

STAR: Structure-Aware Adaptive Retrieval for RAG

Yeowon Jeon
Sookmyung Women's University
Seoul, Republic of Korea
wyw24@sookmyung.ac.kr

Chong-Kwon Kim
Korea Institute of Energy Technology
Naju, Republic of Korea
ckim@kentech.ac.kr

Yoonhyuk Choi
Sookmyung Women's University
Seoul, Republic of Korea
chldbsgur123@gmail.com

Abstract

Retrieval-augmented generation (RAG) often relies on a fixed retrieval budget even though the amount and structure of evidence required to answer a question vary substantially across queries. As a result, the retrieved evidence can remain redundant, incomplete, or only weakly supportive of the final answer, even when top-ranked passages appear relevant. We present STAR, a structure-aware adaptive retrieval framework for RAG that treats this mismatch as a problem of diagnosing evidence sufficiency. Rather than collapsing the retrieved set into a single global confidence value, STAR represents the current evidence state with four interpretable signals: retrieval relevance, redundancy, coverage, and supportiveness. A transparent controller aggregates these signals into a sufficiency score and makes a one-step **STOP/EXPAND** decision. Our central claim is that adaptive retrieval benefits from a control signal that preserves structurally distinct insufficiency patterns rather than collapsing them into a single scalar confidence estimate. In our target regime, this yields a competitive quality-cost operating point relative to confidence-based adaptive retrieval and approaches the answer quality of fixed larger- k retrieval while using fewer documents.

CCS Concepts

• **Information systems** → **Retrieval models and ranking**; • **Computing methodologies** → *Natural language generation*.

Keywords

Retrieval-augmented generation, adaptive retrieval, evidence sufficiency, retrieval efficiency

ACM Reference Format:

Yeowon Jeon, Chong-Kwon Kim, and Yoonhyuk Choi. 2026. STAR: Structure-Aware Adaptive Retrieval for RAG. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '26)*, Aug. 9-13, 2026, Jeju, South Korea. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

1 Introduction

Retrieval-augmented generation (RAG) improves the factual grounding of large language models by retrieving external evidence before generation [7–9, 12, 15]. This general pipeline has proved effective across knowledge-intensive language tasks, open-domain question

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

KDD '26, Aug. 9-13, 2026, Jeju, South Korea

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1329-3/25/03

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

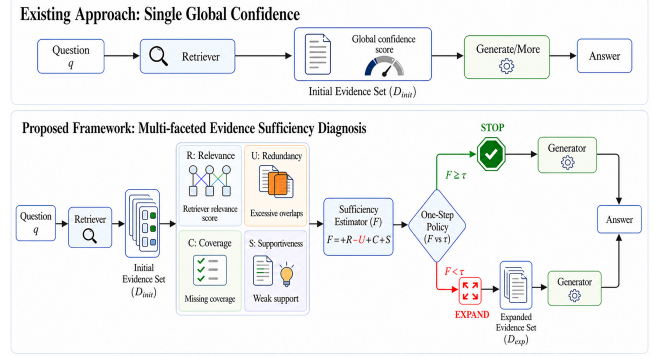


Figure 1: Motivating comparison between single-confidence retrieval control and multi-faceted evidence sufficiency diagnosis. The figure illustrates why single-confidence signals fail to distinguish structurally different failure modes such as missing coverage, redundancy, and weak support.

answering, and retrieval-heavy few-shot settings [4, 14]. Yet retrieving relevant documents does not necessarily mean that the current evidence set is sufficient to answer the question. In many cases, the retrieved documents are individually relevant but collectively incomplete, redundant, or only weakly supportive of the final answer, which can lead to unreliable or only partially grounded responses [2, 23–25]. This problem is especially visible under limited retrieval budgets. Retrieving more documents may improve recall, but it also increases latency, cost, and exposure to distracting context [8, 20]. Stopping too early creates the opposite failure mode: the evidence may look relevant while still lacking the support needed for a grounded answer. The issue is especially acute in open-domain and multi-hop QA, where success often depends on assembling complementary rather than merely top-ranked evidence [4, 22, 23, 25]. Recent adaptive retrieval approaches rely on different decision signals. Some choose retrieval strategies from query complexity before examining retrieved evidence [10], whereas others adapt after retrieval using a global notion of quality, confidence, or self-reflection [2, 20, 24]. These approaches provide limited visibility into why the current evidence is insufficient. In practice, failure often has a structural form: retrieved documents may repeat the same fact, cover only one aspect of the question, or remain weakly supportive despite high retriever scores [12, 23].

We study adaptive retrieval as a control problem over the current evidence set. To address this mismatch, we propose STAR, a structure-aware adaptive retrieval framework for RAG. STAR represents the controller state with decomposed evidence signals that expose insufficiency patterns such as redundancy, missing coverage, and weak support. It uses four interpretable signals to decide

whether the system should answer immediately or perform one additional retrieval step. Rather than redesigning the retriever or generator, STAR adds a diagnostic layer that can sit on top of a standard RAG stack [8, 15, 24].

We restrict the setting to a one-step decision between **STOP** and **EXPAND**. This isolates an under-specified problem in adaptive RAG: deciding when additional retrieval is justified by the structure of the evidence already in hand. Our main claim is that expansion should be conditioned on a decomposed evidence state, not on query complexity alone or on a single global confidence score [2, 22, 24]. We therefore cast expansion control as evidence-state diagnosis and instantiate it with a sufficiency estimator over interpretable signals that captures structurally distinct failure modes. Our contributions are as follows:

- We propose STAR, which formulates adaptive RAG as an evidence-set control problem and explicitly decomposes evidence sufficiency into structurally distinct failure modes rather than treating it as a single confidence estimate.
- We instantiate this formulation with a structure-aware controller that uses four interpretable evidence signals to score sufficiency and make a one-step **STOP/EXPAND** decision.
- We evaluate STAR against fixed-budget and confidence-based baselines using answer-quality and retrieval-efficiency metrics.

2 Related Work

Research on retrieval-augmented generation has established strong retriever-generator baselines for knowledge-intensive natural language processing tasks. Early and influential work showed that combining dense retrieval with generative models can substantially improve factual access and downstream performance [7, 12, 15]. Subsequent systems strengthened the generator side through fusion over multiple passages, large-scale retrieval-augmented pretraining, in-context retrieval augmentation, and retrieval from very large external corpora [3, 8, 9, 18, 19]. This line of work demonstrates the value of retrieval, but it often assumes a fixed retrieval budget or emphasizes retrieval capacity more than the question of whether the current evidence is already sufficient for grounded answer generation.

Adaptive retrieval in RAG has been studied from both query-based and retrieval-aware perspectives. Query-based methods such as Adaptive-RAG select a retrieval strategy from estimated question complexity before inspecting the retrieved evidence [10]. Retrieval-aware methods such as CRAG, Self-RAG, DSP, REPLUG, and Active Retrieval Augmented Generation instead react to the retrieved evidence itself [2, 11, 20, 22, 24]. Our work is closest to the latter direction, but differs in scope. Instead of learning a query policy or introducing another single-score confidence heuristic, we focus on the narrower control problem of representing the retrieved set as a decomposed evidence state and using that representation for a one-step expansion decision.

Our work is also related to research on retrieval quality, reranking, hierarchical retrieval, and open-domain QA benchmarks. Dense retrieval has improved first-stage retrieval quality, but high retrieval relevance does not guarantee low redundancy or strong answer

support [4, 12]. Reader-guided reranking, query reranking, and few-shot reranking methods further highlight the importance of passage interaction and evidence selection after initial retrieval [5, 13, 17]. In HotpotQA and related open-domain QA settings, answering often requires complementary supporting facts across documents, which motivates multi-hop and hierarchical retrieval approaches [1, 6, 16, 21, 23, 25]. Natural Questions provides a complementary open-domain QA setting built from real user queries [14]. These observations motivate our decision to evaluate evidence sets through multiple structural signals rather than collapsing retrieval quality into a single scalar score.

3 Methodology

We study STAR in an adaptive retrieval-augmented generation (RAG) setting in which a retriever returns an initial top- k document set $D = \{d_1, \dots, d_k\}$ for a question q . Our goal is not to redesign the retriever or generator, but to control when retrieval should stop. We define evidence sufficiency as whether the current document set provides enough useful support for grounded answer generation, and we represent that control state with four interpretable signals: retrieval relevance, redundancy, coverage, and supportiveness. The controller therefore reasons over a decomposed evidence state rather than over a single undifferentiated confidence score.

The framework has five stages: *Initial Retrieval*, *Structural Diagnosis*, *Sufficiency Estimation*, *Policy Decision*, and *Optional Follow-up Retrieval*. Unlike approaches that adapt retrieval from query complexity or a single global confidence score, STAR evaluates the retrieved set itself and ties expansion to the insufficiency pattern exposed by the evidence already in hand.

3.1 Problem Setup and Initial Retrieval

Given a question q , a base retriever returns the top- k documents $D = \{d_1, \dots, d_k\}$, where $k = |D|$ denotes the size of the current retrieved evidence set. In experiments, this corresponds to a fixed initial retrieval budget k_{init} . This initial retrieval stage is applied uniformly to all queries and serves as the basis for later diagnosis. Our objective is not to redesign the retriever itself, but to determine whether the current evidence set is already sufficient for grounded answer generation. When expansion is selected, the retriever performs one additional pass to produce a larger evidence set D^+ with budget $k_{\text{exp}} > k_{\text{init}}$. In the current implementation, retrieval is instantiated with dense search over a task-specific corpus slice, and both retrieval budgets are selected on the calibration slice and then fixed for evaluation.

3.2 Structural Diagnosis of Evidence Sufficiency

Instead of compressing the retrieved set into a single confidence value, we decompose its state into four set-level signals: retrieval relevance, redundancy, coverage, and supportiveness. A confidence-based controller asks whether the current retrieval result looks globally reliable; our formulation asks which insufficiency pattern remains visible in the evidence already in hand. The four signals are designed to capture complementary weaknesses of the retrieved set, such as repeated evidence, missing information aspects, or weak answer support, while preserving overall retrieval strength through relevance.

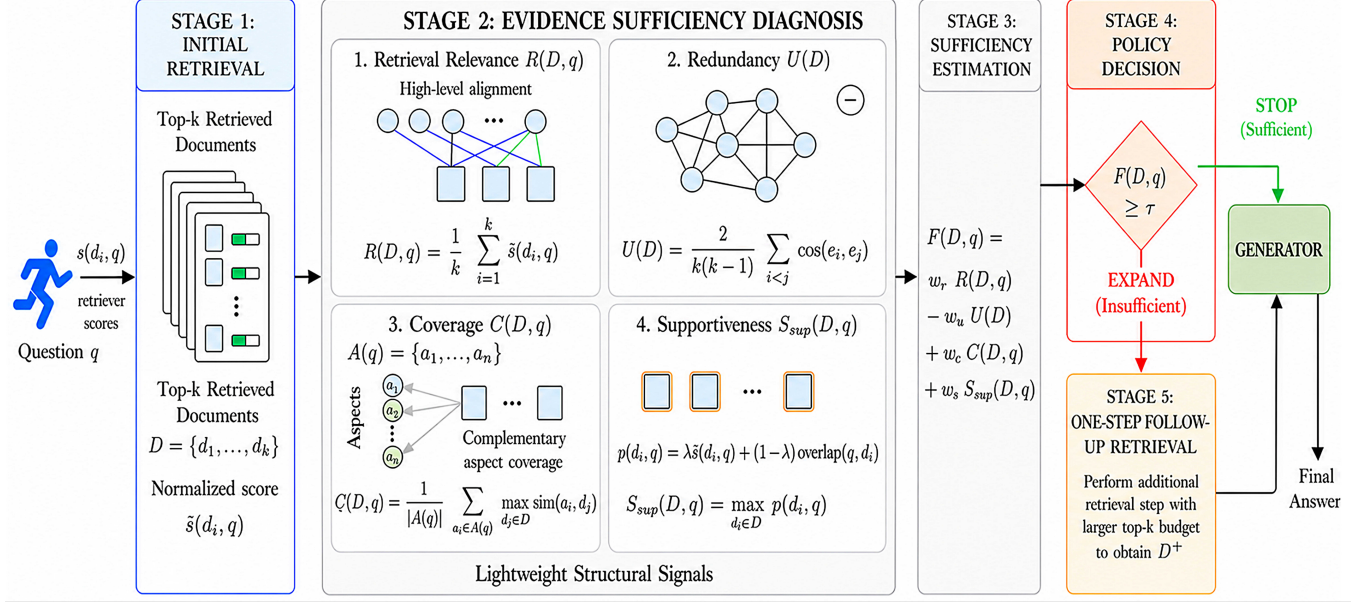


Figure 2: Overview of the proposed structure-aware adaptive RAG framework. Given a question, the system first performs initial retrieval and then diagnoses the sufficiency of the retrieved evidence using four interpretable signals: retrieval relevance, redundancy, coverage, and supportiveness. A weighted sufficiency estimator makes a one-step decision between STOP and EXPAND. If the current evidence is sufficient, the generator answers immediately. Otherwise, the system performs one additional retrieval step with a larger top- k budget before generation.

Let $s(d_i, q)$ denote the retriever score for document d_i and let $\tilde{s}(d_i, q) \in [0, 1]$ be its min-max normalized version within the retrieved set D . Let e_i be the embedding of d_i , and let $A(q) = \{a_1, \dots, a_m\}$ denote a heuristically extracted set of salient question aspects from content-bearing query terms, local token bigrams, and simple entity-like cues in q . In the current implementation, we lowercase and normalize the question, remove stopword-like function words, keep content-bearing unigrams and bigrams, and retain simple entity-like spans detected by capitalization or numeric patterns when available. These aspect candidates are used as heuristic proxies for question sub-needs rather than as fully parsed noun phrases. We then define the following signals.

Retrieval relevance. We measure how strongly the retrieved set aligns with the input question under the base retriever:

$$R(D, q) = \frac{1}{k} \sum_{i=1}^k \tilde{s}(d_i, q). \quad (1)$$

A high relevance score indicates that the current set is strongly matched to the question, but it does not by itself guarantee diversity or sufficiency.

Redundancy. This measures the extent to which the retrieved documents repeat similar evidence. Because it is defined purely over the retrieved set, it depends only on D and not directly on q . We compute it as the average pairwise similarity among document embeddings:

$$U(D) = \frac{2}{k(k-1)} \sum_{i < j} \cos(e_i, e_j). \quad (2)$$

Higher redundancy means that the retrieved set is concentrated on overlapping content, which can limit the marginal utility of additional top-ranked documents even when they are individually relevant.

Coverage. This measures whether the evidence set spans multiple information aspects implied by the question. Because our goal is interpretable control rather than full semantic verification, we instantiate coverage as a heuristic proxy designed for interpretability and straightforward integration. It is intended to detect aspect-level sparsity that often precedes insufficiency in multi-aspect questions. For each aspect $a \in A(q)$, let

$$c(a, D) = \max_{d_i \in D} \text{sim}(a, d_i), \quad (3)$$

where $\text{sim}(a, d_i)$ is an embedding-based aspect-document similarity proxy. In the current implementation, we represent each aspect by averaging its token embeddings and score it against each document using cosine similarity to the document embedding. In addition, we add a safety fallback to lexical matching when aspect-level encoding is unreliable or unavailable. We then define

$$C(D, q) = \frac{1}{|A(q)|} \sum_{a \in A(q)} c(a, D). \quad (4)$$

This term is high when the retrieved set covers multiple complementary parts of the question. Coverage is not intended to measure semantic completeness. Instead, it serves as a heuristic proxy for aspect-level sparsity, especially when global relevance remains high but the evidence is concentrated on only part of the question. We

evaluate its usefulness empirically through ablations and qualitative analysis.

Supportiveness. This signal captures whether at least some retrieved documents are likely to directly support answer generation, rather than merely being topically related. We intentionally avoid introducing a separate verifier or NLI classifier at this stage. Instead, for each document we combine normalized retriever strength with a question-document lexical-overlap proxy, where $\text{overlap}(q, d_i)$ measures the fraction of normalized content-bearing question tokens that appear in d_i . In the current implementation, these two terms are combined with equal weight ($\lambda = 0.5$) as a simple default that avoids introducing another heavily tuned hyperparameter into the controller. This score should be interpreted as a localized answer-bearing cue rather than as evidence that the document semantically entails the answer:

$$p(d_i, q) = \lambda \tilde{s}(d_i, q) + (1 - \lambda) \text{overlap}(q, d_i), \quad (5)$$

We define set-level supportiveness as follows:

$$S_{\text{sup}}(D, q) = \max_{d_i \in D} p(d_i, q). \quad (6)$$

This signal favors evidence sets containing at least one document that is not only relevant under the retriever, but also likely to contain answer-bearing content. Unlike relevance, which captures broad question-document alignment across the retrieved set, supportiveness highlights whether at least one retrieved document exhibits localized answer-bearing cues. As with coverage, we evaluate its usefulness empirically through calibration, ablation, and case analysis, rather than treating lexical overlap itself as a sufficient semantic test.

3.3 Sufficiency Estimation and Retrieval Policy

We combine the four signals with a transparent scoring rule:

$$F(D, q) = w_r R(D, q) - w_u U(D) + w_c C(D, q) + w_s S_{\text{sup}}(D, q), \quad (7)$$

where $w_r, w_u, w_c, w_s \geq 0$ are nonnegative weights estimated on the calibration slice by simple grid search jointly with the threshold τ . Using nonnegative weights preserves the intended monotonic behavior of the controller: increasing relevance, coverage, or supportiveness should not decrease $F(D, q)$, while increasing redundancy should not increase it. Relevance, coverage, and supportiveness therefore contribute positively to evidence sufficiency, whereas redundancy contributes negatively. This explicit decomposition distinguishes STAR from confidence-based adaptive retrieval: a confidence-based controller compresses the retrieved set into a single global value, whereas Eq. 7 preserves distinct evidence failure modes and ties the decision to the specific reason the current set may be inadequate.

The retrieval policy π then applies a one-step threshold rule:

$$\pi(D, q) = \begin{cases} \text{STOP}, & \text{if } F(D, q) \geq \tau, \\ \text{EXPAND}, & \text{otherwise,} \end{cases} \quad (8)$$

where τ is a decision threshold selected on the calibration slice jointly with the weights in Eq. 7. We intentionally restrict the action space to this binary one-step policy in order to isolate the contribution of evidence sufficiency diagnosis without introducing a more complex multi-step controller. This formulation is intentionally simple. It is not meant to serve as a complex predictive model,

but as a transparent control signal that preserves structurally distinct evidence failure modes. While the sufficiency estimator takes the form of a simple linear combination, the key contribution of STAR lies not in the scoring function itself, but in the explicit decomposition of the evidence state into structurally distinct signals that enable targeted retrieval decisions.

3.4 Follow-up Retrieval and Calibration

When the policy selects **EXPAND**, the retriever runs once more with budget k_{exp} , and the generator answers from the expanded evidence set D^+ . When the policy selects **STOP**, the generator answers from the initial evidence set returned with budget k_{init} . The final answer therefore uses the evidence selected by the policy.

We calibrate the policy with silver labels derived from evidence outcomes. For each query in the calibration slice, we compare the initial evidence set D with the expanded set D^+ using a weak-support criterion. An evidence set is treated as weakly supportive only if (i) the gold answer is matched in at least one retrieved document after normalization and (ii) the matched document also satisfies the same question-document overlap condition used in the supportiveness signal. This rule reduces false positives from incidental answer-string matches in loosely related documents. If weak support is already present in D , we label the initial evidence as sufficient; if it appears only in D^+ , we label the initial evidence as insufficient. Examples with no weak support in either set are excluded. This evidence-based labeling does not directly measure downstream generation quality, but it provides a simple and reproducible proxy for calibration. The codebase also includes an optional hybrid variant that additionally requires exact-match correctness from the generator, but the default setup uses the evidence-based variant for reproducibility. The weights in Eq. 7, the budgets k_{init} and k_{exp} , and the threshold in Eq. 8 are selected on the calibration slice before final evaluation.

4 Experiments

We evaluate STAR through three research questions:

- **RQ1:** Does structure-aware diagnosis improve answer quality over simpler baselines? Is it more cost-efficient than always-expanded retrieval?
- **RQ2:** Which evidence signals matter most for STAR’s behavior and performance?
- **RQ3:** How does STAR make retrieval decisions on representative examples in practice?

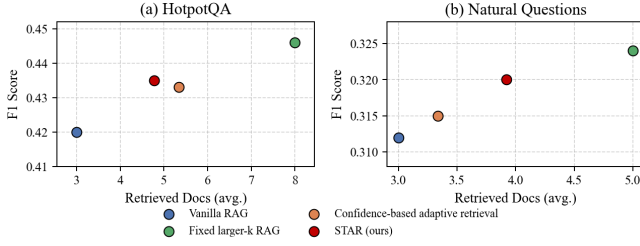
4.1 Datasets and Baselines

We evaluate on HotpotQA and Natural Questions. HotpotQA stresses multi-hop evidence composition, whereas Natural Questions tests whether the same controller remains useful for standard open-domain QA [14, 25]. In the main results, both datasets use fixed 1,000-example validation slices. Under the slice-based corpus setup, both retrieval and evaluation are restricted to the validation split. We compare STAR against three representative baselines:

- **Vanilla RAG** uses only the initial top- k retrieved evidence for answer generation.

Table 1: Results on two benchmark datasets. Higher is better for EM/F1, and lower is better for retrieval calls, retrieved documents, and expansion rate (RQ1).

Method	HotpotQA					Natural Questions				
	EM	F1	Calls	Docs	Exp. Rate	EM	F1	Calls	Docs	Exp. Rate
Vanilla RAG	0.315	0.420	1.000	3.000	0.000	0.223	0.312	1.000	3.000	0.000
Fixed larger- k RAG	0.334	0.446	1.000	8.000	1.000	0.233	0.324	1.000	5.000	1.000
Confidence-based adaptive retrieval	0.322	0.433	1.468	5.340	0.468	0.226	0.315	1.169	3.338	0.169
STAR (ours)	0.328	0.435	1.355	4.775	0.355	0.227	0.320	1.461	3.922	0.461

**Figure 3: Quality-cost trade-off of retrieval policies on HotpotQA and Natural Questions. STAR tends to occupy a competitive operating point relative to confidence-based adaptive retrieval, improving answer quality while avoiding the full document cost of fixed larger- k retrieval.**

- **Fixed larger- k RAG** always expands the retrieval budget regardless of evidence sufficiency.
- **Confidence-based adaptive retrieval** decides whether to retrieve more using a single global confidence signal derived from the initial retrieval result.

4.2 Main Results (RQ1)

Table 1 compares STAR with the three baselines in terms of answer quality and retrieval efficiency. Because all methods share the same generator and prompting setup, differences in EM/F1 primarily reflect the retrieval policy rather than the generation model. Figure 3 complements Table 1 by visualizing the quality-cost trade-off of each retrieval policy. Across both datasets, STAR occupies a competitive quality-cost operating point relative to confidence-based adaptive retrieval and fixed larger- k retrieval. The advantage is clearest on HotpotQA, whereas on Natural Questions the gains are smaller and should be interpreted as modest but directionally consistent under the current controlled setting.

Answer Quality. On HotpotQA, STAR achieves higher EM/F1 than both Vanilla RAG and the confidence-based adaptive baseline, increasing EM from 0.315 to 0.328 and F1 from 0.420 to 0.435 relative to Vanilla RAG. The difference from the stronger confidence baseline is modest in absolute F1 (0.433 to 0.435), but STAR achieves it while using fewer retrieved documents. On Natural Questions, the differences are smaller: STAR achieves higher EM/F1 than the confidence baseline (0.226 to 0.227 EM and 0.315 to 0.320 F1) while remaining below the Fixed larger- k result of 0.324 F1 under a smaller expansion budget. This suggests that the same

Table 2: Ablation results for STAR on HotpotQA. Higher is better for F1 and decision accuracy; lower is better for retrieved documents (RQ2).

Variant	F1	Docs	Decision Acc.
STAR	0.4348	4.775	0.647
STAR w/o Relevance	0.4348	4.775	0.647
STAR w/o Redundancy	0.4417	5.155	0.628
STAR w/o Coverage	0.4396	5.070	0.605
STAR w/o Supportiveness	0.4213	3.915	0.759

controller can remain useful outside a strongly multi-hop setting, while its clearest benefits appear when evidence insufficiency is more structural and multi-hop.

Retrieval Efficiency. On HotpotQA, the calibrated confidence baseline is a meaningful adaptive competitor, expanding on 46.8% of queries. Even against this baseline, STAR uses fewer retrieved documents on average (4.775 vs. 5.340), slightly reduces the premature-stop rate (0.028 vs. 0.031), and still improves F1. On Natural Questions, STAR expands more aggressively than confidence (0.461 vs. 0.169), increasing average retrieved documents from 3.338 to 3.922; that extra cost is accompanied by a lower premature-stop rate (0.018 vs. 0.030) and higher F1 (0.320 vs. 0.315).

Compared with Fixed larger- k RAG, STAR remains more economical on both datasets while approaching the quality of always-expanded retrieval. Overall, STAR is neither uniformly cheapest nor uniformly best in F1, but it offers a competitive quality-cost trade-off by avoiding some structurally premature stops without paying the full document cost of always-expanded retrieval.

4.3 Ablation Study (RQ2)

Table 2 shows that the HotpotQA ablations assign distinct roles to the four signals. Relevance is inactive under the current calibration because its learned weight is zero, and removing it does not change the reported results. A post-hoc analysis on 527 labeled instances from the HotpotQA evaluation slice suggests that this is not explained by simple linear or rank redundancy with supportiveness, as their correlations are near zero (Pearson $r = 0.036$, Spearman $\rho = 0.044$). Instead, relevance shows little ability to distinguish sufficient from insufficient evidence sets (0.469 ± 0.130 vs. 0.500 ± 0.171 ; AUROC = 0.464). This suggests that, in the current dense-retrieval setup, relevance captures topical alignment but does not reliably measure whether the evidence adequately covers and supports the

answer. The zero relevance weight should therefore be interpreted as specific to the current retriever and calibration setting, rather than as evidence that relevance is generally unnecessary; it may become informative again under different retrievers or noisier retrieval conditions. By contrast, removing supportiveness causes the clearest answer-quality drop, reducing F1 from 0.4348 to 0.4213 and increasing the premature-stop rate from 0.028 to 0.047, which suggests that supportiveness is most directly tied to answer quality. Removing redundancy or coverage slightly increases F1 to 0.4417 and 0.4396, respectively, but these variants do so by expanding more often, retrieving more documents, and reducing decision accuracy. We therefore do not interpret those variants as stronger controllers overall; rather, they illustrate a different operating point that trades more retrieval cost for small answer-quality gains while weakening the quality of the **STOP/EXPAND** decision itself. This pattern suggests that redundancy and coverage mainly shape the cost-quality frontier and control behavior, whereas supportiveness contributes most directly to answer quality. The corresponding Natural Questions ablation (Table 5 in the appendix) shows the same qualitative trend: removing supportiveness hurts answer quality most clearly, whereas removing redundancy or coverage increases document cost and expansion behavior while degrading decision quality.

4.4 Failure-Mode and Case Analysis (RQ3)

To better understand STAR’s quality-cost trade-off, Table 3 aggregates cases in which confidence selects **STOP** but STAR selects **EXPAND**. STAR corrects 14 premature stops on HotpotQA and 16 on Natural Questions, mainly in high-redundancy and mixed-insufficiency cases. This concentration supports our claim that STAR’s gains come from identifying structurally insufficient evidence states rather than from uniformly expanding retrieval. At the same time, the larger number of **STOP**→**EXPAND** cases than corrections indicates that not every expansion improves the answer, consistent with the overall quality-cost trade-off. These cases clarify STAR’s role as a retrieval-control mechanism: expansion is useful when evidence that appears sufficient under a global confidence signal remains structurally narrow or weakly supportive. The non-corrected expansions also show that these structural signals are imperfect proxies, so STAR should be interpreted as a policy that navigates a quality-cost trade-off rather than as an oracle expansion policy.

Table 3: Failure-mode analysis of cases where the confidence baseline selects STOP but STAR selects EXPAND. Corrected premature stops are concentrated in high-redundancy and mixed-insufficiency cases, supporting the interpretation that STAR improves retrieval control by identifying structural evidence failures rather than by expanding indiscriminately.

Dataset	STAR reason	Stop→Expand	Corrected stops
HotpotQA	High redundancy	46	8
HotpotQA	Mixed insufficiency	114	6
Natural Questions	High redundancy	196	10
Natural Questions	Mixed insufficiency	157	6

Table 4 shows two representative corrections and illustrates how structural signals revise decisions that confidence would stop too early. For the HotpotQA question about Ruffin Horne McNeill Jr., confidence stops with *career*; STAR detects high redundancy, expands, and recovers *tenure*. For the Natural Questions art deco query, confidence returns *1920s and 1930s*, whereas STAR expands under mixed insufficiency and recovers *1910-1939*. Together with the aggregate counts, these examples illustrate targeted rather than indiscriminate expansion.

Table 4: Representative case studies illustrating STAR’s retrieval decisions (RQ3).

Dataset	Query (abridged)	Confidence-based adaptive retrieval	STAR
HotpotQA	Ruffin Horne McNeill Jr.: winning seasons in every year of what?	STOP → <i>career</i>	EXPAND (high redundancy) → <i>tenure</i>
Natural Questions	When did the design period known as art deco occur?	STOP → <i>1920s and 1930s</i>	EXPAND (mixed insufficiency) → <i>1910-1939</i>

4.5 Scope and Limitations

Our controlled evaluation uses slice-based benchmark corpora, a single **STOP/EXPAND** decision, and heuristic coverage and supportiveness proxies. These proxies may miss implicit or paraphrased question aspects and should not be interpreted as direct measurements of semantic entailment. Future work could augment them with learned semantic verifiers, natural language inference models, or generator-based evidence attribution. Although this setup isolates evidence-state diagnosis, it excludes web-scale and iterative multi-round retrieval. The fixed 1,000-query slices also motivate evaluation across additional slices, retrievers, and generators. STAR further adds pairwise similarity, aspect-document matching, lexical overlap, and multi-signal aggregation over the small initial evidence set, although document embeddings can be cached. Further systems optimization through batched aspect encoding and aspect-embedding caching remains future work. A full systems evaluation should jointly measure controller latency, retrieval latency, and end-to-end throughput.

5 Conclusion

We presented STAR, an adaptive RAG controller that treats expansion as diagnosis over the current evidence state rather than as thresholding a single confidence value. STAR combines interpretable evidence signals to decide whether retrieval should stop or expand. Our results suggest that adaptive retrieval should consider not only how confident the system appears, but also why the current evidence remains insufficient. Under the fixed evaluation setting, STAR shows higher F1 and fewer premature stops than confidence-based adaptive retrieval, while approaching always-expanded retrieval at lower document cost. Gains are clearest on multi-hop HotpotQA and remain directionally consistent on Natural Questions under a smaller expansion budget. STAR is not uniformly cheapest or best, but offers a competitive quality-cost trade-off by diagnosing structural evidence insufficiency. Future work should test the formulation on larger retrieval indices and more expressive generation pipelines.

References

- [1] Manoj Guhan Arivazhagan, Lan Liu, Peng Qi, Xinchu Chen, William Yang Wang, and Zhiheng Huang. 2023. Hybrid Hierarchical Retrieval for Open-Domain Question Answering. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 10680–10689. <https://doi.org/10.18653/v1/2023.findings-acl.679>
- [2] Akari Asai, Zequi Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. <https://openreview.net/forum?id=hSyW5go0v8> ICLR 2024.
- [3] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving Language Models by Retrieving from Trillions of Tokens. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, Baltimore, Maryland, USA, 2206–2240. <https://proceedings.mlr.press/v162/borgeaud22a.html>
- [4] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to Answer Open-Domain Questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Vancouver, Canada, 1870–1879. <https://doi.org/10.18653/v1/P17-1171>
- [5] Yung-Sung Chuang, Wei Fang, Shang-Wen Li, Wen-tau Yih, and James Glass. 2023. Expand, Rerank, and Retrieve: Query Reranking for Open-Domain Question Answering. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 12131–12147. <https://doi.org/10.18653/v1/2023.findings-acl.768>
- [6] Omer Feldman and Ran El-Yaniv. 2019. Multi-Hop Paragraph Retrieval for Open-Domain Question Answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 2296–2309. <https://doi.org/10.18653/v1/P19-1222>
- [7] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. REALM: Retrieval Augmented Language Model Pre-Training. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*. PMLR, Virtual, 3929–3938. <https://proceedings.mlr.press/v119/guu20a.html>
- [8] Gautier Izacard and Edouard Grave. 2021. Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, Online, 874–880. <https://doi.org/10.18653/v1/2021.eacl-main.74>
- [9] Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot Learning with Retrieval Augmented Language Models. *Journal of Machine Learning Research* 24, 251 (2023), 1–43. <https://jmlr.org/papers/v24/23-0037.html>
- [10] Soyeon Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Question Complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics, Mexico City, Mexico, 7036–7050. <https://doi.org/10.18653/v1/2024.naacl-long.389>
- [11] Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active Retrieval Augmented Generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Singapore, 7969–7992. <https://doi.org/10.18653/v1/2023.emnlp-main.495>
- [12] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 6769–6781. <https://doi.org/10.18653/v1/2020.emnlp-main.550>
- [13] Muhammad Khalifa, Alexander Fabbri, Mehrdad Moradshahi, Zhengxiang Wang, Raymond Fok, Yulia Yang, Zhilin Wang, Nanyun Peng, Yejin Choi, and Yulia Tsvetkov. 2023. Few-Shot Reranking for Multi-Hop QA via Language Model Prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*. Association for Computational Linguistics, Toronto, Canada, 4305–4330. <https://doi.org/10.18653/v1/2023.findings-acl.266>
- [14] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics* 7 (2019), 453–466. https://doi.org/10.1162/tacl_a_00276
- [15] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems* 33. Curran Associates, Inc., Red Hook, NY, USA, 9459–9474. <https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html>
- [16] Ye Liu, Kazuma Hashimoto, Yingbo Zhou, Semih Yavuz, Caiming Xiong, and Philip Yu. 2021. Dense Hierarchical Retrieval for Open-Domain Question Answering. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Association for Computational Linguistics, Punta Cana, Dominican Republic, 188–200. <https://doi.org/10.18653/v1/2021.findings-emnlp.19>
- [17] Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Reader-Guided Passage Reranking for Open-Domain Question Answering. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Association for Computational Linguistics, Online, 344–350. <https://doi.org/10.18653/v1/2021.findings-acl.29>
- [18] Hongjing Qian, Zhenghao Du, Yifei Wang, Zhuoran Wang, Weize Gao, Yutao Shen, Xueqi Cheng, Lei Hou, Wayne Xin Zhao, and Ji-Rong Wen. 2024. Grounding Language Models with Chunking-Free In-Context Retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Miami, Florida, USA, 9688–9702. <https://doi.org/10.18653/v1/2024.emnlp-main.552>
- [19] Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-Context Retrieval-Augmented Language Models. *Transactions of the Association for Computational Linguistics* 11 (2023), 1316–1331. https://doi.org/10.1162/tacl_a_00605
- [20] Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. REPLUG: Retrieval-Augmented Black-Box Language Models. <https://doi.org/10.48550/arXiv.2301.12652>
- [21] Haitian Sun, Tania Bedrax-Weiss, and William W. Cohen. 2019. PullNet: Open Domain Question Answering with Iterative Retrieval on Knowledge Bases and Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 2380–2390. <https://doi.org/10.18653/v1/D19-1242>
- [22] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. 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 (Volume 1: Long Papers)*. Association for Computational Linguistics, Toronto, Canada, 10014–10037. <https://doi.org/10.18653/v1/2023.acl-long.557>
- [23] Wenhan Xiong, Xiang Li, Srinivasan Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oguz. 2021. Answering Complex Open-Domain Questions with Multi-Hop Dense Retrieval. <https://openreview.net/forum?id=EMHoBG0avc1> ICLR 2021.
- [24] Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective Retrieval Augmented Generation. <https://doi.org/10.48550/arXiv.2401.15884> CoRR abs/2401.15884.
- [25] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium, 2369–2380. <https://doi.org/10.18653/v1/D18-1259>

A Reproducibility Supplement

This supplement summarizes the protocol details needed to reproduce the experiments.

A.1 Protocol and Splits

For each dataset, we build a task-specific retrieval corpus and evaluate on a fixed 1,000-example validation slice. Each run follows the same sequence: initial retrieval with budget k_{init} , structural diagnosis of the retrieved evidence set, sufficiency scoring, a one-step **STOP/EXPAND** decision, and answer generation from the final evidence set. If the policy chooses **EXPAND**, the system performs exactly one additional retrieval step with budget $k_{\text{exp}} > k_{\text{init}}$; there is no iterative multi-round retrieval loop in the reported setting. In the main paper configuration, the slice-based corpus setup draws both the retrieval corpus and the query set from the validation split, with a shared initial budget of $k_{\text{init}} = 3$. For a slice size of 1,000, `src/run_experiments.py` uses disjoint query slices by default: indices 0-999 for calibration and 1000-1999 for evaluation. These defaults can be overridden with explicit calibration and evaluation query-range flags. Overlapping calibration and evaluation slices are rejected unless `-allow-overlap-splits` is explicitly set. The expansion budget is dataset-specific: HotpotQA uses $k_{\text{exp}} = 8$, whereas Natural Questions uses $k_{\text{exp}} = 5$. For Natural Questions, we additionally use shorter document chunks with a maximum length of 180 tokens and a stride of 90 to reduce distractors under expansion. Unless otherwise noted, runs use a fixed random seed of 42, the dense encoder BAAI/bge-small-en-v1.5, the evidence-based labeling strategy, and the same generator and prompt template across all compared methods. The released configuration preserves these settings in the run metadata, and the main experiments are orchestrated by `src/run_experiments.py`.

A.2 Signals and Calibration

The controller uses four interpretable signals: retrieval relevance, redundancy, coverage, and supportiveness. Relevance is computed from min-max normalized retriever scores; redundancy is the average pairwise cosine similarity among retrieved document embeddings; coverage is aggregated over heuristically extracted question aspects from normalized content-bearing unigrams, bigrams, and simple entity-like cues; and supportiveness combines normalized retriever strength with question-document lexical overlap. Coverage and supportiveness are intentionally heuristic proxies rather than semantic verifiers, so the paper evaluates them through downstream control quality, ablations, and failure-mode analysis rather than treating them as direct measurements of sufficiency. We calibrate nonnegative signal weights together with the decision threshold τ using silver labels derived from evidence outcomes. The initial evidence set D is labeled as sufficient if weak support for the gold answer already appears in D , and insufficient if it appears only after one expansion to D^+ . Weak support requires both normalized answer-string matching and non-trivial question-document lexical alignment, with the overlap threshold fixed at 0.2 in the default configuration. Examples with no weak support in either D or D^+ are excluded from calibration rather than forced into either class, reducing distortion from uninformative negatives. Calibration

is implemented in `src/calibrate.py`. The default paper setting uses `label_strategy=evidence`; the code also exposes a more expensive `hybrid_generation` variant for sensitivity analysis. For the confidence baseline, we calibrate a separate threshold over a score defined only from the retriever top-1 score and the top1-top2 margin, selecting a threshold that keeps the expansion rate within 0.2-0.6 so that the baseline does not collapse to either Vanilla RAG or Fixed larger- k retrieval. Released calibration files additionally record the seed, slice ranges, overlap setting, overlap threshold, prompt version, label strategy, and any ablated signal so that full STAR and variants such as *STAR w/o Coverage* can be reproduced consistently.

Table 5. Natural Questions ablation summary. The pattern matches HotpotQA but is weaker: removing supportiveness causes the clearest answer-quality drop, removing redundancy or coverage increases document cost while lowering decision accuracy, and relevance has little effect. Higher is better for F1 and decision accuracy; lower is better for retrieved documents.

Variant	F1	Docs	Decision Acc.
STAR	0.3195	3.922	0.597
STAR w/o Relevance	0.3195	3.922	0.597
STAR w/o Redundancy	0.3180	4.058	0.505
STAR w/o Coverage	0.3192	4.148	0.512
STAR w/o Supportiveness	0.3118	3.386	0.751

A.3 Evaluation and Artifacts

We report both answer-quality and retrieval-efficiency metrics. Answer quality is measured with exact match (EM) and token-level F1 under the same evidence-constrained generation setting across all methods, and retrieval efficiency is measured using the number of retrieval calls, the number of retrieved documents, and the expansion rate. Evaluation is implemented in `src/evaluate.py`. In the released paper setting, answer generation uses the OpenAI generation path with `gpt-4.1-mini`, while EM and token-level F1 are computed afterward by the local evaluation script; this separation is important because generation and scoring do not happen in the same component. To minimize generator confounds, the evaluation logs record the generator type, model version, prompt-template version, embedding model, calibration source, and confidence-calibration source, and comparisons in the main tables keep these fixed while varying only the retrieval policy. For ablations, we remove one signal at a time and re-calibrate the remaining controller weights on the same calibration slice; we report these variants using the naming convention *STAR w/o <Signal>*. The repository also includes scripts for case analysis and result summarization. We save calibration JSON files, confidence-baseline calibration JSON files, evaluation CSV files, ablation summaries, case-analysis tables, and sidecar `*.meta.json` files under a `results/` directory. These metadata files record the generator type, model version, label strategy, prompt-template version, embedding model, calibration source, confidence-calibration source, retrieval cache directory, shared OpenAI cache path, and OpenAI cache statistics when applicable; if a manifest is provided, they also preserve the manifest path and identifier. This makes reruns, audit trails, and exact STAR-versus-baseline comparisons easier to reconstruct from saved artifacts. Code, configuration files, and example outputs are available at <https://github.com/owonye/KDD.git>.