ZeroEA: A Zero-Training Entity Alignment Framework via Pre-Trained Language Model

Nan Huo
The University of Hong Kong
Hong Kong, China
huonan@cs.hku.hk

Reynold Cheng
The University of Hong Kong
Hong Kong, China
ckcheng@cs.hku.hk

Ben Kao
The University of Hong Kong
Hong Kong, China
kao@cs.hku.hk

Wentao Ning
The University of Hong Kong
Hong Kong, China
nwt9981@cs.hku.hk

Nur Al Hasan Haldar
The University of Western Australia
nur.haldar@uwa.edu.au

Xiaodong Li
The University of Hong Kong
Hong Kong, China
xdli@cs.hku.hk

Jinyang Li
The University of Hong Kong
Hong Kong, China
jl0725@cs.hku.hk

Mohammad Matin Najafi
The University of Hong Kong
Hong Kong, China
matin@cs.hku.hk

Tian Li
TCL Research
Hong Kong, China
tian23.li@tcl.com

Ge Qu
The University of Hong Kong
Hong Kong, China
quge@cs.hku.hk

ABSTRACT

Entity alignment (EA), a crucial task in knowledge graph (KG) research, aims to identify equivalent entities across different KGs to support downstream tasks like KG integration, text-to-SQL, and question-answering systems. Given rich semantic information within KGs, pre-trained language models (PLMs) have shown promise in EA tasks due to their exceptional context-aware encoding capabilities. However, the current solutions based on PLMs encounter obstacles such as the need for extensive training, expensive data annotation, and inadequate incorporation of structural information. In this study, we introduce a novel zero-training EA framework, ZeroEA, which effectively captures both semantic and structural information for PLMs. To be specific, Graph2Prompt module serves as the bridge between graph structure and plain text by converting KG topology into textual context suitable for PLM input. Additionally, in order to provide PLMs with concise and clear input text of reasonable length, we design a motif-based neighborhood filter to eliminate noisy neighbors. The comprehensive experiments and analyses on 5 benchmark datasets demonstrate the effectiveness of ZeroEA, outperforming all leading competitors and achieving state-of-the-art performance in entity alignment. Notably, our study highlights the considerable potential of EA technique in improving the performance of downstream tasks, thereby benefitting the broader research field.

PVLDB Reference Format:
Nan Huo, Reynold Cheng, Ben Kao, Wentao Ning, Nur Al Hasan Haldar, Xiaodong Li, Jinyang Li, Mohammad Matin Najafi, Tian Li, and Ge Qu. ZeroEA: A Zero-Training Entity Alignment Framework via Pre-Trained Language Model. PVLDB, 17(7): 1765 - 1774, 2024.
doi:10.14778/3654621.3654640

Xiaodong Li is the corresponding author.

Proceedings of the VLDB Endowment, Vol. 17, No. 7 ISSN 2150-8097.
doi:10.14778/3654621.3654640

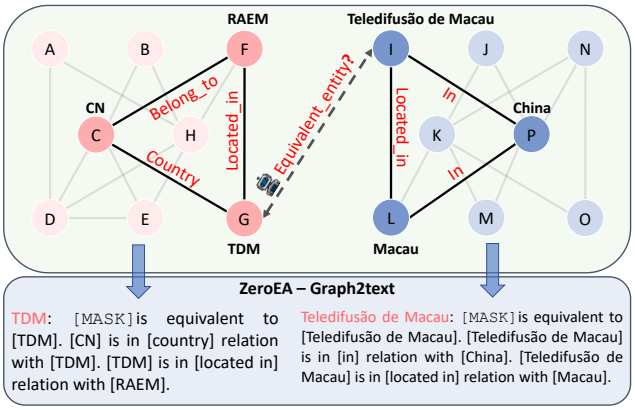

Figure 1: Illustrating entity alignment with ZeroEA.

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/Nan-Huo/ZeroEA.

1 INTRODUCTION

Knowledge graph (KG), a popular format of knowledge bases [4, 23, 26, 36], has emerged as a crucial technique to store structural and semantic information of large scale [21, 50, 65]. KGs have been playing a pivotal role in powering intelligent systems with knowledge-based reasoning [13, 14, 18, 19, 46, 62]. Entity alignment (EA) is one of the critical tasks of KGs, aiming to identify and link equivalent entities across different KGs. As the toy example shows in Figure 1, EA aims to identify whether the entities "TDM" and "Teledifusão de Macau" are equivalent in the real world and can link them together for a more comprehensive neighborhood. Thus EA can benefit downstream tasks such as KG integration [46, 62, 65], recommender system [17, 21, 28], and Text-to-SQL [22, 57].

Most existing EA solutions encode the entities and relations from different KGs into the same vector space through intensive training and then make predictions based on similarity measurements [46, 62]. The key to successful EA is encoding *structural*

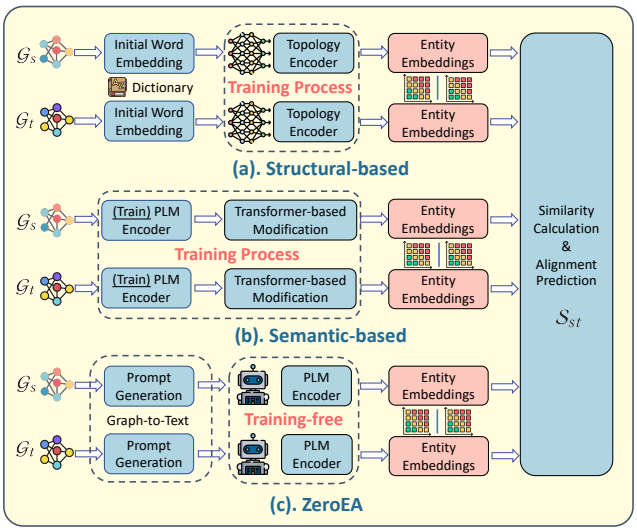

Figure 2: EA on knowledge graphs: (a) GNN encoder, wich trains a GNN with word embeddings as initial input; (b) PLM encoder, which trains a PLM and modifies it with with transformer variants; and (c) ZeroEA, which employs discrete prompt generation with training-free PLM encoder.

information [3, 32, 54] and *semantic* information [47, 65] properly. The two dimensions have formed the two main groups of existing EA methods. Most existing EA methods belong to the structural-based group, which carefully designs a graph topology encoder such as TransE [3] and Graph Neural Networks (GNNs) [46, 54], as shown in Figure 2(a).

On the other hand, the semantic-based group leverages the PLMs (*e.g.,* BERT [12]) to capture textual semantic information of KGs, as shown in Figure 2(b). This group achieves state-of-the-art performance among existing solutions. For example, BERT-INT [47] fine-tunes (defined in section 2) the PLMs on KG semantic information but fails to combine structural information. SDEA [65] leverages a PLM to encode entity attribute information and train a transformer-based [48] neural network to capture semantic information of neighbors.

After a detailed investigation and comparison among popular semantic-based methods, we have the following key observations. (a) They highly depend on intensive training or fine-tuning on PLMs and rely on vast data label annotation, which is costly in web-scale KGs and even sometimes unavailable in the real world. (b) Their definition of neighbors is based on edge connection. However, popular entity nodes have too many edge-connected neighbors in large-scale KGs, which distract the EA models and introduce noise [32], leading to inferior performance. Furthermore, given different neighbors contribute differently to the target node, they should be assigned varying levels of focus, as suggested by recent studies [65].
(c) In the existing literature, the impact of EA on downstream tasks has not been investigated. In this work, we focus on text-to-SQL as our primary downstream task, with the goal of bridging this research gap and providing valuable insights for the development of EA-enhanced downstream applications.

In this work, we propose ZeroEA, a novel zero-training EA framework using PLMs, as shown in Figure 1(c), which gets rid of the intensive fine-tuning process and data annotation by providing high-quality discrete prompt (*i.e.,* input text sequence of PLMs) to evoke the knowledge inherent in PLMs. ZeroEA adopts a Graph2Prompt module to transfer the KG topology information into discrete prompts with plenty of contexts. As illustrated in Figure 1, the target entity "TDM" and its two edges are transformed into the discrete prompt of "TDM". The Graph2Prompt module enables graph techniques (*e.g.,* frequent and small subgraphs, or motifs) to be understandable and used by PLMs. As motifs can identify stable structures (or *higher-order* structures) that are resistant to noise [10, 39, 56], our proposed motif-based neighborhood filter can be used with PLMs to remove noise and capture information precisely. Hence, compared to other supervised semantic-based approaches, ZeroEA is free from fine-tuning and can capture richer structural information while not losing semantic information. Additionally, we also observe that the accurate EA can benefit the downstream tasks such as text-to-SQLs. To summarize, our contributions are:
(1) We propose ZeroEA, a novel zero-training entity alignment framework via PLMs, which gets rid of the extensive PLM fine-tuning and data annotation by using high-quality discrete prompts to evoke the knowledge inherent in PLM.
(2) To capture richer structural information, we propose a motif-based neighborhood filter, which filters out noisy neighbors and captures higher-order KG structure information via motif.
(3) We conduct comprehensive experiments on five benchmark datasets, which indicates that ZeroEA outperforms state-of-the-art supervised approaches and significantly outperforms other unsupervised solutions.
(4) Finally, We also adopt ZeroEA in one of the state-of-the-art solutions in Text-to-SQL with impressive improvement.To do this, we redefine schema-linking, a critical intermediate step in the text-to-SQL process, as an EA problem. Our experimental results also indicate that improving the accuracy of EA results in the enhancement of text-to-SQL performance.

The remaining sections are organized as follows. Section 2 formally defines the entity alignment problem. Section 3 describes our ZeroEA framework for EA task. Section 4 provides the experimental results on benchmark datasets and downstream tasks. Section 5 discussed related works. Section 6 concludes the paper.

2 PRELIMINARIES

Problem Formulation. We first introduce some necessary notations. A KG consists of a collection of triples $\mathcal{T}$ with the format of *(head entity, relation, tail entity)*, an entity set $\mathcal{E}$ and a relation set $\mathcal{R}$. In the $\mathcal{R} = \{r_1, ..., r_n\}$, any r_i refers to a one-hop relation between nodes and a multi-hop relation with l hops is defined as $r^{mul} = r_1 \circ r_2 \circ ... \circ r_l$, which is the composition of one-hop relation. A KG is formally defined as $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{T}\}$.

Following the problem definition in [32], in the EA task, two different KGs are given: $\mathcal{G}_s = \{\mathcal{E}_s, \mathcal{R}_s, \mathcal{T}_s\}$ and $\mathcal{G}_t = \{\mathcal{E}_t, \mathcal{R}_t, \mathcal{T}_t\}$. EA aims to find the equivalent entities from $\mathcal{E}_s$ to $\mathcal{E}_t$ and vice versa. Finally, an alignment set $\mathcal{S}_{st}$ between the entities of the two KGs is

generated and defined in Eq. 1.

$$\mathcal{S}_{st} = \{(e_i, e_j) | e_i \in \mathcal{E}_s, e_j \in \mathcal{E}_t, e_i \Leftrightarrow e_j\} \tag{1}$$

where $\Leftrightarrow$ denotes equivalent entities in the real world, for example, "Teledifusão de Macau" and "TDM" shown in Figure 1.

Masked Pre-trained Language Model. Masked Language Models (MLMs), such as BERT [12] and RoBERTa [34], employ a training strategy that involves masking specific tokens in input sequences and replacing them with `[MASK]` tokens. The primary goal of the MLM is to predict the original tokens at the masked positions, thereby maximizing the likelihood of the correct tokens. Given a textual input $X = x_1, x_2, \ldots, x_n$, where the i-th token is masked, the objective function can be expressed as:

$$-\log \frac{\exp(c(\texttt{[MASK]}) \cdot \mathbf{E}_{x_i})}{\sum_{v \in V} \exp(c(\texttt{[MASK]}) \cdot \mathbf{E}_v)}, \tag{2}$$

where term $\mathbf{E}_v$ in this study denotes the word embedding of v, which belongs to the vocabulary set V.

Tuning-free Prompt for MLMs. The approach of tuning-free prompting for MLMs generates answers or embeddings directly by freezing all parameters of MLMs. This is achieved simply based on a given discrete prompt [30, 33], as outlined in the 4. In our method, we directly extract $c(\texttt{[MASK]})$ as contextualized representations of entities.

Text-to-SQL and Schema-linking. Text-to-SQL aims to convert natural language queries into SQLs, enabling the automatic return of results from relational databases for data science applications. A critical aspect of the text-to-SQL process is schema-linking, which entails mapping question tokens to their corresponding schema elements (tables or columns) resulting in more accurate SQL queries.

Consider a natural language question $\mathcal{Q} = \{q_1, \ldots, q_{|\mathcal{Q}|}\}$ and a database schema $\mathcal{S} = \langle \mathcal{C}, \mathcal{T} \rangle$, where $\mathcal{C} = \{c_1, \ldots, c_{|\mathcal{C}|}\}$ and $\mathcal{T} = \{t_1, \ldots, t_{|\mathcal{T}|}\}$ denote the columns and tables, respectively. The text-to-SQL task aims to generate a corresponding SQL query y for a given question $\mathcal{Q}$ in the context of schema $\mathcal{S}$.

Schema linking is a crucial technique for text-to-SQL generation since it can discriminate the relationship of arbitrary pairs of a question token and a schema item [5, 31, 49]. We represent schema-linking pairs as follows:

$$\mathcal{S}_l = \{(q_i, s_j) \mid q_i \in \mathcal{Q}, s_j \in \langle \mathcal{C}, \mathcal{T} \rangle, q_i \Leftrightarrow s_j\}. \tag{3}$$

In our experiments, we observe that improved schema-linking accuracy leads to generating more accurate SQL queries.

3 METHODOLOGY

This ZeroEA framework consists of three main components: (i) Prompt Generation Module (PGM), which transforms the KG topology into textual discrete prompts with plenty of context information from a filtered neighborhood. (ii) Embedding Module (EM) takes the discrete prompts generated by the PGM as the input of a selected PLM and outputs the context-aware embedding of each target entity. (iii) EA Prediction Module, which calculates the similarity between candidate entities and make alignment prediction based on it. The general overview of our proposed ZeroEA is shown in Figure 3.

Remarks. In this work, we use BERT [12] as the encoder.

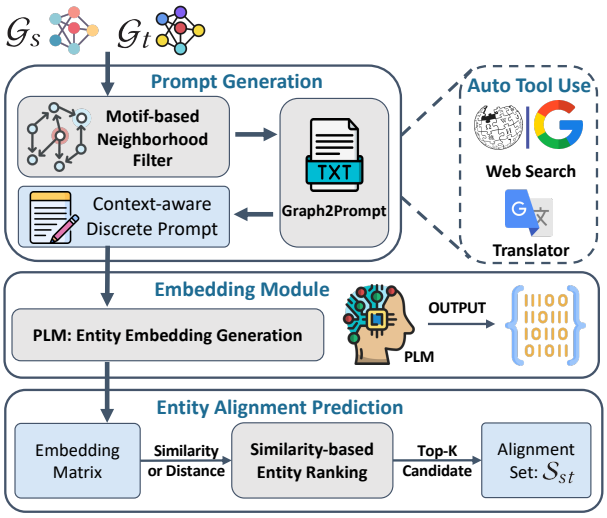

Figure 3: Architecture of ZeroEA.

3.1 Prompt Generation Module (PGM)

As shown in Figure 3, the main components of PGM consist of the motif-based neighborhood filter, Graph2Prompt, and optional auto tool use, which are illustrated in the following parts.

3.1.1 Motif-based Neighborhood Filter. The neighborhood filter is designed to filter out noisy neighbors of target nodes. The most popular way to capture structural information is to aggregate information from neighbors [32, 65]. However, "which neighbors should be covered?" is still in dispute. Recently, many works have claimed that it is beneficial to leverage multi-hop neighbors [53, 54]; however, other works find that one-hop neighbors can provide enough information [32, 60]. As argued in SelfKG [32] and BERT-INT [47], including multi-hop neighbors harms EA performance due to noisy neighbors. Also, PLMs have an input length limitation. For example, BERT [12] can only take input sequences up to 512 tokens in length, which means if having too many neighbors, the input prompt will exceed input limitation and harm the EA performance. To control input length, BERT-INT [47] selects neighbors with high similarity in BERT embedding, which is not sufficient if only semantic similarity between neighbors is considered.

To filter out noise and control the input length of PLMs, we propose a motif-based neighborhood filter. A motif, or a "fundamental building block" of a graph G, is a recurring and significant subgraph pattern of G [10, 24, 27, 28, 35, 52]. In the literature, motifs are said to express the higher-order relationship among nodes, and reduce noise in graph analytics [2, 8, 38, 64]. Motif and motif instance are defined in Def. 1 and 2 respectively, and a simple example of the motif is shown in Figure 1, *i.e.,* the triangles in bold color and lines [64]. Note that the binary matrix M represents the *adjacency matrix*, where each element $M_{i,j}$ denotes the existence of edge (i, j). Thus, besides regular neighbors connected by edges, each node in KGs also has its motif neighbors, which are connected by motif instances. As the example shown in Figure 1, when the motif is a triangle, node C and F are both edge neighbors and motif neighbors. However, node C and B are edge neighbors but not motif neighbors

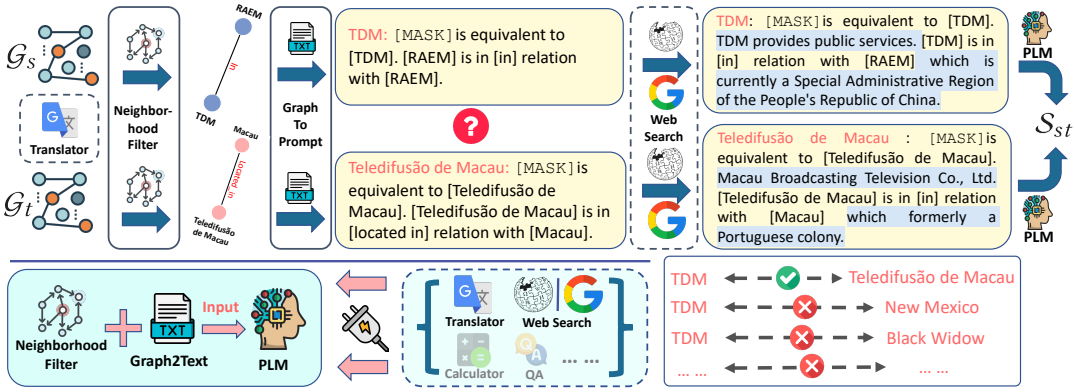

Figure 4: Illustrating the prompt generation module (PGM), where solid line boxes denote the components in PGM, while dash line boxes denote the optional tool use.

since there's no shared triangle motif between them. Compared to node pair C and B, node pair C and F has a smaller chance to rise from noise, because C and F constitute a motif instance, meaning that C and F participate in a higher-order relationship dictated by the motif, which occurs repeatedly in G. Thus, there is a higher confidence for F to be a neighbor of C [2, 39].

For the motif-based neighbor filter, we compute the embedding of each graph node by (1) finding its motif neighbors and their edges; (2) translating each of these edges to a sentence; and (3) combining the sentences into a paragraph and (4) passing the paragraph to the PLM model to generate an embedding for the node. To enumerate motif instances, we have used *E-CLoG* [11], a state-of-the-art local subgraph enumeration algorithm that can find instances for large subgraphs efficiently, e.g., several minutes for all p-node motifs on datasets of million scales, $p = 3, 4, 5$ [11]. That algorithm is also commonly used in big graph analytics (e.g., [25, 43, 51, 59]). Table 3 shows the overhead of motif enumeration for the datasets used in our experiments. As we can see, the required time is not large, and constitutes only 0.5% of the total running time of ZeroEA.

Definition 1 (Motif [64]). *A motif H of i nodes is defined as a tuple $(\boldsymbol{M}, V)$, where $V \subset \{1, 2, 3, ..., i\}$ denotes a set of anchor nodes which is the interested nodes set. And $\boldsymbol{M}$ is a binary matrix of size $i \cdot i$ representing the edge pattern of H.*

Definition 2 (Motif-instance [1]). *Given a graph $G = (V, E)$ and a motif $H = (V_H, E_H)$, the motif-instance $m = (V_m, E_m)$ of H is a subgraph of G which is isomorphic to H, denoted as $m \simeq H$.*

To maintain a high-quality neighborhood with reasonable size, we assign different importance values to neighbors and select the top k important neighbors. we adopt IND as one of the baselines in this module, where a higher node degree means higher importance value [63].

As mentioned, motifs have the ability to reduce noise and capture higher-order relations compared with edge-based relations [64]. In this module, to make the most use of motifs, we explore different ways to use motifs to help us select top k important neighbors, where k is an integer decided by users. We have the following baseline methods to select neighbors in the neighborhood filter module:

(1) n-hop neighbors: all n-hops neighbors are selected, where n is an integer decided by users.

(2) n-hop motif neighbors: all n-hop motif neighbors (*i.e.,* all neighbors that have a n-hop motif-path to target node, where a motif-path is a concatenation of one or more motif instances) are selected.

(3) IND [63]: edge-based neighbors are ranked based on node degrees.

(4) M-IND: motif neighbors are ranked based on motif degree (*i.e.,* the number of motif instances including the given node) values.

However, IND and M-IND focus on the popularity of each neighbor and neglect measuring the interconnection between neighbors and the target node. To capture different interconnection levels between different neighbors and the target node, we propose a *motif-relevance neighborhood filter* where the importance value of each neighbor is measured by the number of shared motifs with the target node. After that, all neighbors are ranked based on the importance value, and we select top k important neighbors. The selected neighbor set S_{ne} with k neighbors of given node e_i in KG $\mathcal{G}_a$ is defined as: $S_{ne} = \text{Neighborhood-Filter}(e_i, \mathcal{G}_a) = \{(e_i, r_1, e_1), (e_i, r_2, e_2), ..., (e_i, r_n, e_n)\}$.

3.1.2 Graph2Prompt. After applying the Graph2Prompt operation, the selected top k neighbors from the neighborhood filter are concatenated together to be the discrete prompt, then can be input to PLMs. The discrete prompt is defined as:

$$\begin{aligned} Prompt_{e_i} = \text{Concat}([&\texttt{[MASK]}\text{is equivalent to } [e_i], \\ &[e_i] \text{ is in } [r_1] \text{ relation with } [e_1], \\ &\ldots\ \ldots \\ &[e_i] \text{ is in } [r_n] \text{ relation with } [e_n]]\,) \end{aligned} \tag{4}$$

where "[MASK]" is a special token in BERT and "Concat(·)" denotes concatenate operation. r_n is the relation between the target entity e_i and the n-th selected neighbor e_n.

If the Web Search tool is applied based on the percipient strategy defined in Section 3.1.5, the short external knowledge about e_i is injected, as shown in equation 5.

$$Prompt_{e_i} = \text{Concat}(\,\text{WebSearch}(e_i), Prompt_{e_i}\,) \tag{5}$$

An illustration of this process can be found in Figure 4.

3.1.3 Embedding Module (EM). The embedding (i.e., semantic representation) of the input token list T is first encoded by a multi-layer bidirectional Transformer [48]. Each Transformer layer has two sub-layers, i.e., a Multi-Head self-Attention network (MHA) and a Fully-connected Forward Network (FFN). In summary, The last layer of BERT semantic hidden states of the entity e_i can be acquired as shown in Eq. 6.

$$\mathbf{E}_{e_i} = \text{Enc}_{\Theta}(Prompt_{e_i}) \tag{6}$$

where Enc denotes the PLM encoder, which is BERT in this work. And Θ is the original parameter of the PLM encoder.

And the final entity embedding of given entity e_i is the hidden states of the special token "`[MASK]`" which represent the context-aware embedding of e_i, and "`[CLS]`" which is usually recognized as the embedding of the whole input prompt [12], as formulated in Eq. 7:

$$\mathbf{c}(\texttt{[MASK]}) = \frac{\mathbf{E}_{ei}(\texttt{[MASK]}) + \mathbf{E}_{ei}(\texttt{[CLS]})}{2} \tag{7}$$

3.1.4 Entity Alignment Prediction. After acquiring all the entity embeddings, the similarity score between the target entity embedding $\mathbf{E}_t$ and a candidate entity embedding $\mathbf{E}_c$ can be measured by cosine similarity as follows:

$$\cos(\mathbf{E}_t, \mathbf{E}_c) = \frac{\mathbf{E}_t \cdot \mathbf{E}_c}{\|\mathbf{E}_t\| \|\mathbf{E}_c\|} \tag{8}$$

The higher similarity score means the target entity and the candidate entity are more likely to be aligned.

3.1.5 Auto Tool Use Strategy: Percipient. Similar to the use of linguistic translation tool in [32, 53, 54], to address the limitations of PLMs, *e.g.,* inability to access the up-to-date knowledge, we propose a novel tool-based framework under which ZeroEA can automatically use tools to expand its capacities. *In this work, we propose the percipient strategy to use tools that should fulfill the following requirements*: (1) The use of tools should be in an automatic way without any human supervision of annotations. (2) The use of tools should be in an on-demand manner and decide *when* and *how* to use tools instead of using all tools everywhere.

We take the Web Search tool as an example in this section. Intuitively, this tool should be applied when the quality of translation is unsatisfying or the node degree of a specific entity is low. To measure the quality of the translation, we adopt the Rouge-L score [29], which is one of the most widely used metrics in the machine translation field.

As for the translation quality measurement process, when given the source text sequence S of length m and target text sequence T of length n, the Rouge-L score is measured as follows: where $LCS(S, T)$ denotes the common sub-sequence with maximum length of S and T, and $\beta = P_{lcs}/R_{lcs}$.

Let F_{lcs} denote the Rouge-L score, α represents the Rouge-L threshold, and γ indicates the entity degree threshold set by the users. If F_{lcs} is lower than α or the entity degree is less than γ, the Web Search tool is applied. In this case, the Web Search tool outputs additional information to enhance the entity representation. Mathematically, the objective function can be expressed in Eq. 9:

$$\text{WebSearch}(e_i) = \begin{cases} K_E, & \text{if } F_{lcs} < \alpha \text{ or } \text{degree}(e_i) < \gamma \\ \text{None}, & \text{otherwise} \end{cases} \tag{9}$$

where: $\text{WebSearch}(e_i)$ is the output of the Web Search tool for entity e_i. K_E represents the external knowledge retrieved through the Web Search tool, including the correct entity name in English and a brief introduction or definition, as the highlighted text in Figure 4. $\text{degree}(e_i)$ is the degree of entity e_i in the KG, indicating its connectedness within the graph. α is the Rouge-L threshold set by users, indicating the minimum acceptable translation quality. γ is the entity degree threshold set by users, indicating the minimum acceptable entity degree. Limited by space, we report the strategy to prevent information leakage in tool use, which rarely happens, in our GitHub repository.

4 EXPERIMENTS

4.1 Experimental Setup

Datasets (1) The **DBP15K** dataset [44] is a well-recognized benchmark for entity alignment, containing three smaller subsets for cross-lingual EA, each with 15,000 aligned entity pairs. (2) The **DWY100K** dataset [16] comprises two subsets of a medium scale for monolingual EA, with each subset including 100,000 aligned entity pairs and around one million triples. (3) The **DBP1M** dataset [15] is among the largest EA benchmarks to date, featuring two cross-lingual subsets, each with over one million entities and nearly **ten million** triples. (4) **SPIDER** [57]: In our work, we consider text-to-SQL as the primary downstream task in our study, aiming to study the impact of entity alignment on the performance of downstream applications. The SPIDER is a large-scale, complex, and cross-domain text-to-SQL dataset. It contains 10,181 questions, 5,693 unique SQL queries, over 200 databases across 138 domains.

Evaluation Metrics: Consistent with benchmark works [46, 62], we adopt two evaluation metrics: hits@K and mean reciprocal rank (MRR). Higher hits@K and MRR mean better performance.

In SPIDER, Exact Match (EM) and Execution Accuracy (EX) are the two primary metrics used to evaluate the end text-to-SQL performance [41]. In addition, we employ the precision, recall, and f1 scores to measure the effectiveness of Schema Linking supported by Entity Alignment (EA) models. The correlation between Entity Alignment EA-enhanced schema linking and its subsequent text-to-SQL performance can reveal that more accurate EA can benefit downstream tasks.

Experimental Settings: We follow the original split of datasets, where 70% of seed alignment data is used as test data, and the other 30% is used as the training data and validation data for supervised methods. In our text-to-SQL implementation, we employ the technique outlined in [22]. Graphix-T5, a state-of-the-art model, treats question tokens and schema items as two small KGs and performs schema linking via string-matching techniques. In the default graph constructed within Graphix-base, the newly added relationships through EA are labeled as `semanticmatch`. Due to the high computational cost of text-to-SQL, we perform our comparative analysis of EA methods to the base version of Graphix-T5.

Compared Methods: We compare our ZeroEA with: 1. Supervised methods that need to use 100% training set data of EA.

Table 1: Recent results on DBP15K and DWY100K. ZeroEA(dir) means adopting directed KGs and (undir) means undirected KGs.

Model		$DBP15K_{zh\_en}$			$DBP15K_{ja\_en}$			$DBP15K_{fr\_en}$			$DWY100K_{dbp\_wd}$			$DWY100K_{dbp\_yg}$		
		Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR
Supervised																
Trans.	MTransE [7]	0.308	0.614	0.364	0.279	0.575	0.349	0.244	0.556	0.335	0.281	0.520	0.362	0.252	0.493	0.376
	JAPE [44]	0.412	0.745	0.490	0.363	0.685	0.476	0.324	0.667	0.430	0.318	0.589	0.378	0.236	0.484	0.364
	BootEA [45]	0.629	0.848	0.703	0.622	0.854	0.701	0.653	0.874	0.731	0.748	0.791	0.898	0.761	0.894	0.818
	TransEdge [3]	0.735	0.919	0.801	0.719	0.932	0.795	0.710	0.941	0.796	0.788	0.938	0.832	0.792	0.936	0.889
GNN	GCN-Align [53]	0.413	0.744	-	0.399	0.745	-	0.373	0.745	-	0.477	0.562	0.514	0.601	0.642	0.623
	MuGNN [6]	0.494	0.844	0.611	0.501	0.857	0.621	0.495	0.870	0.621	0.616	0.897	0.732	0.741	0.937	0.856
	RDGCN [54]	0.708	0.846	0.746	0.767	0.895	0.812	0.886	0.957	0.911	0.902	0.954	0.923	0.864	0.889	0.973
	CEAFF [58]	0.795	-	-	0.860	-	-	0.964	-	-	1.000	-	-	1.000	-	-
	MEAformer [9]	0.949	0.993	0.965	0.978	0.999	0.986	0.991	1.00	0.995	-	-	-	-	-	-
PLM	BERT-INT [47]	0.968	0.990	0.977	0.964	0.991	0.975	0.995	0.998	0.995	0.992	0.999	0.999	0.999	0.999	0.999
	SDEA [65]	0.870	0.966	0.910	0.848	0.952	0.890	0.969	0.995	0.980	0.980	0.996	0.990	0.999	1.0	1.0
Unsupervised & Self-supervised																
Trans.	MultiKE [61]	0.509	0.576	0.532	0.393	0.489	0.432	0.639	0.712	0.665	0.915	0.974	0.932	0.880	0.962	0.916
GNN	SelfKG [32]	0.829	0.919	-	0.890	0.953	-	0.959	0.992	-	0.983	0.998	-	0.998	1.000	-
PLM	ZeroEA(dir)	0.972	0.990	0.981	0.975	0.992	0.981	0.983	0.992	0.988	0.986	0.991	0.988	0.999	1.000	0.999
	ZeroEA(undir)	**0.985**	**0.993**	**0.991**	**0.982**	**0.995**	**0.989**	**0.998**	**0.999**	**0.998**	**0.998**	**0.999**	**0.996**	**0.999**	**1.000**	**0.999**

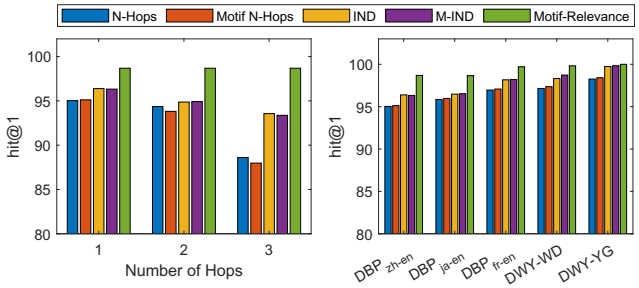

Figure 5: Ablation study on neighborhood filter of ZeroEA.

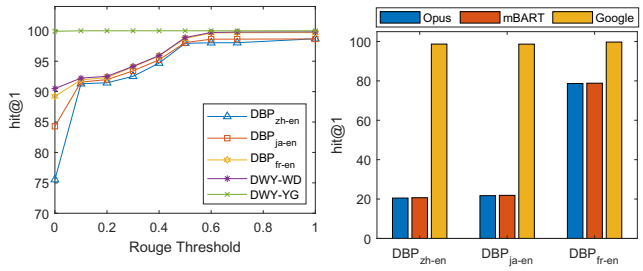

Figure 6: Ablation study on translator of ZeroEA.

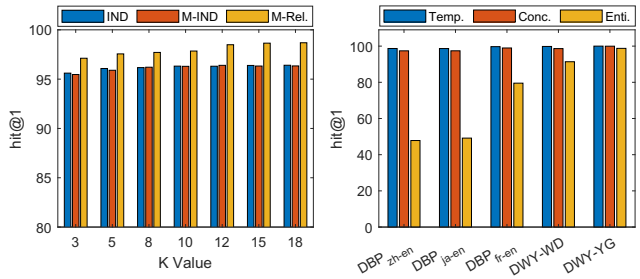

Figure 7: Ablation study on neighborhood filter and Graph2Prompt of ZeroEA.

Supervised methods are further grouped into (1) translation-based methods ("Trans." in Table 1, which are variants of TransE [3]. (2) GNN-based methods ("GNNs" in Table 1), which are variants of GNNs. (3) PLM-based methods("PLM" in Table 1), which using a PLM as the encoder. Group (1), (2) are the structure-based methods and group (3) are semantic-based methods. 2. Unsupervised and self-supervised that don't need to utilize any training set data of EA. They are grouped similarly to the supervised group above. In text-to-SQL settings, we select Graphix-T5-base as the baseline model. We enhance schema linking via these EA methods and the ZeroEA with different thresholds.

4.2 Experimental Results

4.2.1 Overall Results. **Results on multi-lingual datasets:** As shown in Table 1, ZeroEA outperforms both supervised and unsupervised & self-supervised baselines groups by a significant margin.

(1) Compared with the supervised group ZeroEA outperforms the best baseline by 1.9%, 2.1% and 0.5% (ZeroEA achieves 99.9% already) on ZH-EN, JA-EN, and FR-EN respectively, which demonstrates that intensive training is not a necessity and the PLMs' capability to deal with structured KG data is outstanding with proper contextual information. As argued in BERT-INT, the supervised state-of-the-art model in EA, semantic information is even more important than structural information of KGs [47], which is also demonstrated by the performance of our ZeroEA. It reveals that the semantic-information-based embedding method, especially the PLM-based one, is a promising way in EA task, which is not a well-explored solution compared with numerous GNN-based methods.

(2) In the unsupervised & self-supervised group ZeroEA outperforms the SelfKG, the novel GNN-based unsupervised solution, by 16.2%, 7.7% and 3.8% (ZeroEA achieves 99.9% already) on ZH-EN, JA-EN, and FR-EN respectively, which indicates ZeroEA can encode the structural and semantic information of KGs in a more effective way under the same low-resource condition. And ZeroEAbecomes the new state-of-the-art model in EA and doesn't even

Table 2: Ablation study on DBP15K.

Model	$DBP15K_{zh\_en}$			$DBP15K_{ja\_en}$			$DBP15K_{fr\_en}$			macro Hit@1
	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	
ZeroEA	**0.985**	**0.993**	**0.991**	**0.982**	**0.995**	**0.989**	**0.998**	**0.999**	**0.998**	**0.988**
-Neighborhood Filter	0.950	0.978	0.968	0.956	0.978	0.963	0.969	0.999	0.979	0.958
-Graph2Prompt	0.478	0.736	0.518	0.492	0.768	0.532	0.795	0.826	0.831	0.686
-Web Search	0.837	0.885	0.891	0.879	0.933	0.913	0.923	0.976	0.986	0.881
-Translator	0.253	0.291	0.313	0.265	0.301	0.275	0.724	0.839	0.788	0.589

have a training process, demonstrating the strong generalization ability of ZeroEAin zero-shot condition.

4.2.2 Ablation Study. (1) Without Graph2Prompt, the foundation of our proposed model, the performance declines sharply by approximately 50%. This suggests that Pre-trained Language Models (PLMs) excel in handling textual data rather than structured graph data. In the absence of Graph2Prompt, our model would underperform in Entity Alignment tasks. (2) When removing the neighborhood filter module, the performance is reduced from 2.7% to 3.8%, shows that higher-order information brought by motifs is very beneficial for EA task. (3) The reduction of the Web Search tool's performance ranging from 4.6% to 15% demonstrates its strong ability to handle noise from low-quality translations and the substantial proportion of entities possessing limited structural and semantic information. As noted in [65], approximately 40% of entities in DBP15k exhibit degrees less than 5, thereby possessing limited structural information. This observation highlights the significance of leveraging external knowledge to supplement incomplete information within a single Knowledge Graph (KG), ultimately benefiting tasks that rely on such knowledge. (4) When removing the Translator tool, the performance undergoes a great drop, even more than 70%. The reduction is correlated with the similarity between 2 languages. It indicates the limited ability of BERT to process low-resource (*i.e.,* non-English) languages.

4.2.3 Auto tool use. (1) As mentioned before, we adopt the Rouge score to measure translation quality. Figure 6 shows the performance trend with the increase of Rouge threshold, and the turning point is at 0.5, which means that it is the most efficient to set Rouge threshold to be 0.5 when applying web search on fewer entities but have similar performance. (2) As Figure 6 shows, when using small-scale transformer-based machine translation, the performance is extremely low because it can translate most entities; however, when adopting Google translation, which can generate translation to any text, the performance becomes reasonable.

4.2.4 Error Analysis. We conducted a detailed investigation into the errors of ZeroEA using the challenging ZH-EN DBP15K dataset. Out of 136 errors, three main types were identified. The majority, **wrong translation (68.5%)**, leads to significant deviations in embeddings and incorrect predictions. For instance, "Divas in Distress" was mistranslated as "wanna mama...", and "Sukhoi PAK FA" as "KAI T50 Golden Eagle". The second type, **Low degree entities (18%)**, struggle with encoding due to web search discrepancies and sparse structural information. An example is "Teledifusão de Macau S.A.", with an aligned entity name "TDM", requiring more structural context for accurate recognition. The third error type is **wrong label (6%)**, such as the misalignment of "Kvitrafn" and "Einar Selvik". Lastly, **bad WebSearch calls (11%)** can introduce irrelevant data, increasing uncertainty for models. And the last type is **bad WebSearch calls (11%)**. The bad WebSearch calls can inject irrelevant knowledge, thus increase the uncertainty of entities to models.

Table 3: The frequency and counting time (C.Time) by *ESCAPE* [40], with the sum of local motif enumerating time (E. Time) for each node by *E-CLoG* [11], with triangle motif.

Datasets		Frequency	C.Time	E.Time
$DBP15K_{ZH-EN}$	ZH	21,514	0.979s	0.073s
	EN	39,654	1.040s	0.131s
$DBP15K_{JA-EN}$	JA	37,036	1.021s	0.091s
	EN	45,250	1.084s	0.166s
$DBP15K_{FR-EN}$	FR	76,813	1.070s	0.210s
	EN	68,393	1.197s	0.344s
DWY_{DBP-YG}	DBP	179,813	1.367s	1.797s
	YG	185,724	1.489s	3.025s
DWY_{DBP-WD}	DBP	161,124	1.608s	1.355s
	WD	125,722	1.187s	1.106s
$DBP1M_{FR-EN}$	FR	1,105,683	4.070s	154.658s
	EN	778,216	2.644s	146.525s
$DBP1M_{DE-EN}$	DE	786,579	3.202s	101.882s
	EN	331,686	2.502s	51.690s

4.2.5 Neighborhood Filter. As mentioned in Section 3.1.1, we have four baselines and our proposed method of neighborhood filter. In this study, we compare our motif-relevance neighborhood filter to four baselines on five benchmark datasets. Our findings indicate that motif-based n-hop neighbors and selecting n-hop neighbors exhibit comparable performance. The superior performance of IND and M-IND can be ascribed to their capacity to handle lengthy PLM inputs by selecting the k neighbors with the highest importance values. Our motif-relevance strategy consistently outperforms the baselines, supporting the hypothesis that contribution value should be determined by the correlation between two nodes. In all our datasets, we use a "triangle" as our motif ($\triangle$). This is because the instances of $\triangle$ are abundant, as shown in Table 3. Moreover, the overhead for counting $\triangle$ instances, based on the algorithm in [40], is little, just taking 1 to 4 seconds (Table 3). We also remark that $\triangle$ is commonly used in motif-based analytics [2, 8, 25, 38, 39, 42, 43, 64].

4.2.6 Scalability and Efficiency. One of the major contributions of ZeroEA comes from its model scalability. To convincingly demonstrate the scalability of ZeroEA, we have conducted extensive experiments across datasets of varying sizes, including one of the

Table 4: Experimental results on schema linking of Spider.

Model	Recall	Precison	F1
Graphix-base	0.683	0.594	0.635
Graphix-TransE	0.756	0.708	0.731
Graphix-GCN	0.791	0.732	0.760
Graphix-ZeroEA	**0.902**	**0.883**	**0.892**

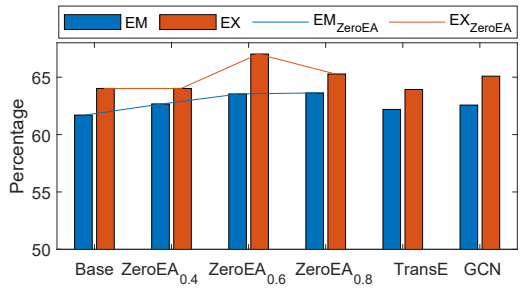

Figure 8: Experimental results on Text-to-SQL. $\mathrm{EM}_{\mathrm{ZeroEA}}$ **shows the trend of EM across different ZeroEA settings.**

largest EA datasets, known as DBP1M [15]. This dataset is notably from ten to a hundred times larger than other datasets used. The experiments cover three distinct levels of dataset magnitude.

By employing datasets that span small, medium, and large scales, as well as both monolingual and cross-lingual contexts, we are able to thoroughly evaluate the robustness of ZeroEA in scaling to different data volumes. All the other EA baselines either fail to operate due to exceeding GPU memory limitations (12G) or require more than three days of computation time. Moreover, these methods are generally incapable of handling large knowledge graphs (KGs) because their training procedures have to load the entire graph into memory. In contrast, ZeroEA is designed to bypass these limitations by avoiding the training phase altogether and instead loading only small subgraphs that correspond to the selected one-hop neighborhood for one target entity each time. This approach effectively decouples the performance of ZeroEA from the size of the KG, thus affirming its scalability to very large KGs without the common constraints faced by other models. Also, ZeroEA performs well on these large datasets for accuracy evaluation. On $\mathrm{DBP1M}_{\mathrm{FR-EN}}$, ZeroEA obtains Hit@1= 0.594, Hit@10=0.635, and MRR=0.400; on $\mathrm{DBP1M}_{\mathrm{DE-EN}}$, ZeroEA obtains Hit@1= 0.616, Hit@10=0.648, and MRR=0.395, which are also state-of-the-art [37]. These results show that ZeroEA not only scales effectively but also maintains state-of-the-art performance across various dataset categories, including cross-lingual, monolingual, and very large-scale subsets.

4.3 Downstream Task Application: Text-to-SQL

Figure 8 presents the text-to-SQL performance on the SPIDER dataset, equipped with various EA-enhanced schema linkings. The `Base` denotes the vanilla model known as Graphix-base, whereas `ZeroEA_`α represents the Graphix-base model with a α threshold for the inclusion of new relationships. Our results demonstrate that (a) The Graphix-base model with ZeroEA outperforms the standard Graphix-base model in downstream tasks, achieving optimal performance when the threshold is set to 0.6.
(b) Furthermore, Table 4 shows that EA offers the most significant enhancement to schema linking for the Graphix-base model.
(c) It is worth noting that a clearly positive correlation exists between the schema-linking F1 and the performance in the end text-to-SQLs, as illustrated in Figure 8. This demonstrates that EA can benefit downstream tasks, thus motivating further exploration of its impact on various downstream tasks.

5 RELATED WORKS

Entity alignment research in recent decades can be grouped into rule-based[62], crowdsourcing-based[46], deep learning (DL)[32, 53, 54], and PLM-based approaches[47, 65]. DL-based methods, particularly embedding-based strategies, have shown superior performance. These methods often use TransE [3] to train KG embeddings, but newer approaches consider KG structures and use graph neural networks [53] or attention-based mechanisms [32]. Some recent work focus on multi-modal EA, for example MEAformer [9]. Some also incorporate semantic information [53–55, 58, 61] or attribute values [47] for improved performance. However, PLM training is costly and time-consuming, so our work aims to use structural and semantic information from KGs without extensive training.

Despite the progress in Entity Alignment (EA) tasks, their impact on downstream tasks like schema-linking in text-to-SQL and Knowledge Graph-based Question Answering (KGQA) [20, 55] is underexplored. EA is crucial for these complex tasks, which can be significantly improved with precise entity alignment. We are the first to investigate EA's effects on downstream tasks.

6 CONCLUSION

We introduce ZeroEA, a novel zero-training framework for entity alignment (EA) that adeptly harnesses the contextual encoding capabilities of pre-trained language models. ZeroEA can adeptly incorporate both semantic and structural information from knowledge graphs via the Graph2Prompt module and the motif-based neighborhood filter. The experimental results achieved on five benchmark datasets not only position ZeroEA as a cutting-edge solution, but also highlight its potential to enhance complex downstream tasks and contribute to the ongoing progress in knowledge graph research. In the future, we plan to develop a trained version of ZeroEA, which can obtain higher performance at the cost of longer training time. We report preliminary results in our GitHub repository and plan to further develop it in the future.

ACKNOWLEDGMENTS

Reynold Cheng, Nan Huo, Wentao Ning, Jinyang Li, Xiaodong Li, Mohammad Matin Najafi, and Ge Qu are supported by the Hong Kong Jockey Club Charities Trust (Project 260920140), the University of Hong Kong (Project 109000579), the France/Hong Kong Joint Research Scheme 2020/21 (Project F-HKU702/20), and the HKU Outstanding Research Student Supervisor Award 2022-23. [add ack from other authors...] We would also like to thank Dr. Jiajun Shun (Google Deepmind) for his advice. We would like to express our deepest thanks to Dr. Jiajun Shen (Google Deepmind) for his invaluable advice in the early stage of this project.

REFERENCES

[1] Nesreen K Ahmed, Jennifer Neville, Ryan A Rossi, and Nick Duffield. 2015. Efficient graphlet counting for large networks. In *2015 IEEE international conference on data mining*. IEEE, 1–10.

[2] Austin R Benson, David F Gleich, and Jure Leskovec. 2016. Higher-order organization of complex networks. *Science* 353, 6295 (2016), 163–166.

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems* 26 (2013).

[4] Hongtai Cao, Qihao Wang, Xiaodong Li, Mohammad Matin Najafi, Kevin Chen-Chuan Chang, and Reynold Cheng. 2024. Large Subgraph Matching: A Comprehensive and Efficient Approach for Heterogeneous Graphs. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE.

[5] Ruisheng Cao, Lu Chen, Zhi Chen, Yanbin Zhao, Su Zhu, and Kai Yu. 2021. LGESQL: Line Graph Enhanced Text-to-SQL Model with Mixed Local and Non-Local Relations. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, 2541–2555.

[6] Yixin Cao, Zhiyuan Liu, Chengjiang Li, Zhiyuan Liu, Juanzi Li, and Tat-Seng Chua. 2019. Multi-Channel Graph Neural Network for Entity Alignment. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1452–1461. https://doi.org/10.18653/v1/P19-1140

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

[8] Xiaowei Chen and John CS Lui. 2018. Mining graphlet counts in online social networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 12, 4 (2018), 1–38.

[9] Zhuo Chen, Jiaoyan Chen, Wen Zhang, Lingbing Guo, Yin Fang, Yufeng Huang, Yichi Zhang, Yuxia Geng, Jeff Z Pan, Wenting Song, et al. 2023. Meaformer: Multi-modal entity alignment transformer for meta modality hybrid. In *Proceedings of the 31st ACM International Conference on Multimedia*. 3317–3327.

[10] Reynold Cheng, Chenghao Ma, Xiaodong Li, Yixiang Fang, Ye Liu, Victor Wong, Esther Lee, Tai Hing Lam, Sai Yin Ho, Man Ping Wang, Weijie Gong, Wentao Ning, and Ben Kao. 2022. The Social Technology and Research (STAR) Lab in the University of Hong Kong. *ACM SIGMOD Record* 51, 2 (2022), 63–68.

[11] Vachik S Dave, Nesreen K Ahmed, and Mohammad Al Hasan. 2017. E-CLoG: counting edge-centric local graphlets. In *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 586–595.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[13] Yixiang Fang, Reynold Cheng, Xiaodong Li, Siqiang Luo, and Jiafeng Hu. 2017. Effective community search over large spatial graphs. *Proceedings of the VLDB Endowment (PVLDB)* 10, 6 (2017), 709–720. https://dl.acm.org/doi/10.14778/3055330.3055337

[14] Yixiang Fang, Zheng Wang, Reynold Cheng, Xiaodong Li, Siqiang Luo, Jiafeng Hu, and Xiaojun Chen. 2018. On spatial-aware community search. *IEEE Transactions on Knowledge and Data Engineering (TKDE)* 31, 4 (2018), 783–798. https://ieeexplore.ieee.org/document/8375664

[15] Congcong Ge, Xiaoze Liu, Lu Chen, Baihua Zheng, and Yunjun Gao. 2021. LargeEA: Aligning Entities for Large-scale Knowledge Graphs. *Proc. VLDB Endow.* 15, 2 (2021), 237–245. https://doi.org/10.14778/3489496.3489504

[16] Lingbing Guo, Zequn Sun, and Wei Hu. 2019. Learning to exploit long-term relational dependencies in knowledge graphs. In *International conference on machine learning*. PMLR, 2505–2514.

[17] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering* 34, 8 (2020), 3549–3568.

[18] Xiaolin Han, Reynold Cheng, Tobias Grubenmanny, Silviu Maniuz, Chenhao Ma, and Xiaodong Li. 2022. Leveraging Contextual Graphs for Stochastic Weight Completion in Sparse Road Networks. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*. SIAM.

[19] Xiaolin Han, Tobias Grubenmann, Reynold Cheng, Sze Chun Wong, Xiaodong Li, and Wenya Sun. 2020. Traffic Incident Detection: A Trajectory-based Approach. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, 1866–1869. https://ieeexplore.ieee.org/document/9101794

[20] Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards Reasoning in Large Language Models: A Survey. arXiv:2212.10403 [cs.CL]

[21] Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 105–113.

[22] Jinyang Li, Binyuan Hui, Reynold Cheng, Bowen Qin, Chenhao Ma, Nan Huo, Fei Huang, Wenyu Du, Luo Si, and Yongbin Li. 2023. Graphix-T5: Mixing Pre-trained Transformers with Graph-Aware Layers for Text-to-SQL Parsing. In *AAAI*. AAAI Press, 13076–13084.

[23] Xiaodong Li. 2019. DURS: A Distributed Method for k-Nearest Neighbor Search on Uncertain Graphs. In *2019 20th IEEE International Conference on Mobile Data Management (MDM)*. IEEE, 377–378. https://ieeexplore.ieee.org/document/8788813

[24] Xiaodong Li, Tsz Nam Chan, Reynold Cheng, Caihua Shan, Chenhao Ma, and Kevin Chang. 2019. Motif paths: A new approach for analyzing higher-order semantics between graph nodes. *HKU Technique Reports* 3 (2019), 4.

[25] Xiaodong Li, Reynold Cheng, Kevin Chen-Chuan Chang, Caihua Shan, Chenhao Ma, and Hongtai Cao. 2021. On analyzing graphs with motif-paths. *Proceedings of the VLDB Endowment* 14, 6 (2021), 1111–1123.

[26] Xiaodong Li, Reynold Cheng, Yixiang Fang, Jiafeng Hu, and Silviu Maniu. 2018. Scalable evaluation of k-nn queries on large uncertain graphs. In *21st International Conference on Extending Database Technology (EDBT)*. 181–192. https://openproceedings.org/2018/conf/edbt/paper-69.pdf

[27] Xiaodong Li, Reynold Cheng, Matin Najafi, Kevin Chang, Xiaolin Han, and Hongtai Cao. 2020. M-Cypher: A GQL Framework Supporting Motifs. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM)*. 3433–3436. https://dl.acm.org/doi/10.1145/3340531.3417440

[28] Xiaodong Li, Vincent KC Yan, Xuxiao Ye, Min Ou, Ruibang Luo, Qingpeng Zhang, Bo Tang, Benjamin J Cowling, Ivan Hung, Chung Wah Siu, Ian CK Wong, Reynold CK Cheng, and Esther W Chan. 2021. Drug Repurposing for the Treatment of COVID-19: A Knowledge Graph Approach. *Advanced Therapeutics* 4 (2021), 2100055. Issue 7. https://onlinelibrary.wiley.com/doi/10.1002/adtp.202100055

[29] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[30] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *Comput. Surveys* 55, 9 (2023), 1–35.

[31] Qian Liu, Dejian Yang, Jiahui Zhang, Jiaqi Guo, Bin Zhou, and Jian-Guang Lou. 2021. Awakening Latent Grounding from Pretrained Language Models for Semantic Parsing. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, Online, 1174–1189.

[32] Xiao Liu, Haoyun Hong, Xinghao Wang, Zeyi Chen, Evgeny Kharlamov, Yuxiao Dong, and Jie Tang. 2022. Selfkg: self-supervised entity alignment in knowledge graphs. In *Proceedings of the ACM Web Conference 2022*. 860–870.

[33] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 61–68.

[34] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* abs/1907.11692 (2019). arXiv:1907.11692

[35] Chenhao Ma, Reynold Cheng, Laks VS Lakshmanan, Tobias Grubenmann, Yixiang Fang, and Xiaodong Li. 2019. LINC: a motif counting algorithm for uncertain graphs. *Proceedings of the VLDB Endowment (PVLDB)* 13, 2 (2019), 155–168. https://dl.acm.org/doi/10.14778/3364324.3364330

[36] Chenhao Ma, Yixiang Fang, Reynold Cheng, Laks VS Lakshmanan, Xiaolin Han, and Xiaodong Li. 2023. Accelerating directed densest subgraph queries with software and hardware approaches. *The VLDB Journal* 33, 1 (2023), 207–230. https://doi.org/10.1007/s00778-023-00805-0

[37] Xinnian Mao, Wenting Wang, Yuanbin Wu, and Man Lan. 2022. LightEA: A Scalable, Robust, and Interpretable Entity Alignment Framework via Three-view Label Propagation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 825–838.

[38] Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. 2002. Network motifs: simple building blocks of complex networks. *Science* 298, 5594 (2002), 824–827.

[39] Mohammad Matin Najafi, Chenhao Ma, Xiaodong Li, Laks V.S. Lakshmanan, and Reynold Cheng. 2023. MOSER: Scalable Network Motif Discovery using Serial Test. *Proceedings of the VLDB Endowment (PVLDB)* 17, 3 (2023), 591–603.

[40] Ali Pinar, Comandur Seshadhri, and Vaidyanathan Vishal. 2017. Escape: Efficiently counting all 5-vertex subgraphs. In *Proceedings of the 26th international conference on world wide web*. 1431–1440.

[41] Bowen Qin, Binyuan Hui, Lihan Wang, Min Yang, Jinyang Li, Binhua Li, Ruiying Geng, Rongyu Cao, Jian Sun, Luo Si, Fei Huang, and Yongbin Li. 2022. A Survey on Text-to-SQL Parsing: Concepts, Methods, and Future Directions. arXiv:2208.13629 [cs.CL]

[42] Ryan A Rossi, Nesreen K Ahmed, Aldo Carranza, David Arbour, Anup Rao, Sungchul Kim, and Eunyee Koh. 2020. Heterogeneous graphlets. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 1 (2020), 1–43.

[43] Ping Shao, Yang Yang, Shengyao Xu, and Chunping Wang. 2021. Network embedding via motifs. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16, 3 (2021), 1–20.

[44] Zequn Sun, Wei Hu, and Chengkai Li. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *The Semantic Web–ISWC 2017: 16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017, Proceedings, Part I 16*. Springer, 628–644.

[45] Zequn Sun, Wei Hu, Qingheng Zhang, and Yuzhong Qu. 2018. Bootstrapping entity alignment with knowledge graph embedding.. In *IJCAI*, Vol. 18.

[46] Zequn Sun, Qingheng Zhang, Wei Hu, Chengming Wang, Muhao Chen, Farahnaz Akrami, and Chengkai Li. 2020. A benchmarking study of embedding-based entity alignment for knowledge graphs. *Proceedings of the VLDB Endowment* 13, 12 (2020).

[47] Xiaobin Tang, Jing Zhang, Bo Chen, Yang Yang, Hong Chen, and Cuiping Li. 2020. BERT-INT: a BERT-based interaction model for knowledge graph alignment. *interactions* 100 (2020), e1.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[49] Bailin Wang, Richard Shin, Xiaodong Liu, Oleksandr Polozov, and Matthew Richardson. 2020. RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 7567–7578.

[50] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *The world wide web conference*. 3307–3313.

[51] Kaixin Wang, Cheng Long, Da Yan, Jie Zhang, and HV Jagadish. 2023. Reinforcement learning enhanced weighted sampling for accurate subgraph counting on fully dynamic graph streams. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 1084–1097.

[52] Qihao Wang, Hongtai Cao, Xiaodong Li, Kevin Chen-Chuan Chang, and Reynold Cheng. 2024. From Motif to Path: Connectivity and Homophily. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE.

[53] Zhichun Wang, Qingsong Lv, Xiaohan Lan, and Yu Zhang. 2018. Cross-lingual knowledge graph alignment via graph convolutional networks. In *Proceedings of the 2018 conference on empirical methods in natural language processing*. 349–357.

[54] Y Wu, X Liu, Y Feng, Z Wang, R Yan, and D Zhao. 2019. Relation-Aware Entity Alignment for Heterogeneous Knowledge Graphs. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence.

[55] Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. UnifiedSKG: Unifying and Multi-Tasking Structured Knowledge Grounding with Text-to-Text Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 602–631.

[56] Shuo Yu, Yufan Feng, Da Zhang, Hayat Dino Bedru, Bo Xu, and Feng Xia. 2020. Motif discovery in networks: A survey. *Computer Science Review* 37 (2020), 100267.

[57] Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, et al. 2018. Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3911–3921.

[58] W Zeng, X Zhao, J Tang, and X Lin. 1912. Collective embedding-based entity alignment via adaptive features, CoRR abs. *arXiv preprint arXiv:1912.08404* (1912).

[59] Hao Zhang, Jeffrey Xu Yu, Yikai Zhang, Kangfei Zhao, and Hong Cheng. 2020. Distributed subgraph counting: a general approach. *Proceedings of the VLDB Endowment* 13, 12 (2020), 2493–2507.

[60] Jing Zhang, Bo Chen, Xianming Wang, Hong Chen, Cuiping Li, Fengmei Jin, Guojie Song, and Yutao Zhang. 2018. Mego2vec: Embedding matched ego networks for user alignment across social networks. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 327–336.

[61] Qingheng Zhang, Zequn Sun, Wei Hu, Muhao Chen, Lingbing Guo, and Yuzhong Qu. 2019. Multi-view knowledge graph embedding for entity alignment. *arXiv preprint arXiv:1906.02390* (2019).

[62] Rui Zhang, Bayu Distiawan Trisedya, Miao Li, Yong Jiang, and Jianzhong Qi. 2022. A benchmark and comprehensive survey on knowledge graph entity alignment via representation learning. *The VLDB Journal* 31, 5 (2022), 1143–1168.

[63] Huan Zhao, Xiaogang Xu, Yangqiu Song, Dik Lun Lee, Zhao Chen, and Han Gao. 2018. Ranking users in social networks with higher-order structures. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[64] Huan Zhao, Xiaogang Xu, Yangqiu Song, Dik Lun Lee, Zhao Chen, and Han Gao. 2019. Ranking users in social networks with motif-based pagerank. *IEEE Transactions on Knowledge and Data Engineering* 33, 5 (2019), 2179–2192.

[65] Ziyue Zhong, Meihui Zhang, Ju Fan, and Chenxiao Dou. 2022. Semantics driven embedding learning for effective entity alignment. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2127–2140.