Contextualization Distillation from Large Language Model for Knowledge Graph Completion

Anonymous ACL submission

Abstract

While textual information significantly enhances the performance of pre-trained language models (PLMs) in knowledge graph completion (KGC), the static and noisy nature of existing corpora collected from Wikipedia articles or synsets definitions often limits the potential of PLM-based KGC models. To surmount these challenges, we introduce the *Contextualization Distillation* strategy, a versatile plug-in-and-play approach compatible with both discriminative and generative KGC frameworks. Our method begins by instructing large language models (LLMs) to transform compact, structural triplets into context-rich segments. Subsequently, we introduce two tailored auxiliary tasks—reconstruction and contextualization—allowing smaller KGC models to assimilate insights from these enriched triplets. Comprehensive evaluations across diverse datasets and KGC techniques highlight the efficacy and adaptability of our approach, revealing consistent performance enhancements irrespective of underlying pipelines or architectures. Moreover, our analysis makes our method more explainable and provides insight into how to generate high-quality corpora for KGC, as well as the selection of suitable distillation tasks.

1 Introduction

Knowledge graph completion (KGC) is a fundamental task in natural language processing (NLP), aiming at unveiling hidden insights within diverse knowledge graphs to explore novel knowledge patterns. Traditional KGC methods (Nickel et al., 2011; Bordes et al., 2013) typically predict the missing part of the triplets by learning the representation of each entity and relation based on their structural information. However, such embedding-based methods tend to overlook the rich textual information of the knowledge graph. Therefore, pre-trained language models (PLMs) have been introduced to KGC and achieved promising results (Kenton and Toutanova, 2019; Xie et al., 2022).

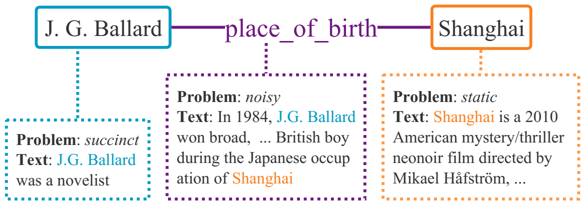

Figure 1: An example to illustrate the limitations of the current textual information for KGC.

Methods	H@1	H@3	H@8/10
ChatGPT-1-shot	15.6	17.6	19.6
PaLM2-1-shot	15.7	20.8	25.4
KG-S2S (Chen et al., 2022a)	**28.5**	**38.8**	**49.3**

Table 1: ChatGPT and PaLM2's unsatisfactory performance on the test set of FB15k-237N compared to a smaller KGC model, KG-S2S (Chen et al., 2022a).

While it has been well-discovered that textual information can be beneficial for PLM-based KGC models (Yao et al., 2019; Wang et al., 2021b; Chen et al., 2022a, 2023a), prior attempts to augment KGC models with textual data from Wikipedia article (Zhong et al., 2015) or synsets definitions (Yao et al., 2019) have encountered certain limitations: (*i*) Entity descriptions, often succinct and static, may inhibit the formation of a comprehensive understanding of entities within KGC models. (*ii*) The incorporation of triplet descriptions, albeit potentially enriching, can introduce substantial noise, particularly when derived through automatic entity alignment (Sun et al., 2020). Figure 1 demonstrates an example to illustrate the aforementioned limitations. The description for the head "*J. G. Ballard*" is limited and for the tail "*Shanghai*", it mistakenly uses the definition of the movie also named "Shanghai". Also, while the two entities show up in the triplet description, it falls short in conveying the semantic essence of the relation "*place_of_birth*".

In light of these limitations, our attention shifts to Large Language Models (LLMs) (Brown et al., 2020; Zhang et al., 2022; Anil et al., 2023; Touvron et al., 2023), renowned for their capability in generating articulate and high-quality data (Dai et al., 2023; Shridhar et al., 2023; Zheng et al., 2023). Our exploration commences with a scrupulous evaluation of LLMs, such as ChatGPT and PaLM2, in KGC, benchmarking them across several esteemed KGC datasets (Dettmers et al., 2018; Garcia-Duran et al., 2018; Mahdisoltani et al., 2013). Utilizing 1-shot In-Context Learning (ICL), we deduce missing heads or tails in triplets and report evaluation metrics. It reveals a significant performance discrepancy of two LLMs in comparison to KG-S2S (Chen et al., 2022a) despite its reliance on a smaller foundational model, T5-base (Raffel et al., 2020). This insight propels us toward the conclusion that direct utilization of LLMs for KGC tasks, while intuitive, is outperformed by the fine-tuning of more diminutive, specialized KGC models. This observation aligns with findings from (Liang et al., 2022; Sun et al., 2023; Zhao et al., 2023), which highlighted the limitations of LLMs in knowledge-centric tasks. Experiment results and analysis on more KGC datasets can be found in Appendix A.

To optimally harness LLMs for KGC, we draw inspiration from recent works (Xiang et al., 2022; Kim et al., 2022a) and introduce a novel approach, *Contextualization Distillation*. Contextualization Distillation first extracts descriptive contexts from LLMs with well-designed prompts, thereby securing dynamic, high-quality context for each entity and triplet. Subsequent to this, two auxiliary tasks are proposed to train smaller KGC models with these informative, descriptive contexts. The plug-in-and-play characteristic of our contextualization distillation enables us to apply and evaluate it on various KGC datasets and baseline models. Through extensive experiments, we affirm that Contextualization Distillation consistently enhances the performance of smaller KGC models, irrespective of architectural and pipeline disparities. Additionally, we provide an exhaustive analysis of each step of Contextualization Distillation, encouraging further insights and elucidations.

The contributions of this work can be summarized into three main aspects:

- We identify the constraints of the current corpus for PLMs-based KGC models and introduce a plug-in-and-play approach, Contextualization Distillation, to enhance smaller KGC models with extracted rationale from LLMs.
- We conduct extensive experiments across several widely recognized KGC datasets and utilize various baseline models. Through these experiments, we validate the effectiveness of Contextualization Distillation in consistently improving smaller KGC models.
- We delve into a comprehensive analysis of our proposed method and provide valuable insights and guidance on how to generate high-quality corpora for distillation, as well as the selection of suitable distillation tasks.

2 Related Work

2.1 Knowledge Graph Completion

Traditional KGC methods (Nickel et al., 2011; Bordes et al., 2013) involve embedding entities and relations into a representation space. In pursuit of a more accurate depiction of entity-relation pairs, different representation spaces (Trouillon et al., 2016; Xiao et al., 2016) have been proposed considering various factors, e.g., differentiability and calculation possibility (Ji et al., 2021). During training, two primary objectives emerge to assign higher scores to true triplets than negative ones: 1) Translational distance methods gauge the plausibility of a fact by measuring the distance between the two entities under certain relations (Lin et al., 2015; Wang et al., 2014); 2) Semantic matching methods compute the latent semantics of entities and relations (Yang et al., 2015; Dettmers et al., 2018).

To better utilize the rich textual information of knowledge graphs, PLMs have been introduced in KGC. Yao et al. (2019) first propose to use BERT (Kenton and Toutanova, 2019) to encode the entity and relation's name and adopt a binary classifier to predict the validity of given triplets. Following them, Wang et al. (2021a) leverage the Siamese network to encode the head-relation pair and tail in a triplet separately, aiming to reduce the time cost and make the inference scalable. Lv et al. (2022) convert each triple and its textual information into natural prompt sentences to fully inspire PLMs' potential in the KGC task. Chen et al. (2023a) design a conditional soft prompts framework to maintain a balance between structural information and textual knowledge in KGC. Recently, there are also some works trying to leverage generative PLMs to perform KGC in a sequence-to-sequence manner and achieve promising results (Xie et al., 2022; Saxena et al., 2022; Chen et al., 2022a).

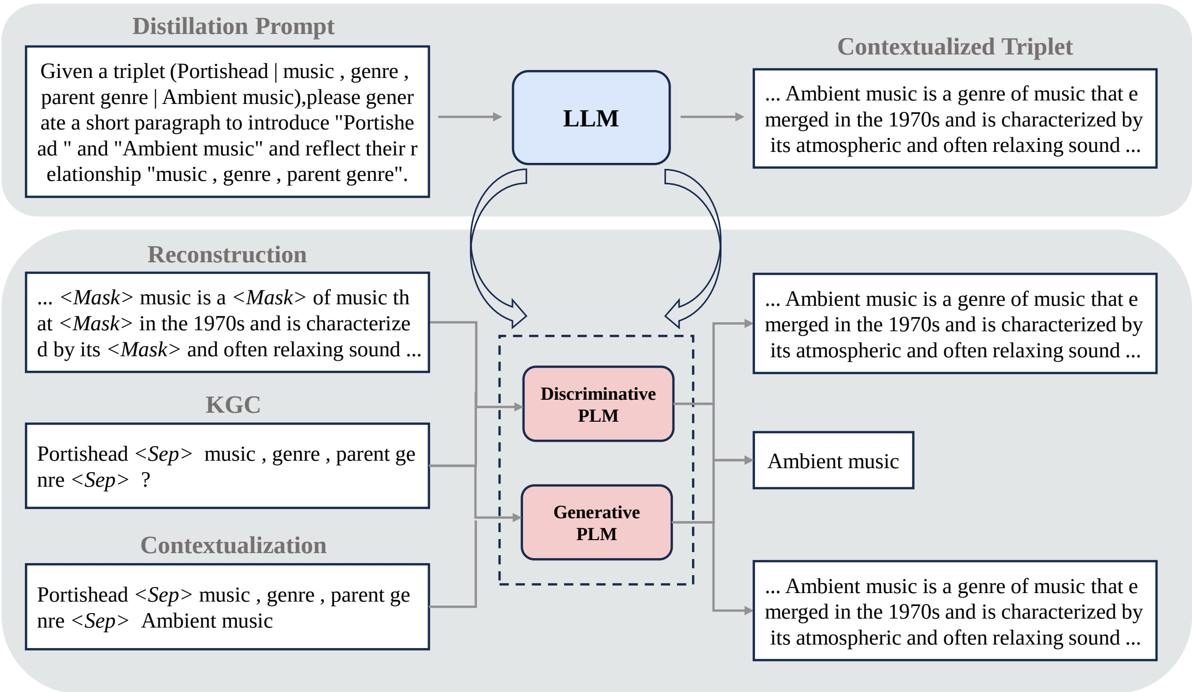

Figure 2: An overview pipeline of our Contextualization Distillation. We first extract descriptive contexts from LLMs (Section 3.1). Then, two auxiliary tasks, reconstruction (Section 3.2.1) and contextualization (Section 3.2.2) are designed to train the smaller KGC models with the contextualized information.

2.2 Distillation from LLMs

Knowledge distillation has proven to be an effective approach for transferring expertise from larger, highly competent teacher models to smaller, affordable student models (Buciluă et al., 2006; Hinton et al., 2015; Beyer et al., 2022). With the emergence of LLMs, a substantial body of research has concentrated on distilling valuable insights from these LLMs to enhance the capabilities of smaller PLMs. One of the most common methods is to prompt LLMs to explain their predictions and then use such rationales to distill their reasoning abilities into smaller models (Wang et al., 2022; Ho et al., 2023; Magister et al., 2022; Hsieh et al., 2023; Shridhar et al., 2023). Distilling conversations from LLMs is another cost-effective method to build new dialogue datasets (Kim et al., 2022b; Chen et al., 2023b; Kim et al., 2022a) or augment existing ones (Chen et al., 2022b; Zhou et al., 2022; Zheng et al., 2023). There are also some attempts (Marjieh et al., 2023; Zhang et al., 2023) that focus on distilling domain-specific knowledge from LLMs for various downstream applications.

Several recent studies have validated the contextualization capability of LLMs to convert structural data into raw text. Among them, Xiang et al. (2022) convert triplets in the data-to-text generation dataset into their corresponding descriptions to facilitate disambiguation. Kim et al. (2022a) design a pipeline for synthesizing a dialogue dataset by distilling conversations from LLMs, enhanced with a social commonsense knowledge graph. By contrast, we are the first to leverage descriptive context generated by LLMs as an informative auxiliary corpus to the KGC models.

3 Contextualization Distillation

In this section, we first illustrate how we curate prompts to extract the descriptive context of each triplet from the LLM. Subsequently, we design a multi-task framework, together with two auxiliary tasks—reconstruction and contextualization—to train smaller KGC models with these high-quality context corpus. The overview pipeline of our method is illustrated in Figure 2.

3.1 Extract Descriptive Context from LLMs

Recent studies have highlighted the remarkable ability of LLMs to contextualize structural data and transform it into context-rich segments (Xiang et al., 2022; Kim et al., 2022a). Here we borrow their insights and extract descriptive context from LLMs to address the limitations of the existing KGC corpus we mentioned in Section 1.

Input

Given a triplet (Portishead | music , genre , parent genre | Ambient music), please generate a short paragraph to introduce "Portishead " and "Ambient music" and reflect their relationship "music , genre , parent genre".

Output

Portishead is a British trip hop band formed in Bristol in 1991. They are considered one of the pioneers of the genre, along with Massive Attack and Tricky. Ambient music is a genre of music that emerged in the 1970s and is characterized by its atmospheric and often relaxing sound. Portishead's music is often described as ambient, due to its use of loops, drones, and other sound effects.

Figure 3: An example contains our instruction to LLMs and the generated descriptive context. We use green to highlight entity description prompt/ generation result and blue to highlight triplet description prompt/ generation result.

In particular, we focus on two commonly employed types of descriptions prevalent in prior methodologies: entity description (ED) (Yao et al., 2019; Chen et al., 2022a) and triplet description (TD) (Sun et al., 2020). Entity description refers to the definition and description of individual entities, while triplet description refers to a textual segment that reflects the specific relationship between two entities within a triplet. Given triplets of a knowledge graph $t_i \in T$, we first curate prompt p_i for the ith triplet by filling the pre-defined template:

$$p_i = \text{Template}(h_i, r_i, t_i), \tag{1}$$

where h_i, r_i, t_i are the head entity, relation, and tail entity of the ith triplet. Then, we use p_i as the input to prompt the LLM to generate the descriptive context c_i for each triplet:

$$c_i = \text{LLM}(p_i), \tag{2}$$

As Figure 3 shows, in our Contextualization Distillation, we design the template to generate both entity description and triplet description at one time. The generating path of each descriptive context can be expressed as $T \longrightarrow (ED, TD)$. Without loss of generalization, we conduct an ablation study to adopt different generating paths of auxiliary context in Section 4.3.

3.2 Multi-task Learning with Descriptive Context

Different PLM-based KGC models adopt diverse loss functions and pipeline architectures (Yao et al., 2019; Chen et al., 2022a; Xie et al., 2022; Chen et al., 2023a). **To ensure the compatibility of our Contextualization Distillation to be applied in various PLM-based KGC methods**, we design a multi-task learning framework for these models to learn from both the KGC task and auxiliary descriptive context-based tasks. For the auxiliary tasks, we design *reconstruction* (Section 3.2.1) and *contextualizatioin* (Section 3.2.2) for discriminative and generative KGC models respectively.

3.2.1 Reconstruction

The reconstruction task aims to train the model to restore the corrupted descriptive contexts. For the discriminative KGC models, we follow the implementation of Kenton and Toutanova (2019) and use masked language modeling (MLM). Previous studies have validated that **such auxiliary self-supervised tasks in the domain-specific corpus can benefit downstream applications** (Han et al., 2021; Wang et al., 2021b).

To be specific, MLM randomly identifies 15% of the tokens within the descriptive context. Among these tokens, 80% are tactically concealed with the special token "$< Mask >$", 10% are seamlessly substituted with random tokens, while the remaining 10% keep unchanged. For each selected token, the objective of MLM is to restore the original content at that particular position, achieved through the cross-entropy loss. The aforementioned process can be formally expressed as follows:

$$c_i^{'} = \text{MLM}(c_i), \tag{3}$$

$$\mathcal{L}_{rec} = \frac{1}{N}\sum_{i=1}^{N} \ell(f(c_i^{'}), c_i) \tag{4}$$

The final loss of discriminative KGC models is the combination of the KGC loss[1] and the proposed reconstruction loss:

$$\mathcal{L}_{dis} = \mathcal{L}_{kgc} + \alpha \cdot \mathcal{L}_{rec}, \tag{5}$$

where α is a hyper-parameter to control the ratios between the two losses.

3.2.2 Contextualization

The objective of contextualization is to instruct the model in generating the descriptive context c_i when provided with the original triplet $t_i = h, r, t$. Compared with reconstruction, **contextualization demands a more nuanced and intricate ability from PLM**. It necessitates the PLM to precisely grasp the meaning of both entities involved and the inherent relationship that binds them together, to generate fluent and accurate descriptions.

[1]We give the illustration of the discriminative KGC models we used in Appendix B.1

Specifically, we concatenate head, relation and tail with a special token "$< Sep >$" as input:

$$I_i = \mathrm{Con}(h_i, < Sep >, r_i, < Sep >, t_i) \quad (6)$$

Then, we input them into the generative PLM and train the model to generate descriptive context c_i using the cross-entropy loss:

$$\mathcal{L}_{con} = \frac{1}{N}\sum_{i=1}^{N} \ell(f(I_i), c_i) \quad (7)$$

The final loss of generative KGC models is the combination of the KGC loss[2] and the proposed contextualization loss:

$$\mathcal{L}_{gen} = \mathcal{L}_{kgc} + \alpha \cdot \mathcal{L}_{con} \quad (8)$$

For generative KGC models, it is also applicable to apply reconstruction as the auxiliary task. We have done an ablation study in Section 4.4 to examine the effectiveness of each auxiliary task on generative KGC models.

4 Experiment

In this section, we apply our Contextualization Distillation across a range of PLM-based KGC baselines. We compare our enhanced model with our approach against the vanilla models using several KGC datasets. Additionally, we do further analysis of each component in our contextualized distillation and make our method more explainable by conducting case studies.

4.1 Experimental Settings

Datasets We use WN18RR (Dettmers et al., 2018) and FB15k-237N (Lv et al., 2022) in our experiment. WN18RR serves as an enhanced version of its respective counterparts, WN18 (Bordes et al., 2013). The improvements involve the removal of all inverse relations to prevent potential data leakage. For FB15K-237N, it's a refine version of FB15k (Bordes et al., 2013), by eliminating concatenated relations stemming from Freebase mediator nodes (Akrami et al., 2020) to avoid Cartesian production relation issues.

Baselines we adopt several PLM-based KGC models as baselines and apply the proposed Contextualization Distillation to them. **KG-BERT** (Yao et al., 2019) is the first to suggest utilizing PLMs for the KGC task. we also consider **CSProm-KG** (Chen et al., 2023a), which combines PLMs with traditional Knowledge Graph Embedding (KGE) models, achieving a balance between efficiency and performance in KGC. In addition to these discriminative models, we also harness generative KGC models. **GenKGC** (Xie et al., 2022) is the first to accomplish KGC in a sequence-to-sequence manner, with a fine-tuned BART (Lewis et al., 2020) as its backbone. Following them, **KG-S2S** (Chen et al., 2022a) adopt soft prompt tuning and lead to a new SOTA performance among the generative KGC models.

Implementation details All our experiments are conducted on a single GPU (RTX A6000), with CUDA version 11.1. We use PaLM2-540B(Anil et al., 2023) as the large language model to distill descriptive context. We tune the Contextualization Distillation hyper-parameter $\alpha \in \{0.1, 0.5, 1.0\}$. We follow the hyper-parameter settings in the original papers to reproduce each baseline's result. For all datasets, we follow the previous works (Chen et al., 2022a, 2023a) and report Mean Reciprocal Rank (MRR), Hits@1, Hits@3 and Hits@10. More details about our experiment implementation and dataset statistics are shown in Appendix C.

4.2 Main Result

Table 2 displays the results of our experiments on WN18RR and FB15k-237N. We observe that our Contextualization Distillation consistently enhances the performance of all baseline methods, regardless of whether they are based on generative or discriminative models. This unwavering improvement demonstrates **the robust generalization and compatibility of our approach across various PLMs-based KGC methods**.

Additionally, some baselines we choose to implement our Contextualization Distillation also utilize context information. For example, both KG-BERT and CSProm-KG adopt entity descriptions to enhance entity embedding representation. Nevertheless, our approach manages to deliver additional improvements to these context-based baselines. Among them, it is worth noting that the application of our approach to KG-BERT achieves an overall 31.7% enhancement in MRR. All these findings lead us to the conclusion that **Contextualization Distillation is not only compatible with context-based KGC models but also capable of further enhancing their performance**.

[2] We give the illustration of the generative KGC models we used in Appendix B.2

	WN18RR				FB15k-237N			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
Traditional Methods								
TransE* (Bordes et al., 2013)	24.3	4.3	44.1	53.2	25.5	15.2	30.1	45.9
DisMult* (Yang et al., 2015)	44.4	41.2	47.0	50.4	20.9	14.3	23.4	33.0
ComplEx* (Trouillon et al., 2016)	44.9	40.9	46.9	53.0	24.9	18.0	27.6	38.0
ConvE* (Dettmers et al., 2018)	45.6	41.9	47.0	53.1	27.3	19.2	30.5	42.9
RotatE* (Sun et al., 2018)	47.6	42.8	49.2	57.1	27.9	17.7	32.0	48.1
CompGCN* (Vashishth et al., 2019)	47.9	44.3	49.4	54.6	31.6	23.1	34.9	48.0
PLMs-based Methods								
MTL-KGC* (Kim et al., 2020)	33.1	20.3	38.3	59.7	24.1	16.0	28.4	43.0
StAR* (Wang et al., 2021a)	40.1	24.3	49.1	**70.9**	-	-	-	-
PKGC* (Lv et al., 2022)	-	-	-	-	30.7	23.2	32.8	47.1
KGT5* (Saxena et al., 2022)	50.8	48.7	-	54.4	-	-	-	-
Our Implementation								
KG-BERT (Yao et al., 2019)	21.6	4.1	30.2	52.4	20.3	13.9	20.1	40.3
KG-BERT-CD	30.3	16.5	35.4	60.2	25.0	17.2	26.6	45.5
GenKGC (Xie et al., 2022)	-	28.6	44.4	52.4	-	18.7	27.3	33.7
GenKGC-CD	-	29.3	45.6	53.3	-	20.4	29.3	34.9
KG-S2S (Chen et al., 2022a)	57.0	52.5	59.7	65.4	35.4	28.5	38.8	49.3
KG-S2S-CD	**57.6**	**52.6**	**60.7**	67.2	35.9	**28.9**	39.4	50.2
CSProm-KG (Chen et al., 2023a)	55.2	50.0	57.2	65.7	36.0	28.1	39.5	51.1
CSProm-KG-CD	55.9	50.8	57.8	66.0	**37.2**	28.8	**41.0**	**53.0**

Table 2: Experiment results on WN18RR and FB15k-237. * denotes results we take from Chen et al. (2022a). Methods suffixed with "-CD" indicate the baseline models with our Contextualization Distillation applied. The best results of each metric are in bold.

Paths	FN15k-237N		
	H@1	H@3	H@10
-	18.7	27.3	33.7
$T \longrightarrow ED$	20.0	28.9	34.5
$T \longrightarrow TD$	20.1	29.0	34.6
$T \longrightarrow RA$	19.4	28.2	34.2
$T \longrightarrow ED \longrightarrow TD$	19.8	28.6	34.5
$T \longrightarrow (ED, TD)$	**20.4**	**29.3**	**34.9**

Table 3: Ablation study results GenKGC with different generating paths to distill corpus from LLMs. We conduct the experiment using FB15k-237N. We add the vallina GenKGC in the first row for comparison.

4.3 Ablation Study on Generating Path

We investigate the efficacy of different context types in the distillation process by employing various generative paths. As illustrated in Table 3, we initially explore the impact of entity description and triplet description when utilized separately as auxiliary corpora (denoted as $T \longrightarrow ED$ and $T \longrightarrow TD$). The experimental findings underscore the critical roles played by both entity description and triplet description as distillation corpora, leading to noticeable enhancements in the performance of smaller KGC models. Furthermore, we ascertain that our method's generating path $T \longrightarrow (ED, TD)$, which utilizes these two corpora, achieves more improvements by endowing the models with a more comprehensive and richer source of information.

To gain a comprehensive understanding of the effectiveness of our Contextualization Distillation, we also explored other alternative generative paths. While rationale distillation has demonstrated its potential in various NLP tasks (Hsieh et al., 2023; Shridhar et al., 2023), our investigation delves into the $T \longrightarrow RA$ path, wherein we instruct the LLM to generate rationales for each training sample[3]. Although the model utilizing rationale distillation exhibits improved performance compared to the vanilla one, it falls short when compared with our Contextualization Distillation incorporating entity descriptions and triplet descriptions. One plausible explanation for this disparity lies in the intrinsic nature of rationales, which tend to be intricate and structurally complex. This complexity can pose a greater challenge for smaller models to fully comprehend, in contrast to the more straightforward descriptive text utilized in our approach.

Also, we borrow the insight from Chain-of-CoT (CoT) (Wei et al., 2022) that generates the content step by step, and conducts the experiment of the generation process $T \longrightarrow ED \longrightarrow TD$. Specifically, we initially prompt the LLM to generate descriptions for two entities and subsequently append these entity descriptions to the prompt, instructing

[3]We give further details and examples of our prompt in Appendix E

	FN15k-237N			
	MRR	H@1	H@3	H@10
GenKGC	-	18.7	27.3	33.7
w/ Reconstruction	-	19.4	28.2	34.2
w/ Contextualization	-	**20.4**	**29.3**	**34.9**
KG-S2S	35.4	28.5	38.8	49.3
w/ Reconstruction	35.8	**29.3**	38.9	48.9
w/ Contextualization	**35.9**	28.9	**39.4**	**50.2**

Table 4: Ablation study results on GenKGC and KG-S2S with reconstruction and contextualization as the auxiliary task respectively. We conduct the experiment using FB15k-237N.

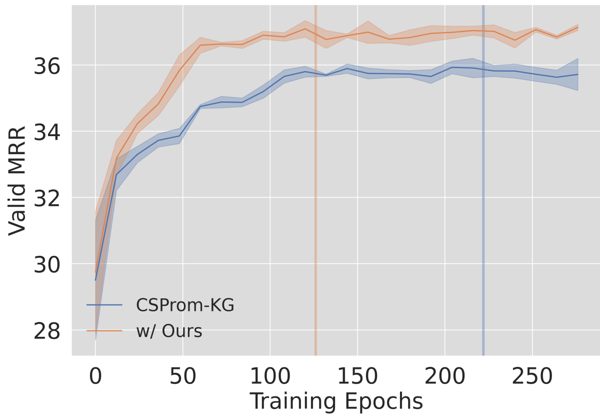

Figure 4: MRR scores on the validation set during the CSProm-KG training on FB15k-237N. We use thin bars to mark the epochs in which the models achieve the best performance in the validation set.

the LLM to generate the triplet description. During training, we concatenate the entity description and triplet description to form the auxiliary corpus for smaller KGC models. Interestingly, our findings indicate that this multi-step generative path also yields suboptimal performance when compared to the single-step generative path. This discrepancy can be attributed to the text incoherence resulting from the concatenation of three segments of descriptions. In light of the insights gained from these observations, we summarize our distillation guidance for KGC as follows: **smaller models can benefit more from comprehensive, descriptive and coherent content generated by LLMs**.

4.4 Ablation Study on Generative KGC Models

In this section, we compare the effectiveness of reconstruction and contextualization in generative KGC models. For GenKGC and KG-S2S, we employ the pre-trained tasks of their respective backbone models (BART for GenKGC and T5 for KG-S2S) as the reconstruction objective. More details of our reconstruction implementation for generative KGC models can be found in Appendix D.

Table 4 presents the ablation study results on FB15k-237N. We find reconstruction is also effective in improving the performance of generative KGC models, showing that KGC models can consistently benefit from the descriptive context with different auxiliary tasks. Comparing the two auxiliary tasks, models with contextualization outperform those with reconstruction on almost every metric, except for Hits@1 in KG-S2S. This implies that **contextualization is a critical capability for generative KGC models to master for better KGC performance**. Generative models have benefited more from the training of converting structural triplets into descriptive context than simply restoring the corrupted corpus.

4.5 Efficiency Analysis

The additional training cost brought by the auxiliary distillation tasks may pose a potential constraint on our approach. However, we also notice baseline models with our method coverage faster on the validation set. Figure 4 presents the validation MRR vs epoch numbers during the CSProm-KG training on FB15k-237N. It is obvious that CSProm-KG with Contextualization Distillation achieves a faster convergence and attains the best checkpoint earlier (at around 125 epochs) compared to the variant without our method (at around 220 epochs). This implies **auxiliary distillation loss can also expedite model learning in KGC**. This trade-off between batch processing time and training steps ultimately results in a training efficiency comparable to that of the vanilla models.

4.6 Case Study

To demonstrate the advantage of our Contextualization Distillation more directable, we conduct a comparative analysis between the description corpus collected by Zhong et al. (2015) and those generated using our method. As presented in Table 5, entity descriptions generated by the LLM effectively address the limitations issue and static shortcomings, resulting in more informative and accurate content. Regarding the triplet description, although the "semi-autobiographical" used in Zhong et al. (2015) somewhat implies J.G. Ballard's connection to Shanghai during his childhood, it still fails to express the semantics of "*place_of_birth*" clearly. In contrast, the descriptive context generated by our method provides a more elaborate and coherent

	Wikipedia (Zhong et al., 2015)	Ours
Head	Ballard was a novelist.	J.G. Ballard (1930-2009) was an English writer. He was born in Shanghai, China, and his early experiences there shaped his writing. His novels often explored themes of alienation, technology, and the future...
Tail	Shanghai is a 2010 American mystery/thriller neo-noir film directed by Mikael Håfström, starring John Cusack and Gong Li...	Shanghai is a city in China. It is one of the most populous cities in the world, and it is a major center of commerce and culture. Shanghai has a long history, and it has been home to many different cultures over the centuries...
Triplet	In 1984, J.G. Ballard won broad, critical recognition for the war novel Empire of the Sun, a semi-autobiographical story of the experiences of a British boy during the Japanese occupation of Shanghai.	Ballard was born in Shanghai in 1930. He lived there until he was eight years old, when his family moved to England. Ballard's early experiences in Shanghai had a profound impact on his writing...

Table 5: Descriptive context of the triplet *(J.G. Ballard, place_of_birth, Shanghai)*. The text in green represents positive content and the text in red represents negative content.

Query	*(The Devil's Double, genre, ?)*
Ground Truth	*Biographical film*
Baseline	*War film*
Ours	*Biographical film*
Our Context	The Devil's Double is a **biographical film** that tells the story of Latif Yahia, **a young Iraqi man who was forced to impersonate Saddam Hussein's son Uday Hussein**...

Table 6: Case study on FB15K-237N with KG-S2S. we also let the model generate a descriptive context for each test sample. The text in **bold** represents informative content in the generated descriptive context.

contextualization of the "*place_of_birth*" between "*J.G. Ballard*" and "*Shanghai*". These comparisons highlight the effectiveness of our method in addressing the previous corpus' limitation.

Furthermore, We showcase how the auxiliary training with descriptive context enhances the baseline models. Table 6 presents the results of KG-S2S performance in a test sample of FB15k-237N, both with and without our contextualization distillation. In this case, the vanilla KG-S2S wrongly predicts the genre of the film "*The Devil's Double*" as "'*War film*', whereas the KG-S2S trained with our auxiliary task correctly labels it as "*Biographical film*". Also, by making the model contextualize each triplet, we find the model with our method applied successfully captures many details about the movie, such as the genre and plot, and presents this information as fluent text. In summary, **the model not only acquires valuable insights about the triplets but also gains the ability to adeptly contextualize this information through our Contextualization Distillation**.

5 Conclusion

In this work, we propose Contextualization Distillation, addressing the limitation of the existing KGC textual data by prompting LLMs to generate descriptive context. To ensure the versatility of our approach across various PLM-based KGC models, we have designed a multi-task learning framework. Within this framework, we incorporate two auxiliary tasks, reconstruction and contextualization, which aid in training smaller KGC models in the informative descriptive context. We conduct experiments on several mainstream KGC benchmarks and the results show that our Contextualization Distillation consistently enhances the baseline model's performance. Furthermore, we conduct in-depth analyses to make the effect of our method more explainable, providing guidance on how to effectively leverage LLMs to improve KGC as well. In the future, we plan to adapt our method to other knowledge-driven tasks, such as entity linking and knowledge graph question answering.

6 Limitation

One limitation of our approach is that the descriptive context extraction stage is only tested with the PaLM2 model due to its unlimited API. The behavior of other LLMs of varying sizes in generating auxiliary corpora for KGC remains unexplored. Due to limitations in computing resources, we evaluate our method on two RE datasets, while disregarding scenarios such as temporal knowledge graph completion (Garcia-Duran et al., 2018), few-shot knowledge graph completion (Xiong et al., 2018) and commonsense knowledge graph completion (Li et al., 2022). In future research, we plan to investigate the effectiveness of our method in border scenarios.

References

Farahnaz Akrami, Mohammed Samiul Saeef, Qingheng Zhang, Wei Hu, and Chengkai Li. 2020. Realistic re-evaluation of knowledge graph completion methods: An experimental study. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, pages 1995–2010.

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. 2022. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10925–10934.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541.

Chen Chen, Yufei Wang, Bing Li, and Kwok-Yan Lam. 2022a. Knowledge is flat: A seq2seq generative framework for various knowledge graph completion. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4005–4017.

Chen Chen, Yufei Wang, Aixin Sun, Bing Li, and Kwok-Yan Lam. 2023a. Dipping plms sauce: Bridging structure and text for effective knowledge graph completion via conditional soft prompting. *arXiv preprint arXiv:2307.01709*.

Maximillian Chen, Alexandros Papangelis, Chenyang Tao, Seokhwan Kim, Andy Rosenbaum, Yang Liu, Zhou Yu, and Dilek Hakkani-Tur. 2023b. Places: Prompting language models for social conversation synthesis. *arXiv preprint arXiv:2302.03269*.

Maximillian Chen, Alexandros Papangelis, Chenyang Tao, Andy Rosenbaum, Seokhwan Kim, Yang Liu, Zhou Yu, and Dilek Hakkani-Tur. 2022b. Weakly supervised data augmentation through prompting for dialogue understanding. In *NeurIPS 2022 Workshop on Synthetic Data for Empowering ML Research*.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Zihao Wu, Lin Zhao, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, et al. 2023. Chataug: Leveraging chatgpt for text data augmentation. *arXiv preprint arXiv:2302.13007*.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Alberto Garcia-Duran, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4821.

Janghoon Han, Taesuk Hong, Byoungjae Kim, Youngjoong Ko, and Jungyun Seo. 2021. Fine-grained post-training for improving retrieval-based dialogue systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1549–1558.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14852–14882, Toronto, Canada. Association for Computational Linguistics.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8003–8017, Toronto, Canada. Association for Computational Linguistics.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and S Yu Philip. 2021. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE transactions on neural networks and learning systems*, 33(2):494–514.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2.

Bosung Kim, Taesuk Hong, Youngjoong Ko, and Jungyun Seo. 2020. Multi-task learning for knowledge graph completion with pre-trained language models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1737–1743.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, et al. 2022a. Soda: Million-scale dialogue distillation with social commonsense contextualization. *arXiv preprint arXiv:2212.10465*.

Hyunwoo Kim, Youngjae Yu, Liwei Jiang, Ximing Lu, Daniel Khashabi, Gunhee Kim, Yejin Choi, and Maarten Sap. 2022b. Prosocialdialog: A prosocial backbone for conversational agents. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4005–4029.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Dawei Li, Yanran Li, Jiayi Zhang, Ke Li, Chen Wei, Jianwei Cui, and Bin Wang. 2022. C3kg: A chinese commonsense conversation knowledge graph. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1369–1383.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Xin Lv, Yankai Lin, Yixin Cao, Lei Hou, Juanzi Li, Zhiyuan Liu, Peng Li, and Jie Zhou. 2022. Do pretrained models benefit knowledge graph completion? a reliable evaluation and a reasonable approach. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3570–3581.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian M Suchanek. 2013. Yago3: A knowledge base from multilingual wikipedias. In *CIDR*.

Raja Marjieh, Ilia Sucholutsky, Pol van Rijn, Nori Jacoby, and Thomas L Griffiths. 2023. What language reveals about perception: Distilling psychophysical knowledge from large language models. *arXiv preprint arXiv:2302.01308*.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 809–816.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992.

Apoorv Saxena, Adrian Kochsiek, and Rainer Gemulla. 2022. Sequence-to-sequence knowledge graph completion and question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2814–2828.

Kumar Shridhar, Alessandro Stolfo, and Mrinmaya Sachan. 2023. Distilling reasoning capabilities into smaller language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7059–7073, Toronto, Canada. Association for Computational Linguistics.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2023. Head-to-tail: How knowledgeable are large language models (llm)? aka will llms replace knowledge graphs? *arXiv preprint arXiv:2308.10168*.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuan-Jing Huang, and Zheng Zhang.

2020. Colake: Contextualized language and knowledge embedding. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3660–3670.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2018. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

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

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*, pages 2071–2080. PMLR.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based multi-relational graph convolutional networks. In *International Conference on Learning Representations*.

Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. 2021a. Structure-augmented text representation learning for efficient knowledge graph completion. In *Proceedings of the Web Conference 2021*, pages 1737–1748.

PeiFeng Wang, Aaron Chan, Filip Ilievski, Muhao Chen, and Xiang Ren. 2022. Pinto: Faithful language reasoning using prompt-generated rationales. In *The Eleventh International Conference on Learning Representations*.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021b. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*, volume 28.

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.

Jiannan Xiang, Zhengzhong Liu, Yucheng Zhou, Eric Xing, and Zhiting Hu. 2022. Asdot: Any-shot data-to-text generation with pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1886–1899.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016. Transg: A generative model for knowledge graph embedding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2316–2325.

Xin Xie, Ningyu Zhang, Zhoubo Li, Shumin Deng, Hui Chen, Feiyu Xiong, Mosha Chen, and Huajun Chen. 2022. From discrimination to generation: Knowledge graph completion with generative transformer. In *Companion Proceedings of the Web Conference 2022*, pages 162–165.

Wenhan Xiong, Mo Yu, Shiyu Chang, Xiaoxiao Guo, and William Yang Wang. 2018. One-shot relational learning for knowledge graphs. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1980–1990.

Bishan Yang, Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the International Conference on Learning Representations (ICLR) 2015*.

Shiping Yang, Renliang Sun, and Xiaojun Wan. 2023. A new benchmark and reverse validation method for passage-level hallucination detection. *arXiv preprint arXiv:2310.06498*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193*.

Hongbo Zhang, Junying Chen, Feng Jiang, Fei Yu, Zhihong Chen, Jianquan Li, Guiming Chen, Xiangbo Wu, Zhiyi Zhang, Qingying Xiao, et al. 2023. Huatuogpt, towards taming language model to be a doctor. *arXiv preprint arXiv:2305.15075*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Chujie Zheng, Sahand Sabour, Jiaxin Wen, Zheng Zhang, and Minlie Huang. 2023. AugESC: Dialogue augmentation with large language models for emotional support conversation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1552–1568, Toronto, Canada. Association for Computational Linguistics.

Huaping Zhong, Jianwen Zhang, Zhen Wang, Hai Wan, and Zheng Chen. 2015. Aligning knowledge and text embeddings by entity descriptions. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 267–272.

Pei Zhou, Hyundong Cho, Pegah Jandaghi, Dong-Ho Lee, Bill Yuchen Lin, Jay Pujara, and Xiang Ren. 2022. Reflect, not reflex: Inference-based common ground improves dialogue response quality. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10450–10468.

Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. *arXiv preprint arXiv:2305.13168*.

A Large Language Model Performance on KGC

We follow Zhu et al. (2023) to assess the performance of directly instructing LLMs to perform KGC and Table 7 gives an example of our input to LLMs. For PaLM, we utilize the API parameter “candidate_count”, while for ChatGPT, we use “n” to obtain multiple candidates, enabling the calculation of Hit@1, Hit@3, and Hit@10 metrics. After obtaining the model’s outputs, we use the Sentence-BERT (Reimers and Gurevych, 2019) to guarantee each output result matches a corresponding entity in the dataset’s entity set.

Table 8 displays the additional experimental results for ChatGPT and PaLM2 across several KGC datasets. It is evident that the performance of ICL of LLM falls short of KG-S2S’s in every dataset. One potential explanation for this subpar performance can be attributed to the phenomenon of hallucination in LLMs (Ji et al., 2023; Yang et al., 2023), leading to incorrect responses when the LLM encounters unfamiliar content.

We also conducted an analysis of the influence of the number of demonstration samples. As Table 9 shows, we find while the number of demonstrations increases, the performance of LLMs shows a corresponding improvement. It appears that augmenting the number of demonstrations in the prompt could be a potential strategy for enhancing the capabilities of LLMs in KGC. Nonetheless, it’s essential to note that incorporating an excessive number of relevant samples as demonstrations faces practical challenges, primarily due to constraints related to input length and efficiency considerations.

Triplet	*(Stan Collymore, play_for, England national football team)*
Tail Prompt	Predict the tail entity [MASK] from the given (Keko (footballer, born 1973), plays for, [MASK]) by completing the sentence "what is the plays for of Keko (footballer, born 1973)? The answer is ". The answer is UE Figueres, so the [MASK] is UE Figueres. Predict the tail entity [MASK] from the given (Stan Collymore, plays for, [MASK]) by completing the sentence "what is the plays for of Stan Collymore? The answer is ". The answer is
Head Prompt	Predict the head entity [MASK] from the given ([MASK], plays for, UE Figueres) by completing the sentence "UE Figueres is the plays for of what? The answer is ". The answer is Keko (footballer, born 1973), so the [MASK] is Keko (footballer, born 1973). Predict the head entity [MASK] from the given ([MASK], plays for, England national football team) by completing the sentence "England national football team is the plays for of what? The answer is ". The answer is

Table 7: The prompt we use to directly leverage LLMs to perform KGC. Tail Prompt and Head Prompt mean the input to predict the missing tail and head entity respectively.

	ChatGPT			PaLM2			KG-S2S		
	H@1	H@3	H@10	H@1	H@3	H@8	H@1	H@3	H@10
WN18RR	11.4	13.5	15.4	11.5	16.6	21.3	52.5	59.7	65.4
FB15k-237	9.7	11.2	12.4	11.5	16.6	21.7	25.7	39.3	49.8
FB15k-237N	15.6	17.6	19.6	15.7	20.8	25.4	28.5	38.8	49.3
YAGO-3-10	4.5	5.0	5.4	6.4	8.8	11.4	-	-	-

Table 8: ChatGPT and PaLM2’s results on other KGC datasets.

B Details of Various KGC Pipelines

B.1 Discriminative KGC Pipelines

KG-BERT (Yao et al., 2019) is the first to propose utilizing PLMs for triplet modeling. It employs a special “[CLS]” token as the first token in input sequences. The head entity, relation, and tail entity are represented as separate sentences, with segments separated by [SEP] tokens. The input token representations are

	FB15k-237N		
	H@1	H@3	H@8
PaLM2-1-shot	15.7	20.8	25.4
PaLM2-2-shot	16.9	22.1	26.8
PaLM2-4-shot	17.7	23.1	27.9

Table 9: Experiment results of the demonstration number's effect on LLMs when performing KGC.

constructed by combining token, segment, and position embeddings. Tokens in the head and tail entity sentences share the same segment embedding, while the relation sentence has a different one. The input is fed into a BERT model, and the final hidden vector of the "[CLS]" token is used to compute triple scores. The scoring function for a triple (h, r, t) is calculated as $s = f(h, r, t) = sigmoid(CWT)$, where s is a 2-dimensional real vector and CWT is the embedding of the "[CLS]" token. Cross-entropy loss is computed using the triple labels and scores for positive and negative triple sets.

CSProm-KG (Chen et al., 2023a) combines PLM and traditional KGC models together to utilize both textual and structural information. It first concatenates the entity description and relation description behind a sequence of conditional soft prompts as the input. The input is then fed into a PLM, denoted as P, where the model parameters are held constant. Subsequently, CSProm-KG extracts embeddings from the soft prompts, which serve as the representations for entities and relations. These representations are then supplied as input to another graph-based KGC model, labeled as G, to perform the final predictions. It also introduces a local adversarial regularization (LAR) method to enable the PLM P to distinguish tCSProm-KGextually similar entities. Finally, CSProm-KG utilizes the standard cross entropy loss with label smoothing and LAR to optimize the whole pipeline.

B.2 Generative KGC Pipelines

In GenKGC (Xie et al., 2022), entities and relations are represented as sequences of tokens, rather than unique embeddings, to connect with pre-trained language models. For missing tail entities in triples $(e_i, r_j, ?)$, descriptions of e_i and r_j are concatenated to form the input sequence, which is then used to generate the output sequence. BART is employed for model training and inference, and a relation-guided demonstration approach is proposed for encoder training. This method leverages the fact that knowledge graphs often exhibit long-tailed distributions and constructs demonstration examples guided by the relation r_j. The final input sequence format is defined as: $x =< BOS > demonstration(r_j) < SEP > d_{e_i}, dr_j < SEP >$, where d_{e_i} and dr_j are description of the head entity and relation respectively. And $demonstration(r_j)$ means the demonstration examples with the relation r_j. Given the input, the target of GenKGC in the decoding stage is to correctly generate the missing entity. Additionally, an entity-aware hierarchical decoding strategy has been proposed to improve the time efficiency.

Following them, KG-S2S (Chen et al., 2022a) adds the entity description in the decoder end, training the model to generate both the missing entity and its corresponding description. It also maintains a soft prompt embedding for each relation to facilitate the model to distinguish the relations with similar surface meanings. Additionally, it adopts a sequence-to-sequence dropout strategy by randomly masking some content in the entity description to avoid model overfitting in the training stage.

C Additional Implementation Details

We show the detailed statistics of the KGC datasets we use in Table 10. Table 11 displays the hyper-parameters we adopt for each baseline model and dataset.

Dataset	# Entity	# Relation	# Train	# Valid	# Test
WN18RR	40,943	11	86,835	3,034	3,134
FB15k-237N	14,541	93	87,282	7,041	8,226

Table 10: Statistics of the Datasets.

model	dataset	batch size	learning rate	epoch	α
KG-BERT	WN18RR	32	5e-5	5	0.1
	FB15k-237N	32	5e-5	5	0.1
CSProm-KG	WN18RR	128	5e-4	500	1.0
	FB15k-237N	128	5e-4	500	1.0
GenKGC	WN18RR	64	1e-4	10	1.0
	FB15k-237N	64	1e-4	10	1.0
KG-S2S	WN18RR	64	1e-3	100	0.5
	FB15k-237N	32	1e-3	50	0.5

Table 11: Details of hyper-parameter settings for each baseline and dataset.

D Implementation Details of Reconstruction for Generative KGC Models

In the case of GenKGC, we adhere to the denoising pre-training methodology used in BART (Lewis et al., 2020). This approach commences by implementing a range of text corruption techniques, such as token masking, sentence permutation, document rotation, token deletion, and text infilling, to shuffle the integrity of the initial text. The primary objective of BART's reconstruction task is to restore the original corpus from the corrupted text.

For KG-S2S, we follow the pre-training approach proposed by T5 (Raffel et al., 2020). This approach employs a BERT-style training objective and extends the concept of single token masking to encompass the replacement of text spans. In this process, we apply a 15% corruption ratio for each segment, randomly substituting a span of text with a designated special token "<extra_id>". Here we employ a span length of 3. The ultimate goal of T5's reconstruction task is to accurately predict the content associated with these special tokens.

E Additional Case Study

In this section, we provide detailed examples to illustrate the input and output of each generating path we adopt in the descriptive context/ rationale extraction stage. We present examples in Table 12, 13, 14, 15, 16.

Triplet	(J.G. Ballard, place_of_birth, Shanghai)
Prompt	Given a triplet (J.G. Ballard| people, person, place_of_birth | Shanghai),please generate a paragraph to introduce "J.G. Ballard" and "Shanghai" and reflect their relationship "people, person, place_of_birth".
Output	J.G. Ballard (1930-2009) was an English writer. He was born in Shanghai, China, and his early experiences there shaped his writing. His novels often explored themes of alienation, technology, and the future. Ballard's work has been praised for its originality and its insights into modern society. He is considered one of the most important writers of the 20th century. Shanghai is a city in China. It is one of the most populous cities in the world, and it is a major center of commerce and culture. Shanghai has a long history, and it has been home to many different cultures over the centuries. The city is known for its skyscrapers, its nightlife, and its vibrant atmosphere. J.G. Ballard was born in Shanghai in 1930. He lived there until he was eight years old, when his family moved to England. Ballard's early experiences in Shanghai had a profound impact on his writing. His novels often explore themes of alienation, technology, and the future. Ballard's work has been praised for its originality and its insights into modern society. He is considered one of the most important writers of the 20th century.

Table 12: Descriptive context obtained from the generating path $T \longrightarrow (ED, TD)$.

Triplet	(J.G. Ballard, place_of_birth, Shanghai)
Prompt	Given a triplet (J.G. Ballard | people, person, place_of_birth | Shanghai), please generate a short paragraph to introduce "J.G. Ballard" and "people, person, place_of_birth". "J.G. Ballard" and "people, person, place_of_birth" must be concluded in the generated text.
Output	J.G. Ballard (1930-2009) was a British novelist, short story writer, and essayist. Born in Shanghai, China, he was raised there until the age of eight, when his family returned to England. Ballard's early experiences in China had a profound impact on his writing, and many of his novels and stories explore themes of alienation, isolation, and dystopia. He is considered one of the most important science fiction writers of the 20th century.

Table 13: Descriptive context obtained from the generating path $T \longrightarrow ED$.

Triplet	(J.G. Ballard, place_of_birth, Shanghai)
Prompt	Given a triplet (J.G. Ballard | people, person, place_of_birth | J.G. Ballard), please generate a paragraph to reflect their relationship people, person, place_of_birth. "J.G. Ballard" and "Shanghai" must concluded in the generated text.
Output	J.G. Ballard was born in Shanghai, China on November 15, 1930. He was a British writer, best known for his science fiction novels and short stories. His work often explored themes of alienation, technology, and the human condition. Ballard's books have been translated into more than 30 languages and have sold over 15 million copies worldwide. He died in London on April 19, 2009.

Table 14: Descriptive context obtained from the generating path $T \longrightarrow TD$.

Triplet	(J.G. Ballard, place_of_birth, Shanghai)
Prompt-Head	Given a triplet (J.G. Ballard | people, person, place_of_birth | Shanghai), please generate a short paragraph to introduce "J.G. Ballard"
Prompt-Tail	Given a triplet (J.G. Ballard | people, person, place_of_birth | Shanghai), please generate a short paragraph to introduce "Shanghai"
Prompt-Triplet	Given a triplet (J.G. Ballard | people, person, place_of_birth | Shanghai) and the description of the two entities J.G. Ballard: <Output-Head>Shanghai: <Output-Tail>please generate a short paragraph to reflect their relationship people, person, place_of_birth. "J.G. Ballard" and "Shanghai" must be concluded in the generated text.
Output-Head	J. G. Ballard was born in Shanghai and is a British writer. He is best known for his novels about science fiction and post-apocalyptic fiction. Some of his most famous works include "The Drowned World", "The Crystal World", and "Empire of the Sun".
Output-Tail	Shanghai is a metropolis on the east coast of China. It is the largest city in China and one of the most populous cities in the world. Shanghai is a major economic and cultural center, and is home to many skyscrapers and other landmarks. The city is also known for its nightlife and its vibrant street life. J.G. Ballard, the author of "Empire of the Sun", was born in Shanghai in 1930.
Output-Triplet	J. G. Ballard was born in Shanghai, China in 1930. He is a British writer best known for his novels about science fiction and post-apocalyptic fiction. Some of his most famous works include "The Drowned World", "The Crystal World", and "Empire of the Sun".

Table 15: Descriptive context obtained from the generating path $T \longrightarrow ED \longrightarrow TD$. <Output-Tail> and <Output-head> refer to the tail description and head description generated by the LLM in previous steps.

Triplet	(J.G. Ballard, place_of_birth, Shanghai)
Prompt-Tail	Given a triplet (J.G. Ballard ǀ people, person, place_of_birth ǀ [MASK]), please generate rationale to explain why [MASK] should be Shanghai
Prompt-Head	Given a triplet ([MASK] ǀ people, person, place_of_birth ǀ J.G. Ballard), please generate rationale at least 100 words to explain why [MASK] should be J.G. Ballard
Output-Tail	The rationale for [MASK] should be Shanghai is that:1. J.G. Ballard was born in Shanghai, China.2. Ballard's father was a British engineer who worked in China.3. Ballard spent his early childhood in Shanghai.4. Ballard's experiences in Shanghai influenced his writing.5. Ballard's novels and short stories often feature themes of alienation, isolation, and dystopia.6. Ballard's work has been praised for its originality, its insights into human nature, and its depiction of the future.In conclusion, the rationale for [MASK] should be Shanghai is that Ballard was born and raised there, and his experiences in Shanghai shaped his writing.
Output-Head	J.G. Ballard was born in Shanghai on 15 November 1930. His father was a British businessman and his mother was a Chinese woman. Ballard lived in Shanghai until the Japanese invasion in 1937, when he and his family were evacuated to England. Ballard's experiences in Shanghai during the war had a profound impact on his writing, and many of his novels are set in dystopian cities that are reminiscent of Shanghai. Ballard's work is often characterized by its bleak vision of the future, but it also contains a strong element of hope. Ballard believed that the future could be a place of beauty and wonder, but only if we are willing to confront the challenges that we face.

Table 16: Rationale obtained from the generating path $T \longrightarrow RA$