SGIC: A Self-Guided Iterative Calibration Framework for RAG

Guanhua Chen[1], **Yutong Yao**[1], **Lidia S. Chao**[1], **Xuebo Liu**[2], **Derek F. Wong**[1*]
[1]NLP^{2}CT Lab, Department of Computer and Information Science, University of Macau
[2]Institute of Computing and Intelligence, Harbin Institute of Technology, Shenzhen, China
{nlp2ct.guanhua, nlp2ct.yutong}@gmail.com
liuxuebo@hit.edu.cn
{derekfw, lidiasc}@um.edu.mo

Abstract

Recent research in retrieval-augmented generation (RAG) has concentrated on retrieving useful information from candidate documents. However, numerous methodologies frequently neglect the calibration capabilities of large language models (LLMs), which capitalize on their robust in-context reasoning prowess. This work illustrates that providing LLMs with specific cues substantially improves their calibration efficacy, especially in multi-round calibrations. We present a new **SGIC**: **S**elf-**G**uided **I**terative **C**alibration Framework that employs uncertainty scores as a tool. Initially, this framework calculates uncertainty scores to determine both the relevance of each document to the query and the confidence level in the responses produced by the LLMs. Subsequently, it reevaluates these scores iteratively, amalgamating them with prior responses to refine calibration. Furthermore, we introduce an innovative approach for constructing an iterative self-calibration training set, which optimizes LLMs to efficiently harness uncertainty scores for capturing critical information and enhancing response accuracy. Our proposed framework significantly improves performance on both closed-source and open-weight LLMs.

1 Introduction

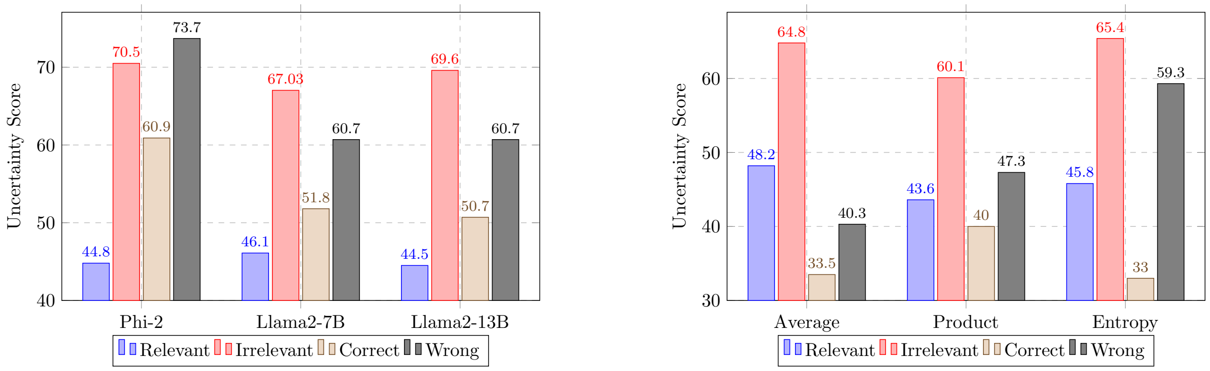
Retrieval-augmented generation (RAG) necessitates intricate reasoning across various candidate-retrieved documents with more than one hop. The advent of exceptionally large language models (LLMs) such as GPT-3.5 and Claude has markedly augmented the capabilities of robust in-context reasoning prowess, enabling significant advances in RAG performance via in-context learning or multi-step reasoning (Wang et al., 2023a; Khalifa et al., 2023), without additional training. Despite these enhancements, the deployment of such LLMs in local settings is frequently impractical, because of their proprietary nature and voluminous parameter sets. Consequently, researchers focus on fine-tuning open-weight LLMs to improve performance in specific downstream applications (Zheng et al., 2023; Du et al., 2022; Penedo et al., 2023).

However, lots of existing works on RAG primarily concentrate on extracting relevant documents or the refinement of specialized instructions (Asai et al., 2022; Ziems et al., 2023; Wang et al., 2023b; Sun et al., 2023; Ma et al., 2023b; Tang et al., 2023), which does not fully leverage the in-context reasoning abilities intrinsic to LLMs. Inspired by recent studies that employ LLMs for self-calibration to generate better answers through self-feedback mechanisms (Peng et al., 2023; Dhuliawala et al., 2023; Shinn et al., 2023), we argue that LLMs can optimize generated answers of RAG by utilizing prior responses coupled with strategic hints to facilitate in-context reasoning. The preliminary experimental findings, detailed in Section A.2, lend credence to our hypothesis. In addition, Figure 1a reveals a distinct gap in the uncertainty scores produced by LLMs when distinguishing correct/incorrect answers and the relevant/irrelevant documents. Meanwhile, several widely used uncertainty estimation approaches (Figure 1b) confirm the generality of this phenomenon.

Thus, we introduce a novel **SGIC**: **S**elf-**G**uided **I**terative **C**alibration Framework that harnesses the robust in-context reasoning capabilities of LLMs to self-calibrate previously generated answers. We adopt the uncertainty scores in the inference stage to iteratively rephrase the input prompts, incorporating the uncertainty scores of previously generated answers to steer the LLMs towards in-context reasoning and self-calibration. Additionally, we introduce document uncertainty scores to assess the relevance between each document and the question, assisting LLMs in retrieving the most pertinent documents. For small-scale models with limited long-document understanding capabilities, we fur-

[*] Corresponding authors.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 28357–28370
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics



(a) Three different baseline models.

(b) Three different uncertainty estimations.

Figure 1: The uncertainty score of the relevant/irrelevant documents and correct/wrong generated answers on 2,000 samples extracted from HotpotQA(Yang et al., 2018) dataset.

ther propose a strategy for reconstructing a self-calibration training dataset following the procedure in Figure 2, to fine-tune the LLMs to utilize the uncertainty scores of the document and previous responses to generate a better answer.

Empirically, we evaluate our framework across two benchmarks, HotpotQA (Yang et al., 2018) and Natural Question (NQ) (Kwiatkowski et al., 2019), on two strong close-source LLMs, **GPT-4o** and **GPT-4o mini** (Achiam et al., 2023) and two open-weight LLMs, which are **Phi-3.5** (Li et al., 2023) and **Llama2-7B-Chat** (Touvron et al., 2023). The experimental results demonstrate the validity and strong potential of our framework.

2 Related Work

2.1 Retrieval-Augmented Generation (RAG)

RAG retrieves relevant documents from a knowledge base and employs a generator to produce coherent and accurate responses based on the retrieved documents (Lewis et al., 2020). Recent studies (Jiang et al., 2023; Chen et al., 2024; Fan et al., 2024) have also demonstrated that RAG can effectively address the hallucination and incorrect reasoning problems of LLMS in the Question Answering (QA) and downstream tasks, such as Document Question Answering (DQA), which require LLMS to reason between multiple documents. Trivedi et al. (2022) mutually integrated the Chain-of-Thoughts (CoT) into the retrieval step to enhance the retrieval capability of LLMs for multi-hop QA. Ma et al. (2023a) proposed query rewriting RAG through Adaptive queries generated by a fine-tuned rewriter. Ma et al. (2023c) proposed an adaptive filter-then-rerank paradigm, prompting LLMs to rerank few-shot hard samples filtered by small LMs, which enhances the perception of key information of the LLMs. Jeong et al. (2024) proposed an adaptive QA framework, dynamically selecting optimal RAG strategy from simple to sophisticated through estimating query complexity by a trained smaller LLM. Zhang et al. (2024) proposed a retrieval-augmented fine-tuning strategy, enabling LLM to identify distractor documents and adapt to domains. As for datasets, there are several multi-hop QA corpora, such as HotpotQA (Yang et al., 2018) and WikiHop (Yang et al., 2018), which are widely used in RAG tasks.

2.2 Self-Calibration

Self-calibration refers to LLMs learning from automated feedback to improve their behavior and adapt over time, avoiding costly human feedback (Pan et al., 2023). Traditional approaches (Li et al., 2019; Unanue et al., 2021; Wu et al., 2021) utilize meticulously designed external matrix to measure the generative generation quality of LMs to guide the model to perform the self-calibration. Akyürek et al. (2023); Yan et al. (2023); Li et al. (2024b) further expanded these methods by modifying the matrix or introducing external revising modules. To avoid the increasingly complex matrix design, researchers construct frameworks such as CoT to calibrate the LLMs leveraging their strong inference ability iteratively(Zelikman et al., 2022; Wang et al., 2022b,a). In addition, several methods (Du et al., 2023; Li et al., 2024a; Liang et al., 2024) employ multi-LLM debating frameworks to calibrate model-generated answers. However, these approaches require fine-tuning multiple LLM agents or utilizing LLMs with extremely large parameters, such as ChatGPT, for many rounds of interaction.

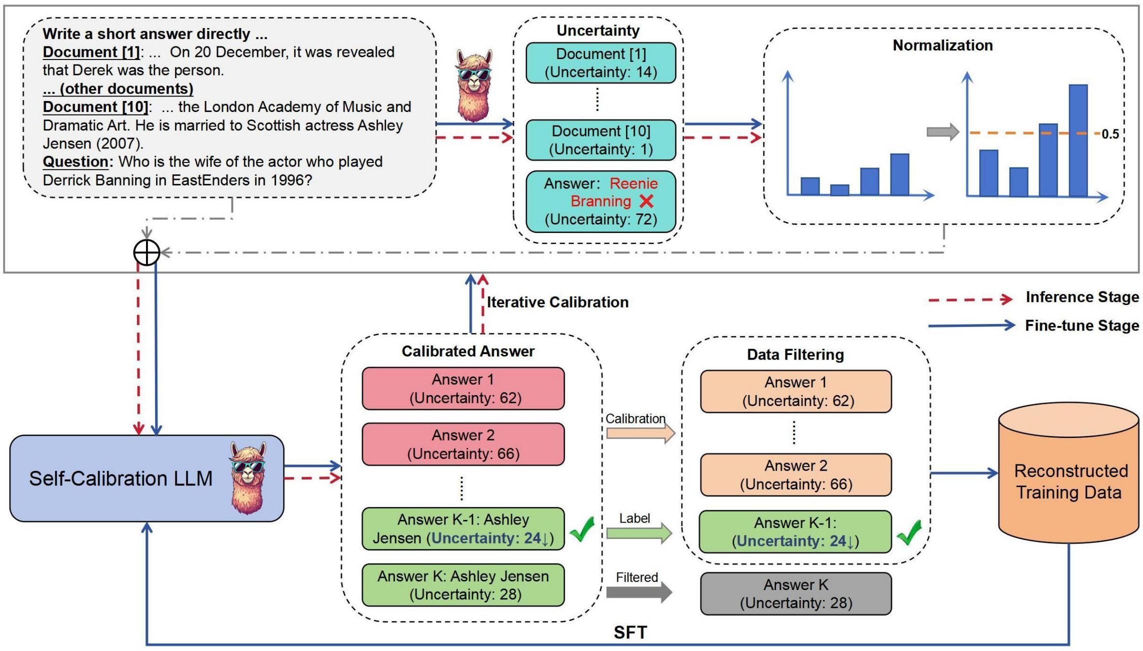

Write a short answer directly ...
Document [1]: ... On 20 December, it was revealed that Derek was the person.
... (other documents)
Document [10]: ... the London Academy of Music and Dramatic Art. He is married to Scottish actress Ashley Jensen (2007).
Question: Who is the wife of the actor who played Derrick Banning in EastEnders in 1996?
Uncertainty
Document [1] (Uncertainty: 14)
Document [10] (Uncertainty: 1)
Answer: Reenie Branning ✗ (Uncertainty: 72)
Normalization
0.5
Iterative Calibration
Inference Stage
Fine-tune Stage
Calibrated Answer
Answer 1 (Uncertainty: 62)
Answer 2 (Uncertainty: 66)
Answer K-1: Ashley Jensen (Uncertainty: 24↓) ✓
Answer K: Ashley Jensen (Uncertainty: 28)
Calibration
Label
Filtered
Data Filtering
Answer 1 (Uncertainty: 62)
Answer 2 (Uncertainty: 66)
Answer K-1: (Uncertainty: 24↓) ✓
Answer K (Uncertainty: 28)
Self-Calibration LLM
Reconstructed Training Data
SFT

Figure 2: The overview of our framework.

In contrast, our approach requires fine-tuning the LLMs only once, enabling them to perform comprehensive reasoning and self-calibration based on retrieved documents, previously generated answers, and their associated uncertainty scores.

3 Method

Figure 2 provides an overview of our proposed framework, which comprises three main components: (1) estimating the uncertainty scores of each document and the generated answers (Section 3.1); (2) iteratively utilizing the generated answers and their corresponding uncertainty scores from the validation set to perform the self-calibration process during the inference stage (Section 3.2); and (3) designing a strategy to reconstruct a new training set to fine-tune a self-guided iterative calibration LLM with uncertainty awareness (Section 3.3).

3.1 Uncertainty Estimation

To achieve our self-guided iterative calibration framework, we first employ the uncertainty score of the generated answer to evaluate the necessity of calibration according to the observation in Figure 1. For a given input sequence X=[x_1,x_2, ...,x_n] consisting of n tokens, the large language model generates a corresponding output sequence Y=[y_1,y_2, ...,y_m] with m tokens, accompanied by the corresponding token-level logits. We begin by applying the softmax function to the logits to derive the highest probability associated with each token, denoted as P=[p_1, p_2, …, p_m]. The uncertainty is then estimated by computing the product of these maximum probabilities, a widely adopted method for uncertainty estimation:

$$s_{ans} = (p_1 \times p_2 \times ... \times p_m) \qquad (1)$$

To ensure the consistency of this score across both the training and inference stages, we redefine Equation 1 as follows:

$$\hat{s} = \begin{cases} \dfrac{s_{ans} - \overline{s}_{ans}}{1 - \overline{s}_{ans}} & \text{if } s_{ans} < \overline{s}_{ans} \\ \dfrac{s_{ans} - \overline{s}_{ans}}{1 - \overline{s}_{ans}} & \text{otherwise} \end{cases} \qquad (2)$$
$$s'_{ans} = 100 \times (0.5 + 0.5 \times \hat{s})$$

where $\overline{s}_{ans}$ is the average uncertainty scores calculated by all generated answers in the training, validation, or test set seperately.

Furthermore, we incorporate uncertainty scores to evaluate the model's confidence regarding the relevance of each document to the given question. A lower uncertainty score indicates a higher likelihood of the document being relevant, thereby aiding the LLM in information retrieval. Inspired by Duan et al. (2023), we estimate the uncertainty score of each document by calculating the product of the maximum probabilities of the generated answers, obtained by combining the given question with each document individually.

Write a short answer ...
Document [1]: Yellowcraig, less commonly ...
... (other documents)
Document [10]: Dirleton Castle ...
Question: A medieval fortress in Dirleton, ...
Previous Generated Answer:
Round 1: Lord High ... (Uncertainty Score: 73)
Round 2: United States ...(Uncertainty Score: 51)
Answer:

Table 1: The input format of iterative self-calibration in our framework.

$$\begin{aligned} s_{doc} &= 1 - (p_1 \times p_2 \times ... \times p_m) \\ s'_{doc} &= 100 \times \frac{(s^i_{doc} - \text{Min}(s_{doc}))}{(\text{Max}(s_{doc}) - \text{Min}(s_{doc}))} \end{aligned} \tag{3}$$

where s^i_{doc} is the product of maximum probability with the i-th document subtracted from one. A higher product indicates the LLM is more confident that the document is relevant to the question and contains more adequate information. To estimate the uncertainty score, we subtracted this product from one and normalized the uncertainty scores of all documents to reduce sensitivities.

Notably, the uncertainty estimation method can be replaced with any improved metric that more accurately measures uncertainty values. We also discuss the precision of uncertainty estimation for SGIC in Appendix A.5.

3.2 Iterative Self-Calibration

During the inference stage, we primarily generate an answer based on the full documents and ascertain its uncertainty score employing Equation 2. Subsequently, we appraise the uncertainty score associated with the answer generated by each document following Equation 3. Then, we rephrase the input with these elements. As demonstrated in Figure 2, we integrate the documents with corresponding uncertainty scores, while attaching the primary answer combined with uncertainty scores as the first-round original answer. The reformulated input is supplied to the LLM, which then engages in in-context reasoning and calibrates the answer under the directive of the uncertainty scores.

Furthermore, self-calibration is iteratively conducted K rounds to obtain a satisfactory answer with minimal uncertainty. Specifically, in each round, the generated answer is re-input into the model for further calibration. During each calibration, we calculate the uncertainty scores of the answer according to Equation 2 and incorporate the answer and score into the "Previous Generated Answer" as illustrated in Table 1 to reconstruct the input of subsequent round, enabling the answer to be calibrated iteratively based on the given documents and the previously generated answers.

3.3 Uncertainty-aware Fine-tuning

After quantifying the uncertainty scores for the training data, we reformulate the representation of each data sample to incorporate these uncertainty scores, as illustrated in Figure 2. Using this reformulation, we construct an uncertainty-aware self-calibration dataset derived from the original training corpus. Following the pipeline outlined in Section 3.2, we restructure the input by including the content and uncertainty scores of documents alongside the primary answers.

The LLM is then employed to iteratively calibrate the answers for each data sample until the answer is correct or the round limit k is reached. After k rounds of calibration, samples with incorrect answers are removed. For the remaining samples, the final reformulated input is used as the training set input. Since the final input contains multi-round information in the "Previous Generated Answer" field, it provides the LLM with comprehensive knowledge of the calibration process. This pruning of unsuitable samples allows the LLM to focus on learning how to leverage the additional information encoded in the uncertainty scores for improved calibration. In parallel, a substitution operation is carefully designed to refine the model's ability to address answers with high uncertainty while preserving or optimizing answers with low uncertainty. This prevents the model from erroneously altering correct answers.

The refined training set enhances the model's ability to calibrate responses accurately while better capturing relevant information through the uncertainty scores. After the training set is restructured, we apply a standard supervised fine-tuning (SFT) procedure to fine-tune the self-guided iterative calibration LLM and evaluate its performance as described in Section 3.2.

4 Experiments

4.1 Setup

Dataset We evaluate our proposed framework on **HotpotQA** (Yang et al., 2018) and **Natural Question** (NQ) (Kwiatkowski et al., 2019) corpus. The

	HotpotQA		Natural Question (NQ)	
Model	EM	F1	EM	F1
Close-Source LLMs				
GPT-4o-mini	69.2	63.0	62.9	47.5
GPT-4o-mini (Ours)	**74.1**	**66.7**	**64.4**	**48.8**
GPT-4o	73.7	68.1	63.3	53.0
GPT-4o (Ours)	**76.5**	**70.8**	**65.2**	**55.0**
Open-Weight LLMs				
Phi-3.5-mini (Full Tuning)	42.8	50.3	58.7	64.3
Phi-3.5-mini (Ours)	**55.3**	**60.2**	**65.0**	**67.5**
Llama2-7B-Chat (LoRA Tuning)	69.1	73.5	74.7	77.9
Llama2-7B-Chat (Ours)	**77.2**	**80.5**	**79.0**	**81.2**

Table 2: The main experimental results (%) on the dev set of HotpotQA and Natural Question (NQ) datasets. **Bold** numbers indicate the better result for each baseline, which sampling K times following the iterative calibration process.

Dataset	Train	Validation	Test
HotpotQA	50,000	7,405	7,405
NQ	40,000	2,000	2,000

Table 3: The statistics of **HotpotQA** and **Natural Question (NQ)** in our experimental setting.

Question Type	EM	F1
Bridge	65.0	72.9
Bridge (Ours)	**75.7**	**79.4**
Comparison	69.6	73.2
Comparison (Ours)	**83.1**	**84.8**

Table 4: The results of different types of question (%) in HotpotQA dataset on Llama2-7B-Chat. **Bold** numbers indicate the best results.

statistic of these two datasets is shown in Table 3. **HotpotQA**[1] dataset includes 113k multi-hop questions. There are two types of questions: bridge and comparison. The final answer in the distractor setting is generated through 10 passages. Each question has at least 2 relevant passages. We also reconstruct the **Natural Question** (NQ)[2] (Kwiatkowski et al., 2019) dataset in the distractor setting similar to **HotpotQA** to evaluate the robustness of our framework. The question of the NQ dataset is comprised of a Google query and a corresponding Wikipedia page. Each page has a passage that can answer the question. We take nine other passages from the same page as distractors. Because of the high demand for computational resources in LLM and the computing resource limitations, we use part of the training set in our experiments.

Evaluation For all experiments in this work, we employ two widely used metrics, Exact Match (EM) and F1 score, to evaluate the performance of our framework. As noted in Huang et al. (2024), it is unreasonable to assume that the ground truth will always be present among all calibrated answers, as real-world scenarios typically lack access to the ground truth. To address this, our methodology involves performing K calibration iterations on samples that require calibration. From these iterations, we select the response with the minimal uncertainty score as the final correct answer.

Models For closed-source LLMs, we evaluate our proposed method on the **GPT-4o-mini** and **GPT-4o** models (Hurst et al., 2024) through the OpenAI API without fine-tuning on the downstream datasets. As for open-weight baselines, we employ two strong decoder-only large language models (LLMs) that vary in scale and architecture: **Phi-3.5** (Li et al., 2023) and **Llama2-7B-Chat** (Touvron et al., 2023). The implementation of experiments will be explained in Appendix A.1.

4.2 Main Results

As shown in Table 2, we show our main experimental results on two closed-source LLMs and two open-weight LLMs, which are evaluated with HotpotQA and Natural Question (NQ) datasets. For close-source LLMs, which are GPT-4o-mini and

[1] https://hotpotqa.github.io/

[2] https://github.com/google-research-datasets/natural-questions

Model	EM	F1
Llama2-7B-Chat (LoRA Tuning)	69.1	73.5
+ Calibration	71.8	75.3
+ Calibration & Answer Uncertainty	76.2	79.6
+ Calibration & Document Uncertainty (Ours)	**77.2**	**80.5**

Table 5: The ablation study results (%) on the dev set of HotpotQA dataset. **Bold** numbers indicate the best results.

Model	EM	F1
Llama2-7B-Chat (LoRA Tuning)	69.1	73.5
External Relevant Score	75.4	78.8
Oracle Uncertainty	**85.7**	**85.1**
Llama2-7B-Chat (Ours)	77.2	80.5

Table 6: The experimental results (%) on the dev set of HotpotQA dataset with different settings of the uncertainty scores in our method. **Bold** numbers indicate the best results.

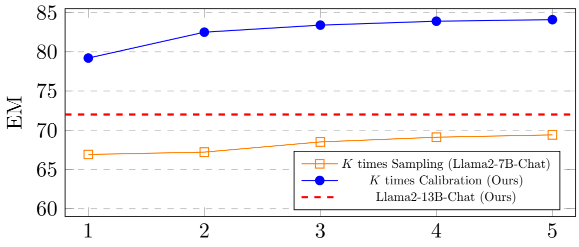


Figure 3: The EM scores of the Llama2-7B-Chat fine-tuned with our method to perform the K times calibrations on the dev set of the HotpotQA dataset.

GPT-4o, our method consistently achieves better performance on both EM and F1 scores compared with sampling the response K times directly. As for the open-weight LLMs, the Phi-3.5-mini and Llama2-7B-Chat models, which are fine-tuned with our proposed framework, also outperform the baselines on all EM and F1 scores, whether with the LoRA tuning or full tuning.

To explore the effectiveness of our approach on different types of problems, we also comparatively analyze the effectiveness of our approach on bridge and comparison types of questions within the dev set of HotpotQA dataset, which are shown in Table 4. Our framework obtained more than 10% improvements on both two types of questions. This suggests that our approach works not only on comparison questions with binary answers, but also on bridge questions with open answers, which demonstrate the robustness and generality of our method.

4.3 Ablation Study

To evaluate the various components of our proposed framework, we conducted a series of experiments to assess the performance impact of combining: (a) only the initial answer, (b) the initial answer with its uncertainty score, and (c) the uncertainty scores of documents. It should be noted that, for experiment (a), we selected the calibrated answer repeated in the second iteration as the final answer, following Huang et al. (2024), since uncertainty scores are not available when only the initial answer is used. As shown in Table 5, the combination of most components yields superior results in both EM and F1 evaluation metrics, demonstrating the effectiveness of our proposed methodology.

5 Analysis

5.1 The Impact of Uncertainty Scores

Despite the depiction of the observed uncertainty distribution in Figure 1, we conducted experiments to validate the necessity of this pattern within our framework. As evidenced in Table 6, we replaced the uncertainty scores of the documents with the relevant scores calculated by the **Multi-qa-mpnet-base-dot-v1** model, a powerful document extraction model proposed by Reimers and Gurevych (2019). We believe this is because the uncertainty scores computed by the model itself are strongly correlated with the model's capabilities, and therefore, more effectively guide the model in iterative self-calibration. Moreover, we consider an extreme case where the uncertainty scores for relevant/irrelevant documents and correct/wrong answers are perfectly accurate. In this scenario, we assign uncertainty scores ranging from 0 to 20 for relevant documents and correct initial answers, while irrelevant documents and wrong answers were given uncertainty scores spanning from 80 to 100. The EM scores improved to 85.7, and the F1 scores improved to 85.1, which further justifies the potential of our framework. Furthermore, we provide a theoretical analysis regarding the application of uncertainty scores in Appendix A.4.

5.2 The Performance of Iterative Calibration

We conducted additional experiments to thoroughly investigate the implications of the K-times calibration mechanism. The blue solid curve in Figure 3 illustrates the experimental outcomes with $K = \{1, 2, 3, 4, 5\}$ iterations of self-calibration.

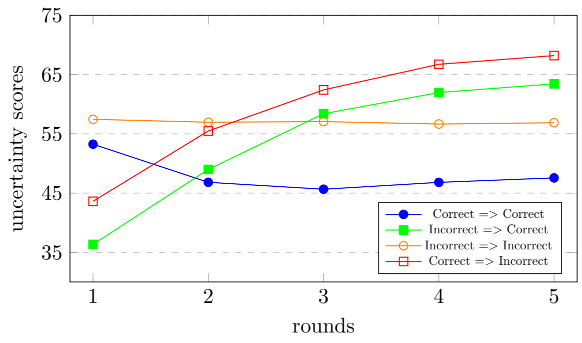


Figure 4: The uncertainty scores of each round of calibrations in four modes of answer variation.

Intuitively, the performance improves as the number of calibration iterations increases. Remarkably, it even surpasses the fine-tuned Llama2-13B-Chat model and the same model fine-tuned with our framework, demonstrating the potential of our approach. For a fair comparison, we also performed 1 to 5 sampling iterations with a temperature of 0.7 for the fine-tuned Llama2-7B-Chat model, represented by the solid orange line in Figure 3. As expected, the performance of the model gradually improves with an increasing number of samples. However, it is evident that our framework still shows a significant performance advantage over this baseline. Additionally, we compare the inference cost of our self-calibration method with the sampling baseline in Appendix A.3.

5.3 Uncertainty Change of Iterative Calibration

To clarify the iterative calibration process, we present the average uncertainty scores for how answers change in each calibration round, as illustrated in Figure 4. The uncertainty for "Correct => Correct" and "Incorrect => Incorrect" remains relatively stable, which makes sense since the answers do not change in these cases. However, the uncertainty scores for "Correct => Incorrect" and "Incorrect => Correct" gradually increase with more calibrations. This likely happens because as more "Previous Generated Answers" accumulate, the model engages in more complex reasoning, leading to higher uncertainty. Interestingly, the uncertainty for "Incorrect => Correct" samples is consistently lower than for "Correct => Incorrect" samples in each round, indicating that the model exhibits lower uncertainty when the calibration results in a correct answer. This can help the model recognize the successful calibrations.

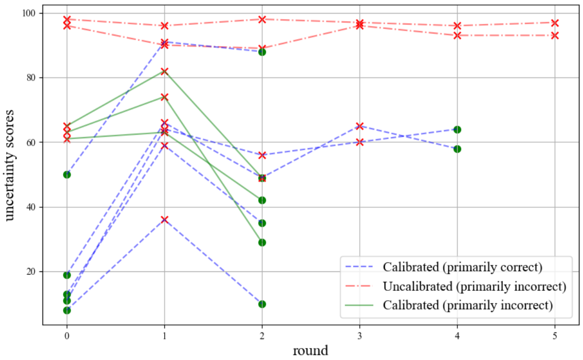


Figure 5: The answer correctness and uncertainty scores of 10 random samples during the iterative calibration. Round 0 refers to the initially generated answer without document uncertainty. Red samples refers to incorrect answers, while Green samples refers to correct answers.

Model	EM
Llama2-7B-Chat (Fine-tuned)	36.2
Llama2-7B-Chat (Ours)	**40.1**

Table 7: The experimental results (%) on the test set of GSM8K dataset. **Bold** numbers are the best results for each base model.

In more detail, we also conduct a more concrete analysis of the 10 randomly extracted samples and results in each round to visualize changes in their answer correctness and corresponding uncertainty scores throughout our calibration procedure. The result is demonstrated in Figure 5, which indicates three types of calibration procedures. For the samples that remain uncorrected after five times calibrations, the uncertainty scores persist at high levels during the whole process. This reveals the LLM is confused by the question and documents. In contrast, for those samples calibrated within five times, the LLM starts with being suspicious of the incorrect answer, as indicated by an increase in uncertainty scores in the first time calibration. The LLM then adjusts the suspected answer in the following iteration with a lower uncertainty score. Moreover, we observe that some initially correct answers are altered to be incorrect in the first round, As evidenced by a significant rise in uncertainty scores, the alternations resemble uncertain attempts. Subsequently, the LLM identifies the errors and calibrates the answers in subsequent iterations.

5.4 Generalizability

In addition to evaluating our approach to RAG tasks, we conduct experiments on other types of reasoning tasks to demonstrate their broader ap-

Dataset	EM	F1
Fine-tuned on NQ		
HotpotQA (Sampling 5 times)	54.5	57.5
HotpotQA (Ours)	**71.0**	**74.4**
Fine-tuned on HotpotQA		
NQ (Sampling 5 times)	58.7	67.4
NQ (Ours)	**76.5**	**78.8**

Table 8: The results of Llama2-7B-Chat on the test set of one dataset, fine-tuned on the other. **Bold** numbers indicate the best results.

plicability. Specifically, we employ the **GSM8K** dataset (Cobbe et al., 2021), which contains basic mathematical problems necessitating multi-step reasoning, to fine-tune both the baseline and our method on the Llama2-7B-Chat model. The GSM8K dataset comprises 7,473 training instances and 1,319 testing instances. For evaluation, we compute the Exact Match (EM) score using the final computation result, without considering the intermediate reasoning steps. Additionally, we treat each step in the reasoning process generated by the pre-trained LLM as a document in the RAG task to calculate the uncertainty score, which is then used for fine-tuning and self-calibration.

As presented in Table 7, fine-tuning the model by reconstructing the input to match the format shown in Table 12 yields an approximate 4% improvement in model performance with only one-time calibration. This consistent enhancement highlights the generalizability of our approach and its potential applicability to a range of reasoning tasks.

5.5 Transferability

We investigate the transferability of calibration capabilities in LLMs fine-tuned with our framework. Specifically, we assess the impact of using LLMs fine-tuned on one dataset when applied to the dev set of another dataset. As shown in Table 8, sampling five times with the LLM fine-tuned on a different dataset results in significantly lower performance compared to the baseline fine-tuned on the corresponding training set (Table 2). This occurs because, in the original QA task, LLMs primarily learn to extract answers from documents, often overlooking in-context reasoning abilities. Additionally, the single-hop of NQ and the multi-hop of HotpotQA influence the LLMs' transferability. However, our framework's self-calibration capability outperforms the baseline and nearly matches the performance of LLMs fine-tuned with the specific training set. This is due to our approach's emphasis on reasoning through multiple rounds of answers and documents, using uncertainty scores to identify more plausible answers, which is a common logic applicable across all datasets.

5.6 Error Analysis

To explore deeper into the distinctions between our method and baseline, we performed an error analysis using the HotpotQA dataset. In the baseline, error samples with low uncertainty are typically more likely to be calibrated successfully, whereas those with high uncertainty often fail calibration. Our analysis of calibration performance across 7,405 samples showed that 1,440 samples (19.5%) were successfully calibrated, while 1,080 samples (14.6%) failed. Among the successful calibrations, 75% (1,081 samples) were calibrated within the first two rounds. This was due to the high initial uncertainty in the baseline answers, which prompted divergent responses. The remaining 359 samples required iterative calibration: baseline answers with low uncertainty gradually accumulated doubt over rounds until a threshold was reached, prompting revised responses. Correct answers reinforced confidence, reducing uncertainty, while incorrect ones sustained high uncertainty, prompting further revisions. For the failures, 693 cases retained incorrect answers with declining uncertainty. A small subset proposed alternative incorrect answers accompanied by sharp drops in uncertainty, halting calibration. Notably, 387 failures reached the 5-round limit with rising uncertainty, suggesting potential calibration success if the rounds were extended. This indicates the framework's ability to iterative self-calibration, although round limits currently constrain efficacy.

5.7 Case Study

We present two examples in Table 13 to empirically showcase the capabilities of SGIC. In Example 1, our uncertainty-aware self-calibration for LLMs calibrates errors in generated answers using uncertainty scores from source documents and the initial answer. In Example 2, we illustrate the framework's ability to refine partially correct answers, enhancing both the completeness and accuracy of the responses. These examples provide compelling evidence of the effectiveness of our framework in improving the quality and reliability of LLM-generated answers.

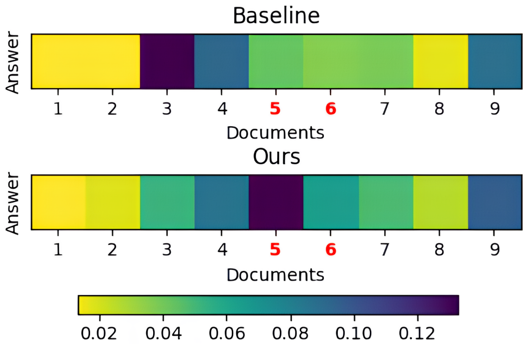


Figure 6: The visualization of the attention distribution of the given documents in a sample. Red means the relevant two documents.

5.8 Attention Distribution

To further investigate how our framework enables LLMs to self-calibrate, we visualize the attention distribution across various documents to analyze its impact on LLM parameterization. Figure 6 presents a comparison between the attention allocation of Llama2-7B-Chat fine-tuned with our framework (lower) and the baseline model (upper). Notably, the baseline model exhibits low attention distribution to relevant documents, whereas our self-calibration framework directs attention more effectively toward these relevant documents. This observation indicates that our framework enhances the model's ability to calibrate its answers by focusing attention on pertinent information.

For a more detailed evaluation, we examined the attention distribution for the whole test set of the HotpotQA dataset. Since the relevant documents may appear in different positions among all candidates, we quantified the difference using the $R_{10}@k$ score. This metric represents the proportion of the top two relevant documents ranked within the top k positions, based on the attention distribution in the LLM. We compared the attention weights from the baseline model and after the first calibration. The results, displayed in Table 10, reveal that even a single calibration with our framework significantly improves the model's focus on relevant documents compared to the baseline.

6 Conclusion

In this paper, we observed significant gaps in uncertainty scores from various LLMs and estimation methods between relevant/irrelevant documents and correct/incorrect answers in the RAG task. To address this, we proposed a novel framework that guides iterative calibration using the model's in-context reasoning abilities. Our framework consistently improves performance for both open-weight and closed-source models by utilizing uncertainty scores of documents and generated answers. These findings underscore the potential of uncertainty-aware self-calibration in enhancing the accuracy and reliability of large language models.

7 Limitation

Even though our proposed novel SGIC framework can utilize the in-context reasoning capabilities of large language models (LLMs) to iteratively self-calibrate the answers by leveraging the uncertainty scores of the given documents and the initially generated answers, it is contingent upon the precision of the underlying uncertainty estimation. While much current work has devoted considerable attention to methodologies for gauging the uncertainty indicative of the veracity of generated responses, there remains a dearth of studies addressing the assessment of the confidence regarding the pertinence of documents to the question. In the future, we will further explore how to improve the accuracy of uncertainty estimation, which can maximize the effectiveness of our framework.

Acknowledgments

This work was supported in part by the Science and Technology Development Fund of Macau SAR (Grant Nos. FDCT/0007/2024/AKP, FDCT/0070/2022/AMJ, FDCT/060/2022/AFJ), the National Natural Science Foundation of China (Grant Nos. 62261160648, 62266013), the China Strategic Scientific and Technological Innovation Cooperation Project (Grant No. 2022YFE0204900), and the UM and UMDF (Grant Nos. MYRG-GRG2023-00006-FST-UMDF, MYRG-GRG2024-00165-FST-UMDF, EF2024-00185-FST, EF2023-00151-FST, EF2023-00090-FST).

References

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

Afra Feyza Akyürek, Ekin Akyürek, Aman Madaan, Ashwin Kalyan, Peter Clark, Derry Wijaya, and Niket Tandon. 2023. Rl4f: Generating natural language feedback with reinforcement learning for repairing model outputs. *arXiv preprint arXiv:2305.08844*.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2022. Task-aware retrieval with instructions. *arXiv preprint arXiv:2211.09260*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17754–17762.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2023. Chain-of-verification reduces hallucination in large language models. *arXiv preprint arXiv:2309.11495*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Jinhao Duan, Hao Cheng, Shiqi Wang, Chenan Wang, Alex Zavalny, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2023. Shifting attention to relevance: Towards the uncertainty estimation of large language models. *arXiv preprint arXiv:2307.01379*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. In *The Twelfth International Conference on Learning Representations*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*.

Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. Few-shot reranking for multi-hop qa via language model prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15882–15897.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ming Li, Jiuhai Chen, Lichang Chen, and Tianyi Zhou. 2024a. Can llms speak for diverse people? tuning llms via debate to generate controllable controversial statements. *arXiv preprint arXiv:2402.10614*.

Moxin Li, Wenjie Wang, Fuli Feng, Fengbin Zhu, Qifan Wang, and Tat-Seng Chua. 2024b. Think twice before assure: Confidence estimation for large language models through reflection on multiple answers. *arXiv preprint arXiv:2403.09972*.

Siyao Li, Deren Lei, Pengda Qin, and William Yang Wang. 2019. Deep reinforcement learning with distributional semantic rewards for abstractive summarization. *arXiv preprint arXiv:1909.00141*.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: **phi-1.5** technical report. *arXiv preprint arXiv:2309.05463*.

Jingcong Liang, Rong Ye, Meng Han, Ruofei Lai, Xinyu Zhang, Xuanjing Huang, and Zhongyu Wei. 2024.

Debatrix: Multi-dimensinal debate judge with iterative chronological analysis based on llm. *arXiv preprint arXiv:2403.08010*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023a. Query rewriting for retrieval-augmented large language models. *arXiv preprint arXiv:2305.14283*.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023b. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*.

Yubo Ma, Yixin Cao, YongChing Hong, and Aixin Sun. 2023c. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! *arXiv preprint arXiv:2303.08559*.

Liangming Pan, Michael Saxon, Wenda Xu, Deepak Nathani, Xinyi Wang, and William Yang Wang. 2023. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, et al. 2023. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*.

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*. Association for Computational Linguistics.

Stephen E Robertson. 1977. The probability ranking principle in ir. *Journal of documentation*, 33(4):294–304.

Noah Shinn, Beck Labash, and Ashwin Gopinath. 2023. Reflexion: an autonomous agent with dynamic memory and self-reflection. *arXiv preprint arXiv:2303.11366*.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agent. *arXiv preprint arXiv:2304.09542*.

Raphael Tang, Xinyu Zhang, Xueguang Ma, Jimmy Lin, and Ferhan Ture. 2023. Found in the middle: Permutation self-consistency improves listwise ranking in large language models. *arXiv preprint arXiv:2310.07712*.

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*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*.

Inigo Jauregi Unanue, Jacob Parnell, and Massimo Piccardi. 2021. Berttune: Fine-tuning neural machine translation with bertscore. *arXiv preprint arXiv:2106.02208*.

Jinyuan Wang, Junlong Li, and Hai Zhao. 2023a. Self-prompted chain-of-thought on large language models for open-domain multi-hop reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2717–2731, Singapore. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2023b. Query2doc: Query expansion with large language models. *arXiv preprint arXiv:2303.07678*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022a. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022b. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Qingyang Wu, Lei Li, and Zhou Yu. 2021. Textgail: Generative adversarial imitation learning for text generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14067–14075.

Hao Yan, Saurabh Srivastava, Yintao Tai, Sida I Wang, Wen-tau Yih, and Ziyu Yao. 2023. Learning to simulate natural language feedback for interactive semantic parsing. *arXiv preprint arXiv:2305.08195*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shuyang Yu, Runxue Bao, Parminder Bhatia, Taha Kass-Hout, Jiayu Zhou, and Cao Xiao. 2024. Dynamic uncertainty ranking: Enhancing in-context learning for long-tail knowledge in llms. *arXiv preprint arXiv:2410.23605*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah Goodman. 2022. Star: Bootstrapping reasoning with reasoning. *Advances in Neural Information Processing Systems*, 35:15476–15488.

Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Noah Ziems, Wenhao Yu, Zhihan Zhang, and Meng Jiang. 2023. Large language models are built-in autoregressive search engines. *arXiv preprint arXiv:2305.09612*.

A Appendix

A.1 Implementation Details

We use the LLaMA-Factory[3] GitHub repository to fine-tune the **Llama2-7B-Chat** with LoRA (Hu et al., 2021) and **Phi-3.5-mini** with full-parameters mode because of the limitation of computation resource. We train all models using the batch size of 16. We set the initial learning rate to 5e-5 and fine-tuned all models 3 epochs. We truncate each document and ensure that its length is less than 200 tokens for all the open-weight LLMs. All the experiments have been completed on one 80G H800 GPU or 80G A100 GPU. Besides, all experiments are calibrated at most five rounds. During the calibration phase, we ensure an equitable evaluation by sampling k times for the baseline model the same as an equivalent number of experimental trials with the baseline model.

A.2 The Impact of the Tag In Context

Firstly, we leverage large language models (LLMs) without fine-tuning to conduct experiments aimed at evaluating the effectiveness of using specific tags as hints within the context to guide LLM reasoning. Table 11 illustrates the input format utilized in these experimental settings. We selected a random subset of 2,000 instances from the development set of the HotpotQA dataset. Each instance adds the “<key>” tags before the relevant documents.

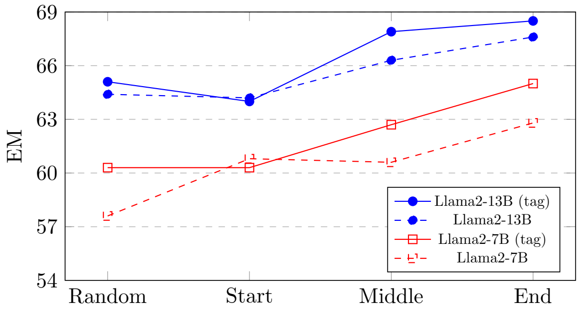


Figure 7: The experimental results of using the “<KEY>” tag to guide LLM on the RAG task at different order of documents. The solid line is the result of adding the “<KEY>” tag, while the dashed line is the baseline result.

Dataset	HotpotQA		NQ	
	EM	F1	EM	F1
Self-RAG	48.7	29.9	61.3	70.8
Ours	**77.2**	**80.5**	**79.0**	**81.2**

Table 9: The results of comparing our method with the Self-RAG framework.

To mitigate potential biases arising from input sequencing in LLMs, as identified by Liu et al. (2023), we systematically varied the positions of the given documents. In separate trials, the two relevant documents are placed at the beginning, middle, and end of the input sequence, as well as in random positions. The results of these experiments are depicted in Figure 7. Notably, the use of tags to direct the model's attention to relevant documents demonstrates an improvement in performance on retrieval-augmented generation (RAG) tasks. This finding supports our hypothesis that the model has the potential to calibrate its generated answers through in-context reasoning guided by specific tags.

A.3 Analysis of Inference Cost

One of the concerns regarding the self-calibration framework is its inference cost, as it requires generating an initial answer and then calibrating it. To address this, we conducted experiments to analyze the inference cost of our proposed method. Initially, we compared the first calibration cost with the baseline implementation of Llama2-7B-Chat fine-tuned with LoRA, using the same number of sampling iterations. Our framework retains 72% of the baseline's inference speed while achieving superior performance, as demonstrated in Figure

[3] https://github.com/hiyouga/LLaMA-Factory

3. This empirically validated trade-off highlights our framework's ability to deliver significant performance improvements with minimal latency increase.

Additionally, we conducted a comparative analysis with one of the state-of-the-art frameworks, Self-RAG (Asai et al., 2023). The experimental results, detailed in Table 9, provide significant insights. Using the Llama2-7B model as per Self-RAG's setup, our framework, with 10 retrieved documents, shows notable improvements over Self-RAG. Specifically, our SGIC framework enhances Exact Match (EM) scores by 28.5 on the HotpotQA dataset and 17.7 on the Natural Questions (NQ) dataset.

A key strength of our framework is its enhanced computational efficiency, achieved through batch processing of document uncertainties. On the HotpotQA dataset, SGIC achieves 210% of Self-RAG's inference speed. Although this speed advantage decreases to 90% on the NQ dataset, this reduction is due to Self-RAG's shorter generation pattern on the simpler single-hop NQ dataset, which reduces time spent on retrieval, critique, and generation phases. Furthermore, Self-RAG requires full parameter fine-tuning and relies on GPT-4 to generate extensive critique training data, increasing training overhead. In contrast, our framework eliminates the need for external knowledge and accelerates training using Parameter-Efficient Fine-Tuning (PEFT) methods like LoRA, achieving superior results compared to Self-RAG.

A.4 Theoretical Analysis

In this section, we delve into the theoretical underpinnings of why our method is effective. The Probability Ranking Principle (PRP), as introduced by Robertson (1977), posits that ranking documents based on their probability of relevance ensures optimal performance in ad-hoc retrieval tasks. This principle relies on models that provide well-calibrated probability estimates. Recent studies have shown that integrating uncertainty into ranking processes can significantly boost learning and performance in information retrieval scenarios (Yu et al., 2024). Our method adheres to the PRP by enhancing model calibration through robust uncertainty estimation.

While Duan et al. (2023) points out challenges such as generative inequalities in uncertainty quantification, our research emphasizes how uncertainty can be harnessed to improve retrieval models. By fine-tuning models to effectively utilize uncertainty scores, we achieve outcomes that align with the PRP, thereby demonstrating the reliability of our approach. Furthermore, employing advanced estimation techniques has the potential to further elevate performance, highlighting the robustness and versatility of our method. Beyond retrieval, our approach is applicable to long document challenges, such as translation and summarization, showcasing its adaptability across a wide range of tasks.

Method	R_{10}@2	R_{10}@5
Baseline	42.9	70.1
1st Time Calibration	**49.8**	**75.4**

Table 10: The $R_{10}@k$ scores of ranking the candidate documents from largest to smallest in terms of the weight of the attention distribution in Llama2-7B-Chat on the HotpotQA dataset.

A.5 Discussion of Uncertainty Estimation

In this work, our primary focus is on leveraging uncertainty scores to identify the best-calibrated answers by comparing these scores, rather than solely concentrating on the precision of uncertainty estimation. Nonetheless, we also assess the precision of uncertainty estimation for both our method and the baseline using a consistent estimation framework. We utilize the AUROC score as a metric and evaluate it on the HotpotQA test set. Following the 1st time calibration, our approach achieves an AUROC score of 68.4, surpassing the baseline's score of 65.5. This indicates that our method not only excels in selecting well-calibrated answers but also enhances uncertainty estimation, even though this is not our primary objective.

Write a short answer directly ...

<KEY> means this document contains key information of the question.

(Other docuemtns) ...

Document [6]: <KEY> Yellowcraig, less commonly known as Broad Sands Bay, is a coastal area ...

Document [7]: <KEY> Dirleton Castle is a medieval fortress in the village of Dirleton, East ...

Table 11: The input format of adding <KEY> tags.

Explain your reasoning and give a final answer for the given question. The initial

Question: Josh decides to try flipping a house. He buys a house for $80,000 and ...

Reasoning Steps:

1. The house was worth $80,000 before the repairs.(Uncertainty Score: 57)
2. The repairs cost $50,000.(Uncertainty Score: 53)
3. So the house is now worth $80,000 + $50,000 = $«80000+ ... (Uncertainty Score: 4)
4. The repairs increased the value of the house by 150%.(Uncertainty Score: 36)
5. So the value of the house increased by 150% of $50,000 = $«150 ... (Uncertainty Score: 100)
6. So the house is now worth $130,000 - $75,000 = $«130000-75000= ... (Uncertainty Score: 0)
7. So Josh made a profit of $55,000. (Uncertainty Score: 49)

Final Answer:

Table 12: The input format of fine-tuning our method on the GSM8K dataset.

Example 1

Write a short answer directly without any explanation or introduction for the ...

(Documents and their uncertainty scores) ...

Question: A medieval fortress in Dirleton, East Lothian, Scotland borders on the south side of ...

The initial answer is:

(Previous Rounds Generated Answer)

Round 2: Firth of Forth (Uncertainty Score: 90)

The correct answer is: Yellowcraig

Example 2

Write a short answer directly without any explanation or introduction for the...

(Documents and their uncertainty scores) ...

Question: The director of the romantic comedy “Big Stone Gap” is based in what New York city?

The initial answer is:

(Previous Rounds Generated Answer)

Round 3: Greenwich Village (Uncertainty Score: 66)

The correct answer is: Greenwich Village, New York City

Table 13: Two examples of how our proposed framework corrects the answer. Red means the answer is wrong, while green indicates the answer is correct. Blue indicates that the answer contains partially correct answer.