

# Towards Practical Knowledge Graph Alignment

Bing Liu<sup>1,\*</sup>

<sup>1</sup>The University of Queensland, St Lucia QLD 4072, Australia

## Abstract

Entity Alignment (EA) is a primary step of Knowledge Graph fusion. Though neural EA models have been widely studied, they can hardly be deployed in practical applications for expensive annotation costs, limited effectiveness, and poor scalability. Towards practical EA, we investigate (1) how to recognise the most informative data samples for annotation, (2) how to devise a more effective training framework for EA models, and (3) how to divide a large-scale EA task into small subtasks without losing EA effectiveness. Our work would bridge the gap between the EA models designed in laboratory settings and the requirements of industrial applications.

## Keywords

Knowledge Graph, Entity Alignment, Active Learning, Compatibility, Semi-supervised Learning, Task Division

## 1. Introduction

Knowledge Graphs (KGs) store structured knowledge – entities and their relationships – in the form of a graph. Modern Information Retrieval (IR) systems leverage KGs to provide better information access services. For example, search engines prefer to return exact answers stored in KGs instead of entries of documents [1]; Recommender systems exploit the relationships between entities to suggest interesting things to users [2]. Due to the limitations in either knowledge source or construction techniques, most KGs suffer from incompleteness, which limits the impact of KGs on downstream applications. Meanwhile, different KGs often contain complementary knowledge because they are built independently. This phenomenon makes fusing complementary KGs a promising solution for building a more comprehensive KG. Entity Alignment (EA), which identifies equivalent entities (i.e. entity mappings) between two KGs, is essential for KG fusion. For example, Fig. 1 presents two small KGs that contain complementary information related to *Donald Trump*. By recognising equivalent entities of the two KGs (i.e., *Donald Trump*  $\equiv$  *D.J. Trump* and *US*  $\equiv$  *America*), we then can merge the two KGs into a richer one.

With the development of deep learning techniques, neural EA methods become current mainstream direction. Various neural EA models have been devised and achieve current state-of-the-art EA performance [3, 4, 5, 6]. These methods use pre-aligned mappings to learn an

---

Woodstock'22: Symposium on the irreproducible science, June 07–11, 2022, Woodstock, NY

\*Corresponding author.


✉ [bing.liu@uq.edu.au](mailto:bing.liu@uq.edu.au) (B. Liu)

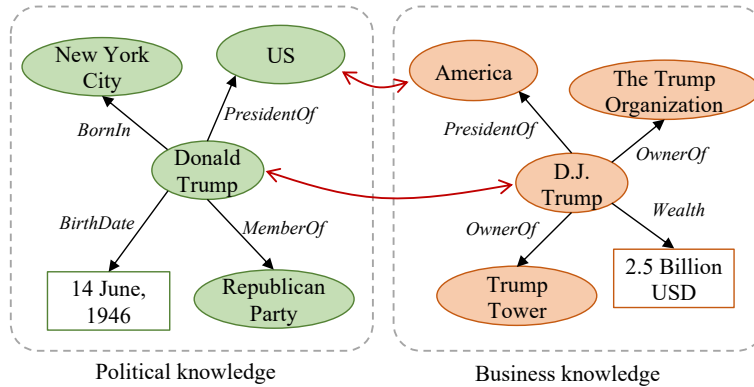
🌐 <https://uqbingliu.github.io/> (B. Liu)

🆔 0000-0002-7858-7468 (B. Liu)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

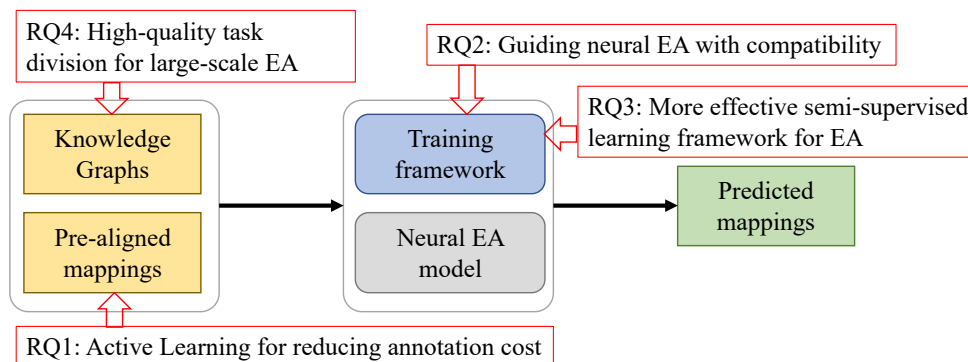
 CEUR Workshop Proceedings (CEUR-WS.org)



**Figure 1:** Example of Entity Alignment. Entity Alignment aims to find equivalent entities in different KGs, e.g.  $Donald\ Trump \equiv D.J.\ Trump$ ,  $US \equiv America$ .

EA model: it encodes entities into informative embeddings and then, for each source entity, selects the closest target entity in the vector space as its counterpart. Though significant progress has been achieved, some problems prevent neural EA methods from being deployed in practical applications: (1) Expensive annotation cost. Typically, these neural EA models rely on a seed alignment as training data. The annotation process of the seed alignment, which involves sampling a set of entities from one KG, and seeking their counterparts from the other KG manually, is very labour-intensive. However, the annotation budget is usually limited in practice. (2) Limited effectiveness. When only the structure information of KGs is available, the existing EA models can hardly achieve satisfying performance, especially when the training data are not sufficient [4]. In addition, the experimental study reported that the neural EA models have much worse performance in industry settings than in laboratory settings even if the attributes are available [5]. Thus, more effort is required to improve the effectiveness of neural EA. (3) Poor scalability. Most neural EA models are evaluated on small KGs sampled from real-world KGs. However, two key problems arise when these models are used for large-scale KGs, i.e. KGs with millions or billions of entities. Problem 1: GPU memory – storing all entity embeddings requires more memory than what is available on the computer infrastructure undertaking the task, easily causing GPUs (which often have a more limited memory than CPU systems) to raise Out-of-Memory exceptions and errors. Problem 2: efficiency – the time required for performing the EA task on large KGs is infeasible, despite running these on powerful GPU infrastructure.

To address these problems, we will investigate: (1) How to reduce the annotation cost of training EA models; (2) How to boost the performance of EA models, especially in few data scenarios; (3) How to apply the neural EA models to large-scale KGs. As such, we are going further towards a practical EA system.



**Figure 2:** Overview of our research questions. Our focus is on: reducing the annotation cost of pre-aligned mappings, more effective training framework, and division of large-scale EA task.

## 2. Methodology

Fig. 2 shows an overview of our Research Questions (RQs). Typically, a neural EA model, taking two KGs and some pre-aligned mappings as input, is optimised in a training framework firstly. Afterwards, it can be used to predict more potential mappings. In the landscape of EA, the neural EA model is under the spotlight, while other involved components are much less concerned. Instead, our focus in RQs is on the other components:

- (RQ1) To reduce the annotation cost, we apply Active Learning to recognise the most informative entities to annotate. Our goal is to achieve the best EA effectiveness with the least annotation cost.
- (RQ2) For improving the performance of EA models, we propose to introduce compatibility as another power of guiding neural EA models apart from labelled data. The learning objectives of an EA model should not only be fitting the labelled data but also making compatible predictions.
- (RQ3) In addition, we devise a more effective semi-supervised learning framework so that we can exploit the unlabelled data to boost the training of EA models.
- (RQ4) To solve the scalability issue of existing EA models, we investigate how to divide a large-scale EA task into a group of small ones without losing EA effectiveness. In this way, the existing EA models can be used to align large KGs.

### RQ1: Active Learning for Neural Entity Alignment.

We seek to reduce the cost of annotating seed alignment data by investigating methods capable of selecting the most informative entities for labelling so as to obtain the best EA model with the least annotation cost: we do so using Active Learning. Active Learning (AL) [7] is a Machine Learning (ML) paradigm where the annotation of data and the training of a model are performed iteratively so that the sampled data is highly informative for training the model. Basically, its steps within one iteration include (1) training the model with existing labelled data; (2) sampling more data, which are thought informative to current model, from the pool of unlabelled data to

annotate; (3) annotating the sampled data manually and adding them to the labelled data; (4) starting a new iteration by updating the model with the updated labelled data.

Though many general AL strategies have been proposed [8, 9], there are some unique challenges in applying AL to EA. The first challenge is **how to exploit the dependencies between entities**. In the EA task, neighbouring entities (context) in the KGs naturally affect each other. For example, in the two KGs of Fig. 1, we can infer *US* corresponds to *America* if we already know that *Donald Trump* and *D.J. Trump* refer to the same person: this is because a single person can only be the president of one country. Therefore, when we estimate the value of annotating an entity, we should consider its impact on its context in the KG. Most AL strategies assume data instances are independent, identically distributed and cannot capture dependencies between entities [7]. In addition, neural EA models exploit the structure of KGs in different and implicit ways [3]. It is not easy to find a general way of measuring the effect of entities on others.

The second challenge is **how to recognise the entities in a KG that do not have a counterpart in the other KG** (i.e., *bachelors*). In the first KG of Fig. 1, *Donald Trump* and *US* are matchable entities while *New York City* and *Republican Party* are bachelors. Selecting bachelors to annotate will not lead to any aligned entity pair. From the perspective of data annotation, recognising bachelors would automatically save annotation budget (because annotators will try to seek a corresponding entity for some time before giving up) and allow annotators to put their effort in labelling matchable entities. This is particularly important for the existing neural EA models, which *only* consider matchable entities for training: thus selecting bachelors in these cases is a waste of annotation budget.

## RQ2: Guiding Neural Entity Alignment with Compatibility

While numerous neural EA models have been devised, they are mainly learned using labelled data only. The *dependencies* between entities, which is the nature of graph data, is under-explored. In an EA task, the *counterparts* of different entities within one KG should be *compatible* – can exist at the same time – w.r.t. the underlying dependencies. Through an experimental study, we verified that more effective EA models make more compatible predictions. Therefore, we argue that making compatible predictions should be one of the objectives of training a neural EA model, other than fitting the labelled data. Unfortunately, compatibility has thus far been neglected by the existing neural EA works.

To fill this gap, we propose a training framework which exploits compatibility to improve existing neural EA models. Few critical problems make it challenging to drive a neural EA model with compatibility: (1) A first problem is how to measure the overall compatibility of all EA predictions. We notice some reasoning rules defined in traditional reasoning-based EA works [10] can reflect the dependencies between entities well. To inherit their merits, we devise a compatibility model which can reuse them. In this way, we contribute one mechanism of combining reasoning-based and neural EA methods. (2) The second problem is how to improve the compatibility of the EA model. Compatibility is measured on the counterparts (i.e. labels) sampled from the EA model, but the sampling process is not differentiable and thus the popular approach of regularising an item in the loss is infeasible. By overcoming these two critical problems, we can guide the training of the EA model with an extra power – compatibility.

### **RQ3: More Effective Semi-supervised Learning Framework for EA**

Semi-supervised learning is an approach to machine learning that combines a small amount of labelled data with a large amount of unlabeled data during training. For example, in self-training, which is a branch of semi-supervised learning, the trained model is applied to the unlabeled data to generate more labelled examples as input for continuing training the model in the next step. Generally, only the labels the model is most confident in are added at each step. In the EA task, some semi-supervised learning strategies have been explored. In the SOTA semi-supervised EA method [11], if two entities, one from the source KG and another from the target KG, are mutually nearest counterpart candidates in the embedding space, they form one pseudo mapping which is added into the training data. Though these semi-supervised learning methods can improve the neural EA models significantly, their generated pseudo-mappings are usually very noisy. To improve their used heuristics, we consider introducing the compatibility measure mentioned in Sec. 2 to assist in generating more accurate pseudo-mappings.

### **RQ4: High-quality Task Division for Large-scale Entity Alignment**

One promising solution to the scalability issue of existing EA models is dividing a large EA task into several small and self-contained subtasks [12, 13]. Each subtask only consists of two small subgraphs produced from the original KGs so that it fits a limited amount of memory. Existing neural EA models can be applied to each subtask to detect unknown mappings according to the included seed mappings, and the mappings found in all subtasks jointly form the final alignment results. Besides solving the memory issue, such a solution also naturally provides the opportunity to further speed up the EA task through parallel processing.

Nevertheless, task division inevitably degrades the overall alignment effectiveness, and notable challenges arise in order to mitigate this side effect. The first challenge is how to guarantee high coverage of potential mappings in the subtasks. The subtasks obtained by dividing a large EA task are processed independently from each other (allowing extreme parallelisation). It is then crucial that equivalent entities are allocated to the same subtask – that is, the task division needs to guarantee a high coverage of the potential mappings in the subtasks. However, it is not straightforward to detect the potential counterparts without using the EA model.

The second challenge is how to achieve high final EA effectiveness. A subtask should contain sufficient *evidence*, such as the KG structure and the supervision signals from seed mappings, in order to achieve high alignment effectiveness. For a set of unmatched entities from one KG, we call the entities that provide alignment evidence as *context entities*, and the subgraph containing unmatched entities and context entities is called *context graph*. How to measure the informativeness of context graphs and how to search the most informative context graphs are both under-explored.

We are going to address these challenges so as to build one general EA division framework, which can derive high-quality EA subtasks.

### 3. Conclusion

The neural Entity Alignment models designed in laboratory environments can hardly be used in practice for expensive annotation cost, limited effectiveness, and poor scalability. To solve these problems, we investigate several open questions: (1) Active Learning for reducing annotation cost; (2) Guiding the training of neural EA models with compatibility; (3) More effective semi-supervised learning framework; (4) High-quality task division for large-scale EA. Our research complements the existing EA works and expands the landscape of EA to the under-explored sub-areas. By driving practical EA further, our work will help produce more complete KGs, which can support better information access services.

### References

- [1] R. Reinanda, E. Meij, M. de Rijke, Knowledge graphs: An information retrieval perspective, *Found. Trends Inf. Retr.* 14 (2020) 289–444. URL: <https://doi.org/10.1561/15000000063>. doi:10.1561/15000000063.
- [2] S. Ji, S. Pan, E. Cambria, P. Marttinen, P. S. Yu, A survey on knowledge graphs: Representation, acquisition, and applications, *IEEE Trans. Neural Networks Learn. Syst.* 33 (2022) 494–514. URL: <https://doi.org/10.1109/TNNLS.2021.3070843>. doi:10.1109/TNNLS.2021.3070843.
- [3] Z. Sun, Q. Zhang, W. Hu, C. Wang, M. Chen, F. Akrami, C. Li, A benchmarking study of embedding-based entity alignment for knowledge graphs, *Proc. VLDB Endow.* 13 (2020) 2326–2340. URL: <http://www.vldb.org/pvldb/vol13/p2326-sun.pdf>.
- [4] X. Zhao, W. Zeng, J. Tang, W. Wang, F. M. Suchanek, An experimental study of state-of-the-art entity alignment approaches, *IEEE Trans. Knowl. Data Eng.* 34 (2022) 2610–2625. URL: <https://doi.org/10.1109/TKDE.2020.3018741>. doi:10.1109/TKDE.2020.3018741.
- [5] Z. Zhang, H. Liu, J. Chen, X. Chen, B. Liu, Y. Xiang, Y. Zheng, An industry evaluation of embedding-based entity alignment, in: A. Clifton, C. Napoles (Eds.), *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020 - Industry Track*, Online, December 12, 2020, International Committee on Computational Linguistics, 2020, pp. 179–189. URL: <https://doi.org/10.18653/v1/2020.coling-industry.17>. doi:10.18653/v1/2020.coling-industry.17.
- [6] X. Mao, W. Wang, Y. Wu, M. Lan, Boosting the speed of entity alignment 10 ×: Dual attention matching network with normalized hard sample mining, in: J. Leskovec, M. Grobelnik, M. Najork, J. Tang, L. Zia (Eds.), *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021, ACM / IW3C2*, 2021, pp. 821–832. URL: <https://doi.org/10.1145/3442381.3449897>. doi:10.1145/3442381.3449897.
- [7] C. C. Aggarwal, X. Kong, Q. Gu, J. Han, P. S. Yu, Active learning: A survey, in: C. C. Aggarwal (Ed.), *Data Classification: Algorithms and Applications*, CRC Press, 2014, pp. 571–606. URL: <http://www.crcnetbase.com/doi/abs/10.1201/b17320-23>.
- [8] B. Settles, *Active Learning*, *Synthesis Lectures on Artificial Intelligence and Machine Learning*, Morgan & Claypool Publishers, 2012. URL: <https://doi.org/10.2200/S00429ED1V01Y201207AIM018>. doi:10.2200/S00429ED1V01Y201207AIM018.

- [9] P. Ren, Y. Xiao, X. Chang, P. Huang, Z. Li, X. Chen, X. Wang, A survey of deep active learning, CoRR abs/2009.00236 (2020). URL: <https://arxiv.org/abs/2009.00236>. arXiv:2009.00236.
- [10] F. M. Suchanek, S. Abiteboul, P. Senellart, PARIS: probabilistic alignment of relations, instances, and schema, Proc. VLDB Endow. 5 (2011) 157–168. URL: [http://www.vldb.org/pvldb/vol5/p157\\_fabianmsuchanek\\_vldb2012.pdf](http://www.vldb.org/pvldb/vol5/p157_fabianmsuchanek_vldb2012.pdf). doi:10.14778/2078331.2078332.
- [11] X. Mao, W. Wang, H. Xu, M. Lan, Y. Wu, MRAEA: an efficient and robust entity alignment approach for cross-lingual knowledge graph, in: J. Caverlee, X. B. Hu, M. Lalmas, W. Wang (Eds.), WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020, ACM, 2020, pp. 420–428. URL: <https://doi.org/10.1145/3336191.3371804>. doi:10.1145/3336191.3371804.
- [12] C. Ge, X. Liu, L. Chen, B. Zheng, Y. Gao, Largeea: Aligning entities for large-scale knowledge graphs, Proc. VLDB Endow. 15 (2021) 237–245. URL: <http://www.vldb.org/pvldb/vol15/p237-gao.pdf>.
- [13] W. Zeng, X. Zhao, X. Li, J. Tang, W. Wang, On entity alignment at scale, The VLDB Journal (2022) 1–25.