such as TransE [3, 9, 22–24] and graph attention networks [3]. The embedding spaces of different KGs are connected by seed entity pairs. In accordance to the distance in the unified embedding space, the alignment results can hence be predicted.

Methods in the second category iteratively label likely EA pairs as the training set and gradually improve alignment results [15, 22–25]. A more detailed discussion about these methods and the difference from our framework is provided in Section 5. Methods in the third category incorporate the side information to offer a complementing view to the KG structure, including the attributes [10, 26–30], entity descriptions [16, 31], and entity names [12, 25, 32–35]. These methods devise various models to encode the side information and consider them as features parallel to the structural information. In comparison, the side information in this work has an additional role, i.e., generating pseudo-labeled data for learning unified structural representations.

Unsupervised entity alignment A few methods have investigated the alignment without labeled data. Qu et al. [36] propose an unsupervised approach toward knowledge graph alignment with the adversarial training framework. Nevertheless, the experimental results are extremely poor. He et al. [37] utilize the shared attributes between heterogeneous KGs to generate aligned entity pairs, which are used to detect more equivalent attributes. They perform entity alignment and attribute alignment alternately, leading to more high-quality aligned entity pairs, which are used to train a relation embedding model. Finally, they combine the alignment results generated by attribute and relation triples using a bivariate regression model. The overall procedure of this work might seem similar to our proposed model. However, there are many notable differences; for instance, the KG embeddings in our work are updated progressively, which can lead to more accurate alignment results, and our model can deal with unmatchable entities. We empirically demonstrate the superiority of our model in Sect. 6.

We notice that there are some entity resolution (ER) approaches established in a setting similar to EA, represented by PARIS [38]. They adopt collective alignment algorithms such as similarity propagation so as to model the relations among entities. We include them in the experimental study for the comprehensiveness of the article.

3 Preliminaries

In this section, we first introduce the outline of our proposal. Then, we elaborate the processing of side information to produce preliminary alignment seeds.

3.1 Model Outline

As shown in Fig. 2, given two KGs, CUEA first mines useful features from the *side information*. These features are forwarded to the *unmatchable entity prediction* module to generate initial alignment results with confidence scores, which are regarded as pseudo-labeled data. Then, the *progressive learning framework* uses these pseudo seeds, along with the probability scores, to connect two KGs and learn unified entity structural embeddings. It further combines the alignment signals from the side information and *structural information* to provide a more comprehensive

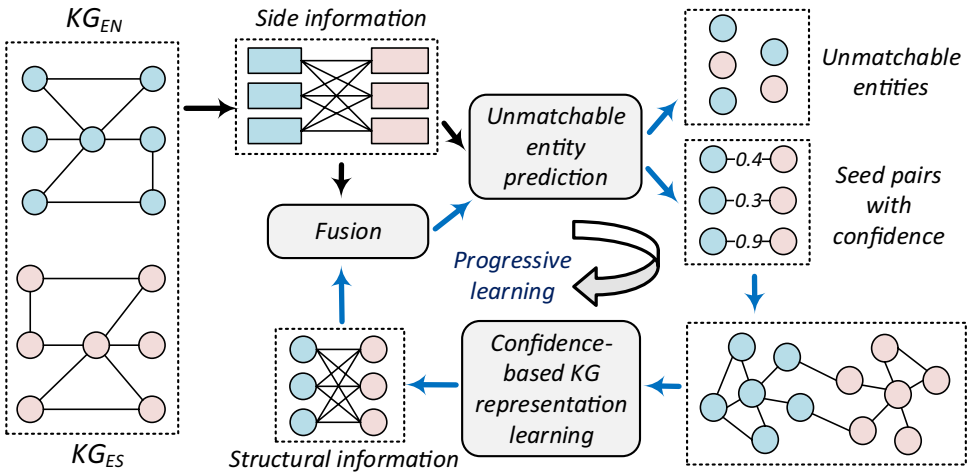

Fig. 2 Outline of CUEA. Arrows in blue represent the progressive learning process. By setting the confidence to 1, the UEA model [19] can be restored

view for alignment. Finally, it progressively improves the quality of structural embeddings and augments the alignment results by iteratively updating the pseudo-labeled data with results from the previous round, which also leads to increasingly better alignment. Note that by assigning the confidence score of 1 to all entity pairs, CUEA turns into the UEA model [19].

3.2 Side Information

There is abundant side information in KGs, such as the attributes, descriptions and classes. In this work, we use a particular form of the attributes—the entity name, as it exists in the majority of KGs. To make the most of the entity name information, inspired by [12], we exploit it from the semantic level and string-level and generate the textual distance matrix between entities in two KGs.

More specifically, we use the averaged word embeddings to represent the semantic meanings of entity names. Given the semantic embeddings of a source and a target entity, we obtain the semantic distance score by subtracting their cosine similarity score from 1. We denote the semantic distance matrix between the entities in two KGs as $\mathbf{M^n}$, where rows represent source entities, columns denote target entities, and each element in the matrix denotes the distance score between a pair of source and target entities. As for the string-level feature, we adopt the Levenshtein distance [39] to measure the difference between two sequences. We denote the string distance matrix as $\mathbf{M^l}$.

To obtain a more comprehensive view of alignment, we combine the two distance matrices and generate the textual distance matrix as $\mathbf{M^t} = \alpha\mathbf{M^n} + (1-\alpha)\mathbf{M^l}$, where α is a hyper-parameter balancing the weights. Then, we forward the textual distance matrix $\mathbf{M^t}$ to the unmatchable entity module to produce alignment results, which are considered as the pseudo-labeled data for training KG structural embeddings. The details are introduced in the next subsection.

Remark The goal of this step is to exploit available side information to generate useful features for alignment. Other types of side information, e.g., attributes and entity descriptions, can also be leveraged. Besides, more advanced textual encoders, such as misspelling oblivious word embeddings [40] and convolutional embedding for edit distance [41], can be utilized. We will investigate them in the future.

4 Unmatchable Entity Prediction

State-of-the-art EA solutions generate for each source entity a corresponding target entity and fail to consider the potential unmatchable issue. Nevertheless, as discussed in [20], in real-life settings, KGs contain entities that other KGs do not contain. For instance, when aligning YAGO 4 and IMDB, only 1% of entities in YAGO 4 are related to movies, while the other 99% of entities in YAGO 4 necessarily have no match in IMDB. These unmatchable entities would increase the difficulty of EA. Therefore, in this work, we devise an unmatchable entity prediction module to predict the unmatchable entities and filter them out from the alignment results.

4.1 Thresholded Bi-directional Nearest Neighbor Search

We put forward a novel strategy, i.e., thresholded bi-directional nearest neighbor search (TBNNS), to generate the alignment results, and the resulting unaligned entities are predicted to be unmatchable. Specifically, given a source entity u and a target entity v, if u and v are the nearest neighbor of each other, and the distance between them is below a given threshold θ, we consider (u, v) as an aligned entity

pair. Note that $\mathbf{M}(u, v)$ represents the element in the u-th row and v-th column of the distance matrix $\mathbf{M}$.

The TBNNS strategy exerts strong constraints on alignment, since it requires that the matched entities should both prefer each other the most, and the distance between their embeddings should be below a certain value. Therefore, it can effectively predict unmatchable entities and prevent them from being aligned. Notably, the threshold θ plays a significant role in this strategy. A larger threshold would lead to more matches, whereas it would also increase the risk of including erroneous matches or unmatchable entities. In contrast, a small threshold would only lead to a few aligned entity pairs, and almost all of them would be correct. This is further discussed and verified in Sect. 6.3. Therefore, our progressive learning framework dynamically adjusts the threshold value to produce more accurate alignment results (to be discussed in the next section).

Algorithm 1: Confidence-based TBNNS

Input : $\mathcal{G}_1$ and $\mathcal{G}_2$: two KGs to be aligned; $\mathcal{E}_1$ and $\mathcal{E}_2$: the entity sets; θ threshold; $\mathbf{M}$: a distance matrix.
Output : $\mathcal{U}$: unmatchable entities; $\mathcal{S}$: seed entity pairs with confidence scores.

foreach $u \in \mathcal{E}_1$ **do**
  $v \leftarrow \arg\min_{\hat{v} \in \mathcal{E}_2} \mathbf{M}(u, \hat{v})$;
  if $\arg\min_{\hat{u} \in \mathcal{E}_1} \mathbf{M}(v, \hat{u}) = u$ **and** $\mathbf{M}(u, v) < \theta$ **then**
    $v' \leftarrow \arg\min_{\hat{v} \in \mathcal{E}_2 - \{v\}} \mathbf{M}(u, \hat{v})$, $\Delta_1 = \mathbf{M}(u, v') - \mathbf{M}(u, v))$;
    $u' \leftarrow \arg\min_{\hat{u} \in \mathcal{E}_1 - \{u\}} \mathbf{M}(v, \hat{u})$, $\Delta_2 = \mathbf{M}(v, u') - \mathbf{M}(v, u))$;
    $\Theta(u, v) \leftarrow \Delta_1 + \Delta_2$;
    $\mathcal{S} \leftarrow \mathcal{S} \cup \{(u, v), \Theta(u, v)\}$;
  else
    $\mathcal{U} \leftarrow \mathcal{U} \cup \{u\}$
foreach $v \in \mathcal{E}_2$ **do**
  if $v \notin \mathcal{S}$ **then**
    $\mathcal{U} \leftarrow \mathcal{U} \cup \{v\}$
return $\mathcal{S}, \mathcal{U}$.

4.2 Confidence-based TBNNS

Considering that the aligned entity pairs generated by TBNNS are of different qualities (i.e., some are true while some are not), we further put forward confidence-based TBNNS, C-TBNNS, to measure the confidence of an entity pair (of being true). Specifically, we define the confidence score Θ of an entity pair (u, v) as:

$$\Theta(u, v) = \mathbf{M}(u, v') - \mathbf{M}(u, v) + \mathbf{M}(v, u') - \mathbf{M}(v, u), \tag{1}$$

where $\Delta_1 = \mathbf{M}(u, v') - \mathbf{M}(u, v)$ denotes the gap between the distance scores of the top-2 closest entities (i.e., v and v') to entity u, while $\Delta_2 = \mathbf{M}(v, u') - \mathbf{M}(v, u)$ denotes the gap between the distance scores of the top-2 closest entities (i.e., u and u') to entity v. This is based on the intuition that, for an entity pair (u, v), if the distance between them is the smallest from both sides, and there are larger margins between the distances of the top-2 candidates, it would be more confident to consider them as a correct entity pair. We further restrict the confidence scores to a certain range:

$$\Theta(\mathcal{S}) = (1 - \lambda)\frac{\Theta(\mathcal{S}) - \min\{\Theta(\mathcal{S})\}}{\max\{\Theta(\mathcal{S})\} - \min\{\Theta(\mathcal{S})\}} + \lambda \tag{2}$$

where $\Theta(\mathcal{S})$ represents the confidence scores of the entity pairs in $\mathcal{S}$. The core of Eq. (2) is the min-max normalization, which converts the confidence scores to $[0, 1]$. We add a hyper-parameter $\lambda \in [0, 1]$ to further restrict the range of the confidence scores to $[\lambda, 1]$. As thus, by setting λ to 1, all entity pairs would have the same confidence score of 1, and C-TBNNS can be restored to TBNNS. Hence, C-TBNNS can be regarded as a general case of TBNNS, which introduces the concept of confidence (probability) into the alignment result generation process.

5 The Progressive Learning Framework

To exploit the rich structural patterns in KGs that could provide useful signals for alignment, we design a progressive learning framework to combine structural and textual features for alignment and improve the quality of structural embeddings and alignment results in a self-training fashion.

5.1 Knowledge Graph Representation Learning

As mentioned above, we forward the textual distance matrix $\mathbf{M^t}$ generated by using the side information to the unmatchable entity prediction module to produce the preliminary alignment results, which are considered as pseudo-labeled data for learning unified KG embeddings. Concretely, following [26], we adopt GCN[1] to capture the neighboring information of entities. We leave out the implementation details since this is not the focus of this paper, which can be found in [26].

Alignment objective Since the representations of source and target KGs are learned individually, they need to be projected into a unified embedding space, where the entities across KGs could be compared directly. To this end, we use the semi-supervised loss function to enforce the distance between the embeddings of the entities in the labeled entity

[1] More advanced structural learning models, such as recurrent skipping networks [21], could also be used here. We will explore these alternative options in the future.

pairs to be small and meanwhile the negative samples (i.e., nonequivalent entity pairs) to be large. Formally:

$$\mathcal{L} = \sum_{(u,v)\in\mathcal{S}} \sum_{(u',v')\in\mathcal{S}'_{(u,v)}} [d(\mathbf{u},\mathbf{v}) + \gamma - d(\mathbf{u}',\mathbf{v}')]_+, \tag{3}$$

where $[\cdot]_+ = \max\{0, \cdot\}$, (u, v) is a labeled entity pair from the training data, and $\mathcal{S}'_{(u,v)}$ represents the set of negative entity pairs obtained by corrupting (u, v) using nearest neighbor sampling [3]. $\mathbf{u}$ and $\mathbf{v}$ represent the embeddings of source and target entities learned by GCN, respectively. $d(\cdot,\cdot)$ is the distance function that measures the distance between two embeddings. γ is a hyper-parameter separating positive samples from negative ones.

Confidence-based objective Considering that the pseudo labeled entity pairs have different confidences of being true, we incorporate such probabilities into the alignment objective to learn more accurate structural embeddings:

$$\mathcal{L}_c = \sum_{(u,v)\in\mathcal{S}} \sum_{(u',v')\in\mathcal{S}'_{(u,v)}} \Theta(u,v) * [d(\mathbf{u},\mathbf{v}) + \gamma - d(\mathbf{u}',\mathbf{v}')]_+, \tag{4}$$

where $\Theta(u, v)$ is the confidence score attached to each entity pair. As thus, the more confident entity pairs would play a more important role during the training process, while the less confident pseudo entity pairs would have a smaller effect on the training, such that the impact from the false positives could be mitigated. We will empirically demonstrate its effectiveness in Section 6.

Feature fusion Given the learned structural embedding matrix $\mathbf{Z}$, we calculate the structural distance score between a source and a target entity by subtracting the cosine similarity score between their embeddings from 1. We denote the resultant structural distance matrix as $\mathbf{M^s}$. Then, we combine the textual and structural information to generate more accurate signals for alignment: $\mathbf{M} = \beta\mathbf{M^t} + (1-\beta)\mathbf{M^s}$, where β is a hyper-parameter that balances the weights. The fused distance matrix $\mathbf{M}$ is used to generate more accurate matches.

5.2 The Progressive Learning Algorithm

The amount of training data has an impact on the quality of the unified KG embeddings, which in turn affects the alignment performance [10, 42]. As thus, we devise an algorithm (Algorithm 2) to progressively augment the pseudo training data, so as to improve the quality of KG embeddings and enhance the alignment performance. The algorithm starts with learning unified structural embeddings and generating the fused distance matrix $\mathbf{M}$ by using the preliminary pseudo-labeled data $\mathcal{S}_0$ (line 1-2). Then, the fused distance matrix is used to produce the new alignment results $\Delta\mathcal{S}$ using C-TBNNS (line 4). These newly generated entity pairs $\Delta\mathcal{S}$ are added to the alignment results, which are used for generating the fused distance matrix in the next round (line 6-7). The entities in $\mathcal{S}$ are removed from the entity sets (line 9-10). In order to progressively improve the quality of KG embeddings and detect more alignment results, we perform the aforementioned process recursively until the number of newly generated entity pairs is below a given threshold μ. Finally, we consider the entity pairs in $\mathcal{S}$ as the final alignment results Ψ.

Algorithm 2: Progressive learning.

Input : $\mathcal{G}_1$ and $\mathcal{G}_2$: KGs to be aligned; $\mathcal{E}_1$ and $\mathcal{E}_2$: the entity sets; $\mathbf{M^t}$: textual distance matrix; $\mathcal{S}_0$: preliminary labeled data; θ_0: the initial threshold.
Output : Ψ: Alignment results.

1 $\mathcal{S} \leftarrow \mathcal{S}_0$;
2 Use $\mathcal{S}$ to learn structural embeddings and generate $\mathbf{M}$;
3 $\theta \leftarrow \theta_0$;
4 $\Delta\mathcal{S}, \mathcal{U} \leftarrow$C-TBNNS $(\mathcal{G}_1, \mathcal{G}_2, \mathcal{E}_1, \mathcal{E}_2, \theta, \mathbf{M})$;
5 **while** $|\Delta\mathcal{S}| \geq \mu$ **do**
6 $\quad \mathcal{S} \leftarrow \mathcal{S} + \Delta\mathcal{S}$;
7 $\quad$ Use $\mathcal{S}$ to learn structural embeddings and generate $\mathbf{M}$;
8 $\quad \theta \leftarrow \theta + \eta$;
9 $\quad \mathcal{E}_1 \leftarrow \{e | e \in \mathcal{E}_1, e \notin \mathcal{S}\}$;
10 $\quad \mathcal{E}_2 \leftarrow \{e | e \in \mathcal{E}_2, e \notin \mathcal{S}\}$;
11 $\quad \Delta\mathcal{S}, \mathcal{U} \leftarrow$C-TBNNS $(\mathcal{G}_1, \mathcal{G}_2, \mathcal{E}_1, \mathcal{E}_2, \theta, \mathbf{M})$;
12 $\Psi \leftarrow \mathcal{S}$;
13 **return** Ψ.

Notably, in the learning process, once a pair of entities is considered as a match, the entities will be removed from the entity sets (line 9-10). This could gradually reduce the alignment search space and lower the difficulty for aligning the rest entities. Obviously, this strategy suffers from the error propagation issue, which, however, could be effectively mitigated by the progressive learning process that dynamically adjusts the threshold. We will verify the effectiveness of this setting in Sect. 6.2.4.

5.3 Dynamic Threshold Adjustment

It can be observed from Algorithm 2 that the matches generated by the unmatchable entity prediction module are not only part of the eventual alignment results, but also the pseudo training data for learning subsequent structural embeddings. Therefore, to enhance the overall alignment performance, the alignment results generated in each round should, ideally, have both large *quantity* and high *quality*. Unfortunately, these two goals cannot be achieved at the same time. This is because, as stated in Sect. 4, a larger threshold in TBNNS can generate more alignment results (large quantity), whereas some of them might be erroneous

Table 1 The statistics of the evaluation benchmarks

Dataset	KG pairs	#Triples	#Entities	#Labeled Ents	#Relations	#Test set
$\texttt{DBP15K}_{\texttt{ZH-EN}}$	DBpedia(Chinese)	70,414	19,388	15,000	1,701	14,888
	DBpedia(English)	95,142	19,572	15,000	1,323	10,500
$\texttt{DBP15K}_{\texttt{JA-EN}}$	DBpedia(Japanese)	77,214	19,814	15,000	1,299	15,314
	DBpedia(English)	93,484	19,780	15,000	1,153	10,500
$\texttt{DBP15K}_{\texttt{FR-EN}}$	DBpedia(French)	105,998	19,661	15,000	903	15,161
	DBpedia(English)	115,722	19,993	15,000	1,208	10,500
$\texttt{SRPRS}_{\texttt{EN-FR}}$	DBpedia(English)	36,508	15,000	15,000	221	10,500
	DBpedia(French)	33,532	15,000	15,000	177	10,500
$\texttt{SRPRS}_{\texttt{EN-DE}}$	DBpedia(English)	38,363	15,000	15,000	222	10,500
	DBpedia(German)	37,377	15,000	15,000	120	10,500

(low quality). These wrongly aligned entity pairs can cause the error propagation problem and result in more erroneous matches in the following rounds. In contrast, a smaller threshold leads to fewer alignment results (small quantity), while almost all of them are correct (high quality).

To address this issue, we aim to balance between the quantity and the quality of the matches generated in each round. An intuitive idea is to set the threshold to a moderate value. However, this fails to take into account the characteristics of the progressive learning process. That is, in the beginning, the quality of the matches should be prioritized, as these alignment results will have a long-term impact on the subsequent rounds. In comparison, in the later stages where most of the entities have been aligned, the quantity is more important, as we need to include more possible matches that might not have a small distance score. Hence, we set the initial threshold θ_0 to a very small value so as to reduce potential errors. Then, in the following rounds, we gradually increase the threshold by η, so that more possible matches could be detected. We will empirically validate the superiority of this strategy over the fixed weight in Sect. 6.2.4.

Noteworthily, our proposed confidence-based framework CUEA can further help mitigate the low-quality issue, as we calculate and assign a confidence score to each entity pair, where the wrongly-aligned entity pairs would presumably have lower confidence scores and thus exert smaller influence on the subsequent alignment process.

Remark As mentioned in the related work, there are some existing EA approaches that exploit the iterative learning (bootstrapping) strategy to improve EA performance. Particularly, BootEA calculates for each source entity the alignment likelihood to every target entity, and includes those with likelihood above a given threshold in a maximum likelihood matching process under the 1-to-1 mapping constraint, producing a solution containing EA pairs [23]. This strategy is also adopted by [15, 24]. Zhu et al. use a threshold to select the entity pairs with very close distances as the pseudo-labeled data [22]. DAT employs a bi-directional margin-based constraint to select the confident EA pairs as labels [25]. Our progressive learning strategy differs from these existing solutions in four aspects: (1) we exclude the entities in the confident EA pairs from the test sets; and (2) we use the dynamic threshold adjustment strategy to control the pace of learning process; and (3) our strategy can deal with unmatchable entities; and (4) we attach a confidence score to each selected entity pair, which can mitigate the negative influence of the false positives on the KG representation learning process as well as the alignment results. The superiority of our strategy is validated in Sect. 6.

6 Experiment

This section reports the experiment results with in-depth analysis. The source code is available at https://github.com/DexterZeng/UEA.

6.1 Experiment Settings

In this subsection, we first introduce the datasets, and then we detail the parameter settings. Next, we introduce the evaluation metrics and the baseline models used for comparison.

6.1.1 Datasets

Following existing works, we adopt the DBP15K dataset [10] for evaluation. This dataset consists of three multilingual KG pairs extracted from DBpedia. Each KG pair contains 15 thousand inter-language links as gold standards. The statistics can be found in Table 1. We note that state-of-the-art studies merely consider the labeled entities and divide them into training and testing sets. Nevertheless, as shown in Table 1, there exist unlabeled entities, e.g., 4,388 and 4,572 entities in the Chinese and English KG of $\texttt{DBP15K}_{\texttt{ZH-EN}}$, respectively. In this connection, we adapt the dataset by including the unmatchable entities. Specifically, for each KG

pair, we keep 30% of the labeled entity pairs as the training set (for training the supervised or semi-supervised methods). Then, to construct the test set, we include the rest of the entities in the first KG and the rest of the labeled entities in the second KG, so that the unlabeled entities in the first KG become unmatchable. The statistics of the test sets can be found in the *Test set* column in Table 1.

In addition, we also use the SRPRS dataset for evaluation. Concretely, we adopt the two cross-lingual datasets, $\text{SRPRS}_{\text{EN-FR}}$ and $\text{SRPRS}_{\text{EN-DE}}$, which are extracted from the multilingual KGs of DBpedia. Note that we do not use the mono-lingual KG pairs in SRPRS since using the side information can already achieve the ground-truth results [20]. There are no unmatchable entities in SRPRS.

6.1.2 Parameter Settings

For the *side information* module, we utilize the fastText embeddings [43] as word embeddings. To deal with cross-lingual KG pairs, following [33], we use Google translate to translate the entity names from one language to another, i.e., translating Chinese, Japanese and French to English. α is set to 0.5. For the *structural information learning*, we set β to 0.5. Following [26], we set γ in the alignment objectives to 3 and adopt Manhattan distance as $d(\cdot,\cdot)$. Regarding C-TBNNS, we set λ to 0.4. For *progressive learning*, we set the initial threshold θ_0 to 0.05, the incremental parameter η to 0.1, the termination threshold μ to 30. Note that if the threshold θ is over 0.45, we reset it to 0.45. These hyper-parameters are default values since there is no extra validation set for hyper-parameter tuning. We will conduct the parameter analysis in the experiment.

6.1.3 Evaluation Metrics

We use *precision* (P), *recall* (R), and *F1 score* as evaluation metrics. The *precision* is computed as the number of correct matches divided by the number of matches found by a method. The *recall* is computed as the number of correct matches found by a method divided by the number of gold matches. The *F1 score* is the harmonic mean between *precision* and *recall*.

6.1.4 Competitors

We select the most performant state-of-the-art solutions for comparison. Within the group that solely utilizes structural information, we compare with:

- BootEA [23], which employs the bootstrapping strategy to iteratively label likely entity alignment as training data for learning alignment-oriented KG embeddings;
- TransEdge [15], which proposes a novel edge-centric embedding model that contextualizes relation representations in terms of specific head-tail entity pairs;
- MRAEA [42], which models entity embeddings by attending over the node's incoming and outgoing neighbors and its connected relations' meta semantics;
- SSP [44], which jointly leverages the global KG structure and entity-specific relational triples for better entity alignment.
- RREA [45], which leverages relational reflection transformation to obtain relation specific embeddings for each entity and achieves effective entity alignment.

Among the methods incorporating other sources of information, we compare with:

- GCN-Align [26], which employs GCN to learn structural embeddings and attribute embeddings for alignment;
- HMAN [16], which harnesses the attributes and textual descriptions of entities to complement the structural information;
- HGCN [11], which jointly learns entity and relation representations for EA;
- RE-GCN [46], which exploits multiple structural graph convolution driven by triadic graph and primal graph to learn entity and relation embeddings;
- DAT [25], which proposes an EA framework with emphasis on long-tail entities;

We also include the unsupervised approaches IMUSE [37] and PARIS [38]. To make a fair comparison, we only use entity name labels as the side information.

6.2 Results

In this subsection, we first introduce the main alignment results. Then, we report the performance of unsupervised approaches given side information in low quality. Finally, we conduct the ablation study to provide insights into UEA.

6.2.1 Main Alignment Results

Table 2 reports the alignment results, which shows that state-of-the-art supervised or semi-supervised methods have rather low precision values. This is because these approaches cannot predict the unmatchable source entities and generate a target entity for each source entity (including the unmatchable ones). Particularly, methods incorporating additional information attain relatively better performance than the

Table 2 Alignment results

	$DBP15K_{ZH-EN}$			$DBP15K_{JA-EN}$			$DBP15K_{FR-EN}$			$SRPRS_{EN-FR}$			$SRPRS_{EN-DE}$		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BootEA	0.444	0.629	0.520	0.426	0.622	0.506	0.452	0.653	0.534	0.365	0.365	0.365	0.503	0.503	0.503
TransEdge	0.518	0.735	0.608	0.493	0.719	0.585	0.492	0.710	0.581	0.400	0.400	0.400	0.556	0.556	0.556
MRAEA	0.534	0.757	0.626	0.520	0.758	0.617	0.540	0.780	0.638	0.403	0.403	0.403	0.543	0.543	0.543
SSP	0.521	0.739	0.611	0.494	0.721	0.587	0.512	0.739	0.605	0.372	0.372	0.372	0.521	0.521	0.521
RREA	0.565	0.801	0.663	0.550	0.802	0.652	0.573	0.827	0.677	0.468	0.468	0.468	0.601	0.601	0.601
GCN-Align	0.291	0.413	0.342	0.274	0.399	0.325	0.258	0.373	0.305	0.296	0.758	0.426	0.428	0.428	0.428
HMAN	0.614	0.871	0.720	0.641	0.935	0.761	0.674	**0.973**	0.796	0.400	0.400	0.400	0.528	0.528	0.528
HGCN	0.508	0.720	0.596	0.525	0.766	0.623	0.618	0.892	0.730	0.670	0.670	0.670	0.763	0.763	0.763
RE-GCN [1]	0.518	0.735	0.608	0.548	0.799	0.650	0.646	0.933	0.764	-	-	-	-	-	-
DAT	0.556	0.788	0.652	0.573	0.835	0.679	0.639	0.922	0.755	0.758	0.758	0.758	0.876	0.876	0.876
CUEA-sup	**0.921**	**0.913**	**0.917**	**0.946**	**0.942**	**0.944**	**0.956**	0.953	**0.954**	**0.988**	**0.972**	**0.980**	**0.991**	**0.983**	**0.987**
IMUSE	0.608	0.862	0.713	0.625	0.911	0.741	0.618	0.892	0.730	0.905	0.905	0.905	0.916	0.916	0.916
PARIS	**0.976**	0.777	0.865	**0.981**	0.785	0.872	**0.972**	0.793	0.873	0.990	0.870	0.926	0.990	0.930	0.959
UEA	0.913	**0.902**	**0.907**	0.940	0.932	0.936	0.953	**0.950**	**0.951**	0.987	0.969	0.978	**0.988**	**0.976**	**0.982**
CUEA	0.912	0.901	0.906	0.943	**0.935**	**0.939**	0.953	0.949	**0.951**	**0.988**	**0.970**	**0.979**	**0.988**	0.975	0.981

The best alignment results are denoted in bold

[1] We omit the results of RE-GCN on $SRPRS_{EN-FR}$ and $SRPRS_{EN-DE}$ since they are not provided in the original paper, and our implementation cannot reproduce the reported performance.

[2] On $SRPRS_{EN-FR}$ and $SRPRS_{EN-DE}$, all of the entities are matchable, and the number of gold matches equals to the number of entities in a KG. Besides, for most methods, they generate matches for all the entities in a KG. Therefore, the number of matches produced by these methods is equal to the number of gold matches, and the values of precision, recall, and F1 score are equal

methods in the first group, demonstrating the benefit of leveraging such additional information.

Regarding the unsupervised methods, although IMUSE cannot deal with the unmatchable entities and achieves a low precision score, it outperforms most of the supervised or semi-supervised methods in terms of recall and F1 score. This indicates that, for the EA task, the KG side information is useful for mitigating the reliance on labeled data. In contrast to the methods discussed before, PARIS attains very high precision, since it only generates matches that it believes to be highly possible, which can effectively filter out the unmatchable entities. It also achieves the second best F1 score among all approaches, showcasing its effectiveness when the unmatchable entities are involved. Our proposals, UEA and CUEA, attain the best balance between precision and recall and obtain the best F1 scores, outperforming the second-best by a large margin, validating their effectiveness. Notably, although our proposed models do not require labeled data, they achieve even better performance than the most performant supervised methods. This could be attributed to the facts that (1) Our proposals are capable of dealing with unmatchable entities and hence achieve a good balance between precision and recall, while all the supervised approaches fail to identify the unmatchable entities and make alignment predictions for every source entity (including the unmatchable ones), thus attaining a low precision and in turn a low F1 score; (2) Most of the state-of-the-art supervised approaches merely perform the one-time alignment and cannot benefit from the progressive learning framework that utilizes the pseudo-labeled data for better training; (3) Some supervised approaches fail to make use of the side information that could provide useful signals for alignment. To verify the effectiveness of our proposed modules in the supervised setting, we allow CUEA to make use of labeled data, resulting in CUEA-sup. The results in Table 2 reflect that CUEA-sup attains much better performance than the state-of-the-art supervised approaches, as well as the unsupervised variant CUEA.

Furthermore, it can be seen that, by integrating the notion of confidence into UEA, CUEA achieves comparable results to UEA. At first sight, it seems that assigning confidence scores to entity pairs do not have a large influence on the representation learning and the alignment results, which, however, could be ascribed to the fact that the side information is too effective on these datasets (solely using the string information can achieves an F1 score of 0.814, to be shown in Table 5), and hence rendering the structural information (largely affected by the confidence scores) less contributive to the overall results. Next, we will show that the confidence-based framework would be much more useful on datasets with side information in low quality.

Table 3 Alignment results given low-grade side information

	$\text{DBP15K}_{\text{ZH-EN}}$			$\text{DBP15K}_{\text{JA-EN}}$		
	P	R	F1	P	R	F1
IMUSE	0.056	0.080	0.066	0.053	0.077	0.063
PARIS	**0.921**	0.066	0.123	**0.911**	0.060	0.113
UEA	0.654	0.088	0.155	0.617	0.084	0.148
CUEA	0.682	**0.093**	**0.164**	0.690	**0.090**	**0.159**

The best alignment results are denoted in bold

Table 4 Comparison of time costs (in seconds)

	DBP15K			SRPRS	
	ZH-EN	JA-EN	FR-EN	EN-FR	EN-DE
RREA	600	597	713	335	351
HMAN	5,489	5,404	5,611	4,423	4,371
IMUSE	67	70	80	52	53
PARIS	10	7	7	5	5
UEA	464	377	384	241	256
CUEA	434	355	370	272	319

target KGs are disparate. Hence, we aim to examine the effectiveness of these unsupervised approaches when the side information is in low quality and cannot provide many useful signals for alignment.

We report the results on $\text{DBP15K}_{\text{ZH-EN}}$ and $\text{DBP15K}_{\text{JA-EN}}$ in Table 3, where the direct comparison between entity name strings serves as the side information. It can be observed that the F1 scores of all methods are very low (compared with those in Table 2), revealing that the quality of side information does affect the overall alignment results. Besides, given the low-quality side information, our proposed models UEA and CUEA still outperform the baselines IMUSE and PARIS in terms of the F1 score,

Table 5 Ablation results

	ZH-EN			JA-EN			FR-EN		
	P	R	F1	P	R	F1	P	R	F1
UEA	0.913	0.902	**0.907**	0.940	0.932	**0.936**	0.953	0.950	**0.951**
w/o Unm	0.553	0.784	0.648	0.578	0.843	0.686	0.603	0.871	0.713
w/o Prg	0.942	0.674	0.786	0.966	0.764	0.853	0.972	0.804	0.880
w/o Adj	0.889	0.873	0.881	0.927	0.915	0.921	0.941	0.936	0.939
w/o Excl	**0.974**	0.799	0.878	0.982	0.862	0.918	0.985	0.887	0.933
MWGM	0.930	0.789	0.853	0.954	0.858	0.903	0.959	0.909	0.934
TH	0.743	**0.914**	0.820	0.795	**0.942**	0.862	0.807	**0.953**	0.874
DAT-I	**0.974**	0.805	0.881	**0.985**	0.866	0.922	**0.988**	0.875	0.928
UEA $-$ $\mathbf{M}^{\mathbf{l}}$	0.908	0.902	0.905	0.926	0.924	0.925	0.937	0.931	0.934
$\mathbf{M}^{\mathbf{l}}$	0.935	0.721	0.814	0.960	0.803	0.875	0.948	0.750	0.838
UEA$-\mathbf{M}^{\mathbf{n}}$	0.758	0.727	0.742	0.840	0.807	0.823	0.906	0.899	0.903
$\mathbf{M}^{\mathbf{n}}$	0.891	0.497	0.638	0.918	0.562	0.697	0.959	0.752	0.843

The best alignment results are denoted in bold

6.2.2 Results Using Low-Quality Side Information

We compare the unsupervised approaches under a practical scenario where the side information is in low-quality. Specifically, we assume that the pre-trained word embeddings as well as the machine translation tools are not available. Under this circumstance, to use the entity name information, a viable solution is to compare the name strings directly. However, the direct string comparison would be ineffective for cross-lingual datasets such as $\text{DBP15K}_{\text{ZH-EN}}$ and $\text{DBP15K}_{\text{JA-EN}}$, where the languages in the source and

demonstrating the effectiveness of the progressive learning framework and the unmatchable entity prediction module. Moreover, it is notable that CUEA achieves better results than UEA in terms of all metrics. This could be attributed to the confidence-based alignment results generation process, which could enable the entity pairs of higher confidence (higher probability of being correct, presumably) to have a larger impact on the representation learning and alignment process.

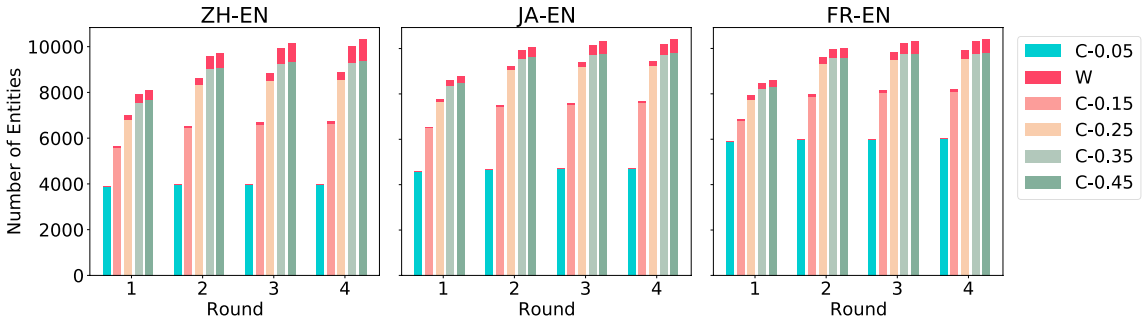

Fig. 3 Alignment results given different threshold values. C-θ refers to the number of correct matches generated by the progressive learning framework at each round given the threshold value θ. W refers to the number of erroneous matches generated in each round

6.2.3 Efficiency Comparison

In this subsection, we evaluate the alignment efficiency. In Table 4, we report the running time of unsupervised approaches, as well as two most performant supervised approaches. Corresponding alignment performance results can be found in Table 2. It reads from Table 4 that, generally speaking, the time costs of our proposals are acceptable, which mainly come from the progressive learning process. PARIS and IMUSE have high efficiency since they adopt simple models to capture the KG structure information and mainly rely on the existing side information for alignment, while the supervised models conduct complicated modeling of KG structure and hence require more time.

6.2.4 Ablation Study

In this subsection, we examine the usefulness of proposed modules by conducting the ablation study. First, by comparing the results of CUEA and UEA in Tables 2 and 3, we can conclude that the confidence-based framework is of great use, especially in cases when the side information is inferior. Next, we perform the ablation study on the basis of UEA.

More specifically, in Table 5, we report the results of UEA w/o Unm, which excludes the unmatchable entity prediction module, and UEA w/o Prg, which excludes the progressive learning process. It shows that, removing the unmatchable entity prediction module (UEA w/o Unm) brings down the performance on all metrics and datasets, validating its effectiveness of detecting the unmatchable entities and enhancing the overall alignment performance. Besides, without the progressive learning (UEA w/o Prg), the precision increases, while the recall and F1 score values drop significantly. This shows that the progressive learning framework can discover more correct aligned entity pairs and is crucial to the alignment progress.

To provide insights into the progressive learning framework, we report the results of UEA w/o Adj, which does not adjust the threshold, and UEA w/o Excl, which does not exclude the entities in the alignment results from the entity sets during the progressive learning. Table 5 shows that setting the threshold to a fixed value (UEA w/o Adj) leads to worse F1 results, verifying that the progressive learning process depends on the choice of the threshold and the quality of the alignment results. We will further discuss the setting of the threshold in the next subsection. Besides, the performance also decreases if we do not exclude the matched entities from the entity sets (UEA w/o Excl), validating that this strategy indeed can reduce the difficulty of aligning entities.

Moreover, we replace our progressive learning framework with other state-of-the-art iterative learning strategies (i.e., MWGM [23], TH [22] and DAT-I [25]) and report the results in Table 5. It shows that using our progressive learning framework (UEA) can attain the best F1 score, verifying its superiority.

6.3 Quantitative Analysis

In this subsection, we perform quantitative analysis of the modules in UEA and CUEA. We first investigate the unmatchable entity prediction module. Then, we examine the robustness of the progressive learning framework by varying the hyper-parameters. Finally, we provide the analysis on the side information, i.e., the influence of the quality of side information on the overall results, and the usefulness of the preliminary alignment results generated by the side information.

6.3.1 Analysis on Unmatchable Entity Prediction

Regarding the unmatchable entity prediction module, we aim to examine: (1) whether the unmatchable entities can be accurately detected; and (2) the influence of θ in TBNNS on the overall performance; and (3) the influence of λ in C-TBNNS on the overall performance.

Table 6 The influence of λ on the alignment results

	$\text{DBP15K}_{\text{ZH-EN}}$			$\text{DBP15K}_{\text{JA-EN}}$		
	P	*R*	*F*1	*P*	*R*	*F*1
UEA	0.682	**0.093**	**0.164**	**0.690**	**0.090**	**0.159**
$\lambda = 0.0$	0.685	**0.093**	**0.164**	0.680	**0.090**	**0.159**
$\lambda = 0.1$	0.687	0.092	0.162	0.683	**0.090**	**0.159**
$\lambda = 0.2$	**0.689**	**0.093**	0.163	0.680	**0.090**	**0.159**
$\lambda = 0.3$	0.678	0.092	0.162	0.679	0.089	0.158
$\lambda = 0.5$	0.661	0.091	0.160	0.672	**0.090**	**0.159**
$\lambda = 0.6$	0.670	0.090	0.159	0.648	0.088	0.155
$\lambda = 0.7$	0.666	0.089	0.158	0.631	0.085	0.150
$\lambda = 0.8$	0.647	0.088	0.156	0.640	0.088	0.155
$\lambda = 0.9$	0.649	0.088	0.155	0.586	0.081	0.142
$\lambda = 1.0$	0.654	0.088	0.155	0.617	0.084	0.148

The best alignment results are denoted in bold

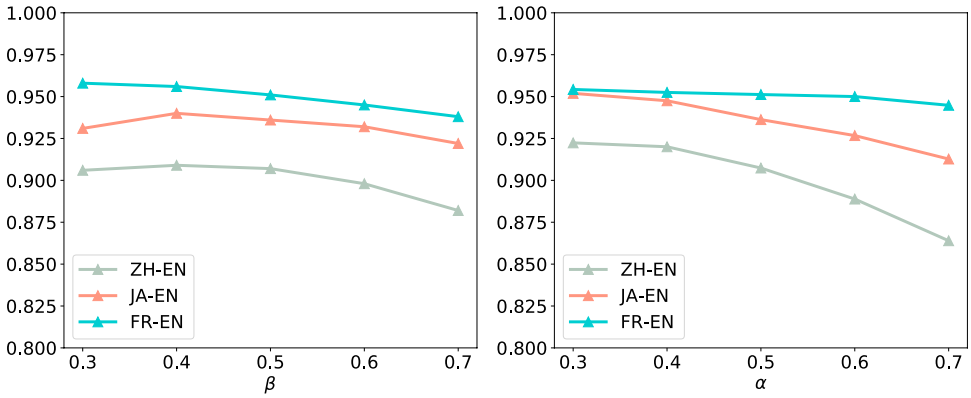

Fig. 4 The F1 scores by setting α and β to different values

Unmatchable entity prediction Zhao et al. [20] propose an intuitive strategy (U-TH) to predict the unmatchable entities. They set an NIL threshold, and if the distance value between a source entity and its closest target entity is above this threshold, they consider the source entity to be unmatchable. We compare our unmatchable entity prediction strategy with it in terms of the percentage of unmatchable entities that are included in the final alignment results and the F1 score. On $\text{DBP15K}_{\text{ZH-EN}}$, replacing our unmatchable entity prediction strategy with U-TH attains the F1 score at 0.837, which is 8.4% lower than that of UEA. Besides, in the alignment results generated by using U-TH, 18.9% are unmatchable entities, while this figure for UEA is merely 3.9%. This demonstrates the superiority of our unmatchable entity prediction strategy.

The threshold θ in TBNNS We discuss the setting of θ to reveal the trade-off between the risk and gain from generating the alignment results in the progressive learning. Identifying a match leads to the integration of additional structural information, which benefits the subsequent learning. However, for the same reason, the identification of a false positive, i.e., an incorrect match, potentially leads to mistakenly modifying the connections between KGs, with the risk of amplifying the error in successive rounds. As shown in Fig. 3, a smaller θ (e.g., 0.05) brings low risk and low gain; that is, it merely generates a small number of matches, among which almost all are correct. In contrast, a higher θ (e.g., 0.45) increases the risk, and brings relatively higher gain; that is, it results in much more aligned pairs, while a certain portion of them are erroneous. Additionally, using a higher threshold leads to increasingly more alignment results, while for a lower threshold, the progressive learning process barely increases the number of matches. This is in consistency with our theoretical analysis in Sect. 4.

The hyper-parameter λ in CUEA We then analyze the influence of λ in Eq. (2), which determines the range of the confidence scores, on the final alignment results. To highlight its influence on the structural representation learning, we follow the settings in 6.2.2 and report the results in Table 6.

It reads from Table 6 that the alignment performance is relatively stable when λ is not too large. Nevertheless, when setting λ to a large value (e.g., 1, to restore UEA), the results drop sharply. This reveals that assigning probability scores to the entity pairs according to their confidence of being true can facilitate the alignment. Besides, generally speaking, CUEA is robust to the perturbation of λ (as long as it is not too large).

6.3.2 Analysis on Progressive Learning Framework

Influence of hyper-parameters α and β As mentioned in Sect. 6.1, we set α and β to 0.5 since there are no training/validation data. Here, we aim to prove that different values of the parameters do not have a large influence on the final results. More specifically, we keep α at 0.5, and choose β from [0.3, 0.4, 0.5, 0.6, 0.7]; then we keep β at 0.5, and choose α from [0.3, 0.4, 0.5, 0.6, 0.7]. It can be observed from Fig. 4 that, although smaller α and β lead to better results, the performance does not change significantly.

6.3.3 Analysis on Side Information

We first analyze the influence of the side information on the final alignment results. Then, we examine the usefulness of preliminary alignment results generated by using the side information.

Influence of input side information We adopt different side information as input to examine the performance of UEA. More specifically, we report the results of UEA $-$ $\mathbf{M^l}$, which merely uses the string-level feature of entity names as input, UEA$-\mathbf{M^n}$, which only uses the semantic embeddings of entity names as input. We also provide the results of $\mathbf{M^l}$ and $\mathbf{M^n}$, which use the string-level and semantic information to directly generate alignment results (without progressive learning), respectively.

As shown in Table 3, the performance of solely using the input side information is not very promising ($\mathbf{M^l}$ and $\mathbf{M^n}$). Nevertheless, by forwarding the side information into our model, the results of UEA $-$ $\mathbf{M^l}$ and UEA$-\mathbf{M^n}$ become much better. This unveils that UEA can work with different types of side information and consistently improve the alignment results. Additionally, by comparing UEA $-$ $\mathbf{M^l}$ with UEA$-\mathbf{M^n}$, it is evident that the input side information does affect the final results, and the quality of the side information is of significance to the overall alignment performance.

Pseudo-labeled data We further examine the usefulness of the preliminary alignment results generated by the side information, i.e., the pseudo-labeled data. Concretely, we replace the training data in HGCN with these pseudo-labeled data, resulting in HGCN-U, and then compare its alignment results with the original performance. Regarding the F1 score, HGCN-U is 4% lower than HGCN on $\texttt{DBP15K}_{\texttt{ZH-EN}}$, 2.9% lower on $\texttt{DBP15K}_{\texttt{JA-EN}}$, 2.8% lower on $\texttt{DBP15K}_{\texttt{FR-EN}}$. The minor difference validates the effectiveness of the pseudo-labeled data generated by the side information. It also demonstrates that this strategy can be applied to other supervised or semi-supervised frameworks to reduce their reliance on labeled data.

7 Conclusion

In this article, we propose unsupervised EA solutions that are capable of dealing with unmatchable entities. We first exploit the side information of KGs to generate preliminary alignment results, which are considered as pseudo-labeled data and forwarded to the progressive learning framework to produce better KG embeddings and alignment results in a self-training fashion. We also devise an unmatchable entity prediction module to detect the unmatchable entities. The experimental results validate the usefulness of our proposed models and their superiority over state-of-the-art approaches.

Acknowledgements This work was partially supported by NSFC under grants Nos. 61872446, 62002373, 71971212, and the Science and Technology Innovation Program of Hunan Province under grant No. 2020RC4046.

References

1. Hao Y, Zhang Y, He S, Liu K, Zhao J (2016) A joint embedding method for entity alignment of knowledge bases. In: CCKS, pp. 3–14
2. Shi X, Xiao Y (2019) Modeling multi-mapping relations for precise cross-lingual entity alignment. In: EMNLP, pp. 813–822
3. Li C, Cao Y, Hou L, Shi J, Li J, Chua T-S (2019) Semi-supervised entity alignment via joint knowledge embedding model and cross-graph model. In: EMNLP, pp. 2723–2732
4. Sun Z, Wang C, Hu W, Chen M, Dai J, Zhang W, Qu Y (2020) Knowledge graph alignment network with gated multi-hop neighborhood aggregation. In: AAAI, pp. 222–229
5. Xu K, Song L, Feng Y, Song Y, Yu D (2020) Coordinated reasoning for cross-lingual knowledge graph alignment. In: AAAI, pp. 9354–9361

6. Chen J, Gu B, Li Z, Zhao P, Liu A, Zhao L (2020) SAEA: self-attentive heterogeneous sequence learning model for entity alignment. In: DASFAA, pp. 452–467
7. Wu Y, Liu X, Feng Y, Wang Z, Zhao D (2020) Neighborhood matching network for entity alignment. In: ACL, pp. 6477–6487
8. Sun Z, Zhang Q, Wei H, Wang C, Chen M, Akrami F, Li C (2020) A benchmarking study of embedding-based entity alignment for knowledge graphs. Proc. VLDB Endow. 13(11):2326–2340
9. Chen M, Tian Y, Yang M, Zaniolo C (2017) Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In: IJCAI, pp. 1511–1517
10. Sun Z, Hu W, Li C (2017) Cross-lingual entity alignment via joint attribute-preserving embedding. In: ISWC, pp. 628–644
11. Wu Y, Liu X, Feng Y, Wang Z, Zhao D (2019) Jointly learning entity and relation representations for entity alignment. In: EMNLP, pp. 240–249
12. Zeng W, Zhao X, Tang J, Lin X (2020) Collective entity alignment via adaptive features. In: ICDE, pp. 1870–1873
13. Bordes A, Usunier N, García-Durán A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: NIPS, pp. 2787–2795
14. Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. In: CoRR, arXiv:abs/1609.02907
15. Sun Z, Huang JC, Hu W, Chen M, Guo L, Qu Y (2019) Transedge: translating relation-contextualized embeddings for knowledge graphs. In: ISWC, pp. 612–629
16. Yang H-W, Zou Y, Shi P, Lu W, Lin J, Sun X (2019) Aligning cross-lingual entities with multi-aspect information. In: EMNLP, pp. 4430–4440
17. Cao Y, Liu Z, Li C, Liu Z, Li J, Chua T-S (2019) Multi-channel graph neural network for entity alignment. In: ACL, pp. 1452–1461
18. Hertling S, Paulheim H (2020) The knowledge graph track at OAEI - gold standards, baselines, and the golden hammer bias. In: ESWC 12123: 343–359
19. Zeng W, Zhao X, Tang J, Li X, Luo M, Zheng Q (2021) Towards entity alignment in the open world: An unsupervised approach. In: DASFAA, Springer, pp. 272–289.
20. Zhao X, Zeng W, Tang J, Wang W, Suchanek F (2020) An experimental study of state-of-the-art entity alignment approaches. IEEE Trans Knowl Data Eng pp. 1. https://ieeexplore.ieee.org/document/9174835
21. Guo L, Sun Z, Hu W (2019) Learning to exploit long-term relational dependencies in knowledge graphs. In: ICML, pp. 2505–2514
22. Zhu H, Xie R, Liu Z, Sun M (2017) Iterative entity alignment via joint knowledge embeddings. In: IJCAI, pp. 4258–4264
23. Sun Z, Hu W, Zhang Q, Qu Y (2018) Bootstrapping entity alignment with knowledge graph embedding. In: IJCAI, pp. 4396–4402
24. Zhu Q, Zhou X, Wu J, Tan J, Guo L (2019) Neighborhood-aware attentional representation for multilingual knowledge graphs. In: IJCAI, pp. 1943–1949
25. Zeng W, Zhao X, Wang W, Tang J, Tan Z (2020) Degree-aware alignment for entities in tail. In: SIGIR, pp. 811–820
26. Wang Z, Lv Q, Lan X, Zhang Y (2018) Cross-lingual knowledge graph alignment via graph convolutional networks. In: EMNLP, pp. 349–357
27. Trisedya BD, Qi J, Zhang R (2019) Entity alignment between knowledge graphs using attribute embeddings. In: AAAI, pp. 297–304
28. Yang K, Liu S, Zhao J, Wang Y, Xie B (2020) COTSAE: co-training of structure and attribute embeddings for entity alignment. In: AAAI, pp. 3025–3032
29. Chen B, Zhang J, Tang X, Chen H, Li C (2020) Jarka: modeling attribute interactions for cross-lingual knowledge alignment. In: PAKDD 12084: 845–856
30. Tang X, Zhang J, Chen B, Yang Y, Chen H, Li C (2020) BERT-INT: a bert-based interaction model for knowledge graph alignment. In: IJCAI, pp. 3174–3180
31. Chen M, Tian Y, Chang K-W, Skiena S, Zaniolo C (2018) Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. In: IJCAI, pp. 3998–4004
32. Xu K, Wang L, Yu M, Feng Y, Song Y, Wang Z, Yu D (2019) Cross-lingual knowledge graph alignment via graph matching neural network. In: ACL, pp. 3156–3161
33. Wu Y, Liu X, Feng Y, Wang Z, Yan R, Zhao D (2019) Relation-aware entity alignment for heterogeneous knowledge graphs. In: IJCAI, pp. 5278–5284
34. Fey M, Lenssen JE, Morris C, Masci J, Kriege NM (2020) Deep graph matching consensus. In: ICLR
35. Zeng W, Zhao X, Tang J, Lin X, Groth P (2021) Reinforcement learning-based collective entity alignment with adaptive features. ACM Trans Inf Syst 39(3):1–31
36. Qu M, Tang J, Bengio Y (2019) Weakly-supervised knowledge graph alignment with adversarial learning. In: CoRR, arXiv:abs/1907.03179
37. He Fuzhen, Li Zhixu, Yang Qiang, Liu An, Liu Guanfeng, Zhao Pengpeng, Zhao Lei, Zhang Min, Chen Zhigang (2019) Unsupervised entity alignment using attribute triples and relation triples. In *DASFAA*, pages 367–382
38. Suchanek FM, Abiteboul S, Senellart P (2011) PARIS: probabilistic alignment of relations, instances, and schema. In: PVLDB, 5(3):157–168
39. Levenshtein VI (1966) Binary codes capable of correcting deletions, insertions, and reversals. In: Soviet physics doklady, 10: 707–710. https://www.semanticscholar.org/paper/Binary-codes-capable-of-correcting-deletions%2C-and-Levenshtein/b2f8876482c97e804bb50a5e2433881ae31d0cdd
40. Edizel B, Piktus A, Bojanowski P, Ferreira R, Grave E, Silvestri F (2019) Misspelling oblivious word embeddings. In: NAACL-HLT, pp. 3226–3234
41. Dai X, Yan X, Zhou K, Wang Y, Yang H, Cheng J (2020) Convolutional embedding for edit distance. In: SIGIR, pp. 599–608
42. Mao X, Wang W, Xu H, Lan M, Wu Y (2020) MRAEA: an efficient and robust entity alignment approach for cross-lingual knowledge graph. In: WSDM, pp. 420–428
43. Bojanowski P, Grave E, Joulin A, Mikolov Tomas (2017) Enriching word vectors with subword information. Trans Assoc Comput Linguist 5:135–146
44. Nie H, Han X, Sun L, Wong CM, Chen Q, Wu S, Zhang W (2020) Global structure and local semantics-preserved embeddings for entity alignment. In: IJCAI, pp. 3658–3664
45. Mao X, Wang W, Xu H, Wu Y, Lan M (2020) Relational reflection entity alignment. In: CIKM, pp. 1095–1104
46. Yang J, Zhou W, Wei L, Lin J, Han J, Hu S (2020) RE-GCN: relation enhanced graph convolutional network for entity alignment in heterogeneous knowledge graphs. In: DASFAA, pp. 432–447