

A Related Works

A.1 Uncertainty Calibration in Language Models

Traditional calibration methods rely on token-level log probabilities [10], but modern LLMs generate text autoregressively by multiplying conditional probabilities [2]. Estimating semantic-level probabilities would require marginalization over all possible sequences, which is computationally intractable. As a result, token-level probabilities often fail to provide reliable confidence estimates for long-form text generation.

Prompt-based approaches aim to address this problem by eliciting verbalized confidence scores [46, 53]. For example, a model can be prompted with: “Express your confidence as a number between 0 and 100.” If it responds with “90”, this value is interpreted as its confidence level. However, LLMs often exhibit overconfidence in zero-shot settings, resulting in poorly calibrated outputs [36]. Although RAG can mitigate this issue, when the retrieved context is unreliable, LLM may still demonstrate overconfidence, leading to misleading conclusions. Addressing this challenge remains essential for improving LLM reliability in complex decision-making tasks.

A.2 Methods for Enhancing RAG Robustness

Recent advancements in reranking for RAG have largely focused on enhancing the relevance of retrieved documents with respect to the input query. For example, LLM-based rerankers leverage semantic representations to reorder documents based on their relevance [45], while cross-encoder-based rerankers jointly encode query-document pairs to model their interaction more precisely [32]. These approaches are highly effective in improving retrieval relevance and downstream QA performance. However, they are fundamentally designed to rank documents by relevance, not to assess how the retrieved information influences the correctness of the final user decision based on the LLM-generated answer. Thus, the resulting scores, although often normalized between 0 and 1, are not calibrated probabilities of correctness and cannot be directly used for decision calibration.

Similarly, Self-RAG [3] introduces the notion of utility scores for retrieved documents to identify potentially helpful content. While this provides a signal for filtering noisy documents, the utility score reflects plausibility rather than empirical correctness. As such, these scores are neither optimized for nor aligned with standard calibration metrics such as ECE, NLL, or Brier Score.

In contrast, our approach directly addresses this gap by training a forecasting function to output calibrated confidence scores that reflect the actual correctness of decisions made by a surrogate user model. We explicitly supervise the forecasting function using binary labels that indicate whether the model’s final prediction is correct, and optimize this function using strictly proper scoring rules. This ensures that the predicted confidence scores match the empirical likelihood of correctness, thus enabling true decision calibration rather than merely relevance estimation.

This fundamental difference in supervision **signal** (relevance vs. correctness) and **objective** (ranking vs. calibration) delineates the core novelty of our work from prior reranking-based approaches. By aligning the model’s confidence estimates with empirical decision accuracy, our method offers a principled and interpretable framework for improving trustworthiness in RAG systems.

B Experimental details

Our implementation builds on key libraries such as PyTorch 2.1.2 [37], Hugging Face Transformers 4.45.1 [51], and PEFT 0.7.1,¹ providing a robust foundation for experimentation. We employ the Llama-3.1-8B-Instruct model, an open-source multilingual LLM available on Hugging Face.² Our experiments are conducted on NVIDIA RTX 3090 and RTX A6000 GPUs. Additionally, we utilize the official facebookresearch-contriever repository³ and the elastic-research-bm25 repository⁴ for our retrieval model. We also use MedMCT

¹<https://github.com/huggingface/peft>

²<https://huggingface.co/meta-llama/Meta-Llama-3.1-8B-Instruct>

³<https://github.com/facebookresearch/contriever>

⁴<https://www.elastic.co/>

based on the MedRAG framework.⁵ For training calibration tuning baselines, we reference the calibration-tuning repository.⁶

B.1 Datasets

Train Datasets SQuAD [39, 38] is a reading comprehension dataset sourced from Wikipedia, containing questions answered by text spans from the articles. WikiQA [55] is a question-sentence pair dataset from Wikipedia, designed for open-domain question answering and includes unanswerable questions for research on answer triggering. TriviaQA [19] is a reading comprehension dataset with questions authored by trivia enthusiasts, paired with evidence documents from Wikipedia and other web sources. We randomly sampled 10,000 data points each from TriviaQA and SQuAD2.0, and collected all 873 training samples from WikiQA. In addition, we incorporated non-overlapping samples from SQuAD1.0, resulting in a combined training dataset of 61,886 examples after deduplication. For the validation set, we gathered 2,000 samples each from TriviaQA and SQuAD, along with 126 samples from WikiQA, and added non-overlapping samples from SQuAD1.0, yielding a total of 12,643 validation data points. All null values were removed prior to finalization. We downloaded all these datasets in Hugging Face datasets⁷.

For the construction of the labeled dataset \mathcal{S} used to train the forecasting function of CalibRAG, we sample a temperature $t \sim \text{Uniform}[1.0, 2.0]$ for each query q and retrieved document d . For each triplet (t, q, d) , we perform user decoding 10 times and assign a soft label b indicating the ratio of generated answers that contain the ground truth. The final dataset \mathcal{S} thus consists of tuples in the form (t, q, d, b) . **The dataset will be made publicly available upon the publication of this work.**

Evaluation Datasets For zero-shot evaluation, we employ several datasets covering diverse domains and question types. HotpotQA [56] is a multi-hop question-answering dataset requiring reasoning across multiple supporting documents from Wikipedia to find answers, emphasizing a more complex retrieval and reasoning process. WebQA [7] is an open-domain question-answering dataset consisting of natural, conversational questions paired with web documents, targeting real-world, context-rich scenarios. Natural Questions (NQ) [25] is another large-scale question-answering dataset, designed to answer questions based on Wikipedia articles, containing both long-form and short-form answers. These datasets are used without additional training, providing a robust evaluation of the generalization capabilities of CalibRAG across different domains and question types.

We also evaluate domain-specific datasets, including BioASQ [22], a biomedical QA dataset containing factoid, list, and yes/no questions derived from PubMed articles, as well as Medical Information Retrieval-Augmented Generation Evaluation (MIRAGE) [52] and a textbook corpus.

B.2 Hyperparameters

Table 2: Hyperparameters for LLM Training

Base Model Hyperparameters		LoRA Hyperparameters	
Hyperparameter	Value	Hyperparameter	Value
Learning Rate	$\{10^{-4}, 10^{-5}\}$	LoRA Rank	8
Batch Size	$\{1, 4\}$	LoRA Alpha	16
Max Steps	10,000	LoRA Dropout	0.1
Optimizer	AdamW		
Dropout Rate	0.0		
Gradient Accumulation Steps	[1, 4]		
Weight Decay	0.01		
Gradient Clipping	1.0		
Warmup Steps	500		
Scheduler	Linear		

⁵<https://github.com/Teddy-XiongGZ/MedRAG>

⁶<https://github.com/activatededgeek/calibration-tuning>

⁷<https://github.com/huggingface/datasets>

931 Table 2 outlines the hyperparameters used for training the base model and LoRA, including key
 932 parameters such as learning rate, batch size, and LoRA-specific settings like rank and alpha.

933 B.3 Evaluation metrics

934 To evaluate long-form text, we utilized `gpt-4o-mini` to compare the ground-truth answers with
 935 the predicted answers in all cases. Based on this comparison, we labeled each instance as correct or
 936 incorrect accordingly.

937 B.3.1 Calibration metrics

- 938 • **Expected Calibration Error** [ECE; 35]:

$$\text{ECE} = \sum_{m=1}^M \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$$

939 where B_m is the set of predictions in bin m , $\text{acc}(B_m)$ is the accuracy, and $\text{conf}(B_m)$ is
 940 the average confidence of the predictions in that bin. ECE measures how well the model’s
 941 predicted probabilities are calibrated.

- 942 • **Brier Score** [BS; 6]:

$$\text{BS} = \frac{1}{N} \sum_{i=1}^N (f_i - y_i)^2$$

943 where f_i is the predicted probability and y_i is the true label. BS combines both the accuracy
 944 and confidence of the predictions, penalizing overconfident and underconfident predictions.

- 945 • **Negative Log Likelihood (NLL)**:

$$\text{NLL} = -\frac{1}{N} \sum_{i=1}^N \log p(y_i | x_i)$$

946 where $p(y_i | x_i)$ is the probability assigned to the correct class y_i given input x_i . NLL
 947 evaluates the model’s probabilistic predictions and lower values indicate better calibration.

948 B.4 CalibRAG Details

949 **Feature extraction details.** To extract features for the forecasting function, we use the hidden
 950 state of the last token from the second-to-last layer of the LLM \mathcal{M} . We empirically found that
 951 representations from the second-to-last layer yield better calibration performance than those from
 952 other layers. This hidden state serves as the input representation $f_{\text{feat}}(q, d)$ for the classifier.

953 **Positional encoding details.** We use a Fourier positional encoding with $N = 6$ frequency com-
 954 ponents to encode the temperature parameter t . This encoding covers the range $t \in [1.0, 2.0]$, and
 955 during training data construction, we sample t uniformly from this range to simulate diverse user
 956 behaviors.

957 C Examples of query reformulations

958 In CalibRAG, the initial query is generated to simulate how a human decision-maker might pose a
 959 simple query based on the input. For example, a decision-maker faced with a problem such as "Is a
 960 tomato a fruit or a vegetable?" might craft a straightforward query like "Classification of tomatoes"
 961 to query a language model. Using this setup, we employed an LLM generator to create simple yet
 962 relevant queries and retrieved documents based on these queries. If the retrieved documents were
 963 insufficiently informative, the query was reformulated in Stage 3. This reformulation emphasized
 964 key terms to refine the query and improve the quality of retrieved documents. The specific prompt
 965 used for this process is detailed in App. G.

966 To help readers understand the transformation from the initial query to its reformulated version,
 967 Table 3 provides examples that illustrate how queries evolve during the refinement process, offering
 968 practical insights into the mechanism.

Table 3: Examples of Query Reformulation

Case	Original Query	Reformulated Query
1	Write a paragraph about the effect of TRH on myocardial contractility.	Write a paragraph about the effect of Thyrotropin-Releasing Hormone (TRH) on myocardial contractility.
2	Write a paragraph about the clinical trials for off-label drugs in neonates as cited in the literature.	Write a paragraph about clinical trials for off-label drug use in neonates as reported in the medical literature.
3	Write a paragraph about the current representatives from Colorado.	Write a paragraph about the current representatives from the state of "Colorado" in the United States.
4	Write a paragraph about the current minister of local government in Zimbabwe and their role within the government.	Write a paragraph about the current Minister of Local Government and Public Works in Zimbabwe and their role within the government.

Table 4: Effect of Threshold Selection on Performance. Experiments on the BioASQ dataset show how increasing ϵ affects accuracy and calibration metrics.

ϵ	AUROC	ACC	ECE	BS
0.0	71.21 \pm 0.83	35.03 \pm 0.14	0.2500 \pm 0.01	0.2900 \pm 0.01
0.4	76.15 \pm 1.50	35.05 \pm 0.25	0.2608 \pm 0.00	0.2830 \pm 0.00
0.5	76.50 \pm 4.98	35.98 \pm 0.38	0.2667 \pm 0.00	0.2779 \pm 0.01
0.6	77.20 \pm 4.10	36.50 \pm 0.45	0.2707 \pm 0.00	0.2800 \pm 0.01

D Additional experiments

D.1 Analysis of ϵ

In our experiments, ϵ was set as a balanced choice to manage the trade-off between accuracy and calibration error. As shown in Table 4, increasing ϵ results in retrieving a larger number of new queries, incorporating more relevant information, and thereby improving accuracy. However, this increase can potentially lead to higher calibration errors. Specifically, while better retrieval enhanced prediction accuracy, the confidence scores for these predictions only increased marginally. This mismatch between improved accuracy and relatively low confidence resulted in underconfident predictions, which contributed to a slight increase in calibration error.

To assess the impact of different ϵ values on model performance, we conducted experiments on the BioASQ dataset. Based on these observations, we selected $\epsilon = 0.5$ as a reasonable compromise to balance accuracy improvements with calibration reliability.

D.2 Evaluation on BEIR Benchmark

To provide a more comprehensive evaluation, we conducted experiments using two datasets from the BEIR benchmark: SciFact and TREC-COVID. These evaluations aim to validate the effectiveness of CalibRAG beyond its primary focus on well-calibrated decision-making, which predicts the probability of a correct decision when a user relies on the generated guidance to solve a given problem. While CalibRAG is not specifically designed as a reranking method to optimize retrieval performance, it inherently supports both calibration and retrieval.

For the experiments, we followed the standard retrieval pipeline, retrieving documents using BM25 and reranking the top-100 results. We compared CalibRAG with the Cross-Encoder baseline, and the results, presented in Table 5, demonstrate that CalibRAG consistently outperforms the Cross-

Table 5: Evaluation results on TREC-COVID and SciFact datasets, a subset of the BEIR benchmark. The evaluation metric is Normalized Discounted Cumulative Gain (NDCG@K).

Model	Dataset	NDCG@5	NDCG@10
Cross-Encoder	TREC-COVID	0.7655	0.7576
	SciFact	0.6668	0.6914
CalibRAG	TREC-COVID	0.7863	0.7660
	SciFact	0.6872	0.7114

Table 6: Results of Verbalized Confidence Fine-Tune Evaluation on the MMLU Dataset using Llama-3.1-8B-Instruct. Evaluation metrics are ACC and ECE.

Case	ACC	ECE
Continuous-Number	43.63	0.3190
Discrete-Number	44.96	0.1605
Linguistic	45.03	0.1585

991 Encoder. These findings validate that CalibRAG not only enables well-calibrated decision-making
 992 but also enhances retrieval performance, reinforcing its utility in relevant scenarios.

993 D.3 Analysis of Verbalized Confidence Representations

994 CalibRAG does not rely on linguistic or numerical confidence in its primary approach. Instead, it
 995 provides confidence scores based on probability predictions generated by the forecasting function.
 996 Verbalized confidence, however, was used as a baseline in the comparative models. Verbalized
 997 confidence is typically expressed as a continuous number within the range [0, 100] Tian et al. [46]
 998 and Xiong et al. [53], but LLMs often struggle to interpret these numerical values precisely.

999 To address this limitation, alternative representations were explored in the baselines: (1) linguistic
 1000 expressions (e.g., “likely”), and (2) discrete numerical values ranging from 0 to 10. These ap-
 1001 proaches were termed Linguistic and Number, respectively, with detailed prompt designs provided
 1002 in Appendix E.

1003 To further analyze verbalized confidence, we conducted experiments on the MMLU dataset using the
 1004 Llama-3-8B model. We evaluated the effectiveness of three confidence representations: continuous
 1005 number, discrete number, and linguistic. As shown in Table 6, both discrete number and linguistic
 1006 representations outperformed the continuous number baseline. Linguistic confidence, in particular,
 1007 addressed the limitations of the model’s understanding of numerical relationships and improved
 1008 calibration.

1009 D.4 Ablation on user model U

1010 We additionally conduct an ablation evaluation on various user models U , considering that human
 1011 users may make different decisions depending on their knowledge background in real-world sce-
 1012 narios. We evaluate the performance of CalibRAG and baseline methods on the NQ and WebQA
 1013 datasets using two retriever models, BM25 and Contriever. For this, we compare the performance
 1014 of Phi-4 [1] and DeepSeek-Distill [11], which represent state-of-the-art user models.

1015 As shown in Fig. 6 and Fig. 7, our results demonstrate that CalibRAG consistently achieves better
 1016 accuracy and calibration error across different user models compared to other baselines.

1017 D.5 Ablation on Fine-Tuning for Reranking Baselines

1018 To ensure a fair comparison between CalibRAG and the reranking baseline, we also evaluated a fine-
 1019 tuned reranker model which fine-tuned using our synthetic datasets. However, as discussed in the

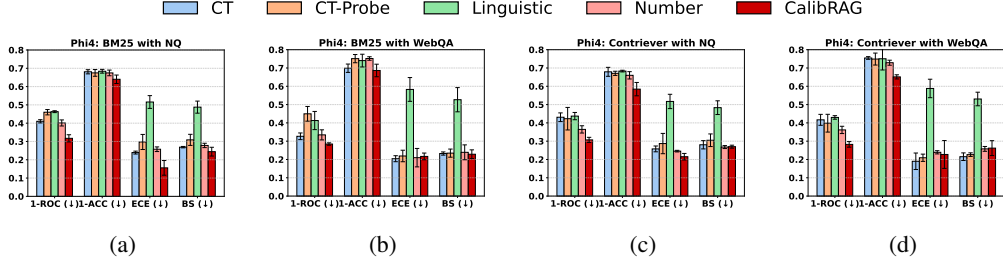


Figure 6: Evaluation results of the baselines and CalibRAG utilizing two retriever models: BM25 (a, b) and Contriever (c, d) on NQ (a, c) and WebQA (b, d). Here, we utilize *Phi-4* [1] as our user model U . We report four metrics—1-AUROC, 1-ACC, ECE, and Brier Score—where lower values indicate better performance.

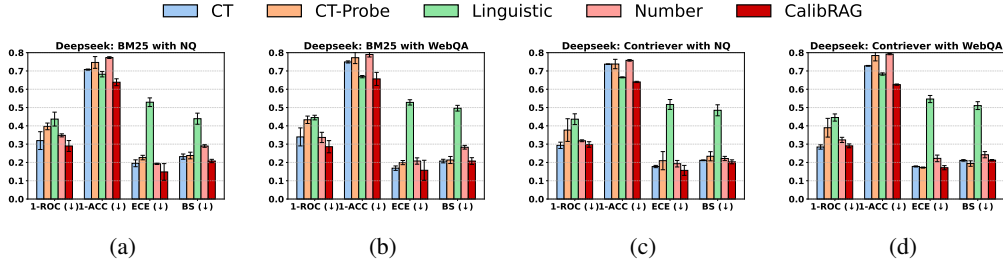


Figure 7: Evaluation results of the baselines and CalibRAG utilizing two retriever models: BM25 (a, b) and Contriever (c, d) on NQ (a, c) and WebQA (b, d). Here, we utilize *DeepSeek-Distill* [11] as our user model U . We report four metrics—1-AUROC, 1-ACC, ECE, and Brier Score—where lower values indicate better performance.

Table 7: Comparison of fine-tuned RAG reranking methods using our synthetic training data on HotpotQA.

Methods	AUROC (↑)	ACC (↑)	ECE (↓)	BS (↓)
Cross-encoder	60.74	34.98	0.477	0.477
Cross-encoder (Fine-tuned)	61.55	32.54	0.008	2.555
CalibRAG	72.47	42.37	0.106	0.206

Table 8: Evaluation metrics of CalibRAG without reranking on WebQA using BM25

Methods	AUROC (↑)	ACC (↑)	ECE (↓)	BS (↓)
Number	69.38 ± 2.84	36.04 ± 0.50	0.1931 ± 0.0131	0.2293 ± 0.0102
CalibRAG w/o Rerank	75.73 ± 0.00	41.99 ± 0.03	0.0780 ± 0.0312	0.1981 ± 0.0025

§ 3.2, training was challenging due to the difficulty of feature extraction without using an embedding model to generate the guidance variable z . And this difficulty let fine-tuned model underfit to the training dataset. As shown in Table 7, the reranker model underperforms compared to the zero-shot setting. Therefore, in the **Comparison with reranking and robust RAG baselines** experiments in Sec. 4.1, we evaluated the CalibRAG model alongside zero-shot reranker models.

D.6 Ablation on CalibRAG without reranking

To isolate the effect of reranking in our confidence calibration framework, we evaluate CalibRAG without using any reranking, where the model directly uses retrieved contexts without reordering them based on predicted confidence. As shown in Table 8, even without reranking, CalibRAG substantially outperforms the Number baseline in both accuracy and calibration metrics. These results indicate that the learned calibration itself, without requiring reranking, still provides significant benefit, demonstrating the robustness of CalibRAG’s alignment mechanism.

Table 9: Evaluation metrics of Number + Rerank and CalibRAG on WebQA

Retriever	Methods	AUROC (\uparrow)	ACC (\uparrow)	ECE (\downarrow)	BS (\downarrow)
BM25	Number + Rerank	75.06 \pm 0.00	42.42 \pm 0.01	0.2075 \pm 0.0167	0.2397 \pm 0.0109
	CalibRAG	77.29 \pm 0.42	43.77 \pm 0.54	0.0567 \pm 0.0332	0.1983 \pm 0.0045
Contriever	Number + Rerank	76.84 \pm 0.00	43.08 \pm 0.00	0.2088 \pm 0.0127	0.2390 \pm 0.0083
	CalibRAG	76.24 \pm 0.37	44.19 \pm 2.60	0.0997 \pm 0.0122	0.2095 \pm 0.0062

Table 10: Comparison of zero-shot evaluation of calibration baselines on NQ and WebQA datasets using BM25 (lexical) retrieval. Results are averaged over three random seeds.

Methods	NQ				WebQA			
	AUROC	ACC	ECE	BS	AUROC	ACC	ECE	BS
CT-LoRA	73.51 \pm 1.65	37.70 \pm 0.28	0.2479 \pm 0.024	0.2709 \pm 0.0133	74.36 \pm 1.17	38.09 \pm 0.28	0.2487 \pm 0.0303	0.2681 \pm 0.0200
CT-probe	60.92 \pm 0.94	37.59 \pm 3.03	0.3490 \pm 0.0236	0.3536 \pm 0.0223	58.52 \pm 2.51	37.75 \pm 4.39	0.3491 \pm 0.0329	0.3539 \pm 0.0332
Linguistic-LoRA	57.12 \pm 4.35	39.42 \pm 0.94	0.4529 \pm 0.0223	0.4362 \pm 0.0284	56.44 \pm 1.93	40.58 \pm 1.18	0.4536 \pm 0.0071	0.4385 \pm 0.0091
Number-LoRA	67.48 \pm 1.42	34.38 \pm 0.71	0.1922 \pm 0.0165	0.2294 \pm 0.0076	69.38 \pm 2.84	36.04 \pm 0.50	0.1931 \pm 0.0131	0.2293 \pm 0.0102
CalibRAG	77.29 \pm 0.12	<u>42.66</u> \pm 0.97	0.0600 \pm 0.0039	0.1983 \pm 0.0017	77.29 \pm 0.42	43.77 \pm 0.54	0.0567 \pm 0.0332	0.1983 \pm 0.0045
CalibRAG-multi	<u>76.73</u> \pm 0.22	46.16 \pm 0.05	<u>0.1397</u> \pm 0.0022	<u>0.2138</u> \pm 0.0016	<u>76.40</u> \pm 0.28	45.84 \pm 0.25	<u>0.1372</u> \pm 0.0007	<u>0.2175</u> \pm 0.0008

Table 11: Comparison of zero-shot evaluation of calibration baselines on NQ and WebQA datasets using Contriever (dense) retrieval. Results are averaged over three random seeds.

Methods	NQ				WebQA			
	AUROC	ACC	ECE	BS	AUROC	ACC	ECE	BS
CT-LoRA	69.89 \pm 4.94	39.93 \pm 1.26	0.2800 \pm 0.0585	0.3008 \pm 0.0435	69.81 \pm 6.82	37.83 \pm 1.25	0.2646 \pm 0.0510	0.2860 \pm 0.0394
CT-probe	63.84 \pm 6.14	37.92 \pm 2.80	0.3225 \pm 0.0634	0.3343 \pm 0.0498	62.65 \pm 8.10	36.43 \pm 4.03	0.3072 \pm 0.0670	0.3180 \pm 0.0565
Linguistic-LoRA	57.05 \pm 3.91	41.50 \pm 0.37	0.4368 \pm 0.0267	0.4252 \pm 0.0290	56.30 \pm 2.70	39.76 \pm 0.77	0.4657 \pm 0.0124	0.4477 \pm 0.0162
Number-LoRA	71.16 \pm 0.61	35.99 \pm 0.54	0.1827 \pm 0.0124	0.2214 \pm 0.0016	<u>73.47</u> \pm 1.01	35.61 \pm 0.12	<u>0.1754</u> \pm 0.0124	<u>0.2141</u> \pm 0.0040
CalibRAG	73.89 \pm 1.50	<u>46.55</u> \pm 2.45	0.0312 \pm 0.0073	0.2074 \pm 0.0062	76.24 \pm 0.37	<u>44.19</u> \pm 2.60	0.0970 \pm 0.0122	0.2095 \pm 0.0062
CalibRAG-multi	<u>72.73</u> \pm 0.08	49.42 \pm 0.07	<u>0.1656</u> \pm 0.0019	0.2375 \pm 0.0013	72.95 \pm 0.08	46.78 \pm 0.02	0.1901 \pm 0.0012	0.2488 \pm 0.0009

D.7 Using Uncertainty baseline confidence scores for reranking

In this section, we investigate the effectiveness of using uncertainty baseline confidence scores for reranking in the RAG pipeline. As described in the main paper, these confidence scores are derived from verbalized scalar predictions generated by the LLM, typically representing values from 0 to 100.

While such scalar confidence values can be used to rerank retrieved documents, this approach incurs significant computational overhead. Specifically, the Number baseline requires generating full guidance z for every (q, d) pair before estimating confidence, as the model conditions on both the query and document to generate scalar outputs. In contrast, CalibRAG directly estimates confidence from the (q, d) pair using a lightweight forecasting function $f(q, d)$, thus avoiding this expensive intermediate generation.

Despite this additional cost, we perform an ablation to compare the reranking performance of Number-based confidence scores versus CalibRAG. As shown in Table 9, CalibRAG consistently outperforms the baseline across both BM25 and Contriever retrievers on the WebQA dataset.

These results demonstrate that CalibRAG not only provides better-calibrated decisions but does so more efficiently without requiring guidance generation for every document candidate. This highlights the dual advantage of CalibRAG in both performance and computational cost.

Table 12: Comparison of zero-shot evaluation of calibration baselines on **BioASQ-Y/N**, **MMLU-Med**, and **PubMedQA** datasets. Results are averaged over three random seeds.

Methods	Dataset	AUROC	ACC	ECE	BS
<i>CT-LoRA</i>	BioASQ-Y/N	65.20 \pm 2.32	54.31 \pm 0.73	0.5167 \pm 0.012	0.5099 \pm 0.0146
	MMLU-Med	66.94 \pm 0.68	47.20 \pm 1.52	0.4293 \pm 0.0088	0.4262 \pm 0.0084
	PubMedQA	56.67 \pm 3.16	43.80 \pm 0.91	0.4307 \pm 0.0099	0.4300 \pm 0.0094
<i>CT-probe</i>	BioASQ-Y/N	59.73 \pm 4.27	57.98 \pm 1.45	0.5664 \pm 0.009	0.5630 \pm 0.0094
	MMLU-Med	55.39 \pm 3.24	49.49 \pm 5.00	0.4771 \pm 0.0384	0.4758 \pm 0.0375
	PubMedQA	54.56 \pm 0.61	46.60 \pm 1.88	0.4506 \pm 0.012	0.4510 \pm 0.0121
<i>Linguistic-LoRA</i>	BioASQ-Y/N	48.24 \pm 2.26	57.82 \pm 0.50	0.3193 \pm 0.0027	0.3464 \pm 0.0030
	MMLU-Med	51.30 \pm 0.93	55.43 \pm 0.94	0.3262 \pm 0.0078	0.3544 \pm 0.0049
	PubMedQA	49.13 \pm 0.79	47.13 \pm 2.25	0.4047 \pm 0.0336	0.4225 \pm 0.021
<i>Number-LoRA</i>	BioASQ-Y/N	52.43 \pm 2.19	53.72 \pm 1.69	0.4664 \pm 0.0355	0.4659 \pm 0.0332
	MMLU-Med	53.47 \pm 2.54	41.44 \pm 1.01	0.3394 \pm 0.0168	0.3541 \pm 0.0135
	PubMedQA	50.34 \pm 0.25	43.60 \pm 0.59	0.3866 \pm 0.0029	0.3954 \pm 0.0032
<i>CalibRAG</i>	BioASQ-Y/N	66.66 \pm 1.34	70.82 \pm 3.34	0.2414 \pm 0.0427	0.2606 \pm 0.0386
	MMLU-Med	68.93 \pm 1.32	57.20 \pm 0.21	0.0625 \pm 0.0653	0.2226 \pm 0.0112
	PubMedQA	66.57 \pm 2.00	62.20 \pm 3.53	0.2250 \pm 0.0353	0.2691 \pm 0.0072

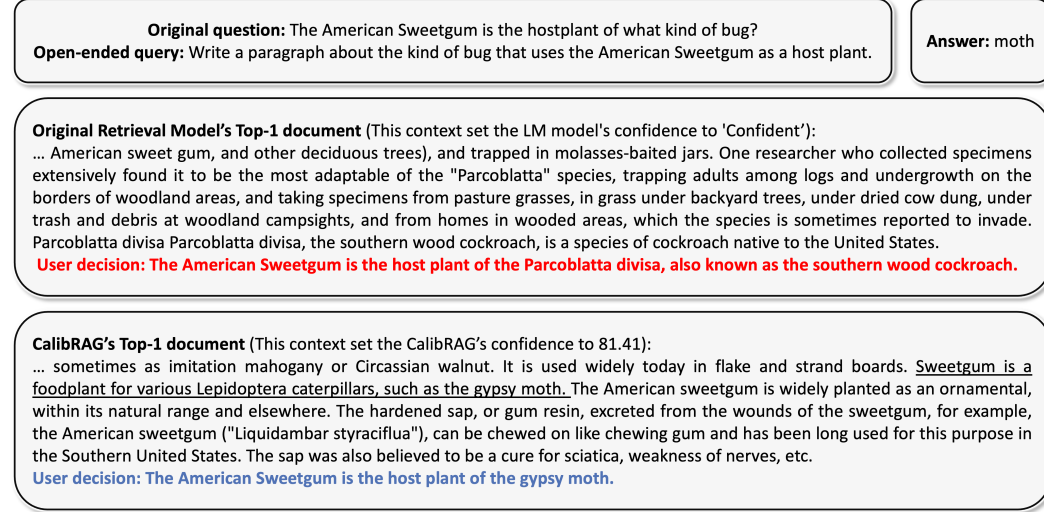


Figure 8: Qualitative comparison of original retrieval model from CalibRAG.

1049 D.8 Full numerical results for main experiments

1050 Table 10, Table 11 and Table 12 present the complete numerical results from the primary experi-
 1051 ments. For the *Base* model, we utilized a pretrained model, sampling sentences across three different
 1052 seeds. For the other methods, training was conducted across three random seeds to ensure robust
 1053 evaluation. We highlight the best-performing value in **bold** and the second-best in underline.

1054 D.9 Qualitative Results

1055 While quantitative metrics alone may not fully capture all the benefits of CalibRAG, we present
 1056 examples highlighting its ability to identify relevant documents and assign calibrated confidence
 1057 scores. Given the query "Write a paragraph about the kind of bug that uses the American Sweetgum
 1058 as a host plant.", the base retriever focuses only on the keyword "American Sweetgum," retrieving

loosely relevant content and marking its confidence as ‘Confident’ (10/11) as illustrated in Fig. 8. This led to the incorrect conclusion that the sweetgum is the host plant of *Parcoblatta divisa*, the southern wood cockroach. In contrast, CalibRAG captures the full context, retrieving documents specifically about the gypsy moth, which uses the sweetgum as a host plant, and correctly assigns a confidence level of 81.41. This demonstrates the capability of CalibRAG to find a relevant document and assign a confidence level correlated with the accuracy of the downstream surrogate user. Additional examples can be found in App. E.

E Data Examples

Fig. 9 shows the top 20 examples of queries and their corresponding labels. The full set of data examples will be released upon publication of the paper. Fig. 9 shows that the ranking of the retrieved documents is not correlated with the accuracy of the user decision. As seen in this example, the top-ranked document is not helpful for the user model in decision-making, whereas the second-ranked document provides information that can lead the user model to make a correct decision. This illustrates the importance of CalibRAG’s forecasting function f in effectively modeling the probability that a decision made using document d is correct, emphasizing the need for reranking documents based on this modeling.

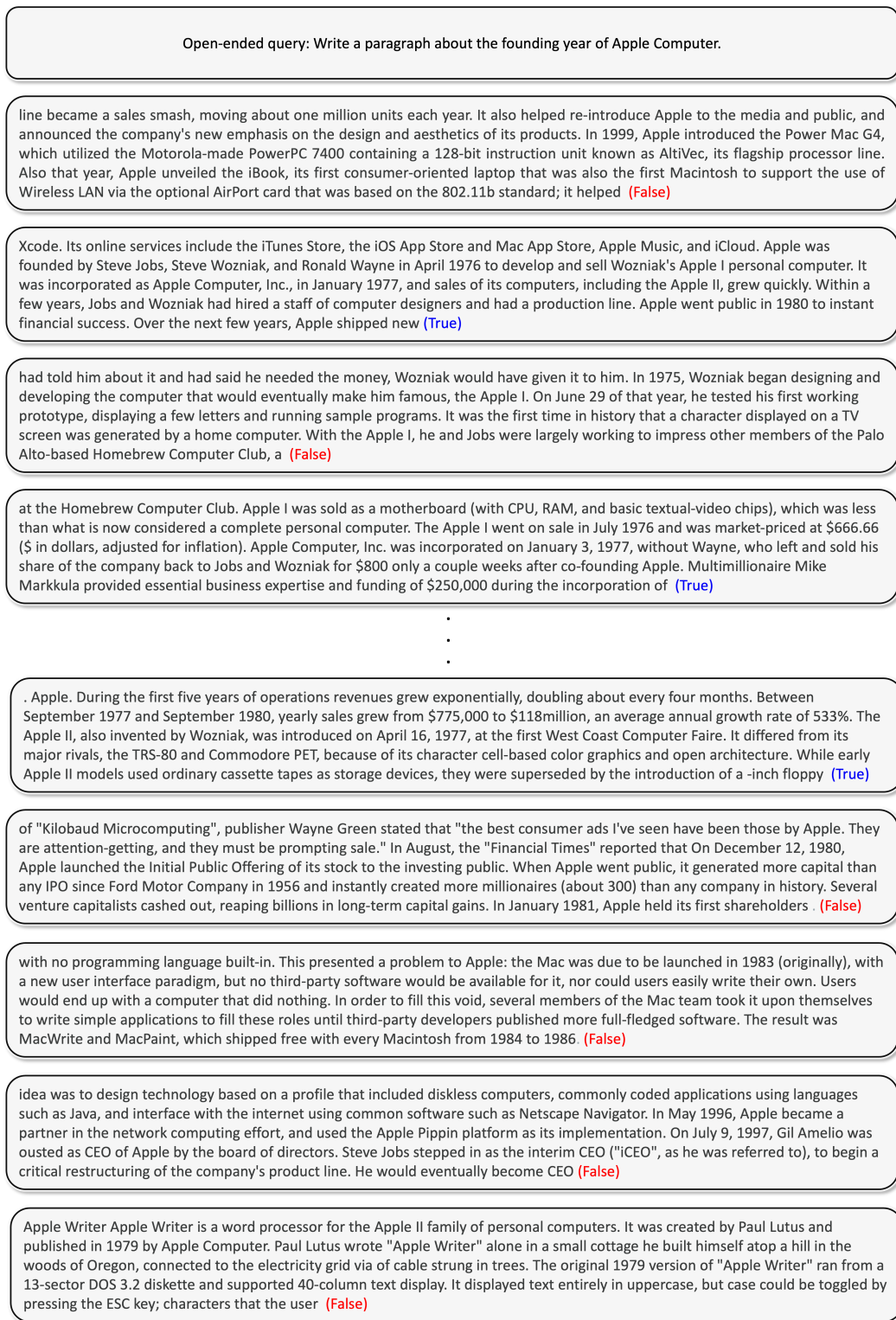


Figure 9: Top-20 retrieved document examples.

1075 F Qualitative Examples

1076 Here, we present additional qualitative examples for comparison with other baselines. In Fig. 10,
 1077 Fig. 11, Fig. 12, and Fig. 13, the examples demonstrate that while the baselines retrieve documents
 1078 that provide incorrect answers to the queries, they still assign high confidence to the retrieved docu-
 1079 ments. In contrast, CalibRAG effectively reranks and retrieves documents that are highly relevant to
 1080 the decision problem x , allowing us to confirm that the guidance generated from these retrieved docu-
 1081 ments is well-predicted to be helpful for decision-making. Additionally, we can confirm that when
 1082 the document with the highest rank does not aid in decision-making for x , CalibRAG successfully
 1083 assigns a lower confidence level, helping to prevent the user from over-relying on the guidance.

<p>Original question: When was the American lawyer, lobbyist and political consultant who was a senior member of the presidential campaign of Donald Trump born?</p> <p>Open-ended query: Write a paragraph about the American lawyer, lobbyist, and political consultant who was a senior member of Donald Trump's presidential campaign, including details about when he was born.</p>	<p>Answer: April 1, 1949</p>
<p>Original Retrieval Model's Top-1 document (This context set the LM model's confidence to 'Certain'): ... Page testified that he did not "directly" express support for lifting the sanctions during the meeting with Baranov, but that he might have mentioned the proposed Rosneft transaction. Carter Page Carter William Page (born June 3, 1971) is an American petroleum industry consultant and a former foreign-policy adviser to Donald Trump during his 2016 Presidential election campaign. User decision: June 3, 1971.</p>	
<p>CalibRAG's Top-1 document (This context set the CalibRAG's confidence to 83.93): <u>Paul Manafort Paul John Manafort Jr. (born April 1, 1949) is an American lobbyist, political consultant, lawyer, and felon. A Republican, he joined Donald Trump's presidential campaign team in March 2016, and was campaign chairman from June to August 2016. Manafort was an adviser to the U.S. presidential campaigns of Republicans Gerald Ford, Ronald Reagan, George H. W. Bush, and Bob Dole. In 1980, he co-founded the Washington, D.C.-based lobbying firm Black, Manafort & Stone, along with principals Charles R. Black Jr., and Roger J. Stone, joined by Peter G. Kelly in 1984. Manafort often lobbied on behalf of ...</u> User decision: April 1, 1949.</p>	

Figure 10: **CalibRAG vs Linguistic-LoRA**. In the case of CalibRAG, a document about the person in question is retrieved with a confidence level of 83.93%. In contrast, the document retrieved by the base retrieval model is related to Donald Trump but does not match the specific person in the query. Nevertheless, the Linguistic-LoRA model trust the document confidently.

<p>Original question: Where did the allies go after north Africa?</p> <p>Open-ended query: Write a paragraph about the next steps taken by the Allies following their involvement in North Africa.</p>	<p>Answer: Italy</p>
<p>Original Retrieval Model's Top-1 document (This context set the LM model's confidence to 6/10): ... Abteilung 350 procedures, subsequently the German High Command was completely taken by surprise by the bulk of the Allies Mediterranean theatre buildup and subsequent landing in French North Africa on 7 November 1942, that was known by the allies as Operation Torch. Not until the Tunisian Campaign was well under way did W-Leit 2 at Taormina pull itself together and contribute to the general effort by intercepting air support messages. It is worth noting that General Albert Praun, who was chief Signals of the Wehrmacht thought that the Allied Invasion of French North Africa was not detected due to radio... User decision: Based on the context provided, the Allies advanced towards Tunisia after North Africa.</p>	
<p>CalibRAG's Top-1 document (This context set the CalibRAG's confidence to 77.98): 1943, at the Casablanca Conference, the Allies agreed to demand Axis unconditional surrender, a war aim which implied the physical occupation of Germany with land forces. While building up strength for an invasion of continental Europe, the Allies pursued an indirect strategy by invading Europe from the South. <u>After defeating Axis forces in North Africa (the invasion of French North-Africa), Sicily and southern Italy were invaded, leading to the defeat of Fascist Italy.</u> Churchill especially favoured a Southern strategy, aiming to attack the "soft underbelly" of Axis Europe through Italy, Greece and the Balkans in a strategy similar to the> Question: Write a paragraph about the next steps taken by the Allies following their involvement in North Africa. User decision: The Allies went to Sicily.</p>	

Figure 11: **CalibRAG vs Number-LoRA**. In the case of CalibRAG, an accurate document about the location following North Africa was retrieved, allowing the user model to make a correct decision. In contrast, the base retrieval model brought a different document. Nevertheless, Number-LoRA model set this context with a confidence level of 6 out of 10, leading the user to ultimately make an incorrect decision.

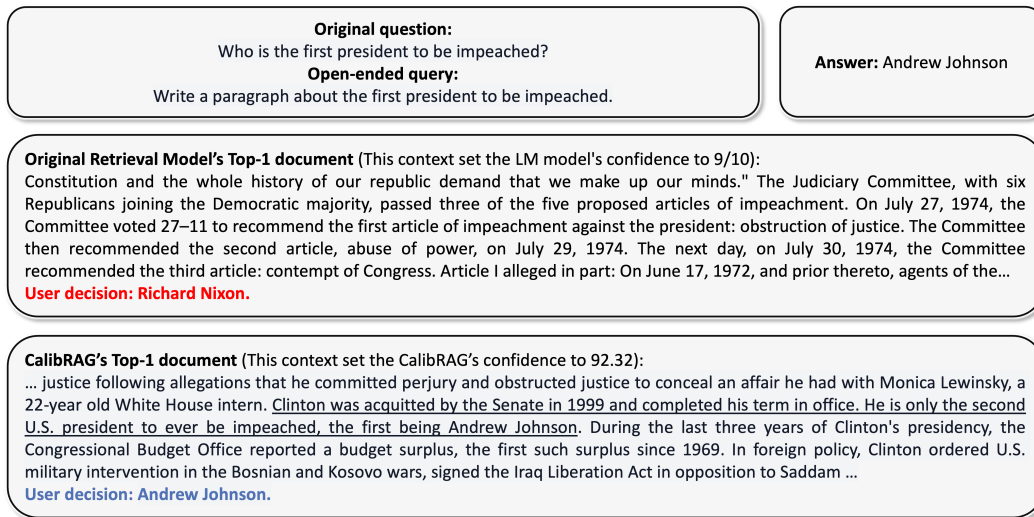


Figure 12: **CalibRAG vs Number-LoRA.** The base retrieval model focused solely on the word 'impeached' and retrieved a related document, missing the context of 'first.' Despite this, the Number-LoRA model set a confidence level of 9 out of 10, causing the user to make an incorrect decision. In contrast, CalibRAG retrieved an accurate document that, while not explicitly containing 'first impeached,' included the phrase 'first being.' It set a confidence level of 92.32%, allowing the user to arrive at the correct answer.

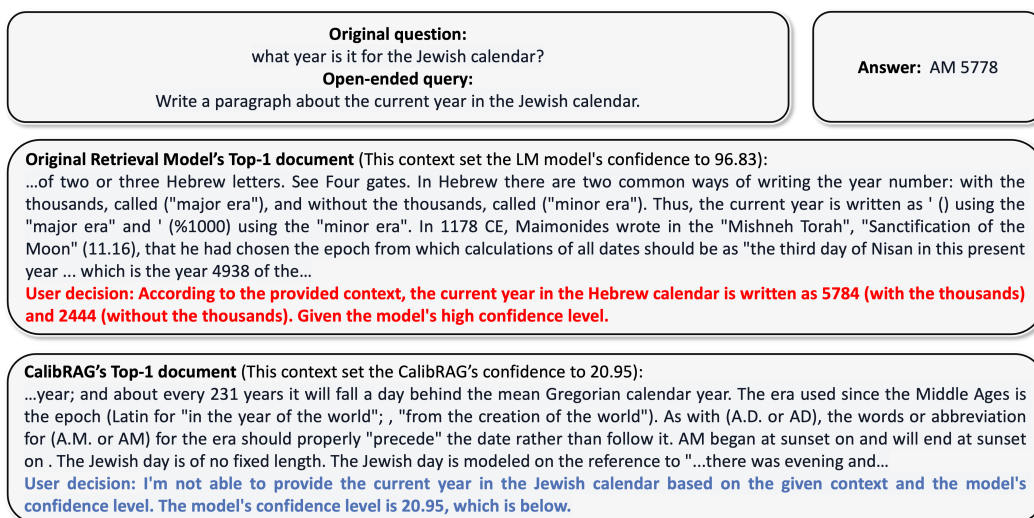


Figure 13: **CalibRAG vs CT-LoRA.** In the case of CalibRAG, the top-20 confidence score is 20.95 for incorrect information, causing the user to hesitate in making a decision. However, with the CT-LoRA model, incorrect information is assigned a confidence score of 96.83, leading the user to make an incorrect decision.

1084 G Prompt Examples

1085 In this section, we present prompt examples used during training and inference. Figure 14a shows
 1086 the prompt that encourages the user model U to act like a human decision-maker, leading it to over-
 1087 rely on the guidance provided by the LLM. Figure 14b displays the prompt that generates the open-
 1088 ended query q from the decision task x . Figure 14c presents the prompt that induces the generation
 1089 of guidance z from M based on the retrieved document d . Figure 15a is used when grading the user
 1090 model U 's decision against the true answer using \mathcal{G} . Figure 16a, Figure 16b, and Figure 16c are
 1091 prompts used to instruct \mathcal{M} to generate confidence in terms of linguistic or numerical calibration.
 1092 Lastly, Figure 15b is the prompt used during **Stage 3** of the inference process.

Decision prompt

```
The task is to answer questions based on a context generated
by a language model in response to a question about relevant
information, along with the model's confidence level in the
provided answer.
Context: {context}
Question: {question}
Model Confidence: {confidence}
Answer:
```

(a) Prompt designed to guide the user model U in making decisions based on the LLM-generated guidance z and confidence c .

Prompt that generates open-ended query q from the decision task x

```
You are an automated assistant tasked with rephrasing specific
questions into open-ended queries to encourage detailed exploration
and discussion of the key topics mentioned.
Your goal is to prompt someone to write a paragraph exploring the
topic without directly revealing the answer.
Examples for Guidance:
Example 1:
Question 1: Which sea creature is the world's largest
invertebrate?
Question 2: Write a paragraph about the world's largest
invertebrate.
...
Now, please rephrase the following question:
Question 1: {question}
Question 2:
```

(b) This prompt was first suggested by [Band et al. \[4\]](#), and we have modified part of the proposed prompt for our use here. We use this prompt as an input when generating the query q based on the decision task x .

Guidance z generation prompt

```
Directly state the answer without phrases like 'the correct answer
is.
Given the retrieved context, answer the question as accurately as
possible.
Question: {question}
Retrieved Context: {title} - {context}
Answer:
```

(c) This prompt guides the LLM \mathcal{M} to provide direct, concise guidance z based on a given retrieved document d .

Figure 14: Prompt used for (a) user model making decisions, (b) generating q from x , and (c) generating z .

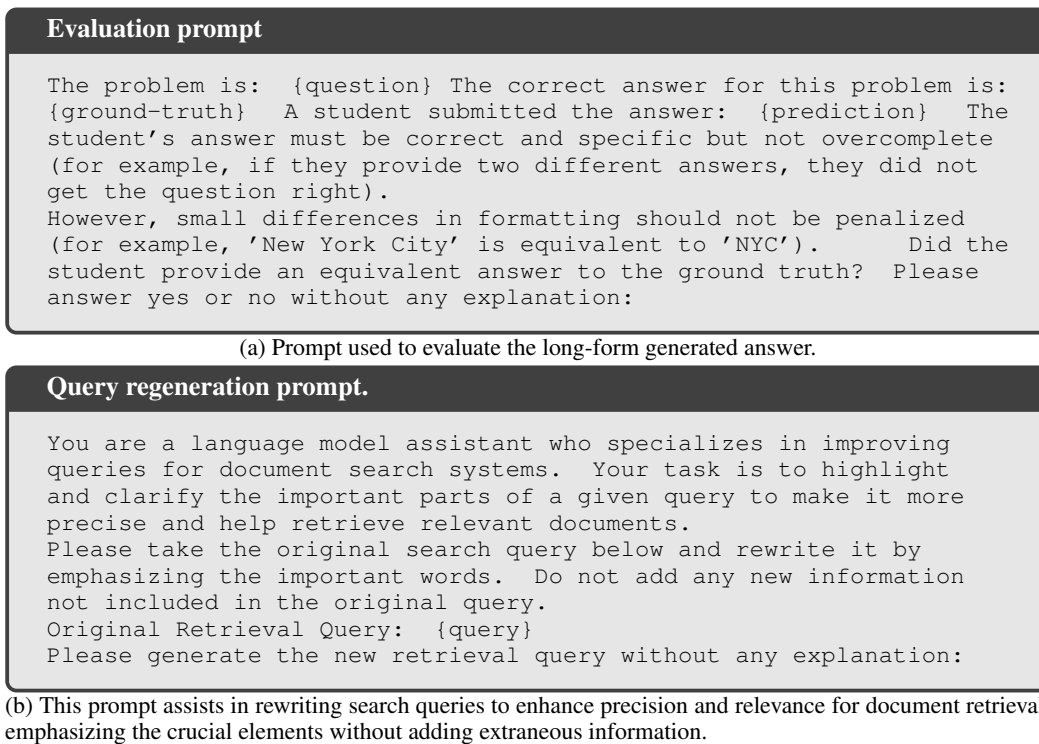


Figure 15: Prompt used for (a) evaluation and (b) query regeneration.

Calibration tuning prompt

```
Is the proposed answer correct?  
Choices:  
(i): no  
(ii): yes  
Answer:
```

(a) This prompt was first suggested by [Kapoor et al. \[20\]](#). It poses a straightforward question to verify the correctness of a proposed answer with binary choices for evaluation. We used this prompt when training our baselines.

Linguistic calibration prompt

```
Provide the certainty level of answer using the given 11 certainty  
levels. Give ONLY your certainty level, no other words or  
explanation.  
Certainty Levels: Unlikely, Doubtful, Uncertain, Ambiguous,  
Probable, Likely, Possible, Specified, Confirmed, Certain,  
Inevitable.  
For example: Certainty: <ONLY the certainty level that Answer  
is correct, without any extra commentary whatsoever; just the  
certainty level!>  
Certainty:
```

(b) This prompt requires the model to evaluate the certainty of an answer using a predefined set of linguistic levels of certainty. We used this prompt for our baselines that utilize linguistic calibration.

Number calibration prompt

```
Provide the certainty level of answer using the given 11 certainty  
levels. Give ONLY your certainty level, no other words or  
explanation.  
Certainty Levels: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.  
For example: Certainty: <ONLY the certainty level that Answer is  
correct, without any extra commentary whatsoever; just the number!>  
Certainty:
```

(c) This prompt is similar to the linguistic calibration prompt but uses numerical certainty levels (from 0 to 10) to rate the confidence in the answer provided. We used this prompt for our baselines that utilize number calibration.

Figure 16: Prompt used for baseline experiments.