Know the Known and the Unknown: Reasonable Answer Generation with Knowledge-Informed Citations

Yichi Zhang[♠◇], **Zhuo Chen**[♠◇], **Lingbing Guo**[♠◇], **Jun Xu**[♣], **Mengshun Sun**[♣], **Zhizhen Liu**[♣], **Lei Liang**[♣], **Wen Zhang**[♠◇*], **Huajun Chen**[♠◇♡*]
[♠] Zhejiang University [♣] Ant Group
[◇] Zhejiang University - Ant Group Joint Laboratory of Knowledge Graph
[♡] Zhejiang Key Laboratory of Big Data Intelligent Computing
{zhangyichi.each, zhang.wen, huajunsir}@zju.edu.cn

Abstract

Question answering (QA) with reference texts is a classic application scenario for large language models (LLMs), where high standards for the credibility and traceability of generated answers are crucial. Many existing approaches focus on generating multi-level citations linked to specific references within the answer, making it verifiable and trustworthy. However, they often overlook key challenges such as citation granularity, the awareness of unknown information, and the adoption of effective training strategies. In this paper, we introduce **K**nowledge-in**F**ormed **C**itation (KFC), which addresses these issues through a novel data construction pipeline, a new benchmark, and an innovative training strategy. With ∼42K samples spanning 19 distinct domains, KFC includes both traditional citations referencing known entity-level information and specialized citations referring to unknown knowledge in the given question. This structure provides a more granular approach to citations, guiding the model to recognize and explicitly indicate unknown information, thus enhancing the quality and credibility of the response. Additionally, we propose a self-correction paradigm, SELF-KFC, designed to fine-tune LLMs by refining poorly cited answers into more accurate ones, making it particularly suitable for citation-dependent scenarios. We present comprehensive experimental results to demonstrate the effectiveness and generalization of SELF-KFC on the KFC benchmark.

1 Introduction

Large language models (LLMs) (Llama Team, 2024) have revolutionized research with their advanced text generation capabilities. Among various applications, question answering (QA) has emerged as a key domain, where LLMs either generate direct answers or base their responses on retrieved reference documents. QA with LLMs has found widespread use across diverse fields, including finance (Liu et al., 2023), healthcare (Nori et al., 2023), and customer service (Zhang et al., 2024b).

Table 1: Comparasion of existing citation benchmarks.

Benchmark	Citation Level	Unknown Awareness	Training	Data Scale
ALCE	Chunk	✗	SFT	∼3K
LongCite	Sentence	✗	SFT	∼45K
SYNSCIQA	Sentence	✗	SFT	∼3K
WebCite	Chunk	✗	SFT	∼7K
KFC (Ours)	Entity	✓	SELF-KFC	∼42K

Many application domains place stringent demands on the accuracy and verifiability of LLM-generated responses. For instance, when LLMs are tasked with answering questions or summarizing information from multiple sources, key phrases in the generated content must be properly attributed to their corresponding sources. Citations (Gao et al., 2023) serve this purpose by marking key phrases with references, akin to scientific papers, enabling validation of whether the model appropriately cites source material and mitigating the risk of hallucinated content. With the rise of retrieve-augmented generation (RAG) (Gao et al., 2024), the need for enhanced citation generation capabilities in LLMs is becoming increasingly critical in practical applications, making it a significant area of current LLM research.

Though many existing research works (Gao et al., 2023; Deng et al., 2024; Zhang et al., 2024a) focus on developing benchmarks with citation integration and corresponding evaluation protocols to assess the citation generation capability of LLMs. However, these efforts overlook several critical issues from three perspectives: citation granularity, awareness of unknown information, and effective training strategies, as shown in Table 1. First, their data construction pipelines tend to be simplistic, often relying on coarse-grained approaches such as

[*] Corresponding authors.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 18996–19011

chunk-to-entity or sentence-to-entity matching for citation labeling in QA pairs. This lack of precision complicates the creation of high-quality datasets. Secondly, the datasets they construct are often idealized, assuming that the reference texts alone can answer the questions effectively. In practice, however, reference texts may be incomplete, leaving certain aspects of the question unanswered. As citations mark known information, we propose that unknown information also be marked like a warning, indicating missing key details in the reference, and readers should also be careful to verify its authenticity. This would guide LLMs to acknowledge gaps and avoid generating hallucinated answers. Lastly, existing studies generally rely on conventional supervised fine-tuning (SFT) for model training, without exploring more advanced or tailored training strategies that could better address the complexities of citation-enhanced answer generation.

To address these issues, we propose a citation benchmark dataset called **K**nowledge-in**F**ormed **C**itation (KFC) accompanied by a newly designed construction pipeline. Building on the foundational QA pair formulation, KFC integrates a chain-of-thought (CoT) prompt, which facilitates a structured analysis of text chunks and input questions to identify both known and unknown entities relevant to the current query. This dual-grained entity knowledge, encapsulating both known and unknown information, is incorporated into the CoT, enhancing the subsequent answer generation process. Special citation markers like $[1], [2], [?]$ are employed in the answer to indicate references and highlight unknown information. An automated pipeline, leveraging LLMs, is developed to generate the KFC benchmark. In addition, a set of evaluation metrics is introduced to assess the citation capabilities in terms of both known and unknown knowledge. To further improve performance, we propose the self-correcting training strategy, SELF-KFC, which combines the strengths of supervised fine-tuning (SFT) and preference alignment (PA) methods (Rafailov et al., 2023). This approach allows for efficient training via self-correction, leading to more accurate citation generation compared with SFT and PA, which are commonly used by existing citation generation works. We conduct extensive experiments on KFC using leading LLM backbones to demonstrate the effectiveness of SELF-KFC through a comprehensive analysis. Our contributions can be summarized as:

- **Benchmark Construction**: We introduce a novel citation generation benchmark, KFC, along with an automated pipeline. The QA data pairs in KFC integrate structured knowledge via chain-of-thought (CoT), guiding LLMs to analyze both known and unknown information at the entity level, thereby improving answer quality.

- **Training Strategy Design**: We propose a new self-correction-based training strategy that enhances the LLM's citation generation capabilities. This approach leverages the strengths of vanilla SFT and DPO, widely used by existing citation generation works to achieve superior performance.

- **Extensive Experiments**: We conduct comprehensive experiments using the KFC benchmark and SELF-KFC training strategy to demonstrate the effectiveness, self-evolutionary, and generalization of our design.

2 Related Works

Generating verifiable citations in the text is an interesting topic for LLM research, aiming to enhance the reliability and factual correctness of the answers. ALCE (Gao et al., 2023) is the first step of this field, which collects a lot of corpora to build the citation benchmark for evaluation. A further work (Huang et al., 2024) employs fine-grained rewards to teach LLM to generate better citations on ALCE. WebCiteS (Deng et al., 2024) is another Chinese benchmark for the query-oriented summarization task with citation enhancement. LongCite (Zhang et al., 2024a) proposes a coarse-to-fine pipeline to build a sentence-level citation benchmark. SynSciQA (Schimanski et al., 2024) builds a citation dataset for evidence-based QA. AGREE (Ye et al., 2024) focuses on test-time adaptation for LLM citation generation. SelfCite (Chuang et al., 2025) proposes a self-supervised method for citation generation. Some LLM-based agents (Wang et al., 2024) can generate paper-level citations for long survey paper generation guided by complex agentic pipelines. Compared with these works, our work focuses on entity-level fine-grained citation with both known and unknown information while attempting to explore a new training strategy different from vanilla SFT and DPO.

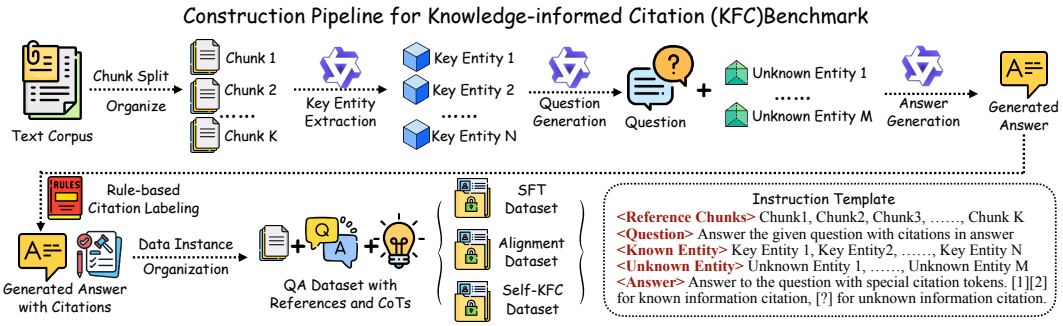

Figure 1: Overview of the construction pipeline for KFC benchmark, which consists of 6 steps called corpus preparation, key entity extraction, question generation, citation labeling, and data instance organization to build datasets for SFT and alignment.

3 Problem Formulation

We first formalize the QA with citations with the following notations. Given a question $\mathcal{Q}$ with k reference documents $\mathcal{R} = \{\mathcal{R}_1, \mathcal{R}_2, \cdots, \mathcal{R}_k\}$, the LLM $\mathcal{M}$ would generate a response $\mathcal{A}$, which would consists of a series of citations $\mathcal{C} = \{c_1, c_2, \cdots, c_m\}$ which would be an index for one of the documents in $\mathcal{R}$. Besides, we introduce a new kind of citation marks $\mathcal{U} = \{u_1, u_2, \cdots, u_m\}$ to mark the unknown information with special tokens rather than an index number. Note that for any citation $c_i \in \mathcal{C} \cup \mathcal{U}$, there is a statement s_i to form a pair (c_i, s_i), which means the citation c_i is used to mark s_i to a certain known document or unknown knowledge. They can be extracted from the generated answer for further evaluation. The overall target is to equip $\mathcal{M}$ with the citation generation capability by fine-tuning $\mathcal{M}$ with manually construed QA pairs.

4 Methodology

4.1 KFC Dataset Construction Pipeline

In this paper, we introduce a new citation generation benchmark, called knowledge-informed citation (KFC), which includes not only traditional citations for well-established information in the references but also citations for previously unknown information. We begin by outlining the detailed construction pipeline of KFC, which comprises 6 steps: corpus preparation, key entity extraction, question generation, answer generation, citation annotation, and data instance organization. The pipeline overview is presented in Figure 1.

Step1: Corpus Preparation The construction of KFC follows a clear, systematic process: we first prepare reference texts, then generate questions based on these texts, and finally construct specific Chain-of-Thought (CoT) prompts and answers using these reference texts. Thus, the first step involves preparing the reference corpus. For KFC, we use the corpus provided by UltraDomain (Qian et al., 2025), a multi-disciplinary collection of long texts spanning 19 different domains. UltraDomain offers long-form documents extracted from textbooks, which we then split into text chunks, each with a maximum word length of 256. We group consecutive k chunks into a set of references, denoted as $\mathcal{R}$, which will be further used to generate the complete dataset.

Step2: Key Entity Extraction Using the reference set $\mathcal{R}$, we employ a robust large language model (LLM) $\mathcal{M}_0$, as the data generator to synthesize the entire dataset. The next step is to perform structural understanding on the reference set $\mathcal{R}$. For each $\mathcal{R}_i \in \mathcal{R}$, we apply $\mathcal{M}_0$ to extract key entities within the chunk, denoted as $\mathcal{E}_i = (e_1, e_2, \ldots, e_n) = \mathcal{M}_0(\mathcal{I}ee, \mathcal{R}i)$, where $\mathcal{I}ee$ is the instruction template for entity extraction. Note that $\mathcal{E}_i$ represents an entity list containing at most n entities. Consequently, we obtain $\mathcal{E} = (\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_k)$, which constitutes the set of known entity extraction results for the k reference chunks.

Step3: Question Generation The next step is to generate a question, denoted as $\mathcal{Q}$, based on the reference corpus $\mathcal{R}$. It is important to note that we need to consider not only questions that can be answered using the current reference, but also those that can not be answered effectively with the available information. To address this, we employ a novel strategy for constructing eligible questions. Specifically, we manually design two instruction templates, $\mathcal{I}know$ and $\mathcal{I}unknow$ respectively, which guide $\mathcal{M}_0$ to generate questions that can / can not be answered perfectly based

on the reference $\mathcal{R}$, respectively. During the question generation process, we randomly select one of the templates as input and generate the question. As a result, the two types of questions will be approximately balanced (1:1) in the final dataset. Furthermore, for questions that can not be answered completely, we employ $\mathcal{M}_0$ to identify the key unknown entity or concept related to the question, denoted as $\mathcal{E}'$. For questions that can be answered fully, $\mathcal{E}'$ is set to the empty set, i.e., $\mathcal{E}' = \emptyset$.

Step4: Answer Generation At this point, we have the reference $\mathcal{R}$, the question $\mathcal{Q}$, and the known/unknown entity lists $\mathcal{E}$ and $\mathcal{E}'$. We then prompt $\mathcal{M}_0$ to generate a gold answer $\mathcal{A}$ based on the provided information as $\mathcal{A} = \mathcal{M}_0(\mathcal{I}_{ans}, \mathcal{R}, \mathcal{Q}, \mathcal{E}, \mathcal{E}')$, where $\mathcal{I}_{ans}$ is an instruction template designed to guide $\mathcal{M}_0$ in answering the question using the reference and key entities. The incorporation of entity-level knowledge serves to direct the model's attention to both the known and unknown information, ensuring that relevant details are emphasized in the generated answer. In addition, we generate another answer, $\mathcal{A}'$, using a different instruction template, $\mathcal{I}'ans$, which guides $\mathcal{M}_0$ to produce an answer with citation marks. While $\mathcal{M}_0$ may not generate high-quality citations, this answer is still valuable for further usage.

Step5: Citation Annotation Using the original answer $\mathcal{A}$, we label the answer with entity-level citations through a rule-based strategy. Specifically, we apply regular expressions to identify known and unknown entities from $\mathcal{E}$ and $\mathcal{E}'$ that appear in $\mathcal{A}$. In our initial exploration, we found that LLMs are less reliable for this task, as they tended to generate more errors in the citations. As a result, we adopted a rule-based approach for citation annotation. Following this process, we obtain a well-labeled answer, denoted as $\widehat{\mathcal{A}}$.

Step6: Data Instance Organization Finally, we organize the data obtained from the pipeline into instances suitable for LLM training and evaluation. Each data instance consists of an input that includes the reference documents and the question, which can be denoted as $\mathcal{X} = (\mathcal{R}, \mathcal{Q})$. The output, unlike the traditional direct generation of answers using references, incorporates a structured understanding process as a CoT prompt in the LLM. Thus, the output $\mathcal{Y} = (\mathcal{E}, \mathcal{E}', \widehat{\mathcal{A}})$ contains a detailed CoT that analyzes the known and unknown key entities in the current context, followed by the cited answer.

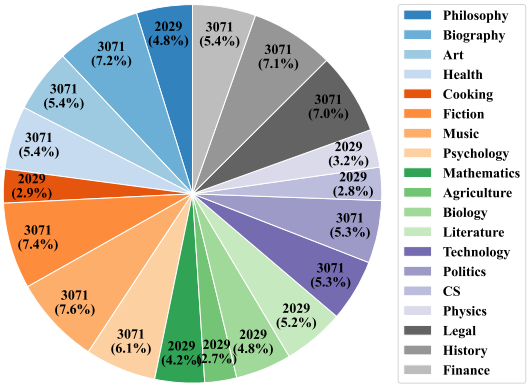

Figure 2: Overview of the KFC benchmark. KFC consists of ∼40K data instances from 19 different domains.

We then construct a dataset, $\mathcal{D}_{sft} = \{(\mathcal{X}_i, \mathcal{Y}_i)\}_{i=1}^{N}$, that supports vanilla supervised fine-tuning (SFT) to incorporate citation generation capabilities for LLMs. In addition to SFT, another common LLM training approach involves preference-based data for preference alignment (e.g., DPO (Rafailov et al., 2023)). For this, we construct a separate compliant dataset. The golden answer $\widehat{\mathcal{A}}$ serves as the preferred answer, while the cited answer $\mathcal{A}'$ generated by $\mathcal{M}_0$ is treated as the unpreferred answer. The preference data pair is denoted as $\mathcal{Y}^{(p)} = (\mathcal{E}, \mathcal{E}', \widehat{\mathcal{A}})$ and $\mathcal{Y}^{(u)} = (\mathcal{E}, \mathcal{E}', \mathcal{A}')$, where $\mathcal{Y}^{(p)}$ and $\mathcal{Y}^{(u)}$ represent the preferred and unpreferred answers, respectively. The preference alignment dataset is then given by $\mathcal{D}_{align} = \{(\mathcal{X}_i, \mathcal{Y}_i^{(p)}, \mathcal{Y}_i^{(u)})\}_{i=1}^{N}$.

KFC Benchmark Overview We provide an overview of the composition of KFC in Figure 2, which contains 42624 data instances across 19 different domains. The data is further divided into training, validation, and test sets in an 8:1:1 ratio. Note that while the dataset scales for both SFT and alignment remain the same, the data format differs.

4.2 Evaluation Protocol

This evaluation protocol assesses LLM-generated answers through three key metrics: (1) Entity Extraction Compatibility, measured by F1-score for both known and unknown entities based on precision and recall of predicted versus true entity sets; (2) Citation Quality, evaluated separately for known and unknown citations using F1-score to measure accuracy against gold standards (Gold) and self-consistency with Chain-of-Thought prompts (Self); and (3) Overall Answer Quality, scored by a strong LLM judge due to the complexity of long-form QA

data, providing a scalar assessment against gold labels. These metrics collectively form a comprehensive framework for benchmarking LLM performance. We present more detailed information on the evaluation protocol in Appendix B.

4.3 Self-Correct KFC Training Strategy

Naive Solutions for LLM Training With KFC, we can train LLMs with its training set to incorporate citation generation capability into LLMs. Vanilla SFT employs $\mathcal{D}_{sft}$ with vanilla next token prediction objective for decoder LLMs, which can be denoted as:

$$\mathcal{L}_{sft} = -\mathbb{E}_{(\mathcal{X}_i,\mathcal{Y}_i)\sim\mathcal{D}_{sft}} \log P_{\mathcal{M}}(\mathcal{Y}_i \mid \mathcal{X}_i) \quad (1)$$

With preference dataset $\mathcal{D}_{align}$, classic alignment methods like DPO (Rafailov et al., 2023) fine-tunes LLMs as:

$$\mathcal{L}_{dpo} = -\mathbb{E}_{(\mathcal{X}_i,\mathcal{Y}_i^{(p)},\mathcal{Y}_i^{(u)})\sim\mathcal{D}_{align}} \log \sigma\beta \left(\log \frac{P_{\mathcal{M}}(\mathcal{Y}_i^{(p)} \mid \mathcal{X}_i)}{P_{\mathcal{M}_{ref}}(\mathcal{Y}_i^{(p)} \mid \mathcal{X}_i)} - \log \frac{P_{\mathcal{M}}(\mathcal{Y}_i^{(u)} \mid \mathcal{X}_i)}{P_{\mathcal{M}_{ref}}(\mathcal{Y}_i^{(u)} \mid \mathcal{X}_i)} \right) \quad (2)$$

where σ is the sigmoid function, β is the temperature parameter and $\mathcal{M}_{ref}$ is the reference model for DPO. With such an objective, DPO would improve the probability of preferred answers and reduce the probability of unpreferred answers. Also, DPO has many improved versions like ORPO (Hong et al., 2024) and SimPO (Meng et al., 2024), which optimize DPO with certain new designs and can also be used for LLM training on the KFC dataset.

Although vanilla SFT and alignment methods are effective, we've found they are neither efficient nor well-suited for citation generation. SFT focuses only on teaching LLMs basic output formats, without enhancing annotation ability. Methods like DPO, which rely on preferred data, are time-consuming to train. Additionally, citation annotation is rarely accurate on the first attempt, leading to many errors and misannotations. To address this, we propose a new strong baseline to train LLMs for citation generation that combines the strengths of SFT and alignment methods while enabling self-evolution for improved citation generation.

Our SELF-KFC Method We propose self-correct KFC (SELF-KFC) to address the issues. Motivated by aligner (Ji et al., 2024), we use a simple SFT paradigm with optimized prompts to enhance the citation generation capability of LLMs. We design a new self-correct prompt data $(\mathcal{X}_i', \mathcal{Y}_i')$ where $\mathcal{X}_i' = (\mathcal{X}_i, \mathcal{Y}_i^{(u)}), \mathcal{Y}_i = \mathcal{Y}_i^{(p)}$ for $(\mathcal{X}_i, \mathcal{Y}_i^{(p)}, \mathcal{Y}_i^{(u)}) \sim \mathcal{D}_{align}$. In this process, we reorganize the preference data pair $(\mathcal{X}_i, \mathcal{Y}_i^{(p)}, \mathcal{Y}_i^{(u)})$ into new format by provide a bad answer in the given input instruction and guide the LLMs try to optimize the bad answer with the given context information. The training objective would be:

$$\begin{aligned}\mathcal{L}_{self} &= -\mathbb{E}_{(\mathcal{X}_i',\mathcal{Y}_i')\sim\mathcal{D}_{self}} \log P_{\mathcal{M}}(\mathcal{Y}_i' \mid \mathcal{X}_i') \\ &= -\mathbb{E}_{(\mathcal{X}_i,\mathcal{Y}_i^{(p)},\mathcal{Y}_i^{(u)})\sim\mathcal{D}_{align}} \log P_M(\mathcal{Y}_i^{(p)} \mid \mathcal{X}_i, \mathcal{Y}_i^{(u)})\end{aligned} \quad (3)$$

This method strikes a balance between alignment and SFT by shifting the preference alignment process to an in-context approach, encouraging LLMs to learn to self-correct and refine citations in poor answers. It leverages the constructed preference dataset for training via Self-KFC, while also enabling iterative self-evolution through a cycle of training, data generation, and re-training. We will demonstrate the effectiveness of this design in our experiments.

5 Experiments and Evaluation

In this section, we introduce the basic settings and the implementation details for our experiments. We further present the experiments on the effectiveness, self-evolutionary, reasonability, and generalization of SELF-KFC.

5.1 Experimental Settings

5.1.1 Baseline Methods

We compare our proposed SELF-KFC with several different LLM training strategies, including vanilla SFT and popular alignment methods DPO (Rafailov et al., 2023), ORPO (Hong et al., 2024), and SimPO (Meng et al., 2024). The experiments are conducted are open-source LLM backbones including LLaMA3-8B (Llama Team, 2024), Qwen2.5-7B (QwenTeam, 2024), and Qwen3-8B (QwenTeam, 2025).

5.1.2 Implementation Details

We implement our training and inference process with LLaMA-Factory (Zheng et al., 2024) and vLLM (Kwon et al., 2023), which are two famous open-source projects. The experiments are conducted on a Linux server with 8×NVIDIA A100 GPUs. We set the max text length to 8192 and the batch size to 8 with BF16 precision. We train LLMs with LoRA (Hu et al., 2022) and set the

Table 2: The main results. The best result of each metric is **bold** and the sub-optimal result is underlined.

Backbone	Method	Entity		Citation (Gold)		Citation (Self)		Answer	Overall
		Known	Unknown	Known	Unknown	Known	Unknown	Quality	Score
LLaMA3-8B	**Zero-shot**	-	-	-	-	-	-	71.33	-
	SFT	63.17	68.96	20.14	61.55	26.47	77.47	77.20	56.42
	ORPO	63.56	69.20	20.06	61.16	26.17	77.55	77.39	56.44
	DPO	64.02	69.88	20.14	62.26	26.19	77.34	77.59	56.77
	SimPO	63.57	69.46	20.20	61.38	26.11	77.14	77.29	56.45
	SELF-KFC	**64.89**	**71.10**	**20.38**	**63.31**	**26.49**	**77.71**	**77.91**	**57.39**
Qwen2.5-7B	**Zero-shot**	-	-	-	-	-	-	74.26	-
	SFT	60.13	65.19	18.80	59.99	24.59	**77.77**	74.71	54.45
	ORPO	61.40	67.30	19.07	59.39	24.56	75.89	75.29	54.70
	DPO	57.97	63.64	17.24	57.64	22.76	75.98	74.60	52.83
	SimPO	59.96	64.49	18.24	59.21	23.81	77.97	73.68	53.91
	SELF-KFC	**63.62**	**69.39**	**19.80**	**61.77**	**25.71**	77.27	**76.08**	**56.23**
Qwen3-8B	**Zero-shot**	-	-	-	-	-	-	74.95	-
	SFT	60.64	65.81	19.06	59.44	25.15	**77.58**	76.18	54.83
	ORPO	63.11	70.30	19.90	**62.79**	25.70	77.57	77.66	56.71
	DPO	63.17	70.33	19.89	61.97	25.67	76.85	77.32	56.45
	SimPO	62.98	69.01	19.95	61.32	25.95	76.21	77.41	56.11
	SELF-KFC	**63.92**	**70.41**	**20.30**	61.44	**26.18**	77.22	**77.76**	**56.75**

LoRA rank to 8. The learning rate is searched in $\{1e^{-5}, 1e^{-4}, 3e^{-4}\}$ and we train all models with 3 epochs. For DPO, OPRO, and SimPO, we set the weight λ_{sft} to 0.1. Other hyper-parameters follow the default settings of LLaMA-Factory. During evaluation, we use Qwen2.5-72B (QwenTeam, 2024) as the LLM judger to measure the answer quality in the range [0, 100]. We present all the instruction templates in Appendix C.

5.2 Main Experiment Results

The main experimental results are presented in Table 2, which shows that SELF-KFC outperforms baseline methods in both overall score and most individual metrics. For instance, SELF-KFC achieves a 1.1% and 2.8% improvement in overall performance on LLaMA3-8B and Qwen2.5-7B, respectively. However, the model's citation generation capability remains relatively low compared to the gold standard, indicating room for further improvement. Notably, our method shows the largest gains in entity recognition, with more modest improvements in unknown entity citation. This is due to the cascading thinking process in our CoT design, where accurate recognition of key entities leads to better citation annotations, while the generally lower number of unknown entities makes them easier to annotate in the generated answer.

5.3 Self-evolution Experiments

As mentioned earlier, SELF-KFC trains LLMs in a self-evolutionary manner, allowing the initial dataset $\mathcal{D}_{self}$ to be updated for iterative training. In this approach, an LLM-generated answer is treated as a new "bad" answer to replace the old one, guiding the model towards a golden answer even when starting with a non-trivial initial answer. This design emerged from the observation that, over time, the bad answers in the original dataset became increasingly trivial, making them less effective as bad examples. By generating new answers, we encourage it to create more meaningful counterexamples that motivate LLMs to improve. More detailed settings is presented in Appendix D.3.

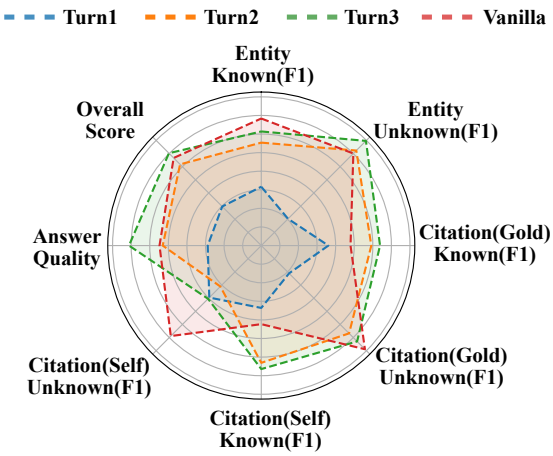

Figure 3: Self-evolution experiments of SELF-KFC. We use min-max normalization to normalize all the metrics.

We conduct self-evolution experiments over three turns, with one epoch per turn. After each epoch, the training dataset is updated with the LLM-generated answers. The results, compared to vanilla training without self-evolution, are shown in

Setting	Full	w/o know	w/o unknow	w/o all
AQ	77.91	75.51	77.17	74.95

Table 3: Ablation study on answer quality (AQ) of different prompt designs.

Figure 3. As seen, all metrics improved as training progressed. After three turns, the LLM demonstrated better performance in answer quality and overall score. Although entity recognition and citation metrics fluctuated, the overall quality of the responses showed significant improvement, indicating that this self-evolutionary approach effectively enhances the model's performance.

5.4 Ablation Study

Compared to vanilla SFT, SELF-KFC incorporates a "bad" answer for self-correction, which improves the final answer quality, as shown in Table 2. We also conduct an ablation study to evaluate the impact of the CoT prompts in KFC on answer quality. Unlike existing approaches, we introduce a CoT process to analyze both known and unknown entities in the documents and questions. To assess the effectiveness of these components, we perform three experiments: without known entities, without unknown entities, and without CoT. As shown in Table 3, removing any part of the CoT leads to a decrease in performance, highlighting the importance of a detailed CoT for question and reference analysis. Additionally, the results indicate that the analysis of known entities contributes more to the final answer.

5.5 Generalization Exploration

To further explore the generalization capability of SELF-KFC, we conduct two additional sets of experiments: one on low-resource KFC data and another on LLM backbones of varying sizes.

5.5.1 Low-Resource Data Experiments

The low-resource experiments aim to assess the generalization capability of SELF-KFC in data-scarce scenarios. We conducted these experiments on LLaMA3-8B, as shown in Figure 4, where the data proportion varied from 10% to 50%. The results indicate that SELF-KFC outperforms baseline methods even with limited data. This demonstrates its robustness and ability to generalize effectively under low-resource conditions.

In contrast, traditional SFT struggles significantly in extremely low-resource settings, with its performance dropping sharply when only 10% of the data is used. The improvement of SELF-KFC is especially notable in these scenarios, highlighting the effectiveness of the new prompt design. By incorporating a tailored CoT process, we optimize how instruction data is utilized, enabling the model to leverage the limited data more effectively and achieve superior performance despite the scarcity.

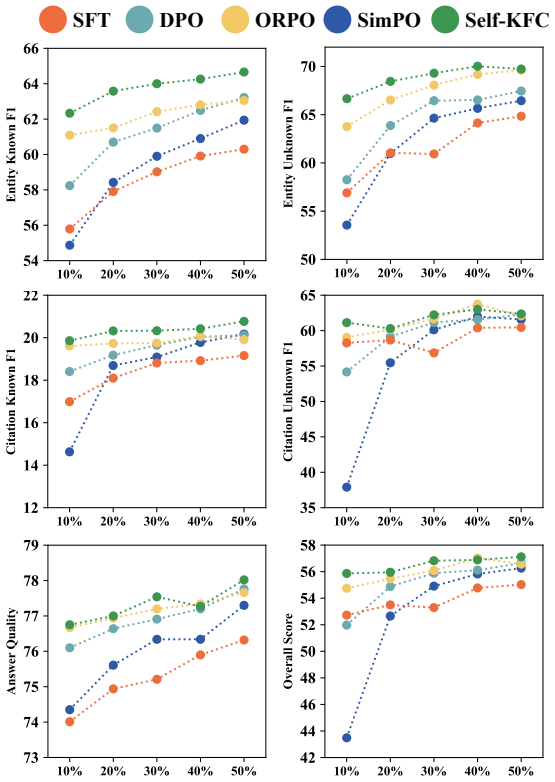

Figure 4: Low-resource data experiment results.

5.5.2 Different LLM Backbone Experiments

In the previous experiments, we mainly used LLMs of 7B-8B sizes. To further validate the generalization of SELF-KFC on different LLMs, we conduct experiments on the Qwen2.5 model series, including 0.5B/1.5B/3B/7B/14B, and/34B. As shown in Figure 5, SELF-KFC still outperforms baseline methods on all-size backbones. The results indicate that SELF-KFC is more specialized in the enhancement of citation ability in LLMs with sizes between 1.5B and 7B. Qwen2.5-0.5B has an obvious performance bottleneck due to its low upper capacity limit, which makes it difficult to get better results no matter what optimization method is used, while the larger LLM has a high lower capacity limit, which makes different training methods effective, of which ours performs the best.

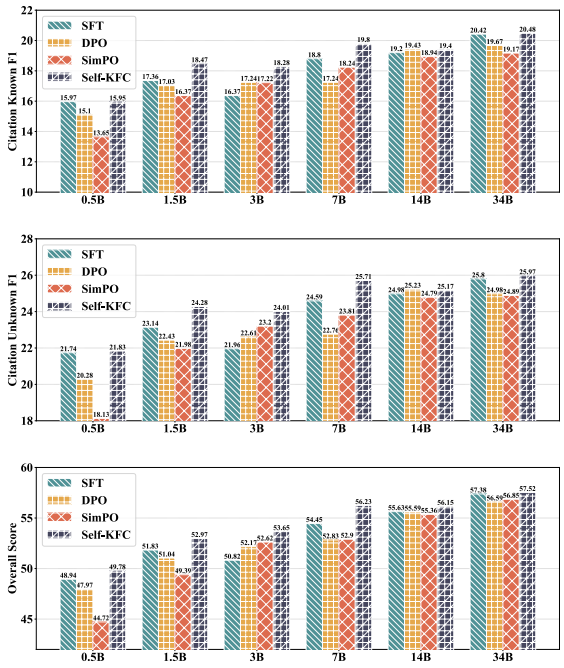

Figure 5: Results on Qwen2.5 with different sizes.

5.5.3 Overall Generalization Ability

Combining the results of the two experiments, we find that SELF-KFC excels despite data scarcity and smaller model sizes. Since real-world applications often face similar challenges, such as limited data and deployment scale, this suggests that SELF-KFC is highly adaptable to practical LLM deployment scenarios. We also conduct cross-domain generalization experiments and present the detailed results in Appendix D.4, which also indicates interesting conclusions on the citation generation capability of LLMs on the KFC benchmark.

5.6 Case Study

We further present some detailed cases in Figure 6 to show the effectiveness of SELF-KFC. Since the overall context information was too long to put down in the main paper, we chose to put out the quoted response section. In this particular case, the question has no relevant information in the reference, meaning the key entities in the question are unknown entities in the given context. We can observe that there is little difference between the answers produced by different training methods and the gold standard in terms of content—there is no noticeable performance gap in the factual accuracy of the answers. However, when it comes to citation quality, our method outperforms both SFT and DPO. Unlike these methods, our approach avoids significant omissions, such as the failure to cite "Populist [2]." This intuitive case demonstrates the effectiveness of our method in enhancing the citation generation capabilities of LLM.

Case Study

Question: What role did Johnny Vogel play in the Populist Party's efforts to address the concerns of debt-ridden farmers in the Midwest?

Golden Answer: Johnny Vogel [4] ran unsuccessfully for Iowa's state senate, representing The Populist [2] Party's efforts to address the concerns of debt-ridden farmers in the Midwest.

SFT Answer: Johnny Vogel [4] ran unsuccessfully for Iowa's state senate, representing the Populist [2] Party [4] 's efforts to address the concerns of debt-ridden farmers in the Midwest.

DPO Answer: Johnny Vogel [4] ran unsuccessfully for Iowa's state senate, becoming one of the few genuine farmers to join the Populist Party [4] 's ranks.

SELF-KFC Answer: Johnny Vogel [4] played a significant role in addressing the concerns of debt-ridden farmers [4] in the Midwest through his involvement with the Populist [2] Party.

Figure 6: A case to shown the effectiveness of SELF-KFC compared with SFT and DPO employed on LLaMA3-8B.

6 Conclusion

In this paper, we address a novel challenge in LLM applications: generating two types of citations in the QA scenario to improve the credibility and reliability of LLM responses. We introduce a new benchmark, KFC, along with innovative pipelines that incorporate structural analysis of the provided documents and questions through a chain-of-thought (CoT) process, further enhancing answer quality. Additionally, we propose a new training method, SELF-KFC, which enables LLMs to learn in a self-correcting manner. SELF-KFC strikes a balance between traditional supervised fine-tuning (SFT) and alignment techniques, such as DPO. Our results demonstrate the effectiveness of this approach in the new problem setting.

Limitations

Our work in this paper still consists of the following limitations:

Scope of the Work. Our work focuses on exploring QA-related problems, with limited in-depth exploration of other NLP tasks. In fact, many other types of tasks also require a combination of citation generation and the ability to handle unseen information. The dataset constructed in this work does not consider tasks beyond QA.

Limited Data Scale. Although the dataset we constructed is larger than most benchmarks and comparable to the largest LongCite dataset, it remains limited to tens of thousands of entries and could be further expanded. Achieving this goal requires more diverse data sources and automated pipelines.

Further Exploration on Methodology. Our newly designed approach represents a trade-off between SFT and PA. While this approach demonstrates a degree of innovation, it still holds significant room for improvement. For instance, it could incorporate finer-grained prompt design and optimization, as well as the introduction of reinforcement learning techniques.

Ethical Considerations

All experiments conducted in our work utilize open-source datasets and LLM models. No data collection or usage practices violating scientific ethics occurred during dataset construction or the experimental phase.

Acknowledgements

This work is founded by National Natural Science Foundation of China (NSFC62306276 / NSFCU23B2055), New Generation Artificial Intelligence-National Science and Technology Major Project 2030 (2025ZD0122800), Yongjiang Talent Introduction Programme (2022A-238-G), and Fundamental Research Funds for the Central Universities (226-2023-00138). This work was supported by Ant Group.

References

Yung-Sung Chuang, Benjamin Cohen-Wang, Zejiang Shen, Zhaofeng Wu, Hu Xu, Xi Victoria Lin, James R. Glass, Shang-Wen Li, and Wen tau Yih. 2025. Selfcite: Self-supervised alignment for context attribution in large language models. In *Forty-second International Conference on Machine Learning*.

Haolin Deng, Chang Wang, Xin Li, Dezhang Yuan, Junlang Zhan, Tianhua Zhou, Jin Ma, Jun Gao, and Ruifeng Xu. 2024. Webcites: Attributed query-focused summarization on chinese web search results with citations. In *ACL (1)*, pages 15095–15114. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *EMNLP*, pages 6465–6488. Association for Computational Linguistics.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

Jiwoo Hong, Noah Lee, and James Thorne. 2024. ORPO: monolithic preference optimization without reference model. In *EMNLP*, pages 11170–11189. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net.

Chengyu Huang, Zeqiu Wu, Yushi Hu, and Wenya Wang. 2024. Training language models to generate text with citations via fine-grained rewards. In *ACL (1)*, pages 2926–2949. Association for Computational Linguistics.

Jiaming Ji, Boyuan Chen, Hantao Lou, Donghai Hong, Borong Zhang, Xuehai Pan, Tianyi Qiu, Juntao Dai, and Yaodong Yang. 2024. Aligner: Efficient alignment by learning to correct. In *NeurIPS*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Xiao-Yang Liu, Guoxuan Wang, and Daochen Zha. 2023. Fingpt: Democratizing internet-scale data for financial large language models. *CoRR*, abs/2307.10485.

AI @ Meta Llama Team. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. In *NeurIPS*.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of GPT-4 on medical challenge problems. *CoRR*, abs/2303.13375.

Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. 2025. Memorag: Boosting long context processing with global memory-enhanced retrieval augmentation. In *Proceedings of the ACM Web Conference 2025 (TheWebConf 2025)*, Sydney, Australia. ACM. ArXiv:2409.05591.

QwenTeam. 2024. Qwen2.5: A party of foundation models.

QwenTeam. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*.

Tobias Schimanski, Jingwei Ni, Mathias Kraus, Elliott Ash, and Markus Leippold. 2024. Towards faithful and robust LLM specialists for evidence-based question-answering. In *ACL (1)*, pages 1913–1931. Association for Computational Linguistics.

Ling Team and Inclusion AI. 2025. Every activation boosted: Scaling general reasoner to 1 trillion open language foundation. *CoRR*, abs/2510.22115.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. 2024. Autosurvey: Large language models can automatically write surveys. In *NeurIPS*.

Xi Ye, Ruoxi Sun, Sercan Ö. Arik, and Tomas Pfister. 2024. Effective large language model adaptation for improved grounding and citation generation. In *NAACL-HLT*, pages 6237–6251. Association for Computational Linguistics.

Jiajie Zhang, Yushi Bai, Xin Lv, Wanjun Gu, Danqing Liu, Minhao Zou, Shulin Cao, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2024a. Longcite: Enabling llms to generate fine-grained citations in long-context QA. *CoRR*, abs/2409.02897.

Yichi Zhang, Zhuo Chen, Yin Fang, Yanxi Lu, Fangming Li, Wen Zhang, and Huajun Chen. 2024b. Knowledgeable preference alignment for llms in domain-specific question answering. In *ACL (Findings)*, pages 891–904. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

A Manual Quality Control and Assessment

In the dataset construction process, we make great effort to control the data quality of the KFC benchmark. To better control the dataset quality, we designed the synthesis approach for the KFC Benchmark by integrating the synthetic methodology developed for citation generation benchmarks. Since citation generation requires extensive human expertise, we employed a hybrid approach combining rule-based methods with LLM-driven automation for data synthesis. Specifically, we used LLM prompts to automatically generate responses while simultaneously applying rule-based techniques such as regular expressions to annotate and validate citations. Following synthesis, we conducted small-scale sampling and manual post-checking for each domain to ensure overall dataset quality. Specifically, when constructing the dataset, the authors randomly sampled 100 entries from the entire dataset and conducted manual verification to ensure a data qualification rate exceeding 95%. This process was repeated multiple times to assess the overall data quality.

The pipeline we designed integrates a small number of human-crafted rules with extensive LLM automation, aligning with current research trends. Numerous studies demonstrate that LLMs can achieve weak-to-strong generalization even when training data quality is imperfect, and models can learn crucial capabilities from it. By combining these insights with their vast existing knowledge and internalizing them, LLMs emerge with significantly enhanced capabilities.

B Evaluation Protocols

We design a set of evaluation metrics to assess the quality of LLM-generated answers on the KFC benchmark from multiple perspectives: the compatibility of entity extraction, the accuracy of two types of citations, and the overall quality of the answer. As mentioned earlier, we employ a Chain-of-Thought (CoT) prompt before the formal answer to facilitate a structured understanding of both the reference chunks and the question. This prompt

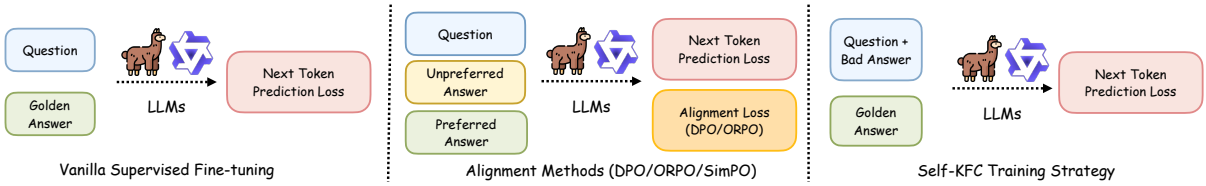

Figure 7: Comparison for SELF-KFC and existing paradigms like SFT and alignment methods.

incorporates both the known entities present in the documents and the unknown entities introduced in the question. Consequently, we evaluate the compatibility of the recognized entities (both known and unknown) using the F1-score.

$$P = \frac{|\mathcal{E}_{pred} \cap \mathcal{E}_{true}|}{\mathcal{E}_{pred}}, R = \frac{|\mathcal{E}_{pred} \cap \mathcal{E}_{true}|}{\mathcal{E}_{true}} \quad (4)$$

where $\mathcal{E}_{pred}$ and $\mathcal{E}_{true}$ represent the predicted and true entity sets, respectively. The F1-score for known entities can then be computed from these precision and recall values. For the extraction of unknown entities, the calculation process follows a similar procedure.

Meanwhile, we evaluate the citation quality for both known and unknown citations from two perspectives: the compatibility of the citation pairs with the golden answers and their self-consistency with the CoT prompt. For a citation-statement pair (c_i, s_i) or (u_i, s_i) in the generated answer, we consider it a correct prediction if it matches either the ground truth answer (Gold) or the previous CoT output (Self). The former metric follows the classical approach for citation evaluation, where citation F1 is used to measure the accuracy of citations related to known and unknown information separately. The latter metric aims to evaluate the consistency between the current citations and the CoT using the F1 metric. These metrics are calculated similarly to the known/unknown entity F1-scores.

Finally, we evaluate the general quality of the generated answers with the LLM-as-a-judge paradigm (Gu et al., 2025), which employs a strong LLM backbone to score the answers. As our datasets consist of long-form QA data, simple metrics like accuracy are not suitable, which is the motivation for such a design. A fixed scoring instruction $\mathcal{I}_{score}$ is used to compare all generated answers with the gold labels by assigning a scalar score. All of the metrics together form our comprehensive evaluation protocol, which synthesizes the performance differences across various LLMs and training methods on the SELF-KFC benchmark.

C Prompt Templates

Due to the page limit, we present the instruction template used in our work in the Appendix. During the construction process, we use several instructions in steps 2- 4, which are presented in Figure 8, Figure 9, and Figure 10. We further present the instruction data format of SFT and SELF-KFC in Figure 11 and Figure 12.

D Experiment Details

D.1 Training Settings

When training DPO/ORPO/SimPO, we initiated training from scratch and incorporated the SFT loss for the preferred answer with a weighting factor of 0.1 during loss calculation. This process imposes no additional burden on the original training. In practice, we found that this approach yielded no significant difference in performance compared to performing SFT first followed by DPO. For Self-KFC, more details can be found in Q4.

D.2 Training Time Costs

At present, our Self-KFC method has generally achieved significant performance gains. While enhancing performance, we have also improved efficiency compared to alignment methods like DPO. This is because we simplified DPO—which requires more GPU memory and runtime—into Self-KFC, a variant of SFT. This process actually reduces training time, which is another advantage of Self-KFC. According to our experiment logs, the training time tested in the same develop environment for different methods is:

- SFT: about 3 hours
- DPO/ORPO/SimPO: about 7 hours
- Self-KFC (our): about 4 hours

Instruction for Step2 Key Entity Extraction

You are an domain expert employed by us to construct domain knowledge graphs. Given the background knowledge document. Please help me to extract the most important key entities (no more than 5 entities) and relations from each documents. Report the entities, relations, and knowledge triples by a json-style file in this form: 'entity': [entity1, entity2, ...], 'relation': [relation1, relation2, ...], 'triple': [(entity1, relation1, entity2), ...] Note that the entities and relation should keep consistent with the documents. Do not include elements in the returned triple that do not appear in the entity and relation lists.
Here is the document:
{#Document}
Please return your json file without any other text in the response.

Figure 8: Instruction template for step2.

Overall, our approach achieves performance improvements while making efficiency trade-offs, thus demonstrating its value for research. On the other hand, this also indicates that the data for our synthetic citation generation task is quite challenging, making it difficult for the model to achieve significantly outperforming results at present.

D.3 Details of Self-evolution Experiments

In our instruction construction process, the first (unpreferred) bad answer for the self-KFC method comes from the zero-shot output of LLMs themseleves (not another one), while the (preferred) good answer is the label from our previously synthesized SFT data. We constructed our prompt in this manner. Additionally, for the self-evolution experiments in Section 5.3, we regenerate a batch of bad answers after each epoch of training and synthesize new instruction data, enabling the LLM to update more dynamically. Through this approach, we have achieved the self-evolution experiment of LLMs.

D.4 Cross-domain Generalization Experiments

We conduct another cross-domain generalization experiment on the KFC benchmark to explore the out-of-distribution (OOD) properties of the LLMs. We first divide the 19 domains in KFC into 6 larger domains:

- Social Science: Finance, Legal, Politics
- Technology: Technology, CS, Agriculture
- Science: Psychology, Physics, Mathematics, Biology
- Humanities: Fiction, Literature, History, Biography, Philosophy
- Arts: Music, Art
- Others: Health, Cooking

We then conduct OOD experiments on the 6 domains by training LLMs on one domain and evaluating them on different domains. The citation generation results are presented in Table 5. These

Table 4: Results on Ling-mini-2.0.

Benchmark	Answer Quality
Base	69.88
SFT	71.99
SimPO	71.48
SELF-KFC	74.61

results demonstrate that LLMs possess a certain degree of out-of-domain (OOD) generalization capability for citation generation. The model's training does not rely on domain-specific knowledge; instead, it acquires a general ability to incorporate citations during the output process. It is evident that training results vary significantly across different domains, and under the current task settings, out-of-distribution (OOD) performance does not necessarily outperform that achieved on the same distribution. For instance, in the case of Arts, training on Humanities data yielded superior results compared to training on the Arts dataset itself.

These results indicate that LLM citation generation capabilities can achieve cross-domain generalization and do not rely on domain-specific knowledge. The model's acquisition of this ability may

Table 5: The cross-domain (OOD) experiment results on Qwen2.5-7B.

Train	Evaluation	Entity		Citation (Ground Truth)		Citation (Self)		LLM Score
		Known	Unknown	Known	Unknown	Known	Unknown	
Arts	**Arts**	53.34	46.63	18.29	41.41	22.08	53.50	75.73
	Humanities	58.10	47.41	19.03	34.84	22.97	45.98	74.47
	Science	49.22	51.20	17.20	38.56	20.99	47.27	74.77
	Social	54.20	40.33	19.97	25.20	23.96	43.99	74.62
	Technology	49.26	43.04	16.19	24.54	19.59	42.75	77.49
	Other	51.09	46.92	14.88	28.56	18.94	41.13	75.60
Humanities	**Arts**	57.92	61.25	23.76	59.56	30.73	82.60	75.74
	Humanities	62.74	59.52	25.01	57.42	32.02	77.31	74.88
	Science	54.08	62.45	23.49	59.80	31.67	80.24	75.28
	Social	58.54	56.37	23.26	49.88	29.67	69.27	76.78
	Technology	54.04	59.70	22.17	51.13	29.98	71.38	79.08
	Other	56.43	55.69	21.14	54.49	29.00	75.04	76.24
Science	**Arts**	55.59	55.38	21.01	45.74	26.13	59.86	74.16
	Humanities	60.13	55.09	22.09	40.64	28.26	53.47	74.43
	Science	53.08	59.33	21.80	50.64	28.26	62.71	74.66
	Social	56.45	50.43	21.99	30.39	27.51	48.14	75.16
	Technology	52.34	51.05	20.51	37.40	27.56	55.40	74.90
	Other	54.59	54.96	17.47	38.89	24.68	54.18	77.95
Social	**Arts**	55.96	50.86	18.97	44.63	23.10	60.81	74.73
	Humanities	59.39	54.29	19.43	41.81	24.45	55.44	74.34
	Science	50.99	59.87	18.37	47.59	23.73	59.12	74.62
	Social	57.79	53.53	21.54	40.17	25.71	53.81	76.43
	Technology	51.51	53.57	16.97	41.16	21.74	64.05	78.56
	Other	53.39	54.34	15.37	40.10	20.42	55.41	75.78
Technology	**Arts**	51.08	44.98	10.85	23.27	12.85	35.08	72.60
	Humanities	55.01	45.27	11.74	22.18	15.00	31.50	72.11
	Science	48.18	52.01	12.42	28.96	15.71	40.80	73.16
	Social	51.79	37.81	14.99	18.16	17.67	31.84	71.52
	Technology	48.37	48.41	11.71	20.18	14.91	32.28	75.97
	Other	49.54	48.93	10.08	21.83	13.44	30.60	73.90
Other	**Arts**	47.07	40.85	5.36	13.07	6.76	19.32	71.42
	Humanities	50.24	41.81	5.63	10.08	7.19	14.61	70.81
	Science	41.86	48.80	7.03	14.53	9.02	19.92	71.75
	Social	46.08	31.32	9.71	8.19	11.61	16.53	69.88
	Technology	42.55	35.99	6.44	9.37	8.87	15.94	75.34
	Other	46.39	36.99	5.05	9.84	6.91	14.54	72.84

Instructions for Step3 Quetion Generation

Template1:
You are a high level NLP data annotator. Given the following series of documents, each document also gives several key entities.
You should generate a question that can be comprehensively answered by these documents, noting that the question you generate needs to be combined with the content of multiple documents.
After you have generated the question, you need to analyze the entities needed for the question and distinguish between those entities that are known in the documents and those that are unknown.
Here are the documents:
{#Documents}
Return your answer in json format, containing three keys, 'generated question' (string type), 'known entities' (list type), and 'unknown entities' (list type) to store your generated question and the entitity analysis results. For the case where all entities are known, unknown entities is an empty list. Note that the known entities should be consistent with the documents. Do not return any other text in the response.
Template2:
You are a high level NLP data annotator. Given the following series of documents, each document also gives several key entities.
You need to generate a question that is not entirely relevant and cannot be answered well by the provided documents. But it can not be too irrelevant.
After you have generated the question, you need to analyze the entities needed for the question and distinguish between those entities that are known in the documents and those that are unknown.
Here are the documents:
{#Documents}
Return your answer in json format, containing three keys, 'generated question' (string type), 'known entities' (list type), and 'unknown entities' (list type) to store your generated question and the entitity analysis results. The unknown entities should be presented in the 'unknown entities' list. Note that the known entities should be consistent with the documents. Do not return any other text in the response.

Figure 9: Instruction template for step3.

be more significantly influenced by other factors in the training data, such as quantity and quality.

D.5 Results on Ling-mini-2.0

We also conduct experiments on Ling-mini-2.0 (Team and AI, 2025). Here is a simple view of the results in Fugre 4. The results also indicates the effectiveness of SELF-KFC.

E Other Statements

In this paper, we employ existing LLM as our assistant to polish the writing of our paper.

Instructions for Step4 Answer Generation

You are a high level NLP data annotator. Given the following series of documents with several key entities. You need to generate the answer of the given question. We have analyzed the known key entities and unknown entities that need to be involved in the current problem.
Here are the documents:
{#Documents}
The question is:
{#Question}
The known entities for current question are:
{#Known Entities}
The unknown entities for current question are:
{#Unknown Entities}
Please return your answer directly. The answer should be concise. Do not return any other text in the response.

Figure 10: Instruction template for step4.

Instruction Template for SFT

Instruction:
Answer the given question with the following documents. Mark the important parts from the documents with citation reference in your answer.
Input Prompt
Documents:
{#Documents}
Question:
{#Question}
Output Prompt
Known Entitities:{}
Unknown Entities:{}
Answer:{}

Figure 11: Instruction template for SFT data.

Instruction Template for Alignment (DPO/ORPO/SimPO)

Answer the given question with the following documents. Mark the important parts from the documents with citation reference in your answer. We present you a bad answer for reference and you should give a better answer.
Input Prompt
Documents:
{#Documents}
Question:
{#Question}
Here is a bad answer for example:
{#Bad Answer} Your answer is: **Output Prompt**
Known Entitities:{}
Unknown Entities:{}
Answer:{}

Figure 12: Instruction template for SELF-KFC data.