Artificial Intelligence Review (2024) 57:229
https://doi.org/10.1007/s10462-024-10866-4

A survey: knowledge graph entity alignment research based on graph embedding

Beibei Zhu[1] · Ruolin Wang[2,3] · Junyi Wang[2,3] · Fei Shao[2,3] · Kerun Wang[2,3]

Abstract

Entity alignment (EA) aims to automatically match entities in different knowledge graphs, which is beneficial to the development of knowledge-driven applications. Representation learning has powerful feature capture capability and it is widely used in the field of natural language processing. Compared with traditional EA methods, EA methods based on representation learning have better performance and efficiency. Hence, we summarize and analyze the representative EA approaches based on representation learning in this paper. We present the problem description and data preprocessing for EA and other related fundamental knowledge. We propose a new EA framework for the latest models, which includes information aggregation module, entity alignment module, and post-alignment module. Based on these three modules, the various technologies are described in detail. In the experimental part, we first explore the effect of EA direction on model performance. Then, we classify the models into different categories in terms of alignment inference strategy, noise filtering strategy, and whether additional information is utilized. To ensure fairness, we perform the comparative analysis of the performance of the models within the categories separately on different datasets. We investigate both unimodal and multimodal EA. Finally, we present future research perspectives based on the shortcomings of existing EA methods.

Keywords Entity alignment · Knowledge graph · Natural language processing · Representation learning

1 Introduction

Google proposed knowledge graphs to improve the accuracy of search engines and the efficiency of user retrieval (Liu et al. 2016). Knowledge graphs can help mine the semantic information of user needs and eliminate ambiguities. The semantic network (Bernerslee et al. 2001) is the predecessor of the knowledge graph. The semantic network focuses more on describing the relationship between concepts and concepts, while the knowledge graph is more inclined to describe the association between entities. The emergence of knowledge graphs is more in line with the development trend of computer semantics.

Extended author information available on the last page of the article

Published online: 03 August 2024

In recent years, knowledge graphs have been widely used in many fields. The data sources used to build knowledge graphs can be structured data, semi-structured data, unstructured data, generic knowledge graphs, etc. While different organizations choose data sources according to their business needs. In addition, there is no unified industry standard for the methods of building knowledge graphs in different domains. These all lead to heterogeneity and redundancy problems among different knowledge graphs. For example, the introduction of "Childhood" by Interactive Encyclopedia and Wikipedia is complementary and repetitive. If the information of the two encyclopedias can be correlated, users will have a more detailed and comprehensive knowledge of the book. To make full use of the information of entities, more and more researchers are fusing different knowledge graphs (Lin et al. 2020).

As a way to integrate knowledge (Mishra et al. 2017), EA extracts entities that refer to the same real-world objects in different knowledge graphs (Yang et al. 2020), which is beneficial for knowledge-driven applications. Traditional EA methods rely on machine translation or feature engineering, which are labor-intensive. And hand-designed features contain subjective factors, so the accuracy of traditional methods depends heavily on the quality of translation and the definition of features. Recently, representation learning techniques are proven to capture structural information better. Therefore, more and more researchers adopt representation learning techniques for knowledge graph EA. The embedding-based EA method frees from the reliance on manually constructed features or rules.

Several works (Zhao et al. 2022; Zhang et al. 2021; Sun et al. 2020; Fanourakis et al. 2022; Chaurasiya et al. 2022) have been done to review the development of EA. However, the field of EA is evolving rapidly and the existing review papers do not include the latest EA models. In addition, their work is not presented in sufficient detail to facilitate the reader's understanding. Zhao et al. (2022) divide the EA framework into four parts: embedding learning module, alignment module, prediction module, and additional information module. They divide the existing state-of-the-art methods into three groups and perform group evaluations to compare the experimental results of the same models in different datasets. Representative methods from each module are selected to generate possible combinations, and the effectiveness of different methods in these modules is understood by comparing the performance of different combinations. Most EA in their experiments are based on local alignment, and they do not introduce multi-modal knowledge graph EA and Chinese knowledge graph EA. Zhang et al. (2021) conduct a comprehensive survey and analysis for embedding-based knowledge graph EA, and they divide the knowledge graph EA framework into two processes: embedding and alignment. Embedding history and methods based on TransE and graph convolutional networks are presented, and nearly 30 representative structural embedding models using these two embedding approaches are listed. Zhang et al. analyze the embedding models by mining whether EA models add attribute information, whether they use relational predicates as input, and whether they add seed. The classification of information is not refined enough. Sun et al. (2020) investigate 23 embedding-based EA methods. They classify the 23 methods according to the techniques and features and construct an open-source library. The library contains 12 representative embedding-based EA methods and evaluation methods for these techniques. However, their focus is on the dataset construction and experimental results, and no models are introduced according to the classification of techniques, and their technical discussions cannot meet the needs of subsequent new models. Fanourakis et al. (2022) do not provide a comprehensive introduction to embedding-based EA tasks, for example, path sequence models in embedding methods, multimodal EA, dangling EA and alignment inference strategies have not been presented. Chaurasiya et al. (2022) focus on four aspects of Degree distribution, Non-isomorphic neighbourhood, and Name bias, and the details of the various parts of the process are not presented.

This paper presents comprehensive research in this field to fill the gap in existing reviews, with the following main contributions.

(1) This paper proposes a new EA framework, which is divided into three parts: information aggregation module, alignment module, and post-alignment module. Each module has unique functions. In the information aggregation module, this paper not only introduces different embedding initialization methods, but also further refines the subsequent parts into global structure embedding and local semantic information. Compared to existing reviews, which tend to treat relations simply as structures, this paper not only considers the structural aspects of relations on the macro level, but also captures the local semantic information of relations on the micro level. In addition, this paper details the interaction between global structure and local semantics, revealing their complementarities and collaborations in the entity alignment process. In the alignment module, this paper introduces alignment optimization strategies and non-alignable entity prediction methods, which are rarely mentioned in previous reviews. Moreover, this paper also comprehensively analyzes different alignment inference strategies from both global and local perspectives. In the post-alignment module, this paper compares and analyzes a variety of iterative strategies to provide guidance for practical applications.

(2) In the experimental part, this paper introduces the performance of unimodal EA and multimodal EA. Among them, when introducing unimodal EA, Chinese EA is also introduced. Considering that entity alignment is a bi-directional matching problem, this paper introduces the effect of EA direction on the model performance. By comparing the experimental results, it is found that the direction has an effect on the model performance, which provides a reference for researchers to optimize the model. In the section on comparative analysis of unimodal experiments, this paper classifies the representative models in a different way from the existing reviews, and also compares them in several aspects. This paper classifies the representative models from four aspects: whether to apply global alignment, whether to apply noise filtering strategy, utilizing only global structure, and combining global structure and local semantics. The existing methods are also compared and analyzed.

(3) This paper follows the latest research trends in the current field and details the advanced methods used for knowledge graph entity alignment. This paper not only introduces existing methods, but also proposes a series of innovative research ideas to researchers. In particular, the paper suggests combining other features such as video with textual information to further achieve more accurate multimodal entity alignment. In addition, to improve the robustness and applicability of entity alignment techniques, the paper emphasizes the importance of constructing datasets that are close to real-world situations from multiple dimensions. The paper also proposes that the knowledge graph can be mapped to more complex vector spaces (e.g., complex spaces) to obtain better quality entity embedding representations. Meanwhile, this paper proposes that spatial and temporal dimensions should be considered comprehensively, to cope with the dynamic changes of the knowledge graph and enhance the generalization ability of the model. This paper provides a reference for advancing the research progress in the field of knowledge graph entity alignment, as well as solving the challenges in real-world problems.

2 Preliminary

2.1 Knowledge graph EA problem description

A knowledge graph is the knowledge base that organizes data from a semantic perspective, and it is a more general framework for describing formal semantic knowledge. A knowledge graph can be formalized as $KG = (E, R, T)$, where E, R, and T represent entities, relations, and triples, respectively. A knowledge graph is a graph structure in which the nodes in the graph represent entities and the edges in the graph represent relationships. There are two types of triples. The first type is the relational triple, such as (Yuan_Longping, Birthplace, China). The other type is called the attribute triple, e.g. (Yuan_Longping, Gender, "Male"). The task of knowledge graph EA is to find equivalent entities in two knowledge graphs (Sun et al. 2020), which is defined formally as:

$$Align_{\text{entity}}\left(KG_1, KG_2\right) = \left\{\left(e_1, e_2\right) \mid e_1 \in KG_1, e_2 \in KG_2, e_1 \sim e_2\right\}, \tag{1}$$

where KG_1, KG_2 denotes two knowledge graphs, e_1, e_2 denotes entities, and $\sim$ denotes equivalence relations. Usually, the subset $Align'_{\text{entity}}\left(KG_1, KG_2\right) \subset Align_{\text{entity}}\left(KG_1, KG_2\right)$ is called the seed set, which is known in advance and used as training data.

2.2 Data preprocessing

Data preprocessing performs operations before the main processing of the data to obtain more targeted data and improve the subsequent alignment process. Data preprocessing in the EA task can be divided into syntactic regularization and data regularization. Usually, most EA algorithms, such as Zhu et al. (2021) perform alignment directly after simply organizing the data format and removing noisy data. While some other EA algorithms employ special data preprocessing. For example, Trisedya et al. (2019) first align predicates and then name similar predicates uniformly to embed relations and entities into the same vector space. Chen et al. (2020c) use radial basis functions to specialize continuous values. To mine the hidden information in the knowledge graph, Jiang et al. (2019) use logic rules to derive new triples in order to enrich the number of triples. And rule-based methods are generally divided into deductive reasoning and transfer rules. RpAlign (Huang et al. 2022) expands the training data using data augmentation to produce supervised triples across the knowledge graph, which can exchange information from different knowledge graphs.

3 Related foundations

3.1 Translation model

The translation model uses vector translation of the embedding space to represent relationships. TransE (Bordes et al. 2013) is the representative of the translation model family and it is widely used. Based on the vector representation of entities and relationships, TransE treats the relationships of a triple as the translation from the head entity to the tail entity. The purpose is to embed all entities and relations in the knowledge graph into a low-dimensional vector space. The energy function of the relation triple $\left(e_1, r_1, e_2\right)$ is defined as:

$$\varphi\left(e_1, r_1, e_2\right) = \left\| e_1 + r_1 - e_2 \right\|, \tag{2}$$

where $\|\cdot\|$ denotes the L_1-parameter or L_2-parameter of the vector. TransE has become the baseline standard for vectorized representation of knowledge graphs and has derived many different variants, such as TransR (Lin et al. 2015a), TransC (Lv et al. 2018) and KG2E (He et al. 2015), etc.

3.2 Deep model

Deep model uses deep learning techniques to learn embedding representations, and commonly used deep models include graph neural network (GNN), graph convolutional network (GCN), and graph attention network (GAT).

GNN (Zhou et al. 2020) is based on an information propagation mechanism, where each node updates its state by exchanging information with each other until it reaches some stable value. The goal of GNN is to learn the state embedding of each neighbor node, and the final output can be obtained using h_v. The formula for h_v is as follows:

$$h_v = f\left(X_v, X_{co[v]}, h_{ne[v]}, X_{ne[v]}\right), \tag{3}$$

where $f(\cdot)$ is a local transformation function with parameters to update the current node state according to the state adjustment of neighboring nodes; X_v denotes the feature vector of node v; $X_{co[v]}$ denotes the feature vector of the edge where node v is located; $h_{ne[v]}$ denotes the state vector of node v's neighboring node; $x_{ne[v]}$ denotes the feature vector of the neighboring nodes of node v.

GCN (Kipf and Welling 2017) includes input, layer propagation and output. When the knowledge graph is embedded into a low-dimensional vector space, entities are considered as nodes. GCN uses the activation function to continuously update the neighbor node information, which is formulated as follows.

$$H^{(l+1)} = \sigma\left(D^{-\frac{1}{2}} \widehat{A} D^{-\frac{1}{2}} H^{(l)} W^{(l)}\right), \tag{4}$$

where $\widehat{A} = A + I$, A is the adjacency matrix, I is the identity matrix; D is the degree matrix of A; H is the feature of a layer, which is equivalent to X at the input layer and is called the hidden layer during propagation; W is the weight matrix; σ is the activation function.

GAT (Velickovic et al. 2018) introduces an attention mechanism to assign corresponding weights to neighboring nodes and obtains information about the whole network from local information. The attention coefficients a_{ij} are obtained by normalizing.

$$a_{ij} = \frac{\exp\left(\mathrm{LeakyRe}lu\left(\mathrm{e}_{ij}\right)\right)}{\sum_{k \in N_i} \exp\left(\mathrm{LeakyRe}lu\left(\mathrm{e}_{ik}\right)\right)}. \tag{5}$$

Using the computed attention coefficients, the features are weighted and summed to obtain the new features incorporating the neighborhood information.

$$h_i' = \sigma\left(\sum_{j\in N_i} a_{ij} W\overrightarrow{h_j}\right). \tag{6}$$

Generally, GAT also uses multi-headed attention to enhance the ability of the model and stabilize the training process by concatenating or averaging.

3.3 Semantic matching model

The semantic matching model calculates the similarity from the semantic level based on entities and relations in the vector space. Semantic matching models include RESCAL (Nickel et al. 2011), DistMult (Yang et al. 2015) and MLP (Multi-Layer Perceptron, Dong et al. 2014), etc.

RESCAL associates entities and vectors to obtain semantic information. The relationships are represented as matrices to model the pairwise interactions between potential factors with the following score functions.

$$f_r(h,t) = \mathbf{h}^\top \mathbf{M}_r \mathbf{t} = \sum_{i=0}^{d-1}\sum_{j=0}^{d-1} \left[\mathbf{M}_r\right]_{ij} \cdot [\mathbf{h}]_i \cdot [\mathbf{t}]_j, \tag{7}$$

where $\mathbf{h}, \mathbf{t} \in \mathbb{R}^d$ is the vector representation of the entity and $\mathbf{M}_r \in \mathbb{R}^{d\times d}$ is the matrix associated with the relationship.

DistMult simplifies RESCAL by restricting $\mathbf{M}_r$ to the diagonal matrix. For each relation r, it introduces a vector embedding $\mathbf{r} \in \mathbb{R}^d$ and requires that $\mathbf{M}_r = \operatorname{diag}(\mathbf{r})$.

$$f_r(h,t) = \mathbf{h}^\top \operatorname{diag}(\mathbf{r})\mathbf{t} = \sum_{i=0}^{d-1} [\mathbf{r}]_i \cdot [\mathbf{h}]_i \cdot [\mathbf{t}]_i. \tag{8}$$

MLP is relatively simple, where each relation (and entity) is associated with a vector and concatenates the vectors h, r, and t in the input layer and maps to the nonlinear hidden layer. The scoring function is as follows.

$$f_r(h,t) = \mathbf{w}^\top \tanh\left(\mathbf{M}^1\mathbf{h} + \mathbf{M}^2\mathbf{r} + \mathbf{M}^3\mathbf{t}\right), \tag{9}$$

where $\mathbf{M}^1, \mathbf{M}^2, \mathbf{M}^3 \in \mathbb{R}^{d\times d}$ are the first level weights and $\mathbf{W} \in \mathbb{R}^d$ are the second level weights, which are all shared in different relations.

3.4 Random walk

Random walk (RW) learns node embeddings by generating sequences. If nodes appear simultaneously on a random walk path in the graph, it means they have similar embeddings. When sampling paths for the knowledge graph, the generated sequences are cross-combinations of nodes and relations (Chen et al. 2020e). DeepWalk (Perozzi et al. 2014) and node2vec (Grover and Leskovec 2016) are pioneering works that introduce deep learning techniques into network analysis to learn node embeddings. When node2vec is applied to the knowledge graph, the transfer probability of reaching the next entity is calculated as follows.

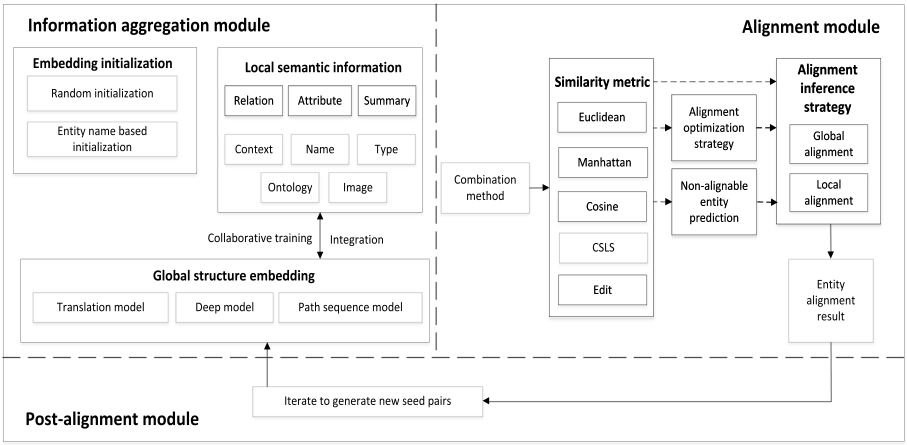

Fig. 1 A basic framework for EA based on representation learning

$$P(e_{i+1} \mid e_i) = \begin{cases} \alpha_{pq}(e_i, e_{i+1}) \cdot w & \exists r \in R, (e_i, r, e_{i+1}) \in T, \\ 0 & \text{otherwise} , \end{cases} \tag{10}$$

where e_i is the ith entity in a certain walk, which needs to decide the next entity e_{i+1}, and if there is a relationship r between e_i and e_{i+1}, the transfer probability from e_i to e_{i+1} needs to be evaluated, and w is the edge weight between entities e_i and e_{i+1}.

3.5 Multimodal knowledge graph

Knowledge graph techniques have been widely used to deal with structured and textual data, but relatively less attention has been paid to unstructured data such as images. Few effective technical means are designed to extract structured knowledge from image data. Therefore, multimodal knowledge graphs are proposed for constructing entities under multiple modalities (e.g., image modality). Multimodal knowledge graphs can provide sufficient visual information for entities, thus allowing EA on a larger scale. Ultimately, multimodal knowledge graphs enable existing models to perform better because text and image features are considered together.

Although multimodal structural data are heterogeneous in the underlying representation, different modal data of the same entity are unified in the high-level semantics. Therefore, fusing multimodal data is helpful to language representation models. There are few studies on multimodal knowledge graphs, and several important open-source multimodal knowledge graphs include IMGpedia (Ferrada et al. 2017), MMKG (Liu et al. 2019), and Richpedia (Wang et al. 2019).

4 Representation learning-based EA framework

As is shown in Fig. 1, we design a typical representation learning-based knowledge graph EA framework. The framework includes information aggregation module, alignment module, and post-alignment module. When aligning entities, two knowledge graphs are first input and seed data are collected for training. Because the quality of the original data directly affects the final alignment results, the input data is often preprocessed.

In the information aggregation module, the embedding representation needs to be initialized firstly, which is generally done by random initialization or entity name-based initialization. Based on the initial embedding representation, the global structure embedding part updates entity embedding, which is based on translation family model, deep model, and path sequence model. The topological connection of the knowledge graph only provides the global structure information, while the local semantic information of entities, such as relations, attributes, summaries, contexts, names, types, ontologies, and images, has a positive impact on EA. So many methods fuse local semantic information to improve the alignment. The effect of the model can be enhanced by iterative co-training between global structural embedding and local semantic information, or integrate global and local information to enrich the features of entities. The information aggregation module gets the final entity embedding, and the information aggregation module provides a reserve for the entity alignment module.

In the alignment module, the embeddings of the source and target knowledge graphs are first unified into a vector space by using the combination method. Then the distance between the source and target entity vectors is calculated based on the final embedding representation of the entities. The commonly used metric strategies include Euclidean distance, Manhattan distance, Cosine distance, Cross-Domain Similarity Local Scaling (CSLS), and Edit distance. After the distance metric calculation between entity vectors, the entity similarity matrix is obtained. Some studies design optimization strategies and non-alignable entity prediction to improve the accuracy of alignment. After that, the alignment strategy includes global alignment and local alignment. Finally, the alignment module will output the alignment result.

In the post-alignment module, semi-supervised strategies are used to iteratively generate new seed pairs and expand the size of the seed set.

5 Information aggregation module

5.1 Embedding initialization method

5.1.1 Random initialization

Most current EA methods, such as GCN-Align (Wang et al. 2018), BootEA (Sun et al. 2018) and COTSAE (Yang et al. 2020) need to depend on the structure to form the initialization vector of entities. The goal of graph embedding is to obtain the low-dimensional vector representation of a high-dimensional graph. The structure embedding needs to be initialized before the global structure embedding is performed. Random initialization of entity embeddings is easier and more convenient. But random initialization may lead to local optima and produce low-quality embeddings. In real-world knowledge graphs, most entities have low node degrees and little structural information, so using only structural information to initialize entity embeddings may limit the effectiveness of EA models.

5.1.2 Vector initialization based on entity names

The entity name is considered a special attribute that is independent of the node degree. The entity name is an important clue in determining whether two entities are equivalent. If the entity name is available in KG, the entity name vector can be used as the initial feature vector of the entity. GMNN (Xu et al. 2019) uses a word-based LSTM to convert the entity name to its initial feature vector. To better initialize the model, RDGCN (Wu et al. 2019a), UED (Luo and Yu 2022), RNM (Zhu et al. 2021), EAMI (Zhu et al. 2023) and RAGA (Zhu et al. 2021) use Glove based on entity names for model initialization. They translate non-English entity names into English via Google Translate and initialize entity features with pre-trained English word vectors.

5.2 Global structure embedding method

In recent years, the field of EA structure embedding methods are mainly divided into three categories: translation model-based structural embedding methods, deep model-based structural embedding methods, and path sequence-based structural embedding methods. The characteristics of the three embedding methods are compared in Table 1.

5.2.1 Structural embedding based on translation model

Early EA approaches based on representation learning relied on TransE to capture the structural information of the knowledge graph. They directly use TransE for structural embedding, modeling in-graph relationships and treating relationships as translation vectors between entities, such as COTSAE (Yang et al. 2020), MEEA (Chen et al. 2021b), DAEA (Sun et al. 2020), MMEA (Chen et al. 2020c), and JSAE (Munne and Ichise 2020), etc.

Some approaches improve TransE, such as Trisedya et al. (2019) add weight to TransE, which enables aligned triples to receive higher attention and improves the alignment effect. The contribution of different neighbors to EA varies, NAEA (Zhu et al. 2019) uses a neighborhood-aware attention mechanism on top of TransE to aggregate entity neighbors with different importance to obtain a neighborhood-level entity representation. RTEA (Jiang et al. 2019) uses string similarity-based and embedding-based approaches represented by TransE to refine structure embedding. Based on RTEA, ESEA (Jiang et al. 2022a) uses an embedding-based model to filter weakly correlated entities and then explores the final alignment using a symbol-based approach. Based on TransE, AMKE (Shen et al. 2022) sets different margin hyperparameters for different relations and adapts the learning margin parameter.

The variants of TransE are also used to embed structures. TransE only represents the one-hop relationships between entities, ignoring the important multi-hop relationship information, and the modeling of complex relationships is not good enough. So IPTransE (Zhu et al. 2017) uses PTransE (Lin et al. 2015b) for structural embedding. Ps-TransC (Kang et al. 2020) uses TransC for structure embedding. Ps-TransC divides the knowledge graph into an ontology layer and an instance layer, where entities in the ontology layer are

Table 1 Comparative analysis of global structure embedding methods

Category		Technology	Advantage	Disadvantage	Work
Based on translation model	Apply TransE directly	Use TransE to treat relations as translations between entities	The model is simple and easy to explain the model structure (2) TransE uses the same embedding space for both relationship and entity embedding	(1) Not powerful enough, only implicitly capturing information about neighbors (2) Loss of representation model and loss of alignment model are not easily balanced	Yang et al. (2020), Yan et al. (2020), Sun et al. (2020), and Chen et al. (2020c)
	Improvement to TransE	Add weight to TransE			Munne and Ichise (2020), Trisedya et al. (2019), and Zhu et al. (2019)
		Structure embedding using TransC			Kang et al. (2020)
		Use PTransE to represent multi-hop relationships			Zhu et al. (2017)
		Combining embedding methods and traditional methods			Jiang et al. (2022a)
		Adaptive margin parameter			Shen et al. (2022)
Based on deep model	Using GNN	Stacking GNN layers to aggregate neighbor information	Capable of fusing the entity's neighbor information into the entity embedding	(1) Complexity of the model (2) Unable to model the long-term structural dependencies between entities	Sun et al. (2020) Pei et al. (2020)
	Using GCN	Direct structure embedding using GCN			Xiong and Gao (2019), Yang et al. (2019), Ye et al. (2019), Wu et al. (2020), and Nie et al. (2020)
		Consider the effect of noise on the model			Wu et al. (2019a, 2019b) and Zhu et al. (2021)
		Solving the over-smoothing problem			Zeng et al. (2021)
	Using GAT	Attention-based propagation mechanism to collect important neighborhood information			Xie et al. (2020, 2022) and Sun et al. (2020)
Based on path sequence model	Random walk	Random walk method using degree awareness	(1) Leverage long-term structural dependencies (2) High expressiveness and efficient propagation	(1) Complex model (2)The path cannot be sampled according to the characteristics	Chen et al. (2020e)
	Improved random walk	Path sampling method using biased random walk			Guo et al. (2019) and Chen et al. (2020f)

considered as a class. TransC models all the triples of each class as a sphere and all the instances of that class are contained within the sphere.

5.2.2 Structural embedding based on deep model

Although the information of neighbors can be transmitted to the central entity, they are only captured implicitly. Therefore to fuse the information of neighbors into the entity embedding, the structural embedding method based on the GNN family of models is proposed. This method considers complex parameters and relations, so it can learn more expressive embeddings (Yan et al. 2020). The deep method usually stacks more than two layers of GNN to learn entity representations, where the first layer of GNN nodes is randomly initialized and the node representation of the last layer of GNN is the final representation of entities. Representative models for structure embedding based on deep models include REA (Pei et al. 2020) and HyperKA (Sun et al. 2020).

GCN follows the neighborhood aggregation scheme that can iteratively update the representation of each entity node. Representative works include RNM (Zhu et al. 2021), NMN (Wu et al. 2020); and the work by Xiong and Gao 2019). The number of layers for GCNs has an impact on EA, HMAN (Yang et al. 2019) stacks multiple layers of GCNs to collect multi-hop neighbor information. Directly using multi-layer GCNs to aggregate information leads to the propagation of noisy information, so AliNet (Sun et al. 2020) uses the gating mechanism to aggregate multi-hop neighbor information. SSP (Nie et al. 2020) and HGCN (Wu et al. 2019b) use GCN to explicitly encode structure information. Highway gates are also used to control the amount of neighborhood information passed to nodes. Tam et al. (2021) adjust the number of GCNs layers to prevent the information transfer noise in the previous layer as well as the topology loss. To address the over-smoothing problem caused by the increasing number of GCN layers, RAC (Zeng et al. 2021) uses both approximate personalized propagation of neural predictions and GCN models to capture structure information. In addition to solving the over-smoothing problem caused by the increase in the number of GCN layers, EchoEA (Lin et al. 2021) also proposes a solution for the overfitting problem by introducing a four-level (entity-level, feature-level, entity-to-relationship, and relationship-to-entity) attention mechanism to further encode entity features.

GAT is also used to learn global structural embeddings of knowledge graphs. To effectively utilize pre-aligned links in the knowledge graph, CAECGAT (Xie et al. 2020) and DuGa-DIT (Xie et al. 2022) share cross knowledge graph entity embeddings and updates the embeddings using a gate mechanism. By overlaying multiple attention layers, the model can learn multi-hop information. In addition, TTEA (Zhang et al. 2023) uses GAT in the last part of the model to re-aggregate the information of the neighbors.

5.2.3 Structural embedding based on path sequence model

Translation models and deep models do not fully exploit the long-term structural dependencies among entities, which suffer from the limitations of low expressiveness and inefficient information dissemination. To better explore the structural information among entities, SAEA (Chen et al. 2020e) designs a degree-aware random walk method to generate heterogeneous sequence data and capture the long-term structural dependencies within entities. Deep paths have more relational dependencies than triples, and cross-knowledge mapping paths are used as bridges between knowledge graphs to transfer information.

RSNs (Guo et al. 2019); and the work by (Chen et al. 2020f) apply biased random walk path sampling method to effectively explore the deep and cross-KG relational paths for embedding learning.

5.3 Combination of global structure and local semantic

5.3.1 Collaborative training

In the EA framework, different modules can be trained collaboratively and there is a positive influence between modules. The first category is the co-training of relationship alignment and entity alignment. For example, RNM (Zhu et al. 2021) adds the relationship information between entities to the neighbor matching model. And RNM designs a semi-supervised framework so that entity alignment and relationship alignment can enhance each other. HGCN (Wu et al. 2019b) first uses the entity embeddings learned by GCN to approximate the relationship representation. The relational representation is then merged into the entities to iteratively learn better representations.

Some models iteratively perform attribute alignment and entity alignment. For example, IMUSE (He et al. 2019), COTSAE (Yang et al. 2020), and NovEA (Sun et al. 2020). In each iteration, IMUSE first performs EA based on attribute values to build a matching set of entity pairs and then performs attribute alignment to build a matching set of attribute pairs. COTSAE learns entity embeddings using a collaborative training framework with alternating TransE components and pseudo networks. NovEA assumes that all common attributes of the two entities have the same weight and uses the aligned entities for attribute alignment. But when two entities do not have common attributes, aligned entity pairs can be used to find more possible aligned attribute pairs.

The generation and discrimination modules of the EA method applying adversarial learning are trained collaboratively. REA (Pei et al. 2020) first trains the noise-aware module to update the entity embeddings, and then uses the learned embeddings to optimize the noise-detection module. The trust score provided by the noise detection module can be fed back in the next iteration to train the noise-aware entity alignment. SEA (Pei et al. 2019b) uses the adversarial training model to iteratively refine the knowledge graph embedding to improve the perception of entity degree differences. The iterative training process stops considering the effect of degree on embeddings when the discriminator cannot distinguish entities based on degree information.

The knowledge graph complementation module and entity alignment module can also iterate over each other. ALIGNKGC (Singh et al. 2021) uses ComplEx for initialization of the knowledge graph complementation task to define the triple scores, which ensures that the two entities share the same embedding vector. Entity alignment allows the knowledge graph to obtain more facts, and high confidence predictions can facilitate EA.

5.3.2 Integration

Global structure and local semantics complement each other. Usually, combining global structure and local information provides additional help to obtain a better entity representation. The integration of multiple knowledge representations using vector concatenation enhances the complementarity of different information, thus improving the accuracy of EA tasks. For example, the unimodal EA model GCN-Align (Wang et al. 2018) concatenates entity embedding and attribute embedding according to the weights. FuzzyEA (Jiang et al.

2022b) performs the fusion of structural embeddings and local semantic embeddings, based on Dempster's combination rule. The multimodal EA model MMEA (Chen et al. 2020c) migrates multimodal knowledge relational data, visual data, and numerical data embeddings from a separate space to a common space and sets the proportional hyperparameters for each type of knowledge.

In addition, some methods fuse global and local information from the matrix level. For example, CEA (Zeng et al. 2020) and CUEA (Zhao et al. 2022) first computes the global similarity matrix based on embedding and then compute the local name semantic similarity matrix, and then add the weighted sum of these two categories to fuse the global structure and local semantic information.

5.4 Local semantic information

The local semantic information incorporated into entities can complement the structural embedding of the knowledge graph and benefit EA. Analyzed from the perspective of the existing forms of local semantic information, the local semantic information of entities mainly includes relations, attributes, entity summary/description, contexts, entity names, and images. Tables 2 and 3 show the advantages and disadvantages of unimodal and multimodal local semantic information, respectively.

5.4.1 Incorporating relational semantic

To accurately disambiguate entities with similar structures, relational semantics can be used to refine the structure-based representation so that similar entities can be distinguished. The relations in the relational triples are connected by head and tail entities, so the relational embedding can be approximated by connecting its average head and tail entity representations, and representative models include HGCN (Wu et al. 2019b), RDGCN (Wu et al. 2019a), RNM (Zhu et al. 2021), and AVR-GCN (Ye et al. 2019), etc. RREA (Mao et al. 2020a) uses multilayer neural networks to learn entity embeddings. For different relationship types, the same entity is embedded in different relationship spaces, and then the embeddings of the same entity in different relationship spaces are aggregated into one entity embedding. The diversity of relationship structures poses challenges to relationship representation. Therefore, SREA (Zhang et al. 2024) constructs weighted line graphs to model diverse relational structures and learns relational representations independently from entities.

Structural embedding models based on TransE or its variants are unsuitable for encoding multi-mapping relationships. For example, a movie made by a director has multiple actors. For this reason, Shi and Xiao (2019) calibrate the embedding of different KGs by a set of fewer pre-aligned seeds to encode multi-map relations by dot product scaling. Contextualized relational representation improves on the above approach by arguing that relations occurring in different entity contexts should have different embeddings, regardless of whether they have the same surface form. For example, SSP (Nie et al. 2020) computes relational embeddings based on adjacent entities and the relations themselves. The approach is more intuitive and the SSP captures the semantic differences between relations even if they have the same surface form but occur in different contexts. RpAlign (Huang et al. 2022) treats relationships as rotational operations between entities and can handle

Table 2 Comparative analysis of various unimodal local semantic information

Category		Technology	Advantage	Disadvantage	Dataset	Work
Text information	Relationship	Approximate relation representation	(1) Text information is descriptive and semantically clear (2) High alignment efficiency and no need to refer to external resources (3) It is easier to integrate various information	(1) Weak discriminative and unvivid image of textual information (2) Poor performance on sparse k`nowledge graph entity alignment tasks	Unimodal	Wu et al. (2019a, 2019b) and Zhu et al. (2021)
		Distinguish between types of relationships				Mao et al. (2020a)
		Encoding multiple mapping relationships				Shi and Xiao (2019)
		Consider the inverse relationship				Mao et al. (2020b)
		Treating relationships as rotation of entities				Huang et al. (2022)
		Construct weighted line graphs				Zhang et al. (2024)
	Attribute	Attribute Correlation				Sun et al. (2017) and Zhang et al. (2017)
		Similarity of attribute values				Wang et al. (2020) and Guan et al. (2019)
		Number of attributes				He et al. (2019) and Xiong and Gao (2019)
	Entity summary/ description	BERT				Munne and Ichise (2020), Yang et al. (2019), and Munne and Ichise (2023)
		GRU				Chen et al. (2018)
		GCN+Bi-LSTM				Xu et al. (2020)

Table 2 (continued)

Category		Technology	Advantage	Disadvantage	Dataset	Work
	Context	Bi-LSTM				Yang et al. (2019)
		Take the mean				Wang et al. (2018)
		Dependency types				Sun et al. (2019)
		Transformer				Sun et al. (2019) and Xin et al. (2022)
		Message propagation scheme				Wang et al. (2023)
	Entity name	LSTM				Xu et al. (2019)
		Average word vector				Zeng et al. (2020) and Wei-Xin et al. (2020)
		Word embeddings and character embeddings				Lin et al. (2023)
	Type information	Type matching constraint				Lu et al. (2021)
		Encoding function				Song et al. (2021)
		Bilinear product score function				Ge et al. (2023)
		Nonlinear mapping				Zhang et al. (2023)
	Ontology information	Relative position				Xiang et al. (2021)
		Attention mechanism				Zhang et al. (2023)

three relationship patterns: symmetric/antisymmetric, inversion, and composition. Thus RpAlign can learn hybrid knowledge graph embedding.

Directed edges force adjacent information to accumulate only by the direction of flow, so some studies such as MRAEA (Mao et al. 2020b) and SHEA (Yan et al. 2021) create inverse relations for relations. ESEA (Jiang et al. 2022a) uses the symbol-based method to align relationships, and the relationship seeds further affect the alignment of multiple entities.

5.4.2 Incorporating entity attribute

The knowledge graph contains attribute triples that can provide valid information for EA. Sun et al. (2017) use the idea of Skip-gram to predict attribute relevance and refine it by clustering entities with higher attribute relevance. Zhang et al. (2017) define different feature functions based on different features, showing formally the correlation between attributes, and discovering more attribute mappings.

Not all similarities between attributes are beneficial for detecting aligned entities. Therefore to automatically find useful attributes for EA, EPEA (Wang et al. 2020) uses a CNN model to encode the sparse similarity matrix into a short and dense vector to capture the attribute similarity of two entities. AttrGNN (Liu et al. 2020) divides the knowledge graph into four subgraphs according to attribute value categories: name attribute, text attribute, numeric attribute, and no attribute, and uses Bidirectional encoder representations from transformers (BERT) to encode the attribute values. Tang et al. (2020) similarly use BERT to encode entity attribute values and compute the similarity matrix.

A simpler way to apply attribute information is to directly adopt the same way of dealing with attribute triples as with relationship triples. Haihong et al. (2020) and Trisedya et al. (2019) use TransE to learn attribute embeddings and then joint attribute embeddings and relational triple embeddings for alignment. EASA (Huang and Luo 2020) generates semantic aggregations of entities from different attributes and attribute values, and adds attribute attention to distinguish the different roles of different attributes during EA. Wang et al. (2018) and Liu et al. (2021) use GCN to generate a structural feature vector and an attribute feature vector for each entity, where Wang et al. (2018) use one-hot encoding to initialize the most frequently occurring attribute of each entity as an attribute feature vector and then combined with the structural feature vector for EA. However, selecting the most frequently occurring attributes would lead to too low differentiation among entities, so Pang et al. (2019) discard the most frequently occurring attributes to ensure both differentiation among entities and the selected entities are not long-tailed entities. Considering that the distance between attributes and attribute values affects the performance of EA, MultiKE (Zhang et al. 2019) integrates attributes and attribute values into the same matrix when processing attribute information, and then feeds them into the CNN for feature extraction.

The number of entity attributes also contributes to EA to some extent. He et al. (2019) measure entity similarity by counting the number of the same attributes between entities. Sorting the number of attributes is equivalent to setting weights for attributes, so Xiong and Gao (2019) arrange the number of attributes in descending order to improve the embedding of attribute information. Similarly, Yang et al. (2019) use the E-CBOW model for embedding attribute information and use the attention mechanism. The influence of different attributes on EA may differ significantly, so Yang et al. (2020) propose a joint attention method that calculates the attention of attribute values using attribute types, which share an attention weight with their attribute values, and captures the forward and reverse sequence information of attribute values using Bi-GRU. The self-attention mechanism plays an important role in distinguishing similar entities, CG-MuAlign (Zhu et al. 2020) and LinkNBed (Trivedi et al. 2018) also use the attention mechanism, where LinkNBed (Trivedi et al. 2018) first initializes attribute embeddings, then aggregates related embedding vectors to enrich entity and relationship embeddings by the attention mechanism, and finally captures the relationship interaction information between two entities by using the entity and relationship embedding representations.

EA model needs to consider differentiating the target entities when finding target entities for source entities. Yan et al. (2020) learn entity topics from attributes through BTM4EA, which uses entity high-level semantics for attribute modeling to filter weakly related entities. In addition, some scholars automatically generate optimal attributes based on data features to constrain the results of attribute triple alignment, such as NovEA (Sun et al. 2020) who selects optimal attributes as candidate values based on decision trees. Guan et al. (2019) apply probabilistic models to iteratively update the embedding of attributes and attribute values when performing attribute triple learning.

5.4.3 Incorporating entity summary/description

Many entities do not have attribute values, and summary embeddings can be used to reduce discrepancies. Wikidata (Vrandecic and Krötzsch 2014) provides a summary text description of the entity, containing basic information about the entity. Wang et al. (2018) use the first paragraph concepts from article data as entity descriptions, using external resources to enrich the entity embedding.

Munne and Ichise (2020), Yang et al. (2019) and EASAE (Munne and Ichise 2023) use BERT to generate a set of word vectors from the summary of each particular entity to obtain entity embeddings. Chen et al. (2018) use a multilingual word embedding pre-training corpus, and convert each entity description into a vector sequence that is input to the description encoder. A GRU incorporating a self-attentive mechanism is used to highlight sentence parts with important shared information and output the final description embedding representation.

In addition to applying a single embedding method, Xu et al. (2020) propose two text embedding models to embed the description of each entity. The Cross-TextGCN model uses GCN to encode the entities by transferring semantics between words and entities in the knowledge graph. The Cross-TextMatch model uses BiLSTM to encode entity descriptions.

5.4.4 Incorporating entity context

Entity context contains a large amount of information related to entities and relationships in the knowledge graph, with clear information sources and no noise. Therefore, fusing entity context information can enhance the knowledge representation learning ability. Yang et al. (2019) utilize contextual information to enhance the accuracy of EA, and add Jaccard coefficients to enhance the logic of contextual information. The contexts of two equivalent entities are usually similar. The stronger the contextual association of a neighbor entity with the central entity, the more alignment cues this neighbor may provide. Therefore, Wang et al. (2018) use the same encoder-independent embedding for each context, and then generate the context vector. TransEdge (Sun et al.

2019) study multiple relationships and use contextual projections to optimize the EA task under the same relationship type to facilitate the propagation of information in the graph. TransEdge extends the relationship, based on the TransE embedding structure while also using contextual projections to refine the embedding. Given that the TransE model cannot capture neighbor information, FuAlign (Wang et al. 2023) proposes a message propagation scheme to aggregate contextual information between an entity and its neighbors. DAEA (Zhang et al. 2021) generates multiple random walks for each entity to be aligned to capture its 10-hop neighborhood information and long sequence context to guide EA. JEANS (Chen et al. 2020a) performs a grounding process that links entities and text tokens in the same language to a shared vocabulary and thus discovers enough entity contexts for EA. While the above approaches are based on the context of neighbors or paths, IMEA (Xin et al. 2022) utilizes two Transformers to encode multiple contexts, including neighborhood subgraphs and paths.

5.4.5 Incorporating entity name

Given two entities, comparing their names is the simplest way to determine whether they are identical. Entity name embeddings can be used to initialize feature matrix, or they can be used as information enhancement signals for EA. Different names can be represented by similar average embeddings, and representing entity names with average word vectors enhances their ease of use, e.g., Zeng et al. (2020) use a concatenated weighted average word embedding to represent the semantic information of entity names, integrating the features on top of the separate structure and name information learned from the similarity matrix on top of the similarity matrix. Although representing entity names as averaged word vectors enhances their ease of operation, the averaging process inevitably causes a certain degree of semantic loss, which in turn cannot fully represent the semantic information of entity names. For this reason, Wei-Xin Zeng et al. (2020) propose a reordering model based on word shift distance, i.e., on the generated entity ordering results, the word shift distance model is used to further mine the entity name information and combine it with the structural information. To avoid the Out-Of-Vocabulary problem, COEA (Lin et al. 2023) combines word-level embeddings and character embeddings to perform entity alignment.

5.4.6 Incorporating type information

When two KGs have sparsity and domain feature differences in terms of structure, it can lead to significant alignment errors. Entity type information helps resolve some ambiguity and vagueness issues. Therefore JTMEA (Lu et al. 2021) combines the similarity of entity vectors and entity type matching in which type features are first extracted from entities of the same type, and then type matching constraints are applied to the comparison of candidate aligned entities. To fully use the entity type information, JETEA (Song et al. 2021) utilized the encoding function to obtain the type features of the entities to perform type matching, and the common features of the entities were extracted as the representation of the type information. TypeEA (Ge et al. 2023) considers entity type information to perform entity alignment and proposes a semantic matching-based type embedding model that utilizes the bilinear product score function to capture associations

Table 3 Comparative analysis of various multimodal local semantic information

Category	Technology	Advantage	Disadvantage	Dataset	Work
Image	VGGNet	(1) Discriminative, visual and intuitive (2) Visual features disambiguate relational information to a certain extent (3) Beneficial to long-tail knowledge graph	(1) Weak descriptiveness and low-level semantic information of images (2) Complex fusion process of each module (3) It is difficult to extract structured knowledge	Multimodal	Chen et al. (2020c) and Wang et al. (2020)
	ResNet				Liu et al. (2021)
	DenseNet				Guo et al. (2021)
	Inception network				Ni et al. (2023)
	Pretrained visual model				Wang et al. (2024)

between types. To focus on the diversity of entity roles, TTEA (Zhang et al. 2023) uses triple-aware entity augmentation to model the diversity of roles of triple elements, using a nonlinear mapping to generate type embeddings from semantic embeddings.

5.4.7 Incorporating ontology information

Incorporating ontology information contributes to solving semantic heterogeneity problems and it also enhances the generality and extensibility of entity alignment. OntoEA (Xiang et al. 2021) claims that it performs entity alignment for the first time by combining ontology information and embedding techniques, utilizing relative positions to use classes and their membership with entities. In addition, OTIEA (Zhang et al. 2023) uses the attention mechanism and designs an ontology pair enhancement approach in the encoding process to capture complex intrinsic correlations through ontology information, while complementing the semantic triples with ontology information and introducing entity role features in the decoder.

5.4.8 Incorporating image

The relational structure information in knowledge graphs may lead to ambiguity. Image features have a unified visual concept, so image can be a good source of EA information.

To extract visual features, MMEA (Chen et al. 2020c) and ACK-MMEA (Li et al. 2023) implement vectorization of images and learns image embeddings using the VGG16 model, in preparation for subsequent multimodal knowledge fusion. To establish certain linkage with text entities directly, ITMEA (Wang et al. 2020) also uses VGGNet for image feature projection to map 4096-dimensional image feature vectors into n-dimensional entity embedding vectors. EVA (Liu et al. 2021) uses ResNet-152 as a feature extractor for all images. For each image, a forward pass is done and the output of the last layer is taken as the image representation. The feature is sent through a trainable feedforward layer for the final image embedding. HMEA (Guo et al. 2021) models and integrates multimodal information in hyperbolic space, uses DenseNet to learn image embeddings. IKRL (Xie et al. 2017) uses attention to construct image-based representations that jointly consider all image instances of each entity. PoE (Liu et al. 2019) combines multimodal features and measures the plausibility of facts by matching the underlying semantics of entities and mining the relationships contained in the embedding space. Entity embeddings are learned in computing the fact scores under each modality. To allow the selected visual coder to have different receptive fields and to adapt to images from different domains. PSNEA (Ni et al. 2023) utilizes an Inception-based network to extract the visual features of entities. PCMEA (Wang et al. 2024) uses a pretrained visual model (PVM), where the visual representation is obtained through a forward propagation layer to obtain a visual embedding representation.

6 Entity alignment module

6.1 Combination method

EA based on embedding needs to use the distance between entity vectors to determine the probability of alignment. Therefore, different knowledge graphs must be embedded into a unified vector space. There are two general methods to reconcile knowledge graph embedding.

(1) Transformation

Transformation embeds knowledge graphs into different vector spaces and transforms the embedding of one knowledge graph into the vector space of another knowledge graph using a linear transfer matrix.

(2) Sharing

There are three ways to achieve sharing: (a) Let the seed entity pairs in the knowledge graph share the same embedding when creating the model (Sun et al. 2017). (b) Using pre-aligned entity pairs to generate new cross-knowledge graph triples, the triples are used as a bridge between different knowledge graphs, e.g. given a seed entity pair (h1, h2) and a triple (h1, r, t1), the exchange method will generate a new triple (h2, r, t1) (Mao et al. 2020a). (c) Minimize the distance between its vectors directly for each pre-aligned entity pair (Yan et al. 2020).

6.2 Similarity metric

In the process of EA, the similarity between entities needs to be measured, and the common similarity measures are as follows.

(1) Euclidean distance

Euclidean distance is the distance between two points in a multidimensional space. The formula for calculating the Euclidean distance between two points (x_1, y_1) and (x_2, y_2) in the two-dimensional plane is defined as:

$$d = \sqrt{\left(x_1 - x_2\right)^2 + \left(y_1 - y_2\right)^2}. \tag{11}$$

(2) Manhattan distance

Manhattan distance is used to indicate the sum of the absolute axis distances of two points on the standard coordinate system, and the calculation formula is defined as:

$$d = \left\|x_1 - x_2\right\| + \left\|y_1 - y_2\right\|. \tag{12}$$

(3) Cosine distance

Cosine distance is a measure of the magnitude of the difference between two vectors using the cosine of the two angles in vector space. The closer the cosine is to 1, the closer the angle is to 0 degrees, i.e., the more similar the two vectors are, and the formula is defined as:

$$d(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}. \tag{13}$$

(4) Cross-Domain similarity local scaling (CSLS)

To deal with the phenomenon of hubness in high-dimensional space, i.e., the existence of dense regions in vector space where some points are the nearest neighbors of many points. While the previous approach uses cosine distance to select the nearest neighbors, CSLS is calculated as:

$$\mathrm{CSLS}\left(Wx_s, y_t\right) = 2\cos\left(Wx_s, y_t\right) - r_{\mathrm{T}}\left(Wx_s\right) - r_{\mathrm{S}}\left(y_t\right), \tag{14}$$

where $r_T\left(Wx_s\right)$ is the average cosine distance between Wx_s and its K target language nearest neighbors.

(5) Edit distance

Some works use Edit distance to calculate the similarity of strings, such as the similarity calculation of entity name strings. Edit distance is a measure of the difference between two character sequences, and Edit distance between two words refers to the minimum number of single-character edit operations (insertion, deletion, or replacement) required to convert one word to another, and the calculation formula is defined as:

$$d[i,j] = \min\begin{cases} d[i,j-1]+1, \\ d[i-1,j]+1, \\ d[i-1,j-1]+c\left(s_1[i],s_2[j]\right), \end{cases}, \tag{15}$$

$$c\left(s_1[i],s_2[j]\right) = \begin{cases} 1, s_1[i] \neq s_2[j], \\ 0, s_1[i] = s_2[j], \end{cases} \tag{16}$$

where $s_1[i]$ is the ith character in string s_1 and $s_2[j]$ is the jth character in string s_2.

6.3 Alignment optimization strategy

Each element in the entity similarity matrix represents the similarity between entities. If fine-grained features can be combined into the entity similarity matrix, the accuracy of the EA model will be improved. According to prior knowledge, EA is a bidirectional matching problem between two knowledge graphs. Therefore, RAGA (Zhu et al. 2021) calculates the fine-grained similarity matrix by summing the weights of each entity aligned in both directions. Specifically, a softmax operation is applied to both rows and columns of the initial entity similarity matrix. To alleviate the uncertainty and ambiguity of the EA process, FuzzyEA (Jiang et al. 2022b) considers the uncertainty based on intuitionistic fuzzy set. Guo et al. (2022) propose a deep reinforcement learning-based framework to transform the EA problem into a sequential decision-making task. This framework can be adapted to most embedding-based EA models. DATTI (Mao et al. 2022) focuses on the decoding process and uses Adjacency tensor isomorphism equations and Gramian tensor isomorphism equations to enhance the decoding power. DATTI can bring great performance improvements in a little time.

6.4 Non-alignable entity prediction

Most of the existing studies assume that given a test source entity, an equivalent target entity can be found for it. Then in the realistic knowledge graph, some entities have no entities aligned with them (Luo and Yu 2022). SoTead (Luo et al. 2022) and WOGCL (Xu et al. 2023) convert the knowledge graph EA into an Optimal Transport problem by calling unmatched entities as hanging entities. Based on the set pseudo-entity pairs, contrast metric learning is performed to calculate the transmission consumption of the entity pairs and finally match virtual entities for the dangling entities. MHP (Liu et al. 2022a) also uses Optimal Transport for global higher-order similarity computation, and the dangling entity of the Optimal Transport is the part where the source and target entity embeddings differ. MHP considers multi-order neighbor entities when performing local similarity calculations. UEA (Zeng et al. 2021) uses a thresholded bidirectional nearest neighbor strategy to generate EA results, and the unaligned entities generated by this process are considered to be unaligned. Based on UEA, CUEA (Zhao et al. 2022) takes into account the fact that different pseudo-labeled data have different characteristics and uses confidence levels to measure the likelihood that an entity pair is true.

6.5 Alignment inference strategy

Alignment inference strategies are mainly divided into two categories: global alignment and local alignment. Table 4 summarizes and analyzes the two inference strategies.

6.5.1 Global alignment

To constrain one-to-one EA and exploit the interdependence between alignment decisions, some studies set the limit of one-to-one matching. CEA (Zeng et al. 2020) and RAGA (Zhu et al. 2021) use a deferred acceptance algorithm to find stable matching results for any two sets of the equal number of entities. No pair of two entities from different sets are more willing to match each other than the match already assigned to them. The deferred acceptance algorithm guarantees that a solution can be found in $O(N^2)$ time.

Furthermore, the global EA task is transformed into a maximum weighted bipartite graph matching problem. The Hungarian algorithm is the best solution to the task assignment problem, and the algorithm guarantees that a solution can be found in $O(N^4)$ time. GM-EHD-JEA (Xu et al. 2020) and LatsEA (Chen et al. 2021c) transform the EA problem into a task assignment problem, which is essentially a basic combinatorial optimization problem, and the exact solution can be found by the Hungarian algorithm. SEU (Mao et al. 2021) combines the Hungarian algorithm and the Sinkhorn operation. The models combining the Hungarian and other operations perform better than the models using only the Hungarian algorithm.

Although the above works achieve global EA by applying a one-to-one constraint on the EA process, they still do not adequately model the potential interdependencies. Thus, CEAFF (Zeng et al. 2021) investigates the dynamic properties of the decision process and provides a reinforcement learning-based model to implement collectively aligned entities. In the reinforcement learning framework, coherence and exclusivity constraints are designed to characterize interdependencies and restrict collective alignment. UED (Luo and Yu 2022) formulates the EA problem as an optimal transportation problem, finding the optimal global alignment by minimizing the total transportation distance.

6.5.2 Local alignment

The element in entity similarity matrix reflects the distance between the source entity and the target entity in vector space. After obtaining the embedding-based similarity matrix, the EA enters the alignment decision stage. Most of the current embedding-based EA methods, such as RNM (Zhu et al. 2021), HGCN (Wu et al. 2019b) and IMUSE (He et al. 2019) use the independent decision strategy to generate alignment results, applying greedy search strategies to find target entities for the test source entities. Specifically, given vector representations of the knowledge graph and the distance metric function. For each source entity, the alignment model uses the distance metric function to calculate the distance between the source entity vector and all target entity vectors to find alignable target entities. The above plain enumeration approach will increase the workload of EA and lead to less efficient alignment. The many-to-one situation may occur in the entity matching process, i.e., many test source entities are matched with the same test target entity. These are the limitations of the local EA strategy.

7 Post-alignment module

In the post-alignment module, the main focus is to discover more aligned entities by adding newly aligned entities to the seed set. Embedding-based EA methods use pre-aligned entities as seed data, and the performance depends heavily on the quality and quantity of seed data (Chen et al. 2020b). The data size of the knowledge graph is large, so the time and labor cost of manually labeling the aligned seeds is also large. Some studies propose using iterative training to add newly generated aligned entities to the seed set, which expands the size of the seed set and guides the subsequent training process. Table 5 provides a comparative analysis of various iterative strategies for the post-alignment module.

IPTransE (Zhu et al. 2017), RATransE (Haihong et al. 2020) and East (Zeng et al. 2019) design hard alignment strategy and soft alignment strategy. The hard alignment strategy directly applies the parameter sharing model of the joint embedding part to the process of generating new aligned entities, adding the new aligned entity pairs to the seed set. The soft alignment strategy is to solve the problem of error accumulation generated during hard alignment. Kang et al. (2020) and Shize et al. (2019) use a re-initialization strategy in addition to a soft alignment strategy. By reinitializing the embedding and the newly aligned entity pair set in each iteration, the propagation of errors in the next generation is reduced.

The above methods can only introduce a small number of high-confidence entity pairs, which cannot bring significant effect improvement. Wei-Xin Zeng et al. (2020) design an iterative strategy of "easy to hard". The method uses the degree of entity nodes as a measure, considers entities with higher degrees as easy courses and long-tailed entities as difficult courses, and adds high-confidence entity pairs to the training set in a easy-to-hard way. Qu et al. (2019) consider entity pairs with alignment probability higher than a predefined threshold as easy alignment and the rest of entity pairs as hard alignment, and if more than K easy alignments are found in each iteration, they are added to the seed set to continue the iteration, otherwise, the iteration ends. There are problems such as the easy introduction of wrong samples and low efficiency. On this basis, Ge et al. (2021) use a refinement strategy to optimize the quality of new seed alignments generated by iterations and provide a plausible seed generator to generate pseudo-seed alignments.

BootEA (Sun et al. 2018) applies Bootstrapping (Yarowsky 1995) to iteratively expand the size of the seed set. The iterative process inevitably generates incorrect marks, and incorrect training samples can mislead the subsequent training process, so alignment editing methods are used to reduce the error accumulation. Similar to Bootstrapping, Lu et al. (2021) and Song et al. (2021) employ the strategy of error evaluation during the iteration process to make marked entities be marked or unmarked in subsequent iterations. Lin et al. (2021) propose an attribute combination bidirectional full filtering strategy to generate semi-supervised data, no longer using only bootstrap positive samples as input but adding negative samples while iterating. The above methods using Bootstrapping do not consider the effect of seed entity selection on entity vector representation. Therefore, Chen et al. (2020b) consider the centrality and differentiability of entities to select entities in addition to the iterative strategy of BootEA. Better knowledge graph alignment is achieved by using only a small number of high-quality seed-aligned entities.

Bootstrapping has achieved significant performance improvements. However, it is based on complex selection criteria that inevitably introduce a set of hyperparameters. Therefore based on the nature of entity one-to-one correspondence and EA direction asymmetry, MRAEA (Mao et al. 2020b), JEANS (Chen et al. 2020a), EVA (Liu et al. 2021), Inga (Pang et al. 2019), RANM (Cai et al. 2023) and AdaptiveEA (Zhang et al. 2021) propose bidirectional iterative strategies. Specifically, the entity pair (e_i, e_j) is considered as a newly predicted aligned entity in the current iteration when and only when the entities e_i and e_j are the nearest neighbors of each other. This approach effectively alleviates the error propagation problem. However, in some cases even if e_i and e_j satisfy mutual nearest neighbors, the similarity between them may still be low, so a thresholded bidirectional nearest neighbor search strategy is proposed in UEA (Zeng et al. 2021) to generate alignment results. The strategy requires that the distance between them is below a given dynamic threshold θ before (e_i, e_j) is considered aligned. Based on bidirectional iterative strategy, DuGa-DIT (Xie et al. 2022) utilizes the newly increased EA pairs to dynamically adjust the attention score matrix and objective function across graphs. Bidirectional iteration is less often considered for adding negative samples to the input, Lin et al. (2021) propose an attribute combination bidirectional full filtering strategy to generate semi-supervised data, instead of using only bootstrap positive samples as input, negative samples are added while iterating. It is filtered by one-to-one constraints based on the unknown correctness of the local alignment.

The active learning algorithm uses the alignment results of the previous iteration in an iterative self-learning manner to update the embedding. Zeng et al. (2021) employ active learning to select entities to be manually labeled to maximize model performance with minimal effort. Given a label budget B, at each iteration, guided by a query policy, the entities with b (b < B) having the most amount of information for labeling are selected and pairs with these entities are added to the data with labels for training the EA model, iterating until the label budget is exhausted. JEANS (Chen et al. 2020a) captures cross-lingual correspondences of entities and lexical elements in a self-learning manner. Starting with a small number of seed EAs, transitions between language-specific embedding spaces are iteratively induced, and more entity and lexical element alignments are inferred in each iteration. DAGCN (Wang et al. 2022) uses the adversarial idea of degree perception to iteratively train generator and discriminator. DAGCN uses the discriminator to adjust the embedding representation in the generator and updates the parameters of the discriminator based on the embedding. When the entity degree difference is not detected, the effect of the degree difference is eliminated.

8 Negative sampling

In the EA task, using negative samples helps improve the performance of the model. The commonly used negative sampling techniques in the field of EA include uniform negative sampling, truncated negative sampling, and nearest neighbor negative sampling. Some scholars also use other negative sampling methods. Table 6 shows the negative sampling methods applied by EA model.

9 Loss function

Loss function is used to estimate the degree of inconsistency between the predicted value $f(x)$ and the true value Y. Loss function is a non-negative real-valued function usually expressed using $L(Y, f(x))$. The smaller the value, the better the robustness of the model. The loss functions commonly used in the field of EA are shown in Table 7.

10 Benchmarking

10.1 Dataset

10.1.1 Unimodal dataset

Commonly used unimodal English datasets in the field of EA are classified into monolingual and cross-lingual datasets, which are mostly extracted from open-linked datasets according to different requirements. Table 8 provides statistics on the commonly used unimodal English datasets. In addition to the English dataset, we also introduce the commonly used Chinese dataset.

(1) DBP15K

The DBP15K[1] (Sun et al. 2017) dataset includes three cross-lingual sub-datasets constructed from the DBpedia dataset. Links between 15,000 popular entities are extracted from English to Chinese, Japanese and French respectively. Usually, the number of entities involved in each language is much more than 15,000, and attribute triples make up a large proportion of the dataset.

(2) DWY100K

The DWY100K[2] (Sun et al. 2018) dataset is monolingual dataset. DWY100K contains two large-scale sub-datasets drawn from DBpedia, Wikidata, and YAGO3, represented by DBP–WD and DBP–YG, respectively. Each dataset has 100,000 reference entity alignments. The extraction method follows the DBP15K extraction method. Taking DBP–WD as an example, 100,000 aligned entity pairs were randomly extracted from the English version of DBpedia to Wikidata.

(3) SRPRS

RSNs[3] (Guo et al. 2019) first proposed the SRPRS dataset, which can control the degree distribution of entities in the sampled dataset. Here, the degree of an entity is defined as the number of relational triples that the entity is associated with.

(4) DBP v1.1

DBP v1.1[4] (Sun et al. 2020) contains cross knowledge graph and cross-lingual settings. The dataset consists of two versions, where the v1 version is generated using iterative degree-based sampling (IDS) method, while the v2 version first randomly removes entities with degree less than or equal to 5 and then uses the IDS method, which increases the density.

(5) Chinese dataset

[1] https://github.com/nju-websoft/JAPE/tree/master/data.

[2] https://github.com/nju-websoft/BootEA/tree/master/dataset.

[3] https://github.com/nju-websoft/RSN.

[4] https://github.com/nju-websoft/OpenEA.

Table 4 Alignment comparison analysis

Category			Feature	1-to-1 constraint	Plain enumeration	Time complexity	Work
Global alignment	Collective search	Deferred acceptance algorithm	Fast, guaranteed stable matching results, requires close number of entities on both sides of the input	✓	×	Low	Zeng et al. (2020) and Zhu et al. (2021)
		Task assignment algorithm	High accuracy, not easy to solve	✓	×	High	Xu et al. (2020) and Chen et al. (2021c)
		Optimal transportation problem	Geometric methods for modeling probability distributions and measuring distances between probability distributions	✓	×	Low	Luo and Yu (2022)
	Reinforced learning	Sequence decision	High model performance, optimizing decisions for all entities instead of optimizing each decision individually	×	×	Low	Zeng et al. (2021)
Local alignment		Greedy algorithm	Simple and easy to implement, aligned entities are not deprivable, and error propagation is severe	×	✓	Low	Zhu et al. (2021), Wu et al. (2019b), and He et al. (2019)

Huang and Luo (2020)[5] collect data from Baidu Encyclopedia and Interactive Encyclopedia in the military domain and extracted triples from the infoboxes, named Dataset-1 on a small scale. Similarly, the entertainment dataset was collected from Baidu Encyclopedia and Interactive Encyclopedia, named Dataset-2 on a large scale. As is shown in Table 9, the Chinese dataset includes the number of entities, the number of relationships, and the total number of fact triples. The triples of the Interactive Encyclopedia and Baidu Encyclopedia are merged.

10.1.2 Multimodal dataset

In the EA domain, two multimodal datasets constructed in MMKG[6] (Liu et al. 2019), namely FB15K–DB15K and FB15K–YAGO15K. FB15K is a representative subset extracted from the Freebase knowledge base. To maintain an approximation to the number of entities in FB15K, DBpedia's DB15K and YAGO's YAGO15K are formed mainly based on FB15K, using links to align the entities in FB15K with those in other knowledge graphs. Table 10 describes the statistics of the multimodal datasets. Each dataset contains nearly 15,000 entities and over 11,000 entity image sets.

10.2 Evaluation metric

Three evaluation metrics are usually used to evaluate the performance of EA, namely *Hits@k*, *MR*, and *MRR*. In the EA task, *Hits@k* represents the calculation of the proportion of the top k correctly aligned entities, *MR* represents the average ranking of all correctly aligned entities, and *MRR* represents the reciprocal of the average ranking of all correctly aligned entities in the alignment results.

(1) *Hits@k*

The proportion of correctly aligned entities ranked in the top k (k is usually taken as 1, 10). If there are correctly aligned entities among the top k candidates, the exact number of *Hits@k* is increased by 1. The higher the value of *Hits@k*, the better the model results. The *Hits@k* is the exact number of aligned entities divided by the total number of aligned entities and is calculated as follows.

$$\text{Hits@}k = \frac{\text{count}\left(\left\{e \in S \mid \text{rank}_e \leq k\right\}\right)}{\text{count}(S)}, \tag{17}$$

where *count*(S) denotes the number of elements in the set; $rank_e$ is the true rank of entity e, and S is the total number of candidate aligned entities.

(2) Mean rank (*MR*)

The average rank of all correctly aligned entities in the results of EA experiments, the lower the *MR* value, the better the model effect.

$$\text{MR} = \frac{\sum_{e \in S} \text{rank}_e}{\text{count}(S)}, \tag{18}$$

[5] https://github.com/hatleon/EASA/tree/master/data.

[6] https://github.com/mniepert/mmkb.

Table 5 Comparative analysis of post-alignment module

Category	Idea	Feature	Work
Hard alignment and soft alignment	Newly aligned entities are added to the seed set based on confidence level	Reduce the manual labeling time and introduce only a small number of high-confidence entity pairs	Zhu et al. (2017), Haihong et al. (2020), and Zeng et al. (2019)
Initialization strategy	Clear the last generated set of new entity pairs	Reduce error propagation, strategy stacking increases time consumption	Kang et al. (2020) and Shize et al. (2019)
Alignment from easy to hard	Add high-confidence entity pairs to the training set in a simple-to-hard way	Enhance the correctness of the representation, but inefficient	Wei-Xin et al. (2020) and Qu et al. (2019)
Bootstrpping and its variants	Use Bootstrapping to mark aligned samples and repeat to get new samples that can represent the parent samples	Suitable for large-scale data training, there will be marker conflicts and error propagation problems	Sun et al. (2018), Lu et al. (2021), Song et al. (2021), Lin et al. (2021), and Chen et al. (2020b)
Bidirectional alignment	An entity pair is considered aligned only if both entities in the pair are nearest neighbors of each other	Avoid side bias and mitigate error propagation	Mao et al. (2020b), Chen et al. (2020a), Liu et al. (2021), Pang et al. (2019), Zhang et al. (2021) Zeng et al. (2021), Lin et al. (2021), Xie et al. (2022), and Cai et al. (2023)
Self-learning	Incorporate active learning algorithms into iterations to label entities with high information content	Can reduce the cost of labeling, information complexity increases, there is an accumulation of errors	Zeng et al. (2021) and Chen et al. (2020a)
Adversarial learning	Iterative training of generator and discriminator	Eliminate degree difference	Wang et al. (2022)

where *count*(*S*) denotes the number of elements in the set; $rank_e$ is the true rank of entity *e*, and *S* is the total number of candidate aligned entities.

(3) Mean reciprocal rank (*MRR*)

MRR is the reciprocal of the average rank of all correctly aligned entities in the EA results. The higher the value of *MRR*, the better the model effect, which is calculated as follows.

$$MRR = \frac{1}{\mathrm{count}(S)} \sum_{e \in S} \frac{1}{\mathrm{rank}_e}, \tag{19}$$

where *count*(*S*) denotes the number of elements in the set; $rank_e$ is the true rank of entity *e*, and *S* is the total number of candidate aligned entities.

11 Experiment and analysis

11.1 Effect of direction on performance

EA is a bidirectional alignment problem. To explore the influence of alignment direction on the model, we study the performance of some representative models such as GCN-Align, HMAN, and GMNN in two directions on DBP15K. The experimental results are shown in Tables 11, 12, 13. For example, $DBP15K_{ZH-EN}$ in Table 11 represents the source knowledge graph as a Chinese dataset, while the target knowledge graph represents an English dataset.

The results show that the model performs better in most cases when the English knowledge graph is used as the target knowledge graph. This is because the knowledge graph of English is denser compared to the knowledge graphs of other languages. However, HMAN, AKE, and DAEA perform better or close to the forward alignment effect when reverse alignment is performed on the test dataset. All three models use external factors of entity description, attributes, or entity frequencies. This indicates that external information can alleviate the lack of sparsity of knowledge graph structure and facilitate EA. Therefore, this paper presents the analysis of the experiments on English knowledge graphs as the target knowledge graphs.

11.2 Experimental setting

To ensure fairness, the model performance on different datasets is analyzed separately in this paper. Under the unimodal dataset, 30% of the seeds are used for training, and the remaining 70% are used for testing. Under the multimodal dataset, 20%, 50%, and 80% of the seeds are used for training, and the rest are used for testing. Unless otherwise specified, the experimental results in this paper are from the original paper. Table 14 shows the information of representative EA models.

Combined with the current research trends in the field, existing EA methods are classified into four categories in this paper: category 1, global alignment models, category 2, using noise filtering strategy, category 3, using only global structural information, and category 4, combining global structural and local semantic information. Some models may belong to different categories, for example, RAGA can be classified into categories 1, 2,

Table 6 Negative sampling method

Category	Introduction	Work
Uniform negative sampling	Randomly select one entity from all entities to replace the head entity or tail entity of a positive triple	Trisedya et al. (2019), Zhu et al. (2017), Chen et al. (2020c), Munne and Ichise (2020), Kang et al. (2020), Yang et al. (2019), Ye et al. (2019), Sun et al. (2020), Xie et al. (2020), Mao et al. (2020a), Mao et al. (2020b), Zhu et al. (2020), Sun et al. (2017), and Chen et al. (2020d) Wang et al. (2018), Zhang et al. (2019), Tang et al. (2020), He et al. (2019), Trivedi et al. (2018), Sun et al. (2020), Zhang et al. (2017), Zhang et al. (2021), Zeng et al. (2019), Zeng et al. (2021), Xu et al. (2020), Chen et al. (2020a), Pei et al. (2019b), Yu et al. (2020), and Chen et al. (2017)
Truncated negative sampling	Limit the sampling to the set of candidate entities when replacing an entity in the triplet	Yang et al. (2020), Sun et al. (2018), Sun et al. (2019), Shi and Xiao (2019), Nie et al. (2020), Cao et al. (2019), and Sun et al. (2020)
Nearest neighbor negative sampling	For entities in positive triples, the set of negative samples is the top nearest neighbor of the entity vector obtained from the negative sampling model	Zhu et al. (2021), Jiang et al. (2019), Yan et al. (2020), Sun et al. (2020), Zhu et al. (2019), Zhu et al. (2021), Wu et al. (2020), Wu et al. (2019b), Wu et al. (2019a), Liu et al. (2020), Chen et al. (2020b), and Li et al. (2019)
Negative sampling of noise distribution	The sampling process requires specifying the overall noise distribution. Commonly used noise distributions are Bernoulli distributions	Guo et al. (2019), Li et al. (2018), and Zhang et al. (2018)
Type-aware negative sampling	Based on the locally closed world assumption, which is bounded not at the class level but at the instance level of the triple	Lin et al. (2019)

and 4. However, we set the classification priority so that if a model is already classified into a category, it will not participate in the intra-class comparison of subsequent categories. Therefore, RAGA is classified into the first category. We perform a comparative analysis of models within categories.

The first category is EA from the global perspective. Most of the current EA models use plain enumeration alignment, which has room for improvement in both accuracy and alignment efficiency. Global EA can effectively exploit the interdependencies between alignment decisions to ensure one-to-one entity matching. Therefore, we separate global alignment and local alignment for experimental comparison and analysis.

The second category is models that employ a noise filtering strategy. As the number of network layers increases, the model can more effectively aggregate information from neighbors and capture structural representations. But this inevitably introduces noise that is not conducive to the learning of entity representations. Therefore, we analyze whether noise filtering has a positive impact on model performance.

The third category is the models using only structural information. Representation learning of knowledge graph is essentially the process of mining graph features, and the structure of the knowledge graph is an important basis for obtaining entity representations. Therefore, we classify the models that utilize only global structural information into a separate category.

The fourth category is the models that combine global structure and local semantics. When the knowledge graph structure is sparse, local semantics such as attributes will provide useful alignment signals for EA.

11.3 Comparison of unimodal EA models

11.3.1 Comparative analysis of models on DBP15K

The performance of model on DBP15K is shown in Table 15. DBP15K is a small, dense cross-lingual dataset. We divide the models into four classes and then performed intra-class comparisons as follows.

Category 1 among global EA models, SoTead performs the best on Hits@1 compared to the other models on the three sub-datasets of DBP15K. SEU is the second-best performer in the first category. Both SoTead and SEU models are unsupervised approaches, and their good performance lies in the fact that SoTead transforms the EA problem into an optimal transport problem, while SEU transforms the EA into a task assignment problem, significantly reducing the complexity of operations on the neural network. LatsEA performs the weakest. The other models consider entity names or relational local semantic information, while LatsEA puts the focus on global structural embedding. CEA, CEAFF, and GM-EHD-JEA all focus on the collective alignment process. While GM-EHD-JEA treats EA as a task assignment problem and uses the Hungarian algorithm. CEA views EA as a stable matching problem and solves it using a deferred acceptance algorithm. CEAFF collectively aligns source entities through reinforcement learning, which can adequately capture the interdependencies between EA decisions. Thus CEAFF achieves better results than the collective alignment methods CEA and GM-EHD-JEA. RAGA achieves better and more significant results than the collective alignment methods CEA, CEAFF, and GM-EHD-JEA. This is because RAGA incorporates the local semantics of relations and fine-tunes the

Table 7 Loss function

Name	Introduction	Work
Margin-based ranking loss	In vector space, to keep matched entity pairs as close as possible and mismatched entity pairs as far away as possible, Margin-based ranking loss is defined as follows. $L = \sum_{(p,q)\in L} \sum_{(p',q')\in L'} \max\left\{0, D(p,q) - D\left(p',q'\right) + \gamma\right\}$ where γ is the margin hyperparameter; L' represents the set of negative examples of L	Trisedya et al. (2019), Zhu et al. (2017), Chen et al. (2020c), Sun et al. (2020), Kang et al. (2020), Zhu et al. (2021), Xiong and Gao (2019), Yang et al. (2019), and Ye et al. (2019) Wu et al. (2020), Xie et al. (2020), Wu et al. (2019b), Wu et al. (2019a), Mao et al. (2020a), Mao et al. (2020b), Wang et al. (2020), and Pei et al. (2020) Haihong et al. (2020), Wang et al. (2018), Pang et al. (2019), Liu et al. (2020), Tang et al. (2020), He et al. (2019), Li et al. (2019), and Lin et al. (2019) Trivedi et al. (2018), Sun et al. (2020), Zhang et al. (2021), Zeng et al. (2019), Zeng et al. (2021), Xu et al. (2020), andPei et al. (2019) Hao et al. (2016) and Yu et al. (2020)
Contrastive loss	Similar samples are still guaranteed to be similar in the lower dimensional space after the dimensionality reduction operation, and conversely, dissimilar samples are still guaranteed to be dissimilar in the lower dimensional space after the dimensionality reduction operation with the following equation. $L = \frac{1}{2N} \sum_{n=1}^{N} yd^2 + (1-y)\max(\text{margin} - \text{d}, 0)^2$ where d represents the Euclidean distance of two sample features; y is the label of whether two samples match; margin is the threshold	Chen et al. (2018) and Zhu et al. (2021, 2020)
Triplet loss	Triplet loss clusters the anchor examples with the positive examples by training and learning the neural network so that the anchor examples are as close to the positive examples and as far away from the negative examples as possible. The formula is as follows. $L = \sum_i^N \left[\left\|f\left(\mathrm{x}_i^a\right) - f\left(\mathrm{x}_i^p\right)\right\|_2^2 - \left\|f\left(\mathrm{x}_i^a\right) - f\left(textrmx_i^n\right)\right\|_2^2 + \alpha\right]_+$ where α is the distance hyperparameter; $f\left(\mathrm{x}_i^a\right)$, $f\left(\mathrm{x}_i^p\right)$ and $f\left(\mathrm{x}_i^n\right)$ are the embedding representations of the anchor example, positive example and negative example, respectively	Cao et al. (2019)

Table 7 (continued)

Name	Introduction	Work
Cross-entropy loss	Cross-entropy loss uses the gradient descent method for parameter updating, and the smaller the value of cross-entropy, the closer the two probability distributions are. The formula is as follows. $L = -[y \cdot \log(p) + (1 - y) \cdot \log(1 - p)]$ where p denotes the probability that the sample is predicted to be positive; y denotes the label of the sample	Wang et al. (2018) , Yang et al. (2019) , and de Assis Costa and de Oliveira (2018)
Logistic loss	Logistic loss is mainly used in binary classification tasks. When embedding entities and relations, it can be used to handle binary relation tasks such as symmetric and asymmetric relations. Logistic loss is defined as follows. $L = \sum_{i=1}^{N} \log\left(1 + \exp\left(-y_i\eta_i\right)\right)$ where $y_i \in \{\pm 1\}$	Shi and Xiao (2019) and Zhang et al. (2019)
Limited loss	Limited loss puts the score of a positive triple in a low value range. Limited loss is defined as follows. $L = \sum_{(h,r,t)\in\Delta} [f(h,r,t) - \gamma]_+$ where $[x]_+ = \max(0, x)$ denotes; γ is a hyperparameter	Sun et al. (2018) and Zhu et al. (2019)

Table 8 Unimodal English dataset

Dataset			Entity	Relation	Attribute	Relation triple	Attribute triple	Aligned entity	Aligned relation
DBP15K	ZH–EN	Chinese	66,469	2830	8113	153,929	379,684	15,000	890
		English	98,125	2317	7173	237,674	567,755		
	JA–EN	Japanese	65,744	2043	5882	164,373	354,619	15,000	529
		English	95,680	2096	6066	233,319	497,230		
	FR–EN	French	66,858	1379	4547	192,191	528,665	15,000	212
		English	105,889	2209	6422	278,590	576,543		
DWY100K	DBP–WD	DBpedia	100,000	330	351	463,294	381,166	–	–
		Wikidata	100,000	220	729	448,774	789,815	–	–
	DBP–YG	DBpedia	100,000	302	334	428,952	451,646	–	–
		YAGO3	100,000	31	23	502,563	118,376	–	-
SRPRS	DBP–WD	DBpedia	15,000	253	–	38,421	–	15,000	–
		Wikidata	15,000	144	–	40,159	–		–
	DBP–YG	DBpedia	15,000	223	–	33,571	–	15,000	–
		YAGO3	15,000	30	–	34,660	–		–
	EN–FR	DBpedia	15,000	221	–	36,508	–	15,000	–
		DBpedia	15,000	177	–	33,532	–		–
	EN–DE	DBpedia	15,000	222	–	38,281	–	15,000	–
		DBpedia	15,000	120	–	37,069	–		–
DBP v1.1	EN–FR	DBpedia	–	267	308	47,334	73,121	–	–
	15K(v1)	DBpedia	–	210	404	40,864	67,167	–	–
	EN–FR	DBpedia	–	193	189	96,318	66,899	–	–
	15K(v2)	DBpedia	-	166	221	80,112	68,779	–	–
	D–W	DBpedia	-	248	342	38,265	68,258	–	–
	15K(v1)	Wikidata	–	169	649	42,746	138,246	–	–
	D–W	DBpedia	–	167	175	73,983	66,813	–	–
	15K(v2)	Wikidata	-	121	457	83,365	175,686	–	–

Table 9 Chinese dataset

Dataset		Entity	Relation	Triple
Dataset-1	Baidu Encyclopedia	12,714	1,406	20,707
	Interactive Encyclopedia	11,181	910	19,693
Dataset-2	Baidu Encyclopedia	110,952	3,789	227,909
	Interactive Encyclopedia	106,737	2,935	236,245

similarity matrix to consider fine-grained semantic features. This shows that combining local semantic information as well as fine-tuning the similarity matrix is meaningful for EA.

Category 2 on DBP15K, RNM performs steadily in the top position on the three sub-datasets. HGCN, RDGCN, and SSP have similar performances because they all use local relational semantics. NMN performs slightly better than HGCN, RDGCN, and SSP. This is because, in addition to the embedding structure, NMN considers neighborhood matching. The performance of RNM is much improved than NMN, which is because RNM further introduces relational based on NMN while using collaborative training. This shows

Table 10 Multimodal dataset

Dataset	Entity	Relation	Attribute	Relation triple	Attribute triple	Image	Link
FB15K	14,951	1,345	116	592,213	29,395	13,444	–
DB15K	12,842	279	225	89,197	48,080	12,837	12,846
YAGO15K	15,404	32	7	122,886	23,532	11,194	11,199

the importance of combining global structure and local semantics, while co-training can promote the performance of the model.

Category 3 MuGNN and KECG obtain better results than MTransE on DBP15K, which indicates that the deep model is more capable of capturing structural features compared to the translation model. Specifically, MuGNN uses a multi-channel graph neural network to capture structural information. KECG adopts a similar idea by jointly learning entity embeddings and encoding intra-graphic relations and neighborhood information. However, their performance is still weaker than that of RSNs. This is because RSNs improve performance by considering long-term relational dependencies between entities, which captures more structural signals for alignment. BootEA ranks first in all metrics for all sub-datasets in the category 3 that utilize structure only, because BootEA uses a bootstrap strategy that adds the generated new EA pairs to the seed set. This indicates that the bootstrap strategy has a positive impact on improving performance.

Category 4 the results of JTMEA are higher than JAPE and close to GCN-Align. JTMEA has better ability to capture semantic information during embedding. Both JTMEA and JETEA use entity type information and achieve close performance. TTEA uses type enhancement while considering triple specificity and role diversity to perform entity alignment, resulting in superior performance. Compared to GCN-Align and JAPE, HMAN works better because HMAN considers entity description information in addition to attribute information. This indicates that exploring multiple local semantic information is useful. Compared to HMAN, JAPE, and GCN-Align, GM-Align performs better because GM-Align uses an entity name-based initialization. While HMAN, JAPE, and GCN-Align use a random initialization embedding. MRAEA, TransEdge, and NAEA all use an iterative strategy along with local semantic information. The overall performance of the methods using iteration exceeds that of the other methods. MRAEA performs better than TransEdge and NAEA. TransEdge and NAEA use a bootstrap iteration strategy and they do not consider direction, while MRAEA uses a bidirectional iteration strategy, which indicates that the effect of direction on EA deserves attention. OTIEA utilizes ontology-enhanced triple encoders by mining intrinsic associations and ontology pair information with better results. In this category, FuzzyEA performs the best, FuzzyEA fuses entity names and descriptions and takes into account the uncertainty caused by a single metric of the alignment process.

11.3.2 Comparative analysis of models on DWY100K

The performance of each model on DWY100K is shown in Table 16. DWY100K is a large-scale, dense monolingual dataset. We classify the models into four categories and then the classification comparison is performed as follows.

Table 11 The effect of direction on the model on $DBP15K_{ZH-EN}$

Model	ZH–EN		EN–ZH	
	Hits@1	Hits@10	Hits@1	Hits@10
GCN-Align (Wang et al. 2018)	0.413	0.744	0.365	0.699
HMAN (Yang et al. 2019)	0.562	0.851	0.537	0.834
GMNN (Xu et al. 2019)	0.679	0.785	0.653	0.796
AKE (Lin et al. 2019)	0.249	0.665	0.253	0.679
Inga (Pang et al. 2019)	0.505	0.794	0.494	0.761
MMREA (Shi and Xiao 2019)	0.681	0.867	0.677	0.866
DGM (Fey et al. 2020)	0.801	0.875	0.768	0.836
DAEA (Zhang et al. 2021)	0.828	0.924	0.888	0.961
STGCN (Xu et al. 2020)	0.465	0.779	0.406	0.718
JAPE (Sun et al. 2017)	0.412	0.745	0.402	0.711

Category 1 on DWY100K, both CEA and GM-EHD-JEA focus on the collective alignment process. While GM-EHD-JEA uses the Hungarian algorithm to maximize the local similarity score under one-to-one constraints. CEA uses the deferred acceptance algorithm to guarantee one-to-one alignment, i.e., optimal assignment of one source entity. CEA consistently outperforms GM-EHD-JEA on all sub-datasets of DWY100K, which indicates that GM-EHD-JEA using the search space separation strategy can harm the performance of the model. Moreover, the Hits@1 metric of CEA has reached 1. On DBP15K, the performance of CEA is much less than that of DWY100K, and the same phenomenon occurs for the GM-EHD-JEA model. This is because the similarity between the names of entities to be aligned is higher for the DWY100K dataset compared to DBP15K.

Category 2 on DWY100K, the performance of HGCN and NMN is much better than that of SSP, while the model performance of HGCN and NMN is closer. Both HGCN and NMN use initialization based on entity names, and HGCN considers local relational semantics in addition to using entity names. While SSP also makes use of relational information, it is not initialized based on entity names, so the performance of SSP is much inferior to HGCN and NMN, which shows that the initialization embedding method has an impact on the model performance.

Category 3 on DWY100K, MTransE performs the worst because it only considers the original topological information. MuGNN and KECG perform much better than MTransE because MuGNN uses a multi-channel graph neural network to capture structural information at different levels. KECG jointly learns entity embeddings and proximity information is considered. It is worth noting that the RSNs model does not outperform MuGNN and KECG on DWY100K. RSNs improve performance by considering long-term relational dependencies between entities, but when structural data are more adequate, long-term relational dependencies may not be significantly effective. BootEA still ranks first in the metrics for both datasets of DWY100K, which indicates that the bootstrap strategy improves the performance of the EA model on both large-scale and small-scale datasets. In addition, the performance of each model on DWY100K is significantly improved compared to the performance on small-scale data like DBP15K. For example, MuGNN has a significant improvement in Hits@1, which is due to the larger

Table 12 The effect of direction on the model on $DBP15K_{JA--EN}$

Model	JA–EN		EN–JA	
	Hits@1	Hits@10	Hits@1	Hits@10
GCN-Align (Wang et al. 2018)	0.399	0.745	0.384	0.718
HMAN (Yang et al. 2019)	0.567	0.869	0.565	0.866
GMNN (Xu et al. 2019)	0.740	0.872	0.713	0.846
AKE (Lin et al. 2019)	0.281	0.691	0.273	0.682
Inga (Pang et al. 2019)	0.515	0.795	0.511	0.770
MMREA (Shi and Xiao 2019)	0.655	0.859	0.652	0.858
DGM (Fey et al. 2020)	0.848	0.897	0.811	0.868
MEEA (Chen et al. 2021a)	0.556	0.790	0.548	0.788
DAEA (Zhang et al. 2021)	0.870	0.951	0.910	0.970
STGCN (Xu et al. 2020)	0.454	0.759	0.436	0.745
JAPE (Sun et al. 2017)	0.363	0.685	0.384	0.673

Table 13 The effect of direction on the model on $DBP15K_{FR-EN}$

Model	FR–EN		EN–FR	
	Hits@1	Hits@10	Hits@1	Hits@10
GCN-Align (Wang et al. 2018)	0.373	0.745	0.368	0.731
HMAN (Yang et al. 2019)	0.540	0.871	0.543	0.867
GMNN (Xu et al. 2019)	0.894	0.952	0.882	0.948
AKE (Lin et al. 2019)	0.240	0.674	0.247	0.693
Inga (Pang et al. 2019)	0.505	0.794	0.494	0.761
MMREA (Shi and Xiao 2019)	0.677	0.890	0.673	0.891
DGM (Fey et al. 2020)	0.933	0.960	0.920	0.953
MEEA (Chen et al. 2021a)	0.578	0.811	0.571	0.792
DAEA (Zhang et al. 2021)	0.936	0.971	0.952	0.980
STGCN (Xu et al. 2020)	0.521	0.827	0.514	0.823
JAPE (Sun et al. 2017)	0.324	0.667	0.330	0.659

dataset size and more structural information of the model, which can provide more support for EA.

Category 4 RALG performs best because it creates a heterogeneous line graph and it is used to independently learn the relational representation of entities. SelfKG achieves the second-best performance and it uses self-supervised alignment. The DWY100K dataset is a monolingual dataset with high entity name similarity, so for monolingual datasets, alignment is easier and supervised learning is not necessary. SHEA has better performance on DWY100K. Because SHEA considers both intra-graph and cross-graph attention mechanisms when it learns alignment-oriented entity embeddings. EASAE uses both summary and attribute embedding, so alignment performance is great. GM-Align performs well. Because GM-Align implements

Table 14 Representative EA model information

Model	Embedding				Alignment			Post alignment	Category
	Initialization	Global structure	Interaction	Local semantic	Combination method	Distance metric	Alignment method		
LatsEA (Chen et al. 2021c)	Random	HAKE+ GCN	No	No	Transformation	Euclidean	Global	No	1
CEA (Zeng et al. 2020)	Name	GCN	Concatenation	Name	Sharing	Cosine	Global	No	
GM-EHD-JEA (Xu et al. 2020)	Name	GCN	Collaboration	Name	Sharing	–	Global	No	
RAGA (Zhu et al. 2021)	Name	GCN	Concatenation	Name Relation	Sharing	Manhattan	Global	No	
CEAFF (Zeng et al. 2021)	Random	GCN	Concatenation	Name	Sharing	Bray–Curtis dissimilarity	Global	No	
SEU (Mao et al. 2021)	Name	No	No	Name Context	Transformation	Sinkhorn	Global	No	
HGCN (Wu et al. 2019b)	Name	GCN	Collaboration	Name Relation	Sharing	Manhattan	Local	No	2
RDGCN (Wu et al. 2019a)	Name	DPGCNN	Concatenation	Name Relation	Sharing	Manhattan	Local	No	
RNM (Zhu et al. 2021)	Name	GCN	Collaboration	Name Relation	Sharing	Manhattan	Local	No	
NMN (Wu et al. 2020)	Name	GCN	Concatenation	Name	Sharing	Manhattan	Local	No	
SSP (Nie et al. 2020)	Random	GCN	No	Relation	Sharing	–	Local	No	
DAGCN (Wang et al. 2022)	Name	GCN	No	No	Sharing	Manhattan	Local	No	
MTransE (Chen et al. 2017)	Random	TransE	No	No	Transformation	Euclidean	Local	No	3
MuGNN (Cao et al. 2019)	Random	GNN+ GAT	No	No	Sharing	Cosine	Local	No	
KECG (Li et al. 2019)	Random	TransE+ GAT	No	No	Transformation	Euclidean	Local	No	
RSNs (Guo et al. 2019)	Random	Path sequence	No	No	Sharing	Cosine	Local	No	
BootEA (Sun et al. 2018)	Random	TransE	No	No	Sharing	Cosine	Local	Yes	

Table 14 (continued)

Model	Embedding				Alignment			Post alignment	Category
	Initialization	Global structure	Interaction	Local semantic	Combination method	Distance metric	Alignment method		
NAEA (Zhu et al. 2019)	Random	TransE	No	Relation	Sharing	Cosine	Local	Yes	4
TransEdge (Sun et al. 2019)	Random	TransE	Concatenation	Context	Sharing	Cosine	Local	Yes	
MRAEA (Mao et al. (2020b))	Random	GNN	Concatenation	Relation	Sharing	Manhattan	Local	Yes	
GM-Align (Xu et al. 2019)	Name	GCN	Concatenation	Name	Sharing	Cosine	Local	No	
HMAN (Yang et al. 2019)	Random	GCN	Concatenation	Description Attribute	Sharing	Euclidean	Local	No	
JAPE (Sun et al. 2017)	Random	TransE	Concatenation	Attribute	Sharing	Cosine	Local	No	
GCN-Align (Wang et al. 2018)	Random	GCN	Concatenation	Attribute	Sharing	Manhattan	Local	No	
SHEA (Yan et al. 2021)	Name	GAT	Concatenation	Relation Name	Sharing	Euclidean	Local	No	
JTMEA (Lu et al. 2021)	Random	TransH	No	Attribute	Transformation	Euclidean	Local	Yes	
EASY (Ge et al. 2021)	Name	GMD	Concatenation	Name	Sharing	Manhattan	Local	Yes	
ESEA (Jiang et al. 2022a)	Random	TransE	Collaboration	Relation	Sharing	Cosine Edit	Local	No	
DuGa-DIT (Xie et al. 2022)	Random Name	GAT	Collaboration	Context	Sharing	Manhattan	Local	Yes	
SelfKG (Liu et al. 2022b)	LaBSE	GAT	No	No	Sharing	Euclidean	Local	No	

cross-graph matching, the other models learn their respective structural representations independently.

11.3.3 Comparative analysis of models on SRPRS

Table 17 shows the performance of each model on the SRPRS dataset. We classify the models into four categories and then the classification comparison is performed as follows.

Category 1 compared to DBP15K and DWY100K, the SRPRS dataset is sparser. The SRPRS dataset is closer to the realistic dataset. By observing Table 17, we can find that the performance of each model degrades significantly on the sparse dataset. However, the performance of SEU still ranks first, and SEU uses word vector embedding as well as character embedding in the embedding part. The SRPRS dataset contains a large number of proper names, and the performance of the CEA and CEAF models also remains high, with Hits@1 on DBP–WD and DBP–YG achieving almost exactly correct predictions. This indicates that both the collective search algorithm and the task assignment algorithm are applicable to sparse datasets.

Category 2 HGCN and RDGCN both utilize GCN for structural embedding and they both use entity names and relations. So the performance of the two models is close. We can find the same sparse datasets, but these two models perform a bit better on the sparse dataset of a single language, which proves that the heterogeneity of language is a bigger obstacle to the EA task compared with the sparse knowledge graph.

Category 3 in the EA models using only structural information, the model trends are close to those of DBP15K and DWY100K. The models using the iterative strategy perform a little better. In addition, the performance of RSNs on sparse datasets is close to that of BootEA using the iterative strategy, which indicates that long-term dependencies help to obtain more accurate representations when the structure of the knowledge graph is relatively sparse.

Category 4 the performance of GM-Align still maintains better results because GM-Align considers graph matching, incorporates local matching information between different knowledge graphs, and has less reliance on structure. NAEA, TransEdge, and MRAEA using the iterative strategy degrade significantly, but the performance of MRAEA degrades less than TransEdge and NAEA, which indicates that the bidirectional iterative strategy is more robust than the bootstrapping iterative strategy. On sparse datasets, the performance of JAPE and GCN-Align models, which also apply attributes, does not differ much. This indicates that when the structure of the knowledge graph is sparse, GCN cannot learn good structural representation. It is worth noting that COEA and FGWEA still maintain a better performance even on sparse datasets, with values above 0.9 on all metrics, because both models not only take into account structural and semantic information but also improve the alignment process. COEA converts entity alignment into a combination optimization problem, and FGWEA uses optimal transport to solve entity alignment.

11.3.4 Comparative analysis of models on DBP v1.1

Table 18 shows the performance of the model on the DBP v1.1 dataset. The model data marked with * is from reference (Ge et al. 2023), and the other model data is from reference (Xiang et al. 2021). By observing Table 18, we can find that models that only utilize structural information do not work well, such as the MTransE and RSNs

Table 15 Model performance comparison on *DBP*15*K* dataset

Category	Model	DBP15K								
		ZH–EN			JA–EN			FR–EN		
		Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
Global alignment	LatsEA (Chen et al. 2021c)	0.522	0.763	0.613	0.539	0.772	0.625	0.538	0.787	0.632
	CEA (Zeng et al. 2020)	0.776	–	–	0.853	–	–	0.967	–	–
	GM-EHD-JEA (Xu et al. 2020)	0.595	0.779	0.660	0.635	0.830	0.710	0.792	0.936	0.850
	RAGA (Zhu et al. 2021)	0.798	0.930	0.847	0.831	0.950	0.875	0.914	0.983	0.940
	CEAFF (Zeng et al. 2021)	0.811	–	–	0.868	–	–	0.972	–	–
	SEU (Mao et al. 2021)	0.900	0.965	0.924	0.956	0.991	0.969	0.988	0.999	0.992
	UED (Luo and Yu 2022)	0.826	0.943	0.870	0.863	0.960	0.900	0.938	0.987	0.957
	SoTead (Luo et al. 2022)	0.932	–	–	0.968	–	–	0.995	–	–
Noise filtering strategy	HGCN (Wu et al. 2019b)	0.720	0.857	0.768	0.766	0.897	0.814	0.892	0.961	0.917
	RDGCN (Wu et al. 2019a)	0.708	0.846	0.746	0.767	0.895	0.812	0.886	0.957	0.911
	RNM (Zhu et al. 2021)	0.840	0.919	0.870	0.872	0.944	0.899	0.938	0.981	0.954
	NMN (Wu et al. 2020)	0.733	0.869	0.781	0.785	0.912	0.827	0.902	0.967	0.924
	SSP (Nie et al. 2020)	0.739	0.925	0.808	0.721	0.935	0.800	0.739	0.947	0.818
	DAGCN (Wang et al. 2022)	0.692	0.866	–	0.758	0.901	–	0.910	0.974	–
Only structure	MTransE (Chen et al. 2017)	0.209	0.512	0.310	0.250	0.572	0.360	0.247	0.577	0.360
	MuGNN (Cao et al. 2019)	0.494	0.844	0.611	0.501	0.857	0.621	0.495	0.870	0.621
	KECG (Li et al. 2019)	0.477	0.835	0.598	0.489	0.844	0.610	0.486	0.851	0.610
	RSNs (Guo et al. 2019)	0.508	0.745	0.591	0.507	0.737	0.590	0.516	0.768	0.605
	BootEA Sun et al. (2018)	0.629	0.847	0.703	0.622	0.853	0.701	0.653	0.874	0.731

Table 15 (continued)

Category	Model	DBP15K								
		ZH–EN			JA–EN			FR–EN		
		Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
Structure and semantics	NAEA (Zhu et al. 2019)	0.650	0.867	0.720	0.641	0.872	0.718	0.673	0.894	0.752
	TransEdge (Sun et al. 2019)	0.735	0.919	0.801	0.719	0.932	0.795	0.710	0.941	0.796
	MRAEA (Mao et al. 2020b)	0.757	0.930	0.827	0.758	0.934	0.826	0.781	0.948	0.849
	GM-Align (Xu et al. 2019)	0.679	0.785	–	0.739	0.872	–	0.894	0.952	–
	HMAN (Yang et al. 2019)	0.561	0.859	0.670	0.557	0.860	0.670	0.550	0.876	0.660
	JAPE (Sun et al. 2017)	0.414	0.741	0.530	0.365	0.695	0.480	0.318	0.668	0.440
	GCN-Align (Wang et al. 2018)	0.434	0.762	0.550	0.427	0.762	0.540	0.411	0.772	0.530
	SHEA (Yan et al. 2021)	0.763	0.914	0.835	0.821	0.938	0.860	0.905	0.970	0.902
	JTMEA (Lu et al. 2021)	0.422	0.759	0.530	0.387	0.724	0.500	0.371	0.756	0.460
	ESEA (Jiang et al. 2022a)	0.725	0.869	0.775	0.750	0.878	0.795	0.794	0.909	0.834
	RpAlign (Huang et al. 2022)	0.748	0.889	0.794	0.730	0.890	0.782	0.752	0.900	0.801
	DuGa-DIT (Xie et al. 2022)	0.807	0.882	0.832	0.914	0.952	0.928	0.982	0.992	0.985
	FuzzyEA (Jiang et al. 2022b)	0.863	0.984	0.909	0.898	0.985	0.933	0.977	0.998	0.986
	SelfKG (Liu et al. 2022b)	0.829	0.919	–	0.890	0.953	–	0.959	0.992	–
	JETEA (Song et al. 2021)	0.427	0.750	–	0.364	0.724	–	0.365	0.718	–
	TTEA (Zhang et al. 2023)	0.802	0.938	0.852	0.831	0.954	0.876	0.924	0.986	0.947
	OTIEA (Zhang et al. 2023)	0.801	0.936	0.851	0.831	0.954	0.877	0.924	0.986	0.947

models. Although the BootEA model utilizes only structural information, it uses an iterative strategy to expand the size of the seed set, so it achieves better performance. Comparatively, combining structural and semantic information can improve the performance of the model. For example, the RDGCN model uses noise filtering strategy and relational semantics, and the OntoEA model introduces ontology information, both of which achieve better performance. TypeEA-B, TypeEA-R, and TypeEA-M apply type information to BootEA, RDGCN, and MultiKE, and through comparison, it is found that introducing entity type information facilitates entity alignment.

11.3.5 Comparative analysis of models on Chinese dataset

As shown in Table 19, representative models on the Chinese dataset include TransH (Wang et al. 2014), TransD (Ji et al. 2015), IEAJKE (Zhu et al. 2017), AttrE (Trisedya et al. 2019), and EASA (Huang and Luo 2020). In Dataset-1 and Dataset-2, EASA performs the best with Hits@1 and Hits@10 and is much ahead of the second-best performing model AttrE in terms of Hits values. The MR metric of EASA is second only to the top-ranked IEAJKE in Dataset-1 and Dataset-2, which proves the importance of making full use of the semantic information of entities. Although AttrE uses the frequency ratio of relationships and attributes as weights for EA, it cannot capture the importance of semantic aggregation generated by many attributes. TransH, TransD, and IEAJKE do not consider entity semantic integration and attribute weights, and thus perform below EASA and AttrE.

11.4 Comparison of multimodal EA models

Tables 20 and 21 show the experimental results of representative multimodal knowledge graph EA models on the FB15K–DB15K and FB15K–YAGO15K datasets. This paper compares the performance of each model on the datasets providing 20%, 50%, and 80% of seed entity pairs as training sets, respectively. The performance of each model gradually increases as the percentage of seed entity pairs increases. The fastest growth rate of each metric for MMEA and ACK-MMEA indicates that MMEA and ACK-MMEA have strong robustness and adaptability. By observing the experimental data, we find that PCMEA has the best results. PCMEA filters modal-specific noise and utilizes pseudo-labeling calibration methods and contrast learning, which can reduce the effect of noise and improve the quality of pseudo-labeling. The value of each metric of MMEA is significantly better than IKRL at each proportion of seed sets, which indicates that MMEA is applicable to real-world multimodal knowledge graphs. On the FB15K–DB15K dataset, for 80% of the seed entity pairs, MMEA outperforms HMEA with Hits@1 by nearly 20%. On the FB15K–YAGO15K dataset, for 80% of the seeded entity pairs, MMEA outperforms HMEA and GCN-Align by more than 15% in Hits@1 metrics. Even when the percentage of seeds is only 20%, MMEA shows better results compared to HMEA, with an improvement of about 14% for Hits@1 and over 15% for Hits@10. The above analysis shows that migrating multimodal knowledge embeddings from separate spaces to a common space is an effective method for EA.

Table 16 Model performance comparison on *DWY*100*K* dataset

Category	Model	DWY100K					
		DBP–WD			DBP–YG		
		Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
Global alignment	CEA (Zeng et al. 2020)	1.000	–	–	1.000	–	–
	GM-EHD-JEA (Xu et al. 2020)	0.863	0.922	0.892	0.783	0.823	0.802
Noise filtering strategy	HGCN (Wu et al. 2019b)	0.984	0.992	0.994	0.992	0.999	0.993
	NMN (Wu et al. 2020)	0.981	0.992	0.973	0.960	0.982	0.975
	SSP (Nie et al. 2020)	0.772	0.960	0.842	0.811	0.968	0.869
Only structure	MTransE (Chen et al. 2017)	0.238	0.507	0.330	0.227	0.414	0.290
	MuGNN (Cao et al. 2019)	0.604	0.894	0.701	0.739	0.937	0.810
	KECG (Li et al. 2019)	0.631	0.888	0.720	0.719	0.904	0.790
	RSNs (Guo et al. 2019)	0.607	0.793	0.673	0.689	0.878	0.756
	BootEA (Sun et al. 2018)	0.748	0.898	0.801	0.761	0.894	0.808
Structure and semantics	TransEdge (Sun et al. 2019)	0.788	0.938	0.824	0.792	0.936	0.832
	MRAEA (Mao et al. 2020b)	0.794	0.930	0.856	0.819	0.951	0.875
	GM-Align (Xu et al. 2019)	0.863	0.922	0.890	0.783	0.823	0.800
	HMAN (Yang et al. 2019)	0.655	0.897	0.740	0.776	0.938	0.830
	JAPE (Sun et al. 2017)	0.339	0.609	0.430	0.216	0.457	0.300
	GCN-Align (Wang et al. 2018)	0.513	0.777	0.610	0.596	0.837	0.680
	SHEA (Yan et al. 2021)	0.971	0.993	0.976	0.968	0.998	0.982
	JTMEA (Lu et al. 2021)	0.325	0.601	0.470	0.329	0.593	0.450
	RpAlign (Huang et al. 2022)	0.826	0.934	0.862	0.838	0.945	0.872
	SelfKG (Liu et al. 2022b)	0.983	0.998	–	1.000	1.000	–
	ESEA (Jiang et al. 2022a)	0.926	0.958	0.939	0.964	0.965	0.965
	EASAE (Munne and Ichise 2023)	0.920	0.978	–	0.909	0.966	–
	RALG (Zhang et al. 2023)	0.989	1.000	0.993	0.993	1.000	0.994

12 Research prospect

12.1 Study of other multimodal data

The relational structure information of knowledge graphs sometimes leads to ambiguity, so multimodal knowledge (Liu et al. 2019) has a key role in the knowledge embedding process. Although a few studies (Chen et al. 2020c; Wang et al. 2020) have applied

Table 17 Model performance comparison on *SRPRS* dataset

Category	Model	SRPRS											
		EN–FR			EN–DE			DBP–WD			DBP–YG		
		Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR	Hits@1	Hits@10	MRR
Global alignment	CEA (Zeng et al. 2020)	0.962	–	–	0.971	–	–	1.000	–	–	1.000	–	–
	GM-EHD-JEA (Xu et al. 2020)	0.574	0.646	0.605	0.681	0.748	0.713	–	–	–	–	–	–
	CEAFF (Zeng et al. 2021)	0.968	–	–	0.981	–	–	1.000	–	–	1.000	–	–
	SEU (Mao et al. 2021)	0.982	0.995	0.986	0.983	0.996	0.987	–	–	–	–	–	–
Noise filtering strategy	HGCN (Wu et al. 2019b)	0.670	0.770	0.713	0.763	0.863	0.807	0.989	0.999	0.996	0.991	0.997	0.994
	RDGCN (Wu et al. 2019a)	0.672	0.767	0.714	0.779	0.886	0.822	0.974	0.994	0.982	0.990	0.997	0.992
Only structure	MTransE (Chen et al. 2017)	0.213	0.447	0.290	0.107	0.248	0.160	0.188	0.382	0.260	0.196	0.401	0.270
	MuGNN (Cao et al. 2019)	0.131	0.342	0.208	0.245	0.431	0.310	0.151	0.366	0.220	0.175	0.381	0.240
	KECG (Li et al. 2019)	0.298	0.616	0.403	0.444	0.707	0.540	0.323	0.646	0.430	0.350	0.651	0.450
	RSNs (Guo et al. 2019)	0.350	0.636	0.440	0.484	0.729	0.570	0.391	0.663	0.480	0.393	0.665	0.490
	BootEA (Sun et al. 2018)	0.365	0.649	0.460	0.503	0.732	0.580	0.384	0.667	0.480	0.381	0.651	0.470
Structure and semantics	NAEA (Zhu et al. 2019)	0.177	0.416	0.260	0.307	0.535	0.390	0.182	0.429	0.260	0.195	0.451	0.280
	TransEdge (Sun et al. 2019)	0.400	0.675	0.490	0.556	0.753	0.630	0.461	0.738	0.560	0.443	0.699	0.530
	MRAEA (Mao et al. 2020b)	0.460	0.768	0.559	0.594	0.818	0.666	0.509	0.795	0.597	0.485	0.768	0.574
	GM-Align (Xu et al. 2019)	0.574	0.646	0.602	0.681	0.748	0.710	0.762	0.830	0.792	0.804	0.837	0.825
	HMAN (Yang et al. 2019)	0.401	0.705	0.500	0.528	0.778	0.620	0.433	0.744	0.549	0.461	0.765	0.566
	JAPE (Sun et al. 2017)	0.241	0.544	0.345	0.268	0.547	0.367	0.212	0.502	0.316	0.193	0.500	0.305
	GCN-Align (Wang et al. 2018)	0.243	0.522	0.340	0.385	0.600	0.460	0.291	0.556	0.380	0.319	0.586	0.410
	COEA (Lin et al. 2023)	0.986	0.997	0.990	0.987	0.996	0.991	–	–	–	–	–	–
	FGWEA (Tang et al. 2023)	0.996	0.999	–	0.997	1.000	–	–	–	–	–	–	–

Table 18 Model performance comparison on *DBPv*1.1 dataset

Category	Model	DBPv1.1											
		EN–FR 15k(v1)			EN–FR 15k(v2)			D–W 15k(v1)			D–W 15k(v2)		
		Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR	Hits@1	Hits@5	MRR
Noise filtering strategy	RDGCN (Wu et al. 2019a)	0.755	0.854	0.800	0.847	0.919	0.880	0.515	0.669	0.584	0.623	0.757	0.684
Only structure	MTransE (Chen et al. 2017)	0.247	0.467	0.351	0.240	0.436	0.336	0.259	0.461	0.354	0.271	0.490	0.376
	SEA (Pei et al. 2019a)	0.280	0.530	0.397	0.360	0.651	0.494	0.360	0.572	0.458	0.567	0.770	0.660
	BootEA (Sun et al. 2018)	0.507	0.718	0.603	0.660	0.850	0.745	0.572	0.744	0.649	0.821	0.926	0.867
	RSNs (Guo et al. 2019)	0.393	0.595	0.487	0.579	0.759	0.662	0.441	0.615	0.521	0.723	0.854	0.782
Structure and semantics	JAPE (Sun et al. 2017)	0.262	0.497	0.372	0.292	0.524	0.402	0.250	0.457	0.348	0.262	0.484	0.368
	GCN-Align (Wang et al. 2018)	0.338	0.589	0.451	0.414	0.698	0.542	0.364	0.580	0.461	0.506	0.743	0.612
	AliNet (Sun et al. 2020)	0.258	0.437	0.339	0.359	0.569	0.453	0.270	0.403	0.331	0.522	0.698	0.601
	AttrE (Trisedya et al. 2019)	0.481	0.671	0.569	0.535	0.746	0.631	0.299	0.467	0.381	0.489	0.695	0.585
	MultiKE (Zhang et al. 2019)	0.749	0.819	0.782	0.864	0.909	0.885	0.411	0.521	0.468	0.495	0.646	0.569
	OntoEA (Xiang et al. 2021)	0.797	0.871	0.832	0.901	0.981	0.931	0.553	0.787	0.656	0.795	0.943	0.860
	TypeEA-B* (Ge et al. 2023)	0.543	0.769	0.643	0.882	0.939	0.909	0.608	0.771	0.681	0.830	0.940	0.889
	TypeEA-R* (Ge et al. 2023)	0.794	0.879	0.832	0.906	0.959	0.930	0.590	0.738	0.656	0.668	0.809	0.729
	TypeEA-M* (Ge et al. 2023)	0.796	0.861	0.827	0.891	0.929	0.908	0.455	0.589	0.522	0.534	0.703	0.612

Table 19 Model performance comparison on Chinese dataset

Model	Dataset-1			Dataset-2		
	Hits@1	Hits@10	MR	Hits@1	Hits@10	MR
TransH (Wang et al. 2014)	0.816	0.913	0.133	0.829	0.900	0.657
TransD (Ji et al. 2015)	0.823	0.917	0.143	0.847	0.901	0.572
IEAJKE (Zhu et al. 2017)	0.823	0.933	0.084	0.856	0.924	0.342
AttrE (Trisedya et al. 2019)	0.846	0.932	0.471	0.875	0.926	2.100
EASA (Huang and Luo 2020)	0.901	0.949	0.130	0.933	0.964	0.402

Table 20 Model performance comparison analysis in dataset FB15K–DB15K

Model	20%		50%		80%	
	Hits@1	Hits@10	Hits@1	Hits@10	Hits@1	Hits@10
PoE (Liu et al. 2019)	11.10	17.80	23.50	33.00	34.40	40.60
GCN-Align (Wang et al. 2018)	5.35	17.11	13.85	34.31	22.18	48.95
IKRL (Xie et al. 2017)	1.01	2.40	2.77	5.79	5.41	11.09
HMEA (Guo et al. 2021)	12.65	36.86	26.23	58.08	41.68	78.55
MMEA (Chen et al. 2020c)	26.48	54.11	41.65	70.35	59.03	86.87
PCMEA (Wang et al. 2024)	67.63	88.72	73.75	91.54	82.04	96.44
ACK-MMEA (Li et al. 2023)	30.40	54.90	56.00	73.60	68.20	87.40

Table 21 Model performance comparison analysis in dataset FB15K–YAGO15K

Model	20%		50%		80%	
	Hits@1	Hits@10	Hits@1	Hits@10	Hits@1	Hits@10
PoE (Liu et al. 2019)	8.70	13.30	18.50	24.70	28.90	34.30
GCN-Align (Wang et al. 2018)	6.76	17.99	16.47	35.85	28.75	53.05
IKRL (Xie et al. 2017)	0.86	1.75	1.95	3.73	3.57	7.14
HMEA (Guo et al. 2021)	10.51	31.27	26.50	58.08	43.30	80.11
MMEA (Chen et al. 2020c)	23.39	48.00	40.26	64.51	59.76	83.89
PCMEA (Wang et al. 2024)	58.96	83.47	67.02	88.57	75.56	94.24
ACK-MMEA (Li et al. 2023)	28.90	49.60	53.50	69.90	67.60	86.40

static image data to multimodal EA, other multimodal data have not been fully explored. For example, dynamic video data have not been applied to EA tasks. Video data contains more rich information and conveys more intuitive information compared with static image data. Therefore, how to incorporate advanced video features as well as other features into multimodal knowledge graph EA may be the focus of future research.

12.2 Study of realistic datasets

There are significant differences between existing datasets and real-world knowledge graphs. This makes it difficult for existing EA models to run on real-world knowledge graphs. The current datasets have more neighboring entities and contain more semantic information, thus these high-level entities are relatively easy to align. In addition, the current dataset focuses on only one aspect of heterogeneity, such as multilingualism, and ignores pattern and scale differences. Therefore, considering multiple perspectives to create datasets that are closer to real-world knowledge graphs deserves further research.

12.3 Study of other vector spaces

A key step in the embedding-based knowledge graph EA model is to learn the embedding representation, and the quality of the embedding has a direct impact on the subsequent EA performance. It has been shown that the non-Euclidean space has better graph structure embedding performance than the Euclidean space (Nickel and Kiela 2017). Most of the current EA methods use the Euclidean space. Although some models use hyperbolic spaces (Guo et al. 2021) and spherical space (Huang et al. 2022), many other vector spaces (e.g., complex spaces Sun et al. 2019) are still worth studying.

12.4 Study of dynamic knowledge graph EA

At present, most datasets applied in the field of EA are static knowledge graphs. While realistic knowledge graphs are frequently changing, so it is necessary to consider dynamic factors in the process of EA of knowledge graphs. The dynamics of dynamic knowledge graphs are mainly reflected in two dimensions of time and space (Zheng et al. 2020). Although some scholars have studied the EA of temporal knowledge graphs (Song et al. 2022), no scholars have considered the dynamics of knowledge graphs from the spatial dimension. Therefore, it is worthwhile to design spatial–temporal knowledge graph EA models from both temporal and spatial dimensions.

Acknowledgements Thanks to all the authors for their hard work.

Author contributions Beibei Zhu: collect literature, write the first draft of the article, draw graphs and tables, and revise the article; Ruolin Wang: collect literature, write the first draft of the article, and revise the article; Junyi Wang: check the article layout, write the first draft of the article, and revise the article; Fei Shao: collect literature, write the first draft of the article, and revise the article; Kerun Wang: write the first draft of the article and revise the article.

Funding This work is supported by Research Initiation Program for New PhDs at Liaoning Normal University.

Data availability The data analysed during the current study are included in this paper.

Code availability There are no experiments or code involved in this paper.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Competing interests The authors declare no competing interests.

References

Bernerslee T, Hendler J, Lassila O (2001) The semantic web. Sci Am 284(5):34–43

Bordes A, Usunier N, García-Durán A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: Paper presented at 27th annual conference on neural information processing systems 2013, Lake Tahoe, Nevada, United States, 5–8 December 2013

Cai W, Ma W, Wei L, Jiang Y (2023) Semi-supervised entity alignment via relation-based adaptive neighborhood matching. IEEE Trans Knowl Data Eng 35(8):8545–8558

Cao Y, Liu Z, Li C, Liu Z, Li J, Chua T (2019) Multi-channel graph neural network for entity alignment. In: Paper presented at the 57th conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019

Chaurasiya D, Surisetty A, Kumar N, Singh A, Dey V, Malhotra A, Dhama G, Arora A (2022) Entity alignment for knowledge graphs: progress, challenges, and empirical studies. CoRR abs/2205.08777. arxiv:2205.08777

Chen M, Tian Y, Yang M, Zaniolo C (2017) Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. In: Paper presented at the 26th international joint conference on artificial intelligence, IJCAI 2017, Melbourne, Australia, 19–25 August 2017

Chen M, Tian Y, Chang K, Skiena S, Zaniolo C (2018) Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. In: Paper presented at the 27th international joint conference on artificial intelligence, IJCAI 2018, Stockholm, Sweden, 13–19 July 2018

Chen M, Shi W, Zhou B, Roth D (2020a) Cross-lingual entity alignment for knowledge graphs with incidental supervision from free text (Preprint). arxiv:2005.00171

Chen X, Wang L, Tang Y, Han W, Gu Z (2020b) Seeds optimization for entity alignment in knowledge graph embedding. In: Paper presented at the 5th IEEE international conference on data science in cyberspace, DSC 2020, Hong Kong, 27–30 July 2020

Chen L, Li Z, Wang Y, Xu T, Wang Z, Chen E (2020c) MMEA: entity alignment for multi-modal knowledge graph. In: Paper presented at the 13th international conference, KSEM 2020, Hangzhou, China, proceedings, Part I, 28–30 August 2020

Chen B, Zhang J, Tang X, Chen H, Li C (2020d) JarKA: modeling attribute interactions for cross-lingual knowledge alignment. In: Paper presented at the 24th Pacific–Asia conference, PAKDD 2020, Singapore, 11–14 May 2020

Chen J, Gu B, Li Z, Zhao P, Liu A, Zhao L (2020e) SAEA: self-attentive heterogeneous sequence learning model for entity alignment. In: Paper presented at the 25th international conference, DASFAA 2020, proceedings, Part I, Jeju, South Korea, 24–27 September 2020

Chen J, Li Z, Zhao P, Liu A, Zhao L, Chen Z, Zhang X (2020f) Learning short-term differences and long-term dependencies for entity alignment. In: Paper presented at the 19th international semantic web conference, Athens, Greece, 2–6 November

Chen L, Tian X, Tang X, Cui J (2021a) Multi-information embedding based entity alignment. Appl Intell 51(12):8896–8912. https://doi.org/10.1007/s10489-021-02400-8

Chen L, Tian X, Tang X, Cui J (2021b) Multi-information embedding based entity alignment. Appl Intell 51(12):8896–8912. https://doi.org/10.1007/s10489-021-02400-8

Chen W, Chen X, Xiong S (2021c) Global entity alignment with gated latent space neighborhood aggregation. In: Paper presented at the 20th China national conference, CCL 2021, Hohhot, China, 13–15 August 2021. https://doi.org/10.1007/978-3-030-84186-7_25 de Assis Costa G, de Oliveira JMP (2018) Linguistic frames as support for entity alignment in knowledge graphs. In: Paper presented at the 20th international conference on information integration and web-based applications and services, iiWAS 2018, Yogyakarta, Indonesia, 19–21 November 2018

Dong X, Gabrilovich E, Heitz G, Horn W, Lao N, Murphy K, Strohmann T, Sun S, Zhang W (2014) Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In: Paper presented at the 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '14, New York, NY, USA, 24–27 August 2014

Fanourakis N, Efthymiou V, Kotzinos D, Christophides V (2022) Knowledge graph embedding methods for entity alignment: an experimental review. CoRR. arxiv:2203.09280

Ferrada S, Bustos B, Hogan A (2017) IMGpedia: a linked dataset with content-based analysis of Wikimedia images. In: Paper presented at the 16th international semantic web conference, Vienna, Austria, 21–25 October 2017

Fey M, Lenssen JE, Morris C, Masci J, Kriege NM (2020) Deep graph matching consensus. In: Paper presented at the 8th international conference on learning representations, ICLR 2020, Addis Ababa, Ethiopia, 26–30 April 2020

Ge C, Liu X, Chen L, Zheng B, Gao Y (2021) Make it easy: an effective end-to-end entity alignment framework. In: Paper presented at the 44th international ACM SIGIR conference on research and development in information retrieval, virtual event, Canada, 11–15 July 2021

Ge X, Wang YC, Wang B, Kuo C-CJ (2023) TypeEA: type-associated embedding for knowledge graph entity alignment. APSIPA Trans Signal Inf Process 12(1):1–18

Grover A, Leskovec J (2016) Node2Vec: scalable feature learning for networks. In: Paper presented at the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, San Francisco, CA, USA, 13–17 August 2016

Guan S, Jin X, Wang Y, Jia Y, Shen H, Li Z, Cheng X (2019) Self-learning and embedding based entity alignment. Knowl Inf Syst 59(2):361–386. https://doi.org/10.1007/s10115-018-1191-0

Guo L, Sun Z, Hu W (2019) Learning to exploit long-term relational dependencies in knowledge graphs. In: Paper presented at the 36th international conference on machine learning, ICML, Long Beach, California, USA, 9–15 June 2019

Guo H, Tang J, Zeng W, Zhao X, Liu L (2021) Multi-modal entity alignment in hyperbolic space. Neurocomputing 461:598–607. https://doi.org/10.1016/j.neucom.2021.03.132

Guo L, Han Y, Zhang Q, Chen H (2022) Deep reinforcement learning for entity alignment. In: Paper presented at the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, 22–27 May 2022

Haihong E, Cheng R, Song M, Zhu P, Wang Z (2020) A joint embedding method of relations and attributes for entity alignment. Int J Mach Learn Comput 10(5):605–611

Hao Y, Zhang Y, He S, Liu K, Zhao J (2016) A joint embedding method for entity alignment of knowledge bases. In: Paper presented at knowledge graph and semantic computing: semantic, knowledge, and linked big data—first China conference, CCKS 2016, Beijing, China, 19–22 September 2016

He S, Liu K, Ji G, Zhao J (2015) Learning to represent knowledge graphs with Gaussian embedding. In: Paper presented at the 24th ACM international conference on information and knowledge management, CIKM 2015, Melbourne, VIC, Australia, 19–23 October 2015

He F, Li Z, Yang Q, Liu A, Liu G, Zhao P, Zhao L, Zhang M, Chen Z (2019) Unsupervised entity alignment using attribute triples and relation triples. In: Paper presented at the 24th international conference, DASFAA 2019, Chiang Mai, Thailand, 22–25 April 2019

Huang L, Luo X (2020) EASA: entity alignment algorithm based on semantic aggregation and attribute attention. IEEE Access 8:18162–18170

Huang H, Li C, Peng X, He L, Guo S, Peng H, Wang L, Li J (2022) Cross-knowledge-graph entity alignment via relation prediction. Knowl Based Syst 240:107813. https://doi.org/10.1016/j.knosys.2021.107813

Ji G, He S, Xu L, Liu K, Zhao J (2015) Knowledge graph embedding via dynamic mapping matrix. In: Paper presented at the 53rd annual meeting of the Association for Computational Linguistics and the 7th international joint conference on natural language processing of the Asian Federation of Natural Language Processing, ACL 2015, Beijing, China, 26–31 July 2015

Jiang T, Bu C, Zhu Y, Wu X (2019) Two-stage entity alignment: combining hybrid knowledge graph embedding with similarity-based relation alignment. In: Paper presented at the 16th Pacific Rim international conference on artificial intelligence, Cuvu, Yanuca Island, Fiji, proceedings, Part I, 26–30 August 2019

Jiang T, Bu C, Zhu Y, Wu X (2022a) Combining embedding-based and symbol-based methods for entity alignment. Pattern Recognit 124:108433. https://doi.org/10.1016/j.patcog.2021.108433

Jiang W, Liu Y, Deng X (2022b) Fuzzy entity alignment via knowledge embedding with awareness of uncertainty measure. Neurocomputing 468:97–110. https://doi.org/10.1016/j.neucom.2021.10.026

Kang S, Ji L, Li Z, Hao X, Ding Y (2020) Iterative cross-lingual entity alignment based on TransC. IEICE Trans Inf Syst 103–D(5):1002–1005. https://doi.org/10.1587/transinf.2019DAL0001

Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: Paper presented at the 5th international conference on learning representations, ICLR 2017, conference track proceedings, Toulon, France, 24–26 April 2017

Li S, Li X, Ye R, Wang M, Su H, Ou Y (2018) Non-translational alignment for multi-relational networks. In: Paper presented at the 27th international joint conference on artificial intelligence, IJCAI 2018, Stockholm, Sweden, 13–19 July 2018

Li C, Cao Y, Hou L, Shi J, Li J, Chua T (2019) Semi-supervised entity alignment via joint knowledge embedding model and cross-graph model. In: Paper presented at the 9th international joint conference on natural language processing, EMNLP–IJCNLP 2019, Hong Kong, China, 3–7 November 2019

Li Q, Guo S, Luo Y, Ji C, Wang L, Sheng J, Li J (2023) Attribute-consistent knowledge graph representation learning for multi-modal entity alignment. In: Paper presented at proceedings of the ACM web conference 2023, WWW 2023, Austin, TX, USA, 30 April–4 May 2023

Lin Y, Liu Z, Sun M, Liu Y, Zhu X (2015a) Learning entity and relation embeddings for knowledge graph completion. In: Paper presented at the 29th AAAI conference on artificial intelligence, Austin, Texas, USA, 25–30 January 2015

Lin Y, Liu Z, Luan H, Sun M, Rao S, Liu S (2015b) Modeling relation paths for representation learning of knowledge bases. In: Paper presented at the 2015 conference on empirical methods in natural language processing, EMNLP 2015, Lisbon, Portugal, 17–21 September 2015

Lin X, Yang H, Wu J, Zhou C, Wang B (2019) Guiding cross-lingual entity alignment via adversarial knowledge embedding. In: Paper presented at 2019 IEEE international conference on data mining, ICDM 2019, Beijing, China, 8–11 November 2019

Lin L, Liu J, Lv Y, Guo F (2020) A similarity model based on reinforcement local maximum connected same destination structure oriented to disordered fusion of knowledge graphs. Appl Intell 50(9):2867–2886. https://doi.org/10.1007/s10489-020-01673-9

Lin X, EH, Song W, Luo H (2021) EchoEA: echo information between entities and relations for entity alignment (Preprint). arxiv:2107.03054

Lin L, Zu L, Guo F, Fu S, Lv Y, Guo H, Liu J (2023) Using combinatorial optimization to solve entity alignment: an efficient unsupervised model. Neurocomputing 558:126802–126815

Liu Q, Li Y, Duan H, Liu Y, Qin Z (2016) Knowledge graph construction techniques. J Comput Res Dev 53(3):582–600

Liu Y, Li H, García-Durán A, Niepert M, Oñoro-Rubio D, Rosenblum DS (2019) MMKG: multi-modal knowledge graphs. In: Paper presented at the 16th international conference, ESWC 2019, Portorož, Slovenia, 2–6 June 2019

Liu Z, Cao Y, Pan L, Li J, Chua T (2020) Exploring and evaluating attributes, values, and structures for entity alignment. In: Paper presented at the 2020 conference on empirical methods in natural language processing, EMNLP 2020, online, 16–20 November 2020

Liu F, Chen M, Roth D, Collier N (2021) Visual pivoting for (unsupervised) entity alignment. In: Paper presented at the eleventh symposium on educational advances in artificial intelligence, EAAI 2021, virtual event, 2–9 February 2021

Liu J, Sun Z, Hooi B, Wang Y, Liu D, Yang B, Xiao X, Chen M (2022a) Dangling-aware entity alignment with mixed high-order proximities. CoRR abs/2205.02406. arxiv:2205.02406

Liu X, Hong H, Wang X, Chen Z, Kharlamov E, Dong Y, Tang J (2022b) SelfKG: self-supervised entity alignment in knowledge graphs. In: Paper presented at the ACM web conference 2022, virtual event, Lyon, France, 25–29 April 2022

Lu G, Zhang L, Jin M, Li P, Huang X (2021) Entity alignment via knowledge embedding and type matching constraints for knowledge graph inference. J Ambient Intell Humaniz Comput 4:1–11

Luo S, Yu S (2022) An accurate unsupervised method for joint entity alignment and dangling entity detection. In: Paper presented at the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, 22–27 May 2022

Luo S, Cheng P, Yu S (2022) Semi-constraint optimal transport for entity alignment with dangling cases. CoRR abs/2203.05744. arxiv:2203.05744

Lv X, Hou L, Li J, Liu Z (2018) Differentiating concepts and instances for knowledge graph embedding. In: Paper presented at the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, 31 October–4 November 2018

Mao X, Wang W, Xu H, Wu Y, Lan M (2020a) Relational reflection entity alignment. In: Paper presented at the 29th ACM international conference on information and knowledge management, virtual event, Ireland, 19–23 October 2020

Mao X, Wang W, Xu H, Lan M, Wu Y (2020b) MRAEA: an efficient and robust entity alignment approach for cross-lingual knowledge graph. In: Paper presented at the 13th ACM international conference on web search and data mining, Houston, TX, USA, 3–7 February 2020

Mao X, Wang W, Wu Y, Lan M (2021) From alignment to assignment: frustratingly simple unsupervised entity alignment. In: Paper presented at the 2021 conference on empirical methods in natural language processing, EMNLP 2021, virtual event, Punta Cana, Dominican Republic, 7–11 November 2021

Mao X, Ma M, Yuan H, Zhu J, Wang Z, Xie R, Wu W, Lan M (2022) An effective and efficient entity alignment decoding algorithm via third-order tensor isomorphism. In: Paper presented at the 60th annual meeting of the Association for Computational Linguistics: long papers, ACL 2022, Dublin, Ireland, 22–27 May 2022, vol 1

Mishra S, Saha S, Mondal S (2017) GAEMTBD: genetic algorithm based entity matching techniques for bibliographic databases. Appl Intell 47(1):197–230. https://doi.org/10.1007/s10489-016-0874-z

Munne RF, Ichise R (2020) Joint entity summary and attribute embeddings for entity alignment between knowledge graphs. In: Paper presented at the 15th international conference, HAIS 2020, Gijón, Spain, 11–13 November 2020

Munne RF, Ichise R (2023) Entity alignment via summary and attribute embeddings. Log J IGPL 31(2):314–324

Ni W, Xu Q, Jiang Y, Cao Z, Cao X, Huang Q (2023) PSNEA: pseudo-Siamese network for entity alignment between multi-modal knowledge graphs. In: Paper presented at the 31st ACM international conference on multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023–3 November 2023

Nickel M, Kiela D (2017) Poincaré embeddings for learning hierarchical representations. In: Paper presented at annual conference on neural information processing systems 2017, Long Beach, CA, USA, 4–9 December 2017

Nickel M, Tresp V, Kriegel H (2011) A three-way model for collective learning on multi-relational data. In: Paper presented at the 28th international conference on machine learning, ICML 2011, Bellevue, Washington, USA, 28 June–2 July 2011

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: Paper presented at the 29th international joint conference on artificial intelligence, IJCAI 2020, 2020

Pang N, Zeng W, Tang J, Tan Z, Zhao X (2019) Iterative entity alignment with improved neural attribute embedding. In: Paper presented at the 16th extended semantic web conference 2019 (ESWC 2019), Portoroz, Slovenia, 2 June 2019

Pei S, Yu L, Hoehndorf R, Zhang X (2019a) Semi-supervised entity alignment via knowledge graph embedding with awareness of degree difference. In: Paper presented at the World Wide Web Conference, WWW 2019, San Francisco, CA, USA, 13–17 May 2019

Pei S, Yu L, Hoehndorf R, Zhang X (2019b) Semi-supervised entity alignment via knowledge graph embedding with awareness of degree difference. In: Paper presented at the World Wide Web conference, WWW 2019, San Francisco, CA, USA, 13–17 May 2019

Pei S, Yu L, Yu G, Zhang X (2020) REA: robust cross-lingual entity alignment between knowledge graphs. In: Paper presented at the 26th ACM SIGKDD conference on knowledge discovery and data mining, virtual event, CA, USA, 23–27 August 2020

Perozzi B, Al-Rfou R, Skiena S (2014) DeepWalk: online learning of social representations. In: Paper presented at the 20th ACM SIGKDD international conference on knowledge discovery and data mining, KDD '14, New York, NY, USA, 24–27 August 2014

Qu M, Tang J, Bengio Y (2019) Weakly-supervised knowledge graph alignment with adversarial learning (Preprint). arxiv:1907.03179

Shen L, He R, Huang S (2022) Entity alignment with adaptive margin learning knowledge graph embedding. Data Knowl Eng 139:101987. https://doi.org/10.1016/j.datak.2022.101987

Shi X, Xiao Y (2019) Modeling multi-mapping relations for precise cross-lingual entity alignment. In: Paper presented at the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, EMNLP–IJCNLP 2019, Hong Kong, China, 3–7 November 2019

Shize K, Lixin J, Shuxin L, Yuehang D (2019) Cross-lingual entity alignment model based on the similarities of entity descriptions and knowledge embeddings. Acta Electron Sin 47(9):1841–1847

Singh H, Jain P, Mausam Chakrabarti S (2021) Multilingual knowledge graph completion with joint relation and entity alignment (Preprint). arxiv:2104.08804

Song X, Zhang H, Bai L (2021) Entity alignment between knowledge graphs using entity type matching. In: Paper presented at the 14th international conference, KSEM 2021, Tokyo, Japan, 14–16 August 2021

Song X, Bai L, Liu R, Zhang H (2022) Temporal knowledge graph entity alignment via representation learning. In: Paper presented at the 27th international conference, DASFAA 2022, virtual event, 11–14 April 2022

Sun Z, Hu W, Li C (2017) Cross-lingual entity alignment via joint attribute-preserving embedding. In: Paper presented at the 16th international semantic web conference, Vienna, Austria, 21–25 October 2017

Sun Z, Hu W, Zhang Q, Qu Y (2018) Bootstrapping entity alignment with knowledge graph embedding. In: Paper presented at the 27th international joint conference on artificial intelligence, IJCAI 2018, Stockholm, Sweden, 13–19 July 2018

Sun Z, Deng Z, Nie J, Tang J (2019a) RotatE: knowledge graph embedding by relational rotation in complex space. In: Paper presented at the 7th international conference on learning representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019

Sun Z, Huang J, Hu W, Chen M, Guo L, Qu Y (2019b) TransEdge: translating relation-contextualized embeddings for knowledge graphs. In: Paper presented at the 18th international semantic web conference, Auckland, New Zealand, 26–30 October 2019

Sun Z, Zhang Q, Hu W, Wang C, Chen M, Akrami F, Li C (2020a) A benchmarking study of embedding-based entity alignment for knowledge graphs. Proc VLDB Endow 13(11):2326–2340

Sun J, Zhou Y, Zong C (2020b) Dual attention network for cross-lingual entity alignment. In: Paper presented at the 28th international conference on computational linguistics, COLING 2020 (online), Barcelona, Spain, 8–13 December 2020

Sun Z, Chen M, Hu W, Wang C, Dai J, Zhang W (2020c) Knowledge association with hyperbolic knowledge graph embeddings. In: Paper presented at the 2020 conference on empirical methods in natural language processing, EMNLP 2020, online, 16–20 November 2020

Sun Z, Wang C, Hu W, Chen M, Dai J, Zhang W, Qu Y (2020d) Knowledge graph alignment network with gated multi-hop neighborhood aggregation. In: Paper presented at the 10th AAAI symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020

Sun T, Zhai J, Wang Q (2020e) NovEA: a novel model of entity alignment using attribute triples and relation triples. In: Paper presented at the 13th international conference, KSEM 2020, Hangzhou, China, 28–30 August 2020

Tam NT, Trung HT, Yin H, Vinh TV, Sakong D, Zheng B, Hung NQV (2021) Entity alignment for knowledge graphs with multi-order convolutional networks (extended abstract). In: Paper presented at the 37th IEEE international conference on data engineering, ICDE 2021, Chania, Greece, 19–22 April 2021

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: Paper presented at the 29th international joint conference on artificial intelligence, IJCAI 2020, 2020

Tang J, Zhao K, Li J (2023) A fused Gromov–Wasserstein framework for unsupervised knowledge graph entity alignment. In: Paper presented at findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, 9–14 July 2023

Trisedya BD, Qi J, Zhang R (2019a) Entity alignment between knowledge graphs using attribute embeddings. In: Paper presented at the 9th AAAI symposium on educational advances in artificial intelligence, EAAI 2019, Honolulu, Hawaii, USA, 27 January–1 February 2019

Trisedya BD, Qi J, Zhang R (2019b) Entity alignment between knowledge graphs using attribute embeddings. In: Paper presented at the 33rd AAAI conference on artificial intelligence, AAAI 2019, the thirty-first innovative applications of artificial intelligence conference, IAAI 2019, the 9th AAAI symposium on educational advances in artificial intelligence, EAAI 2019, Honolulu, Hawaii, USA, 27 January–1 February 2019

Trivedi R, Sisman B, Dong XL, Faloutsos C, Ma J, Zha H (2018) LinkNBed: multi-graph representation learning with entity linkage. In: Paper presented at the 56th annual meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018

Velickovic P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y (2018) Graph attention networks. In: Paper presented at the 6th international conference on learning representations, ICLR 2018, Vancouver, BC, Canada, 30 April–3 May 2018

Vrandecic D, Krötzsch M (2014) Wikidata: a free collaborative knowledgebase. Commun ACM 57(10):78–85. https://doi.org/10.1145/2629489

Wang Z, Zhang J, Feng J, Chen Z (2014) Knowledge graph embedding by translating on hyperplanes. In: Paper presented at the 28th AAAI conference on artificial intelligence, Québec City, Québec, Canada, 27–1 July 2014

Wang Z, Lv Q, Lan X, Zhang Y (2018a) Cross-lingual knowledge graph alignment via graph convolutional networks. In: Paper presented at the 2018 conference on empirical methods in natural language processing, Brussels, Belgium, 31 October–4 November 2018

Wang L, Bhagavatula C, Neumann M, Lo K, Wilhelm C, Ammar W (2018b) Ontology alignment in the biomedical domain using entity definitions and context. In: Paper presented at proceedings of the BioNLP 2018 workshop, Melbourne, Australia, 19 July 2018

Wang M, Qi G, Wang H, Zheng Q (2019) Richpedia: a comprehensive multi-modal knowledge graph. In: Paper presented at the 9th joint international conference, JIST 2019, Hangzhou, China, 25–27 November 2019

Wang H, Lun B, Zhang X, Sun X (2020a) Multimodal entity alignment based on joint knowledge representation learning. Control Decis 35(12):10

Wang Z, Yang J, Ye X (2020b) Knowledge graph alignment with entity-pair embedding. In: Paper presented at the 2020 conference on empirical methods in natural language processing, EMNLP 2020, online, 16–20 November 2020

Wang H, Wang Y, Li J, Luo T (2022) Degree aware based adversarial graph convolutional networks for entity alignment in heterogeneous knowledge graph. Neurocomputing 487:99–109. https://doi.org/10.1016/j.neucom.2022.02.002

Wang C, Huang Z, Wan Y, Wei J, Zhao J, Wang P (2023) FuAlign: cross-lingual entity alignment via multi-view representation learning of fused knowledge graphs. Inf Fusion 89:41–52

Wang L, Qi P, Bao X, Zhou C, Qin B (2024) Pseudo-label calibration semi-supervised multi-modal entity alignment. In: Paper presented at the 38th AAAI conference on artificial intelligence, AAAI 2024, 36th conference on innovative applications of artificial intelligence, IAAI 2024, 14th symposium on educational advances in artificial intelligence, EAAI 2024, Vancouver, Canada, 20–27 February 2024

Wei-Xin Z, Xiang Z, Jiu-Yang TZT (2020) Iterative entity alignment via re-ranking. J Comput Res Dev 57(7):12

Wu Y, Liu X, Feng Y, Wang Z, Yan R, Zhao D (2019a) Relation-aware entity alignment for heterogeneous knowledge graphs. In: Paper presented at the 28th international joint conference on artificial intelligence, IJCAI 2019, Macao, China, 10–16 August 2019

Wu Y, Liu X, Feng Y, Wang Z, Zhao D (2019b) Jointly learning entity and relation representations for entity alignment. In: Paper presented at the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, EMNLP–IJCNLP 2019, Hong Kong, China, 3–7 November 2019

Wu Y, Liu X, Feng Y, Wang Z, Zhao D (2020) Neighborhood matching network for entity alignment. In: Paper presented at the 58th annual meeting of the Association for Computational Linguistics, ACL 2020, online, 5–10 July 2020

Xiang Y, Zhang Z, Chen J, Chen X, Lin Z, Zheng Y (2021) OntoEA: ontology-guided entity alignment via joint knowledge graph embedding. In: Paper presented at findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, online event, 1–6 August 2021

Xie R, Liu Z, Luan H, Sun M (2017) Image-embodied knowledge representation learning. In: Paper presented at the 26th international joint conference on artificial intelligence, IJCAI 2017, Melbourne, Australia, 19–25 August 2017

Xie Z, Zhu R, Zhao K, Liu J, Zhou G, Huang JX (2020) A contextual alignment enhanced cross graph attention network for cross-lingual entity alignment. In: Paper presented at the 28th international conference on computational linguistics, COLING 2020 (online), Barcelona, Spain, 8–13 December 2020

Xie Z, Zhu R, Zhao K, Liu J, Zhou G, Huang JX (2022) Dual gated graph attention networks with dynamic iterative training for cross-lingual entity alignment. ACM Trans Inf Syst 40(3):1–30. https://doi.org/10.1145/3471165. (**Article 44**)

Xin K, Sun Z, Hua W, Hu W, Zhou X (2022) Informed multi-context entity alignment. In: Paper presented at the 15th ACM international conference on web search and data mining, virtual event, Tempe, AZ, USA, 21–25 February 2022

Xiong F, Gao J (2019) Entity alignment for cross-lingual knowledge graph with graph convolutional networks. In: Paper presented at the 28th international joint conference on artificial intelligence, IJCAI 2019, Macao, China, 10–16 August 2019

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: Paper presented at the 57th conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019

Xu K, Song L, Feng Y, Song Y, Yu D (2020a) Coordinated reasoning for cross-lingual knowledge graph alignment. In: Paper presented at the 10th AAAI symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020

Xu W, Chen C, Jia C, Shen Y, Ma X, Lu W (2020b) Boosting cross-lingual entity alignment with textual embedding. In: Paper presented at the 9th CCF international conference, NLPCC 2020, Zhengzhou, China, 14–18 October 2020

Xu J, Li Y, Xie X, Li Y, Hu N, Zheng H, Jiang Y (2023) Investigating graph structure information for entity alignment with dangling cases. CoRR abs/2304.04718

Yan Z, Peng R, Wang Y, Li W (2020) CTEA: Context and Topic Enhanced Entity Alignment for knowledge graphs. https://doi.org/10.1016/j.neucom.2020.06.054

Yan Z, Peng R, Wang Y, Li W (2021) Soft-self and hard-cross graph attention network for knowledge graph entity alignment. Knowl Based Syst 231:107415. https://doi.org/10.1016/j.knosys.2021.107415

Yang B, Yih W, He X, Gao J, Deng L (2015) Embedding entities and relations for learning and inference in knowledge bases. In: Paper presented at the 3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015

Yang Y, Hao M, Huo Y, Chen L, Gao Z (2019a) Entity alignment algorithm based on dual-attention and incremental learning mechanism. IEEE Access 7:162179–162191. https://doi.org/10.1109/ACCESS.2019.2951785

Yang H, Zou Y, Shi P, Lu W, Lin J, Sun X (2019b) Aligning cross-lingual entities with multi-aspect information. In: Paper presented at the 9th international joint conference on natural language processing, EMNLP–IJCNLP 2019, Hong Kong, China, 3–7 November 2019

Yang K, Liu S, Zhao J, Wang Y, Xie B (2020) COTSAE: CO-Training of Structure and Attribute Embeddings for entity alignment. In: Paper presented at the 10th AAAI symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020

Yarowsky D (1995) Unsupervised word sense disambiguation rivaling supervised methods. In: Paper presented at the 33rd annual meeting of the Association for Computational Linguistics, MIT, Cambridge, Massachusetts, USA, 26–30 June 1995

Ye R, Li X, Fang Y, Zang H, Wang M (2019) A vectorized relational graph convolutional network for multi-relational network alignment. In: Paper presented at the 28th international joint conference on artificial intelligence, IJCAI 2019, Macao, China, 10–16 August 2019

Yu D, Yang Y, Zhang R, Wu Y (2020) Generalized multi-relational graph convolution network (Preprint). arxiv:2006.07331

Zeng W, Tang J, Zhao X (2019) Iterative representation learning for entity alignment leveraging textual information. In: Paper presented at international workshops of ECML PKDD 2019, Würzburg, Germany, 16–20 September 2019

Zeng W, Zhao X, Tang J, Lin X (2020a) Collective entity alignment via adaptive features. In: Paper presented at the 36th IEEE international conference on data engineering, ICDE 2020, Dallas, TX, USA, 20–24 April 2020

Zeng W, Zhao X, Wang W, Tang J, Tan Z (2020b) Degree-aware alignment for entities in tail. In: Paper presented at the 43rd international ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, virtual event, China, 25–30 July 2020

Zeng W, Zhao X, Tang J, Lin X, Groth P (2021a) Reinforcement learning-based collective entity alignment with adaptive features. ACM Trans Inf Syst 39(3):26–12631. https://doi.org/10.1145/3446428

Zeng W, Zhao X, Tang J, Li X, Luo M, Zheng Q (2021b) Towards entity alignment in the open world: an unsupervised approach. In: Paper presented at the 26th international conference, DASFAA 2021, Taipei, Taiwan, 11–14 April 2021

Zeng W, Zhao X, Tang J, Fan C (2021c) Reinforced active entity alignment. In: Paper presented at the 30th ACM international conference on information and knowledge management, virtual event, Queensland, Australia, 1–5 November 2021

Zhang Y, Paradis T, Hou L, Li J, Zhang J, Zheng H (2017) Cross-lingual infobox alignment in Wikipedia using entity-attribute factor graph. In: Paper presented at the 16th international semantic web conference, Vienna, Austria, 21–25 October 2017

Zhang Y, Liu L, Fu S, Zhong F (2018) Entity alignment across knowledge graphs based on representative relations selection. In: Paper presented at the 5th international conference on systems and informatics, ICSAI 2018, Nanjing, China, 10–12 November 2018

Zhang Q, Sun Z, Hu W, Chen M, Guo L, Qu Y (2019) Multi-view knowledge graph embedding for entity alignment. In: Paper presented at the 28th international joint conference on artificial intelligence, IJCAI 2019, Macao, China, 10–16 August 2019

Zhang J, Zhang Z, Zhang H, Ma Z, Ye Q, He P, Zhou Y (2021a) From electronic health records to terminology base: a novel knowledge base enrichment approach. J Biomed Inform 113:103628. https://doi.org/10.1016/j.jbi.2020.103628

Zhang G, Zhou Y, Wu S, Zhang Z, Dou D (2021b) Cross-lingual entity alignment with adversarial kernel embedding and adversarial knowledge translation (Preprint). arxiv:2104.07837

Zhang Y, Gao T, Lu J, Cheng Z, Xiao G (2021c) Adaptive entity alignment for cross-lingual knowledge graph. In: Paper presented at the 14th international conference, KSEM 2021, Tokyo, Japan, 14–16 August 2021

Zhang R, Trisedya BD, Li M, Jiang Y, Qi J (2021d) A comprehensive survey on knowledge graph entity alignment via representation learning (Preprint). arxiv:2103.15059

Zhang Y, Wu J, Yu K, Wu X (2023a) Independent relation representation with line graph for cross-lingual entity alignment. IEEE Trans Knowl Data Eng 35(11):11503–11514

Zhang Z, Tan C, Wang H, Zhao X, Yang M (2023b) Type-enhanced ensemble triple representation via triple-aware attention for cross-lingual entity alignment (Preprint). arxiv:2305.01556

Zhang Z, Tan C, Zhao X, Yang M, Jiang C (2023c) OTIEA: ontology-enhanced triple intrinsic-correlation for cross-lingual entity alignment. CoRR. arxiv:2305.01561

Zhang Y, Wu J, Yu K, Wu X (2024) Diverse structure-aware relation representation in cross-lingual entity alignment. ACM Trans Knowl Discov Data 18(4):1–23

Zhao X, Zeng W, Tang J, Li X, Luo M, Zheng Q (2022a) Toward entity alignment in the open world: an unsupervised approach with confidence modeling. Data Sci Eng 7(1):16–29. https://doi.org/10.1007/s41019-022-00178-4

Zhao X, Zeng W, Tang J, Wang W, Suchanek F (2022b) An experimental study of state-of-the-art entity alignment approaches. IEEE Trans Knowl Data Eng 34(6):2610–2625

Zheng C, Fan X, Wang C, Qi J (2020) GMAN: a graph multi-attention network for traffic prediction. In: Paper presented at the 34th AAAI conference on artificial intelligence, AAAI 2020, the thirty-second innovative applications of artificial intelligence conference, IAAI 2020, the tenth AAAI symposium on educational advances in artificial intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020

Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C, Sun M (2020) Graph neural networks: a review of methods and applications. AI Open 1:57–81

Zhu H, Xie R, Liu Z, Sun M (2017a) Iterative entity alignment via joint knowledge embeddings. In: Paper presented at the 26th international joint conference on artificial intelligence, IJCAI 2017, Melbourne, Australia, 19–25 August 2017

Zhu H, Xie R, Liu Z, Sun M (2017b) Iterative entity alignment via joint knowledge embeddings. In: Paper presented at the 26th international joint conference on artificial intelligence, IJCAI 2017, Melbourne, Australia, 19–25 August 2017

Zhu Q, Zhou X, Wu J, Tan J, Guo L (2019) Neighborhood-aware attentional representation for multilingual knowledge graphs. In: Paper presented at the 28th international joint conference on artificial intelligence, IJCAI 2019, Macao, China, 10–16 August 2019

Zhu Q, Wei H, Sisman B, Zheng D, Faloutsos C, Dong XL, Han J (2020) Collective multi-type entity alignment between knowledge graphs. In: Paper presented at the web conference 2020, Taipei, Taiwan, 20–24 April 2020

Zhu R, Ma M, Wang P (2021a) RAGA: Relation-Aware Graph Attention networks for global entity alignment. In: Paper presented at the 25th Pacific–Asia conference, PAKDD 2021, virtual event, proceedings, Part I, 11–14 May 2021

Zhu Y, Liu H, Wu Z, Du Y (2021b) Relation-aware neighborhood matching model for entity alignment. In: Paper presented at the 11th symposium on educational advances in artificial intelligence, EAAI 2021, virtual event, 2–9 February 2021

Zhu B, Bao T, Han R, Cui H, Han J, Liu L, Peng T (2023) An effective knowledge graph entity alignment model based on multiple information. Neural Netw 162:83–98

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Beibei Zhu[1] · Ruolin Wang[2,3] · Junyi Wang[2,3] · Fei Shao[2,3] · Kerun Wang[2,3]

✉ Beibei Zhu
zhubb24@lnnu.edu.cn

Ruolin Wang
wangrl20@mails.jlu.edu.cn

Junyi Wang
wangjy20@mails.jlu.edu.cn

Fei Shao
shaof20@mails.jlu.edu.cn

Kerun Wang
wangkr19@mails.jlu.edu.cn

[1] School of Computer Science and Artificial Intelligence, Liaoning Normal University, Liushu South Street, Dalian 116081, Liaoning, China

[2] School of Computer Science and Technology, Jilin University, Qianjin Street, Changchun 130012, Jilin, China

[3] Key Laboratory of Symbol Computation and Knowledge Engineering for Ministry of Education, Jilin University, Qianjin Street, Changchun 130012, Jilin, China