2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Exploiting Mention-Entity Graph to Enhance In-context Learning for Collective Entity Linking

Xingyi Li(✉), Boyuan Jia, Mingshuo Chen, Xiang Cheng, and Sen Su

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing, China
{lixingyi2016,jby222,chen.mingshuo,chengxiang,susen}@bupt.edu.cn

Abstract. Entity Linking (EL) aims at mapping mentions to their corresponding entities. It has been shown that in-context learning based approaches can provide better performance. However, they ignore the interdependency between different EL decisions, i.e., the mentions in the same document should be semantically related to each other, leading to inaccuracy of the task. In this paper, we present CIRCLE, a collective entity linking approach via Mention-Entity graph based in-context learning. In CIRCLE, we propose a logic enhanced path information injection method, which leverages comparative and additive logic to enhance the path information. Moreover, we design a submodular function based demonstration selection method which selects the document-level demonstrations considering high coverage of semantic and path information. Furthermore, we design a Tree-of-Thoughts based demonstration format method which uses a four-layer tree structure for hierarchical thinking. Experimental results confirm the effectiveness of our approach.

Keywords: Collective entity linking, In-context learning, Mention-Entity graph, Large language models.

1 Introduction

Knowledge graphs [25, 2, 5, 27], consisting of enormous relational triples, are valuable resources for many downstream natural language processing applications including question answering and recommender systems. Knowledge graph construction has been a widely concerned topic in both academia and industry.

Entity linking (EL) is the core task of knowledge graph construction. Traditional local approaches usually link mentions in a document by assuming them to be independent. The primary drawback of these approaches lies in the neglect of the additional interdependency between different EL decisions, leading to inaccuracy of the entity linking task. As opposed to local approaches, global (a.k.a. collective entity linking) approaches link the mentions in the same document jointly by exploiting the interdependency between them.

Most existing local and global approaches face challenges in disambiguating long-tail entities due to their limited training data. With the development of large language models (LLMs), model can be finetuned based on a small amount of labeled data from the downstream task. In-context learning (ICL) is regarded as a kind of LLM-based approach with three major phases that are demonstration selection, demonstration ordering and demonstration format. Recently, various ICL based entity linking approaches have been proposed, such as ChatEL [10], INSGENEL-ICL [29], and LLMAEL [30]. However, they ignore the interdependency between different EL decisions, leading to inaccuracy of the entity linking task.

In this paper, we present a collective entity linking approach CIRCLE (i.e., exploiting mention-entity graph to enhance in-**C**ontext learn**I**ng fo**R** **C**ol**L**ective **E**ntity linking), which considers the interdependency among the mentions in a document by Mention-Entity graph based in-context learning. In CIRCLE, to capture the interdependency between the mentions, we propose a logic enhanced path information injection method. In this method, we define comparative logic and additive logic to explain the path information from the Mention-Entity graph, and inject them into document-level demonstrations. Moreover, to provide comprehensive and representative guidance for large language models at document-level, we design a submodular function based demonstration selection method, which uses semantic coverage submodular function and path coverage submodular function to select demonstrations. Furthermore, to consider all mentions together with the path information, we design a Tree-of-Thoughts based demonstration format method with four-layer tree structure for each document, which are Mention-Entity graph layer, path information layer, mention layer and candidate entity layer. Based on the tree structure, the hierarchical thinking paradigm assists in better utilizing the four-layer information of each mention.

In summary, our contributions are fourfold:

- We present CIRCLE, a collective entity linking approach via Mention-Entity graph based in-context learning, which considers the interdependency among the mentions in a document.

- We propose a logic enhanced path information injection method which leverages comparative logic and additive logic to explain the path information.

- We design a submodular function based demonstration selection method which selects the document-level demonstrations considering high coverage of semantic and path information. We further devise a Tree-of-Thoughts based demonstration format method which uses a four-layer tree structure for hierarchical thinking.

- We conduct comprehensive experiments on ten public datasets compared with five baselines. The average in-KB micro-F1 on these datasets outperform those of the baselines, confirming the effectiveness of our approach.

2 Related Work

Traditional local approaches focus on resolving mentions independently relying on textual context information from the surrounding words [8, 31], which pay little attention on the interdependency between each target decision. To alleviate this issue, global (i.e. collective) approaches are proposed. Typically, GCEL [14] constructs a simple graph by only including the mentions and their candidates, and runs an iterative weighing process to find the candidate with the largest weight as the best matching. To improve the computation efficiency, NCEL [7] applies Graph Convolutional Network to integrate both local contextual features and global coherence information for entity linking.

Obviously, such approaches struggle to disambiguate long-tail entities due to their limited training data. In-context learning (ICL) is a training-free learning framework which shows strong zero-shot and few-shot performance in entity linking task. ChatEL [10] utilizes LLMs with a three-step framework based on prompt. INSGENEL-ICL [29] feeds a fixed exemplar and task instruction to the LLM as the demonstrations. LLMAEL [30] is a plug-and-play approach to enhance entity linking through LLM data augmentation. However, they link mentions in a document without considering their interdependency, resulting in inaccuracy of the entity linking task.

3 Preliminaries

3.1 In-context Learning

The large language models (LLMs), such as GPT series [6, 22, 1] and LlaMA family [17, 26, 18], are known to have impressive in-context learning abilities. These LLMs have been proven to solve a completely new problem with a small number of examples without any training or finetuning but only with a few examples as instructions or demonstrations, which is called in-context learning (ICL). The generic process of ICL contains three major phases, which are demonstration selection, demonstration ordering and demonstration format [33]. Specifically, from the task dataset, a few examples are chosen to serve as demonstrations first. Then, they are put together in a certain order with created templates. Finally, to format demonstrations, the demonstrations are concatenated together followed by the test instance, which are fed into the LLM to produce the result without parameter updating. In our work, we focus on the demonstration selection and demonstration format phases.

3.2 Submodular Functions

Submodular functions [11] are widely recognized to model notions of diversity, representativeness, and coverage in many applications [4].

Formally, given a set V, the submodular function $F: 2^V \rightarrow \mathbb{R}$, where 2^V denotes the power set of V, F is a submodular function when it satisfies monotonically non-decreasing:

$$F(A \cup \{v\}) - F(A) \geq 0. \tag{1}$$

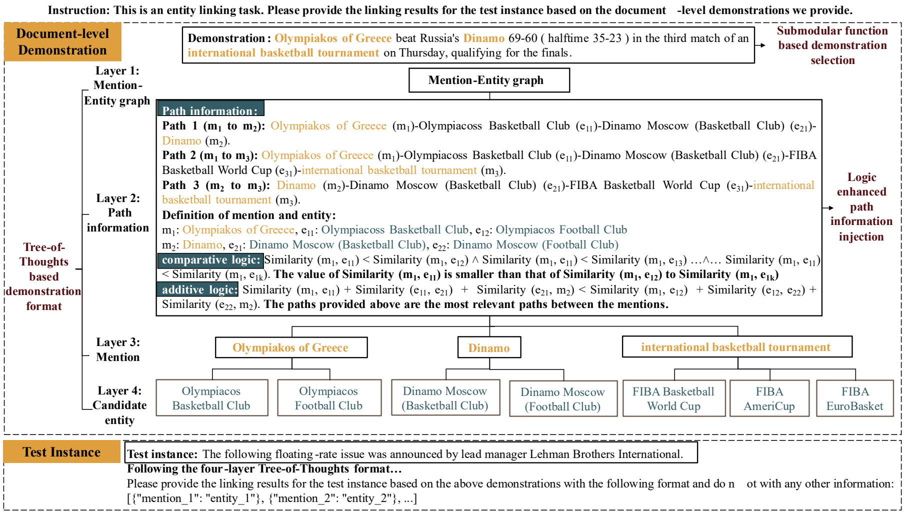

Fig. 1. Overview of our approach CIRCLE.

The diminishing returns property for any $A \subseteq B \subset V$ and $v \in V \backslash B$:

$$F(A \cup \{v\}) - F(A) \geq F(B \cup \{v\}) - F(B). \tag{2}$$

This inequality implies a decreasing incremental gain of inserting element v into the sets of A and B. Moreover, such a function is termed monotone if $F(A) \leq F(B)$. Monotone submodular functions exhibit a rich set of properties. There is one property utilized in this work:

Theorem 1: If F' is a monotone submodular function and $g: \mathbb{R} \rightarrow \mathbb{R}$ is a non-decreasing concave function, then $F = F' \circ g: 2^V \rightarrow \mathbb{R}$ is also a monotone submodular function.

4 Methodology

4.1 Overview

As shown in **Fig. 1**, our approach CIRCLE consists of three phases, namely path information injection, demonstration selection and demonstration format. In the first phase, a logic enhanced path information injection method is proposed, which devises two types of logic to explain the path information. In the second phase, a submodular function based demonstration selection method is proposed to select the document-level demonstrations considering high coverage of semantic and path information. In the third phase, a Tree-of-Thoughts based demonstration format method is proposed which uses four layers to better enhance the learning process of linking all mentions.

4.2 Logic Enhanced Path Information Injection

It is proven that topological information is important for modeling the global interdependency between different entity linking decisions [14]. However, directly incorporating the topology of mention-entity into the demonstrations may lead to comprehension challenges for LLM.

Therefore, we propose a logic enhanced path information injection method, which devises two types of logic for illustrating the path information.

As the basis, we construct a Mention-Entity graph following the existing work [14]. The Mention-Entity graph is a weighted graph $G = (V, E)$, where the node set V contains all mentions in a document and all the possible candidate entities of these mentions. Each node represents either a mention or an entity. Each edge has a weight. An edge between a mention and an entity represents the Compatible relation, while an edge between two entities represents the Semantic-Related relation.

Path Information Generation. To capture the path information from the Mention-Entity graph, we calculate the complementary number for the weight on edge and leverage Dijkstra's algorithm. The new weight between $node_i$ and $node_j$ can be denoted as $Similarity(node_i, node_j)$, where $node_i$ and $node_j$ can be replaced by m_i and e_{ik}. m_i represents the i-th mention, and e_{ik} represents the k-th candidate entity for the i-th mention. In a document, we first fix two mentions as the start and the end nodes and obtain the paths from one mention to another. We then order these paths by calculating the sum of the weights among each path. Finally, the shortest path for each pair of mentions is composed of "m_1-e_{1k}-...-e_{2k}-m_2", in which the interdependency of entities cannot be captured.

For instance, the path information in the example from **Fig. 1** is illustrated in **Fig. 2Fig. 1**.

Logic Rule Design. To further explain the principle of the path information, we devise two types of logic, which are comparative logic and additive logic, respectively.

The comparative logic means the similarity of the mention and its corresponding entities, which is defined as follows:

$$\begin{aligned} Similarity(m_1, e_{11}) &< Similarity(m_1, e_{12}) \\ \wedge\, Similarity(m_1, e_{11}) &< Similarity(m_1, e_{13}) \\ \wedge \ldots \wedge\, Similarity(m_1, e_{11}) &< Similarity(m_1, e_{1k}). \end{aligned} \tag{3}$$

For better explain the comparative logic in natural language, we add "The value of $Similarity(m_1, e_{11})$ is smaller than that of $Similarity(m_1, e_{12})$ to $Similarity(m_1, e_{1k})$" after it.

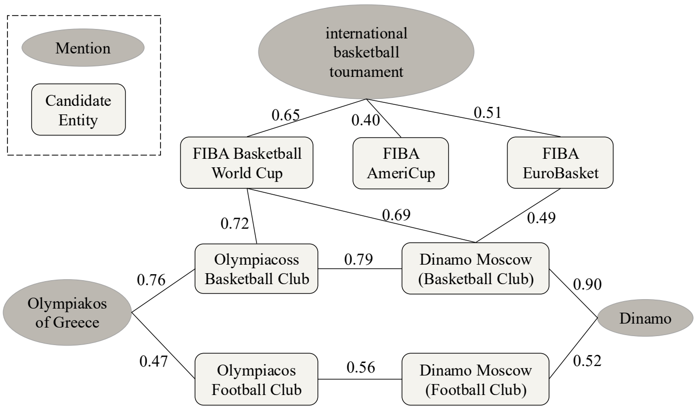

Fig. 2. Mention-Entity graph of the example "Olympiakos of Greece beat Russia's Dinamo 69-60 (halftime 35-23) in the third match of an international basketball tournament on Thursday, qualifying for the finals." in AIDA dataset.

The additive logic means the addition of weights in a path from one mention to another mention. Since the additive logic in each two mentions are the same, we choose the logic from m_1 and m_2 to guide the LLMs in understanding the path information for a document. The following formula defines the additive logic between m_1 and m_2:

$$\begin{gathered} Similarity(m_1, e_{11}) + Similarity(e_{11}, e_{21}) + \\ Similarity(e_{21}, m_2) < Similarity(m_1, e_{12}) + \\ Similarity(e_{12}, e_{22}) + Similarity(e_{22}, m_2). \end{gathered} \tag{4}$$

For better explain the additive logic in natural language, we add "The paths provided above are the most relevant paths between the mentions" after it.

4.3 Submodular Function based Demonstration Selection

Demonstration selection is the initial step of in-context learning. It is proven that the selected demonstrations with a certain amount of representativeness could give more guidance to large language models (LLMs) [16] for achieving better performance. However, quantifying the representativeness of the collective entity linking task is difficult, as it involves the semantic and path representativeness for the document.

Therefore, we propose a submodular function based demonstration selection method, which selects document-level demonstrations with high coverage of both the semantic and path information by using the submodular function.

Semantic Coverage Submodular Function. Since selecting demonstrations based on semantic similarity to the test instance in the embedding space proves to be an effective strategy, we introduce a semantic coverage submodular function at the document-level, which is defined as follows:

$$F_{SC}(S,t) = \sum_{d_i \in S} sim(d_i, t), \tag{5}$$

where $D = \{d_1, d_2, \ldots, d_n\}$ is training set, d_i represents the i-th document in D, S represents the demonstrations selected from D, t represents the test instance. We use Sentence-Bert as the $sim(.)$.

Path Coverage Submodular Function. Since topology is important for document-level demonstrations, we introduce a path coverage submodular function, which is defined as follows:

$$path\_sum(d_i) = \sum_{\substack{m_{ia}, m_{ib} \in M_i \\ a<b}} len(path(m_{ia}, m_{ib})), \tag{6}$$

where $m_{ij} \in M_i$ represents the j-th mention in d_i, $path(.)$ represents the path between m_{ia} and m_{ib}, $len(.)$ represents the path length. We use Dijkstra's algorithm as $len(.)$.

$$F_{PC}(S) = \sum_{d_i \in S} path\_sum(d_i). \tag{7}$$

Finally, we select document-level demonstrations according to the metric $F(S,t)$:

$$F(S,t) = F_{SC}(S,t) + F_{PC}(S). \tag{8}$$

Obviously, $F_{SC}(S,t)$ and $F_{PC}(S)$ behave as the monotone submodular function since they are fixed value regardless of the set. Following the Theorem 1, we can prove $F(S,t)$ is a monotone submodular function. We apply the standard greedy search outlined in Algorithm 1.

Algorithm 1 Submodular Function based Demonstration Selection

Require: Test instance t, candidate set D=$\{d_i\}_{i=1}^{n}$, number of demonstrations K.
Ensure: Selected demonstration set S

Let $S \leftarrow$ empty list.
while $|S| < K$ **do**
    for $d \in D \backslash S$ **do**
        $F(S \cup \{d_i\}, t) - F(S,t) \leftarrow F(S \cup \{d_i\}, t)$
    end for
    $S.append(argmax_d(F(S \cup \{d_i\}, t) - F(S,t)))$
end while
return S

4.4 Tree-of-Thoughts based Demonstration Format

To enhance the demonstration format for effectively guiding LLMs in generating more accurate results, Tree-of-Thoughts (ToT) [32] is proposed to enable exploration over coherent units of text that serve as intermediate steps toward problem solving. However, each mention's linking in a document is treated as a separate branch task by using ToT. That is to say, when linking a mention, the linking results of other mentions cannot be considered, thus failing to achieve collective entity linking.

Therefore, we propose a Tree-of-Thoughts based demonstration format method, which considers all mentions in a document by the tree structure, together with the hierarchical thinking paradigm.

Tree Structure Design. Since there is more than one mention in a document and each mention has path information, it is difficult to refine these mentions with the corresponding augmented information. As a result, directly mixing them together would significantly reduce accuracy. To this end, we devise a four-layer tree structure for formatting. First, the first layer indicates the Mention-Entity graph for the given document-level demonstration. Then, the second layer represents the logic enhanced path information. Next, the third layer means all mentions within a document. Finally, the fourth layer represents all candidate entities for each mention. In this way, the document-level demonstration can be well formatted.

Hierarchical Thinking Paradigm. Given that the ToT only selects one path as the final answer, we generate multiple paths in order to consider all mentions in a document, i.e., hierarchical thinking paradigm. For each document-level demonstration, we start by analyzing from layer 1 to layer 2. This allows the LLM to capture the complete Mention-Entity graph and path information within the document. Subsequently, we determine the entity of the first mention in layer 3, followed by the entity of the second mention, and so on. In this way, we can make accurate decisions with full access to document-level information of demonstrations.

5 Experiments

5.1 Datasets and Evaluation Metrics

To verify the effectiveness of our approach on the entity linking task, we conduct experiments on ten public datasets: MSNBC [9], AQUAINT [19], ACE2004 [23], WNED-CWEB (CWEB) [12], WNED-WIKI (WIKI) [13], KORE50 [15], OKE-2015 (OKE15) [20], OKE-2016 (OKE16) [21], N3-Reuters-128 (R128) [24], and N3-RSS-500 (R500) [24]. The details of these datasets are described in **Table 1**.

Table 1. Statistics of the datasets. # mentions, # doc and # mentions per doc denote the number of mentions, documents and mentions per document, respectively.

Dataset	#mentions	#doc	#mentions per doc
MSNBC	656	20	32.8
AQUAINT	727	50	14.5
ACE2004	257	36	7.1
CWEB	11154	320	34.8
WIKI	6821	320	21.3
KORE50	144	50	2.9
OKE15	536	101	5.3
OKE16	288	173	1.7
R128	650	113	5.8
R500	524	357	1.5

To maintain a fair comparison across datasets, we use the in-KB micro-F1 score as our evaluation metric following the prior work [13]. Specifically, in-KB only considers mentions matching existing knowledge base entities, excluding empty or invalid ones.

5.2 Baselines

In our experiments, we compare CIRCLE with five entity linking baselines. The baselines are categorized based on their underlying backbones. BLINK and ReFinED utilize pretrained language models as their backbones, while ChatEL, LLMAEL and INSGENEL (INSGENEL-R, INSGENEL-ICL^t and INSGENEL-ICL^c) are based on large language models.

- **BLINK** [28] is a two-stage zero-shot linking algorithm.
- **ReFinED** [3] is an efficient zero-shot-capable approach to end-to-end entity linking.
- **ChatEL** [10] is a three-step method to prompt LLMs for entity linking.
- **LLMAEL** [30] leverages LLMs as knowledgeable context augmenters, generating mention centered descriptions as additional input.
- **INSGENEL** [29] enables casual language models to perform entity linking. INSGENEL-R is a retrieval-augmented generative method. INSGENEL-ICL^t and INSGENEL-ICL^c refer to INSGENEL with in-context learning based on text-davinci-003 and code-davinci-002.

To evaluate the effectiveness of introducing the logic rule to explain the path information, we introduce the internal baseline CIRCLE^t, which considers the interdependency of mentions in a document by the path information without injecting the logic rule.

5.3 Implementation Details

We use GPT-4o and Llama-3.3-70B as our backbones. We select the BLINK model as the retrieval model to generate candidate entities following the existing work [28]. For each test instance, we select one document-level demonstration and keep all paths within it. We evaluate the approach using 5-fold cross-validation, where the dataset is partitioned into five equal parts, with each part serving as the test set in turn while the remaining four parts were used for selecting demonstrations. The final performance metric is obtained by averaging the results across all five folds.

5.4 Results

Table 2 and

Table 3 show the results of the entity linking task on datasets with long text and short text, respectively. Fig. 3 shows the average experiment performances of entity linking on ten datasets. The experimental results show that our approach CIRCLE achieves a significant improvement in average in-KB micro-F1 score, outperforming all baselines by 4%, proving the effectiveness of our approach. From

Table 3, we also observe that CIRCLE is better than the baselines particularly in scenarios with limited mentions per document, i.e., on six public datasets. This indicates that CIRCLE effectively exploits the Mention-Entity graph to enhance the ICL. For example, CIRCLE outperforms the ChatEL baseline on KORE50 and R500, with improvement of 13.8% and 8.9%. However, from **Table 2**, for MSNBC, AQUAINT and CWEB datasets, CIRCLE performs slightly worse than ReFinED but is generally on par with them. This proves that CIRCLE has great potential when each document has more mentions.

Table 2. The experiment performances of entity linking on four datasets with long text. The best value is in bold.

Method	Backbone	MSNBC	AQUAINT	CWEB	WIKI
BLINK	BERT	86.2	85.2	69.1	81.1
ReFinED	RoBERTa	**89.1**	**86.1**	**73.8**	84.1
ChatEL	GPT-4	88.1	76.7	-	-
LLMAEL	Llama-3-70B-instruct	86.6	85.2	69.2	81.1
INSGENEL-R	Llama-7B	74.2	-	-	-
INSGENEL-ICL^{t}	Llama-7B	53.3	-	-	-
INSGENEL-ICL^{c}	Llama-7B	47.4	-	-	-
CIRCLE	GPT-4o	88.3	84.6	71.0	85.4
	Llama-3.3-70B	86.3	81.9	66.5	**86.3**

Table 3. The experiment performances of entity linking on six datasets with short text. The best value is in bold.

Method	Backbone	ACE2004	KORE50	OKE15	OKE16	R128	R500
BLINK	BERT	86.0	-	-	-	-	-
ReFinED	RoBERTa	86.4	56.7	78.1	79.4	68.0	70.8
ChatEL	GPT-4	89.3	78.7	75.8	75.2	78.9	82.2
LLMAEL	Llama-3-70B-instruct	86.0	-	-	-	-	-
INSGENEL-R	Llama-7B	-	71.9	64.1	63.3	56.8	45.5
INSGENEL-ICL^{t}	Llama-7B	-	39.2	-	-	-	34.9
INSGENEL-ICL^{c}	Llama-7B	-	39.0	-	-	-	25.4
CIRCLE	GPT-4o	92.7	87.4	85.2	**82.8**	**87.1**	**91.1**
	Llama-3.3-70B	**94.0**	**92.5**	**87.9**	82.6	84.8	89.7

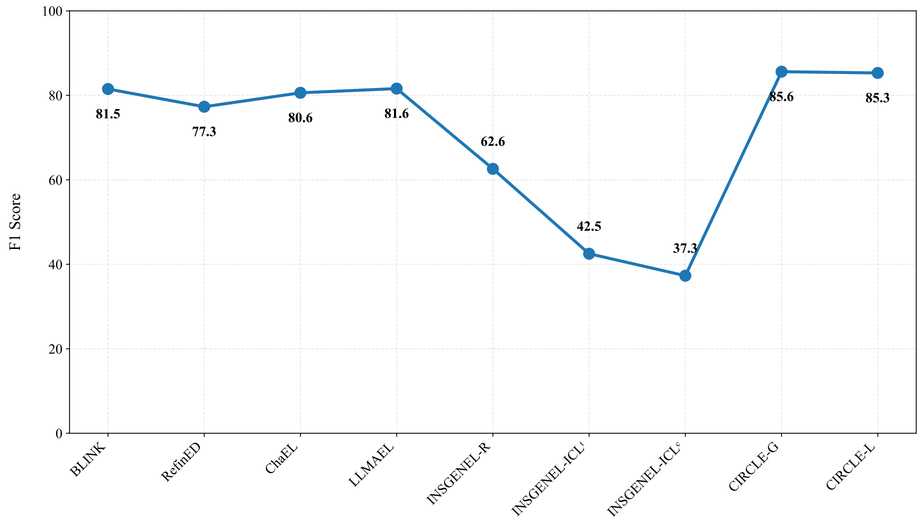

Fig. 3. The average experiment performances of entity linking on ten datasets, which contain short text and long text. CIRCLE-G and CIRCLE-L represent CIRCLE based on GPT-4o and Llama-3.3-70B, respectively.

5.5 Further Analysis

Ablation Study. To analyze the effectiveness of logic-enhanced path information, demonstration selection and format of our approach, we conduct ablation study and report detailed results in **Table 4**. We select the ACE2004 dataset due to its relatively moderate number of mentions per document, which is more representative compared with the other nine datasets. We setup our approach by ablating certain methods:

- W/O path, where no logic-enhanced path information injection method is performed.
- W/O demonstration selection, where no submodular function based demonstration selection method is performed. Instead, we select the demonstrations randomly from the training set.
- W/O demonstration format, where no Tree-of-Thoughts based demonstration format is performed. In other words, we randomly format the path, logic rule, mentions and candidate entities.

From the results in **Table 4**, we observe that: (1) Logic-enhanced path information is significant, as the performance of our approach is degraded when it is not employed. (2) Submodular function based demonstration selection is essential because the performance of our approach improves when a broader coverage of demonstrations are selected. (3) Tree-of-Thoughts based demonstration format is necessary because the performance of our approach is poor when demonstrations are randomly formatted.

Table 4. Ablation study on ACE2004 dataset with the GPT-4o backbone.

Dataset	Method			
	CIRCLE	W/O path	W/O demonstration selection	W/O demonstration format
ACE2004	92.7	87.1	91.2	84.6

Effectiveness of the Logic Rule. From **Table 5**, we can find that CIRCLE achieves better performance on OKE15 than the internal baseline CIRCLE^t, in which the path information is directly introduced in demonstrations without logic rule. This is because simply introducing the path information cannot effectively enable the LLMs to understand.

Table 5. The performance of CIRCLE and CIRCLE^t on OKE15 dataset with the GPT-4o backbone.

Dataset	Method	
	CIRCLE	CIRCLE^t
OKE15	**85.2**	83.8

Impact of the Number of Paths. To illustrate the impact of different number of paths, we conduct further experiments on R128 dataset. From **Table 6**, we can observe that the performance of CIRCLE achieves the best when 30% of the paths are chosen instead of keeping all the paths in each demonstration. This suggests that a smaller number of paths can effectively guide the test instance, especially when the documents are long and contain numerous paths.

Table 6. The performance of CIRCLE with the number of paths on R128 dataset with the GPT-4o backbone.

Dataset	Method					
	CIRCLE	Number of paths (ratio)				
		0.1	0.2	**0.3**	0.5	0.7
R128	**87.1**	86.0	85.5	**88.6**	86.0	83.7

Impact of the Number of Document-level Demonstrations. To illustrate the impact of different number of document-level demonstrations, we conduct further experiments on R128 dataset. From **Table 7**, we can observe that the performance of our approach improves as the shot decreases. This means a smaller number of demonstrations provide sufficient guidance for the test instance.

Table 7. The performance of CIRCLE with the number of demonstrations for a test instance in 1 (CIRCLE), 2, 3, 5-shot settings on R128 dataset with the GPT-4o backbone.

Dataset	Method			
	CIRCLE	Number of demonstrations		
		2	3	5
R128	**87.1**	84.7	83.6	84.3

6 Conclusion

In this paper, we propose CIRCLE, a collective entity linking approach via Mention-Entity graph based in-context learning, which considers the interdependency among the mentions in documents. We propose a logic enhanced path information injection method, which devises two types of logic to explain the path information. Moreover, we design a submodular function based demonstration selection method and Tree-of-Thoughts based demonstration format method to better select and format document-level demonstrations. The experimental results confirm the effectiveness of our approach. In the future, we will extend our approach to cross-lingual settings to build the associations between entities.

Acknowledgments. This work was supported by the National Natural Science Foundation of China (Grant No. 62372051).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)
2. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: Proceedings of ISWC. pp. 722–735 (2007)
3. Ayoola, T., Tyagi, S., Fisher, J., Christodoulopoulos, C., Pierleoni, A.: ReFinED: An efficient zero-shot-capable approach to end-to-end entity linking. In: Proceed- ings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track. pp. 209–220 (2022)
4. Balakrishnan, R., Li, T., Zhou, T., Himayat, N., Smith, V., Bilmes, J.A.: Diverse client selection for federated learning via submodular maximization. In: Proceed- ings of ICLR (2022)
5. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collabo- ratively created graph database for structuring human knowledge. In: Proceedings of SIGMOD. pp. 1247–1250 (2008)
6. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Nee- lakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. In: Proceedings of NIPS (2020)
7. Cao, Y., Hou, L., Li, J., Liu, Z.: Neural collective entity linking. In: Proceedings of COLING. pp. 675–686 (2018)
8. Chisholm, A., Hachey, B.: Entity disambiguation with web links. Trans. Assoc. Comput. Linguistics (2015)
9. Cucerzan, S.: Large-scale named entity disambiguation based on wikipedia data. In: Proceedings of EMNLP. pp. 708–716 (2007)
10. Ding, Y., Zeng, Q., Weninger, T.: Chatel: Entity linking with chatbots. In: Pro- ceedings of LREC/COLING, 20-25 May, 2024, Torino, Italy. pp. 3086–3097 (2024)
11. Fujishige, S.: Submodular functions and optimization. Elsevier (2005)

12. Gabrilovich, E., Ringgaard, M., Subramanya, A.: Facc1: Freebase annotation of clueweb corpora, version 1 (release date 2013-06-26, format version 1, correction level 0). Note: http://lemurproject. org/clueweb09/FACC1/Cited by (2013)
13. Guo, Z., Barbosa, D.: Robust named entity disambiguation with random walks. Semantic Web (2018)
14. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: a graph-based method. In: Proceedings of SIGIR. pp. 765–774 (2011)
15. Hoffart, J., Seufert, S., Nguyen, D.B., Theobald, M., Weikum, G.: KORE: keyphrase overlap relatedness for entity disambiguation. In: Proceedings of CIKM. pp. 545–554 (2012)
16. Levy, I., Bogin, B., Berant, J.: Diverse demonstrations improve in-context compo- sitional generalization. In: Proceedings of ACL. pp. 1401–1422 (2023)
17. Llama: Meta llama (2023), https://llama.meta.com/
18. Llama3: Meta llama 3 (2024), https://llama.meta.com/llama3/
19. Milne, D.N., Witten, I.H.: Learning to link with wikipedia. In: Proceedings of CIKM. pp. 509–518 (2008)
20. Nuzzolese, A.G., Gentile, A.L., Presutti, V., Gangemi, A., Garigliotti, D., Navigli, R.: Open knowledge extraction challenge. In: SemWebEval at ESWC. pp. 3–15 (2015)
21. Nuzzolese, A.G., Gentile, A.L., Presutti, V., Gangemi, A., Meusel, R., Paulheim, H.: The second open knowledge extraction challenge. In: SemWebEval at ESWC. pp. 3–16 (2016)
22. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C.L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P.F., Leike, J., Lowe, R.: Training language models to follow instructions with human feedback. In: Proceedings of NeurIPS (2022)
23. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to wikipedia. In: Proceedings of ACL. pp. 1375–1384 (2011)
24. Röder, M., Usbeck, R., Hellmann, S., Gerber, D., Both, A.: N3 - A collection of datasets for named entity recognition and disambiguation in the NLP interchange format. In: Proceedings of LREC. pp. 3529–3533 (2014)
25. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: Proceedings of WWW. pp. 697–706 (2007)
26. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bash- lykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
27. Vrandečić, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. Com- munications of the ACM (2014)
28. Wu, L., Petroni, F., Josifoski, M., Riedel, S., Zettlemoyer, L.: Scalable zero-shot entity linking with dense entity retrieval. In: Proceedings of EMNLP. pp. 6397– 6407 (2020)
29. Xiao, Z., Gong, M., Wu, J., Zhang, X., Shou, L., Jiang, D.: Instructed language models with retrievers are powerful entity linkers. In: Proceedings of EMNLP. pp. 2267–2282 (2023)
30. Xin, A., Qi, Y., Yao, Z., Zhu, F., Zeng, K., Bin, X., Hou, L., Li, J.: Llmael: Large language models are good context augmenters for entity linking. arXiv preprint arXiv:2407.04020 (2024)
31. Yamada, I., Shindo, H., Takeda, H., Takefuji, Y.: Joint learning of the embedding of words and entities for named entity disambiguation. In: Proceedings of COLING. pp. 250–259 (2016)
32. Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., Narasimhan, K.: Tree of thoughts: Deliberate problem solving with large language models. In: Proceedings of NIPS (2023)

33. Zhao, Z., Wallace, E., Feng, S., Klein, D., Singh, S.: Calibrate before use: Improving few-shot performance of language models. In: Proceedings of ICML. pp. 12697– 12706 (2021)