RELEVANCE SAYS YES, BUT SUFFICIENCY SAYS NO: EFFICIENT ADAPTIVE RETRIEVAL VIA LATENT EVIDENCE SUFFICIENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Adaptive retrieval-augmented generation (RAG) must decide whether additional retrieval is needed, which evidence is worth acquiring, and when retrieval should stop. Existing approaches typically rely on retrieval relevance, uncertainty, or explicit model judgments, yet relevance does not indicate whether the current evidence is sufficient, and highly relevant passages may still be redundant. We further find that explicit sufficiency judgments can be systematically misaligned with the sufficiency information encoded in hidden representations. Building on these observations, we introduce **SuREST**, an adaptive RAG framework that uses a lightweight probe to recover latent evidence sufficiency and employs it as a unified control signal throughout retrieval. Across four multi-hop QA benchmarks, SuREST achieves the best answer quality among the evaluated baselines. With Qwen2.5-7B-Instruct, it improves macro F1 by 2.36 points over IRCoT, the strongest baseline by macro F1, while reducing retained passages by 56.7% and end-to-end inference time by 53.6%. Evidence analysis suggests that these gains come from reducing redundant evidence while largely preserving coverage. These results demonstrate that latent evidence sufficiency provides an effective basis for jointly improving retrieval effectiveness, evidence quality, and efficiency.

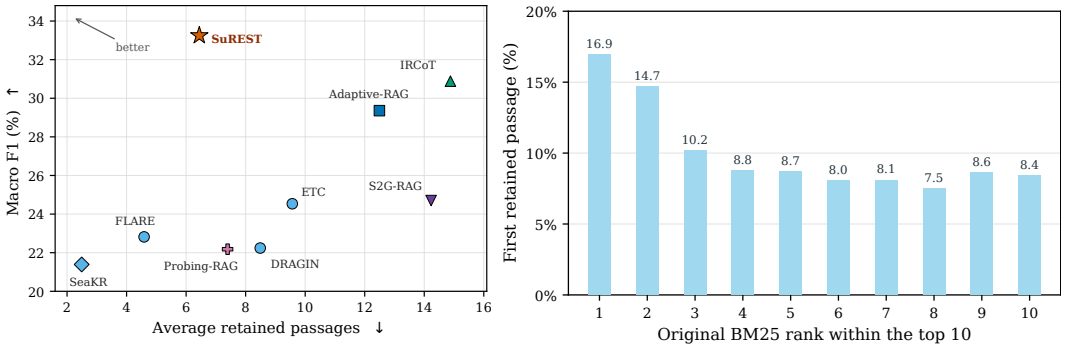
1 INTRODUCTION

Retrieval-augmented generation (RAG) improves language models by grounding generation in external evidence (Lewis et al., 2020), but effective retrieval requires more than retrieving relevant documents (Trivedi et al., 2023; Joren et al., 2025). In multi-hop QA, retrieval needs evolve with the current evidence state (Trivedi et al., 2023; Su et al., 2024): some questions need no retrieval, whereas others require multiple complementary facts. Moreover, unnecessary or irrelevant context can increase cost and distract the model (Yoran et al., 2024). The goal is therefore to acquire sufficient evidence for accurate answers while avoiding unnecessary retrieval rounds and redundant context processing. Adaptive RAG must therefore answer three coupled questions: *whether* additional retrieval is needed, *which* evidence should be acquired, and *when* retrieval should stop.

Most adaptive RAG approaches address these decisions through signals such as question complexity, generation confidence, uncertainty, attention, or predicted retrieval necessity (Trivedi et al., 2023; Jiang et al., 2023; Su et al., 2024; Li et al., 2026a; Jeong et al., 2024; Yao et al., 2025; Baek et al., 2025). More recent work such as S2G-RAG (Li et al., 2026b) explicitly judges evidence sufficiency and identifies missing information for further retrieval. This more directly targets evidence sufficiency, but raises a key question: how reliable are the model's explicit sufficiency judgments? When sufficient evidence is misjudged as insufficient, retrieval may continue without improving answer quality. A reliable sufficiency signal should therefore guide both *whether* to retrieve and *which* evidence to acquire, since passage utility depends on the current evidence state (Trivedi et al., 2023): a passage may be highly relevant yet redundant given what has already been acquired.

We find a systematic mismatch between explicit sufficiency decisions and the information encoded in hidden representations. On controlled evidence pairs, hidden states yield a more discriminative sufficiency signal than the model's native output. Importantly, this gap remains when comparing two readouts of the same final-layer representation, indicating a readout mismatch rather than an absence

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Trade-off (b) Evidence Selection

Figure 1: **Quality-efficiency and evidence selection in SUREST. (a)** Macro F1 versus mean retained passages. **(b)** BM25 rank of the first retained passage across 3,047 evidence-need retrieval calls with at least one retained passage; initial-question retrievals are excluded.

of sufficiency information. A lightweight linear probe reliably reads out this latent sufficiency signal across prompt formulations, label mappings, and training seeds. This motivates using latent evidence sufficiency, rather than explicit decisions, as the control signal for adaptive retrieval.

Building on this finding, we introduce SUREST (**Su**fficiency-guided **R**etrieval, **E**vidence **S**election, and **T**ermination), which uses latent evidence sufficiency as a unified control signal to decide when to retrieve, what to retain, and when to stop. A sufficiency gate avoids unnecessary retrieval, while targeted queries seek missing facts. At its core, *Sufficiency Margin Gain* (SMG) scores each candidate by how much it reduces the insufficiency of the current evidence set, favoring complementary evidence over redundant passages. As Figure 1 (b) shows, the first retained passage is often not BM25's top-ranked result: retrieval rank alone does not determine its value to the current evidence set. The same sufficiency signal stops retrieval once enough evidence has been assembled.

Across four multi-hop QA benchmarks, SUREST consistently outperforms all the evaluated methods while substantially reducing retrieval cost and inference time. The gains generalize across three language-model backbones, and evidence-quality analysis shows that SUREST improves efficiency primarily by reducing redundant evidence while largely preserving coverage.

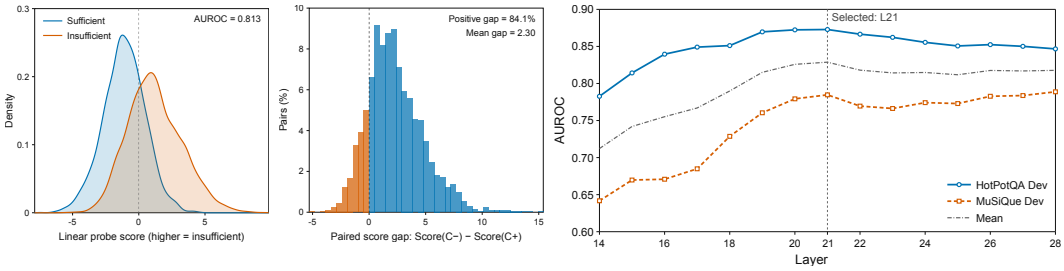
Our main contributions are:

- We identify a systematic mismatch between explicit evidence-sufficiency decisions and the sufficiency information encoded in language-model hidden representations, and show that a lightweight linear probe provides a stable readout of this latent signal.
- We introduce SUREST, which uses latent evidence sufficiency to jointly control retrieval triggering, conditional evidence acquisition through SMG, and adaptive termination.
- Across four open-domain multi-hop QA benchmarks, SUREST consistently improves answer quality while reducing retrieval and inference cost, supporting latent evidence sufficiency as an effective control signal for adaptive retrieval.

2 Related Work

2.1 Adaptive Retrieval-Augmented Generation

Fixed retrieval applies the same strategy regardless of question difficulty or the evidence already acquired, risking either missing evidence or unnecessary retrieval. Adaptive RAG instead responds to evolving information needs. IRCoT (Trivedi et al., 2023) interleaves reasoning and retrieval; FLARE (Jiang et al., 2023), DRAGIN (Su et al., 2024), and ETC (Li et al., 2026a) trigger retrieval from generation-time signals. Adaptive-RAG (Jeong et al., 2024) routes questions by predicted complexity, while SeaKR (Yao et al., 2025) and Probing-RAG (Baek et al., 2025) use internal model states to guide retrieval. Other work makes retrieval sensitive to missing or complementary evidence.



(a) Score distribution (b) Paired score difference (c) Layer-wise decodability

Figure 2: **Latent evidence sufficiency in hidden representations. (a)** Linear-probe score distributions at L21 (the 21st transformer layer) for sufficient and insufficient contexts on the independent MuSiQue confirmatory set. **(b)** Paired score differences $s(q, C^-) - s(q, C^+)$; positive values assign higher insufficiency to C^-. **(c)** Development-set AUROC across the final 15 layers on HotPotQA and MuSiQue; L21 is selected using the mean development AUROC across the two datasets.

MIGRES (Wang et al., 2025) identifies information gaps to guide search and filters retrieved content; S2G-RAG (Li et al., 2026b) uses explicit sufficiency and gap judgments for iterative retrieval. SetR (Lee et al., 2025) selects passages as a set to cover complementary information, while Joren et al. (2025) use context sufficiency to analyze RAG failures.

Relevance alone does not determine evidence sufficiency or utility: a highly relevant passage may still be redundant given the current evidence. Our analysis further shows that the model's native sufficiency decision can underuse sufficiency information encoded in hidden states. We therefore read out latent evidence sufficiency and use its state-dependent change to score candidate passages, while using the same signal to trigger and stop retrieval.

2.2 Internal Representations and Probing

Hidden states can encode task-relevant information that is not fully expressed in model outputs, and probes can recover such information (Burns et al., 2023; Li et al., 2023). In adaptive RAG, SeaKR (Yao et al., 2025) uses internal uncertainty to trigger retrieval and rerank passages, while Probing-RAG (Baek et al., 2025) probes hidden states for retrieval necessity. We instead probe *evidence sufficiency*: whether the current evidence supports a complete answer path. We compare this signal with the model's native sufficiency readout, then use it to decide when to retrieve, which evidence to retain, and when to stop.

3 Latent Evidence Sufficiency in Language Models

Prior work has shown that identifying whether context is sufficient provides a useful lens for analyzing RAG failures (Joren et al., 2025). We ask a complementary question: does a language model internally encode whether the available evidence is sufficient to answer a question? If so, why not rely directly on the model's explicit *Sufficient*/*Insufficient* decision? We study these questions before introducing SUREST. Specifically, we test whether evidence sufficiency is decodable from hidden states and whether the model's native output faithfully exposes this information. All experiments in this section use Qwen2.5-7B-Instruct, our main experimental backbone.

3.1 Probing Latent Evidence Sufficiency

Paired sufficiency data. For each question q_i, we construct paired contexts (C_i^+, C_i^-) that differ primarily in evidence completeness: C_i^+ contains a complete supporting path, whereas C_i^- lacks at least one required reasoning hop. We exclude cases that remain answerable without the intended evidence, and use question-disjoint train, development, and evaluation splits, including an independent MuSiQue confirmatory set. Full construction details are provided in Appendix A.

Table 1: Representation–output alignment on the MuSiQue confirmatory set. BAcc. denotes balanced accuracy at a zero semantic-margin threshold, and Pair denotes paired ranking accuracy.

Readout	Layer	AUROC ↑	BAcc. ↑	C^+ Acc.	C^- Acc.	Pair ↑
Native LM head	28	0.678	0.544	0.142	0.946	0.724
Linear probe	28	0.797	0.723	0.764	0.682	0.820
Linear probe	21	**0.813**	**0.735**	0.761	0.708	**0.841**

Probe construction. We extract the hidden state $h^{(\ell)}$ at the sufficiency-decision position and train a linear probe

$$s_\ell(q, C) = \mathbf{w}_\ell^\top \widetilde{\mathbf{h}}^{(\ell)}(q, C) + b_\ell, \tag{1}$$

where $\mathbf{w}_\ell$ and b_ℓ are the probe parameters. Larger scores indicate greater evidence insufficiency. To reduce dependence on prompt wording and label tokens, training pools multiple instruction variants and swapped label mappings. We train five probes for each of the final 15 layers and average their scores at inference. Layer selection uses only mean development AUROC across HotPotQA and MuSiQue, yielding L21. Full training and selection details are given in Appendix A.

Sufficiency is linearly decodable and stable. Figure 2 shows clear separation between sufficient and insufficient contexts and positive paired score gaps for most examples. The selected L21 probe achieves AUROC 0.872 on held-out HotPotQA and 0.813 on the independent MuSiQue confirmatory set, with 0.841 pair-ranking accuracy. Performance is stable across prompt–mapping conditions (AUROC 0.802–0.821) and training seeds (standard deviation 0.0039), indicating that evidence sufficiency is robustly encoded in hidden representations rather than being an artifact of a particular prompt or label mapping. Additional analyses in Appendix A.5 further verify robustness across probe architectures, prompting conditions, training seeds, answer correctness, and datasets.

3.2 Misalignment Between Latent Sufficiency and the Native Readout

Decodability alone does not justify introducing a probe: the model's native output may already expose the same sufficiency information. We therefore compare the native final-layer readout with a linear probe trained on the same L28 representation. For each label mapping, let $t_{\text{Insufficient}}$ and $t_{\text{Sufficient}}$ denote the label tokens assigned to the two semantic classes, and let $z(t)$ denote the next-token logit at the sufficiency-decision position. We define the native semantic margin as

$$s_{\text{LM}}(q, C) = z(t_{\text{Insufficient}}) - z(t_{\text{Sufficient}}), \tag{2}$$

so that larger values consistently indicate greater evidence insufficiency. Table 1 shows that the native readout is strongly biased toward insufficiency: it correctly classifies 94.6% of insufficient contexts but only 14.2% of sufficient ones, predicting *Insufficient* in 90.2% of cases. In contrast, a linear probe on the same L28 representation improves balanced accuracy from 0.544 to 0.723 and AUROC from 0.678 to 0.797.

The strong native bias, together with the linear decodability of sufficiency from the same representation, points to a readout mismatch rather than an absence of sufficiency information. The L28 probe improves AUROC by 0.120 under paired bootstrap resampling (95% CI: $[0.099, 0.141]$), and recovers 73.5% of sufficient contexts that the native head incorrectly classifies as insufficient. Since AUROC is threshold-independent, the gap cannot be explained by simple threshold miscalibration.

One plausible explanation is that task-relevant information remains encoded in hidden representations even when it is not faithfully expressed in model outputs (Burns et al., 2023; Li et al., 2023), while discrete decisions can additionally be shaped by prompt- and verbalizer-dependent output priors (Zhao et al., 2021; Li et al., 2024). Together with the stronger development-selected L21 representation, these results motivate using latent sufficiency rather than the native direct sufficiency decision as the control signal for SUREST. Section 6.1 further confirms this end to end: replacing the probe with a calibrated native readout degrades both answer quality and retrieval efficiency.

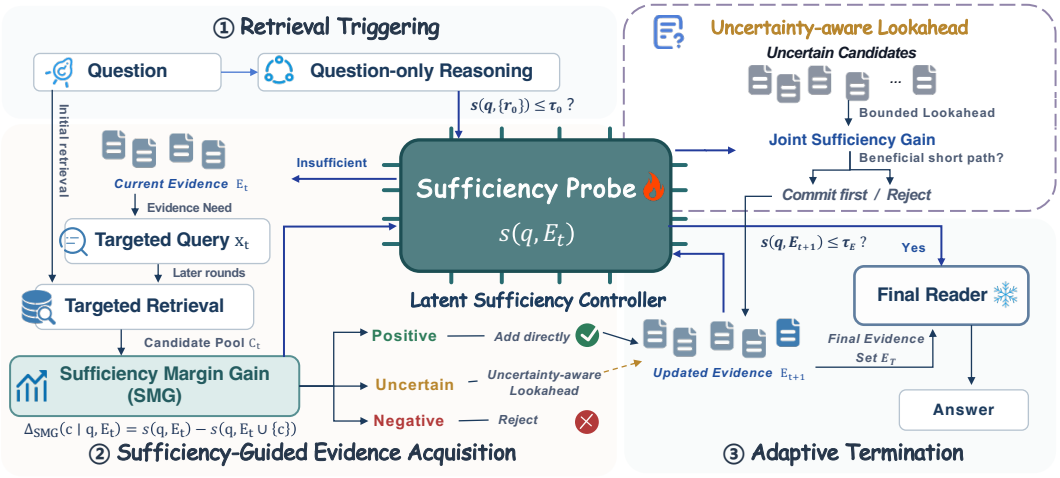


Figure 3: **Overview of SUREST.** A latent sufficiency probe controls retrieval triggering and termination, while SMG guides evidence selection with bounded lookahead for uncertain candidates.

4 SUREST

4.1 OVERVIEW

We introduce SUREST, a sufficiency-guided framework that uses a single latent signal to control three decisions in adaptive RAG: *whether* to retrieve, *which* evidence to acquire, and *when* to stop. Let q denote a question and E_t the evidence retained after step t. We use the ensemble L21 probe score $s(q, E_t) = \bar{s}_{21}(q, E_t)$ from Eq. 8, where larger values indicate greater evidence insufficiency. An evidence set is *sufficient* if it contains the factual information needed to support a valid reasoning path from the question to the answer; sufficiency is therefore a property of the evidence, not a guarantee of answer correctness.

As shown in Figure 3, SUREST uses this score as a closed-loop control signal: it gates question-only answers, scores retrieved candidates by their reduction in latent insufficiency, and terminates once the accumulated evidence is sufficient. Unlike relevance-based selection, each acquisition decision is conditioned on the current evidence state, allowing complementary passages to be favored over redundant ones. Algorithm 1 in Appendix B.1 gives the full procedure. We use a conservative pre-retrieval threshold τ_0 and a retrieved-evidence threshold τ_E; otherwise, the same score governs retrieval triggering, evidence selection, and termination.

4.2 RETRIEVAL TRIGGERING VIA LATENT SUFFICIENCY

Before retrieval, we ask M to produce a question-only reasoning trace r_0 followed by a short answer a_0. The final-answer span is excluded from the probe input, and a deterministic validity check rejects refusals or outputs without an explicit reasoning trace. For valid outputs, SUREST accepts the answer only if

$$s(q, \{r_0\}) \le \tau_0. \tag{3}$$

If the check fails or the score exceeds τ_0, both r_0 and a_0 are discarded, and retrieval then starts from the original question, initializing E_0 with the top-k_0 BM25 passages ($k_0 = 3$).

Because the probe is trained on evidence contexts rather than question-only reasoning traces, we treat this gate as a conservative heuristic transfer rather than a calibrated sufficiency estimate. We therefore use a stricter threshold τ_0 for this pre-retrieval decision, where an incorrect acceptance would terminate retrieval before any external evidence is observed. We calibrate τ_E on the in-domain development splits and use the midpoint of the resulting operating points as a fixed threshold for the out-of-domain benchmarks. Threshold selection and sensitivity are detailed in Appendix B.5.

4.3 SUFFICIENCY-GUIDED EVIDENCE ACQUISITION

When the current evidence E_t remains insufficient, SUREST identifies the most salient unresolved information need conditioned on q and E_t, expresses it as an evidence need g_t, and converts it into a concise lexical query x_t for retrieval. BM25 uses x_t to construct a candidate pool C_t, while latent sufficiency determines which candidates are actually retained.

Sufficiency-margin gain (SMG). For each candidate c, we measure the reduction in latent insufficiency:

$$\Delta_{\mathrm{SMG}}(c \mid q, E_t) = s(q, E_t) - s\big(q, E_t \cup \{c\}\big), \tag{4}$$

where a positive value indicates that c moves the evidence state toward sufficiency. Because SMG is conditioned on E_t, after selecting $c^\star$ we update $E_t \leftarrow E_t \cup \{c^\star\}$ and recompute the gains of the remaining candidates. This rolling update discourages redundancy and captures state-dependent complementarity with the current evidence set.

Uncertainty-aware selection. We use the five probe seeds as an ensemble to stabilize candidate selection. Candidates with consistently positive gain are selected directly according to their mean SMG. When no candidate is confidently beneficial, a bounded lookahead evaluates whether uncertain candidates can jointly reduce insufficiency or reach the sufficiency threshold. Only the first passage of the best plan is committed, after which gains are recomputed against the updated evidence state. If the initial selection step retains no admissible passage, a one-shot query rewrite is used as fallback; otherwise, the next round regenerates the unresolved evidence need. Detailed voting rules, uncertainty criteria, and lookahead settings are provided in Appendix B.

4.4 SUFFICIENCY-GUIDED TERMINATION

SUREST checks the insufficiency score before each retrieval round and after every selected passage. Retrieval terminates as soon as

$$s(q, E_t) \le \tau_E, \tag{5}$$

after which the common final reader receives only the retained evidence set. Retrieval also stops when the retrieval, passage, or context budget is exhausted, when no unseen candidate is returned, or when no valid query can be produced. These are safety constraints; under normal operation, termination is governed by latent evidence sufficiency.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Datasets. We evaluate on four open-domain multi-hop benchmarks: HotPotQA (Yang et al., 2018), MuSiQue (Trivedi et al., 2022), IIRC (Ferguson et al., 2020), and 2WikiMultiHopQA (2Wiki) (Ho et al., 2020). We use the HotPotQA full-wiki validation split, the answerable MuSiQue development subset, the IIRC development split excluding `none`-type answers, and the 2Wiki development split. Our sufficiency probe is trained on paired sufficient–insufficient contexts constructed from the training splits of HotPotQA and MuSiQue, which we treat as in-domain; IIRC and 2Wiki serve as out-of-domain benchmarks. We sample 500 questions from each benchmark using the same random seed 42 and evaluate all methods on the identical sampled subsets.

Baselines. We compare against two non-adaptive controls (w/o retrieval and single-step retrieval), six adaptive methods (FLARE, DRAGIN, ETC, Adaptive-RAG, SeaKR, and Probing-RAG), and two iterative methods (S2G-RAG (Li et al., 2026b) and IRCoT). We follow the corresponding original implementations and describe method-specific settings in Appendix C.1.

Implementation. We use Qwen2.5-7B-Instruct (Qwen et al., 2025) as the common backbone and BM25 (Robertson & Zaragoza, 2009) over the DPR Wikipedia passage corpus (Karpukhin et al., 2020). All methods share global limits of five retrieval rounds, 15 retained passages, and 15,000 evidence tokens. For SUREST, we use the development-selected L21 probe described in Section 3, with operational thresholds detailed in Appendix B.5.

Table 2: QA performance and retrieval efficiency on four multi-hop QA datasets. In **(a)**, ΔF1 denotes the absolute macro-F1 improvement of SUREST over each baseline; best and second-best QA results are shown in **bold** and underline, respectively. In **(b)**, R and P denote retrieval rounds and retained passages, and Time is seconds per question. Macro values are unweighted averages across datasets; darker shading indicates larger F1 gains in (a) and larger cost reductions in (b).

(a) QA Performance

Method	In-domain				Out-of-domain				Macro		
	HotPotQA		MuSiQue		IIRC		2Wiki				
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	ΔF1
Non-adaptive baselines											
w/o retrieval	17.80	25.90	2.40	10.08	13.00	15.94	21.20	24.11	13.60	19.01	+14.24
Single-step retrieval	30.00	38.97	3.20	9.40	19.20	22.63	24.20	29.18	19.15	25.05	+8.20
Adaptive RAG											
FLARE	27.20	35.87	3.00	7.49	15.60	19.04	23.40	28.88	17.30	22.82	+10.43
DRAGIN	19.80	32.09	6.00	13.45	11.20	15.68	21.60	27.75	14.65	22.24	+11.01
ETC	27.40	39.98	6.40	14.87	10.60	15.81	20.60	27.48	16.25	24.53	+8.72
Adaptive-RAG	33.20	44.22	6.40	12.70	20.40	25.19	30.80	35.33	22.70	29.36	+3.89
SeaKR	25.20	34.69	4.20	9.79	13.80	17.43	18.40	23.65	15.40	21.39	+11.86
Probing-RAG	25.20	33.64	2.80	9.54	15.80	19.11	21.60	26.44	16.35	22.18	+11.07
Iterative RAG											
S2G-RAG	28.60	37.29	3.60	11.06	18.80	21.89	24.60	28.56	18.90	24.70	+8.55
IRCoT	34.60	46.00	7.20	13.54	23.00	27.52	31.60	36.48	24.10	30.89	+2.36
SUREST (Ours)	**36.60**	**48.28**	**11.40**	**17.65**	**23.40**	**27.61**	**33.40**	**39.46**	**26.20**	**33.25**	–

(b) Retrieval and Inference Cost

Method	In-domain				Out-of-domain				Macro		
	HotPotQA		MuSiQue		IIRC		2Wiki				
	R	P	R	P	R	P	R	P	R ↓	P ↓	Time ↓
Non-adaptive baselines											
w/o retrieval	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
Single-step retrieval	1.00	6.00	1.00	6.00	1.00	6.00	1.00	6.00	1.00	6.00	0.96
Adaptive RAG											
FLARE	1.51	4.49	1.60	4.80	1.57	4.64	1.50	4.42	1.54	4.59	0.89
DRAGIN	3.47	10.40	4.10	12.29	1.96	5.88	1.79	5.38	2.83	8.49	12.68
ETC	2.65	7.94	3.70	11.09	3.34	10.03	3.07	9.20	3.19	9.57	10.60
Adaptive-RAG	2.90	10.99	4.55	14.50	3.78	13.88	3.51	10.60	3.68	12.49	12.60
SeaKR	2.49	2.49	2.44	2.44	2.45	2.45	2.59	2.59	2.49	2.49	91.92
Probing-RAG	1.66	4.99	3.63	10.90	2.76	8.27	1.80	5.41	2.47	7.39	19.86
Iterative RAG											
S2G-RAG	4.31	12.93	4.95	14.84	4.76	14.29	4.95	14.86	4.74	14.23	16.56
IRCoT	4.94	14.81	4.97	14.90	4.97	14.90	4.97	14.90	4.96	14.88	17.23
SUREST (Ours)	2.95	5.38	4.51	8.54	3.85	6.83	2.82	5.02	3.53	6.44	7.99

5.2 MAIN RESULTS

Overall performance. Table 2 summarizes QA performance and retrieval efficiency. SUREST achieves the best EM and F1 on all four datasets, with a macro-average EM/F1 of 26.20/33.25 and a macro-F1 improvement of $+2.36$ points over IRCoT (95% CI: $[+0.73, +4.04]$). It surpasses Probing-RAG by 9.85 EM and 11.07 F1 points and S2G-RAG by 7.30 EM and 8.55 F1 points, showing that the gains arise from more than probing hidden states or explicitly judging sufficiency. Improvements over IRCoT are consistent across both in-domain and out-of-domain benchmarks, with particularly strong gains on MuSiQue and positive transfer to unseen datasets. This supports latent sufficiency as a robust control signal rather than a dataset-specific retrieval heuristic.

Better effectiveness–efficiency trade-off. Table 2 reports retrieval costs alongside QA performance, and Figure 1(a) visualizes the quality–efficiency trade-off. SUREST's accuracy gains do not

Table 3: Ablation of SUREST components and control signals. Parentheses show changes from the full model, computed before rounding (F1 in points; cost metrics in percent).

Variant	Rounds ↓	Passages ↓	Time (s) ↓	Macro F1 ↑
SUREST	3.53	6.44	7.99	33.25
Native readout (calibrated)	4.31 (+22.1%)	6.34 (−1.7%)	8.89 (+11.2%)	31.26 (−1.99)
w/o No-context Gate	3.83 (+8.6%)	7.01 (+8.8%)	7.77 (−2.7%)	32.71 (−0.53)
w/o SMG	3.73 (+5.7%)	11.60 (+80.1%)	9.81 (+22.8%)	31.52 (−1.73)
w/o Uncertainty-Aware Lookahead	3.58 (+1.5%)	6.02 (−6.6%)	7.16 (−10.4%)	32.35 (−0.90)
w/o Adaptive Termination	4.41 (+24.9%)	7.03 (+9.2%)	10.22 (+27.9%)	33.27 (+0.02)

rely on more retrieval. Compared with IRCoT, it improves macro F1 by 2.36 points while reducing retrieval rounds, retained passages, and inference time by 28.8%, 56.7%, and 53.6%, respectively. Compared with S2G-RAG, it improves macro F1 by 8.55 points with corresponding reductions of 25.5%, 54.7%, and 51.8%. Moreover, 91.3% of S2G-RAG examples exhaust the retrieval budget, suggesting that its explicit sufficiency judgment rarely enables early stopping. Although FLARE incurs lower absolute cost, its QA performance is substantially weaker. Overall, SUREST achieves a stronger quality–efficiency trade-off than the strongest iterative and sufficiency baselines.

6 ANALYSIS

6.1 ABLATION STUDY

Table 3 isolates the effects of the control signal, retrieval gate, SMG, lookahead, and termination on macro F1 and retrieval cost. Per-dataset results appear in Appendix C.3.

Probe-based control outperforms the native readout. Replacing the probe with a calibrated native sufficiency margin lowers macro F1 by 1.99 points and increases retrieval rounds and inference time by 22.1% and 11.2%, despite retaining 1.7% fewer passages. This variant also disables uncertainty-aware lookahead because the native readout has no five-seed ensemble. Removing lookahead alone causes a smaller 0.90-point F1 drop, suggesting that the larger degradation is not solely attributable to the absence of lookahead. Calibration details appear in Appendix C.3.1.

SMG and lookahead improve evidence selection. Replacing SMG with BM25 rank-order selection under the same per-round cap lowers macro F1 by 1.73 points and increases retained passages and inference time by 80.1% and 22.8%, showing that state-dependent sufficiency gain effectively filters redundant or unhelpful evidence. Removing lookahead lowers F1 by 0.90 points overall, with the largest drop on MuSiQue (1.84 points), suggesting that lookahead may be particularly useful for compositional multi-hop reasoning.

Gating and termination mainly control retrieval cost. Removing the question-only retrieval gate (No-context Gate) lowers F1 by 0.53 points and increases rounds and passages by 8.6% and 8.8%, although inference time falls by 2.7%. Thus, the gate trades additional question-only generation overhead for fewer retrieval operations, and may be omitted when retrieval is known a priori to be necessary. Without adaptive termination, F1 is essentially unchanged (+0.02), but rounds rise to 4.41 out of 5 and inference time increases by 27.9%; 80.5% of examples exhaust the retrieval budget. Further analysis appears in Appendix C.3.2.

6.2 ANALYSIS OF EVIDENCE ACQUISITION

Does SUREST acquire better evidence? To understand why SUREST achieves stronger QA performance with fewer retained passages, we evaluate the quality of its final evidence sets in Table 4. SUREST achieves the highest macro evidence-set F1 of 38.54. Compared with IRCoT, SUREST retains fewer than half as many pages (4.58 vs. 9.51), with only modest reductions in gold-evidence recall (52.36 vs. 54.28) and complete-evidence recovery (32.14 vs. 35.06). Compared with Probing-RAG, SUREST substantially improves recall (52.36 vs. 33.81) and complete-evidence

Table 4: Page-level quality of final evidence sets on the common-retrieval subset. Best and second-best results are shown in **bold** and underline, respectively.

Method	Evidence-set F1 ↑				Macro				
	HotPot	MuSiQue	IIRC	2Wiki	Prec. ↑	Rec. ↑	F1 ↑	Complete ↑	Pages ↓
Adaptive-RAG	22.38	15.66	21.18	25.89	13.77	53.59	21.28	33.62	9.43
IRCoT	22.46	15.48	21.20	25.96	13.69	**54.28**	21.27	**35.06**	9.51
Probing-RAG	39.09	**27.76**	33.24	42.62	**41.36**	33.81	35.68	5.24	**2.08**
S2G-RAG	33.36	22.66	32.34	31.12	33.55	31.51	29.87	10.68	2.34
SUREST	**43.03**	25.46	**35.78**	**49.88**	33.30	52.36	**38.54**	32.14	4.58

Table 5: Evidence-selection quality under incomplete gold coverage. Random reports the expected performance under uniform random selection.

Strategy	Missing-gold hit(%) ↑	Recall gain(pp) ↑	Complete gold(%) ↑
Random	5.21	2.35	3.01
BM25 top-1	8.37	3.81	5.21
SUREST	**10.91**	**4.95**	**6.67**

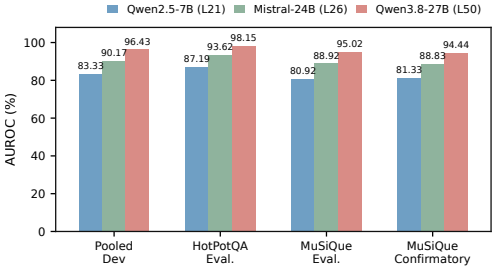


Figure 4: Cross-backbone probe AUROC on sufficiency discrimination.

recovery (32.14 vs. 5.24), showing that deciding whether retrieval is needed alone is insufficient for assembling complete multi-hop evidence. A similar pattern emerges against S2G-RAG: despite comparable precision (33.30 vs. 33.55), SUREST achieves much higher recall (52.36 vs. 31.51) and complete recovery (32.14 vs. 10.68). This suggests that explicit sufficiency and gap judgments alone do not ensure effective evidence accumulation.

Does SMG acquire more useful evidence? Table 5 compares 6,580 steps with incomplete gold coverage, holding the evidence state and candidate pool fixed. SUREST selects a missing gold page more often than BM25 top-1 (10.91% vs. 8.37%; +2.54 points, 95% CI [1.82, 3.24]) and yields greater recall gain (4.95 vs. 3.81 points). Notably, only 17.28% of candidate pools contain a missing gold page; when available, SUREST selects it in 63.15% of cases versus 48.46% for BM25. SMG improves selection, while candidate availability remains the main bottleneck.

6.3 Generalization across Backbones

To test cross-backbone transfer, we evaluate SUREST on Mistral-Small-24B-Instruct-2501 and Qwen3.8-27B (Qwen Team, 2026), reusing thresholds selected on Qwen2.5-7B without recalibration. Across Qwen2.5-7B, Mistral-24B, and Qwen3.8-27B, SUREST achieves macro F1 scores of 33.25, 40.90, and 49.29, exceeding the best evaluated baselines by 2.36, 1.72, and 7.99 points, respectively. Figure 4 shows consistently higher linear-probe AUROC for the larger backbones across all reported settings, suggesting that evidence sufficiency is more linearly separable in their hidden representations. Additional results appear in Appendix C.5.

7 Conclusion

We study adaptive retrieval through the lens of *evidence sufficiency* and show that explicit sufficiency decisions can diverge from the information encoded in language-model hidden representations. We introduce SUREST, which uses latent sufficiency to control retrieval triggering, evidence selection, and termination. Across four multi-hop QA benchmarks and multiple model backbones, SUREST outperforms all the evaluated methods while substantially reducing retrieval cost. These results suggest that latent evidence sufficiency can guide retrieval toward better answers at lower cost.

REFERENCES

Ingeol Baek, Hwan Chang, ByeongJeong Kim, Jimin Lee, and Hwanhee Lee. Probing-RAG: Self-probing to guide language models in selective document retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3287–3304, Albuquerque, New Mexico, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-naacl.181. URL https://aclanthology.org/2025.findings-naacl.181/.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

James Ferguson, Matt Gardner, Hannaneh Hajishirzi, Tushar Khot, and Pradeep Dasigi. IIRC: A dataset of incomplete information reading comprehension questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 1137–1147, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.86. URL https://aclanthology.org/2020.emnlp-main.86/.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7036–7050, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.389. URL https://aclanthology.org/2024.naacl-long.389/.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.495. URL https://aclanthology.org/2023.emnlp-main.495/.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval-augmented generation systems. In *International Conference on Learning Representations (ICLR)*, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Dahyun Lee, Yongrae Jo, Haeju Park, and Moontae Lee. Shifting from ranking to set selection for retrieval augmented generation. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 17606–17619, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.861. URL https://aclanthology.org/2025.acl-long.861/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Bo Li, Tian Tian, Zhenghua Xu, Hao Cheng, Shikun Zhang, and Wei Ye. Modeling uncertainty trends for timely retrieval in dynamic RAG. In *Proceedings of the AAAI Conference on Artificial*

Intelligence, pp. 31527–31535, 2026a. doi: 10.1609/aaai.v40i37.40418. URL https://ojs.aaai.org/index.php/AAAI/article/view/40418.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Minghan Li, Junjie Zou, Xinxuan Lv, Chao Zhang, and Guodong Zhou. S2G-RAG: Structured sufficiency and gap judging for iterative retrieval-augmented QA. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25846–25862, San Diego, California, United States, July 2026b. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.1185.

Shiyang Li, Jun Yan, Hai Wang, Zheng Tang, Xiang Ren, Vijay Srinivasan, and Hongxia Jin. Instruction-following evaluation through verbalizer manipulation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 3678–3692, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.233.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Stephen E. Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. DRAGIN: Dynamic retrieval augmented generation based on the real-time information needs of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 12991–13013, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.702. URL https://aclanthology.org/2024.acl-long.702/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 10014–10037, Toronto, Canada, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Keheng Wang, Feiyu Duan, Peiguang Li, Sirui Wang, and Xunliang Cai. LLMs know what they need: Leveraging a missing information guided framework to empower retrieval-augmented generation. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 2379–2400, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.163/.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Zijun Yao, Weijian Qi, Liangming Pan, Shulin Cao, Linmei Hu, Weichuan Liu, Lei Hou, and Juanzi Li. SeaKR: Self-aware knowledge retrieval for adaptive retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pp. 27022–27043, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-long.1312. URL https://aclanthology.org/2025.acl-long.1312/.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. Making retrieval-augmented language models robust to irrelevant context. In *International Conference on Learning Representations*, 2024.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12697–12706. PMLR, 2021.

AI USE STATEMENT

In this work, we used generative AI tools to aid and polish the writing, assist in identifying potentially relevant prior work, and reproduce the S2G-RAG baseline following its original training procedure, where GPT-4o-mini is used to construct turn-level supervision. Generative AI was not used in the design, training, or evaluation of SUREST. All AI-assisted content, including suggested references, was independently reviewed and verified by us. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies retrieval control for language models using publicly available benchmarks, models, and retrieval corpora. It does not involve human-subject experiments, private or personally identifiable data, or the collection of new user data. Our experiments focus on question answering and retrieval efficiency and do not introduce new capabilities intended for harmful or high-risk applications. We report experimental settings and evaluation procedures to support transparency and reproducibility.

REPRODUCIBILITY STATEMENT

We provide detailed experimental settings, probe construction, retrieval procedures, threshold selection, and additional analyses in the main paper and appendices. As supplementary material, we will submit selected code covering probe training and the core SUREST implementation. The complete codebase will be released subsequently. Together, these materials are intended to support reproduction of our main experiments and analyses.

A SUFFICIENCY PROBE DETAILS

This appendix provides the data construction, representation extraction, and training details for the sufficiency probe used by SUREST. Unless stated otherwise, all settings refer to Qwen2.5-7B-Instruct. The final system uses a five-seed ensemble of linear probes trained on L21 hidden states extracted at the decision position.

A.1 PAIRED SUFFICIENCY DATA CONSTRUCTION

Pair definition. Each example consists of a question q_i and paired contexts (C_i^+, C_i^-). The sufficient context C_i^+ contains the evidence needed to support a complete answer path, whereas the insufficient context C_i^- lacks at least one required supporting hop. We assign labels $y_i^+ = 0$ and $y_i^- = 1$, so larger probe logits indicate greater evidence insufficiency. Because the two contexts share the same question and answer, the paired construction controls for question-level variation and isolates differences in evidence completeness.

HotPotQA. We construct HotPotQA pairs from the distractor training data after first partitioning questions into disjoint train, development, and evaluation pools. For each question, C_i^+ is the original context. To construct C_i^-, we replace one gold supporting paragraph with a length-matched non-supporting paragraph drawn from another question in the same split. Replacement passages must introduce a new title and must not contain the normalized gold answer. This preserves context length and topical plausibility while removing one required supporting hop. Replacement donors never cross split boundaries.

To reduce cases answerable from parametric knowledge or residual evidence, we filter pairs using Qwen2.5-7B-Instruct. We query the model both without context and with only C_i^-. A pair is discarded if either prediction matches the gold answer exactly after normalization or achieves token-level F1 of at least 0.5. We do not filter based on model performance on C_i^+.

MuSiQue. MuSiQue provides paired answerable and unanswerable records with the same example identifier. We use the corresponding contexts as C_i^+ and C_i^- after verifying that the paired records share the same question and answer. We similarly filter questions that can be answered reliably without context. The train, development, evaluation, and independent confirmatory pools are sampled without replacement, kept disjoint by question identifier, and frozen before representation extraction. The confirmatory set is never used for prompt revision, layer selection, threshold calibration, or architecture selection.

A.2 Prompting and Representation Extraction

Sufficiency-decision prompt. All inputs are rendered using the Qwen chat template. The system instruction is:

> *You judge whether the context contains enough information to answer the question. Follow the label mapping in the user message and output exactly one label token.*

The baseline user prompt has the following structure:

> `Question:` q
>
> `Context:` C
>
> `A` = [semantic label for A]
> `B` = [semantic label for B]
>
> `Return exactly A or B.`
> `Label:`

We use both label mappings,

`A=Sufficient/B=Insufficient` and `A=Insufficient/B=Sufficient`.

The display order and output instruction remain fixed; only the semantic assignments of A and B are swapped.

In addition to the baseline prompt, we construct three additional instruction variants. Each prepends the common instruction "Decide whether the current context provides a sufficient basis for answering the question reliably," followed by one of the following sentences:

- **Neutral-standard:** The context meets this standard when it provides adequate support for a reliable answer.
- **Proceed-oriented:** The context meets this standard when any missing information would not materially affect the reliability of the answer.
- **Retrieval-oriented:** The context meets this standard only when the important facts and reasoning connections needed for a reliable answer are established by the context.

Pooling examples across the four instruction variants and both label mappings reduces the risk that the probe exploits prompt-specific wording or a fixed association between semantic classes and output labels. For downstream SUREST inference, we use a single fixed prompt configuration: the baseline template with A=SUFFICIENT and B=INSUFFICIENT. We extract the L21 decision-position

Table 6: Numbers of question-disjoint sufficiency pairs in each split.

Dataset	Train	Dev	Eval.	Confirmatory
HotPotQA	5,000	500	500	–
MuSiQue	1,200	150	150	500

representation from this prompt and average the logits of the five independently trained probes; we do not ensemble across prompt variants or label mappings at inference.

Hidden-state extraction. Qwen2.5-7B-Instruct has 28 transformer blocks with hidden size 3,584. For each rendered prompt, we perform a forward pass and extract representations from the final 15 layers, L14–L28. For consistency with the model's final readout space, we apply the model's final normalization module to each captured layer representation before probe preprocessing.

At each layer, we use the hidden state at the *decision position*, defined as the final token of the rendered prompt at `Label:`, immediately before the model predicts A or B. This representation is used for layer selection, representation analysis, and SUREST control. Probe supervision is determined solely by whether the corresponding context is constructed as sufficient or insufficient; the model's own A/B prediction is not used as a training target.

A.3 Probe Training and Optimization

Preprocessing and domain balancing. For each layer, we estimate a per-dimension mean $\boldsymbol{\mu}_\ell$ and standard deviation $\boldsymbol{\sigma}_\ell$ using only the domain-balanced training bank. The standardized representation is

$$\widetilde{\mathbf{h}}^{(\ell)} = \mathrm{clip}\left(\frac{\mathbf{h}^{(\ell)} - \boldsymbol{\mu}_\ell}{\max(\boldsymbol{\sigma}_\ell, \varepsilon_{\mathrm{num}})}, -10, 10\right), \qquad \varepsilon_{\mathrm{num}} = 10^{-5}, \tag{6}$$

where the maximum and clipping operations are applied elementwise. The training statistics $(\boldsymbol{\mu}_\ell, \boldsymbol{\sigma}_\ell)$ are then fixed and reused for all development, held-out, confirmatory, and downstream inference examples.

Because the HotPotQA training bank is substantially larger than the MuSiQue bank, directly pooling the two would cause HotPotQA to dominate the objective. We therefore repeat the complete MuSiQue training bank four times, yielding 4,800 MuSiQue pair copies alongside 5,000 HotPotQA pairs. For early stopping, the MuSiQue development bank is repeated three times, yielding 450 MuSiQue pairs alongside 500 HotPotQA pairs. Repetition is applied after expansion over context, instruction, and label-mapping conditions, preserving the same condition distribution within each domain.

Objective. For each layer ℓ and random seed r, we fit an affine classifier $s_{\ell,r}$ as defined in Eq. 1. Let $\mathcal{D}^{(\ell)}_{\mathrm{bal}}$ denote the corresponding domain-balanced training bank. We minimize binary cross-entropy with logits:

$$\mathcal{L}_{\ell,r} = -\frac{1}{|\mathcal{D}^{(\ell)}_{\mathrm{bal}}|} \sum_{(\widetilde{\mathbf{h}},y)\in\mathcal{D}^{(\ell)}_{\mathrm{bal}}} \big[y \log \sigma(s_{\ell,r}(\widetilde{\mathbf{h}})) + (1-y)\log(1-\sigma(s_{\ell,r}(\widetilde{\mathbf{h}})))\big]. \tag{7}$$

We optimize each probe with AdamW. Within each layer and seed, we retain the checkpoint with the lowest domain-balanced development BCE. Training stops after ten consecutive epochs without improvement or after 100 epochs. Layer selection is performed separately using development AUROC. Table 7 summarizes the complete optimization configuration.

Table 7: Training configuration of the linear sufficiency probe.

Hyperparameter	Value
Backbone	Qwen2.5-7B-Instruct
Probe input dimension	3,584
Probe architecture	Linear ($3{,}584 \to 1$)
Layers trained	L14–L28
Optimizer	AdamW
Learning rate	1×10^{-3}
Weight decay	1×10^{-4}
Batch size	256
Maximum epochs	100
Early-stopping patience	10
Probe seeds	0, 1, 2, 3, 4
Training objective	Binary cross-entropy

Table 8: Linear and nonlinear probe results on the independent MuSiQue confirmatory set. Pair denotes paired ranking accuracy.

Probe	Layer	AUROC ↑	BAcc. ↑	Pair ↑
Linear	21	0.813	0.735	0.841
MLP	21	0.818	0.738	0.847

A.4 Ensemble Construction and Layer Selection

We train five independently initialized linear probes at each candidate layer and average their logits at inference:

$$\bar{s}_\ell(q, C) = \frac{1}{R} \sum_{r=1}^{R} s_{\ell,r}(q, C), \qquad R = 5. \tag{8}$$

We first construct a five-seed ensemble for each layer and select the final layer using only development data:

$$\ell^* = \arg \max_{\ell \in \{14,\ldots,28\}} \frac{1}{2} \left(\mathrm{AUROC}^{\mathrm{HP}}_{\mathrm{dev},\ell} + \mathrm{AUROC}^{\mathrm{MQ}}_{\mathrm{dev},\ell} \right). \tag{9}$$

This criterion selects L21, whose ensemble achieves development AUROC values of 0.873 on HotPotQA and 0.785 on MuSiQue, yielding a mean selection score of 0.829. For seeds 0–4, the selected L21 checkpoints occur at epochs 1, 2, 2, 1, and 2, respectively. No held-out evaluation or confirmatory example is used for checkpoint or layer selection.

For the representation analyses in Section 3, we use zero as the semantic decision threshold, classifying a context as insufficient when the ensemble score is greater than zero. This threshold is used only for representation-level analysis and is distinct from the operational thresholds used by SUREST for question-only gating and retrieval termination. The latter are calibrated with the end-to-end system and described in Appendix B.5.

A.5 Additional Probe Results

A.5.1 Linear versus Nonlinear Probes

To test whether the sufficiency signal requires a nonlinear decoder, we train an MLP only at L21 after this layer has already been selected using the linear probes and development data. The MLP has architecture $3{,}584 \to 64 \to 1$, with a GELU activation and dropout of 0.2, and follows the same preprocessing, training data, optimizer, random seeds, and early-stopping protocol as the linear probe. No additional layer search or architecture selection is performed on the confirmatory set.

The MLP improves confirmatory AUROC by only 0.006, balanced accuracy by 0.003, and paired ranking accuracy by 0.006. This marginal gain suggests that the relevant sufficiency information is

Table 9: Robustness across all eight instruction–mapping conditions on the independent MuSiQue confirmatory set. "Sufficient" indicates whether A or B is mapped to the semantic label *Sufficient*; the other token is mapped to *Insufficient*. BAcc. denotes balanced accuracy at a zero score threshold, and Pair denotes paired ranking accuracy. The pooled row aggregates 8,000 context instances and 4,000 paired comparisons from 500 question pairs across all eight conditions.

		Native LM head (L28)			Linear probe (L28)			Linear probe (L21)		
Prompt variant	Sufficient	AUROC	BAcc.	Pair	AUROC	BAcc.	Pair	AUROC	BAcc.	Pair
Baseline	A	.699	.596	.734	.801	.735	.830	.812	.738	.834
Baseline	B	.683	.550	.722	.793	.719	.814	.808	.732	.824
Neutral-standard	A	.689	.560	.740	.803	.736	.838	.820	.739	.834
Neutral-standard	B	.683	.515	.718	.800	.720	.824	.811	.734	.856
Proceed-oriented	A	.684	.565	.724	.791	.719	.816	.809	.730	.846
Proceed-oriented	B	.674	.511	.702	.793	.714	.816	.802	.724	.828
Retrieval-oriented	A	.688	.547	.734	.800	.720	.806	.821	.745	.864
Retrieval-oriented	B	.680	.506	.716	.802	.719	.818	.816	.735	.840
All conditions (pooled)		.678	.544	.724	.797	.723	.820	.813	.735	.841

Table 10: Representation–output alignment on the MuSiQue confirmatory set, conditioned on whether the model answers the sufficient context C^+ correctly. "$C^+ \rightarrow$Insuff." denotes the fraction of sufficient contexts classified as *Insufficient* by the native readout.

Group	# Questions	$C^+ \rightarrow$Insuff. (%)	Native AUROC	L21 Probe AUROC
All	500	85.8	0.678	0.813
EM correct	148	83.9	0.735	0.870
EM incorrect	352	86.6	0.654	0.787

Prompt variant	$C^+ \rightarrow$Insuff. (%)	
	EM correct	EM incorrect
Baseline	73.6	75.7
Neutral-standard	85.5	88.8
Proceed-oriented	85.8	89.3
Retrieval-oriented	90.5	92.8

already largely accessible through a linear readout. We therefore retain the linear probe in SUREST, avoiding unnecessary decoder complexity.

A.5.2 Stability across Prompts, Mappings, and Seeds

Table 9 shows that the advantage of the learned readout is consistent across all eight instruction–mapping conditions. In particular, the L21 ensemble achieves AUROC values between 0.802 and 0.821, while swapping the A/B semantic mapping does not materially change performance.

A.5.3 Misalignment Conditional on Answer Correctness

Table 10 tests whether the native insufficiency bias can be explained simply by limited answering ability. Even among the 148 questions that the model answers correctly from C^+, the native readout still classifies 83.9% of sufficient contexts as *Insufficient*. The bias also persists across all prompt variants, with sufficient-context error rates of 73.6–90.5% in the EM-correct group. Although native AUROC improves from 0.654 on EM-incorrect examples to 0.735 on EM-correct examples, it remains well below the L21 probe (0.870). Thus, answer failure contributes to the difficulty of sufficiency judgment but does not account for the systematic representation–output mismatch.

A.5.4 Additional Representation–Output Alignment

Table 11 shows that the representation–output mismatch also holds on the held-out HotPotQA evaluation split. At the same L28 representation, the linear probe improves AUROC from 0.679 to 0.850 and balanced accuracy from 0.612 to 0.768, while the development-selected L21 probe further reaches 0.872 AUROC and 0.975 paired ranking accuracy. This confirms that the mismatch observed on MuSiQue is not specific to the confirmatory dataset.

Table 11: Representation–output alignment on the held-out HotPotQA evaluation split. BAcc. denotes balanced accuracy at a zero semantic-margin threshold, and Pair denotes paired ranking accuracy.

Readout	Layer	AUROC ↑	BAcc. ↑	C^+ Acc.	C^- Acc.	Pair ↑
Native LM head	28	0.679	0.612	0.439	0.786	0.826
Linear probe	28	0.850	0.768	0.759	0.776	0.956
Linear probe	21	**0.872**	**0.791**	**0.788**	**0.795**	**0.975**

Table 12: Qwen2.5-7B-Instruct configuration used in the main SUREST experiments. HP and MQ denote HotPotQA and MuSiQue.

Component	Configuration
Initial retrieval	Original-question BM25, top 3
Targeted retrieval	BM25, top 10 candidates
Retrieval budget	5 calls
Evidence budget	15 passages; 15,000 context tokens
Per-round additions	At most 5 passages
Sufficiency estimator	Five-seed L21 linear-probe ensemble
Candidate scoring	Batched KV-cache reuse; batch size 5
Confident-positive gate	Mean SMG > 0 and $\geq 3/5$ positive votes
Uncertainty boundary	$\epsilon = -0.2$
Uncertain utility	$0.7\times$ normalized mean SMG $+ 0.3\times$ normalized $(-\mathrm{BM25\ rank})$
Lookahead	Beam width 3; depth 2; fallback set-gain threshold 0.5
Query rewrite	At most one rewrite after a failed retrieval step
No-context threshold	$\tau_0 = -5$
Evidence thresholds	HP -2; MQ -4; IIRC -3; 2Wiki -3
Decoding	Greedy (no sampling)

B SUREST IMPLEMENTATION DETAILS

This appendix provides the implementation details omitted from Section 4. Unless otherwise stated, all settings are shared across the four datasets. Table 12 summarizes the main configuration used with Qwen2.5-7B-Instruct.

B.1 COMPLETE RETRIEVAL PROCEDURE

Algorithm 1 gives the complete inference procedure of SUREST, including retrieval triggering, sufficiency-guided evidence selection, adaptive termination, and query-rewrite fallback.

“‘latex “‘

B.2 RETRIEVAL INITIALIZATION AND TARGETED QUERYING

Question-only retrieval gate. Before retrieval, the model produces a reasoning trace r_0 and a short answer a_0 from the question alone using the following zero-shot template:

```
Question: <question>
Reason explicitly before giving the final answer. If a
required fact is uncertain, say so instead of guessing.
Reasoning: <reasoning>
Final answer: <short answer, or Unknown>
```

The answer span is removed before probe scoring, so only r_0 is used as temporary context. Refusals, empty reasoning traces, or missing answers automatically trigger retrieval. Otherwise, the answer is accepted only when $s(q, \{r_0\}) \leq \tau_0$. Rejected reasoning and answers are discarded and do not enter the retrieved context. Because the probe is trained on evidence contexts rather than question-

only reasoning traces, we treat this use as a conservative heuristic transfer rather than a calibrated sufficiency estimate.

Initial retrieval. If retrieval is triggered, the original question is used as a BM25 query and the top three passages initialize E_0. We do not apply SMG selection in this round because no retrieved evidence state has yet been established. E_0 is then evaluated against the evidence-sufficiency threshold.

Targeted query generation. If E_t remains insufficient, the model receives q, the retained evidence E_t, and the previous query history, and generates a query targeting the most important unresolved evidence need:

```
Identify the single most important missing fact or reasoning
connection needed to answer the question reliably from the
current evidence.  Produce a concise lexical search query
that can find that fact.  Return one JSON object with keys
''missing_fact'' and ''query'' only.
```

The output is parsed as (g_t, x_t), and BM25 retrieves the top ten not already retained passages using x_t. These passages form the candidate pool; BM25 relevance does not determine which passages are ultimately retained.

Repeated or empty queries are not executed directly. Instead, the model is allowed one rewrite; if the rewritten query is non-empty and unseen, retrieval continues with the rewritten query. Otherwise, retrieval terminates. A rewrite may also be triggered after retrieval when BM25 returns a non-empty candidate pool but the sufficiency-guided selector retains no admissible passage. If retrieval budget remains, the rewritten query is executed in the next retrieval round.

B.3 Sufficiency-Guided Evidence Selection with SMG

Let E denote the retained evidence and C the remaining candidates in the current retrieval round. For each $c \in C$, we compute the member-wise Sufficiency Margin Gain

$$\Delta_r(c \mid q, E) = s_r(q, E) - s_r\big(q, E \cup \{c\}\big), \tag{10}$$

where s_r is the insufficiency score of ensemble member r. The mean gain is

$$\bar{\Delta}(c \mid q, E) = \frac{1}{5} \sum_{r=1}^{5} \Delta_r(c \mid q, E), \tag{11}$$

which is exactly the SMG in Eq. 4 because the ensemble score is the mean of the five member scores.

We also count the number of ensemble members assigning positive gain,

$$v^+(c \mid E) = \sum_{r=1}^{5} \mathbb{I}\left[\Delta_r(c \mid q, E) > 0\right]. \tag{12}$$

A candidate is *confident-positive* when

$$\bar{\Delta}(c \mid q, E) > 0 \quad \wedge \quad v^+(c \mid E) \geq 3. \tag{13}$$

If any confident-positive candidate exists, we select the one with the largest mean SMG, using BM25 score only as a deterministic tie-breaker. After each selection, the passage is added to E and all remaining gains are recomputed. This rolling update keeps candidate utility conditional on the current evidence state and discourages redundant acquisition.

Selection within a retrieval round stops when the evidence becomes sufficient, five passages have been added, a global context budget is reached, or no admissible candidate remains. If the evidence remains insufficient, control returns to the outer loop, which either invokes a permitted rewrite fallback or generates a new targeted query.

Table 13: Effect of full-prefill recomputation.

Variant	Macro EM ↑	Macro F1 ↑	Rounds ↓	Passages ↓	Time (s) ↓
SuREST	26.20	33.25	3.53	6.44	7.99
Full-prefill	26.10	32.91	3.50	6.31	27.33
Change	−0.10	−0.34	−1.0%	−2.1%	+242.0%

Efficient conditional scoring. To avoid recomputing the shared question–evidence prefix for every candidate, we reuse its attention KV cache and score candidate suffixes in micro-batches of five. The cache is refreshed after each committed passage. This optimization leaves the selection rule unchanged; selection-time scores may differ slightly from separate full-input reevaluations used for stopping checks.

To quantify its computational benefit, we additionally evaluate a full-prefill variant that recomputes the complete accumulated context at each scoring step. Table 13 shows that Full-prefill yields nearly identical QA performance and retrieval behavior, with Macro F1 changing from 33.25 to 32.91, average retrieval rounds from 3.53 to 3.50, and retained passages from 6.44 to 6.31. In contrast, average inference time increases from 7.99s to 27.33s, accompanied by substantially more input tokens and forward calls. These results indicate that cache reuse primarily reduces computation without materially altering SuREST's retrieval behavior.

B.4 Uncertainty-Aware Lookahead and Query Fallback

When no confident-positive candidate exists, we retain candidates with near-nonnegative SMG for bounded lookahead. Let $\epsilon = -0.2$ and define

$$v^{\epsilon}(c \mid E) = \sum_{r=1}^{5} \mathbb{I}[\Delta_r(c \mid q, E) > \epsilon]\,. \tag{14}$$

A non-confident-positive candidate is treated as *uncertain* when

$$\bar{\Delta}(c \mid q, E) > \epsilon \quad \vee \quad v^{\epsilon}(c \mid E) \geq 2. \tag{15}$$

All remaining candidates are excluded.

We rank uncertain candidates by

$$u(c \mid E) = \alpha\, \mathcal{N}_C\big(\bar{\Delta}(c \mid q, E)\big) + (1-\alpha)\, \mathcal{N}_C(-\operatorname{rank}_{\mathrm{BM25}}(c))\,, \tag{16}$$

with $\alpha = 0.7$, where $\mathcal{N}_C$ denotes min–max normalization over the uncertain subset.

We run beam search from the three highest-utility candidates with beam width three and maximum depth two. Each expansion recomputes the insufficiency score under the updated evidence state. We first prefer the shortest plan that reaches the stopping threshold. Otherwise, a two-passage plan is accepted only if

$$\Delta_{\mathrm{set}}(A \mid q, E) = s(q, E) - s(q, E \cup A) > \delta, (\delta = 0.5). \tag{17}$$

Only the first passage of the selected plan is committed, after which all gains are recomputed. If no acceptable plan is found before any passage is committed in the round, the query may be rewritten once, provided that retrieval budget remains.

Normalization and deterministic tie-breaking. Min–max normalization is applied separately within each eligible candidate group to conditional gains and negative retrieval ranks. If the within-group range is at most 10^{-12}, including singleton groups, all normalized values are set to zero.

For lookahead plans that reach the sufficiency threshold, we prefer shorter plans, breaking ties by lower terminal insufficiency score, higher mean stepwise utility, and lexicographically smaller passage-ID sequences. If no plan reaches sufficiency, we consider only plans of length at least two whose total gain exceeds $\delta + 10^{-12}$, ranking them by lower terminal insufficiency score, shorter length, higher mean utility, and passage-ID lexicographic order. If no plan qualifies, no candidate is selected. Only the first passage of the selected plan is committed, after which scores are recomputed.

Table 14: Sensitivity to the retrieved-evidence threshold τ_E on the in-domain development sets. † marks the operating point used in the main experiments.

Dataset	τ_E	F1 ↑	Rounds ↓	Passages ↓
HotPotQA	-1	37.47	2.58	5.08
	$-2^{\dagger}$	40.55	3.03	5.62
	-3	41.02	3.48	6.15
	-4	41.23	3.88	6.59
	-5	41.37	4.26	6.85
MuSiQue	-1	12.31	3.43	6.51
	-2	13.61	3.84	7.23
	-3	16.17	4.17	7.63
	$-4^{\dagger}$	16.49	4.37	7.91
	-5	16.17	4.51	8.02

B.5 Threshold Calibration and Sensitivity

The probe outputs an insufficiency margin, so a more negative threshold imposes a stricter criterion for declaring the current evidence sufficient. We fix the question-only threshold to $\tau_0 = -5$ for all datasets and do not tune it separately.

We analyze the retrieved-evidence threshold τ_E on the previously defined in-domain development splits from HotPotQA and MuSiQue (Appendix A.1). Table 14 shows the resulting quality–efficiency trade-off. On HotPotQA, most of the accuracy gain is obtained by moving from -1 to -2; stricter thresholds provide only modest additional F1 while substantially increasing retrieval cost. We therefore use $\tau_E = -2$. MuSiQue benefits more from stricter stopping and reaches its highest development F1 at -4, which we use as its operating point.

For the out-of-domain benchmarks, we do not tune τ_E. Instead, we use the midpoint of the two in-domain operating points,

$$\tau_E^{\mathrm{OOD}} = \frac{-2 + (-4)}{2} = -3, \tag{18}$$

for both IIRC and 2WikiMultiHopQA.

Overall, stricter thresholds increase retrieval cost, but their marginal quality benefit quickly diminishes. In particular, tightening HotPotQA from -2 to -5 improves F1 by only 0.82 points while increasing retrieval rounds from 3.03 to 4.26.

Termination and safety stops. The learned termination condition is checked before every new retrieval call and after every committed passage. Independent hard stops are triggered by the five-call retrieval budget, the passage or token budget, empty retrieval, duplicate-query exhaustion, or failure to produce an admissible query. These conditions prevent unbounded execution but are not interpreted as evidence sufficiency.

C Experimental Details

C.1 Baseline Implementation Details

Unless otherwise specified, all methods use Qwen2.5-7B-Instruct as the reasoning and answer-generation backbone and retrieve from the same DPR Wikipedia corpus with BM25. We impose a common budget of at most five retrieval calls, 15 accumulated evidence passages, and 15,000 context tokens. Native triggering, query construction, and evidence-management rules are preserved whenever compatible with these constraints. Decoding is greedy and deterministic, except for SeaKR's uncertainty sampling, for which we fix the random seed to 42.

FLARE (Jiang et al., 2023) uses a 64-token lookahead, triggering retrieval when any lookahead token has probability at most 0.8 and masking tokens below 0.4 when forming the query. DRAGIN (Su et al., 2024) uses a RIND threshold of 2.4 and the top 25 attention-selected content words, while

ETC (Li et al., 2026a) uses an uncertainty-trend threshold of 1.0 and the top 35 query terms; both use the official part-of-speech filter. Adaptive-RAG (Jeong et al., 2024) uses a fine-tuned T5-large router to select no/single-step/iterative retrieval; the single-step branch retrieves 15 passages, while the iterative branch follows IRCoT (Trivedi et al., 2023) with three passages per round.

SeaKR (Yao et al., 2025) follows the official controller with decoder layer 15, 20 sampled completions, an eigen-score threshold of -6, and at most five selected documents. Probing-RAG (Baek et al., 2025) follows the official four-shot procedure, retrieves three passages per round, and uses a probe threshold of zero. Its model-specific probes are trained on the same frozen training-question splits as our sufficiency probe. Probing-RAG constructs its own training trajectories rather than reusing our sufficiency-probe instances.

S2G-RAG (Li et al., 2026b) follows the official setup with a Llama-3.2-3B-Instruct model as the S2G-Judge, which explicitly assesses evidence sufficiency and generates evidence gaps when further retrieval is required. Following the original training procedure, we fine-tune the judge on HotPotQA and MuSiQue trajectories, with turn-level states evaluated by GPT-4o-mini to construct supervision. At inference time, we preserve the official sufficiency-and-gap control procedure while using top-3 BM25 retrieval and the shared five-call budget.

FLARE, DRAGIN, ETC, and SeaKR retain their native answer-generation procedures. Adaptive-RAG, IRCoT, Probing-RAG, and S2G-RAG use the same final-reader prompt as SUREST, conditioned on the evidence retained by their respective controllers.

C.2 Inference-Time Measurement

All reported latency experiments are conducted on dedicated NVIDIA H20 GPUs with batch size 1, using the corresponding backbone in bfloat16 under PyTorch 2.4.0, CUDA 12.4, and Transformers 4.45.2.

We report mean per-question wall-clock time over 500 examples per dataset. Timing covers the complete per-example RAG pipeline, including retrieval triggering, query generation, local BM25 retrieval, probe and candidate scoring, iterative reasoning, and final answer generation. All methods query the same local Elasticsearch index. Model loading, retriever initialization, and result serialization are excluded. No separate warm-up examples are discarded.

C.3 Per-Dataset Ablation Results

Table 15 provides per-dataset results for the main ablations. Values in parentheses denote changes relative to the full SUREST model; F1 changes are absolute points, while efficiency changes are relative percentages.

The calibrated native-readout variant reduces F1 on all four datasets while increasing retrieval rounds on all four, with similar overall passage usage, indicating less effective retrieval control rather than a simple increase in retained evidence. The per-dataset results clarify the distinct roles of the controller components. Removing SMG consistently increases evidence consumption, with passage counts rising by 66.5–88.3%, while reducing F1 on all four datasets, highlighting the importance of state-dependent evidence selection. Removing uncertainty-aware lookahead causes its largest degradation on MuSiQue (-1.84 F1), consistent with its role in recovering complementary evidence for harder multi-hop questions. The no-context gate mainly reduces unnecessary retrieval, with the largest effect on 2Wiki, while adaptive termination substantially reduces retrieval cost with almost no change in macro F1.

C.3.1 Native-readout ablation

We further test whether the model's native sufficiency readout can replace the linear probe at inference time. Under the same sufficiency-decision prompt, we define the native semantic margin as

$$s_{\text{native}}(q, C) = z(B = \text{Insufficient}) - z(A = \text{Sufficient}), \tag{19}$$

where lower values indicate greater evidence sufficiency. After calibration described below, this score replaces the probe score throughout the retrieval controller: it is used for the question-only

Table 15: Per-dataset ablation results of SUREST. Values in parentheses denote changes relative to the full model. Changes are computed from unrounded values.

Dataset	Rounds ↓	Passages ↓	Time (s) ↓	F1 ↑
Native readout (calibrated)				
HotPotQA	3.91 (+32.5%)	5.97 (+11.0%)	7.99 (+28.3%)	45.77 (−2.51)
MuSiQue	4.77 (+5.8%)	7.27 (−14.9%)	10.26 (−6.7%)	16.79 (−0.86)
IIRC	4.72 (+22.6%)	6.97 (+2.0%)	9.53 (+14.7%)	26.16 (−1.45)
2Wiki	3.85 (+36.5%)	5.13 (+2.2%)	7.78 (+20.8%)	36.30 (−3.16)
Macro	**4.31** (+22.1%)	**6.34** (−1.7%)	**8.89** (+11.2%)	**31.26** (−1.99)
w/o No-context Gate				
HotPotQA	3.14 (+6.6%)	5.77 (+7.2%)	5.74 (−7.8%)	48.46 (+0.18)
MuSiQue	4.57 (+1.2%)	8.65 (+1.4%)	10.34 (−6.0%)	17.59 (−0.06)
IIRC	3.94 (+2.3%)	7.06 (+3.3%)	7.88 (−5.2%)	27.34 (−0.27)
2Wiki	3.69 (+31.0%)	6.57 (+30.7%)	7.14 (+10.9%)	37.47 (−1.99)
Macro	**3.83** (+8.6%)	**7.01** (+8.8%)	**7.77** (−2.7%)	**32.71** (−0.53)
w/o SMG				
HotPotQA	3.21 (+9.0%)	9.89 (+83.8%)	8.27 (+32.9%)	46.59 (−1.69)
MuSiQue	4.60 (+1.9%)	14.21 (+66.5%)	12.29 (+11.8%)	17.38 (−0.26)
IIRC	4.11 (+6.9%)	12.82 (+87.6%)	10.51 (+26.5%)	25.07 (−2.54)
2Wiki	2.98 (+5.7%)	9.46 (+88.3%)	8.15 (+26.7%)	37.05 (−2.41)
Macro	**3.73** (+5.7%)	**11.60** (+80.1%)	**9.81** (+22.8%)	**31.52** (−1.73)
w/o Uncertainty-Aware Lookahead				
HotPotQA	3.00 (+1.7%)	5.14 (−4.4%)	5.71 (−8.3%)	47.87 (−0.42)
MuSiQue	4.52 (+0.2%)	7.76 (−9.1%)	9.58 (−12.9%)	15.80 (−1.84)
IIRC	3.93 (+2.2%)	6.43 (−5.9%)	7.47 (−10.1%)	26.67 (−0.93)
2Wiki	2.88 (+2.3%)	4.74 (−5.7%)	5.90 (−8.4%)	39.06 (−0.40)
Macro	**3.58** (+1.5%)	**6.02** (−6.6%)	**7.16** (−10.4%)	**32.35** (−0.90)
w/o Adaptive Termination				
HotPotQA	4.49 (+52.3%)	6.42 (+19.2%)	9.39 (+50.9%)	48.47 (+0.19)
MuSiQue	4.80 (+6.2%)	8.71 (+2.0%)	12.47 (+13.4%)	17.82 (+0.18)
IIRC	4.71 (+22.6%)	7.43 (+8.8%)	10.73 (+29.2%)	27.34 (−0.27)
2Wiki	3.62 (+28.5%)	5.57 (+10.8%)	8.27 (+28.4%)	39.47 (+0.01)
Macro	**4.41** (+24.9%)	**7.03** (+9.2%)	**10.22** (+27.9%)	**33.27** (+0.02)

gate, post-retrieval sufficiency termination, and candidate scoring through

$$\Delta_{\text{cal}}(d \mid q, E) = s_{\text{cal}}(q, E) - s_{\text{cal}}(q, E \cup \{d\}), \tag{20}$$

Offline calibration. The native margin and probe score have different numerical scales, so we calibrate the native readout before running the end-to-end ablation. For retrieved-evidence and no-context settings separately, we fit a constrained affine mapping from the native margin to the original probe score,

$$s_{\text{cal}} = a\, s_{\text{native}} + b, \qquad a > 0. \tag{21}$$

The positivity constraint preserves the score ordering, so larger calibrated values continue to indicate greater insufficiency. Calibration uses the frozen HotPotQA and MuSiQue probe-pair development examples, excluding any question IDs sampled into the four end-to-end evaluation sets. We separately map native margins to the probe-score scales used for retrieved evidence and question-only reasoning, allowing the ablation to reuse the same operational thresholds τ_E and τ_0 as the main system.

The calibration stage therefore still requires the original linear probe, but the resulting end-to-end runs do not load or evaluate any probe checkpoint. This experiment should thus be interpreted as replacing the probe at inference time, rather than as a probe-free or training-free alternative.

Selection without ensemble uncertainty. The main system estimates candidate uncertainty from the five-seed probe ensemble. Because the native readout provides only a single semantic margin,

Table 16: Behavior of the no-context gate and post-retrieval sufficiency termination. Post-retrieval stopping rates are computed over questions that enter retrieval. "Missing gold" denotes stopped cases lacking at least one annotated supporting page.

Dataset	No-context accepted	Accepted EM	Post-retrieval stop	Missing gold at stop	Stopped EM
HotPotQA	37/500 (7.4%)	19/37 (51.4%)	303/463 (65.4%)	143/303 (47.2%)	132/303 (43.6%)
MuSiQue	6/500 (1.2%)	0/6 (0.0%)	95/494 (19.2%)	69/95 (72.6%)	27/95 (28.4%)
IIRC	13/500 (2.6%)	6/13 (46.2%)	212/487 (43.5%)	129/212 (60.8%)	59/212 (27.8%)
2Wiki	107/500 (21.4%)	55/107 (51.4%)	189/393 (48.1%)	73/189 (38.6%)	53/189 (28.0%)
Overall	**163/2000 (8.15%)**	**80/163 (49.1%)**	**799/1837 (43.5%)**	**414/799 (51.8%)**	**271/799 (33.9%)**

this ensemble-based uncertainty test and the associated uncertainty-aware lookahead are disabled. All remaining settings, including retrieval budgets, query generation, candidate retrieval, stopping thresholds, and the final reader, are kept unchanged.

Across the four benchmarks, replacing the probe with the calibrated native readout reduces macro EM/F1 from 26.20/33.25 to 24.25/31.26. Retrieval rounds increase from 3.53 to 4.31 and inference time from 7.99 s to 8.89 s, while retained passages remain similar (6.44 vs. 6.34). Together with the smaller degradation observed when disabling uncertainty-aware lookahead alone, these results indicate that the native readout provides a less effective control signal for end-to-end adaptive retrieval.

C.3.2 Gating and termination mainly control retrieval cost

Table 16 further examines the decisions made by the two stopping mechanisms. Across the 2,000 evaluation questions, 325 question-only outputs proceed to the no-context probe, while the remaining questions are rejected by the model and routed directly to retrieval. Among the 325 probed cases, 163 (50.2%) are accepted without retrieval.

The no-context gate does not simply trade answer accuracy for efficiency. Among the 163 questions accepted by the full system, 80 are EM-correct and 83 are EM-incorrect. When the gate is disabled and these same questions instead proceed through retrieval, 26 change from incorrect to correct, but 37 change from correct to incorrect. Thus, the gate yields a net gain of 11 EM-correct answers on this subset; the 83 incorrect question-only answers should therefore not be interpreted as 83 errors caused by early acceptance.

A similar pattern holds for post-retrieval termination. Among the 1,837 questions entering retrieval, SuREST declares the accumulated evidence sufficient and stops on 799 (43.5%). Disabling this early stopping changes 36 of these cases from incorrect to correct, but changes 35 from correct to incorrect. Hence, sufficiency-based termination costs only one net EM-correct answer over all 2,000 questions (0.05 percentage points), while saving an average of 2.19 additional retrieval calls each time it triggers.

Missing annotated support should not be equated with insufficient evidence. Annotated supporting pages are not necessarily the only valid evidence path, and manual spot checks further suggest that a non-trivial fraction of annotated supporting evidence is absent from the fixed retrieval corpus. Conversely, even among the 385 stopped cases that cover all annotated supporting pages, 223 remain EM-incorrect. Thus, incomplete annotated-page coverage explains only part of the residual error; answer generation and reasoning errors remain substantial even when the annotated evidence has been fully recovered.

C.4 Evidence-Quality Evaluation Protocol

We compare the final evidence sets produced by SuREST, Adaptive-RAG, IRCoT, Probing-RAG, and S2G-RAG. To isolate the quality of acquired evidence from the separate decision of whether to retrieve at all, we evaluate on the *common-retrieval subset*: questions for which all five methods retain at least one evidence page and gold supporting pages are available. This yields 198 HotPotQA, 240 MuSiQue, 279 IIRC, and 188 2Wiki examples.

Table 17: Cross-backbone results with Mistral-Small-24B-Instruct-2501. Best and second-best QA results are shown in bold and underline, respectively.

Method	Macro EM	Macro F1	Rounds	Passages	Time (s)
w/o retrieval	18.45	24.17	0.00	0.00	0.59
Single-step retrieval	25.25	32.66	1.00	6.00	1.50
FLARE	18.40	22.94	1.90	5.59	1.41
DRAGIN	27.40	36.50	2.01	6.04	17.40
ETC	27.10	35.81	2.38	7.13	13.69
Adaptive-RAG	27.65	35.18	3.21	10.37	18.70
SeaKR	<u>30.75</u>	<u>39.18</u>	2.68	2.68	177.68
Probing-RAG	22.30	29.50	2.38	7.15	28.55
S2G-RAG	27.65	35.05	4.61	13.83	18.89
IRCoT	29.35	37.63	5.00	14.98	29.12
SUREST	**32.70**	**40.90**	3.53	6.45	17.13

Table 18: QA performance and retrieval cost on Qwen3.8-27B for the evaluated method subset. EM and F1 are reported on each benchmark, while Macro denotes the unweighted average across the four datasets. R and P denote the average numbers of retrieval rounds and retained passages, respectively, and T denotes inference time in seconds per question. Best and second-best QA results among evaluated methods are shown in bold and underline, respectively.

	HotPotQA		2Wiki		IIRC		MuSiQue		Macro				
Method	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	R ↓	P ↓	T(s) ↓
w/o retrieval	23.20	31.29	23.60	28.33	12.20	15.28	7.20	11.85	16.55	21.69	0.00	0.00	6.3
Single-step	41.00	52.91	34.20	40.15	30.20	34.90	12.20	20.35	29.40	37.08	1.00	6.00	2.9
FLARE	32.80	43.48	29.60	33.15	20.20	22.91	8.00	14.10	22.65	28.41	2.35	7.04	10.5
DRAGIN	41.40	53.53	<u>42.80</u>	<u>49.43</u>	31.00	34.68	<u>18.40</u>	<u>27.55</u>	<u>33.40</u>	<u>41.30</u>	2.45	7.35	28.1
ETC	35.20	47.92	41.20	47.84	28.60	32.77	14.40	23.69	29.85	38.06	2.14	6.41	25.2
S2G-RAG	43.20	55.41	40.00	46.90	32.80	37.31	15.80	23.10	32.95	40.68	4.58	13.74	22.3
IRCoT	<u>44.60</u>	<u>57.51</u>	36.20	43.17	<u>33.00</u>	<u>37.54</u>	<u>18.40</u>	26.01	33.05	41.06	4.88	14.63	58.4
SUREST	**50.00**	**63.02**	**47.80**	**54.03**	**40.60**	**45.91**	**25.60**	**34.20**	**41.00**	**49.29**	3.28	5.91	20.5

Evidence is evaluated at the Wikipedia-page level. We normalize page titles by Unicode decomposition, ASCII conversion, lowercasing, replacing underscores with spaces, removing non-alphanumeric characters, and collapsing whitespace. For question i, let G_i and P_i denote the sets of normalized gold and retained page titles. We compute

$$\mathrm{Prec}_i = \frac{|G_i \cap P_i|}{|P_i|}, \qquad \mathrm{Rec}_i = \frac{|G_i \cap P_i|}{|G_i|}, \tag{22}$$

$$\mathrm{F1}_i = \begin{cases} \dfrac{2\,\mathrm{Prec}_i \mathrm{Rec}_i}{\mathrm{Prec}_i + \mathrm{Rec}_i}, & \mathrm{Prec}_i + \mathrm{Rec}_i > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

We additionally report complete-evidence recovery $\mathbb{I}[G_i \subseteq P_i]$ and the number of unique retained pages $|P_i|$. Dataset-level metrics are macro-averaged over questions, and the reported macro results are unweighted averages over the four datasets.

C.5 Additional Cross-Backbone Results

Using the same HotPotQA–MuSiQue development-set selection criterion, the selected probe layers are L26 for Mistral-Small-24B-Instruct-2501 and L50 for Qwen3.8-27B, with corresponding hidden-state dimensions of 5,120 and 5,120, respectively. For both transferred backbones, SUREST directly reuses the operational thresholds selected with Qwen2.5-7B-Instruct, without backbone-specific recalibration. Table 17 reports the full comparison on Mistral-Small-24B-Instruct-2501. To

Table 19: Threshold sensitivity of Probing-RAG across four multi-hop QA benchmarks.

Threshold	Macro EM ↑	Macro F1 ↑	Rounds ↓	Passages ↓
-2	15.95	21.79	2.13	6.38
-1	16.20	22.04	2.30	6.89
0	16.35	22.18	2.47	7.39
1	16.65	22.52	2.62	7.86
2	16.85	22.76	2.77	8.30

Table 20: Generation settings for the prompts used by SUREST.

Stage	Max. tokens	Sampling	Expected output
Question-Only Retrieval Gate	256	No	Reasoning and a short answer
Final reader	256	No	One `Final answer:` line
Targeted query generation	96	No	JSON with an evidence need and query
Query rewrite	64	No	JSON with a revised query

limit the computational cost of evaluation with the larger Qwen3.8-27B backbone, we evaluate a representative subset spanning non-retrieval, fixed retrieval, generation-triggered, explicit-sufficiency, and iterative retrieval methods. Table 18 reports the corresponding per-dataset results. Although w/o retrieval avoids retrieval entirely, it is slower than single-step retrieval because, without evidence, answer generation often continues to the 256-token limit, making the additional decoding cost larger than that of a single BM25 retrieval.

C.6 PROBING-RAG THRESHOLD SENSITIVITY

To ensure a fair comparison, we additionally reproduce the threshold-sensitivity analysis of Probing-RAG over thresholds $\{-2, -1, 0, 1, 2\}$ while keeping all other settings fixed. Table 19 reports macro QA performance and retrieval cost across the four benchmarks. Increasing the threshold leads to progressively more retrieval and slightly higher QA performance, but the overall variation is small: macro F1 changes from 21.79 to 22.76 while retained passages increase from 6.38 to 8.30. The main experiments use threshold 0, yielding a macro F1 of 22.18. The consistently lower performance across the sweep indicates that the gap to SUREST is not attributable to a particular Probing-RAG threshold choice.

D PROMPT TEMPLATES

This appendix reports the prompts used by SUREST. Uppercase angle-bracketed fields such as `<QUESTION>` and `<TITLE 1>` denote run-time substitutions. Angle-bracketed spans inside output-format instructions, such as `<answer>`, are part of the literal prompt text unless stated otherwise. Retrieved evidence is rendered as a numbered sequence of Wikipedia page titles and passage texts. System and user messages are reported separately to preserve their chat roles. Unless otherwise stated, the prompts below correspond to the main Qwen2.5-7B-Instruct experiments; the same templates are reused for the cross-backbone experiments. All deterministic generations use greedy decoding without sampling. Table 20 summarizes the generation limits.

D.1 QUESTION-ONLY RETRIEVAL GATE

Before the first retrieval call, the model produces a reasoning trace and a candidate answer using the following prompt. The final-answer span is removed before the temporary reasoning context is scored by the sufficiency probe. If the validity check or probe gate rejects the candidate, both the reasoning and answer are discarded.

```
No retrieved evidence is available. Reason explicitly before giving
    the final answer. State any uncertainty and do not invent facts.
```

```
    End with 'Final answer: <answer>'. Use only the shortest answer
    span; for yes/no questions, output exactly 'yes' or 'no'.
```

Listing 1: System message for the question-only retrieval gate.

```
Question:
<QUESTION>

Reason explicitly before giving the final answer. If a required fact
    is uncertain, say so instead of guessing.

Reasoning: <reasoning>
Final answer: <short answer, or Unknown>
```

Listing 2: User message for the question-only retrieval gate.

D.2 Final Reader

For a retained evidence set, the same final-reader template is used across datasets, with a fixed dataset-specific demonstration bank. The user message is preceded by a fixed dataset-specific demonstration bank: eight examples for HotPotQA, eight for IIRC, eight for 2WikiMultiHopQA, and six for MuSiQue. The MuSiQue demonstrations include retrieved evidence; the other three banks contain the question, reasoning demonstration, and final answer. The complete fixed demonstration instances are provided in the anonymous supplementary implementation; Listing 4 shows exactly how they are serialized and how the target instance is appended.

```
Answer the question. Reason carefully. You may use the retrieved
    evidence to answer. Return exactly one line: 'Final answer:
    <answer>'. After 'Final answer:', output only the shortest answer
    span or spans, not a sentence. Do not restate the question or add
    explanation. For yes/no questions, output exactly 'yes' or 'no'.
    For numeric questions, include a unit only when the question
    explicitly requests it. If the question requests multiple items,
    include every requested item. Keep only words needed to preserve
    the answer's exact identity.
```

Listing 3: System message for the evidence-conditioned final reader.

```
Official multi-hop QA demonstrations follow. Use their reasoning
    pattern internally and copy their minimal Final answer format.

Example 1
Question: <DEMONSTRATION QUESTION 1>
<OPTIONAL DEMONSTRATION EVIDENCE 1>
Reasoning demonstration: <DEMONSTRATION REASONING 1>
Final answer: <DEMONSTRATION ANSWER 1>

...

Example <N>
Question: <DEMONSTRATION QUESTION N>
<OPTIONAL DEMONSTRATION EVIDENCE N>
Reasoning demonstration: <DEMONSTRATION REASONING N>
Final answer: <DEMONSTRATION ANSWER N>

Now answer the target question. You may use the retrieved evidence.

Question:
<QUESTION>

Retrieved evidence:
[1] <TITLE 1>: <PASSAGE 1>
```

```
[2] <TITLE 2>: <PASSAGE 2>

...

Output requirement:
Return exactly one line: 'Final answer: <answer>'. After 'Final
    answer:', output only the shortest answer span or spans, not a
    sentence. Do not restate the question or add explanation. For
    yes/no questions, output exactly 'yes' or 'no'. For numeric
    questions, include a unit only when the question explicitly
    requests it. If the question requests multiple items, include every
    requested item. Keep only words needed to preserve the answer's
    exact identity.
```

Listing 4: User-message template for the evidence-conditioned final reader. The bracketed evidence block inside each demonstration is present for MuSiQue demonstrations only.

For MuSiQue, each optional demonstration-evidence field in Listing 4 is serialized as follows; for the other datasets, this field is omitted rather than replaced by an empty marker.

```
Retrieved evidence:
[1] <DEMONSTRATION TITLE 1>: <DEMONSTRATION PASSAGE 1>
[2] <DEMONSTRATION TITLE 2>: <DEMONSTRATION PASSAGE 2>
...
```

D.3 Targeted Query Generation

After the initial original-question BM25 retrieval, an insufficient evidence set triggers targeted query generation. Query generation and rewriting share the control-system message in Listing 5.

```
Follow the user instruction exactly and keep the response concise.
```

Listing 5: System message shared by targeted query generation and query rewriting.

```
Identify the single most important missing fact or reasoning
    connection needed to answer the question reliably from the current
    evidence. Produce a concise lexical search query that can find that
    fact. Return one JSON object with keys "missing_fact" and "query"
    only.

Question:
<QUESTION>

Current evidence:
[1] <TITLE 1>: <PASSAGE 1>

[2] <TITLE 2>: <PASSAGE 2>

...

Queries already executed (do not repeat or trivially paraphrase them):
- <QUERY 1>
- <QUERY 2>
...
```

Listing 6: User-message template for targeted query generation. The query-history block is omitted when no earlier query exists.

D.4 QUERY REWRITE

Query rewriting is used in two failure cases. First, if a newly generated targeted query is empty or duplicates a previously executed query, SUREST attempts one rewrite before terminating. Second, after a non-empty BM25 retrieval, if the sufficiency-guided selector retains no admissible candidate, a rewritten query is generated for the next retrieval round, provided that retrieval budget remains. Each failure condition permits at most one rewrite attempt.

```
<FAILURE MESSAGE>

Question:
<QUESTION>

Missing fact:
<MISSING FACT>

Current evidence:
[1] <TITLE 1>: <PASSAGE 1>

[2] <TITLE 2>: <PASSAGE 2>

...

Failed query:
<FAILED QUERY>

Previous queries:
- <QUERY 1>
- <QUERY 2>
...

First judge whether the stated missing fact is a valid unresolved
    reasoning hop given the question and current evidence. If it is
    not, silently identify the correct unresolved hop before writing
    the query. Do not output this judgment. Target one unresolved
    reasoning hop. Use grounded names or titles from the question and
    evidence, but change the search anchor or relation when needed.
    Never replace an unresolved entity with a guessed name. Keep the
    known anchor and the relation that identifies it. Write a 3-10 term
    lexical BM25 query, not a full question. Do not use Boolean
    operators or assume the answer.

Return only: {"query": "<new query>"}
```

Listing 7: User-message template for query rewriting.

The failure message in Listing 7 is selected from the following two verbatim variants.

```
The previous query is empty or repeats an executed search. Target a
    different unresolved reasoning hop, not a paraphrase.
```

Listing 8: Failure message used when the proposed query is empty or duplicates an executed query.

```
The previous BM25 query retrieved no useful new evidence. Create a
    different search query, not a paraphrase of the failed query.
```

Listing 9: Failure message used when the sufficiency-guided selector retains no admissible candidate.

If the missing-fact field parsed from the preceding targeted-query-generation output is empty, the literal fallback text `the most important unresolved fact` is inserted. Query histories are ordered chronologically. The rewritten query is normalized for duplicate detection before it is issued to BM25.

E CASE STUDIES

We present two illustrative examples from each benchmark to qualitatively examine the behavior of SUREST in terms of answer quality and retrieval efficiency. For SUREST, we additionally show the executed control trace, including retrieval triggering, targeted queries, selected evidence, sufficiency-margin changes, and the stopping decision. Selection-time scores obtained through KV-cache-based candidate scoring may differ numerically from full-input reevaluation scores; consequently, the reported margin changes are relative to their respective selection-time baselines and need not sum to the difference between successive stopping-check scores. Rounds denote retrieval calls, and Passages denotes the size of the final accumulated evidence set.

Table 21: Output examples on HotPotQA.

Case 1

Question: In what movie did a Kenyan-Mexican actress, who graduated from Hampshire College, star in in 2015?

Gold answer: Star Wars: The Force Awakens

SuREST	**Answer:** Star Wars: The Force Awakens **EM:** 1 **Rounds:** 2 **Passages:** 4 *Trace.* The question-only output *Unknown* is rejected as a refusal. Call 1 retrieves three passages with the original question; $s_{\text{check}} = -1.23 > -2$, so retrieval continues. Call 2 uses *Lupita Nyong'o 2015 movie* to target the missing film role and retains one additional passage. Its selection-time SMG is 1.67; the independently reevaluated score is $s_{\text{check}} = -2.91 \leq -2$. Retrieval stops and the final reader answers from four retained passages $(3 + 1)$ after two calls.
Adaptive-RAG	**Answer:** Eclipsed **EM:** 0 **Rounds:** 5 **Passages:** 15
ETC	**Answer:** "Selma." **EM:** 0 **Rounds:** 4 **Passages:** 12
IRCoT	**Answer:** Eclipsed **EM:** 0 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** What Katy Did **EM:** 0 **Rounds:** 2 **Passages:** 6
S2G-RAG	**Answer:** Starred in "Selma" **EM:** 0 **Rounds:** 5 **Passages:** 15

Case 2

Question: Which rock band chose its name by drawing it out of a hat, Switchfoot or Midnight Oil?

Gold answer: Midnight Oil

SuREST	**Answer:** Midnight Oil **EM:** 1 **Rounds:** 2 **Passages:** 5 *Trace.* The question-only answer is *Midnight Oil*, but probing its preceding reasoning yields $-4.84 > \tau_0 = -5$, so the gate triggers retrieval. Call 1 retrieves three passages and yields $s_{\text{check}} = -0.73 > -2$. Call 2 uses *Switchfoot name origin drawing hat* and retains two passages, with successive selection-time SMGs of 0.85 and 1.15. After reevaluation, $s_{\text{check}} = -2.52 \leq -2$, so retrieval stops. The final reader returns *Midnight Oil* using five passages $(3 + 2)$ after two calls.
Adaptive-RAG	**Answer:** Switchfoot **EM:** 0 **Rounds:** 0 **Passages:** 0
ETC	**Answer:** Switchfoot **EM:** 0 **Rounds:** 4 **Passages:** 12
IRCoT	**Answer:** Midnight Oil **EM:** 1 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** Switchfoot **EM:** 0 **Rounds:** 0 **Passages:** 0
S2G-RAG	**Answer:** Midnight Oil **EM:** 1 **Rounds:** 5 **Passages:** 15

Table 22: Output examples on MuSiQue.

Case 1

Question: What date did the creator of The Blind Girl die?

Gold answer: 13 August 1896

SuREST	**Answer:** 13 August 1896 **EM:** 1 **Rounds:** 3 **Passages:** 9 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retrieves three passages; $s_{\text{check}} = 6.39 > -4$. Call 2 uses *The Blind Girl creator date of death* and retains three more passages, bringing the total to six. The reevaluated score is $3.71 > -4$, so the next query targets the identified creator: *John Everett Millais date of death*. Call 3 retains three additional passages; reevaluation gives $s_{\text{check}} = -4.37 \leq -4$, triggering termination. The reader answers *13 August 1896* using nine passages ($3 + 3 + 3$). These are round-level checks, not cumulative sums of candidate SMGs.
Adaptive-RAG	**Answer:** 1887 **EM:** 0 **Rounds:** 4 **Passages:** 12
ETC	**Answer:** 13 August 1893 **EM:** 0 **Rounds:** 3 **Passages:** 9
IRCoT	**Answer:** 1887 **EM:** 0 **Rounds:** 4 **Passages:** 12
Probing-RAG	**Answer:** no **EM:** 0 **Rounds:** 5 **Passages:** 15
S2G-RAG	**Answer:** 1903 **EM:** 0 **Rounds:** 5 **Passages:** 15

Case 2

Question: What county is the city that shares a border with the state capital of the state where Andrew Deveaux was born located in?

Gold answer: Richland County; Richland County, South Carolina

SuREST	**Answer:** Richland County **EM:** 1 **Rounds:** 4 **Passages:** 7 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retains three passages, with $s_{\text{check}} = 2.10 > -4$. Call 2 asks for the county containing the city bordering South Carolina's state capital; it adds one passage, and reevaluation gives $0.51 > -4$. Call 3 targets the city bordering Columbia and adds three passages; $s_{\text{check}} = -1.64 > -4$, so sufficiency is still not reached. Call 4 retains no candidate, and the query-rewrite fallback fails. This fallback terminates retrieval; the reader answers from seven passages ($3 + 1 + 3 + 0$). The fourth call counts toward Rounds even though it adds no evidence.
Adaptive-RAG	**Answer:** Beaufort County **EM:** 0 **Rounds:** 5 **Passages:** 15
ETC	**Answer:** Richland County **EM:** 1 **Rounds:** 5 **Passages:** 15
IRCoT	**Answer:** Beaufort County **EM:** 0 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** Beaufort **EM:** 0 **Rounds:** 5 **Passages:** 15
S2G-RAG	**Answer:** Beaufort **EM:** 0 **Rounds:** 5 **Passages:** 15

Table 23: Output examples on IIRC.

Case 1

Question: When was the team founded that beat Rhyl FC 5-0 in 1993?

Gold answer: 1899

SuREST	**Answer:** 1899 **EM:** 1 **Rounds:** 3 **Passages:** 7 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retains three passages, with $s_{\text{check}} = 3.72 > -3$. Call 2 queries the team that beat Rhyl FC in 1993, adds two passages, and yields $s_{\text{check}} = -2.58 > -3$. Call 3 uses *cardiff city 1993 founded*. Its first candidate has SMG -0.047 but is admitted through uncertain lookahead: the logged two-passage plan has gain 0.544 and predicted score $-3.123 \leq -3$. After committing the first passage, the second is rescored and selected with SMG 0.606. Reevaluation gives $s_{\text{check}} = -3.15 \leq -3$. Retrieval stops with seven passages $(3 + 2 + 2)$.
Adaptive-RAG	**Answer:** 1879 **EM:** 0 **Rounds:** 5 **Passages:** 15
ETC	**Answer:** 1878 **EM:** 0 **Rounds:** 5 **Passages:** 15
IRCoT	**Answer:** 1879 **EM:** 0 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** 1947 **EM:** 0 **Rounds:** 5 **Passages:** 15
S2G-RAG	**Answer:** 1993 **EM:** 0 **Rounds:** 5 **Passages:** 15

Case 2

Question: What year did Ebert start critiquing films?

Gold answer: 1967

SuREST	**Answer:** 1967 **EM:** 1 **Rounds:** 2 **Passages:** 5 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retrieves three passages and yields $s_{\text{check}} = 4.16 > -3$. Call 2 uses *Roger Ebert film criticism start year* and retains two passages, with successive selection-time SMGs of 6.28 and 1.07. The independently reevaluated score is $s_{\text{check}} = -3.28 \leq -3$, so retrieval terminates. The final reader answers *1967* using five passages $(3 + 2)$ after two retrieval calls.
Adaptive-RAG	**Answer:** 1968 **EM:** 0 **Rounds:** 1 **Passages:** 15
ETC	**Answer:** 1965 **EM:** 0 **Rounds:** 4 **Passages:** 12
IRCoT	**Answer:** 1967 **EM:** 1 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** 1955 **EM:** 0 **Rounds:** 0 **Passages:** 0
S2G-RAG	**Answer:** 1967 **EM:** 1 **Rounds:** 5 **Passages:** 15

Table 24: Output examples on 2WikiMultiHopQA.

Case 1

Question: When did Albrecht Of Saxe-Weissenfels's mother die?

Gold answer: 11 December 1669

SuREST	**Answer:** 11 December 1669 **EM:** 1 **Rounds:** 2 **Passages:** 4 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retrieves three passages; $s_{\text{check}} = 0.51 > -3$. Call 2 targets *Anna Maria of Mecklenburg-Schwerin death date*, identifying Albrecht's mother as the missing link, and retains one passage. At selection time, the logged baseline 0.509 minus SMG 4.547 gives the candidate score -4.038 (rounded). A separate evidence reevaluation gives $s_{\text{check}} = -3.788 \leq -3$. Retrieval therefore stops with four passages $(3 + 1)$ after two calls.
Adaptive-RAG	**Answer:** 22 January 1686 **EM:** 0 **Rounds:** 5 **Passages:** 15
ETC	**Answer:** - the date is is not provided **EM:** 0 **Rounds:** 3 **Passages:** 9
IRCoT	**Answer:** 22 January 1686 **EM:** 0 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** 9 May 1692 **EM:** 0 **Rounds:** 5 **Passages:** 15
S2G-RAG	**Answer:** 22 January 1686 **EM:** 0 **Rounds:** 5 **Passages:** 15

Case 2

Question: Who was born earlier, Matt Guthmiller or George Nevill, 1St Earl Of Abergavenny?

Gold answer: George Nevill, 1St Earl Of Abergavenny

SuREST	**Answer:** George Nevill, 1st Earl Of Abergavenny **EM:** 1 **Rounds:** 3 **Passages:** 7 *Trace.* The question-only output *Unknown* triggers retrieval. Call 1 retains three passages, with $s_{\text{check}} = 5.67 > -3$. Call 2 uses *Matt Guthmiller birth year* and retains three more passages. Although the last selection-time candidate score is -3.31, the outer-loop reevaluation gives $s_{\text{check}} = -2.93 > -3$; retrieval therefore continues. Call 3 uses the query *Matt Guthmiller birth date* and adds one passage. The new check gives $s_{\text{check}} = -3.58 \leq -3$, terminating retrieval. The reader answers from seven passages $(3 + 3 + 1)$ after three calls.
Adaptive-RAG	**Answer:** Matt Guthmiller **EM:** 0 **Rounds:** 0 **Passages:** 0
ETC	**Answer:** 24 June 1727 **EM:** 0 **Rounds:** 3 **Passages:** 9
IRCoT	**Answer:** George Nevill, 1st Earl Of Abergavenny **EM:** 1 **Rounds:** 5 **Passages:** 15
Probing-RAG	**Answer:** Matt Guthmiller **EM:** 0 **Rounds:** 1 **Passages:** 3
S2G-RAG	**Answer:** George Nevill, 1st Earl Of Abergavenny **EM:** 1 **Rounds:** 5 **Passages:** 15

Algorithm 1 SUREST: Sufficiency-guided retrieval, evidence selection, and termination

Require: Question q, language model M, retriever $\mathcal{R}$, insufficiency score s, thresholds τ_0, τ_E
Require: Retrieval budget B_R, passage budget B_P, initial depth k_0, candidate depth k, per-round selection limit m
Require: Rewrite-eligible selector failures $\mathcal{F}_{\mathrm{rw}}$

$(r_0, a_0, v) \leftarrow \textsc{PriorAnswer}(M, q)$ ▷ **(1) Whether to retrieve**
if v **and** $s(q, \{r_0\}) \le \tau_0$ **then**
  return a_0
end if
$E \leftarrow \textsc{TopK}(\mathcal{R}(q), k_0)$
$b \leftarrow 1$; $(g_{\mathrm{fb}}, x_{\mathrm{fb}}) \leftarrow (\varnothing, \varnothing)$; $\mathcal{H} \leftarrow \{q\}$
while $s(q, E) > \tau_E$ **and** $b < B_R$ **and** $|E| < B_P$ **do**
  if $x_{\mathrm{fb}} = \varnothing$ **then**
    $(g, x) \leftarrow \textsc{DescribeNeed}(M, q, E, \mathcal{H})$
  else
    $(g, x) \leftarrow (g_{\mathrm{fb}}, x_{\mathrm{fb}})$
    $(g_{\mathrm{fb}}, x_{\mathrm{fb}}) \leftarrow (\varnothing, \varnothing)$
  end if
  ▷ One-shot rewrite for empty or repeated queries
  if $x = \varnothing$ **or** $x \in \mathcal{H}$ **then**
    $x' \leftarrow \textsc{Rewrite}(M, q, E, g, x, \mathcal{H})$
    if $x' = \varnothing$ **or** $x' \in \mathcal{H}$ **then**
      break
    end if
    $x \leftarrow x'$
  end if
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{x\}$
  $C \leftarrow \textsc{TopK}(\mathcal{R}(x) \setminus E, k)$
  $b \leftarrow b + 1$
  if $C = \emptyset$ **then**
    break ▷ No unseen candidate returned
  end if
  $j \leftarrow 0$; $\rho \leftarrow \varnothing$
  while $C \neq \emptyset$ **and** $j < m$ **and** $|E| < B_P$ **do**
    for all $c \in C$ **do**
      $\Delta_{\mathrm{SMG}}(c \mid q, E) \leftarrow s(q, E) - s(q, E \cup \{c\})$
    end for
    $(c^\star, \rho) \leftarrow \textsc{SelectEvidence}(C, \Delta_{\mathrm{SMG}}, \mathrm{rank}_{\mathrm{BM25}})$ ▷ **(2) Which evidence to acquire**
    if $c^\star = \varnothing$ **then**
      break
    end if
    $E \leftarrow E \cup \{c^\star\}$; $C \leftarrow C \setminus \{c^\star\}$; $j \leftarrow j + 1$
    if $s(q, E) \le \tau_E$ **then**
      break ▷ **(3) When to stop**
    end if
  end while
  if $s(q, E) \le \tau_E$ **then**
    break
  else if $j = 0$ **and** $\rho \in \mathcal{F}_{\mathrm{rw}}$ **and** $b < B_R$ **then**
    $x' \leftarrow \textsc{Rewrite}(M, q, E, g, x, \mathcal{H})$
    if $x' = \varnothing$ **or** $x' \in \mathcal{H}$ **then**
      break
    end if
    $(g_{\mathrm{fb}}, x_{\mathrm{fb}}) \leftarrow (g, x')$
  end if
end while
return $\textsc{Reader}(M, q, E)$