SEARAG: Semantic Entropy-Guided Adaptive Retrieval for Multi-Hop Question Answering

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) significantly enhances Large Language Models (LLMs) in knowledge-intensive tasks. However, traditional static retrieval strategies fail to adapt to the evolving informational needs during generation, often resulting in insufficient or redundant content. Existing adaptive retrieval methods commonly rely on output probabilities or external heuristics, which fail to accurately reflect the model's true knowledge needs. To address this, we introduce Semantic Entropy-based Adaptive RAG (SEARAG), training a discriminative model to predict binary semantic entropy from intermediate hidden-layer states, quantifying generation uncertainty in real-time. During generation, we perform iterative sentence-by-sentence reasoning. If high semantic entropy is detected in an iteration, external knowledge retrieval is triggered for enhanced generation; otherwise, the process proceeds to the next iteration. This mechanism accurately identifies the model's knowledge needs, reduces redundant retrieval, and improves output quality. Experimental results on five multi-hop QA tasks show SEARAG outperforms existing adaptive RAG methods in performance and efficiency, confirming its effectiveness and generalization. We release our code in our Github repository.

1 Introduction

Retrieval-Augmented Generation (RAG) has emerged as a promising method to address hallucinations and factual inaccuracies inherent in Large Language Models (LLMs), which, despite their exceptional performance across various NLP tasks, frequently generate plausible but incorrect outputs (Brown et al., 2020; Ji et al., 2023; Min et al., 2023; Zhao et al., 2025). Traditional RAG frameworks typically follow a straightforward retrieve-and-generate pipeline (Lewis et al., 2020; Guu et al., 2020; Izacard and Grave, 2021), effectively handling simple queries through a single retrieval step. However, complex multi-step questions or long-form generations necessitate dynamic retrieval strategies capable of adapting the retrieval process according to real-time information needs (Zhang et al., 2023; Shao et al., 2023; Cheng et al., 2023).

Adaptive RAG methods have been developed to dynamically decide when and how external knowledge retrieval should occur. Broadly, these methods can be categorized into confidence-based strategies, model-decided retrieval, and external classifier-based approaches. Confidence-based methods such as FLARE (Jiang et al., 2023b) and DRAGIN (Su et al., 2024) rely on token-level uncertainties or attention distributions to trigger retrieval. Model-decided retrieval approaches, like Self-RAG (Asai et al., 2023), allow the language model itself, either through fine-tuning or prompting, to autonomously determine retrieval necessity. External classifier-based methods, exemplified by Adaptive-RAG (Jeong et al., 2024) and Self-Knowledge Guided Generation (Wang et al., 2023), incorporate additional classifiers to assess query complexity or output uncertainty.

Despite significant advancements in Adaptive Retrieval-Augmented Generation (RAG), precisely determining *when* large language models (LLMs) require external knowledge retrieval and effectively deciding *how* to formulate retrieval queries remain critical challenges. Recent methods such as Probing-RAG (Baek et al., 2025) have employed internal hidden-state probing to assess whether retrieval is necessary, achieving efficiency through lightweight classifiers. However, Probing-RAG is inherently non-adaptive; it lacks the ability to dynamically and continuously monitor information needs during the generation process. Consequently, it fails to detect and address evolving knowledge gaps that arise incrementally, especially in complex, multi-hop scenarios. On the other hand, methods such as SeaKR (Yao et al., 2024) effectively

leverage deep uncertainty signals derived from internal model states to guide adaptive retrieval decisions. However, SeaKR measures uncertainty by repeatedly generating outputs for the same prompt, significantly increasing computational overhead. Moreover, although SeaKR comprehensively addresses knowledge integration strategies, including self-aware re-ranking and reasoning, it neglects efficient and dynamic query formulation. This oversight can lead to mismatches between retrieved documents and the model's real-time informational needs.

However, these existing methods face critical challenges, such as unreliable retrieval triggers arising from superficial uncertainty assessments and ineffective integration of retrieved knowledge, which often leads to unnecessary retrieval steps and information conflicts. Therefore, an ideal Adaptive-RAG framework should effectively combine the capabilities of real-time detection of internal uncertainty during generation and dynamically construct precise retrieval queries that align closely with immediate information requirements.

To overcome the above limitations, we propose a novel Adaptive-RAG framework that leverages internal-state probing to estimate semantic uncertainty. Specifically, we construct training datasets using multi-hop question answering data and train a MLP-based *semantic entropy prober*, inspired by recent work on semantic entropy (Kossen et al., 2024), to predict binary semantic entropy from intermediate hidden states during generation. This enables accurate, real-time decisions on whether to trigger external retrieval. Furthermore, we introduce an advanced Query Formulation strategy: during the generation process, we dynamically identify uncertain words and utilize relevant historical context and entities to formulate targeted retrieval queries. Additionally, if newly retrieved information inadvertently increases the model's uncertainty, we adaptively expand the retrieval scope (top-k documents) and further refine the results using a reranker. Experiments demonstrate that our method consistently achieves performance that matches or surpasses state-of-the-art approaches across both in-domain and out-of-domain datasets. By integrating real-time internal-state probing and precise query formulation, our method effectively balances retrieval efficiency and generation accuracy, substantially advancing the performance of Adaptive-RAG systems.

Our main contributions are:

- We propose **SEARAG**, an adaptive RAG framework that uses a semantic entropy prober to decide when to retrieve based on hidden states of LLMs.
- We empirically show that SEARAG outperforms existing adaptive RAG baselines on multiple knowledge-intensive QA datasets.

2 Related Work

2.1 Adaptive Retrieval-Augmented Generation

Adaptive Retrieval-Augmented Generation (RAG) dynamically decides when and how external knowledge should be retrieved during generation, aiming for improved accuracy and efficiency. Existing methods primarily fall into three categories: confidence-based, model-decided, and external classifier-based. Confidence-based methods, such as FLARE (Jiang et al., 2023b) and DRAGIN (Su et al., 2024), use token-level uncertainty or attention weights to trigger retrieval. Model-decided methods, like Self-RAG (Asai et al., 2023) and MIGRES (Wang et al., 2025), enable models themselves to determine retrieval necessity via fine-tuning or prompting. External classifier-based methods, including Self-Knowledge Guided Generation (Wang et al., 2023) and Adaptive-RAG (Jeong et al., 2024), introduce additional classifiers to evaluate model outputs or query complexity. However, these methods often face issues such as inaccurate retrieval triggers due to superficial uncertainty measures and inefficient integration of external knowledge. Recent approaches, such as SeaKR (Yao et al., 2024), leverage deeper internal hidden-layer uncertainty signals within Large Language Models (LLMs) to better inform adaptive retrieval decisions.

2.2 Self-awareness via Internal States of LLMs

Recent studies have explored leveraging internal hidden states of LLMs for uncertainty detection and self-awareness. Hidden-layer representations effectively capture model confidence and detect hallucinations or inaccuracies. Fomicheva et al. (2020) and Azaria and Mitchell (2023) demonstrate that internal states reflect the truthfulness of model outputs. Approaches such as Probing-RAG (Baek

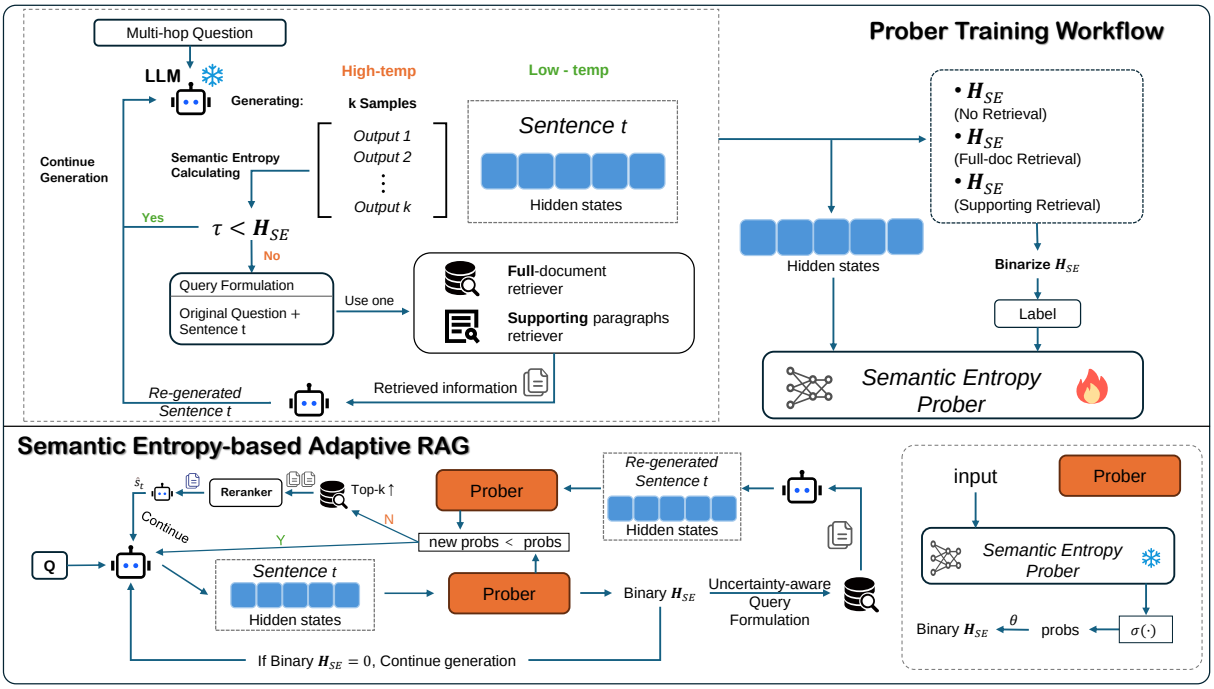

Figure 1: Two-stage framework of our semantic entropy-based adaptive RAG(SEARAG). The upper part constructs training data and supervises the training of a semantic entropy prober. The lower part performs inference, where a prober model decides retrieval using LLM hidden states.

et al., 2025) and SeaKR (Yao et al., 2024) explicitly utilize these internal states to assess knowledge sufficiency and hallucination risks, enhancing adaptive retrieval strategies. Other methods, like Lookback Lens (Chuang et al., 2024a) and Dola (Chuang et al., 2024b), exploit transformer attention and logits across layers to reduce hallucinations and improve output reliability.

2.3 Semantic Entropy

Semantic entropy has emerged as a powerful measure to quantify semantic uncertainty in LLM outputs, surpassing traditional lexical-based metrics. Introduced by Farquhar et al. (2024), semantic entropy captures model uncertainty across diverse semantic interpretations rather than lexical variations. To overcome the computational burden of multiple output generation, Semantic Entropy Probes (SEP) proposed by Kossen et al. (2024) approximate semantic entropy directly from internal states, achieving efficient and robust uncertainty detection. Thus, semantic entropy serves as a nuanced and effective real-time indicator of uncertainty, well-suited for adaptive retrieval tasks.

3 Method

In this section, we present the SEARAG framework in detail (see Figure 1). Specifically, in Section 3.1, we introduce the *Semantic Entropy Prober*, including the training data construction, training strategy, and how it is applied to determine whether retrieval should be triggered. In Section 3.2, we describe our *Uncertainty-aware Query Formulation* method. Finally, Section 3.3 presents the *Uncertainty-aware Reranking Strategy*.

3.1 Semantic Entropy Prober

Hallucination signals in language models appear in early activations before generating the first token (Snyder et al., 2024), making it possible to predict the reliability of subsequent outputs based on its hidden state. Semantic Entropy Probes (SEPs; Kossen et al. (Kossen et al., 2024)) use logistic regression to detect semantic uncertainty from hidden states at the token level. However, SEPs overlook how uncertainty relates to hidden states when relevant facts are introduced via in-context learning (ICL). Considering that the representativeness of the first token may diminish in long-text generation, we iteratively utilize sentence-level first-token signals to predict uncertainty and guide retrieval decisions.

Specifically, our designed Semantic Entropy Probe is a feed-forward network comprising a single hidden layer and an output layer that provides binary classification outcomes, determining

whether additional knowledge retrieval is necessary. Considering that the model's first layer typically captures shallow lexical or syntactic information lacking deep semantic representation (Chuang et al., 2024b), we exclude the first layer and utilize representations from all subsequent layers to leverage mid-to-high-level semantic information comprehensively.

The probe takes as input the direct concatenation of hidden states from all layers excluding the first layer for the first generated token at each step. If the input text length is n, this token is at index n. Assuming the model has L layers, the concatenation is represented as:

$$T' = \text{concat}_{l=2}^{L} H_l[n, :] \in \mathbb{R}^{(L-1)\, d_{\text{model}}}, \quad (1)$$

where $H_l[n, :] \in \mathbb{R}^{d_{\text{model}}}$ is the hidden state at position n of the l-th layer, and d_{model} is the hidden dimension. Subsequently, T' is input into the probe to compute logits, which are transformed via a Sigmoid function and used to make a binary decision on whether retrieval is necessary.

Probe Training Data Construction We iteratively generate and construct training data at the sentence level: first generating sentences at low temperature (low randomness) and recording their first token hidden states, then performing multiple samplings ($N = 10$) at high temperature (high randomness) to estimate semantic entropy for the same input. Low-temperature decoding (e.g., temperature 0.1) is nearly greedy, so hidden states are scarcely affected by sampling noise; they chiefly reflect the genuine context and retrieval state, giving the probe a clean, generalizable signal. Following discrete semantic entropy computation methods by Kuhn et al. (Kuhn et al., 2023), we cluster the sampled sentences semantically and compute entropy:

$$H_{SE}(x) = -\sum_{k=1}^{K} p(C_k|x) \, \log p(C_k|x). \quad (2)$$

Threshold Determination – Minimizing Semantic Constraints: SEPs traditionally employ only Mean Squared Error (MSE) to determine binarization thresholds, neglecting the smoothness of the semantic space. To enhance semantic consistency and robustness, we propose *Semantic-Constraint Entropy Splitting* (SCES), which augments the MSE objective with variance constraints on hidden layer features. For a candidate threshold t, we define:

$$\mathcal{L}(t) = \underbrace{\sum_{H_i<t} (H_i - \mu_{\text{low}})^2 + \sum_{H_i \geq t} (H_i - \mu_{\text{high}})^2}_{\text{MSE}(t)} + \lambda \left(\sigma^2_{\text{low}}(t) + \sigma^2_{\text{high}}(t)\right), \quad (3)$$

where μ_{low} and μ_{high} denote the mean entropy values of samples below and above the threshold t, respectively, and $\sigma^2_{\text{low}}(t)$, $\sigma^2_{\text{high}}(t)$ represent the corresponding variances in hidden-layer feature space. The final threshold τ is selected by minimizing this objective: $\tau = \arg\min_t \mathcal{L}(t)$.

SCES maintains the simplicity of MSE while introducing semantic consistency constraints to yield more robust binary labels.A detailed description of semantic entropy estimation and SCES threshold binarization is provided in Appendix A.

After data collection, semantic entropy values are uniformly calculated, and SCES determines a global threshold τ. Samples are binarized accordingly: if $H_{SE} > \tau$, the case requires retrieval ($y = 0$), otherwise it does not ($y = 1$).

During data collection, if subsequent semantic entropy computation labels a case as requiring retrieval, external knowledge retrieval is triggered during sampling, and the sentence is regenerated; otherwise, the low-temperature-generated sentence is retained directly. This process repeats until completion. We construct two versions: *full-document retrieval* and *support paragraph retrieval*, to analyze the independent effects of uncertainty estimation.

The final training set consists of 4,000 samples derived from the multi-hop question-answering datasets: 2WikiMultiHopQA (Ho et al., 2020), HotpotQA (Yang et al., 2018), IIRC (Ferguson et al., 2020), and StrategyQA (Geva et al., 2021), maintaining approximately balanced positive and negative labels. Examples of the constructed training data are provided in Appendix **??**. The probe is trained using binary cross-entropy loss with logits:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log \sigma(p_i) + (1 - y_i) \log \left(1 - \sigma(p_i)\right) \right] \quad (4)$$

See Appendix B for training hyperparameters.

3.2 Uncertainty-aware Query Formulation

To construct retrieval queries reflecting model uncertainty, we first extract the initial generated sentence and compute word-level entropy by averaging

token-level entropies. A dynamic threshold is then set as:

$$\text{threshold} = \mu_{\text{entropy}} + \alpha \cdot \sigma_{\text{entropy}}, \quad (5)$$

where μ_{entropy} and σ_{entropy} are the mean and standard deviation of the word entropies, and α controls sensitivity. We identify the first high-entropy word exceeding this threshold and use words preceding it as reliable context.

We selectively preserve content words (e.g., nouns, verbs, adjectives, numbers) from this uncertain region. To ensure query robustness, we evaluate overlap—both lexical and semantic—between these words and entities in the previous sentence or the original query. If overlap exists, we adopt the prior sentence as the query base; otherwise, we revert to the original user query. The final retrieval query concatenates the chosen base with these filtered content words, ensuring semantic relevance and contextual coherence.

3.3 Post-Retrieval Generation Continuation

At sentence-level timestep t, the model M first produces a response sentence s_t. The Semantic-Entropy Prober then inspects the hidden-state bundle from this generation step and returns a binary judgement (*certain* vs. *uncertain*). If the judgement is *certain*, s_t is emitted as the output for step t; otherwise, an external retrieval is triggered. A query $\text{qry}(q, s'_{<t}, s_t)$ is formulated with our proposed uncertainty-aware query formulation strategy and issued to the corpus $\mathcal{D}$, yielding contextual passages C_t. Subsequently, we build a new prompt by placing a few in-context examples at the top and concatenating C_t, the original question q, and the previously confirmed sentence sequence $s'_{<t}$. Feeding this Prompt_t back into the generator yields a refined sentence s'_t, which replaces the potentially hallucinated s_t.

Uncertainty-aware Reranking After retrieval and sentence regeneration, we continuously monitor the semantic uncertainty scores p_{unc}. If the regenerated sentence exhibits higher uncertainty than the original sentence by a small margin δ, i.e.,

$$p_{\text{unc}}^{\text{retrieved}} > p_{\text{unc}}^{\text{original}} + \delta, \quad (6)$$

an adaptive *expand–rerank–regenerate* procedure is triggered. This small margin δ ensures robustness by preventing unnecessary fluctuations near the decision boundary.

1. **Dynamic Recall Expansion** Temporarily increase the current top-k by five (capped at ten) and issue a new retrieval.

2. **Semantic Reranking** Re-order the newly retrieved passages with the open-source model `gte-reranker-modernbert-base` (Zhang et al., 2024), obtaining a relevance-sorted list.

3. **Prompt Reconstruction** Replace the original C_t in the prompt with the reranked passages and rebuild the prompt together with the examples and answer prefix.

4. **Regeneration** Invoke the generator on the reconstructed prompt to produce a revised sentence s'_t.

When a non-decreasing rise is detected, this strategy enlarges recall and applies the reranker to mitigate quality drop caused by retrieval noise or missing information.

Iterative Refinement Loop After appending s'_t to the answer, the system advances to timestep $t{+}1$ and repeats: (1) generate a new sentence and probe its uncertainty; (2) if retrieval is required, fetch passages and regenerate the sentence, followed—if necessary—by the uncertainty-aware reranking cycle; (3) if retrieval is not required, append the sentence directly. Each step thus produces a final sentence s'_t, which is appended to the prompt for subsequent reasoning. The loop continues until the answer is complete, coupling retrieval decisions, uncertainty detection, and dynamic reranking within a single refinement framework and thereby reducing the impact of information gaps and retrieval noise on generation quality.

4 Experiment

4.1 Experimental Setup

Datasets We conduct experiments on five publicly available open-source multi-hop QA datasets. Among them, datasets used in the training of our uncertainty prober—2WikiMultiHopQA (Ho et al., 2020), HotpotQA (Yang et al., 2018), IIRC (Ferguson et al., 2020), and StrategyQA (Geva et al., 2021)—are considered *in-domain* datasets, while MuSiQue (Trivedi et al., 2022), which is not used during training, serves as our *out-of-domain* benchmark to evaluate generalization under domain shift.

Evaluation Metrics we sample 500 examples from the test split of each dataset for evaluation. To guide the model's reasoning process and encourage more interpretable outputs, we incorporate Chain-of-Thought prompting (Wei et al., 2022) and few-shot demonstrations (Brown et al., 2020) into the input prompt. Full prompt templates are provided in Appendix A. For StrategyQA, we report the exact match (EM) score, as the target answers are binary ("yes" or "no"). For the remaining datasets—2WikiMultiHopQA, MuSiQue, HotpotQA, and IIRC—we evaluate using both answer-level exact match (EM) and token-level F1, as the answers are free-form textual spans.

For StrategyQA, we report exact match (EM) scores since the answers are binary ("yes" or "no"). For the remaining datasets—2WikiMultiHopQA, MuSiQue, HotpotQA, and IIRC—we use both answer-level EM and token-level F1 scores to assess the accuracy and completeness of generated answers.

Baselines We compare our method with several representative baselines, including the non-retrieval setting (wo-RAG), and three adaptive retrieval-augmented generation methods: FLARE (Jiang et al., 2023b), DRAGIN (Su et al., 2024), and SEAKR (Yao et al., 2024). FLARE triggers retrieval whenever a token's probability falls below a predefined threshold, and constructs the query by removing low-confidence tokens from the last generated sentence. DRAGIN determines retrieval timing based on token-level importance and uncertainty, using attention signals over the full context to formulate semantically rich queries. SEAKR leverages internal hidden-state uncertainty to decide when to retrieve, and integrates retrieved knowledge through self-aware re-ranking and reasoning strategies.

Models We evaluate all methods using three open-source instruction-tuned large language models: LLaMA-2-7b-Chat-hf (Touvron et al., 2023), LLaMA-3.1-8B-Instruct (Dubey et al., 2024), and Mistral-7B-Instruct-v0.3 (Jiang et al., 2023a). All experiments are conducted under a white-box setting to allow extraction of hidden states for adaptive retrieval.

Implementation Details We use English Wikipedia (Karpukhin et al., 2020) as the external knowledge source, segmented into passages of 100 tokens each.For each retrieval *(using the same top-k cutoff for every method on each dataset)*, we return the top k most relevant documents with the BM25 algorithm (Robertson et al., 2009), which ranks passages based on lexical matching with the input query. We directly leverage the token-level hidden states returned by Hugging Face Transformers (setting `output_hidden_states=True`).

Further details are provided in Appendix E.

4.2 Overall Results

Table 1 summarizes the performance of SEARAG and baseline methods across five open-domain QA datasets under three different language models. Several key findings emerge from the results: (1) While many RAG-based methods improve performance over the non-retrieval baseline, this trend is not consistent across all settings. In some cases—particularly on tasks like StrategyQA where the model already performs well without retrieval—certain static or poorly timed retrieval methods (e.g., SeaKR) may underperform compared to direct generation. This suggests that when retrieval is unnecessary or misaligned with the model's information needs, it can introduce noise that harms output quality. (2) SEARAG achieves the best performance on a majority of datasets and evaluation metrics across all three models. Its advantage is particularly evident on multi-hop reasoning datasets such as 2WikiMultiHopQA and HotpotQA, where precise timing and relevance of retrieval play a critical role. Compared to adaptive baselines like FLARE, DRAGIN, and SeaKR, SEARAG exhibits stronger consistency and robustness, especially under larger models such as LLaMA-3.1-8B. (3) Unlike methods such as FLARE, which rely on token-level uncertainty and may suffer from noisy retrieval triggers, SEARAG utilizes internal hidden states to make more informed retrieval decisions, leading to fewer unnecessary retrievals and higher-quality responses. The performance gain is particularly evident under more capable models (e.g., LLaMA-3.1-8B), where SEARAG achieves up to +0.171 EM and +0.153 F1 improvements over the best baseline on 2WikiMultiHopQA. (4) On out-of-domain data (MuSiQue), SEARAG maintains superior performance without access to domain-specific training data, outperforming all baselines in both EM and F1. This highlights its generalization capability beyond in-domain scenarios. (5) Compared to SeaKR, which also leverages internal signals, SEARAG shows clear improvements,

Table 1: Overall results of SEARAG and baselines on five datasets. The best results are in bold, and the second-best are underlined.

	In-Domain							Out-of-Domain	
	2WikiMultiHopQA		HotpotQA		StrategyQA	IIRC		MuSiQue	
	EM	F1	EM	F1	Accuracy	EM	F1	EM	F1
Llama2-7b									
w/o RAG	0.142	0.2181	0.189	0.2710	0.654	0.139	0.1733	0.068	0.1477
FLARE	0.226	0.3089	0.210	0.2874	0.608	0.162	0.1942	0.085	0.1593
DRAGIN	0.234	0.2884	0.235	0.3372	0.660	0.193	0.2429	0.105	0.1733
SeaKR	0.302	0.3601	**0.279**	**0.3970**	0.544	0.195	0.2351	0.042	0.1038
SEARAG (Ours)	**0.303**	**0.3933**	0.248	0.3594	**0.683**	**0.239**	**0.2879**	**0.113**	**0.1987**
Llama3-8b									
w/o RAG	0.257	0.3427	0.193	0.2949	0.757	0.160	0.1931	0.072	0.1675
FLARE	0.322	0.4126	0.260	0.3588	0.732	0.216	0.2595	0.080	0.1681
DRAGIN	0.277	0.3417	0.285	0.3802	0.760	0.196	0.2252	0.080	0.1737
SeaKR	0.302	0.3682	0.300	0.3985	0.640	0.200	0.2397	0.088	0.1642
SEARAG (Ours)	**0.493**	**0.5656**	**0.417**	**0.5163**	**0.767**	**0.317**	**0.3681**	**0.133**	**0.2696**
Mistral-7b									
w/o RAG	0.187	0.2884	0.193	0.2949	0.740	0.183	0.2231	0.070	0.1614
FLARE	0.193	0.3137	0.223	0.3225	0.723	0.143	0.1843	0.075	0.1590
DRAGIN	0.327	0.4397	0.263	0.4102	0.743	0.273	0.3381	0.115	**0.2365**
SeaKR	0.264	0.3452	0.140	0.2771	0.546	0.132	0.1835	0.056	0.1466
SEARAG (Ours)	**0.387**	**0.5018**	**0.357**	**0.4805**	**0.763**	**0.297**	**0.3459**	**0.120**	0.2172

likely due to its semantic-entropy-based probing and uncertainty-aware reranking mechanism that better aligns retrieval timing and content relevance. In summary, SEARAG demonstrates consistent and superior performance across various models, datasets, and reasoning types, particularly excelling in multi-hop and open-domain settings, validating the effectiveness of our semantic-entropy-driven adaptive retrieval design.

4.3 Timing of Retrieval

In this analysis, we investigate the impact of retrieval timing by fixing the query formulation strategy—all methods use the last complete sentence generated by the LLM as the query—and varying only the timing mechanism across different adaptive RAG frameworks. We compare SEARAG against two representative baselines: FLARE, which triggers retrieval whenever any generated token's probability falls below a predefined threshold, and DRAGIN, which determines retrieval timing based on the importance and uncertainty of generated tokens.

As shown in Table 2, SEARAG consistently outperforms both FLARE and DRAGIN on HotpotQA and IIRC across two model backbones (LLaMA3-8B and Mistral-7B). These results demonstrate the effectiveness of our semantic entropy–driven retrieval timing strategy. By leveraging internal hidden state representations to assess sentence-level uncertainty, SEARAG is able to more reliably detect the model's need for external knowledge and initiate retrieval at appropriate moments. This leads to substantial performance gains in both EM and F1 across datasets, highlighting the benefit of using internal confidence signals for adaptive control of the retrieval process.

Table 2: Comparison of adaptive RAG methods under different *retrieval timing* strategies. All methods use the last complete sentence from the LLM as the query. Results on HotpotQA and IIRC with LLaMA3-8B (L8B) and Mistral-7B (M7B). Best results are in bold.

Models	Methods	HotpotQA		IIRC	
		EM	F1	EM	F1
L8B	FLARE	0.233	0.3215	0.200	0.2407
	DRAGIN	0.276	0.3759	0.183	0.2134
	SEARAG	**0.395**	**0.5028**	**0.298**	**0.3469**
M7B	FLARE	0.202	0.3023	0.122	0.1678
	DRAGIN	0.243	0.3945	0.262	0.3256
	SEARAG	**0.343**	**0.4623**	**0.278**	**0.3347**

Table 3: Ablation study of SEARAG on three multi-hop QA datasets: 2Wiki(2WikiMultiHopQA), IIRC

Models	2Wiki		IIRC	
	EM	F1	EM	F1
SEARAG	0.493	0.5656	0.317	0.3681
Ablating Uncertainty-aware Query Formulation				
Previous Sentence Only	0.322	0.3981	0.206	0.2402
Uncertainty Sentence Only	0.424	0.5089	0.298	0.3469
Full History Query	0.368	0.4562	0.291	0.3432
Ablating Uncertainty-aware Reranking				
– U.A. Reranking	0.412	0.5025	0.290	0.3265

4.4 Ablation Study

To assess the contribution of each component in SEARAG, we conduct ablation studies on two multi-hop QA datasets: 2WikiMultiHopQA and IIRC. All experiments are performed using the LLaMA3.1-8B-Instruct model. The results are presented in Table 3.

Ablating Uncertainty-aware Query Formulation. We evaluate the impact of our proposed uncertainty-aware query formulation by replacing it with three alternative strategies: (1) *Previous Sentence Only*, which uses only the last complete sentence from the previously generated output as the query; (2) *Uncertainty Sentence Only*, which directly uses the uncertain sentence that triggered retrieval; and (3) *Full History Query*, which concatenates the question with all previously generated content to form the query. As shown in Table 3, all alternatives result in performance drops compared to our method, demonstrating the effectiveness of uncertainty-guided query construction that not only selectively retains relevant entities from the uncertain sentence but also incorporates context from previously generated content, enabling both precision and continuity in retrieval.

Ablating Uncertainty-aware Reranking. We further assess the impact of our uncertainty-aware reranking strategy by disabling it and directly using the top-k retrieved documents based on the initial retrieval scores, without applying uncertainty-based reranking or top-$(k{+}5)$ expansion. The results in Table 3 show a clear performance drop, indicating that reranking plays a key role in suppressing noisy evidence and refining the retrieved context. Compared to query formulation variants, the removal of reranking causes a comparable or even greater decrease in performance, underscoring its importance in adaptive knowledge integration.

Table 4: Performance comparison of prober training under different supervision signals (Accuracy (Acc) vs. Binary Semantic Entropy (BSE)) and data construction strategies (Single-turn vs. Multi-turn), evaluated on LLaMA-2-7B-Chat across two QA datasets.

Strategy	2Wiki		IIRC	
	EM	F1	EM	F1
Single (Acc)	0.243	0.3478	0.180	0.2262
Single (BSE)	0.273	0.3686	0.210	0.2565
Multi (Acc)	0.273	0.3556	0.207	0.2635
Multi (BSE)	**0.303**	**0.3933**	**0.239**	**0.2879**

4.5 Data Construction Strategy

To assess the impact of data construction and supervision signals on prober performance, we conduct ablation experiments comparing *single-turn* and *multi-turn* training strategies under two types of supervision: *answer accuracy (Acc)* and *binary semantic entropy (BSE)*. All experiments are performed using LLaMA-2-7B-Chat on 2Wiki and IIRC, as shown in Table 4.

In the **single-turn** setting, each training instance is based on an isolated sentence-level generation. In contrast, the **multi-turn** strategy incorporates generation history, better reflecting the iterative reasoning in multi-hop QA.

We observe two trends: (1) multi-turn training consistently improves performance over single-turn (e.g., EM improves from 0.273 to 0.303 on 2Wiki), and (2) BSE supervision yields better results than Acc, especially in F1 (e.g., 0.3933 vs. 0.3556 on 2Wiki). These results support our design choice of using multi-turn construction with BSE labels as default.

5 Conclusion

In this work, we present SEARAG, an adaptive RAG framework that employs a semantic entropy prober to estimate uncertainty from intermediate hidden states and guide retrieval decisions. We further introduce an uncertainty-aware query formulation and reranking strategy to enhance retrieval quality. Experiments on five open-domain QA datasets demonstrate that SEARAG consistently outperforms or matches existing adaptive RAG methods across both in-domain and out-of-domain settings.

6 Limitations

While SEARAG demonstrates strong performance and generalization, it has several limitations. Our framework relies on access to intermediate hidden states via the `output_hidden_states` setting in HuggingFace-style Transformer implementations, which restricts its applicability to open-source models and makes it incompatible with API-based systems. Additionally, to focus on demonstrating the method's effectiveness, we adopt fixed thresholds in several components without extensive tuning, which may limit performance in some settings. The prober is trained only on the hidden states of the first generated token (excluding the first layer), and we do not explore variations in token position or layer depth. Lastly, our experiments are conducted on relatively small models, and the effectiveness of SEARAG on larger-scale LLMs remains to be verified. In future work, we plan to overcome these limitations through methodological improvements and expanded evaluations.

7 Ethics Statement

We have taken active steps to ensure our research complies with ethical standards in responsible AI development. All datasets used in our experiments—such as 2WikiMultiHopQA, HotpotQA, IIRC, StrategyQA, and MuSiQue—are publicly available and curated for research purposes. We avoid the use of any data that may involve personal, sensitive, or manually crowdsourced annotations beyond the original dataset construction. Our experiments rely solely on inference from open-source large language models (e.g., LLaMA-3.1-8B, LLaMA-2-7B-Chat, Mistral-7B-Instruct). These models are released under research-friendly licenses, and we do not perform any additional fine-tuning or gradient-based training. As such, our methodology does not introduce additional bias, memorization risk, or ethical concerns beyond what is already present in the base models. To encourage transparency and reproducibility, we release our code and evaluation protocols to the community. This allows others to validate, improve, and extend our work in a responsible manner.

References

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*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Ingeol Baek, Hwan Chang, ByeongJeong Kim, Jimin Lee, and Hwanhee Lee. 2025. Probing-RAG: Self-probing to guide language models in selective document retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3287–3304, Albuquerque, New Mexico. Association for Computational Linguistics.

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

Xin Cheng, Di Luo, Xiuying Chen, Lemao Liu, Dongyan Zhao, and Rui Yan. 2023. Lift yourself up: Retrieval-augmented text generation with self-memory. In *Advances in Neural Information Processing Systems*, volume 36, pages 43780–43799. Curran Associates, Inc.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James R. Glass. 2024a. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1419–1436, Miami, Florida, USA. Association for Computational Linguistics.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. 2024b. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. 2020. Iirc: A dataset of incomplete information reading comprehension questions. *arXiv preprint arXiv:2011.07127*.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Gautier Izacard and Edouard Grave. 2021. Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. In *EACL 2021 - 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 874–880, Kiev, Ukraine. Association for Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7036–7050, Mexico City, Mexico. Association for Computational Linguistics.

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

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023a. Mistral 7b. *Preprint*, arXiv:2310.06825.

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

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in llms. *Preprint*, arXiv:2406.15927.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

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.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274, Singapore. Association for Computational Linguistics.

Ben Snyder, Marius Moisescu, and Muhammad Bilal Zafar. 2024. On early detection of hallucinations in factual question answering. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 2721–2732, New York, NY, USA. Association for Computing Machinery.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. 2024. Dragin: Dynamic retrieval augmented generation based on the real-time information needs of large language models. *arXiv preprint arXiv:2403.10081*.

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. 9835 musique: Multi-hop questions via single-hop question composition. *Trans. Assoc. Comput. Linguistics*, 10:539–554.

Keheng Wang, Feiyu Duan, Peiguang Li, Sirui Wang, and Xunliang Cai. 2025. LLMs know what they need: Leveraging a missing information guided framework

to empower retrieval-augmented generation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2379–2400, Abu Dhabi, UAE. Association for Computational Linguistics.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10303–10315, Singapore. Association for Computational Linguistics.

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

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Zijun Yao, Weijian Qi, Liangming Pan, Shulin Cao, Linmei Hu, Weichuan Liu, Lei Hou, and Juanzi Li. 2024. Seakr: Self-aware knowledge retrieval for adaptive retrieval augmented generation. *arXiv preprint arXiv:2406.19215*.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, et al. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's song in the ai ocean: A survey on hallucination in large language models. *Preprint*, arXiv:2309.01219.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2025. A survey of large language models. *Preprint*, arXiv:2303.18223.

A Semantic Entropy Estimation and Threshold Binarization

Semantic Entropy Estimation. For a given context x, an auto-regressive LLM defines the joint probability of a sequence $s = (t_1, \ldots, t_n)$ as

$$p(s \mid x) = \prod_{i=1}^{n} p(t_i \mid t_{<i}, x), \tag{7}$$

and the probability mass of a semantic cluster C_k is the sum of the probabilities of all sequences that fall into that cluster,

$$p(C_k \mid x) = \sum_{s \in C_k} p(s \mid x). \tag{8}$$

The semantic entropy of x is then

$$H_{SE}(x) = -\sum_{k=1}^{K} p(C_k \mid x) \, \log p(C_k \mid x). \tag{9}$$

Practical estimation. Direct evaluation of Eqs. (7)–(9) is infeasible: the number of possible token sequences grows exponentially with sequence length. Following the Monte-Carlo protocol of Kuhn et al. (2023), we therefore draw $N = 10$ high-temperature generations $\mathcal{S} = \{s^{(j)}\}_{j=1}^{N}$ and cluster them semantically: two samples are assigned to the same cluster if they mutually entail each other according to an off-the-shelf NLI model. This greedy procedure yields at most $K \leq N$ clusters $\{C_k\}_{k=1}^{K}$. Let n_k denote the number of samples in C_k. The cluster probability is approximated by

$$\hat{p}(C_k \mid x) = \frac{n_k}{N}, \tag{10}$$

and the semantic entropy estimate becomes

$$\widehat{H}_{SE}(x) = -\sum_{k=1}^{K} \hat{p}(C_k \mid x) \, \log \hat{p}(C_k \mid x). \tag{11}$$

This discrete, sampling-based estimator is model-agnostic and supplies a reliable measure of semantic uncertainty for the subsequent threshold-binarization step.

Global Threshold via Semantic-Constraint Entropy Splitting (SCES). To convert the continuous entropy scores into binary labels required by the prober, we determine a global threshold τ on the *training* set. Let $\{H_i\}$ be the set of entropy scores and $\{Z_i\}$ the corresponding hidden-state features of the first output token (concatenated across layers). Previous work (SEPs) minimizes the mean-squared error (MSE) of within-group entropy to select τ, but ignores the topology of the feature space. We therefore introduce **Semantic-Constraint Entropy Splitting** (SCES), which augments the MSE objective with a variance term that penalizes fea-

ture dispersion on both sides of τ:

$$\mathcal{L}(t) = \underbrace{\sum_{H_i<t}(H_i-\mu_{\text{low}})^2 + \sum_{H_i\geq t}(H_i-\mu_{\text{high}})^2}_{\text{MSE}(t)} + \lambda\big(\sigma^2_{\text{low}}(t)+\sigma^2_{\text{high}}(t)\big), \quad (12)$$

where μ_{low} and μ_{high} are the mean entropies below and above t, and $\sigma^2_{\text{low}}(t) = \frac{1}{|H_i<t|}\sum_{H_i<t}\|Z_i - \bar{Z}_{\text{low}}\|_2^2$ denotes the feature variance (analogously for σ^2_{high}). The hyper-parameter λ is kept small to avoid over-regularisation. A fine-grained grid search over t minimises (12), yielding $\tau = \arg\min_t \mathcal{L}(t)$. Each sample is then labelled $y_i = \mathbf{1}[H_i \leq \tau]$, and the same τ is applied at inference to decide whether the current sentence warrants external retrieval.

SCES retains the simplicity of the original MSE split while enforcing semantic smoothness in the hidden-state space, resulting in more stable binary decisions and improved prober performance.

B Semantic Entropy Prober Training Configuration

Table 5: Prober Training Configuration

Hyperparameter	Value
Learning Rate	1e-3
Batch Size	64
Max Epochs	100
Dropout Rate	0.3
Activation Function	ReLU
Optimizer	AdamW
Learning Rate Scheduler	ReduceLROnPlateau
Patience	10
Weight Decay	1e-5
Criterion (Loss Function)	BCEWithLogitsLoss
Device	A100 × 2
Early Stopping	Enabled

This section outlines the key hyperparameters used for training the prober model (see Table 5 for details). We employed early stopping to prevent overfitting, using validation loss as the primary criterion. If the validation loss did not improve for a specified number of epochs (patience), training was halted, and the best model state was retained.

Our focus was primarily on methodological innovation, and we did not conduct extensive hyperparameter tuning. The model's parameters were set to reasonable defaults.

Algorithm 1 Iterative Refinement with Uncertainty-aware Retrieval

Input: question q; corpus $\mathcal{D}$; LLM $\mathcal{M}$; retriever $\mathcal{R}$; prober P; thresholds τ, δ; $\mathcal{M}.\text{generate}(\cdot)$ G
Output: final answer $\mathbf{A}$

$\mathbf{prompt} \leftarrow q$; $\mathbf{A} \leftarrow \emptyset$
while not done **do**
  $(s, h) \leftarrow G(\mathcal{M}, \mathbf{prompt})$
  $u_{\text{orig}} \leftarrow \text{P}(h)$
  if $u_{\text{orig}} > \tau$ **then**
    $C \leftarrow \textsc{Retrieve}(\mathcal{R}, \text{qry}(q, \mathbf{A}, s))$
    $(s', h') \leftarrow G(\mathcal{M}, [\mathbf{prompt}; C])$
    $u_{\text{ret}} \leftarrow \text{P}(h')$
    if $u_{\text{ret}} > u_{\text{orig}} + \delta$ **then**
      $C \leftarrow \textsc{Expand}(C, +5)$
      $C \leftarrow \textsc{Rerank}(C)$
      $(s', h') \leftarrow G(\mathcal{M}, [\mathbf{prompt}; C])$
    end if
    $s^\star \leftarrow s'$
  else
    $s^\star \leftarrow s$
  end if
  $\mathbf{A} \leftarrow \mathbf{A} \,\|\, s^\star$
  $\mathbf{prompt} \leftarrow [\mathbf{prompt}; s^\star]$
end while
return $\mathbf{A}$

C SEARAG: Sentence-level Iterative Generation with Semantic Entropy-guided Retrieval

This algorithm 1 presents the core procedure of **SEARAG**, our proposed sentence-level adaptive generation framework. At each decoding step, the language model generates a sentence and computes its semantic entropy via internal hidden states. If the uncertainty exceeds a learned threshold, the system triggers retrieval to regenerate the sentence using relevant external knowledge. When retrieval fails to reduce uncertainty, SEARAG performs a one-time expansion and reranking of the retrieved results. This process continues iteratively until the full answer is produced, ensuring each sentence is generated with minimal hallucination and maximal grounding.

D Examples of Training Data for the Semantic Entropy Prober

Tables 6, 7, 8 illustrates three representative cases from our training data under different retrieval conditions: no retrieval, full-document retrieval, and

Table 6: Example: No Retrieval

Question	Question: In what city and country was the Jagdgeschwader 77 based?
Generated Answer	/
High-temperature Samples	1. Jagdgeschwader 77 was a German World War II fighter wing. 2. The Jagdgeschwader 77 was based in Neuburg. 3. The Jagdgeschwader 77 was based in Merseburg, Germany. 4. Jagdgeschwader 77 (77th Air Fighter Wing) was based in Merseburg, in the country of Germany. 5. The Jagdgeschwader 77 was based in Berlin, Germany. 6. The Jagdgeschwader 77 (JG 77) was based in Graz, Austria. 7. Jagdgeschwader 77 was a Luftwaffe fighter wing based in the city of Lille in the country of France. 8. Jagdgeschwader 77 was based in Berlin, Germany. 9. Jagdgeschwader 77 (77th Fighter Wing) was based in the city of Döberitz, Germany. 10. Jagdgeschwader 77 was a Luftwaffe fighter wing of World War II.
Semantic Entropy	1.89

supporting-paragraph retrieval. Each example includes the model-generated answer, a set of high-temperature generations, and the corresponding semantic entropy value computed using our NLI-based clustering and Eq. 11.

These entropy scores are subsequently converted into binary supervision labels using our proposed Semantic-Constraint Entropy Splitting (SCES) method (see Section A). Specifically, a global threshold τ is determined by jointly minimizing intra-cluster entropy variance and hidden-state dispersion. Samples with entropy below τ are labelled as confident (i.e., no retrieval required), while those exceeding the threshold are treated as uncertain (i.e., retrieval beneficial). These binary labels serve as training targets for the semantic entropy prober.

E More Details about Experiment Setup

Datasets We conduct experiments on five open-domain, knowledge-intensive QA datasets to evaluate our model's reasoning ability across diverse dimensions, including multi-hop reasoning, reading comprehension under incomplete contexts, and strategic commonsense reasoning. The datasets include 2WikiMultiHopQA (Ho et al., 2020), HotpotQA (Yang et al., 2018), IIRC (Ferguson et al., 2020), StrategyQA (Geva et al., 2021), and MuSiQue (Trivedi et al., 2022).

2WikiMultiHopQA is a multi-hop question answering benchmark constructed over Wikipedia articles. It contains roughly 20,000 questions, each designed to require reasoning over two supporting passages from different articles. This dataset emphasizes the model's ability to synthesize information across multiple sources, trace entity relationships, and perform multi-step reasoning to select the correct answer from a candidate set.

HotpotQA is another large-scale multi-hop QA dataset containing 113,000 questions. Unlike 2WikiMultiHopQA, HotpotQA not only requires reasoning across documents but also includes distractor paragraphs, introducing additional noise and making retrieval more challenging. Questions often have concise factual answers, such as names, dates, or short descriptions, and demand evidence aggregation from multiple documents for correct answering.

MuSiQue (Multi-hop Structured Questions) is a manually curated multi-hop QA dataset that combines high-quality questions with structured multi-step reasoning annotations. It is designed to reduce shortcut artifacts common in other QA datasets and to ensure that answering requires following an explicit multi-fact reasoning chain. MuSiQue is known for its compositional complexity and linguistic variation, providing a stringent testbed for evaluating step-wise retrieval and reasoning capabilities.

IIRC (Incomplete Information Reading Com-

Table 7: Example: Full Document Retrieval

Question	Are The Great Commandment and The Night Is My Kingdom from the same country?
Retrieved Paragraphs from the Full Document	[1] Mir Kino Theatre in Moscow. The film was screened throughout ... [2] Great Is My Country Vast is my Native Land (1958), also known ... [3] Night", set a record in late 2013 as the first male country music...
Generated Answer	The Great Commandment is from the country of United States of America.
High-temperature Samples	1. The Night Is My Kingdom is from the country of Russia. 2. The Night Is My Kingdom is from the country of United Kingdom. 3. The Night Is My Kingdom is from the country of Australia. 4. The Night Is My Kingdom is from the country of Russia. 5. The Night Is My Kingdom is from the country of Russia. 6. The Night Is My Kingdom is from the country of Russia. 7. The Night Is My Kingdom is from the country of Russia. 8. The Night Is My Kingdom is from the country of Republic of Macedonia. 9. The Night Is My Kingdom is from the country of Russia. 10. The Night Is My Kingdom is from the country of Russia.
Semantic Entropy	0.94

prehension) consists of 13,441 questions grounded in Wikipedia paragraphs. Each question is constructed to deliberately lack sufficient context in its immediate paragraph, thereby requiring retrieval of additional information from linked documents. The minimal lexical overlap between questions and their required evidence simulates real-world information-seeking scenarios. Moreover, the dataset contains unanswerable and multi-hop questions, posing a significant challenge to retrieval-based methods.

StrategyQA is a compact yet highly challenging dataset of 2,780 strategic yes/no questions. Each sample includes a high-level question, a decomposition of implicit reasoning steps, and supporting evidence. The questions require models to combine world knowledge with common sense and perform implicit, often abstract reasoning. Adversarial filtering and annotation protocols ensure the dataset avoids superficial cues, making it an ideal benchmark for evaluating reasoning generalization.

Taken together, these datasets cover a wide spectrum of retrieval and reasoning difficulty. While HotpotQA, 2WikiMultiHopQA, and MuSiQue focus on compositional multi-hop reasoning, IIRC emphasizes the ability to bridge incomplete information. StrategyQA evaluates models' competence in abstract commonsense reasoning, where surface-level lexical overlap is minimal and strategic thinking is crucial.

Prompt Settings The prompt template used in all experiments is structured as follows:

```
Examples:
Question: Q1
Answer: Reasoning steps... So the answer is A1.
Question: Q2
Answer: Reasoning steps... So the answer is A2.
...
Context:
[1] Context 1
[2] Context 2
...
Answer in the same format as before.
Question: < input question >
Answer:
```

We adopt few-shot prompting with fixed numbers of exemplars for each dataset, considering both reasoning complexity. Specifically, we use 6 exemplars for 2WikiMultiHopQA, and 8 exemplars for HotpotQA, IIRC, StrategyQA, and MuSiQue. Across all datasets, we set the retrieval top-k to 3 to balance knowledge coverage and prompt compactness.

Table 8: Example: Supporting Paragraph Retrieval

Question	What is the date of birth of the performer of song Titanic (Falco Song)?
Supporting Paragraphs	[1] Titanic (Falco song): "Titanic" is a song by Falco from his 1992 studio album "Nachtflug". [2] Falco (musician): Johann Hölzel (19 February 1957 – 6 February 1998), better known by his stage name Falco, was an Austrian singer and songwriter.
Generated Answer	The date of birth of the performer of the song "Titanic" is 19 February 1957.
High-temperature Samples	1. The date of birth of Falco is 19 February 1957. 2. Falco was born on 19 February 1957. 3. Falco was born on 19 February 1957. 4. Falco was born on 19 February 1957. 5. Falco was born on 19 February 1957. 6. Falco was born on 19 February 1957. 7. Falco's date of birth is 19 February 1957. 8. The date of birth of Falco is 19 February 1957. 9. Falco was born on 19 February 1957. 10. Falco was born on 19 February 1957.
Semantic Entropy	0