Multi-perspective Improvement of Knowledge Graph Completion with Large Language Models

Derong Xu[1,3], Ziheng Zhang[2], Zhenxi Lin[2], Xian Wu[2*], Zhihong Zhu[4], Tong Xu[1*], Xiangyu Zhao[3*], Yefeng Zheng[2] and Enhong Chen[1]
[1]University of Science and Technology of China & State Key Laboratory of Cognitive Intelligence,
[2]Jarvis Research Center, Tencent YouTu Lab, [3]City University of Hong Kong, [4]Peking University
derongxu@mail.ustc.edu.cn, {tongxu, cheneh}@ustc.edu.cn, zhihongzhu@stu.pku.edu.cn,
{zihengzhang, chalerislin, kevinxwu, yefengzheng}@tencent.com, xianzhao@cityu.edu.hk

Abstract

Knowledge graph completion (KGC) is a widely used method to tackle incompleteness in knowledge graphs (KGs) by making predictions for missing links. Description-based KGC leverages pre-trained language models to learn entity and relation representations with their names or descriptions, which shows promising results. However, the performance of description-based KGC is still limited by the quality of text and the incomplete structure, as it lacks sufficient entity descriptions and relies solely on relation names, leading to sub-optimal results. To address this issue, we propose MPIKGC, a general framework to compensate for the deficiency of contextualized knowledge and improve KGC by querying large language models (LLMs) from various perspectives, which involves leveraging the reasoning, explanation, and summarization capabilities of LLMs to expand entity descriptions, understand relations, and extract structures, respectively. We conducted extensive evaluation of the effectiveness and improvement of our framework based on four description-based KGC models and four datasets, for both link prediction and triplet classification tasks.

Keywords: Knowledge Graph Completion, Large Language Models

1. Introduction

A knowledge graph (KG) is a type of multi-relational graph data that contains the name/description of entities and relations and presents relational facts in a triplet format (Zhu et al., 2022). Examples of KGs include Freebase (Bollacker et al., 2008), DBpedia (Lehmann et al., 2015), and YAGO (Mahdisoltani et al., 2015), which have been proven useful in various applications, such as recommender systems (Sun et al., 2020) and knowledge graph question answer (Saxena et al., 2020). Despite their widespread applications, KGs still suffer from the problem of incompleteness (Xu et al., 2023b). Along this line, the task of knowledge graph completion (KGC), which aims at predicting missing facts within a KG, helps both the construction and canonicalization of KGs.

There have been several proposed works on KGC that aim to predict missing facts, such as structure-based KGC (Sun et al., 2019; Dettmers et al., 2018; Yue et al., 2023) and description-based KGC (Wang et al., 2022a,b; Jiang et al., 2023). The structure-based KGC only considers graph structural information from observed triple facts and embeds each entity and relation separately into trainable index embeddings. Unlike structure-based KGC, description-based KGC methods encode the text of entities and relations into a semantic space using pre-trained language models. The plausibility

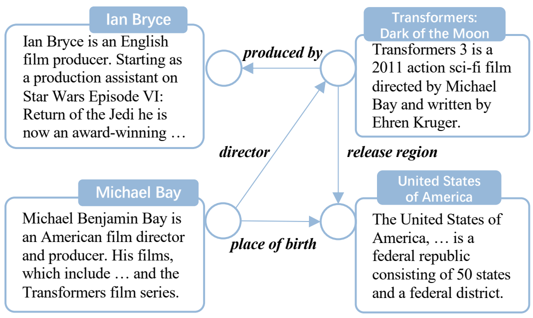

Figure 1: A subgraph with entity descriptions from a KG. The text of relations only includes its name.

of facts is predicted by computing a scoring function of triplet or matching semantic similarity between the [head entity, relation] and tail entity (Wang et al., 2022a). In this way, the textual encoder facilitates easy generalization of the model to unseen graph entities, resulting in better scalability than index entity embedding.

However, despite the significant success achieved by description-based KGCs in learning textual and structural knowledge, their effectiveness is still limited by the quality of Internet-crawled text and incomplete structure. For instance, in Figure 1, the brief descriptions of *"Ian Bryce"* and *"Transformers: Dark of the Moon"* are uninformative. In this case, relying solely on the name of relation *"produced by"* may result in the ambiguous understanding of entity types.

* Corresponding authors. This work was done when Derong Xu was an intern at Tencent.

LREC-COLING 2024, pages 11956–11968
20-25 May, 2024.

Meanwhile, learning structural patterns from known graphs is challenging for long-tailed entities. These limitations make it hard for KGC to achieve high performance in real-world applications that involve insufficient and incomplete knowledge graphs.

Recently, large language models (LLMs) have been shown to have massive context-aware knowledge and advanced capability through the process of pre-training on large-scale corpora (Liang et al., 2022; Peng et al., 2023). Therefore, it is worthwhile to utilize the rich knowledge of LLMs to address the challenges of KGC. However, it raises a question: *how to effectively leverage the capabilities and knowledge of a language model to improve the graph learning?*

To answer the above question, in this paper, we propose a novel technical framework, called MPIKGC, which prompts LLMs to generate auxiliary texts for improving the performance of KGC models. Specifically, to address the problem of incomplete entity information, we propose to expand the entity description using the knowledge captured by LLMs. We achieve this by designing a Chain-of-Thought (CoT) prompt (Wei et al., 2022) that allows the LLM to break down the query into different aspects and generate descriptions step-by-step. In addition to addressing the issue of relation ambiguity, we further propose a solution to improve KGC models' understanding of relation meanings, which involves querying LLMs with three curated prompting strategies, namely global, local, and reverse prompts that capture the association between relations and facilitate better reverse prediction. Moreover, to address the issue of sparse graph links, especially for long-tailed entities, we propose enriching knowledge graphs by querying LLMs to extract additional structural information, using the keywords summarized by LLMs to measure similarity between entities, and creating new triples that construct associations between related entities and enable the formation of new structural patterns in KGC models.

To demonstrate the effectiveness and universality of our proposed framework, we apply the strategies from different perspectives to four description-based KGC models and four datasets separately, improving their performance on both the link prediction task and the triplet classification task. MPIKGC is also evaluated through a variety of ablation and comparison experiments, demonstrating its diversity for performance improvement from different perspectives and its generalizability across different LLMs. The codes and datasets are available in `https://github.com/quqxui/MPIKGC`.

2. Related Work

2.1. Description-based KGC Methods

Different from traditional structured-based KGC methods (Sun et al., 2019), which solely utilized structural information, description-based KGC methods typically represented entities and relations in KGs using pre-trained language models. Specifically, these methods utilized textual descriptions for embedding entities and relations, and for the long-tailed entities, description-based KGC methods usually performed well because of the representation learning brought by entity descriptions. For example, DKRL (Xie et al., 2016) employed a convolutional neural network to encode entity descriptions, while KG-BERT (Yao et al., 2019) utilized pre-trained BERT models to learn the embeddings of entities and relations, and KEPLER (Wang et al., 2021) further adapted pre-trained language models to simultaneously optimize knowledge embedding and language modeling objectives. MMRNS (Xu et al., 2022) leverages the similarity of description to perform negative sampling for hard samples. Also, SimKGC, proposed by Wang et al. (2022a), utilized contrastive learning and various types of negative sampling to improve performance. LMKE (Wang et al., 2022b), on the other hand, leveraged language models and textual information to generate knowledge embeddings for entities, particularly for long-tailed entities. Besides, CSProm-KG (Chen et al., 2023) extended frozen pre-trained language models with structural awareness using the proposed conditional soft prompt.

Despite the promising results of description-based KGC models, they still face difficulties caused by deficiency of textual data and incomplete structure, while our MPIKGC aims to address these challenges by introducing large language models to description-based KGC models.

2.2. Large Language Models for KG

Recently, the emergence of large language models (LLMs), such as GPT4 (OpenAI, 2023), Llama-2 (Hugo et al., 2023), and ChatGLM2 (Zeng et al., 2023), has led to several studies (Pan et al., 2023a) exploring the potential of integrating LLMs and KGs to achieve improved performance by leveraging the strengths of both modalities. KGs can enhance LLMs by providing a means of explicitly storing rich factual knowledge, while LLMs can aid in the construction of KGs by generating new facts (Pan et al., 2023b; Xu et al., 2023a). Specifically, Liang et al. (2022) revealed that LLMs perform well on frequent entities and relations that mostly occur in the pre-training data. The Chain-of-Thought (CoT) (Wei et al., 2022) prompting strategy significantly improved the reasoning performance of LLMs without

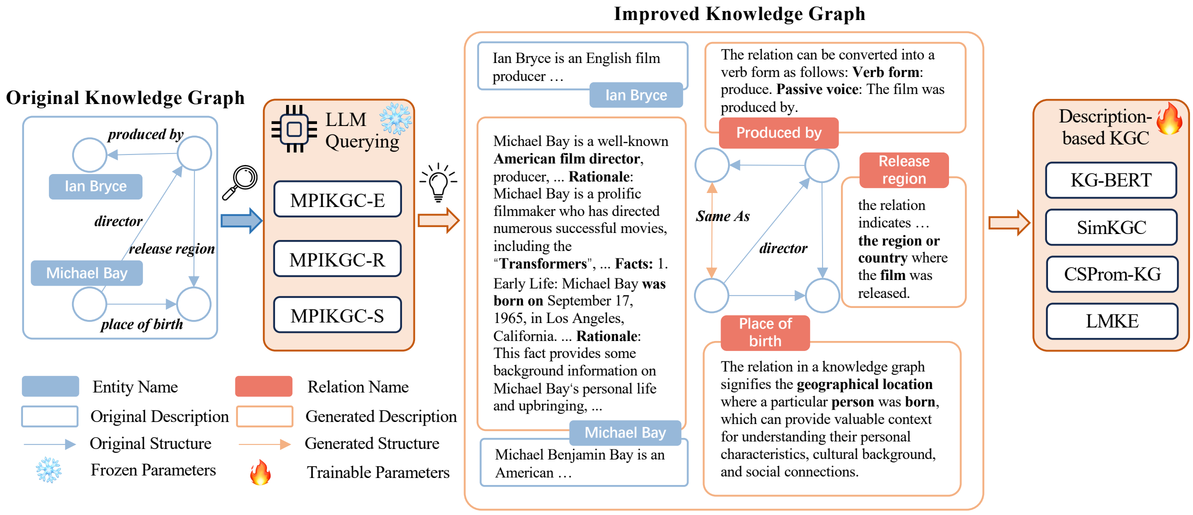

Figure 2: Framework of improving knowledge graph from the perspective of entity, relation, and structure. We evaluate the efficiency of the enhanced knowledge graph by employing various description-based KGC models on link prediction and triplet classification tasks.

requiring further fine-tuning. Besides, Mruthyunjaya et al. (2023) exhibited the considerable potential of LLMs in recalling factual information for symbolic KGs. Moreover, LLMs have shown remarkable zero-shot performance in named entity recognition (Li et al., 2023) as they can extract structural knowledge by utilizing relevant external context information. Given these findings, we propose to investigate the potential of LLMs in terms of the reasoning, explanation, and summarization abilities for improving the KGC task.

3. Methodology

3.1. Problem Definition

Knowledge graph $\mathcal{G}$ is a heterogeneous and direct graph data structure that can be represented as a collection of triplets with descriptions, which could be represented as $\mathcal{G} = \{(h, r, t, d)\} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{D}$, where $\mathcal{E}$ denotes the entity set, $\mathcal{R}$ as the relation set, and $\mathcal{D}$ as the original description of entities and relations. The aim of the triplet classification task is to ascertain the accuracy of a given triplet. The link prediction task of KGC is to infer the missing facts based on the known textual and structural data, which comprises two parts, i.e., predicting the tail entity when given $(h, r, ?)$, and predicting the head entity when provided with $(?, r, t)$. To accomplish this, it is necessary to rank all entities within $\mathcal{E}$ by calculating a score function for both positive and negative triples. Description-based KGC models utilize pre-trained language models to encode $\mathcal{D}$ and learn representations of entities and relations, while our goal is to enhance the textual and structural data as input of KGC models by querying LLMs with curated prompts.

3.2. Multi-perspective Prompting

In this section, we elaborate on the pipeline of our MPIKGC and provide motivation for each prompt, along with an illustrative example for better understanding. Our approach involves enhancing knowledge graph completion by improving entity, relation, and structure data, as shown in Figure 2. Specifically, templates for querying LLMs are shown in Table 1, in which prompts used for querying LLMs follow three fundamental principles. (1) Clarity: It is crucial for LLMs to adhere to our instructions precisely. An excessively complex prompt may result in misunderstandings in the instructions, particularly for small LLMs (with fewer than 10 billion parameters), ultimately reducing the effectiveness of communication. (2) Universality: The prompts we design should be compatible with various LLMs, and the generated text from these LLMs consistently demonstrates improvement on multiple KGC models, in both link prediction and triplet classification tasks. (3) Diversity: Prompts should demonstrate diversity to enrich KG data from various perspectives: including entity, relation, and structure. They can improve the learning of KGC models and show cooperative effects when combined. We evaluated the three claims of our proposed framework in the experiments presented in Section 4.

3.2.1. Description Expansion

Formalizing the comprehensive knowledge of an entity from LLMs is non-trivial, as it is difficult to ascertain whether the LLMs has generated and en-

Strategies	Templates
MPIKGC-E	Please provide all information about {Entity Name}. Give the rationale before answering:
MPIKGC-R Global	Please provide an explanation of the significance of the relation {Relation Name} in a knowledge graph with one sentence:
MPIKGC-R Local	Please provide an explanation of the meaning of the triplet (head entity, {Relation Name}, tail entity) and rephrase it into a sentence:
MPIKGC-R Reverse	Please convert the relation {Relation Name} into a verb form and provide a statement in the passive voice:
MPIKGC-S	Please extract the five most representative keywords from the following text: {Entity Description}. Keywords:

Table 1: Templates of each strategy for querying. MPIKGC-E, -R, and -S are corresponding to the improvement methods in terms of entity, relation, and structure, respectively.

compassed all the information of this entity. Meanwhile, it can be challenging to manually set many instructions for each entity to query about, such as asking the released region or director for a movie, which consumes a lot of manpower and often results in an excessive number of tokens being input into the LLMs, consequently increasing the computational burden of inference. Such long text may also not be suitable for small-scale LLMs and can hamper their performance (Bai et al., 2023).

We propose to design a Chain-of-Thought (CoT) (Wei et al., 2022) prompt strategy, that enables LLMs to break down complex queries into different directions and generate descriptions step-by-step, without the need for explicit manual input. It instructs LLMs to implicitly query relevant information on their own, resulting in more efficient and extensive responses. As demonstrated template in Table 1 **MPIKGC-E**, We request LLMs to provide a comprehensive entity description and provide a rationale before answering, which serves as justification for the answer and improves the recall of KGC models. For instance, Figure 2 presents an example of a famous person *"Michael Bay"*, where LLMs generate a description containing various occupations and personal details of the individual, accompanied by rationales for each response to enhance the LLMs statement.

3.2.2. Relation Understanding

The presence of heterogeneous relations in a knowledge graph plays a crucial role in distinguishing between two entities. However, relying solely on relation names may lead to ambiguous interpretations, particularly for complex relation categories (such as many-to-many and many-to-one). Moreover, the link prediction task requires an additional reverse prediction, i.e., predicting the head entity given $(?, r, t)$. Typically, the performance of reverse prediction for many-to-one relation is significantly lower than that of forward prediction (Yang et al., 2014). Structure-based KGC methods attempt to address this issue by adding a reverse relation for each forward relation, thereby doubling the trainable index embeddings for relations. In contrast, description-based KGC methods, such as SimKGC (Wang et al., 2022a), append a string *"reverse"* to the relation name. We argue that such an approach does not enable models to fully comprehend the meaning of relations, resulting in poor performance.

Therefore, we propose three prompting strategies, namely Global, Local, and Reverse, as depicted by MPIKGC-R in the Table 1. Specifically, **MPIKGC-R Global** aims to deduce the significance of a relation from the perspective of the entire KG, thereby facilitating better association between two relations. For instance, both *"produced by"* and *"director"* are related to the film industry, while *"release region"* and *"place of birth"* are associated with the name of a country or area. In contrast, **MPIKGC-R Local** intends to infer the relation's meaning from the triplet perspective, thereby enhancing comprehension and suggesting possible types of head/tail entities while predicting missing facts. For instance, when querying the meaning of *"(head entity, release region, tail entity)"*, LLMs suggest that the relation may connect to films and regions. In addition, **MPIKGC-R Reverse** entails LLMs to represent relations as verbs, and convert them to the passive voice. For example, *"produce"* can be transformed into *"produced by"*, thereby enhancing comprehension and enabling better reverse prediction. The generated text is appended to the relation name and is processed according to each KGC model's workflow for handling the relation name.

Dataset	#Ent	#Rel	#Train	#Valid	#Test
FB15k237	14,541	237	272,115	17,535	20,466
WN18RR	40,943	11	86,835	3,034	3,134
FB13	75,043	13	316,232	5,908	23,733
WN11	38,696	11	112,581	2,609	10,544

Table 2: Statistics of KGs used for our work.

3.2.3. Structure Extraction

KGC models are capable of learning structural patterns from training triples and generalizing to the missing links in test triples. For instance, a person entity that has the occupation of a producer or director, is probably related to film entities. However, pattern learning from graph structures is limited to sparse links, particularly for long-tailed entities (Li et al., 2022). To address this problem, we propose **MPIKGC-S**, which queries LLMs to generate additional structural information to enrich KGs. To convert the LLMs' generative text into graph-based data, we utilize the summarizing capability of LLMs to extract relevant keywords from description, then calculate a matching score s between entities based on the number of matched keywords:

$$s = len(m)/min(len(k_h), len(k_t)),$$
$$m = intersection(k_h, k_t),$$

where k_h and k_t denote the keywords of head/tail entities, respectively, and m is the intersection of k_h and k_t. After sorting the matching score, we selected top k pairs and created new triples in the form of *(head, Same As, tail)*, which are then appended to the training set. In addition to these similarity-based triplets, we also consider adding a self-loop triplet with the relation "SameAs" to each entity: *(head, Same As, head)*. The motivation is to enhance the KGC model's learning of the "SameAs" relation. These extra triplets construct the association between related entities and allow for the formation of new structural patterns in KGC models. For instance, by adding "SameAs" relation between *"Ian Bryce"* and *"Michael Bay"*, *"Ian Bryce"* can reach to the *"Transformers: Dark of the Moon"* entity with a explicit path, thereby serving as a valuable addition to the KGC model learning process.

4. Experiments

4.1. Experimental Setup

Datasets. We conduct link prediction experiments on two widely used datasets, namely FB15k237 (Toutanova and Chen, 2015) and WN18RR (Dettmers et al., 2018), as well as triplet classification experiments on FB13 (Socher et al., 2013) and WN11 (Socher et al., 2013).

Metrics. We evaluate the performance of KGC models using the following metrics: Mean Rank (MR), Mean Reciprocal Rank (MRR), and Hits@n (H@n, n={1,3,10}) for the link prediction task, and Accuracy for the triplet classification task. Lower MR values indicate better performance, while higher values for other metrics are indicative of better performance.

Baselines. In our study, we compare our improved KGs with the original KGs using four description-based KGC models: KG-BERT (Yao et al., 2019), SimKGC (Wang et al., 2022a), LMKE (Wang et al., 2022b), and CSProm-KG (Chen et al., 2023). The criteria for selecting baselines are based on state-of-the-art performance, the model's novelty, and the experimental time cost. As KGC requires ranking all candidate entities, we prioritized baselines that utilized a 1-to-n scoring method. We also compared against traditional structure-based KGC models, including TransE (Bordes et al., 2013), DistMult (Yang et al., 2014), RotatE (Sun et al., 2019), ConvE (Dettmers et al., 2018), ConvKB (Nguyen et al., 2018), and ATTH (Chami et al., 2020).

Backbones. We rely on Llama-2 (Llama-2-7b-chat) (Hugo et al., 2023), ChatGLM2-6B (Zeng et al., 2023), ChatGPT (gpt-3.5-turbo-0613) [1], GPT4 (gpt-4-0613) (OpenAI, 2023) as our primary text generation backbones. In both LLMs, we set the temperature to 0.2 and the maximum length to 256, and use single-precision floating-point (FP32) for inference. We employ BERT (bert-based-uncased) (Yao et al., 2019) as the backbone to encode generated text for all description-based KGC models, with the hyperparameters being set in accordance with the corresponding models.

Settings. To ensure a fair comparison, we reproduce each method using their open-source codes and utilized the "`bert-based-uncased`" version (Yao et al., 2019) as the backbone for all models. To account for the increased amount of text for both entities and relations in the enhanced KGs, we ensure that different augmentation experiments have the same maximal token length and data processing pipeline. Additionally, we include Mean Rank results, which were not reported by some baselines. Other parameters are set following the default parameters provided in the original paper. To ensure the reproducibility of our results, we provide details on the hyper-parameters used for the four baselines across four benchmarks in Appendix 9.1, which includes information on hyper-parameters such as maximum token length and batch size. Additional experiments are available in Appendix 9.2 and 9.3.

Models	FB15k237 MR↓	MRR↑	H@1↑	H@3↑	H@10↑	WN18RR MR↓	MRR↑	H@1↑	H@3↑	H@10↑
Structure-based Approaches										
TransE (Bordes et al., 2013)	323	27.9	19.8	37.6	44.1	2300	24.3	4.3	44.1	53.2
DistMult (Yang et al., 2014)	512	28.1	19.9	30.1	44.6	7000	44.4	41.2	47.0	50.4
ConvE (Dettmers et al., 2018)	245	31.2	22.5	34.1	49.7	4464	45.6	41.9	47.0	53.1
RotatE (Sun et al., 2019)	177	33.8	24.1	37.5	53.3	3340	47.6	42.8	49.2	57.1
ATTH (Chami et al., 2020)	-	34.8	25.2	38.4	54.0	-	48.6	44.3	49.9	57.3
Description-based Approaches										
CSProm-KG (Chen et al., 2023)	188	35.23	26.05	38.72	53.57	545	**55.10**	**50.14**	**57.04**	64.41
+MPIKGC-E	195	35.51	26.38	38.96	53.74	1244	53.80	49.19	55.65	62.81
+MPIKGC-R	192	35.38	26.29	38.83	53.50	838	53.90	49.35	55.74	62.36
+MPIKGC-S	**179**	**35.95**	**26.71**	**39.52**	**54.30**	**528**	54.89	49.65	56.75	**65.24**
LMKE (Wang et al., 2022b)	**135**	30.31	21.49	33.02	48.07	**54**	55.78	42.91	64.61	79.28
+MPIKGC-E	138	30.83	21.89	33.67	48.75	57	56.35	43.27	65.54	**79.53**
+MPIKGC-R	145	**30.99**	**22.21**	**33.70**	48.83	59	**57.60**	**45.10**	**65.95**	79.35
+MPIKGC-S	**135**	30.68	21.67	33.35	**48.91**	70	50.71	36.91	59.65	76.13
SimKGC (Wang et al., 2022a)	146	32.66	24.13	35.42	49.65	148	65.64	57.08	71.20	80.33
+MPIKGC-E	**143**	33.01	24.37	35.80	50.29	**124**	65.64	57.10	71.09	80.41
+MPIKGC-R	156	31.05	22.63	33.62	47.65	129	**66.41**	**57.90**	**72.08**	**81.47**
+MPIKGC-S	**143**	**33.22**	**24.49**	**36.26**	**50.94**	170	61.48	52.81	66.77	76.94

Table 3: Experimental results on the link prediction task where the best results in each block are **in bold**. ↑: higher is better. ↓: lower is better.

4.2. Main Results

In this section, we conduct a comprehensive performance comparison between structure-based KGC and description-based KGC and evaluate the effectiveness of our MPIKGC in three perspectives by feeding the enhanced KGs to description-based approaches. As presented in Table 3, ATTH outperforms other structure-based methods across four metrics. Besides, we observe that in most situations, structure-based methods exhibited better performance on FB15k237, which contains general world facts (such as an entity of actor), while description-based methods perform better on WN18RR, a subset of WordNet (Miller, 1995) with rich language knowledge suitable for PLMs.

CSProm-KG proposed to focus on both textual and structural information, which results in its superior performance on FB15k237 compared to LMKE and SimKGC. However, it performs much worse on WN18RR. Our proposed methods, particularly the structure extraction approach MPIKGC-S, improve the structure-focused aspect of CSProm-KG and achieve the highest performance on FB15k237 compared to all other baselines, even surpassing the structure-based methods. However, the difference between the FB15k237 and WN18RR datasets is indeed noteworthy. FB15k237 has shown particularly good results with MPIKGC-S, which could be attributed to the fact that FB15k237 has 15K entities and 237 kinds of relations as shown in Table 2, while WN18RR has 40K entities with only 11 relations. Adding extra relation to WN18RR might excessively alter the sparsity and triplet distribution of the KG, leading to poor performance.

On the other hand, our proposed method for relation understanding (MPIKGC-R) demonstrates a 1%-2% improvement in the MRR, Hits@1, and Hits@3 metrics on WN18RR compared to LMKE, while MPIKGC-E achieves higher Hits@10 score of 79.53%. The same improvement trend is also observed for the MPIKGC-R approach when applied to SimKGC on WN18RR. The reason is that both methods are text-focused and have the capability to learn abundant information with enhanced data. However, MPIKGC-S does not exhibit improvement in LMKE and SimKGC on WN18RR. We hypothesize that this may be attributed to the low number of relation types, which could potentially mislead the model when adding new relations. Moreover, the incorporation of extracted structural data for FB15k237 achieves much better performance

[1] https://openai.com/blog/chatgpt

Models	FB13	WN11
Structure-based Approaches		
TransE (Bordes et al., 2013)	81.5	75.9
DistMult (Yang et al., 2014)	86.2	87.1
ConvKB (Nguyen et al., 2018)	88.8	87.6
Description-based Approaches		
KG-BERT (Yao et al., 2019)	84.74	93.34
+MPIKGC-E	**86.29**	**94.13**
+MPIKGC-R	84.51	93.36
+MPIKGC-S	85.35	93.61
LMKE (Wang et al., 2022b)	91.70	93.71
+MPIKGC-E	91.52	93.84
+MPIKGC-R	91.49	**93.93**
+MPIKGC-S	**91.81**	93.91

Table 4: Accuracy on the triplet classification task.

based on LMKE and SimKGC, which proves the effectiveness of MPIKGC-S in the link prediction.

4.3. Triplet Classification

In this section, we evaluate our proposed methods on the triplet classification task, a binary classification task that determines the correctness of a given triplet. Based on the results presented in Table 4, we can conclude that the structure-based methods perform well on the FB13 dataset, while significantly underperform compared to the description-based methods on the WN11 dataset. This outcome is consistent with the findings in the link prediction task discussed in Section 4.2 and can be attributed to the variations between Freebase (Bollacker et al., 2008) and WordNet (Miller, 1995). On the other hand, the results demonstrate that expanding descriptions (MPIKGC-E) is a promising technique to improve the KG-BERT's performance, as it yields a 1.55% higher accuracy score on FB13 and a 0.79% higher accuracy score on WN11. Our methods also exhibit minor enhancements on LMKE and achieve the highest accuracy score of 91.81% in FB13. The overall results indicate the universality of the MPIKGC framework that can enhance the performance of various KGC models in both link prediction and triplet classification tasks.

4.4. Parameter Analysis of Structure Extraction

In this section, we evaluate the significance of the hyper-parameter top k in extracting structural data. We present the results obtained on FB15k237 and compare the settings with and without self-loop with SimKGC. Increasing the value of k implies the addition of more triplets to the training set. For instance, when k is set to 1, we identify the best-matched entity for each entity by computing the matching score. Besides, the self-loop setting involves the inclusion of a triplet for the entity itself. As shown in Figure 3, the curves for all three metrics exhibit an upward trend under both settings as the k increases, which demonstrates that the augmentation perspective in structure indeed improves the performance of KGC on the FB15k237 dataset. Furthermore, we can see that the setting with self-loop consistently outperforms the setting without self-loop. This observation suggests that augmenting KGC's learning of the "SameAs" relation is a promising strategy for enhancing performance. However, we also note that the performance starts to decline when k reaches 4 or 5. This trend indicates that the training set contains excessive extraneous triplets, which could negatively affect the learning of other data.

Models	FB15k237			
	MRR	H@1	H@3	H@10
LMKE	30.31	21.49	33.02	48.07
+MPIKGC-E	30.71	21.97	33.29	48.35
+MPIKGC-R	30.64	21.70	33.22	48.74
+MPIKGC-S	30.68	21.67	33.35	48.91
+MPIKGC-E&R	30.74	21.77	33.57	48.77
+MPIKGC-E&S	30.92	21.85	33.67	**49.50**
+MPIKGC-R&S	**31.21**	**22.26**	33.86	49.42
+MPIKGC-E&R&S	30.97	21.91	**33.90**	49.28

Table 5: Ablation of augmentation methods from different perspectives.

4.5. Ablation of Multi-perspective Prompts

This section presents an analysis of the performance of description expansion (MPIKGC-E), relation understanding (MPIKGC-R), and structure extraction (MPIKGC-S), as well as the performance when combining them together. For instance, MPIKGC-E&R denotes the combination of MPIKGC-E and MPIKGC-R, while the remaining methods follow the same naming convention. We conduct ablation experiments on the FB15k237 dataset using Llama-2-generated texts, with LMKE as the baseline. The results demonstrate that after being enhanced with our method from the perspectives of entities, relations, and structures, the KGC models achieved a nearly 0.5% improvement across all four metrics. Additionally, MPIKGC-E&R combines the generated entity descriptions with the descriptive text of relations, resulting in a slight improvement over using either one individually, which

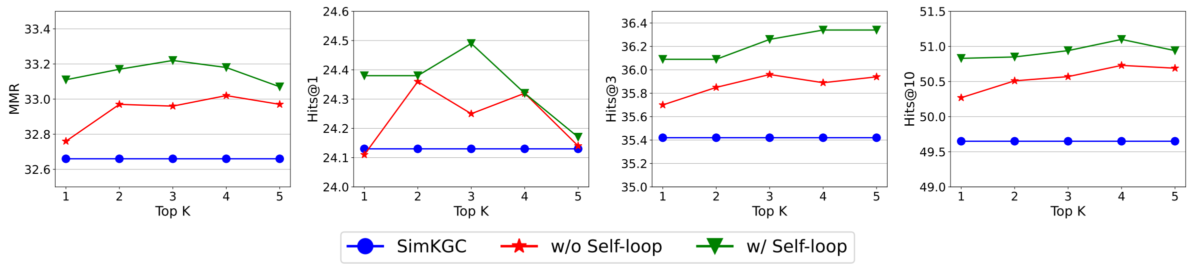

Figure 3: Analysis of hype-parameter k and the self-loop setting on FB15k237.

demonstrates the compatibility of these two methods. Moreover, MPIKGC-E&S achieves the highest H@10 score, while MPIKGC-R&S performs best on MRR and H@1. MPIKGC-E&R&S achieves the best H@3 score. We observe that incorporating the structure extraction method further improves performance by nearly 0.5% across most metrics. For example, MPIKGC-E&R&S gets a H@10 score of 49.28%, which is 0.51% higher than MPIKGC-E&R. The same phenomenon can be seen when adding relation text '-R'. The comprehensive findings indicate that our diverse enhancement methods are compatible and can be integrated to boost overall performance.

4.6. Ablation of Relation Understanding

In this section, we evaluate the performance of various ablation settings for relation understanding on the WN18RR dataset, as shown in Table 6.

Specifically, MPIKGC-R G&L represents the combination of Global and Local descriptions, which are concatenated using a separate token '[SEP]'. Meanwhile, the other methods follow the same rule. The results show that MPIKGC-R Global outperformed the baseline SimKGC by nearly 1% across all four metrics. Additionally, MPIKGC-R Local achieves the highest H@10 score of 81.57%, but has the lowest MRR and H@1 scores. Conversely, MPIKGC-R Reverse achieves over 1% improvement in MRR and H@1 but performs worse in H@10. These results suggest that MPIKGC-R Local prioritizes the top-10 recall of correct entities, while MPIKGC-R Reverse focuses on improving the performance of the best entity (i.e., top-1).

After combining these three strategies, we observe that MPIKGC-R G&L achieves a significant improvement in MRR and H@1, suggesting that Global and Local prompts have a complementary effect. However, other combined strategies exhibit poor performance. We therefore believe that incorporating too many relation descriptions may increase the difficulty of learning the meaning of the relation.

Models	WN18RR			
	MRR	H@1	H@3	H@10
SimKGC	65.64	57.08	71.20	80.33
+MPIKGC-R Global	66.41	57.90	**72.08**	81.47
+MPIKGC-R Local	64.45	54.87	70.65	**81.57**
+MPIKGC-R Reverse	66.53	59.28	70.72	80.09
+MPIKGC-R G&L	**66.97**	**59.88**	70.82	79.77
+MPIKGC-R G&R	65.56	57.00	70.98	80.90
+MPIKGC-R L&R	65.75	57.36	71.03	80.06
+MPIKGC-R G&L&R	65.85	57.47	70.98	80.64

Table 6: Ablation of different relation understanding strategies and combinations on WN18RR.

4.7. Comparison of LLMs

This section is dedicated to exploring the use of various LLMs to enhance KGC on FB15k237. Due to the long querying time and high costs associated with querying each entity and keyword on four benchmarks, we have restricted our analysis to the application of ChatGPT and GPT4 solely for MPIKGC-R. As shown in Table 7, the results indicate that our framework consistently improves KGC based on LMKE with generated text across three perspectives when using various LLMs. This suggests the effectiveness of our designed prompts, which are universal to both large-scale (ChatGPT and GPT4) and small-scale (Llama-2 and ChatGLM2) LLMs. Specifically, ChatGLM2 produces superior results for MPIKGC-E and MPIKGC-S compared to Llama-2, indicating ChatGLM2's advantage of reasoning and summarization abilities. However, Llama-2 and ChatGPT outperform ChatGLM2 in terms of their ability to understand relations. On the other hand, we can see that applying GPT4 into MPIKGC-R has led to a significant improvement in all metrics, owing to the larger model scale that facilitates a more comprehensive understanding of KG relations.

5. Conclusion

In this paper, we proposed MPIKGC, a novel framework that investigates improving the quality of KGs

Models	FB15k237			
	MRR	H@1	H@3	H@10
LMKE	30.31	21.49	33.02	48.07
+MPIKGC-E (Llama-2)	30.56	21.62	33.47	48.15
+MPIKGC-E (ChatGLM2)	**30.83**	**21.89**	**33.67**	**48.75**
+MPIKGC-R (Llama-2)	30.64	21.70	33.22	48.74
+MPIKGC-R (ChatGLM2)	30.24	21.33	32.96	48.27
+MPIKGC-R (ChatGPT)	30.65	21.82	33.24	48.52
+MPIKGC-R (GPT4)	**30.99**	**22.21**	**33.70**	**48.83**
+MPIKGC-S (Llama-2)	30.68	21.67	33.35	**48.91**
+MPIKGC-S (ChatGLM2)	**31.07**	**22.26**	**33.81**	48.82

Table 7: Ablation of different LLMs on FB15k237.

by querying LLMs from three perspectives: expanding the entity descriptions by designing Chain-of-Thought prompt, enhancing the understanding of relation by designing global, local, and reverse prompts, as well as extracting the structural data via keywords summarization and matching.

We evaluated MPIKGC on WN18RR and FB15k237 datasets for the link prediction task and on WN11 and FB13 datasets for the triplet classification task. The results of extensive experiments demonstrated that our method achieved significant improvement over four description-based KGC models. Moreover, additional ablation experiments highlight the potential to combine different enhancement methods for even better performance.

In the future, we plan to explore the possibility of refining the "SameAs" relation into more fine-grained categories, without adding too many triplets. On the other hand, generating KG data by LLMs may encounter problems such as hallucination, toxic and bias, and we plan to develop restricted prompts or fine-tune LLMs to augment text generation controllability and interpretability.

6. Acknowledgement

This work was supported in part by the grants from National Natural Science Foundation of China (No.62222213, U22B2059, U23A20319, 62072423), and the USTC Research Funds of the Double First-Class Initiative (No.YD2150002009). Xiangyu Zhao was partially supported by Research Impact Fund (No.R1015-23), APRC - CityU New Research Initiatives (No.9610565, Start-up Grant for New Faculty of CityU), CityU - HKIDS Early Career Research Grant (No.9360163), Hong Kong ITC Innovation and Technology Fund Midstream Research Programme for Universities Project (No.ITS/034/22MS), Hong Kong Environmental and Conservation Fund (No. 88/2022), and SIRG - CityU Strategic Interdisciplinary Research Grant (No.7020046, No.7020074), and CCF-Tencent Open Fund.

7. Bibliographical References

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2023. LongBench: A Bilingual, Multitask Benchmark for Long Context Understanding. *arXiv preprint arXiv:2308.14508*.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6901–6914. Association for Computational Linguistics.

Chen Chen, Yufei Wang, Aixin Sun, Bing Li, and Kwok-Yan Lam. 2023. Dipping PLMs Sauce: Bridging Structure and Text for Effective Knowledge Graph Completion via Conditional Soft Prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11489–11503. Association for Computational Linguistics.

Touvron Hugo, Martin Louis, Stone Kevin, and others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Pengcheng Jiang, Shivam Agarwal, Bowen Jin, Xuan Wang, Jimeng Sun, and Jiawei Han. 2023. Text-Augmented Open Knowledge Graph Completion via Pre-Trained Language Models. *arXiv preprint arXiv:2305.15597*.

Jinyuan Li, Han Li, Zhuo Pan, and Gang Pan. 2023. Prompt ChatGPT In MNER: Improved multimodal named entity recognition method based on auxiliary refining knowledge from ChatGPT. *arXiv preprint arXiv:2305.12212*.

Yuling Li, Kui Yu, Xiaoling Huang, and Yuhong Zhang. 2022. Learning inter-entity-interaction for few-shot knowledge graph completion. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7691–7700. Association for Computational Linguistics.

Percy Liang, Rishi Bommasani, and Tony et al. Lee. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Vishwas Mruthyunjaya, Pouya Pezeshkpour, Estevam Hruschka, and Nikita Bhutani. 2023. Rethinking Language Models as Symbolic Knowledge Graphs. *arXiv preprint arXiv:2308.13676*.

Tu Dinh Nguyen, Dat Quoc Nguyen, Dinh Phung, et al. 2018. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 327–333.

OpenAI. 2023. GPT-4 Technical Report. *ArXiv*, abs/2303.08774.

Jeff Z Pan, Simon Razniewski, Jan-Christoph Kalo, Sneha Singhania, Jiaoyan Chen, Stefan Dietze, Hajira Jabeen, Janna Omeliyanenko, Wen Zhang, Matteo Lissandrini, et al. 2023a. Large Language Models and Knowledge Graphs: Opportunities and Challenges. *arXiv preprint arXiv:2308.06374*.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2023b. Unifying Large Language Models and Knowledge Graphs: A Roadmap. *arXiv preprint arXiv:2306.08302*.

Wenjun Peng, Guiyang Li, Yue Jiang, Zilong Wang, Dan Ou, Xiaoyi Zeng, Derong Xu, Tong Xu, and Enhong Chen. 2023. Large language model based long-tail query rewriting in taobao search. *arXiv preprint arXiv:2311.03758*.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4498–4507.

Rui Sun, Xuezhi Cao, Yan Zhao, Junchen Wan, Kun Zhou, Fuzheng Zhang, Zhongyuan Wang, and Kai Zheng. 2020. Multi-modal knowledge graphs for recommender systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1405–1414.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *International Conference on Learning Representations*.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022a. SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4281–4294. Association for Computational Linguistics.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Xintao Wang, Qianyu He, Jiaqing Liang, and Yanghua Xiao. 2022b. Language Models as Knowledge Embeddings. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2291–2297. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation Learning of Knowledge Graphs with Entity Descriptions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023a. Large language models for generative information extraction: A survey. *arXiv preprint arXiv:2312.17617*.

Derong Xu, Tong Xu, Shiwei Wu, Jingbo Zhou, and Enhong Chen. 2022. Relation-enhanced Negative Sampling for Multimodal Knowledge Graph Completion. MM '22, page 3857–3866, New York, NY, USA. Association for Computing Machinery.

Derong Xu, Jingbo Zhou, Tong Xu, Yuan Xia, Ji Liu, Enhong Chen, and Dejing Dou. 2023b. Multimodal Biological Knowledge Graph Completion via Triple Co-attention Mechanism. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, pages 3928–3941. IEEE.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. *arXiv preprint arXiv:1412.6575*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for Knowledge Graph Completion. *arXiv preprint arXiv:1909.03193*.

Ling Yue, Yongqi Zhang, Quanming Yao, Yong Li, Xian Wu, Ziheng Zhang, Zhenxi Lin, and Yefeng Zheng. 2023. Relation-aware Ensemble Learning for Knowledge Graph Embedding. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16620–16631, Singapore. Association for Computational Linguistics.

Aohan Zeng, Xiao Liu, and Zhengxiao Du et al. 2023. GLM-130B: An Open Bilingual Pre-trained Model. In *The Eleventh International Conference on Learning Representations*.

Xiangru Zhu, Zhixu Li, Xiaodan Wang, Xueyao Jiang, Penglei Sun, Xuwu Wang, Yanghua Xiao, and Nicholas Jing Yuan. 2022. Multi-modal knowledge graph construction and application: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

8. Language Resource References

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, page 1247–1250. Association for Computing Machinery.

Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. 2018. Convolutional 2D Knowledge Graph Embeddings. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, pages 1811–1818.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. DBpedia–A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web*, 6(2).

Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. 2015. YAGO3: A Knowledge Base from Multilingual Wikipedias. In *CIDR 2015, Seventh Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, January 4-7, 2015, Online Proceedings*.

George A Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. *Advances in neural information processing systems*, 26.

Kristina Toutanova and Danqi Chen. 2015. Observed Versus Latent Features for Knowledge Base and Text Inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 57–66, Beijing, China. Association for Computational Linguistics.

9. Appendices

9.1. Hyper-Parameter Settings

To ensure the reproducibility of our results, we provide detailed information on the hyperparameters used for the four baselines across four benchmarks. This includes information on hyperparameters such as max token length and batch size, as shown in Table 10, 9, 8, and 11.

To ensure a fair evaluation of the performance improvement of our enhanced KG on KG completion compared to the original KG, we maintain the same hyperparameters for them. The parameter settings for the four KGC models mostly follow the defaults in their open-source code.

KG-BERT	FB13	WN11
learning rate	5e-5	5e-5
batch size	32	256
epochs	8	5
max num tokens	70	50
gradient accumulation steps	1	1
warmup proportion	0.1	0.1

Table 8: The hyper-parameter settings of KG-BERT for FB13 and WN11.

SimKGC	FB15k237	WN18RR
learning rate	1e-5	5e-5
batch size	1024	1024
additive margin	0.02	0.02
use amp	True	True
use self-negative	True	True
finetune-t	True	True
pre-batch	2	0
epochs	10	50
max num tokens	70	50

Table 9: The hyper-parameter settings of SimKGC for FB15k237 and WN18RR.

CSProm-KG	FB15k237	WN18RR
learning rate	5e-4	5e-4
batch size	128	128
epochs	60	60
desc max length	70	50
prompt length	10	10
alpha	0.1	0.1
n_lar	8	8
label smoothing	0.1	0.1
embed dim	156	144
k_w	12	12
k_h	13	12
alpha step	0.00001	0.00001

Table 10: The hyper-parameter settings of CSProm-KG for FB15k237 and WN18RR.

LMKE	FB15k237/FB13	WN18RR/WN11
bert_lr	1e-5	1e-5
model_lr	5e-4	5e-4
weight decay	1e-7	1e-7
batch size	256	1024
epochs	8	70
max tokens	70	50
self adversarial	True	True
contrastive	True	True
plm	bert	bert

Table 11: The hyper-parameter settings of LMKE for FB15k237, FB13, WN11, and WN18RR.

9.2. Costs

For the LLM inference stage, we conduct experiments on one NVIDIA V100 32G, and report the average time and config/cost for 100 data samples, as shown in Table 12, 13, and 14. For example, it only costs 11.25 hours of inference time to run MPIKGC-E on FB15k237 with 15k samples, using one V100 32G. For the KGC training stage, our enhancement method requires minimal additional time over the original method, with only an increase of 2G-3G in GPU memory consumption due to the larger maximum token.

It is important to note that the majority of our experiments do not rely on ChatGPT API. Instead, we primarily utilize open-source LLMs such as Llama and ChatGLM (in the main experiments). ChatGPT API is only involved in Section 4.7 for the ablation study of comparing different LLMs.

LLM	config/cost	time
chatglm-2-6b	V100 13G	5.4s
llama-2-7b-chat	V100 14G	6.9s
gpt-3.5-turbo-0613	0.000638$	4.9s
gpt-4-0613	0.0246$	13.0s

Table 12: Generation costs for MPIKGC-E.

LLM	config/cost	time
chatglm-2-6b	V100 13G	3.1s
llama-2-7b-chat	V100 14G	3.3s
gpt-3.5-turbo-0613	0.00021$	2.8s
gpt-4-0613	0.00615$	3.9s

Table 13: Generation costs for MPIKGC-R.

LLM	config/cost	time
chatglm-2-6b	V100 13G	3.4s
llama-2-7b-chat	V100 14G	3.3s
gpt-3.5-turbo-0613	0.00020$	2.7s
gpt-4-0613	0.00318$	5.2s

Table 14: Generation costs for MPIKGC-S.

9.3. Comparison with TagReal

Here, we compare TagReal (Jiang et al., 2023) to demonstrate our differences and advantages for knowledge graph completion. TagReal is a method for open knowledge graph completion that automatically generates query prompts and retrieves support information from text corpora to probe knowledge from pre-trained language models. Theoretically, while TagReal performs well on KGC, we identify several improvements in our method compared to Targal:

- We propose three enhancement strategies that improve performance from different perspectives (entity, relation, structure). Each strategy has its own advantages and, when combined, they further enhance performance.
- Our method exhibits stronger applicability. Our method can enhance most description-based KGC models, adapt to multiple LLMs, and improve the performance of both link prediction and triplet classification tasks.
- Our method is more flexible as it generates supplementary text descriptions using LLMs through instructions. However, the prompt generation of TagReal requires complex steps. We believe that our designing instructions and generating prompts using LLMs would be simpler and more efficient.

As for experimental comparison in Table 15, the method TagReal differs from ours in terms of the benchmarks, backbones, and training methods used. We compared the prompt generated for each relation from TagReal with our method MPIKGC-R in FB15k237. For instance, for the relation "*/sports/sports_team_location/teams*", the enhancement strategies are as follows:

- TagReal –> [X] is located in [Y].
- MPIKGC-R –> The relation denotes the geographical location in which sports teams are located, indicating the connection between teams and their respective locations.

Models	FB15k237			
	MRR	H@1	H@3	H@10
LMKE	30.31	21.49	33.02	48.07
+TagReal	30.48	21.63	33.19	48.37
+MPIKGC-R Global	30.99	22.21	33.70	48.83

Table 15: Comparison with TagReal, in which the enhancement data for MPIKGC-R Global was generated by querying GPT4.

The experimental results show that after supplementing the relation, our method can generate a more detailed explanation for this relation, resulting in better performance, even though TagReal shows performance improvement compared to LMKE.