AELC: Adaptive Entity Linking with LLM-Driven Contextualization

Anonymous ACL submission

Abstract

Entity linking (EL) focuses on accurately associating ambiguous mentions in text with corresponding entities in a knowledge graph. Traditional methods mainly rely on fine-tuning or training on specific datasets. However, they suffer from insufficient semantic comprehension, high training costs, and poor scalability. Large Language Models (LLMs) offer promising solutions for EL, but face key challenges: weak simple-prompt performance, costly fine-tuning, and limited recall and precision due to the lack of LLMs use in candidate generation. Building on this, we introduce a novel framework: **A**daptive **E**ntity **L**inking with LLM-Driven **C**ontextualization. AELC, for the first time, introduces the combination of high-density key information condensation prompt and tool-invocation strategy, using a unified format semantic filtering strategy and an adaptive iterative retrieval mechanism to dynamically optimize the candidate set, significantly enhancing both precision and coverage. Furthermore, we innovatively reformulate the EL task as a multiple-choice problem, enabling multi-round reasoning to substantially improve the model's discriminative capability and robustness. Experiments on four public benchmark datasets demonstrate that AELC achieves state-of-the-art performance. Further ablation studies validate the effectiveness of each module.

1 Introduction

Entity linking plays a vital role in various NLP downstream tasks, including reading comprehension (Andrus et al., 2022) and intelligent question answering (Wang et al., 2022). EL typically involves two stages: *candidate generation* (retrieving potential entities) and *candidate re-ranking* (selecting the most suitable entity from the candidate set). Effective EL improves information retrieval and improves the accuracy and personalization of conversational systems. However, its performance is often constrained by the quality of candidates.

Traditional EL methods typically rely on small pre-trained models for candidate generation (Wu et al., 2019; De Cao et al., 2020). While efficient, these models often lack deep semantic understanding, leading to candidate sets with low recall and precision. When the correct entity is missing from the candidate set, existing methods struggle to cope. Some approaches (Le and Titov, 2019; Arora et al., 2021) overlook this issue during evaluation, while others introduce a 'None' class to bypass it. However, such strategies are unrealistic in practical scenarios, limiting the robustness and applicability of EL in complex settings.

In recent years, the rapid advancement of LLMs has introduced new opportunities for EL, owing to their powerful semantic modeling and reasoning capabilities acquired through training on large-scale datasets. Several studies have explored the integration of LLMs into EL. For example, SumMC (Cho et al., 2022) reformulates the EL task as a multiple-choice problem by generating mention summaries to aid entity selection; ChatEL (Ding et al., 2024) proposes a structured three-stage framework that systematically guides LLMs to produce more accurate linking outputs; and LLMaEL (Xin et al., 2024) enhances EL by enriching input with LLM-generated mention-focused descriptions, while retaining traditional models for task-specific processing. Despite their potential, these approaches still face significant challenges: most rely on simple prompts, which fail to fully exploit the reasoning capabilities of LLMs; fine-tuning or training LLMs remains computationally expensive and impractical in many scenarios; and critically, the candidate generation stage often fails to leverage LLMs, resulting in suboptimal recall and precision, thereby limiting overall EL performance.

To overcome these limitations, we introduce AELC (**A**daptive **E**ntity **L**inking with LLM-Driven

Contextualization), a novel framework designed to fully leverage the capabilities of LLMs for robust EL. The AELC framework consists of three components: LLM-Driven Key Information Condensation (LLM-KIC), Adaptive Semantic Fusion for Dynamic Candidate Generation (ASF-DCG), and LLM-Powered Task Formalization Transformation (LLM-TFT). LLM-KIC refines the document-level entity linking task into paragraph-level subtasks, aligning better with LLM input constraints. It employs a Key Information Condensation Prompt with a built-in chain-of-thought structure to distill the context of a mention into high-density key information. ASF-DCG dynamically generates candidate entities based on an online knowledge graph through a combination of tool invocation, semantic filtering, and adaptive iterative strategies. Finally, LLM-TFT enhances the model's adaptability to the EL task by reformulating it into a multiple-choice format and leveraging in-context learning, thereby significantly improving overall performance.

To comprehensively assess the effectiveness of AELC, we conducted experiments on four widely-used EL benchmark datasets, comparing our approach against both traditional EL models and recent LLM-based methods. Results show that AELC consistently outperforms baselines in linking accuracy. These findings highlight the strength of our framework in enhancing EL performance through the effective integration of external tools and the construction of diverse task-specific prompts.

To summarize, the main contributions of this work are as follows:

1. We propose AELC, a novel entity linking framework that fully leverages LLMs to address key limitations in existing LLM-based EL methods, particularly candidate generation and semantic reasoning.

2. We design three components, LLM-KIC, ASF-DCG, and LLM-TFT, that work in synergy to condense high-density key information, dynamically retrieve high-quality candidates, and reformulate the EL task into a multiple-choice format to enhance LLM adaptability and accuracy.

3. We conduct extensive experiments on four widely-used EL benchmark datasets, demonstrating that AELC outperforms both traditional and recent LLM-based approaches, achieving state-of-the-art performance in linking accuracy.

2 Related Work

2.1 Supervised Entity Linking

Based on embedding models, structural information learning models can extract valuable structured information to perform EL tasks. For example, MTransE (Chen et al., 2016) was the first to propose an approach for EL across knowledge graphs using the TransE (Bordes et al., 2013) embedding model. It predicts linking results based on distances within a unified embedding space. To address the issue of strong reliance on training data in the above methods, a BERT-based model known as BLINK (Wu et al., 2019) emerged as the earliest solution for zero-shot EL tasks. Based on BLINK, a BART-based model namedGENRE (De Cao et al., 2020) was introduced, reportedly outperforming BLINK in performance. Based on probabilistic models, potential mention-candidate pairs are iteratively labeled to form a training dataset, and the linking results are progressively optimized. For instance, the probabilistic model (Fellegi and Sunter, 1969) leverages entity-to-attribute similarity and transforms the EL task into a classification problem, thereby constructing a probabilistic model based on attribute similarity. Graph neural network (GNN)-based methods leverage the advantages of GNNs in identifying isomorphic subgraphs to mine finer-grained structural information for improved EL. For instance, models based on graph attention (Xu et al., 2019) utilize the direct contextual information surrounding the target mention to construct a topic entity graph, thereby transforming the EL task into a graph matching problem. Based on additional information, methods integrate auxiliary data to provide complementary views of the KG structure, including entity attributes (Sun et al., 2017; Tang et al., 2020; D'Auria et al., 2023a), entity descriptions (Sufi, 2022; Yu et al., 2023), and entity names (Zeng et al., 2020; De Cao et al., 2022; D'Auria et al., 2023b). Different models are designed to encode these types of auxiliary information, which serve as pseudo-labeled data for learning a unified structural representation. For instance, this model (Yang et al., 2019) employs graph convolutional networks (GCNs) to combine relational and attribute information of entities in the knowledge graph.

2.2 Unsupervised Entity Linking

τMIL-ND (Le and Titov, 2019) is one of the earliest EL models designed for unlabeled data, leveraging distant supervision to compute compatibility scores between candidate entities and the contextual cues of the target mention. However, the model requires extensive hyperparameter tuning, and its experimental performance is highly sensitive to the wide range of dataset-specific hyperparameters. Zeshel (Logeswaran et al., 2019) utilizes annotated datasets for training and non-annotated datasets for testing, which relies heavily on its inherent semantic understanding to resolve novel target mentions. DSEL (Fan et al., 2015) leverages entity descriptions from Wikipedia articles, generates a large corpus of weakly annotated data, and feeds it to a classifier for linking newly discovered target mentions. Eigentheme (Arora et al., 2021) is a lightweight and scalable EL approach. In geometric space, it assumes that target mentions within a document reside in a low-rank subspace of the complete embedding space formed by candidate entity lists. This subspace is identified using singular value decomposition (SVD), and linking is performed based on the distance between candidate entities and the identified subspace. SumMC (Cho et al., 2022) is a fully unsupervised model that first generates a mention-conditioned summary of the surrounding context, and then reframes the EL task as a multiple-choice question, selecting the correct entity from a predefined list of candidates. ChatEL (Ding et al., 2024) is a structured, three-step framework that systematically guides large language models to generate accurate outputs for entity linking tasks. GEMEL (Shi et al., 2024) is a generative framework for MEL that uses LLMs to directly generate target entity names. LLMaEL (Xin et al., 2024)enhances entity linking by augmenting input with mention-focused descriptions generated by LLMs, while keeping traditional models for task-specific processing.

3 Preliminary

Let D be a single document from the document collection $\mathcal{D}$. Building upon the foundation laid by previous work in entity linking, we assume that the relevant information pertaining to target mentions in document D has been obtained through a named entity recognizer. Thus, let $\mathcal{M}_D = \{m_1, m_2,...,m_n\}$ represent the set of n target mentions contained within document D. Let $\mathcal{E}$ be the set of all entities in the dynamic online knowledge graph $\mathcal{G}$. Let $\mathcal{C}$ be the candidate set for the target mention, where $\mathcal{C} \subseteq \mathcal{E}$.

The input Ψ for unsupervised entity linking comprises a document D containing target mentions $\mathcal{M}_D$ and an online source knowledge graph $\mathcal{G}$. The goal of this task is to find an equivalent candidate c_j for the target mention m_i without pre-training, where candidate c_j is drawn from $\mathcal{G}$:

$$\Psi = \{(m_i, c_j) | m_i \in \mathcal{M}_D, c_j \in \mathcal{E}, m_i \leftrightarrow c_j\}, \quad (1)$$

where $m_i \leftrightarrow c_j$ represents target mention m_i and the candidate entity c_j are equivalent, i.e., m_i and c_j refer to the same real-world object.

4 Approach

In this section, we introduce AELC with the aim of improving the performance of conventional unsupervised EL methods. The overview of the framework is shown in Figure 1.

4.1 Overview

EL typically involves two core steps: candidate generation and candidate re-ranking. Traditional methods often rely on small-scale pre-trained models to retrieve candidates from static knowledge graphs. However, due to limited semantic understanding, these models struggle to achieve high accuracy and coverage. Recent LLM-based approaches have shown promise, yet many fail to leverage the powerful reasoning and comprehension capabilities of LLMs during the candidate generation phase, resulting in low recall and precision, and consequently, suboptimal performance in real-world scenarios. To address these challenges, our framework introduces a key information condensation prompt to enhance mention-level understanding. In addition, semantic filtering and an adaptive iterative retrieval strategy are employed to dynamically refine candidate sets, improving their accuracy. Finally, we reformulate the re-ranking task as a multiple-choice problem, further enhancing linking precision. In Section 4.2, we detail how high-density key information is extracted for mentions. Section 4.3 presents our candidate generation process, incorporating tool-based retrieval, semantic filtering, and adaptive iteration. Section 4.4 explains how we transform the re-ranking task into a multiple-choice format to improve EL efficiency.

4.2 LLM-Driven Key Information Condensation

Decompose Task Granularity by *split(mention)*. As a foundational model, LLMs demonstrate pow-

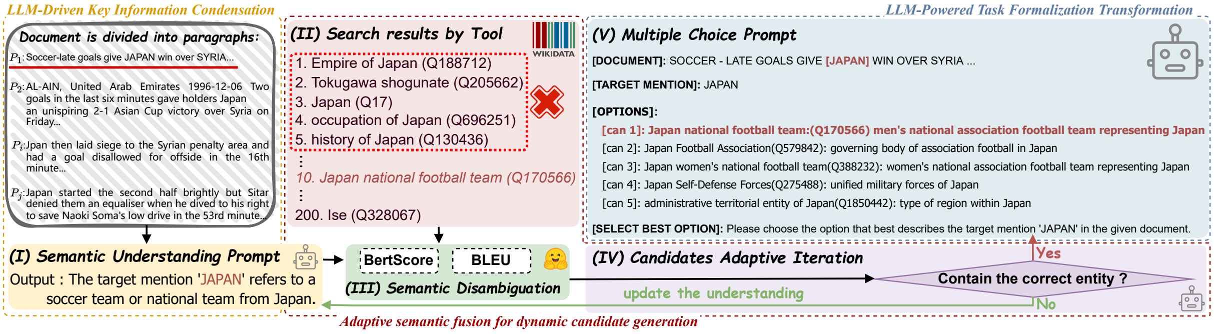

LLM-Driven Key Information Condensation
Document is divided into paragraphs:
P_1: Soccer-late goals give JAPAN win over SYRIA...
P_2: AL-AIN, United Arab Emirates 1996-12-06 Two goals in the last six minutes gave holders Japan an unispiring 2-1 Asian Cup victory over Syria on Friday...
P_i: Jpan then laid siege to the Syrian penalty area and had a goal disallowed for offside in the 16th minute...
P_j: Japan started the second half brightly but Sitar denied them an equaliser when he dived to his right to save Naoki Soma's low drive in the 53rd minute...
(I) Semantic Understanding Prompt
Output : The target mention 'JAPAN' refers to a soccer team or national team from Japan.
(II) Search results by Tool
WIKIDATA
1. Empire of Japan (Q188712)
2. Tokugawa shogunate (Q205662)
3. Japan (Q17)
4. occupation of Japan (Q696251)
5. history of Japan (Q130436)
10. Japan national football team (Q170566)
200. Ise (Q328067)
BertScore
BLEU
(III) Semantic Disambiguation
Adaptive semantic fusion for dynamic candidate generation
LLM-Powered Task Formalization Transformation
(V) Multiple Choice Prompt
[DOCUMENT]: SOCCER - LATE GOALS GIVE [JAPAN] WIN OVER SYRIA ...
[TARGET MENTION]: JAPAN
[OPTIONS]:
[can 1]: Japan national football team:(Q170566) men's national association football team representing Japan
[can 2]: Japan Football Association(Q579842): governing body of association football in Japan
[can 3]: Japan women's national football team(Q388232): women's national association football team representing Japan
[can 4]: Japan Self-Defense Forces(Q275488): unified military forces of Japan
[can 5]: administrative territorial entity of Japan(Q1850442): type of region within Japan
[SELECT BEST OPTION]: Please choose the option that best describes the target mention 'JAPAN' in the given document.
(IV) Candidates Adaptive Iteration
Contain the correct entity ?
Yes
No
update the understanding

Figure 1: **The overview of AELC**, which comprises three core modules: LLM-Driven Key Information Condensation, Adaptive Semantic Fusion for Dynamic Candidate Generation, and LLM-Powered Task Formalization Transformation.

erful NLP capabilities. However, its limitation on input context length limits the range and efficiency of its applications. This restriction in text length may lead to a truncated output, thereby affecting the integrity and accuracy of EL.

For this issue, we were inspired by the least-to-most prompt method (Zhou et al., 2022), which enhances EL by decomposing document-level tasks into sentence-level tasks, as shown by the gray box in Figure 1. Given a document $\mathcal{D}$, we use target mentions as the basis for task segmentation, dividing the entire document into a paragraph set $\mathcal{P}_D$, where each paragraph P_i contains a single mention m. The process follows a repetitive linking paradigm outlined in Algorithm 1.

Algorithm 1 : Sentence-level EL

```
Input: A document $\mathcal{D}$; a mention set $\mathcal{M}_D$ and its
    mentions $m$, $m \in \mathcal{M}_D$; a LLM model $\mathcal{L}$;
        a template $\mathcal{T}$ for EL prompting.
Output: A candidate set $Cans$ for mention $m(s)$.
// Divide $\mathcal{D}$ based on $\mathcal{M}_D$
for $m \in \mathcal{M}_D$ do
    $s(m) \leftarrow split(D)$
end for
// Get a sentence set $\mathcal{S}_D$
$\mathcal{S}_D \leftarrow$ a sentence set for $\mathcal{D}$;
        $s \in \mathcal{S}_D$
         the sentence's mention $m(s)$
// Traversing $\mathcal{S}_D$ for EL
for $s \in \mathcal{S}_D$ do
    $\mathcal{P} \leftarrow \mathcal{T}(s, m(s))$;
    $Cans(s) \leftarrow \mathcal{L}(\mathcal{P})$
end for
```

Key Information Condensation Prompt. Due to the powerful understanding capabilities of LLMs, we use them to condense the key information from the context. We provide the mention and context for LLMs. First, we design a specific task-oriented Chain-of-Thought to guide the LLM's focus on the key information of the mention, preventing it from summarizing noisy or redundant information. Second, to ensure stable output from the LLM, we leverage its in-context learning ability and construct a well-designed key information example as a demonstration, specifying the output format for the LLMs. Then, based on the prompt and contextual information, the LLM condenses the key information for the mention, thereby enhancing its semantic features.

4.3 Adaptive Semantic Fusion for Dynamic Candidate Generation

Candidate Search Tool-Based. We use the online version of Wikidata as the target knowledge graph. However, the online Wikidata functions as an entity-level search engine. For example, searching for sentences like '*Soccer-late goals give JAPAN win over SYRIA.*' or '*The target mention JAPAN refers to a soccer team ...*' does not directly retrieve the entity '*Japan national football team (Q170566)*'. To address this issue, we used Wikidata to search for mention (*JAPAN*) from the sentence and obtain all relevant candidates, as shown in Figure 1 (II). Here, we limit the number of candidates to no more than 200.

Candidate Filtering Strategy. Through the candidate search process, we obtained 200 candidates related to the target mention '*JAPAN*'. The information for each candidate includes its name, Qid, description, and other attributes. However, as shown in the red dashed box in Figure 1 (II), the top-5 search results include entities that are irrelevant to the target mention, such as the Japanese nation and history. To further reduce irrelevant candidates, we adopted a filtering mechanism based on the similarity between the key information of the mention and the descriptions of the candidates. We use BLEU (Papineni et al., 2002) and BERTScore (Reimers and Gurevych, 2019) as metrics to calculate the similarity of information-description pairs, as shown in Figure 1 (III), selecting the top-5 candidates with the highest similarity scores, forming the candidate set $\mathcal{C}$.

Candidate Adaptive Iteration. To further optimize the candidate filtering process, after obtaining the candidate set $\mathcal{C}$, we use LLMs to determine whether the candidate set contains the correct entity corresponding to the mention. As shown in Figure 1 (IV), specifically, if the set includes the correct entity, the process moves to the next module. If not, the current set is fed back into the 'Key Information Condensation' stage, integrating the candidates into a new prompt. The LLM then generates the key information related to the mention based on this feedback and iteratively updates the candidate set based on the newly generated information. This process continues until LLMs confirm that the correct entity exists within the candidate set, at which point the iteration stops.

4.4 LLM-Powered Task Formalization Transformation

The LLM refers to a deep neural network model trained on extensive datasets, which demonstrates enhanced NLP capabilities and superior generative performance compared to traditional machine learning models. Consequently, we transform the string matching task, typically handled by ML models, into a multiple-choice task using LLMs.

Multiple-choice Prompt. As shown in the blue box in Figure 1 (V). The prompt consists of four components: [SENTENCE] refers to the context of mention, [TARGET MENTION] represents the mention to be linked, [OPTIONS] includes all the choices within the candidate set $Cans$ for the multiple-choice question, and [SELECT BEST OPTION] indicates the requirement for the multiple-choice question. Based on this multiple-choice prompt, LLMs are employed for selection.

5 Experiments

5.1 Datasets

We conducted experiments on four commonly used English datasets in EL. The statistics of these datasets are shown in Table 1. The categories *easy*, *medium*, *hard* and *none* each represent the different candidates for a mention.

In order to better measure the performance of our framework, we employed the same dataset as the baseline and conducted experiments on four English datasets in the unsupervised entity linking task: AIDA-CoNLL-testb (AIDA-B)[1], WNED-Wiki[2], WNED-CWEB[2], WikiHow-Wikidata (Wiki-Wiki)[3]. Detailed introductions and statistics of the datasets as shown in the Appendix A.

5.2 Baselines

We selected previous EL models and a series of LLMs as baselines. Specifically, for previous EL models, we opted for the first annotation-free EL model, τMIL-ND (Le and Titov, 2019), the pioneering unsupervised EL model Eigentheme (Arora et al., 2021), and the LLM-based EL approach SumMC (Cho et al., 2022), ChatEL (Ding et al., 2024), GEMEL (Shi et al., 2024) and LLMaEL (Xin et al., 2024). For LLMs, we chose DeepSeek (DeepSeek-AI et al., 2025), GPT-4 (Achiam et al., 2023), Qwen2-7B-Instruct (Yang et al., 2024), and Llama-2-13b-chat-hf (Touvron et al., 2023). For further information of the baselines, please refer to Appendix B.

5.3 Implementation Details

Knowledge Graph. In our framework, we primarily consider the Wikidata[4]. Wikidata is a knowledge graph complementing Wikipedia[5] (providing rich encyclopedic information about world entities), which structures and organizes this encyclopedic knowledge in relational triples.

Experimental Setups. We use a series of LLMs, where the temperature parameter is set to 0.75 (for consistency in fixed output formats) and the maximum token length for the input is set to 256. We use 32 shots in the semantic understanding prompt and 2 shots in the multiple choice prompt and the maximum number of iterations was limited to 5 for the candidates adaptive iteration. Wikidata is used as the source KG, and the number of search results is set to 200. We use precision @1 to evaluate EL effectiveness in all experiments.

[1] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/ambiverse-nlu/aida/downloads

[2] http://dx.doi.org/10.7939/DVN/10968

[3] https://drive.google.com/file/d/1Oebe1sbbixX7FWHX813diCdqReh1IyWH/view

[4] https://wikidata.org/

[5] https://en.wikipedia.org/

Datasets	Details of the mention					Details of the document	
	easy	medium	hard	none	overall	max_mention	overall
AIDA-B	2534 (57%)	1110 (25%)	621 (15%)	148 (3%)	4413	96	231
WNED-WIKI	2731 (41%)	1475 (22%)	1722 (26%)	766 (11%)	6694	46	318
WNED-CWEB	4667 (42%)	3056 (28%)	2653 (24%)	664 (6%)	11040	37	320
Wiki-Wiki	2727 (24%)	8560 (76%)	0 (-%)	0 (-%)	11287	3	7097

Table 1: **Datasets and their statistics.** The categories *easy*, *medium*, *hard*, and *none* each represent the different candidates for a mention: *easy* means the correct answer is the first candidate; *medium* means the correct answer is included but not first; *hard* means the correct answer is absent from the candidates; *none* means no candidates are available. Additionally, max_mention refers to the maximum number of mentions in a single document.

MODELS	AIDA-B				WNED-Wiki				WNED-Cweb				Wiki-Wiki			
	easy	med	diff	All	easy	med	diff	All	easy	med	diff	All	easy	med	diff	All
τMIL-ND	0.70	0.19	0.45	0.45	-	-	-	0.13	-	-	-	0.27	-	-	-	0.31
Eigentheme	0.86	0.50	-	0.62	0.82	0.47	-	0.44	0.77	0.41	-	0.29	0.61	0.53	-	0.50
SumMC	0.80	0.71	-	0.64	0.81	0.65	-	0.47	0.75	0.60	-	0.48	0.62	0.80	-	0.76
ChatEL	0.82	-	-	0.64	0.77	-	-	0.57	0.71	-	-	0.61	0.75	-	-	0.67
GEMEL	0.80	-	-	0.63	0.72	-	-	0.51	0.76	-	-	0.69	0.73	-	-	0.62
LLMaEL	0.86	-	-	0.69	0.85	-	-	0.66	0.75	-	-	0.63	0.73	-	-	0.62
DeepSeek$_{doc}$	0.31	0.29	0.27	0.35	0.48	0.39	0.35	0.46	0.40	0.32	0.24	0.31	0.36	0.33	0.27	0.32
DeepSeek$_{sen}$	0.70	0.66	0.68	0.67	0.63	0.52	0.51	0.55	0.64	0.59	0.60	0.61	0.66	0.67	0.59	0.64
GPT-4$_{doc}$	0.29	0.26	0.25	0.27	0.12	0.10	0.10	0.11	0.22	0.20	0.20	0.21	0.18	0.15	0.14	0.16
GPT-4$_{sen}$	0.68	0.66	0.65	0.66	0.57	0.54	0.50	0.54	0.64	0.63	0.62	0.63	0.60	0.58	0.59	0.59
Llama$_{doc}$	0.22	0.18	0.10	0.17	0.16	0.15	0.13	0.15	0.15	0.23	0.12	0.17	0.34	0.32	0.27	0.31
Llama$_{sen}$	0.64	0.37	0.22	0.41	0.61	0.51	0.54	0.55	0.59	0.54	0.28	0.47	0.62	0.64	0.57	0.61
Qwen$_{doc}$	0.26	0.22	0.14	0.21	0.46	0.34	0.17	0.32	0.36	0.28	0.11	0.25	0.24	0.21	0.23	0.23
Qwen$_{sen}$	0.66	0.52	0.26	0.48	0.62	0.43	0.31	0.45	0.64	0.59	0.18	0.47	0.59	0.56	0.58	0.58
w/o LLM-KIC	0.40	0.35	0.33	0.36	0.26	0.22	0.21	0.23	0.33	0.27	0.28	0.29	0.26	0.25	0.23	0.25
w/o ASF-DSG	0.81	0.70	0.65	0.72	0.81	0.68	0.66	0.72	0.76	0.63	0.67	0.69	0.64	0.76	0.70	0.70
w/o LLM-TFT	0.86	0.74	0.78	0.79	0.84	0.69	0.69	0.74	0.79	0.63	0.67	0.70	0.66	0.80	0.76	0.74
AELC$_{DeepSeek}$	**0.90**	**0.78**	**0.80**	**0.82**	**0.93**	**0.67**	**0.61**	**0.72**	**0.86**	**0.71**	**0.65**	**0.74**	**0.79**	**0.73**	**0.66**	**0.75**
AELC$_{GPT-4}$	0.89	0.76	0.80	0.82	0.77	0.66	0.51	0.65	0.83	0.69	0.64	0.72	0.73	0.64	0.61	0.66
AELC$_{Llama}$	0.82	0.63	0.57	0.67	0.74	0.60	0.50	0.61	0.79	0.66	0.63	0.69	0.69	0.60	0.59	0.63
AELC$_{Qwen}$	0.86	0.72	0.64	0.74	0.92	0.61	0.50	0.68	0.81	0.68	0.57	0.69	0.72	0.61	0.61	0.65

Table 2: **Comparison with Baselines.** EL effectiveness assessed through precision@1. med and diff are abbreviations for *medium* and *hard*, respectively. $Model_{doc}$ defines mention context as the document, whereas $Model_{sen}$ restricts it to the located sentence. LLM-KIC, ASF-DSG, and LLM-TFT are the abbreviations of the three modules. Bold fonts denote the best methods.

5.4 Main Results

There are three question modes (*easy*, *medium* and *hard*) for EL (see Table 1 for details). We continue to employ the results of τMIL-ND and Eigentheme from (Arora et al., 2021), utilizing publicly available datasets. The results of SumMC are collected from (Cho et al., 2022). The specific results are presented in Table 2.

Comparison with Previous EL Models. AELC achieves more competitive performance than other unsupervised EL models on four datasets.

Eigentheme performs well in processing Easy tasks, mainly due to its effective utilization of relationships between mentions and its mastery of global context, thereby highlighting the core information of mentions and significantly improving efficiency in processing Easy tasks. However, Eigentheme neglects the contextual information within documents, which limits its performance in handling other types of tasks.

SumMC shows high performance in processing Medium tasks, with accuracy improvements ranging from 2% to 45%. This is due to SumMC's use of LLMs to compress document content into concise sentences directly related to mentions, which not only enriches the contextual information around the mentions, but also enhances its ability to solve more complex problems. However, the SumMC model has not fully considered the importance of semantic disambiguation, which is an area that needs further improvement.

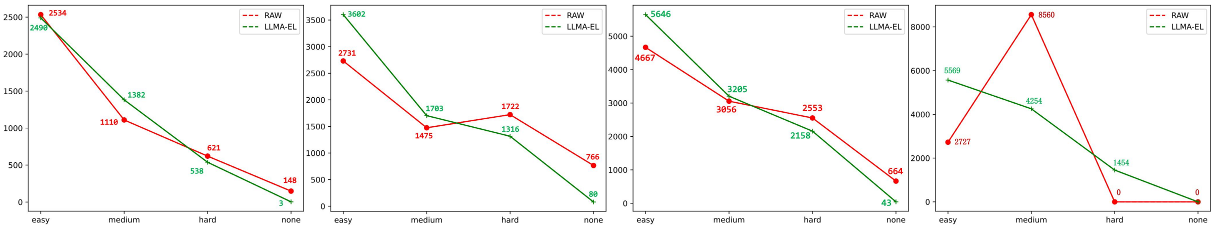


Figure 2: **Compare the quality of candidates for all datasets.** The horizontal axis represents the types of candidates. The vertical axis indicates the number of candidates. The red line shows the distribution of candidates in the raw dataset, while the green line shows the changes in the distribution of candidates after AELC processing.

In existing EL models, tasks without correct options or missing options are usually ignored and not processed. The AELC, however, retrieves the latest and most accurate information by searching online Wikidata, using these search results as candidate options. AELC uses an adaptive iterative strategy to optimise the candidate options, which significantly reduces the proportion of missing problems and improves the accuracy of the processed tasks. This method not only complements the shortcomings of the traditional EL model, but also enhances the performance of the model in handling complex tasks.

Comparison with LLMs. Table 2 evaluates the performance of various LLMs on the EL tasks. Specifically, mentions within documents are initially labeled as [mention]. Subsequently, we categorize the context of the mention into two levels: document-level and sentence-level. Given a set of candidates, the models are tasked with selecting the candidate most closely related to the [mention] based on their understanding of the context. The experimental results indicate that DeepSeek demonstrates the best performance on the EL task, with an average ACC that is 2%-15% higher than other models. Compared to the document-level model ($Model_{doc}$), the sentence-level model ($Model_{sen}$) exhibits superior results. This is likely due to the limitations of input text length, where longer contexts may introduce noise to the mentions, thus reducing the accuracy of EL.

Comparison of AELC Versions. Table 2 presents the results of the AELC model in the absence of different modules, providing an in-depth evaluation of the contributions of each component to the overall performance. *w/o LLM-TFT* resulted in a 2%-13% decrease in accuracy. This indicates that the introduction of multi-choice prompts significantly enhances the LLM's understanding of task context and its ability to accurately match candidates. *w/o ASF-DCG* led to an average accuracy reduction of 9.5%, reflecting that traditional static retrieval methods struggle to effectively capture deep semantic associations within the context. *w/o LLM-KIC* caused a substantial accuracy drop of 41% to 50%. This significant performance degradation validates the crucial role of the adaptive iterative strategy and multi-round interactive optimization in constructing high-quality candidates.

The Quality of Candidates. In the analysis presented in Figure 2, by comparing the number of candidates across different categories, we observe that AELC significantly enhances the quality of candidates. Specifically, on the AIDA, WNED-WIKI, and WNED-CWEB datasets, the implementation of AELC markedly improved the number of candidates categorized as *easy* and *medium*, while significantly reducing those classified as *hard* and *none*. Performance on the Wiki-Wiki dataset appears relatively average, which may be due to the dataset being manually annotated and containing only *easy* and *medium* categories. This discrepancy could stem from the inherent error rate of LLMs reading comprehension and the stringent search requirements of Wikidata, resulting in some mentions originally belonging to the *easy* and *medium* categories being incorrectly classified as *hard*.

5.5 Further Analysis

Effect of key information Condensation Prompt. In Table 3, we compared different key information condensation prompts. The $Prompt_{summary}$ utilizes LLMs to generate summaries for mentions and uses Wikidata to retrieve candidates. $Prompt_{example}$ builds on $Prompt_{summary}$ by adding 32 contextual learning examples from Wikidata to facilitate in-context learning in LLMs. $Prompt_{mention}$ directly searches for mentions in Wikidata. $Prompt^{*}_{X}$ performs a secondary search using mentions when the summary retrieval yields no results. The results show that $Prompt_{mention}$ achieves the best candidate quality, indicating

Prompts	64 tokens		128 tokens	
	L_{sen}	L_{men}	L_{sen}	L_{men}
$Prompt_{summary}$	0.14	0.16	0.15	0.18
$Prompt^{*}_{summary}$	0.27	0.30	0.31	0.38
$Prompt_{example}$	0.31	0.33	0.30	0.32
$Prompt^{*}_{example}$	0.56	0.57	0.56	0.58
$Prompt_{mention}$	0.65	0.66	0.67	0.68

Table 3: **Effect of Tool Adaptation.** X tokens represent the length used to segment document, L_{sen} represents linking all mentions in a sentence at once; L_{men} means linking only one mention in a sentence.

Similarity	$Prompt_{summary}$			
	64 tokens		128 tokens	
	L_{sen}	L_{men}	L_{sen}	L_{men}
BLEU	0.54	0.55	0.53	0.58
BERTScore	0.60	0.61	0.62	0.64

Similarity	$Prompt_{example}$			
	64 tokens		128 tokens	
	L_{sen}	L_{men}	L_{sen}	L_{men}
BLEU	0.60	0.62	0.62	0.65
BERTScore	0.64	0.66	0.67	0.69

Table 4: **Effect of Candidate Filtering.** BLEU and BERTScore are similarity metrics.

that overly lengthy search fields may degrade retrieval performance. $Prompt_{example}$ outperforms $Prompt_{summary}$, demonstrating that Wikidata provides more updated and accurate entity information. Across all prompts, the precision of L_{men} is higher than that of L_{doc}, further validating the effectiveness of advanced tools in improving information retrieval quality.

Effect of Candidate Filtering Strategy. In Table 4, we employ BLEU (Papineni et al., 2002) and BERTScore (Reimers and Gurevych, 2019) to calculate $sim(keyinformation, description)$ for filtering candidates, where $keyinformation$ is the output of the semantic condensation prompt and $description$ is candidate's attribute in wikidata. Primarily, the candidates set filtered by similarity significantly improves the precision of linking. Specifically, in $Prompt_{summary}$, precision saw a twofold increase, and in $Prompt_{example}$, precision was on average boosted by 5.5%. Second, the effect of using BERTScore is significantly better than BLEU. In $Prompt_{summary}$, precision was on average increased by 6.8%, and in $Prompt_{example}$, there was an average improvement of 4.3%. Finally, the results show that Men is more accurate than Doc. The experimental results demonstrate that candidates filtered based on the similarity between key informations and descriptions, extracted using semantic condensation prompts, show significantly improved quality.

Datasets	Str. Mat.	Mul. Cho.
AIDA-B	0.67	0.85
WNED-Wiki	0.44	0.73
WNED-Cweb	0.42	0.73
Wiki-Wiki	0.60	0.76

Table 5: **Effect of Multiple Choice.** Mul. Cho. and Str. Mat. are short for Multiple Choice and String Match.

Effect of Multiple-choice Prompt. As illustrated in Table 5, the use of multiple choice prompt for linking outperforms the traditional string matching approach. In the WNED-Cweb dataset, there is a maximal precision increase of 31%. Due to the dataset containing a large volume of single-character mentions (e.g., m), which challenge traditional models' filtering capabilities. However, LLMs improve precision by using their semantic understanding abilities with the multiple-choice prompt. In the wikiwiki dataset, the minimum precision increase is 16%, as the mentions are mainly tangible daily objects (e.g., water) with few semantically ambiguous ones. This indicates that LLMs possess a vast knowledge base and excel in semantic understanding, surpassing traditional matching models. Leveraging multi-choice prompts allows LLMs to more accurately and efficiently harness their capabilities in entity linking tasks, fully tapping into their deep semantic understanding and significantly enhancing overall performance.

6 conclusion

In this work, we introduce a novel framework,**A**daptive **E**ntity **L**inking with LLM-Driven **C**ontextualization (AELC). By leveraging high-density key information condensation prompt and tool invocation strategy, AELC extracts crucial information for target mentions. Moreover, the candidate filtering strategy combined with the candidate adaptive iterative strategy improves the quality of the candidate set. Furthermore, we reformulate the EL task as a multiple-choice problem, enhancing the adaptability and accuracy of LLMs in performing language tasks. Experimental results demonstrate that AELC achieves state-of-the-art performance on four benchmark datasets. In future work, we will investigate more effective and efficient ways to combine LLMs and Tools for entity linking, e.g.,auto prompt learning, and extend this framework to Multimodal Entity Linking task.

Limitations

Our work has two main limitations. First, we focus solely on entity linking within the text modality and do not consider other forms of modality-specific information. Second, the framework relies on large language model APIs, which incurs costs. Future research should explore incorporating a wider range of modality information and investigate cost-effective ways to leverage LLMs to achieve optimal performance.

Acknowledgments

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

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Berkeley R Andrus, Yeganeh Nasiri, Shilong Cui, Benjamin Cullen, and Nancy Fulda. 2022. Enhanced story comprehension for large language models through dynamic document-based knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10436–10444.

Akhil Arora, Alberto García-Durán, and Robert West. 2021. Low-rank subspaces for unsupervised entity linking. *arXiv preprint arXiv:2104.08737*.

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.

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

Young Min Cho, Li Zhang, and Chris Callison-Burch. 2022. Unsupervised entity linking with guided summarization and multiple-choice selection. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9394–9401.

Daniela D'Auria, Vincenzo Moscato, Marco Postiglione, Giuseppe Romito, and Giancarlo Sperlí. 2023a. Improving graph embeddings via entity linking: A case study on italian clinical notes. *Intelligent Systems with Applications*, 17:200161.

Daniela D'Auria, Vincenzo Moscato, Marco Postiglione, Giuseppe Romito, and Giancarlo Sperlí. 2023b. Improving graph embeddings via entity linking: A case study on italian clinical notes. *Intelligent Systems with Applications*, 17:200161.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2020. Autoregressive entity retrieval. *arXiv preprint arXiv:2010.00904*.

Nicola De Cao, Ledell Wu, Kashyap Popat, Mikel Artetxe, Naman Goyal, Mikhail Plekhanov, Luke Zettlemoyer, Nicola Cancedda, Sebastian Riedel, and Fabio Petroni. 2022. Multilingual autoregressive entity linking. *Transactions of the Association for Computational Linguistics*, 10:274–290.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Yifan Ding, Qingkai Zeng, and Tim Weninger. 2024. Chatel: Entity linking with chatbots. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 3086–3097. ELRA and ICCL.

Miao Fan, Qiang Zhou, and Thomas Fang Zheng. 2015. Distant supervision for entity linking. *arXiv preprint arXiv:1505.03823*.

Ivan P Fellegi and Alan B Sunter. 1969. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210.

Zhaochen Guo and Denilson Barbosa. 2018. Robust named entity disambiguation with random walks. *Semantic Web*, 9(4):459–479.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pages 782–792.

Phong Le and Ivan Titov. 2019. Distant learning for entity linking with automatic noise detection. *arXiv preprint arXiv:1905.07189*.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. *arXiv preprint arXiv:1906.07348*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Erik F Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*.

Senbao Shi, Zhenran Xu, Baotian Hu, and Min Zhang. 2024. Generative multimodal entity linking. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 7654–7665. ELRA and ICCL.

Fahim K Sufi. 2022. Identifying the drivers of negative news with sentiment, entity and regression analysis. *International Journal of Information Management Data Insights*, 2(1):100074.

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*, pages 628–644. Springer.

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:e1.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Yanfeng Wang, Tao Wang, Junhui Wang, Xin Zhou, Ming Gao, and Runmin Liu. 2022. Military chain: construction of domain knowledge graph of kill chain based on natural language model. *Mobile Information Systems*, 2022:1–11.

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2019. Scalable zero-shot entity linking with dense entity retrieval. *arXiv preprint arXiv:1911.03814*.

Amy Xin, Yunjia Qi, Zijun Yao, Fangwei Zhu, Kaisheng Zeng, Bin Xu, Lei Hou, and Juanzi Li. 2024. LLMAEL: large language models are good context augmenters for entity linking. *CoRR*, abs/2407.04020.

Kun Xu, Liwei Wang, Mo Yu, Yansong Feng, Yan Song, Zhiguo Wang, and Dong Yu. 2019. Cross-lingual knowledge graph alignment via graph matching neural network. *arXiv preprint arXiv:1905.11605*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, and 1 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Hsiu-Wei Yang, Yanyan Zou, Peng Shi, Wei Lu, Jimmy Lin, and Xu Sun. 2019. Aligning cross-lingual entities with multi-aspect information. *arXiv preprint arXiv:1910.06575*.

Chuanming Yu, Zhengang Zhang, Lu An, and Gang Li. 2023. A knowledge graph completion model integrating entity description and network structure. *Aslib Journal of Information Management*, 75(3):500–522.

Weixin Zeng, Xiang Zhao, Jiuyang Tang, and Xuemin Lin. 2020. Collective entity alignment via adaptive features. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 1870–1873. IEEE.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and 1 others. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

A Details of Datasets

- The **AIDA-CoNLL** dataset (Hoffart et al., 2011) stands as one of the earliest high-quality manually annotated datasets within the entity linking literature. It is founded on the CoNLL 2003 shared task (Sang and De Meulder, 2003). This dataset is segmented into training, validation, and testing partitions. Given the entirely unsupervised nature of our framework, we solely utilize the testing (CoNLL-Test) subset.
- The **WNED-Wiki** and **WNED-Cweb (WNED-Clueweb)** datasets are introduced benchmark datasets by (Guo and Barbosa, 2018), which were created by uniformly sampling mentions with different levels of prior scores from English Wikipedia ('2013-06-06' dump) and FACC1 [6] respectively.
- The **Wiki-Wiki** dataset (Cho et al., 2022) consists of mentions of common sense knowledge extracted from the WikiHow[7] and their corresponding entity to Wikidata.

B Details of Baselines

- τ**MIL-ND**: The τMIL-ND, designed by (Le and Titov, 2019), represents one of the earliest EL models that does not require annotated datasets. It transforms the EL task into a binary multi-instance learning (MIL) task by employing a noise-based detection classifier with remote supervision
- **Eigentheme**: The Eigentheme, designed by (Arora et al., 2021), stands out as the most mature solution in the realm of fully unsupervised EL tasks. By constructing the full embedding space for entities through graph embeddings, the model identifies the low-rank subspace using Singular Value Decomposition (SVD) and calculates candidate entities based on their proximity to the subspace.
- **SumMC**: The SumMC model, as designed by (Cho et al., 2022), is a pioneering model that was the first to employ a LLM (GPT-3) to achieve a fully unsupervised EL task. Currently, it represents the state-of-the-art among fully unsupervised EL models.

C The Quality of Candidates for Candidate Filtering.

In our research, $Prompt_{summary}$ denotes the initial version of the Key Information Condensation Prompt. LLMs execute this prompt to generate summary outputs, which are then used to perform a retrieval in Wikidata. When the results of the search show a significant deviation, $Prompt^*_{summary}$ performs a secondary retrieval by substituting the summary with mentions to improve accuracy. $Prompt_{example}$ is an improved version of $Prompt_{summary}$ that incorporates 32 additional examples to strengthen the LLM's in-context learning ability. Similarly, after generating output using this prompt, retrieval is performed on Wikidata, and if notable deviations occur, $Prompt^*_{example}$ employs entity mentions as substitutes for a second retrieval.

The experimental results in Table 3 show that both $Prompt^*_{summary}$ and $Prompt^*_{example}$ outperform their respective base versions, $Prompt^*_{summary}$ and $Prompt_{example}$. Consequently, we further analyze how these two prompt types affect candidate quality under different context lengths (64 tokens or 128 tokens) and various Candidate Filtering Strategies.

- **w.o. Can. Fil. :** Without using Candidate Filtering. Using the term 'summary' returned by $Prompt_{summary}$ as the query, call the dynamic online Wikidata to directly search for the query, and obtain the results as candidates.
- **BLEU:** Based on the outputs of $Prompt^*_{summary}$ and $Prompt^*_{example}$, the BLEU metric is used to separately calculate the similarity between the summary-description pairs, which serves as a criterion for candidate filtering. Here, description refers to the attribute information of search entries directly retrieved from mentions in Wikidata.
- **BERTScore:** Similar to the step of BLEU.

As shown in Figure 3, the quality of the candidates filtered by BLEU and BERT-Score is significantly higher than that of the unfiltered candidates (denoted as w.o. Can. Fil.). Specifically, the number of candidates that contain the correct answer (categories *easy* and *medium*) increases notably, while the number of candidates without the correct answer (*hard*) and those with no matching search results (*none*) decreases substantially.

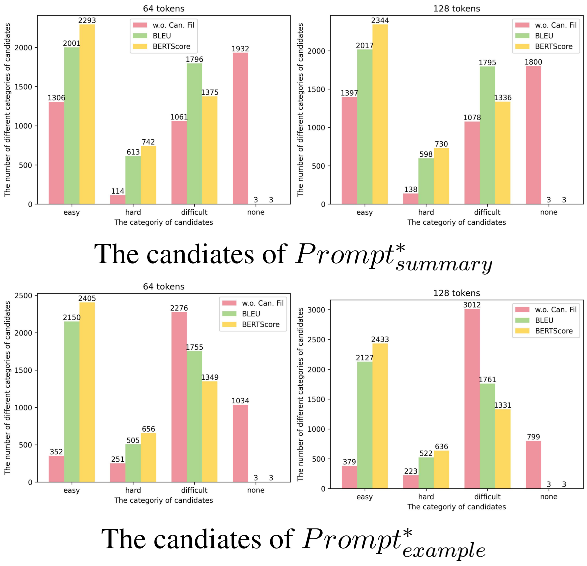
This improvement can be primarily attributed to the retrieval capabilities of Wikidata. As a structured knowledge graph, Wikidata excels at retrieving entities and their attribute information, with

[6] http://lemurproject.org/clueweb12/

[7] https://www.wikihow.com/Main-Page

Prompts	**Examples of semantic understanding prompt (sentence, description)**
$Prompt^{*}_{summary}$	’Soccer-late goals give JAPAN win over SYRIA...’, ’Japan national football team: national association football team.’
$Prompt^{*}_{example}$	’Soccer-late goals give JAPAN win over SYRIA...’, "Japan national football team: men’s national association football team representing Japan."

Table 6: **The different instances in Key Information Condensation Prompts.**



The candiates of $Prompt^{*}_{summary}$

The candiates of $Prompt^{*}_{example}$

Figure 3: **Quality Demonstration Graph of Candidates for Candidate Filtering.**

higher accuracy when the target mention closely matches the entity name. However, the complexity of Wikidata searches increases in practice. For example, when querying the keyword “*JAPAN*”, Wikidata returns the entity representing the country “*Japan (Q17)*”. In contrast, a query such as “*The target mention ‘JAPAN’ refers to a Japanese football team or national team that won a match against Syria*”, Wikidata returns ‘*There were no results matching the query*’. Therefore, directly using queries generated by Key Information Condensation Prompt makes it difficult to effectively perform both semantic disambiguation and entity linking in retrieval.

Comparing candidate quality under BLEU and BERT-Score metrics between $Prompt^{*}_{summary}$ and $Prompt^{*}_{example}$ reveals that the BERT-Score demonstrates a clear advantage. This is because BLEU is based solely on word overlap counts without considering semantic similarity, whereas LLM-generated summaries are expressed in natural language, and pre-trained BERT models can effectively capture deep semantic information, resulting in more accurate similarity evaluations.

Moreover, under the same evaluation conditions, the candidates generated by $Prompt^{*}_{example}$ exhibit higher quality than those generated by $Prompt^{*}_{summary}$. This difference comes primarily from variations in the in-context learning examples provided in each prompt, which lead to different interpretations of the mention content, thereby affecting the similarity scores computed by BLEU and BERT-Score and ultimately influencing candidate quality assessment.

Table 6 presents an example constituted by a **tuple (sentence, description)**, where the **sentence**, indicating the context of the mention, remains consistent between $Prompt^{*}_{summary}$ and $Prompt^{*}_{example}$. The distinction lies in that **description** in $Prompt^{*}_{summary}$ is derived from the explanation of the mention in the original dataset, while **description** in $Prompt^{*}_{example}$ is derived from Wikidata’s description of the mention.

Given that descriptions from Wikidata are more current and accurate, the original dataset’s explanations may contain errors due to their temporal limitations. Consequently, the candidate quality in $Prompt_{example}$ is superior to that in $Prompt_{summary}$, further underscoring the necessity of using the online dynamic Wikidata as a replacement for the original static dataset (knowledge graph).

D Different combinations of candidates $\mathcal{C}$.

Table 7, we present different combinations of Sea_can and Sim_can used to construct the candidate sets $\mathcal{C}$.

Combines	**64 tokens**		**128 tokens**	
	L_{sen}	L_{men}	L_{sen}	L_{men}
Sea_can	0.65	0.66	0.67	0.68
Sim_can	0.47	0.49	0.51	0.54
$Sim\_can + Sea\_can$	0.56	0.59	0.61	0.64
$Sea\_can + Sim\_can$	0.69	0.71	0.70	0.78

Table 7: **Different combinations of Sea_can and Sim_can to construct the $\mathcal{C}$ in AIDA-B dataset.**

- Sea_can refers to candidates constructed from the top-5 results obtained by directly searching the mention on Wikidata.

- Sim_can refers to candidates that are constructed from the top-5 results obtained through BERTScore (summary, description).

- Sea_can + Sim_can refers to the candidates from Sim_can are appended to those in Sea_can.

- Sim_can + Sea_can refers to the candidates from Sea_can are appended to those in Sim_can.

The precision of Sea_can surpasses that of Sim_can. This superiority is attributable to Wikidata's robust search and matching capabilities, which yield the higher quality of results for mention searches. Additionally, when employing the BERTScore model, the model exhibits varying preferences for different vocabularies. Relying solely on the obtained candidates can inadvertently lower the correct candidate's rank among the candidates. Based on these observations, we conducted experiments involving the combination of two subsets of candidates.

The results indicate that $Sea\_can + Sim\_can$ outperforms $Sim\_can + Sea\_can$, mainly due to the candidates generated by Wikidata tend to appear in the upper half of the candidate list. LLMs demonstrate a deeper memory of previously encountered information, which in turn enhances the quality of multiple-choice selection, thereby improving the overall accuracy of entity linking.

E Different question of multiple choice prompts.

In Table 8, $Sentence$ denotes the sentence containing the mention, while $Summary_{SumMC}$ and $Summary_{AELC}$ correspond to the summaries generated by the SumMC and AELC models, respectively.

The results indicate that when employing the three terms as the question of multiple-choice prompts for entity linking, the precision achieved by each prompt does not differ much. Notably, the precision of the $Summary_{AELC}$ is marginally higher, that attributable to the detailed steps incorporated within the key information condensation prompt. These steps effectively exploit the LLMs comprehension capability.

Questions	Can_5	Can_{10}	Can_{20}
$sentence$	0.711	0.774	0.752
$summary_{SumMC}$	0.713	0.774	0.760
$summary_{AELC}$	0.718	0.783	0.764

Table 8: **Effect of the different question of multiple choice prompt.** The can_N indicates the number of candidates is N.

Mentions	**AIDA**	**WNED-Wiki**	**WNED-Cweb**	**Wiki-Wiki**
-70%	0.18	0.14	0.18	0.27
-50%	0.35	0.23	0.26	0.41
-30%	0.45	0.30	0.34	0.52
-10%	0.56	0.35	0.36	0.64
all	0.64	0.47	0.48	0.76

Table 9: **The analysis of static offline KG.** The $-X\%$ indicates random removal of $X\%$ of the raw dataset.

The number of candidates affects multiple-choice results, with 10 candidates giving the best results. Having too few candidates (can_5) means the correct option might not be covered. In contrast, having too many (can_20) makes the text too long and complicates the selection process for LLMs due to their varied memory for differently sequenced information.

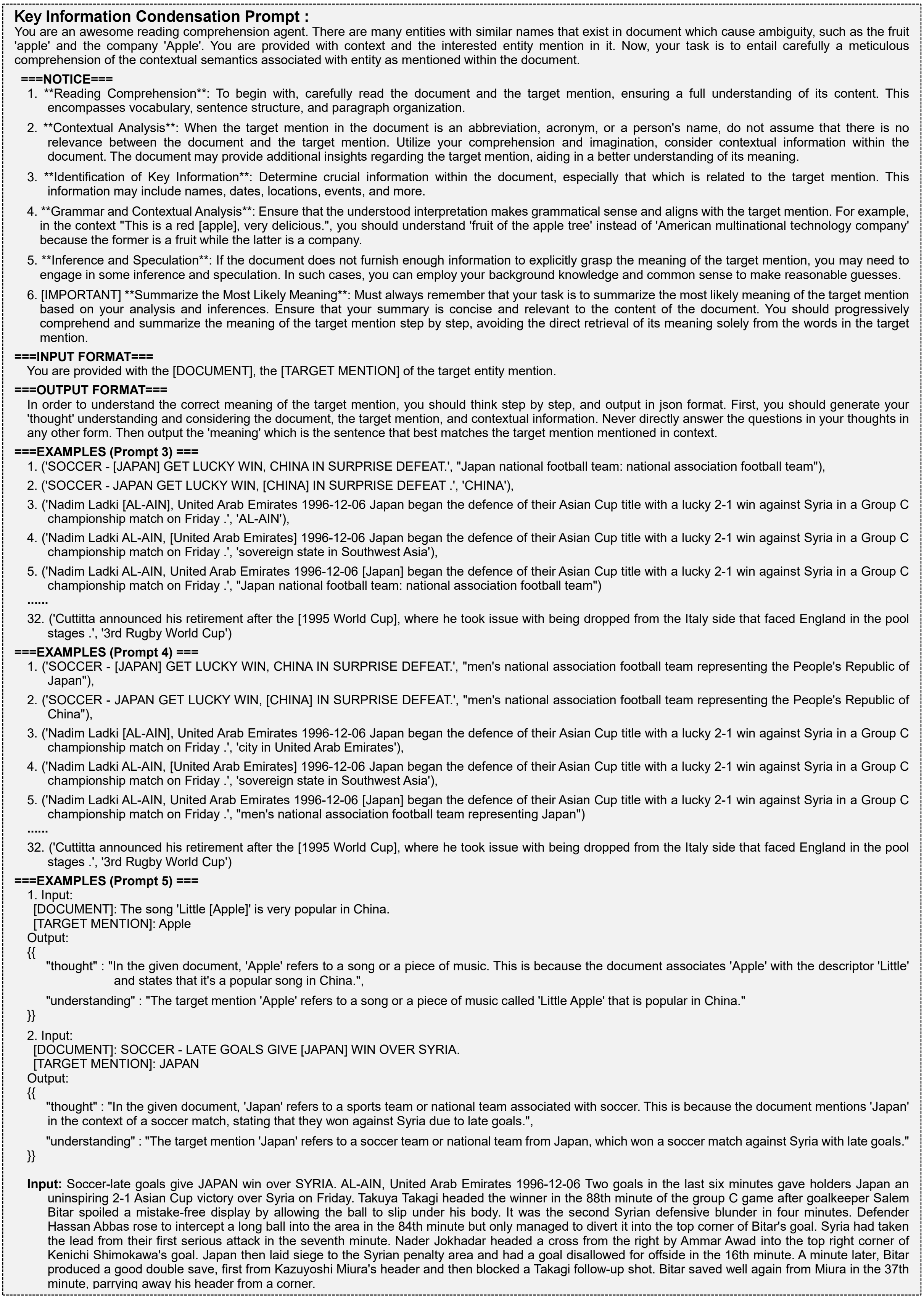
F Incompleteness of static knowledge graph.

In Table 9, we initiate an analysis of the impact of static offline knowledge graphs on the performance of the baseline SumMC. Specifically, we conducted experiments in which we randomly removed 10%, 30%, 50%, and 70% of mentions from the original dataset, simulating a scenario in which static offline knowledge graphs lack updates for mentions in the real world.

The experimental results reveal that when a substantial number of unknown target mentions are present, the effectiveness of SumMC experiences a significant decline. For example, in the case of the Wiki-Wiki dataset, the removal of 70% of target mentions results in a reduction of precision from 76% to 27%. Consequently, the invocation of tools, specifically the substitution of a static offline knowledge graph with a dynamic online KG, becomes highly necessary and meaningful.

G Details of Prompts

The details of the Key Information Condensation Prompt and the Multiple-choice Prompt are shown in Figures 4 and 5, respectively.



Key Information Condensation Prompt :

You are an awesome reading comprehension agent. There are many entities with similar names that exist in document which cause ambiguity, such as the fruit 'apple' and the company 'Apple'. You are provided with context and the interested entity mention in it. Now, your task is to entail carefully a meticulous comprehension of the contextual semantics associated with entity as mentioned within the document.

===NOTICE===

1. **Reading Comprehension**: To begin with, carefully read the document and the target mention, ensuring a full understanding of its content. This encompasses vocabulary, sentence structure, and paragraph organization.
2. **Contextual Analysis**: When the target mention in the document is an abbreviation, acronym, or a person's name, do not assume that there is no relevance between the document and the target mention. Utilize your comprehension and imagination, consider contextual information within the document. The document may provide additional insights regarding the target mention, aiding in a better understanding of its meaning.
3. **Identification of Key Information**: Determine crucial information within the document, especially that which is related to the target mention. This information may include names, dates, locations, events, and more.
4. **Grammar and Contextual Analysis**: Ensure that the understood interpretation makes grammatical sense and aligns with the target mention. For example, in the context "This is a red [apple], very delicious.", you should understand 'fruit of the apple tree' instead of 'American multinational technology company' because the former is a fruit while the latter is a company.
5. **Inference and Speculation**: If the document does not furnish enough information to explicitly grasp the meaning of the target mention, you may need to engage in some inference and speculation. In such cases, you can employ your background knowledge and common sense to make reasonable guesses.
6. [IMPORTANT] **Summarize the Most Likely Meaning**: Must always remember that your task is to summarize the most likely meaning of the target mention based on your analysis and inferences. Ensure that your summary is concise and relevant to the content of the document. You should progressively comprehend and summarize the meaning of the target mention step by step, avoiding the direct retrieval of its meaning solely from the words in the target mention.

===INPUT FORMAT===

You are provided with the [DOCUMENT], the [TARGET MENTION] of the target entity mention.

===OUTPUT FORMAT===

In order to understand the correct meaning of the target mention, you should think step by step, and output in json format. First, you should generate your 'thought' understanding and considering the document, the target mention, and contextual information. Never directly answer the questions in your thoughts in any other form. Then output the 'meaning' which is the sentence that best matches the target mention mentioned in context.

===EXAMPLES (Prompt 3) ===

1. ('SOCCER - [JAPAN] GET LUCKY WIN, CHINA IN SURPRISE DEFEAT.', "Japan national football team: national association football team"),
2. ('SOCCER - JAPAN GET LUCKY WIN, [CHINA] IN SURPRISE DEFEAT .', 'CHINA'),
3. ('Nadim Ladki [AL-AIN], United Arab Emirates 1996-12-06 Japan began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', 'AL-AIN'),
4. ('Nadim Ladki AL-AIN, [United Arab Emirates] 1996-12-06 Japan began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', 'sovereign state in Southwest Asia'),
5. ('Nadim Ladki AL-AIN, United Arab Emirates 1996-12-06 [Japan] began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', "Japan national football team: national association football team")

......

32. ('Cuttitta announced his retirement after the [1995 World Cup], where he took issue with being dropped from the Italy side that faced England in the pool stages .', '3rd Rugby World Cup')

===EXAMPLES (Prompt 4) ===

1. ('SOCCER - [JAPAN] GET LUCKY WIN, CHINA IN SURPRISE DEFEAT.', "men's national association football team representing the People's Republic of Japan"),
2. ('SOCCER - JAPAN GET LUCKY WIN, [CHINA] IN SURPRISE DEFEAT.', "men's national association football team representing the People's Republic of China"),
3. ('Nadim Ladki [AL-AIN], United Arab Emirates 1996-12-06 Japan began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', 'city in United Arab Emirates'),
4. ('Nadim Ladki AL-AIN, [United Arab Emirates] 1996-12-06 Japan began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', 'sovereign state in Southwest Asia'),
5. ('Nadim Ladki AL-AIN, United Arab Emirates 1996-12-06 [Japan] began the defence of their Asian Cup title with a lucky 2-1 win against Syria in a Group C championship match on Friday .', "men's national association football team representing Japan")

......

32. ('Cuttitta announced his retirement after the [1995 World Cup], where he took issue with being dropped from the Italy side that faced England in the pool stages .', '3rd Rugby World Cup')

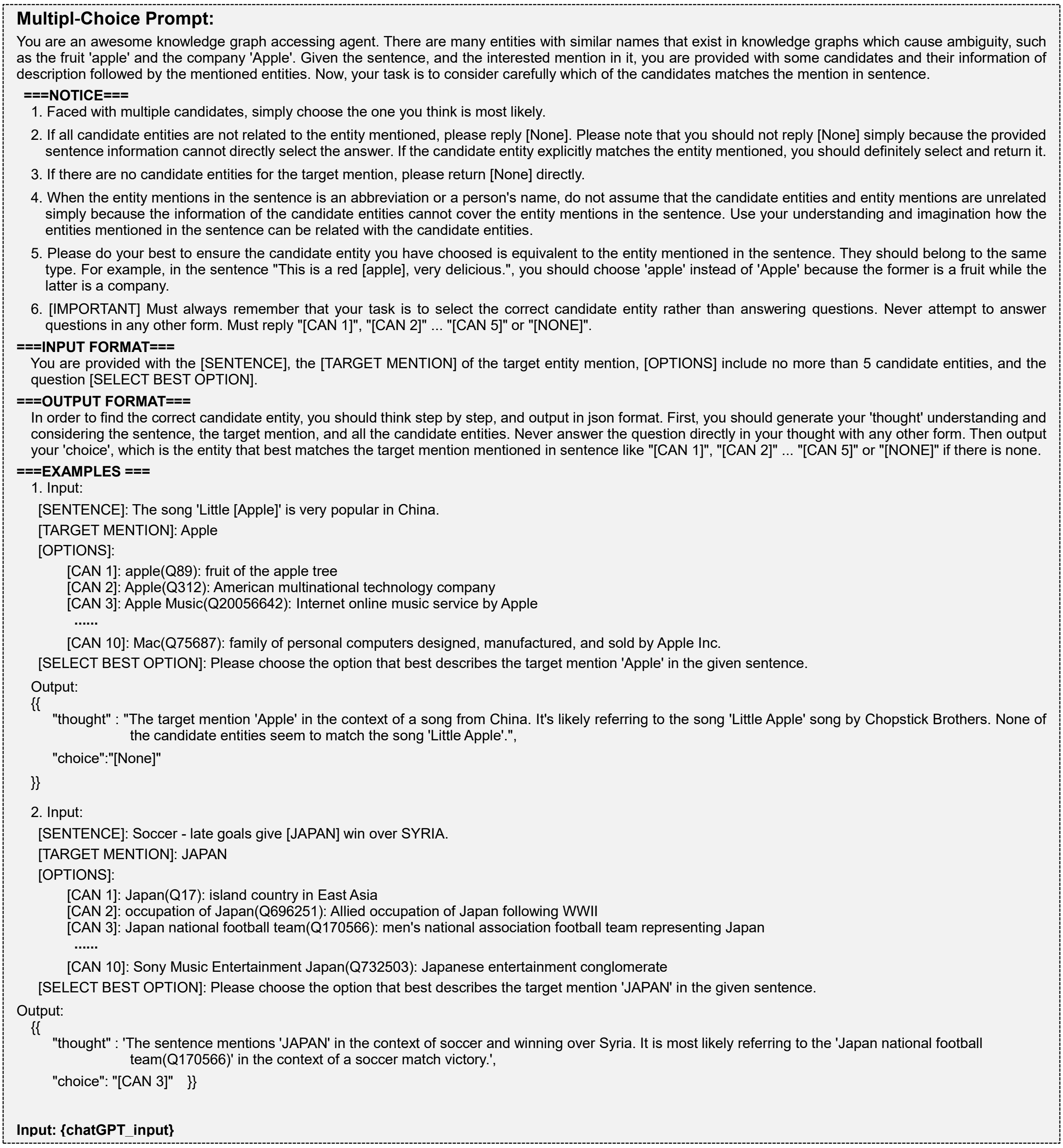
===EXAMPLES (Prompt 5) ===

1. Input:
[DOCUMENT]: The song 'Little [Apple]' is very popular in China.
[TARGET MENTION]: Apple
Output:
{{
"thought" : "In the given document, 'Apple' refers to a song or a piece of music. This is because the document associates 'Apple' with the descriptor 'Little' and states that it's a popular song in China.",
"understanding" : "The target mention 'Apple' refers to a song or a piece of music called 'Little Apple' that is popular in China."
}}

2. Input:
[DOCUMENT]: SOCCER - LATE GOALS GIVE [JAPAN] WIN OVER SYRIA.
[TARGET MENTION]: JAPAN
Output:
{{
"thought" : "In the given document, 'Japan' refers to a sports team or national team associated with soccer. This is because the document mentions 'Japan' in the context of a soccer match, stating that they won against Syria due to late goals.",
"understanding" : "The target mention 'Japan' refers to a soccer team or national team from Japan, which won a soccer match against Syria with late goals."
}}

Input: Soccer-late goals give JAPAN win over SYRIA. AL-AIN, United Arab Emirates 1996-12-06 Two goals in the last six minutes gave holders Japan an uninspiring 2-1 Asian Cup victory over Syria on Friday. Takuya Takagi headed the winner in the 88th minute of the group C game after goalkeeper Salem Bitar spoiled a mistake-free display by allowing the ball to slip under his body. It was the second Syrian defensive blunder in four minutes. Defender Hassan Abbas rose to intercept a long ball into the area in the 84th minute but only managed to divert it into the top corner of Bitar's goal. Syria had taken the lead from their first serious attack in the seventh minute. Nader Jokhadar headed a cross from the right by Ammar Awad into the top right corner of Kenichi Shimokawa's goal. Japan then laid siege to the Syrian penalty area and had a goal disallowed for offside in the 16th minute. A minute later, Bitar produced a good double save, first from Kazuyoshi Miura's header and then blocked a Takagi follow-up shot. Bitar saved well again from Miura in the 37th minute, parrying away his header from a corner.

Figure 4: **The Key Information Condensation Prompt.** The different examples are show in various prompts.



Multipl-Choice Prompt:

You are an awesome knowledge graph accessing agent. There are many entities with similar names that exist in knowledge graphs which cause ambiguity, such as the fruit 'apple' and the company 'Apple'. Given the sentence, and the interested mention in it, you are provided with some candidates and their information of description followed by the mentioned entities. Now, your task is to consider carefully which of the candidates matches the mention in sentence.

===NOTICE===

1. Faced with multiple candidates, simply choose the one you think is most likely.
2. If all candidate entities are not related to the entity mentioned, please reply [None]. Please note that you should not reply [None] simply because the provided sentence information cannot directly select the answer. If the candidate entity explicitly matches the entity mentioned, you should definitely select and return it.
3. If there are no candidate entities for the target mention, please return [None] directly.
4. When the entity mentions in the sentence is an abbreviation or a person's name, do not assume that the candidate entities and entity mentions are unrelated simply because the information of the candidate entities cannot cover the entity mentions in the sentence. Use your understanding and imagination how the entities mentioned in the sentence can be related with the candidate entities.
5. Please do your best to ensure the candidate entity you have choosed is equivalent to the entity mentioned in the sentence. They should belong to the same type. For example, in the sentence "This is a red [apple], very delicious.", you should choose 'apple' instead of 'Apple' because the former is a fruit while the latter is a company.
6. [IMPORTANT] Must always remember that your task is to select the correct candidate entity rather than answering questions. Never attempt to answer questions in any other form. Must reply "[CAN 1]", "[CAN 2]" ... "[CAN 5]" or "[NONE]".

===INPUT FORMAT===

You are provided with the [SENTENCE], the [TARGET MENTION] of the target entity mention, [OPTIONS] include no more than 5 candidate entities, and the question [SELECT BEST OPTION].

===OUTPUT FORMAT===

In order to find the correct candidate entity, you should think step by step, and output in json format. First, you should generate your 'thought' understanding and considering the sentence, the target mention, and all the candidate entities. Never answer the question directly in your thought with any other form. Then output your 'choice', which is the entity that best matches the target mention mentioned in sentence like "[CAN 1]", "[CAN 2]" ... "[CAN 5]" or "[NONE]" if there is none.

===EXAMPLES ===

1. Input:

[SENTENCE]: The song 'Little [Apple]' is very popular in China.

[TARGET MENTION]: Apple

[OPTIONS]:

[CAN 1]: apple(Q89): fruit of the apple tree
[CAN 2]: Apple(Q312): American multinational technology company
[CAN 3]: Apple Music(Q20056642): Internet online music service by Apple
......
[CAN 10]: Mac(Q75687): family of personal computers designed, manufactured, and sold by Apple Inc.

[SELECT BEST OPTION]: Please choose the option that best describes the target mention 'Apple' in the given sentence.

Output:
{{
"thought" : "The target mention 'Apple' in the context of a song from China. It's likely referring to the song 'Little Apple' song by Chopstick Brothers. None of the candidate entities seem to match the song 'Little Apple'.",
"choice":"[None]"
}}

2. Input:

[SENTENCE]: Soccer - late goals give [JAPAN] win over SYRIA.

[TARGET MENTION]: JAPAN

[OPTIONS]:

[CAN 1]: Japan(Q17): island country in East Asia
[CAN 2]: occupation of Japan(Q696251): Allied occupation of Japan following WWII
[CAN 3]: Japan national football team(Q170566): men's national association football team representing Japan
......
[CAN 10]: Sony Music Entertainment Japan(Q732503): Japanese entertainment conglomerate

[SELECT BEST OPTION]: Please choose the option that best describes the target mention 'JAPAN' in the given sentence.

Output:
{{
"thought" : 'The sentence mentions 'JAPAN' in the context of soccer and winning over Syria. It is most likely referring to the 'Japan national football team(Q170566)' in the context of a soccer match victory.',
"choice": "[CAN 3]" }}

Input: {chatGPT_input}

Figure 5: **The Multiple-Choice Prompt.**