

Article

Enhanced Knowledge Graph Completion Based on Structure-Aware and Semantic Fusion Driven by Large Language Models

Jing Hu ^{1,2,*}, Hishammuddin Asmuni ¹, Kun Wang ² and Yingying Li ^{1,2}

¹ Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru 81310, Malaysia; hishamudin@utm.my (H.A.); liyingying@graduate.utm.my (Y.L.)

² School of Software, Shanxi Agricultural University, Jinzhong 030801, China; wangkun@sxau.edu.cn

* Correspondence: hujing@graduate.utm.my

Abstract

Knowledge graphs (KGs) have emerged as fundamental infrastructures for organizing structured information across a wide range of AI applications. Practically, KGs are often incomplete, which limits their effectiveness. Knowledge Graph Completion (KGC) has become a critical research problem. Existing methods of KGC primarily rely on graph structure or textual descriptions independently, often failing to capture the complex interplay between structural topology and rich semantic context. Recent advances in Large Language Models (LLMs) offer promising capabilities in understanding and generating human-like semantic representations. However, effectively integrating such models with structured graph information remains a challenging and underexplored area. In this work, we propose an enhanced KGC framework that leverages a structure-aware and semantic fusion mechanisms driven by the representational power of LLMs. Our method jointly encodes the topological structure of the graph and the textual semantics of entities and relations, allowing for more informed and context-rich KGC. The experimental results of benchmark datasets demonstrate that our approach outperforms existing baselines, particularly in scenarios with sparse graph connectivity or limited textual information. In particular, on the WN18RR dataset, the model demonstrates a 12.4% increase in Hits@3 and an 11.7% increase in Hits@10.

Keywords: knowledge graphs; knowledge graph completion; structure-aware; semantic fusion; large language models



Academic Editor: Arkaitz Zubiaga

Received: 24 October 2025

Revised: 17 November 2025

Accepted: 18 November 2025

Published: 19 November 2025

Citation: Hu, J.; Asmuni, H.; Wang, K.; Li, Y. Enhanced Knowledge Graph Completion Based on Structure-Aware and Semantic Fusion Driven by Large Language Models. *Electronics* **2025**, *14*, 4521. <https://doi.org/10.3390/electronics14224521>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Knowledge graphs (KGs) have been established as a powerful semantic network model grounded in graph theory, designed to represent structured and semi-structured information by capturing entities and their interrelations. Their primary objective is to integrate and unify heterogeneous data sources—ranging from structured databases to unstructured text—to improve a machine's ability to understand, interpret, and reason about complex information. As a result, KGs have been increasingly adopted across a wide range of domains, including, but not limited to, question answering, information retrieval, natural language processing, and personalized recommendation systems [1]. Despite their growing importance, most existing KGs suffer from inherent limitations regarding data completeness. One of the major challenges lies in the restricted scale and scope of the source data available for KG construction, particularly in capturing implicit or commonsense

knowledge that is not explicitly stated in textual corpora or structured databases. Such knowledge, while crucial for human reasoning, is often absent or underrepresented in current KGs. Empirical studies have revealed that many large-scale KGs still exhibit significant sparsity, with a considerable portion of entities and their relations either missing or incompletely defined [2]. To address the issue, Knowledge Graph Completion (KGC) has emerged. The task of KGC is to enhance the structure and improve the quality of KGs by predicting missing entities or relations and the rationality of given triples, thereby uncovering unknown and uncertain facts [3].

Traditional KGC methods [4–6] typically rely on the structural characteristics of a graph and manually defined logical inference rules to predict missing entities or relations. While these approaches have demonstrated effectiveness on relatively small and dense KGs, their limitations become increasingly evident as the scale and complexity of KGs continue to grow. Firstly, the rapid expansion of KGs aggravates the problem of data sparsity, making it increasingly difficult to mine high-quality inference paths or patterns. Secondly, although KGs function as semantic networks where entities and relations carry rich contextual meaning, many traditional KGC models [7] are inherently limited in their ability to model these complex semantic associations. They often depend on shallow structural features, and thus fall short in understanding deeper linguistic or contextual nuances within the data. Moreover, traditional methods generally lack the capacity to incorporate external or contextual information, such as background knowledge from text corpora or user-generated content. To address these limitations, advances have explored KGC methods based on pre-trained language models (PLMs), such as BERT [8] and SimKGC [9]. These models are trained on large-scale text corpora and are capable of capturing nuanced semantic patterns and contextual dependencies in language. When applied to KGC, PLMs enable more robust reasoning by leveraging both the structural information of the KG and external textual evidence. For example, PLM-based models can encode entity and relation descriptions, model implicit commonsense knowledge, and support zero-shot or few-shot inference scenarios. However, PLM-based methods also face challenges, including computational cost, explainability, and the need for careful alignment between textual and structured representations.

Recently, Large Language Models (LLMs) have become a promising solution. By utilizing its extensive pre-training on different corpora and excellent semantic understanding, LLMs offer a promising pathway to enhance the completeness, accuracy, and contextual richness of KGs [10]. Some initial studies [11,12] have explored LLM-based approaches to KGC. While these methods mark a promising direction, they typically reformulate KGC as a text generation or classification problem over individual triples. Most prominently, LLMs frequently fall short in possessing accurate and verifiable factual information, giving rise to the widely recognized problem of “hallucination” [13]. In addition, these approaches tend to neglect the inherently structured nature of knowledge graphs, such as subgraph topology, relational dependencies, and contextual entity relation dynamics. Such structural information, although non-textual, is semantically rich and, if effectively integrated, could significantly improve LLMs’ comprehension and reasoning over KGs.

The interact of structural information and semantic information is a mechanism of information supplement and information error correction. In KGs, structural information is often sparse and incomplete. Semantic information can be a powerful supplement. At the same time, natural language also has polysemy: the same word has different meanings in different contexts. In this way, structural information can provide critical context for information error correction. In order to tackle the constraints in current KGC approaches in semantic generalization and structural utilization, we propose a structure-aware and semantic fusion KGC framework. This framework fully leverages the powerful semantic

understanding and content-generation capabilities of LLMs, aiming to generate unstructured textual representations of subgraphs related to the target triple and high-quality candidate entities or relations for prediction. By combining the structured information and textual descriptions present in the KG with the rich semantic knowledge encoded in the LLM-generated text, our approach maintains the structural integrity of the graph while significantly enhancing the model's ability to understand and reason in complex semantic contexts. Specifically, the framework first performs subgraph extraction centered on the incomplete triple, using this localized structure as the core unit for semantic modeling. This subgraph localization strategy not only reduces the candidate search space and strengthens the semantic alignment between the generated text and the KG structure, but also improves the efficiency of semantic understanding and content generation by the LLM. Subsequently, the incomplete triple is used as input to the LLM to generate candidate entities or relations enriched with external knowledge. Under structural constraints, the LLM is guided to produce natural language descriptions of the subgraph, resulting in more context-aware and semantically coherent outputs, thereby reducing the likelihood of hallucinations during generation. Finally, by integrating the generated subgraph descriptions with existing textual information in the KG, the model assesses the plausibility of each candidate entity or relation and completes the triple accordingly. Meanwhile, the integration of structured relational patterns, entity neighborhood information, and unstructured semantic content empowers the LLM with enhanced interpretability and accuracy during the knowledge completion process. In summary, this paper presents a structure-aware and semantic fusion KGC framework which integrates structural information with the natural language processing capability of LLMs. The aims of this paper are as follows:

- To embed and integrate the structural knowledge information of KGs with the knowledge generated by LLMs, we propose a KGC framework based on structure-aware and semantic fusion driven by LLMs, which can enhance the interpretability and accuracy of inferences.
- To fully utilize the complex structural and textual information in KGs, we propose a subgraph localization and knowledge acquisition strategy that can reduce the search space of KGs and the occurrence of “hallucinations” of LLMs, improving the efficiency of semantic understanding and content generation of LLMs.
- To enrich additional semantic knowledge of the incomplete triple, a candidate set generation and optimization method driven by LLMs is proposed that can obtain high-quality and broader entities or relations.

The rest of this paper is organized as follows: Section 2 introduces the related work. Section 3 describes an enhanced KGC framework driven by LLMs. Section 4 presents the experiment settings and results. Section 5 provides a summary of the research and examines potential avenues for future work.

2. Related Work

2.1. Knowledge Graph Completion

KGC aims to predict the missing entities or relations in a KG, thereby enhancing its coverage and utility. Conventional KGC methods can be generally divided into two main types: embedding-based and rule-based methods. Embedding-based models, exemplified by TransE [14], DistMult [15], and ComplEx [16], encode entities and relations into continuous vector representations and deduce missing connections via algebraic computations. These methods are computationally efficient, but often struggle with capturing complex relational patterns and semantic nuances. Rule-based methods, like KGRL [5] and MF-PGM [6], exploit logical rules and graph patterns to infer new facts. While interpretable, they typically require extensive manual feature engineering and may not generalize well

to unseen data. The KGC methods that employ neural network models harnesses the powerful learning and expressive capacities of neural networks to construct a model of the KG, thereby improving its reasoning capabilities. The ConvE model [17] was introduced for tasks related to link prediction. While shallow models are commonly employed for link prediction in large KGs, they often struggle to capture the deeper underlying features, leading to subpar prediction performance. Shang et al. [18] merged the features of GCN and the ConvE model, introducing an end-to-end structure-aware convolutional network to complete the KG. The utilization of GCN in SACN proves to be a powerful technique for generating embedded representations of nodes. It has the capability to aggregate local information from neighboring nodes within the graph for each individual node. These studies do not consider the influence of neighborhoods' information. Recent advances in PLMs, such as BERT [19], has opened new possibilities for KGC. Unlike traditional methods that rely solely on the graph's structure, approaches grounded in PLMs capitalize on the extensive semantic comprehension embedded within Large Language Models to deduce absent knowledge. These methods typically convert triples into natural language sentences and treat KGC as a masked token prediction, classification, or question answering task. KG-BERT [8,20] demonstrated that PLMs can be fine-tuned on triple classification tasks by encoding entities and relations in textual form. StAR [21] integrates structural constraints into PLMs using entity markers, knowledge-aware attention, or auxiliary graph encoders to address the semantic-structure gap. However, despite their strong generalization capabilities, PLMs still face challenges in incorporating large-scale graph structures, integrating local graph contextual information, and maintaining efficiency in real-world applications.

2.2. LLMs-Based KGC

With the emergence of LLMs such as GPT-3/4, DeepSeek, and LLaMA, there has been a surge of interest in using these models for KGC. Compared to earlier PLM-based methods, LLMs offer stronger generalization through few-shot and zero-shot learning by leveraging prompt engineering and in-context learning, significantly reducing the need for task-specific fine-tuning. One of the pioneering efforts in this direction is the LAMA framework [22], which probes PLMs and LLMs for factual knowledge using cloze-style prompts. To address the limited use of KG structure in LLMs, SIKGC [23] arranged the triples in the KG as the sequences of text. By fusing the descriptions of entities, relations and their structural information as task-aware prompts, they input such prompts into Large Language Models and regard the responses as prediction tasks. Zhang et al. [12] proposed a new method for integrating the structural information of KGs into LLMs to improve the performance of triple classification. Guan et al. [24] proposed a KG-based adjustment framework that effectively reduces the factual illusion in the LLM inference process and improves the reliability of the model by autonomously verifying and adjusting the responses generated by the LLM. Yao et al. [25] introduced an innovative framework called KG-LLM to model these triples. It employs entity and relation descriptions of a triple as prompts and utilizes the response for predictions. These methods transform the KGC task into natural language prompts such as "Where was Barack Obama born?" and evaluate the LLM's ability to complete missing elements. Ji et al. [10] introduced a framework for KGC, termed knowledge-guided LLM reasoning (KLR-KGC), which directs the LLM to sift through and prioritize candidate entities, thereby constraining its output to minimize omissions and erroneous answers. However, this framework is based on traditional knowledge graph embedding (KGE) methods and does not take into account the unstructured information in the KG. The GS-KGC [26] framework utilizes negatives and neighboring information to motivate LLMs to produce a greater number of potential

answers. However, this method over-fit the results generated by the LLMs, and does not make full use of the powerful generation ability of LLMs.

To surmount the inherent constraints of KGC methods in terms of semantic generalization and structural utilization, we propose a structure-aware and semantic fusion KGC-enhanced framework. This framework harnesses the powerful semantic understanding and generative capabilities of LLMs to produce unstructured textual representations of subgraphs centered around the target triple along with high-quality candidate entities or relations for prediction. By effectively integrating the structured information and textual descriptions from the KG with the rich semantic knowledge captured in LLM-generated text, our approach preserves the structural coherence of the graph while significantly improving the model's capacity for semantic reasoning and complex relational inference.

3. Methodology

The framework integrates structure-aware reasoning with semantic fusion to aim to enhance the interpretability and accuracy of KGC by embedding explicit structured and unstructured information from KGs and aligning it with the rich contextual knowledge generated by LLMs. Specifically, we design a subgraph localization and knowledge acquisition strategy to extract relevant structural and textual contexts around the target triple. This strategy effectively reduces the KG search space while preserving critical relational information, thereby improving the semantic understanding and content generation efficiency of LLMs. To further enrich the semantic context of incomplete triples, we introduce an LLM-guided candidate set generation and optimization method that leverages keyword-based inference to produce broader and higher-quality entities or relations. By combining structured and unstructured subgraph information with unstructured LLM-generated semantics, our framework maintains graph coherence and significantly improves the model's capacity for complex relational inference.

3.1. Problem Definition

A KG can be represented as $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{D})$, where $\mathcal{E}$ and $\mathcal{R}$ denote the sets of entities and relations, respectively. The set $\mathcal{T} = \{(h, r, t) \mid h, t \in \mathcal{E}, r \in \mathcal{R}\}$ comprises the triples within the KG. $\mathcal{D}$ represents the set of descriptions for both entities and relations within the KG. The textual description of the entity and relation in the subgraph is represented as $SKG - text = \mathcal{D}(e) \oplus \mathcal{D}(r) \oplus \mathcal{D}(er)$, where $\mathcal{D}(e)$ is the textual description of each entity, $e \in \mathcal{E}$, $\mathcal{D}(r)$ is the textual description of each relation, and $r \in \mathcal{R}$, $\mathcal{D}(er)$ is the co-occurrence textual description of e and r . The KGC tasks involve discovering missing entities or relations and predicting the correctness of the triple, which is denoted as $(h, r, ?)$, $(h, ?, t)$ and $(h, r, t)?$. When applying LLMs to KGC tasks, we denote an LLM as M , which serves as a decoder. A candidate set generated by the LLM is a set of predicted candidate entities or candidate relations, which is denoted as $S_{M|\mathcal{E}|N} = \{e_1, e_2, e_3, \dots, e_n\}$ and $S_{M|\mathcal{R}|N} = \{r_1, r_2, r_3, \dots, r_n\}$. N is the number of candidate items. The LLM identifies all of the entities and relations of the subgraph and generates a relevant textual description, which is represented as $MSKG - text$. The LLM generates an optimal set $OS_{M|\mathcal{E}|K} = \{(e_i, s_i, d_i) \mid e \in \mathcal{E}, s \in \mathcal{S}, d \in \mathcal{DD}\}$ or $OS_{M|\mathcal{R}|K} = \{(r_i, s_i, d_i) \mid r \in \mathcal{R}, s \in \mathcal{S}, d \in \mathcal{DD}\}$ of entities or relations by analyzing the possibility for each candidate item and sorting them in descending order. Here, $\mathcal{S}$ denotes the collection of scores for the optimal items, $\mathcal{DD}$ represents the set of explanations corresponding to these optimal items, and K is the number of optimized and selected items.

3.2. Framework

An enhanced KGC (E-KGC) framework based on structure-aware and semantic fusion driven by LLMs is illustrated in Figure 1. In this framework, the key modules contain the subgraph localization and knowledge acquisition strategy and the candidate set generation and optimization method. In the subgraph localization and knowledge acquisition strategy, the KGC task and the incomplete KG are inputs. The intermediate result is the *Sub-KG* about the KGC task and the output is *SKG-text*. In the candidate set generation and optimization method, LLMs generate and optimize the candidate set of predicted candidate entities or relations based on the KGC task, *SKG-text*, and *MSKG-text*. The final result of the method is the optimal set of predicted entities or relations. It is worth noting that using this framework can accomplish the three KGC tasks. The overall process is presented in Algorithm 1. In this algorithm, the LLM is called four times, with time complexities including candidate set generation $O(1)$, subgraph triple encoding $O(N_{triple})$, subgraph text encoding $O(1)$, and candidate set optimization $O(N)$. The total time complexity of the algorithm is $O(N_{triple} + N)$, where N_{triple} is the number of the triples of the subgraph and N is the number of candidate items.

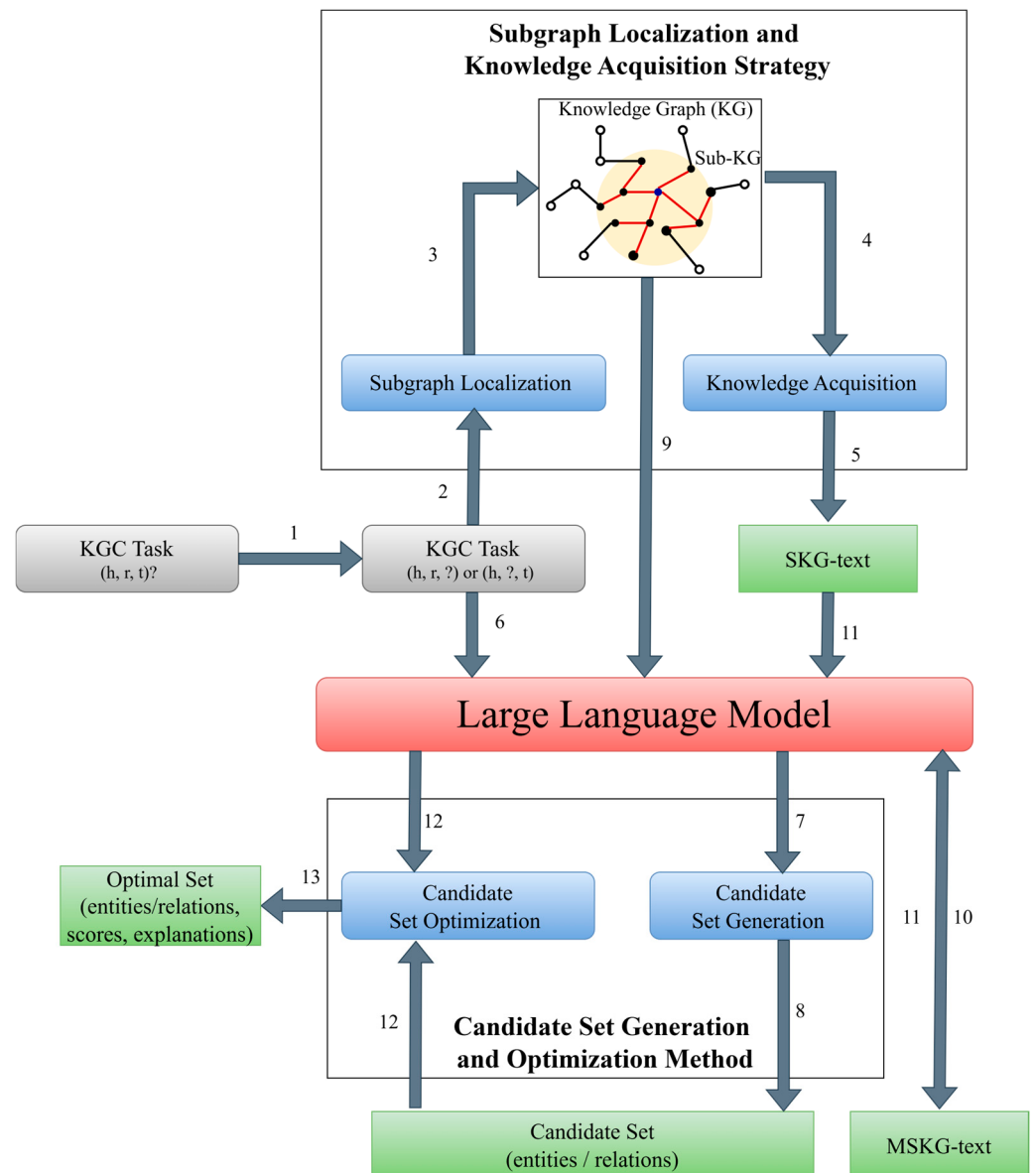


Figure 1. Overview of E-KGC framework process.

Algorithm 1 The E-KGC framework process**Input:** KGC task $(h, r, t)?$ or $(h, r, ?)$ or $(h, ?, t)$; KG $\mathcal{G}$; LLM M **Output:** Optimal set $OS_{M|\mathcal{E}|K}$ or $OS_{M|\mathcal{R}|K}$

```

1: if KGC task is  $(h, r, ?)$  or  $(h, ?, t)$  then
2:   Construct Sub-KG from  $\mathcal{G}$  according to target entity
3:   Obtain SKG-text from Sub-KG
4:   Get  $S_{M|\mathcal{E}|N}$  or  $S_{M|\mathcal{R}|N}$  according to  $h$  and  $r$  or  $h$  and  $t$  using  $M$ 
5:   Generate MSKG-text according to all entities and relations of Sub-KG using  $M$ 
6:    $OS_{M|\mathcal{E}|K} \leftarrow S_{M|\mathcal{E}|N}$  or  $OS_{M|\mathcal{R}|K} \leftarrow S_{M|\mathcal{R}|N}$  according to SKG-text, MSKG-text using  $M$ 
7: if KGC task is  $(h, r, t)?$  then
8:   Convert  $(h, r, t)?$  to  $(h, r, ?)$ 
9:   repeat line 2–line 6
10:  if  $t$  in  $\mathcal{E}$  of  $OS_{M|\mathcal{E}|K}$  then
11:     $(h, r, t)? \leftarrow \text{"Yes"}$ 
12:  else
13:     $(h, r, t)? \leftarrow \text{"No"}$ 

```

The framework enhances KGC interpretability and accuracy by integrating structure-aware reasoning and semantic fusion, embedding structured and unstructured KG information and aligning it with LLM-generated knowledge. It employs a subgraph localization and knowledge acquisition strategy to reduce KG search space and improve LLM semantic understanding and an LLM-guided method to enrich incomplete triples with higher-quality entities and relations.

3.3. Subgraph Localization and Knowledge Acquisition Strategy

Due to the extremely large scale of KGs, acquiring all neighboring entities and relations for a target entity directly from the entire KG imposes prohibitive computational overhead on LLMs. Simultaneously, this approach introduces excessive noise during semantic understanding, thereby compromising the accuracy of the generated content. To mitigate these challenges of resource consumption and LLM “hallucinations”, we propose a strategy of subgraph localization and knowledge acquisition.

Algorithm 2 implements breadth-first subgraph localization centered on target entity and textual knowledge acquisition. Firstly, the target entity e is added to the visited entity set $E_{visited}$, *Sub-KG* entity set E_{sub} , and queue Q , respectively, and the current hop distance is set to 0 (line 1–3). When the queue is not empty, an entity is taken from the queue (line 4–5). Secondly, we iteratively expand its neighborhood using bidirectional traversal of outgoing (line 7–13) and incoming (line 14–20) relations. The expansion terminates when reaching the predefined hop limit max_hops (line 6). The setting of max_hops depends on the richness of descriptions of entities and relations in the KG. The richer the description of entities and relations, the smaller the value of max_hops , and the higher the computational efficiency. For each neighbor entity (the tail entity in the outgoing side, the head entity in the incoming side), if the neighbor has not been visited, it will be marked as visited, added to the queue, and *Sub-KG* entity set E_{sub} , while the current distance increase by 1. we construct the vertex-induced subgraph by including all triples between entities in E_{sub} (line 23–30). Finally, triple-text acquisition (line 31–36) and entity attribute text acquisition from *Sub-KG* are showed. In order to enhance the semantic understanding of LLMs while balancing the simplicity and richness of knowledge information, we use an

extended mode as the entity attribute text generation method to extract and combine entity attribute information from *Sub-KG* (line 37–41). The extended mode is denoted as

$$\mathcal{D}''(e'') = \text{Connection}(e'', \mathcal{A}) \oplus d, \quad (1)$$

where $\mathcal{A}$ is the attribute set of e'' , d is the textual description of e'' and the function *Connection*() is denoted as

$$\text{Connection}(e'', \mathcal{A}) = \bigsqcup_{a \in \mathcal{A}} e''.\text{attr}(a), \quad (2)$$

For example, (Einstein, Albert (German physicist who proposed the theory of relativity)) is $\mathcal{D}''(e'')$. The value of *Connection*() is “Einstein, Albert”. d is “German physicist who proposed the theory of relativity”. $\mathcal{A}$ is “name, alias”.

3.4. LLM-Guided Candidate Set Generation and Optimization Method

To enrich our additional semantic knowledge of the incomplete triple, we proposed an LLM-guided candidate set generation and optimization method that can obtain high-quality and broader entities or relations. This method mainly consists of three steps: (1) LLMs generate the candidate set based on target entities and relations; (2) LLMs generate the text knowledge *MSKG-text* based on *Sub-KG*; and (3) LLMs optimize the candidate set based on the *MSKG-text* and *SKG-text*. The particulars of the method are elaborated upon in Algorithm 3. LLM-guided candidate set generation and optimization method is based on prompt engineering. The first phase is that LLMs generate the candidate set based on *prompt_template_1*, which includes the target entity and relation (line 1–2). The second phase is that the LLMs generate coherent natural language text for each triple and connect and optimize each sentence text generated into a qualified paragraph text (line 3–8). In the last phase, the LLMs analyze each item in the candidate set, sort them in descending order according to the prediction probability, and finally take the top K items as the items in the optimal set (line 9–13).

The example below shows the LLM-guided candidate set generation and optimization method for better interpretability. As shown in Figure 2, this example focuses on the LLM and introduces two types of text examples, *SKG-text* and *MSKG-text*, to improve the model inference. *SKG-text* focuses on the semantic analysis of specific emotional behaviors in the subgraph (such as various expressions of “understanding and entering into another’s feelings”), emphasizing the multidimensional definition of “empathy”; *MSKG-text*, through “hypernym” extension, reveals the broad connotation of the behavior of “understanding” (such as unexplained “empathy”). The two together construct a semantic gradient from concrete instances to abstract concepts. The model output is divided into candidate set and optimal set. The candidate set obtains a wider range of potential related entities (such as “emotion”, “sentiment”, etc.) based on extral knowledge of the LLM, providing a candidate pool for subsequent optimization. The optimal set selects high confidence entities (such as “sympathy”, “trait”, “emotion”) through probability thresholds. Each recommendation includes reason and probability to ensure the interpretability of the results. At the same time, based on the structural information and textual information of the subgraph, it avoids the “hallucination” of entities such as “words” and “language” in the candidate set.

Algorithm 2 Subgraph localization and knowledge acquisition**Input:** Target entity e ; KG $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{D})$; Maximum hop distance max_hops **Output:** Text of the entity and relation in the subgraph SKG_text

```

1:  $E\_visited \leftarrow E\_visited \cup \{e\}$ 
2:  $E\_sub \leftarrow E\_sub \cup \{e\}$ 
3:  $Q.enqueue((e, 0))$  // Add target entity and current hop distance into queue  $Q$ 
4: while  $Q$  is not empty do
5:    $(e\_curr, h\_curr) \leftarrow Q.dequeue$  // Get an entity and hop distance from the queue
6:   if  $h\_curr < max\_hops$  then
7:     for each triple  $(e\_curr, r, e\_tail) \in \mathcal{T}$  do // Outgoing relations
8:       if  $e\_tail \notin E\_visited$  then
9:          $E\_visited \leftarrow E\_visited \cup \{e\_tail\}$ 
10:         $E\_sub \leftarrow E\_sub \cup \{e\_tail\}$ 
11:         $Q.enqueue((e\_tail, h\_curr + 1))$ 
12:       end if
13:     end for
14:     for each triple  $(e\_head, r, e\_curr) \in \mathcal{T}$  do // Incoming relations
15:       if  $e\_head \notin E\_visited$  then
16:          $E\_visited \leftarrow E\_visited \cup \{e\_head\}$ 
17:          $E\_sub \leftarrow E\_sub \cup \{e\_head\}$ 
18:          $Q.enqueue((e\_head, h\_curr + 1))$ 
19:       end if
20:     end for
21:   end if
22: end while
23: for each entity  $e' \in E\_sub$  do
24:   for each triple  $(e_h, r, e) \in \mathcal{T}$  where  $e_h = e'$  or  $e = e'$  do
25:     if  $e_h \in E\_sub$  and  $e \in E\_sub$  then
26:        $T\_sub \leftarrow T\_sub \cup \{(e_h, r, e)\}$ 
27:     end if
28:   end for
29: end for
30:  $Sub\text{-}KG = (E\_sub, T\_sub)$ 
31: for each triple  $(h', r', t') \in T\_sub$  do
32:    $h'\_text \leftarrow D(h')$ 
33:    $r'\_text \leftarrow D(r')$ 
34:    $t'\_text \leftarrow D(t')$ 
35:    $D' \leftarrow D' \cup \{(h'\_text, r'\_text, t'\_text, h'\_text \oplus r'\_text \oplus t'\_text)\}$ 
36: end for
37: for each entity  $e'' \in E\_sub$  do
38:    $base\_text \leftarrow \text{Connection}(e'', A)$  // Multiple attribute connection
39:    $d \leftarrow e''.getAttribute("description")$ 
40:    $D'(e'') \leftarrow D'(e'') \cup \{base\_text \oplus d\}$ 
41: end for
42:  $SKG\_text \leftarrow D' \oplus D'(e'')$ 

```

Algorithm 3 LLM-guided candidate set generation and optimization

Input: Text of the entity and relation in the subgraph $SKG\text{-}text$; Target entity e_t and relation r_t ; LLM M ; Sub-KG; Maximum selected item K ; Number of candidate items N

Output: Optimal set $OS_{M|\mathcal{E}|K}$ or $OS_{M|\mathcal{R}|K}$

- 1: $prompt_template_1 \leftarrow$ "Based on items" $\{e_t\}$ "and" $\{r_t\}$ "predict the candidate item set for another item in the triple. The requirements are: (a) the number of candidate items is" $\{N\}$, "(b) the candidate items are sorted in descending order of prediction probability, (c) output in set format."
- 2: $S_{M|\mathcal{E}|N}$ or $S_{M|\mathcal{R}|N} \leftarrow M(prompt_template_1)$
- 3: **for** each triple $(h, r, t) \in T_{sub}$ **do**
- 4: $prompt_template_2 \leftarrow$ "Please generate a coherent and natural text description based on" $\{(h, r, t)\}$.
- 5: $Sentence \leftarrow Sentence \cup \{M(prompt_template_2)\}$
- 6: **end for**
- 7: $prompt_template_3 \leftarrow$ "Please connect each item in" $\{Sentence\}$ "and optimize it into natural language text according to the following requirements. Requirement: (a) maintain semantic coherence, (b) eliminate redundant expressions, (c) ensure entity referential consistency."
- 8: $MSKG\text{-}text \leftarrow M(prompt_template_3)$
- 9: **for** each item $i \in S_{M|\mathcal{E}|N}$ or $S_{M|\mathcal{R}|N}$ **do**
- 10: $prompt_template_4 \leftarrow$ "Based on" $\{SKG\text{-}text\}$ "and" $\{MSKG\text{-}text\}$ "analyze the possibility of" $\{i\}$ "as a predicted item and display the reasons for the prediction. Requirement: output format (item, prediction probability, reason)."
- 11: $Set \leftarrow Set \cup \{M(prompt_template_4)\}$
- 12: **end for**
- 13: $OS_{M|\mathcal{E}|K}$ or $OS_{M|\mathcal{R}|K} \leftarrow$ Sort in descending order based on probability and select the top K values.

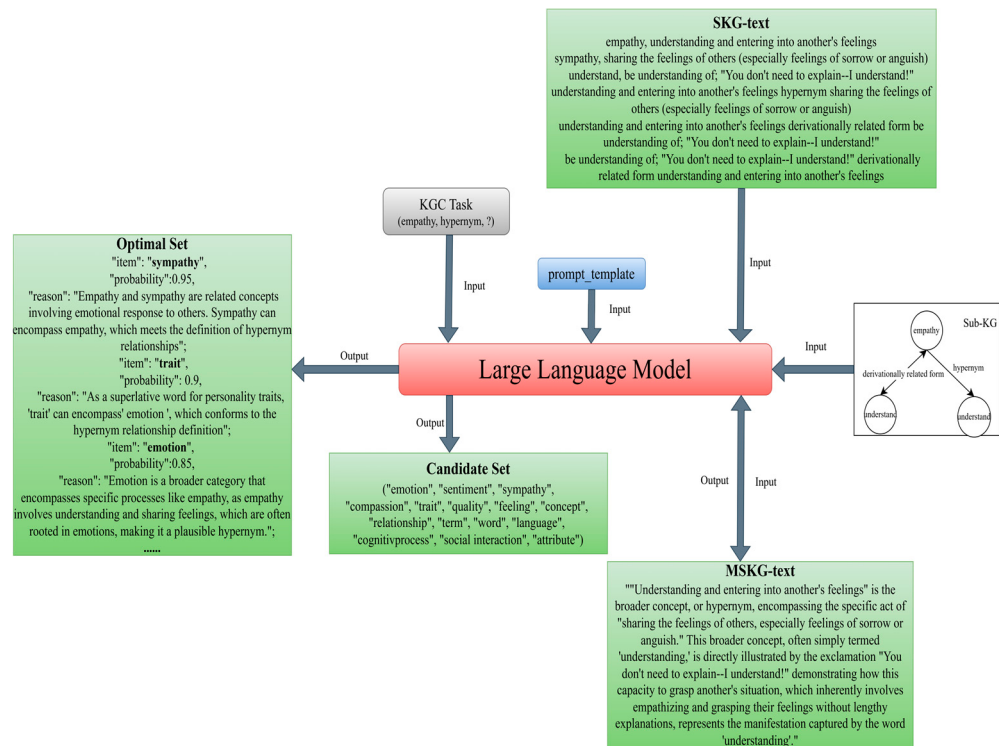


Figure 2. A sample of the LLM-guided candidate set generation and optimization method.

4. Experiment

In this section, we compare the performance of E-KGC with other baseline KGC methods to demonstrate the effectiveness of the E-KGC framework on common public datasets.

4.1. Experiment Settings

4.1.1. Datasets

We evaluate the E-KGC on the following datasets. Detailed information on the datasets is shown in Table 1.

Table 1. Summary statistics of datasets.

Dataset	Entity	Relation	Train	Valid	Test
WN18RR	40,943	11	86,835	3034	3134
FB15k-237	14,541	237	272,115	17,535	20,466

(1) WN18RR [17]: This was mainly used for the KGC tasks. It was extracted and refined from WordNet 18 dataset, aiming to solve the problem of reverse relationships in the original dataset. During the construction process, all relationships in WordNet 18 were screened first to remove the relationship pairs that could be derived from simple reverse relationships so as to ensure the complexity and challenge of the dataset.

(2) FB15k-237 [27]: This is a subset of freebase. It was created by deleting a large number of reversible relationship data in FB15k. All insignificant triples were removed to guarantee that no entities linked within the training set had direct connections to those in the verification set or the test set.

4.1.2. Baselines

To demonstrate the effectiveness, we compared the proposed method with the traditional methods, using the PLM-based and LLM-based methods as baselines. Table 2 shows the baseline descriptions of the three methods.

Table 2. Baselines.

Baseline	Type	Description
TransE [14]	KGE-based	By representing entities and relationships as translational relationships in vector space.
DistMult [15]	KGE-based	The interaction between entities and relationships is represented by bilinear models.
SACN [18]	GNN-based	It has the capability to aggregate local information from neighboring nodes within the graph for each individual node.
R-GCN [28]	GNN-based	It achieves this by uniformly considering the neighborhood of each entity through hierarchical propagation rules, making it well suited for directed graphs.
KG-BERT [8,20]	PLM-based	It demonstrates that PLMs can be fine-tuned on triple classification tasks by encoding entities and relations in textual form.
StAR [21]	PLM-based	It integrates structural constraints into PLMs using entity markers, knowledge-aware attention, or auxiliary graph encoders to address the semantic-structure gap.
KLR-KGC [10]	LLM-based	It directs the LLM to filter and prioritize candidate entities, thereby constraining its output to minimize omissions and erroneous answers.
MPIKGC-S [29]	LLM-based	It compensates for the deficiency of contextualized knowledge and improve KGC by querying LLMs from various perspectives.
GS-KGC [26]	LLM-based	It leverages subgraph details for contextual reasoning and adopts a question-answering (QA) strategy to accomplish the KGC tasks.

4.1.3. Metrics

Frequently employed evaluation metrics for KGC include Hits@K and Mean Reciprocal Rank (MRR) [30].

Hits@K indicates the percentage of all of the accurately identified samples that are ranked among the top-K positions following scoring. Hits@K divides the accumulated value of the number of triples ranked in the top K in the test set by the number of all triples in the test set, and the value range is [0, 1]. The larger the value of Hits@K, the better the KGC model. Hits@K is expressed as

$$\text{Hits@K} = \frac{\sum_{(h,r,t) \in \text{test}} [\text{rank}(h,r,t) \leq K?1:0]}{\text{count}(\text{test})}, \quad (3)$$

Among them, $\text{rank}(h,r,t) \leq K?1:0$ is the conditional expression to judge whether the sample ranking is within the top K. If $\text{rank}(h,r,t) \leq K$ is established, the expression value is 1, otherwise it is 0. $\text{count}(\text{test})$ represents the total number of triples in the test set.

MRR reflects the overall ranking of the correct sample in the candidate set. MRR calculates the reciprocal of the ranking positions, ensuring the outcome lies within the range of [0, 1]. A higher MRR value, approaching closer to 1, signifies a superior model performance. MRR is expressed as

$$\text{MRR} = \frac{\sum_{(h,r,t) \in \text{test}} \frac{1}{\text{rank}(h,r,t)}}{\text{count}(\text{test})}, \quad (4)$$

4.1.4. Experiment Details

Experiments were conducted using an NVIDIA GeForce RTX 4090 (GPU), 12th Gen Intel (R) Core (TM) i7-12700K 3.60 GH (CPU), 64 GB memory, and 1.7 TB hard drive. We adopted the DeepSeek model, deployed locally. The advantage of local deployment is that it is not affected by network fluctuations and cloud service provider failures, and also avoids the experiment interruption or data loss caused by network problems. DeepSeek-R1 is specifically trained on a massive dataset of logic and reasoning problems-solving compared to a general-purpose base model like LLaMA. It is fine-tuned to understand and follow user instructions precisely without requiring additional fine-tuning. The DeepSeek-R1:8b model, as a medium-sized version of the DeepSeek-R1 series, has the characteristics of moderate parameter size, strong performance, and reasonable resource consumption. In particular, it has significant advantages in complex semantic understanding and content generation. Based on our proposed framework and algorithms, DeepSeek-R1:8b is very suitable for application in our experiments. To minimize the randomness in the model's output, the parameters were configured as $\text{top_p} = 0.8$, $\text{temperature} = 0.1$ and $\text{max_hops} = 1$.

4.2. Experiment Results

4.2.1. E-KGC Performance

In this section, nine baselines are used to compare with E-KGC in terms of Hits@K and MRR utilizing the WN18RR and FB15K-237 to verify the performance of E-KGC. The goal of the experiments is the improvement of completion performance. The results are shown in Table 3. The result comparison chart is shown in Figures 3 and 4.

Table 3. Performance comparison in Hits@K. “-” means unavailable results. The bold values are the best values in each block.

	Model	WN18RR				FB15K-237			
		H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR
KGE-based	TransE	0.043	0.441	0.532	0.243	0.198	0.376	0.441	0.279
	DistMult	0.390	0.440	0.490	0.430	0.155	0.263	0.419	0.241
GNN-based	SACN	0.430	0.480	0.540	0.470	0.260	0.390	0.540	0.350
	R-GCN	0.080	0.137	0.207	0.123	0.100	0.181	0.300	0.164
PLM-based	KG-BERT	0.412	0.465	0.524	0.438	0.197	0.289	0.420	0.268
	StAR	0.222	0.436	0.647	0.364	0.171	0.287	0.452	0.263
LLM-based	KLR-KGC	0.476	0.542	0.587	0.516	0.323	0.449	0.574	0.404
	MPIKGC-S	0.497	0.568	0.652	0.549	0.267	0.395	0.543	0.360
	GS-KGC	0.346	0.516	-	-	0.280	0.426	-	-
	E-KGC	0.462	0.692	0.769	0.579	0.200	0.533	0.667	0.372

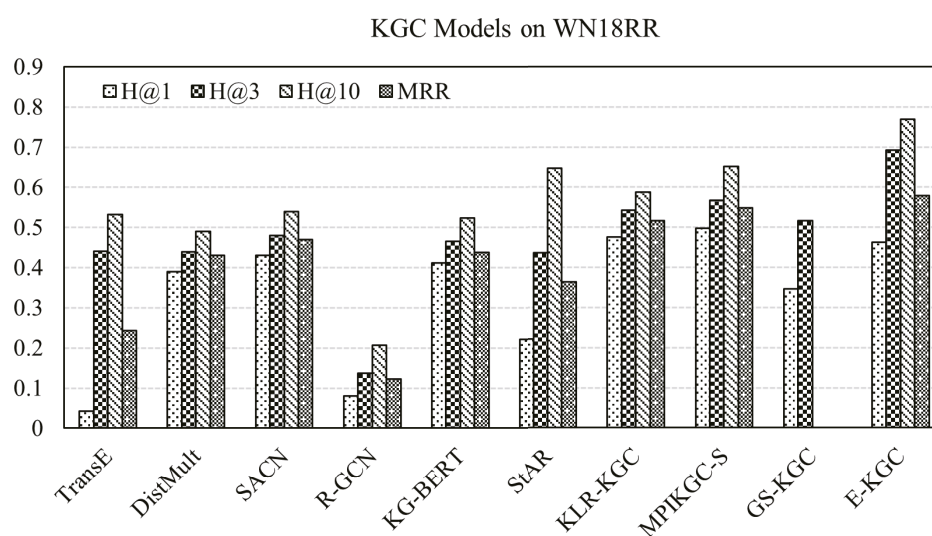


Figure 3. Hits@1, Hits@3, Hits@10 and MRR, in the WN18RR datasets.

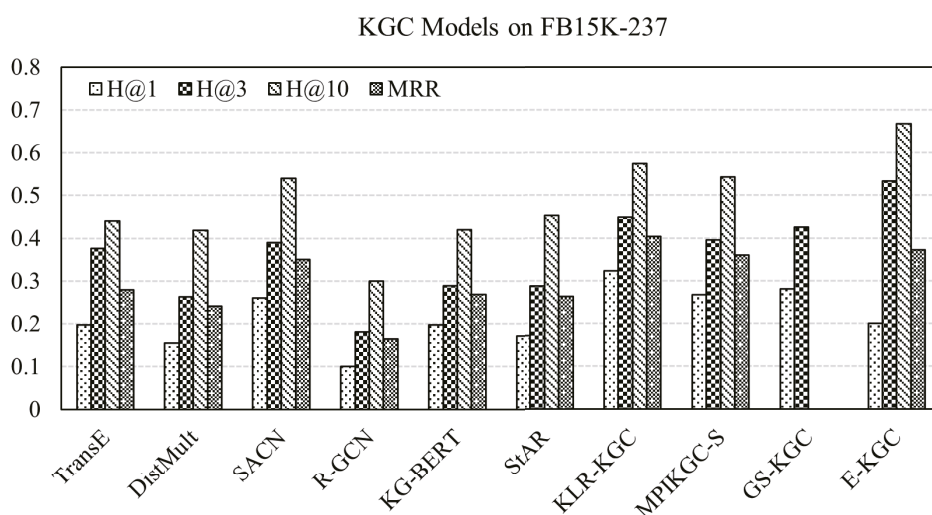


Figure 4. Hits@1, Hits@3, Hits@10, and MRR in the FB15K-237 datasets.

Based on the comprehensive experimental results, our proposed model demonstrates superior performance compared to existing baseline models across multiple evaluation metrics. As shown in Table 3, our model achieves remarkable improvements in KGC

tasks. Specifically, on the WN18RR dataset, H@3 increased by 12.4% and H@10 increased by 11.7%, showing that our model makes a qualitative leap in reasoning the accuracy and recall range. Additionally, our model attains an MRR score of 0.579, marking a 3% enhancement compared to the top-performing baseline model depicted in Figure 3. This notable increase in MRR underscores the model's ability to sustain consistently superior performance across the entire ranking list. Likewise, on the FB15K-237 dataset, the E-KGC model demonstrates a 1.2% improvement in MRR, an 8.4% rise in H@3, and a 9.3% increase in H@10, as shown in Figure 4.

The exceptional performance of our model stems from several pivotal innovations. First, our subgraph localization and knowledge acquisition strategy effectively captures the semantic relationships and contextual information within the KG, improving the efficiency of semantic understanding and content generation of LLMs. Second, the LLM-guided candidate set generation and optimization method provides richer external semantic information. Third, the integration of semantic similarity assessment using the DeepSeek model enables a more nuanced understanding of entity and relation meanings, going beyond surface-level textual matching.

4.2.2. Ablation Experiment

This section presents ablation experiments to assess the efficacy of our framework's core components: subgraph localization and knowledge acquisition strategy and LLM-guided candidate generation and optimization method. The contributions of each component are determined by evaluating Hits@K and MRR across different ablation scenarios. w/o SKG means the E-KGC removes the subgraph localization and knowledge acquisition strategy and directly uses the LLM to predict entities or relationships based on the triples to be predicted. w/o LLM means that E-KGC excludes LLM-guided candidate set generation and optimization method, relying instead on mined logic rules of subgraphs and using embedded model to learn entity and relation vector representation for prediction. w/o prompt means that the LLM-guide method removes all prompts. In other words, the LLM directly utilizes *SKG-text*, the subgraph, and the given entity and relation to obtain corresponding results without using prompts. The specific information is shown in Table 4.

Table 4. Ablation results. The bold values are the best values in each block.

Model	WN18RR				FB15K-237			
	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR
E-KGC	0.462	0.692	0.769	0.579	0.200	0.533	0.667	0.372
w/o SKG	0.325	0.452	0.673	0.525	0.154	0.493	0.572	0.314
w/o LLM	0.203	0.416	0.476	0.478	0.123	0.487	0.523	0.325
w/o prompt	0.109	0.352	0.426	0.464	0.087	0.411	0.443	0.265

Impact of subgraph localization and knowledge acquisition (without SKG): The ablation of the subgraph-guided knowledge module leads to performance declines across both datasets, underscoring its critical role. On WN18RR, the corresponding drops are 5.4% and 9.6%, respectively; on the FB15k-237 dataset, there is a decline of 5.8% in MRR and a decrease of 9.5% in H@10. These results confirm that the structural information provided by subgraph knowledge is essential for helping the LLM interpret entities and relations accurately, particularly in filtering out noise and correctly ranking candidates.

Impact of LLM-guided candidate set generation and optimization (without LLM): The absence of the LLM leads to substantial performance drops, with MRR falling by 10.1% and 4.7% on WN18RR and FB15k-237, respectively. This pronounced degradation, particularly in MRR, demonstrates that the model deprived of the LLM's linguistic prior

and reasoning capability fails to incorporate external knowledge, which severely limits its predictive performance.

Impact of prompt (without prompt): The findings indicate a notable decrease in performance when the prompts are not implemented. For all datasets, MRR and H@10 decreases by over 10% and 20%, respectively. The lack of prompts severely impede the effective reasoning skill of the LLM. These prompts play a crucial role in aiding the LLM to comprehend the reasoning process more thoroughly and arrive at well-informed decisions.

In summary, E-KGC consistently achieved the best performance on both datasets, and ablation studies confirm that each component of our framework contributes to this success.

5. Conclusions

In this paper, we introduce a novel structure-aware and semantic fusion framework for KGC, which effectively bridges the gap between the structural information inherent in KGs and the powerful semantic understanding capabilities of LLMs. E-KGC addresses the key challenges of leveraging LLMs for KGC, including their tendency for “hallucination,” inefficiency in handling large-scale graph data, and lack of explicit structural reasoning. Our contributions offer a unified KGC paradigm that fuses structural knowledge from KGs with the semantic capabilities of LLMs, enhancing both inference accuracy and interpretability. Experimental results on multiple benchmark datasets serve to substantiate the claim that our framework successfully harnesses the synergy between structured knowledge and LLMs, offering a robust, efficient, and interpretable path for advancing KGC. Our current research primarily focuses on static KGs. Extending it to dynamic KGs using different open-source LLMs presents a significant challenge and opportunity. Meanwhile, we recognize that statistical validation is the important standard in empirical research. We advocate for future large-scale studies using full statistical analysis to confirm our research.

Author Contributions: Conceptualization, data curation, methodology and original draft writing, J.H.; project administration, software and supervision, and writing—review and editing, H.A.; conceptualization, funding acquisition and validation, K.W.; formal analysis, validation, and writing—review and editing, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partly supported by the Shanxi Province Basic Research Program (Free Exploration Category), grant number 202403021222112, and the Science and Technology Innovation Foundation of Shanxi Agricultural University under grant 2020QC14.

Data Availability Statement: The data will be provided upon request.

Acknowledgments: The authors are equally grateful to the School of Software, Shanxi Agricultural University, for experimental equipment and providing support.

Conflicts of Interest: The authors affirm that there are no financial or personal relationships that could have influenced the work reported in this paper.

References

1. Kong, Y.; Liu, X.; Zhao, Z.; Zhang, D.; Duan, J. Bolt defect classification algorithm based on knowledge graph and feature fusion. *Energy Rep.* **2022**, *8*, 856–863. [\[CrossRef\]](#)
2. West, R.; Gabrilovich, E.; Murphy, K.; Sun, S.; Gupta, R.; Lin, D. Knowledge base completion via search-based question answering. In Proceedings of the 23rd International Conference on World Wide Web, Seoul, Republic of Korea, 7–11 April 2014; pp. 515–526.
3. GUAN Sai-Ping, J.X.-L. Knowledge Reasoning Over Knowledge Graph: A Survey. *J. Softw.* **2018**, *29*, 2966–2994.
4. Lao, N.; Cohen, W.W. Relational retrieval using a combination of path-constrained random walks. *Mach. Learn.* **2010**, *81*, 53–67. [\[CrossRef\]](#)
5. Wei, Y.; Luo, J.; Xie, H. KGRL: An OWL2 RL Reasoning System for Large Scale Knowledge Graph. In Proceedings of the 12th International Conference on Semantics, Knowledge and Grids (SKG), Beijing, China, 15–17 August 2016; pp. 83–89.

6. Kou, F.; Du, J.; Yang, C.; Shi, Y.; Liang, M.; Xue, Z.; Li, H. A multi-feature probabilistic graphical model for social network semantic search. *Neurocomputing* **2019**, *336*, 67–78. [\[CrossRef\]](#)
7. Zhao, F.; Yan, C.; Jin, H.; He, L. BayesKGR: Bayesian Few-Shot Learning for Knowledge Graph Reasoning. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2023**, *22*, 1–21. [\[CrossRef\]](#)
8. Yao, L.; Mao, C.; Luo, Y. KG-BERT: BERT for Knowledge Graph Completion. *arXiv* **2019**, arXiv:1909.03193. [\[CrossRef\]](#)
9. Wang, L.; Zhao, W.; Wei, Z.; Liu, J. SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL, Dublin, Ireland, 22–27 May 2022; Volume 1, pp. 4281–4294.
10. Ji, S.; Liu, L.; Xi, J.; Zhang, X.; Li, X. KLR-KGC: Knowledge-Guided LLM Reasoning for Knowledge Graph Completion. *Electronics* **2024**, *13*, 5037. [\[CrossRef\]](#)
11. Zhang, L.; Wang, W.; Jiang, Z.; Wang, Z. Exploring Knowledge from Knowledge Graphs and Large Language Models for Link Prediction. In Proceedings of the 2024 IEEE International Conference on Knowledge Graph (ICKG), Abu Dhabi, United Arab Emirates, 11–12 December 2024; pp. 485–491.
12. Zhang, Y.; Chen, Z.; Guo, L.; Xu, Y.; Zhang, W.; Chen, H. Making Large Language Models Perform Better in Knowledge Graph Completion. In Proceedings of the 32nd ACM International Conference on Multimedia, Melbourne, VIC, Australia, 28 October–1 November 2024; pp. 233–242.
13. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.* **2025**, *43*, 1–55. [\[CrossRef\]](#)
14. Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-relational Data. In Proceedings of the Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 2787–2795.
15. Yang, B.; Yih, W.; He, X.; Gao, J.; Deng, L. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of the 3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA, 7–9 May 2015.
16. Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, É.; Bouchard, G. Complex Embeddings for Simple Link Prediction. In Proceedings of the 33rd International Conference on Machine Learning, ICML, New York, NY, USA, 19–24 June 2016; JMLR Workshop and Conference Proceedings. Volume 48, pp. 2071–2080.
17. Dettmers, T.; Minervini, P.; Stenetorp, P.; Riedel, S. Convolutional 2D Knowledge Graph Embeddings. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI-18), New Orleans, LA, USA, 2–7 February 2018; pp. 1462–2061.
18. Shang, C.; Tang, Y.; Huang, J.; Bi, J.; He, X.; Zhou, B. End-to-End Structure-Aware Convolutional Networks for Knowledge Base Completion. In Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI-19), Honolulu, HI, USA, 27 January–1 February 2019; pp. 3060–3067.
19. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
20. Kim, B.; Hong, T.; Ko, Y.; Seo, J. Multi-Task Learning for Knowledge Graph Completion with Pre-trained Language Models. In Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain, 8–13 December 2020; pp. 1737–1743.
21. Zhang, J.; Shen, B.; Zhang, Y. Learning hierarchy-aware complex knowledge graph embeddings for link prediction. *Neural Comput. Appl.* **2024**, *36*, 13155–13169. [\[CrossRef\]](#)
22. Petroni, F.; Rocktäschel, T.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A.H.; Riedel, S. Language models as knowledge bases? *arXiv* **2019**, arXiv:1909.01066. [\[CrossRef\]](#)
23. Li, W.; Ge, J.; Feng, W.; Zhang, L.; Li, L.; Wu, B. SIKGC: Structural Information Prompt Based Knowledge Graph Completion with Large Language Models. In Proceedings of the 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Kuching, Malaysia, 6–10 October 2024; pp. 1116–1123.
24. Guan, X.; Liu, Y.; Lin, H.; Lu, Y.; He, B.; Han, X.; Sun, L. Mitigating large language model hallucinations via autonomous knowledge graph-based retrofitting. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 24 March 2024; Volume 38, pp. 18126–18134.
25. Yao, L.; Peng, J.Z.; Mao, C.S.; Luo, Y. Exploring Large Language Models for Knowledge Graph Completion. In Proceedings of the 2025 IEEE International Conference on Acoustics, Speech, and Signal Processing, Hyderabad, India, 6–11 April 2025.
26. Yang, R.; Zhu, J.; Man, J.; Liu, H.; Fang, L.; Zhou, Y. GS-KGC: A generative subgraph-based framework for knowledge graph completion with large language models. *Inf. Fusion* **2025**, *117*, 102868. [\[CrossRef\]](#)
27. Toutanova, K.; Chen, D.; Pantel, P.; Poon, H.; Choudhury, P.; Gamon, M. Representing Text for Joint Embedding of Text and Knowledge Bases. In Proceedings of the Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 1499–1509.

28. Schlichtkrull, M.; Kipf, T.N.; Bloem, P.; van den Berg, R.; Titov, I.; Welling, M. Modeling Relational Data with Graph Convolutional Networks. *Semant. Web* **2018**, *10843*, 593–607.
29. Xu, D.; Zhang, Z.; Lin, Z.; Wu, X.; Zhu, Z.; Xu, T.; Zhao, X.; Zheng, Y.; Chen, E. Multi-perspective Improvement of Knowledge Graph Completion with Large Language Models. In Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italia, 20–25 May 2024; pp. 11956–11968.
30. Akrami, F.; Saeef, M.S.; Zhang, Q.; Hu, W.; Li, C. Realistic Re-evaluation of Knowledge Graph Completion Methods: An Experimental Study. In Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, 14–19 June 2020; pp. 1995–2010.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.