Towards Robust Retrieval-Augmented Generation via Reliability-Aware Adaptive Evidence Selection

Minkyu Che[1], **Jihwan Oh**[1], **Jeonghwan Choi**[1], **Jeesu Jung**[1,*], **Hwanjun Song**[1,2,*]
[1]Korea Advanced Institute of Science and Technology, [2]Cluvion
{mingyu8020, songhwanjun}@kaist.ac.kr

Abstract

Retrieval-augmented generation (RAG) is increasingly deployed on unverified corpora, where an adversary can inject poisoned documents to steer the model toward an attacker-specified answer. Existing defenses rely on auxiliary heuristics and break down once their assumptions about the attack type or poisoning level are violated. We propose SAFER-RAG, a defense that derives its reliability signal from the adversarial objective itself. Since poisoned documents are explicitly optimized to rank highly for the target query, they remain anchored at top ranks under semantically equivalent rewrites, while clean documents shift naturally. SAFER-RAG captures this rank-stability gap through spectral analysis, partitions retrieved chunks into Safe and Unsafe Sets, and adaptively selects evidence under a safe-first strategy that avoids over-filtering useful evidence on clean or lightly poisoned corpora. Across three QA benchmarks, three attack types, and four poisoning levels, SAFER-RAG consistently achieves the highest accuracy and the lowest attack success rate. The code will be released upon acceptance.

1 Introduction

Retrieval-augmented generation (RAG) has become a standard paradigm for grounding large language models (LLMs) in external knowledge for knowledge-intensive tasks (Maynez et al., 2020; Zhao et al., 2023). However, the very mechanism that makes RAG powerful also makes it fragile: because retrieved documents directly condition the generation, an adversary who injects even a small number of *poisoned* documents into the corpus can steer the model toward an attacker-specified answer (Zou et al., 2025; Bigdeli et al., 2025). This risk is especially pressing for RAG systems built on unverified sources, such as open web pages or user-generated content, where reliable answering demands trust judgments, not just relevance.

Existing defenses against corpus poisoning fall into two broad categories. *Reliability-based* methods score documents using auxiliary cues such as perplexity (Zhong et al., 2023), query–document similarity (Yao et al., 2025), or cross-document contradiction (Shen et al., 2025). *Clustering-based* methods instead separate poisons via embedding-space proximity (Cheng et al., 2025) or semantic-structural grouping (Si et al., 2025). The former assumes a low poisoning level where clean documents still dominate the retrieved set, and breaks down once poisoned documents displace them at the top ranks. The latter assumes a specific attack type with distinct surface artifacts, and collapses under natural-looking poisons, while additionally requiring cluster counts and thresholds tuned per attack. Yet in practice, both the attack type and poisoning level are *unknown* beforehand in real-world RAG deployments, leaving existing defenses unreliable under realistic settings.

We sidestep both assumptions by anchoring our defense on a structural artifact that the adversarial objective itself imposes on poisoned documents. Since poisoned documents are explicitly optimized to rank highly for the target query, they remain anchored at high ranks under semantic-preserving query perturbations such as rewriting, whereas clean documents shift naturally according to which terms each rewrite emphasizes (refer to Section 4.2 for our analysis of this rank-stability gap). Crucially, this contrast stems from the attack objective itself rather than surface patterns, making the signal robust to attack type and level.

Motivating by this insight, we propose a novel method, SAFER-RAG (Selection with Adaptively Filtered Evidence by Reliability for RAG) which turns query-induced *rank behavior* into a reliability signal and uses it to guide evidence selection. As illustrated in Figure 1, SAFER-RAG proceeds in

*Co-corresponding authors.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 49353–49374
October 24-29, 2026 ©2026 Association for Computational Linguistics

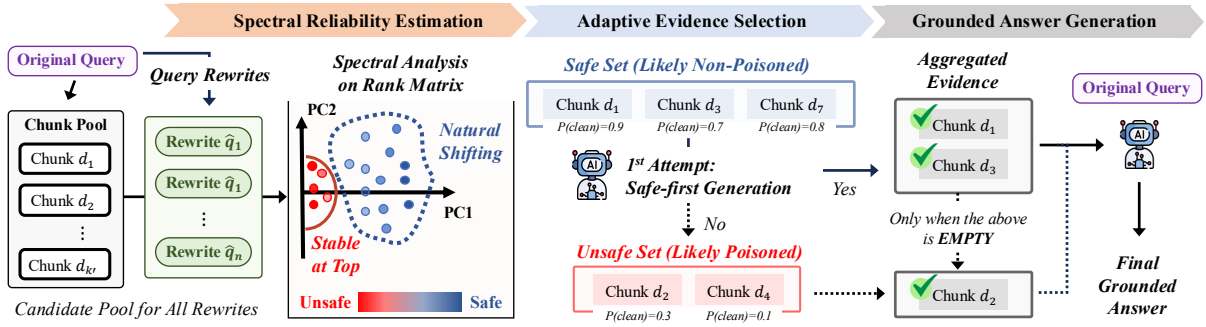


Figure 1: Overview of SAFER-RAG: It rewrites the query multiple times and applies spectral analysis on the rank matrix to split chunks into a Safe Set and an Unsafe Set. Supporting evidence is then adaptively selected from the Safe Set via the highest-reliability candidate answer, falling back to the Unsafe Set only when none is valid. The final answer is grounded solely on the selected supporting evidence, aggregated with the original query.

three stages: **(i) Spectral Reliability Estimation**, where SAFER-RAG generates semantically equivalent rewrites for each query and analyzes how every candidate chunk reorders across them. Since rank vectors collected over many rewrites are high-dimensional and noisy, SAFER-RAG projects them onto their leading principal directions, exposing the dominant axis along which stable-at-top (suspicious) and naturally shifting (clean) behaviors separate. This contrast lets SAFER-RAG partition retrieved chunks into a *Safe Set* of likely reliable evidence and an *Unsafe Set* of suspicious chunks, without manually calibrated thresholds; **(ii) Adaptive Evidence Selection**, where evidence chunks are selected under a *safe-first* strategy. SAFER-RAG generates candidate answers from the Safe Set and selects the one whose supporting evidence has the highest average reliability, escalating to the Unsafe Set only when no valid candidate can be grounded; and **(iii) Grounded Answer Generation**, where the final answer is produced from only the selected supporting evidence, keeping the answering context compact and free of unverified chunks.

SAFER-RAG thus offers three advantages. First, since the signal arises from the attack objective itself, it remains robust across attack types and levels. Second, since the pipeline can run independently per sub-query, it extends to multi-hop reasoning through query decomposition. Third, since selection keeps only answer-supporting chunks, it improves accuracy even on clean corpora.

We evaluate SAFER-RAG across three poisoning attack types and four intensity levels under realistic conditions on two single-hop QA datasets (NQ (Kwiatkowski et al., 2019), MS-MARCO (Bajaj et al., 2016)) and one multi-hop QA dataset (HotpotQA (Yang et al., 2018)). We compare SAFER-RAG with four state-of-the-art baselines, including EcoSafeRAG (Yao et al., 2025), ReliabilityRAG (Shen et al., 2025), SeCon-RAG (Si et al., 2025) and RAGuard (Cheng et al., 2025). SAFER-RAG achieves the highest robustness and the lowest performance variance across attack types and poisoning levels, while avoiding the over-filtering that degrades prior defenses on clean and lightly poisoned corpora.

2 Related Work

Poisoning Attacks on RAG. Modern RAG systems are vulnerable to poisoning attacks, where attackers inject malicious documents into external knowledge bases to steer generation toward attacker-specified answers (Zhong et al., 2023; Zou et al., 2025; Nazary et al., 2025). Effective poisons must be both retrievable under the target query and plausible enough to evade detection. PoisonedRAG (Zou et al., 2025) prioritizes the former by appending query-related content, while FSAP (Bigdeli et al., 2025) prioritizes the latter via few-shot prompting that suppresses surface-level artifacts. Beyond corpus poisoning, other work shifts the attack surface to the retrieval-to-generation interface through indirect prompt injection (Shafran et al., 2025; Chaturvedi et al., 2025). We focus on corpus poisoning, a more severe threat that compromises the knowledge base itself and persists across queries, leaving prompt-level threats to orthogonal defenses.

Defenses against Corpus Poisoning. A growing line of work investigates how to detect or filter poisoned evidence at retrieval or generation time (Xiang et al., 2026; Zhou et al., 2025). Existing methods can be grouped into reliability-based and clustering-based approaches.

Reliability-based defenses assign per-document

reliability scores from auxiliary signals and filter retrieved evidence accordingly. RAGuard (Cheng et al., 2025) combines chunk-wise perplexity with query-document similarity filtering, while ReliabilityRAG (Shen et al., 2025) builds an NLI-based contradiction graph and selects a consistent majority via reliability-weighted maximum independent set. Related approaches extend this line with LLM self-assessment or per-passage voting (Xiang et al., 2026; Zhou et al., 2025). These methods work well under low poisoning, but fail as injection grows and the benign-majority assumption no longer holds, allowing poisons to both pass per-document checks and form consistent majority.

Clustering-based defenses instead assume that malicious documents form separable groups distinct from benign content. EcoSafeRAG (Yao et al., 2025) uses DBSCAN-based clustering with a bait-guided diversity check, SeCon-RAG (Si et al., 2025) uses entity-intent-relation extraction with centroid-based filtering and conflict-aware checking. Other work exploits weak graph connectivity (Zheng et al., 2025) or retriever-stage partitioning and masking (Pathmanathan et al., 2026). These methods are effective against query-conditioned attacks like PoisonedRAG (Zou et al., 2025), where shared query tokens cause poisons to cluster tightly in embedding space, but are less effective against natural-looking attacks like FSAP (Bigdeli et al., 2025), whose LLM-generated poisons scatter among benign documents.

3 Method: SAFER-RAG

We design SAFER-RAG around three objectives, addressed by three corresponding stages: (i) spectral reliability estimation (in Section 3.1), (ii) adaptive evidence selection (in Section 3.2), and (iii) grounded answer generation (in Section 3.3).

First, since the attack type and poisoning level are unknown to the defender in practice, we seek a reliability signal that does not depend on either factor. Second, since aggressive filtering can overly clean benign or lightly poisoned corpora, we adopt an adaptive selection strategy that examines evidence from least to most likely poisoned, falling back to suspicious chunks only when no valid answer can be grounded from cleaner ones. Third, to avoid context overhead and conflicting evidence in generation, we ground the final answer only on the answer-supporting chunks identified.

The overall pipeline of SAFER-RAG is illustrated in Figure 1, while Algorithm 1 in Appendix A details the complete procedure in pseudo-code.

3.1 Spectral Reliability Estimation

The first stage aims at estimating a per-chunk reliability score, the posterior probability that a chunk is clean, by analyzing rank changes across semantically equivalent rewrites of the query.

In detail, we first generate n rewritten queries $\hat{Q} = \{\hat{q}_1, \ldots, \hat{q}_n\}$ from an initial query q in a single LLM rollout, where each rewrite preserves the original information need while varying surface form, lexical choice, and specificity level (the prompt and analysis are in Table 16 and Appendix E). This reveals how the retrieval behavior of each chunk responds to different phrasings of the same question without changing what is being asked.

Candidate Pool Construction. Before measuring rank changes across rewrites $\hat{Q}$, we construct a single set of candidate chunks D_{cand} to evaluate. Otherwise, each rewritten query would retrieve a different top-k set, leaving some chunks present under one rewrite but missing from another and making rank comparison ill-defined. To this end, we retrieve a larger *candidate pool* of size $k' > k$[1] from the *original* query q as:

$$D_{\text{cand}} = \{d_1, d_2, \ldots, d_{k'}\} = R(q, D, k'), \quad (1)$$

where $R(q, D, k')$ denotes the top-k' documents retrieved by the retriever for query q over corpus D. This candidate pool guarantees a well-defined rank for every chunk under every rewrite.

Rank Vector Generation. For each retrieved candidate $d \in D_{\text{cand}}$, we re-rank D_{cand} under each rewritten query $\hat{q}_i \in \hat{Q}$ and collect the resulting ranks into a *rank vector* $\mathbf{v}$, a sequence that records the position of d within D_{cand} across all n rewrites:

$$\mathbf{v}_d = [r(d, \hat{q}_1),\ r(d, \hat{q}_2),\ \ldots,\ r(d, \hat{q}_n)], \quad (2)$$

where $r(d, \hat{q}_i)$ is the rank of d in D_{cand} when re-scored by the retrieval R under $\hat{q}_i$, capturing how d's retrieval priority shifts under rephrasing.

Spectral Reliability Estimation. Recent work have shown that since poisoned chunks are crafted to closely match the target query, they remain highly ranked even under rephrasing of the query

[1] A pool too small fails to capture rank drops, while a pool too large adds noise from low-ranked chunks as candidates. We use $k' = 4k$ as a balance. See Appendix H.2 for analysis.

(Zou et al., 2025; Xian et al., 2025), making paraphrasing- or rewriting-based defenses ineffective at removing them. Instead, *we turn this limitation into a signal*, since poisoned chunks remain anchored at top ranks across rewrites while non-poisoned chunks, both relevant and irrelevant, reorder naturally with each rewrite.

Specifically, because raw rank vectors are high-dimensional and noisy, directly comparing chunks across the vectors is challenging. We therefore extract a *spectral* representation that can capture the dominant axis of variation. We stack the rank vectors of all chunk candidates to form a *rank matrix* $V = \left[\mathbf{v}_{d_1}, \mathbf{v}_{d_2}, \ldots, \mathbf{v}_{d_{k'}}\right]^\top$, where each row corresponds to the rank vector of one candidate chunk. Then, we perform eigenvector decomposition on its covariance matrix $V^\top V$. This yields eigenvectors $\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n\}$ ordered by the variance of rank changes they capture. We only use the leading eigenvector $\mathbf{u}_1$,[2] since the first component captures the dominant axis along which poisoned and non-poisoned chunks separate.

Our goal is to distinguish poisoned from non-poisoned chunks, so we project each chunk's rank vector onto $\mathbf{u}_1$ to obtain a scalar score that summarizes its rank behavior along this dominant axis, $x_d = \mathbf{v}_d^\top \mathbf{u}_1$. Along this axis, $\{x_d\}$ naturally splits into two modes, one for consistently high-rank behavior ("poisoned" chunks) and the other for rewrite-dependent reordering ("non-poisoned" ones). To separate these two modes without any hand-tuned threshold, we fit a two-component Gaussian Mixture Model (GMM) (Reynolds et al., 2009) on $\{x_d\}$ via expectation-maximization.

We assign the component with the smaller mean value to the "unsafe" class, since lower rank values correspond to high-rank positions, and define the *reliability score* of a candidate chunk d as its posterior probability when the mode (g) is clean:

$$p(g = \text{clean} \mid x_d) = p(g)\, p(x_d \mid g)/p(x_d), \quad (3)$$

where $g \in \{\text{clean}, \text{unclean}\}$. Using this reliability score, we partition the candidate pool D_{cand} into two disjoint subsets, the *Safe Set* of chunks that are more likely non-poisoned and the *Unsafe Set* of chunks that are more likely poisoned:

$$\begin{aligned} S_{\text{safe}} &= \{\, d \in D_{\text{cand}} \mid P(g{=}\text{clean} \mid x_d) \geq 0.5 \,\} \\ S_{\text{unsafe}} &= D_{\text{cand}} \setminus S_{\text{safe}}. \end{aligned} \quad (4)$$

3.2 Adaptive Evidence Selection

Given the two sets, SAFER-RAG adopts a *safe-first* strategy for selecting answer-supporting evidence. It first attempts to ground candidate answers in S_{Safe}, and falls back to S_{Unsafe} only when no valid answer can be supported from the Safe Set.

SAFER-RAG first selects the top-k chunks from S_{Safe} by their retrieval scores against the original query q, forming the safe evidence pool D_{Safe}. A generator LLM then produces a set of candidate answer–evidence pairs $\{(a, D_{\text{supp}}(a))\}$ from D_{Safe}, where $D_{\text{supp}}(a) \subseteq D_{\text{Safe}}$ denotes the chunks supporting answer a. A candidate $(a, D_{\text{supp}}(a))$ is considered *valid* if a is non-empty and grounded in at least one chunk, that is, $D_{\text{supp}}(a) \neq \emptyset$.

Even within D_{Safe}, some malicious chunks may remain due to imperfect partitioning, leading the generator LLM to produce multiple valid answer–evidence pairs that are not equally trustworthy. To guard against this, SAFER-RAG selects among them the one whose supporting evidence has the highest expected reliability:

$$p(\text{clean} \mid a) = \mathbb{E}_{d \in D_{\text{supp}}(a)}[\, p(g{=}\text{clean} \mid x_d)\,]. \quad (5)$$

The finally selected supporting evidence is then:

$$D^*_{\text{supp}} = \arg \max_{D_{\text{supp}}(a)} p(\text{clean} \mid a), \quad (6)$$

If no valid answer–evidence pair can be generated from D_{Safe}, SAFER-RAG falls back to the Unsafe Set and repeats the same procedure on S_{Unsafe}, applying the same criterion to obtain D^*_{supp}. This adaptive design avoids both over-filtering on clean or lightly poisoned corpora and under-filtering under heavy poisoning, since the pool consulted adapts to each query's reliability landscape rather than being fixed in advance.

3.3 Grounded Answer Generation

Given the selected evidence, SAFER-RAG produces the final answer with respect to the original query q. For multi-hop queries, SAFER-RAG additionally decomposes q into sub-queries $\{q^{(1)}, \ldots, q^{(h)}\}$ via an LLM, applies the previous two stages independently to each sub-query $q^{(i)}$, and aggregates the resulting supporting evidence into $D_{\text{final}} =$

[2] We find that using additional components beyond the first does not provide further discriminative information (see Appendix B), and that this spectral feature outperforms simpler rank-based statistics such as mean rank, rank variance, and rank entropy (see Appendix C).

Poisoning Level	Method	Single-hop						Multi-hop		
		NQ			MS-MARCO			HotpotQA		
		PoisonedRAG Acc ↑ / ASR ↓	FSAP Acc ↑ / ASR ↓	Mixed Acc ↑ / ASR ↓	PoisonedRAG Acc ↑ / ASR ↓	FSAP Acc ↑ / ASR ↓	Mixed Acc ↑ / ASR ↓	PoisonedRAG Acc ↑ / ASR ↓	FSAP Acc ↑ / ASR ↓	Mixed Acc ↑ / ASR ↓
Clean	LLM-only	– – – – –	56.0 / -	– – – – –	– – – – –	60.5 / -	– – – – –	– – – – –	52.0 / -	– – – – –
	VanillaRAG	– – – – –	79.5 / -	– – – – –	– – – – –	86.6 / -	– – – – –	– – – – –	71.0 / -	– – – – –
	RAGuard	– – – – –	72.2 / -	– – – – –	– – – – –	80.0 / -	– – – – –	– – – – –	59.0 / -	– – – – –
	ReliabilityRAG	– – – – –	76.0 / -	– – – – –	– – – – –	74.5 / -	– – – – –	– – – – –	49.0 / -	– – – – –
	EcoSafeRAG	– – – – –	77.1 / -	– – – – –	– – – – –	85.6 / -	– – – – –	– – – – –	62.0 / -	– – – – –
	SeCon-RAG	– – – – –	77.8 / -	– – – – –	– – – – –	86.2 / -	– – – – –	– – – – –	68.0 / -	– – – – –
	SAFER-RAG	– – – – –	**84.0 / -**	– – – – –	– – – – –	**88.2 / -**	– – – – –	– – – – –	**79.0 / -**	– – – – –
Low	VanillaRAG	52.6 / 44.9	55.6 / 41.6	54.5 / 43.6	64.5 / 30.0	62.5 / 35.6	63.5 / 32.7	47.0 / 45.0	36.0 / 48.0	34.0 / 49.0
	RAGuard	57.1 / 12.9	62.6 / 18.9	60.0 / 18.2	60.2 / 21.8	54.5 / 24.0	60.9 / 22.9	47.0 / 45.0	42.0 / 39.0	34.0 / 49.0
	ReliabilityRAG	47.3 / 40.9	50.4 / 34.4	50.9 / 38.5	62.2 / 24.2	57.3 / 23.6	62.5 / 23.8	44.0 / 44.0	32.0 / 45.0	41.0 / 36.0
	EcoSafeRAG	50.9 / 44.7	55.4 / 41.1	52.7 / 41.8	64.9 / 28.4	65.5 / 32.7	65.3 / 33.5	46.0 / 45.0	33.0 / 47.0	23.0 / 55.0
	SeCon-RAG	68.7 / 9.6	65.8 / 17.6	67.3 / 16.4	72.9 / 9.8	70.9 / 16.4	73.5 / 13.1	65.0 / **5.0**	65.0 / 14.0	60.0 / **11.0**
	SAFER-RAG	**76.4 / 4.4**	**78.4 / 9.6**	**77.1 / 8.0**	**79.8 / 7.1**	**81.8 / 2.6**	**75.1 / 5.1**	**70.0** / 7.0	**66.0 / 9.0**	**67.0** / 12.0
Medium	VanillaRAG	43.8 / 54.4	50.2 / 48.9	45.5 / 52.7	54.9 / 38.2	52.7 / 43.6	53.8 / 40.9	41.0 / 49.0	29.0 / 56.0	27.0 / 57.0
	RAGuard	55.3 / 19.5	57.3 / 28.6	56.4 / 20.0	50.2 / 35.3	49.1 / 27.4	49.6 / 31.3	39.0 / 44.0	44.0 / 42.0	34.0 / 55.0
	ReliabilityRAG	33.8 / 57.1	42.4 / 49.1	38.2 / 49.1	58.2 / 31.8	56.4 / 27.4	58.4 / 29.6	37.0 / 51.0	28.0 / 59.0	38.0 / 46.0
	EcoSafeRAG	42.5 / 55.6	48.0 / 51.1	44.0 / 50.2	58.2 / 35.8	54.5 / 37.5	58.5 / 36.7	43.0 / 43.0	28.0 / 55.0	32.0 / 59.0
	SeCon-RAG	66.4 / 16.5	59.8 / 22.4	62.5 / 21.8	70.4 / 14.7	72.0 / 22.5	72.4 / 18.5	61.0 / **8.0**	58.0 / 19.0	63.0 / 16.0
	SAFER-RAG	**76.6 / 4.2**	**77.5 / 9.1**	**77.5 / 5.5**	**76.9 / 8.2**	**76.4 / 9.6**	**74.0 / 9.3**	**68.0 / 8.0**	**69.0 / 12.0**	**70.0 / 11.0**
High	VanillaRAG	37.8 / 57.1	37.5 / 54.0	38.9 / 58.2	49.3 / 41.5	42.5 / 47.3	45.8 / 48.0	36.0 / 54.0	23.0 / 66.0	28.0 / 57.0
	RAGuard	54.7 / 24.1	56.2 / 25.3	54.0 / 34.5	47.3 / 39.1	47.3 / 28.6	47.8 / 33.8	44.0 / 46.0	37.0 / 56.0	32.0 / 58.0
	ReliabilityRAG	28.7 / 61.1	27.1 / 63.6	29.1 / 61.8	51.3 / 38.2	58.2 / 34.5	58.5 / 38.9	28.0 / 62.0	21.0 / 68.0	26.0 / 56.0
	EcoSafeRAG	38.7 / 57.6	38.6 / 58.6	37.5 / 57.5	52.2 / 39.6	50.9 / 43.6	51.6 / 44.4	37.0 / 49.0	25.0 / 60.0	26.0 / 55.0
	SeCon-RAG	58.5 / 20.0	52.9 / 35.1	57.5 / 21.8	65.3 / 17.8	65.5 / 24.1	65.5 / 20.9	57.0 / **10.0**	53.0 / 26.0	60.0 / 21.0
	SAFER-RAG	**74.5 / 3.1**	**72.2 / 8.2**	**73.6 / 6.9**	**80.5 / 10.9**	**78.4 / 14.5**	**79.5 / 14.0**	**60.0** / 11.0	**68.0 / 14.0**	**62.0 / 10.0**

Table 1: Generation performance (Acc and ASR) of six methods across three poisoning attack types and four poisoning levels on three datasets. The best scores are marked in **bold**.

$\cup_{i=1}^{h} D_{\text{supp}}^{*,(i)}$, where $D_{\text{supp}}^{*,(i)}$ denotes the evidence selected for $q^{(i)}$ and $h = 1$ recovers the single-hop case. The generator LLM then produces the final answer a^* conditioned on q and D_{final}. Refer to Appendix A for details on the query decomposition.

This design also reduces context overhead. Only the chunks supporting the chosen answer for each sub-query are kept, so the final context remains compact, free of residual poisoned or irrelevant chunks, and free of conflicts since each sub-query contributes evidence for a single answer.

4 Evaluation

We conduct experiments to validate three claims about SAFER-RAG: (1) it is robust against diverse corpus poisoning attack types and intensity levels; (2) its robustness stems from our core design components; and (3) it is efficient and generalizes across backbone models. See Appendices F and G for datasets, model and retrieval details.

Datasets. We evaluate on three popular open-domain QA benchmarks: NQ (Kwiatkowski et al., 2019), MS-MARCO (Bajaj et al., 2016) for single-hop QA, HotpotQA (Yang et al., 2018) for multi-hop reasoning QA. We use 550 queries for NQ and MS-MARCO, and 100 queries for HotpotQA.

On each benchmark, we target the retrieval corpus with two representative poisoning attacks of distinct character. *PoisonedRAG* (Zou et al., 2025) appends query-related content to maximize retrievability, while *FSAP* (Bigdeli et al., 2025) uses few-shot prompting to produce natural-looking poisons that evade surface-level detection. We additionally include a *Mixed* setting that combines the two in a 1:1 ratio. Following prior poisoning studies (Xiang et al., 2026; Shen et al., 2025), we vary the attack intensity across four levels in {Clean, Low, Medium, High}, corresponding to $\{0, 2, 4, 6\}$ poisoned documents injected into the retrieval corpus per target query, in order to examine how defenses scale with the poisoning level. Since neither the attack type nor the poisoning level is known to the defender in advance, a deployable defense must remain robust across both axes rather than excel under any single configuration.

Baselines. We compare our method against four representative defenses from two existing categories, along with VanillaRAG, a no-defense baseline. *Reliability-based* defenses score each retrieved document with auxiliary signals and filter accordingly; representatives are RAGuard (Cheng et al., 2025) and ReliabilityRAG (Shen et al., 2025). *Clustering-based* defenses filter via embedding-space or semantic-structural grouping;

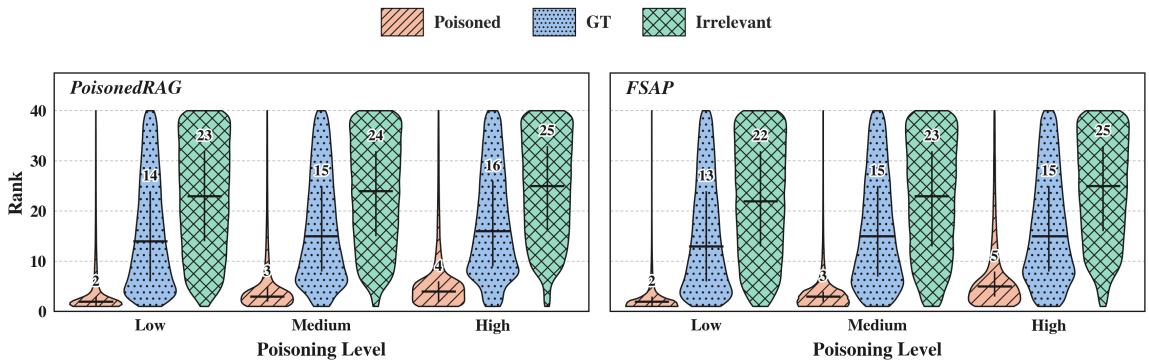


Figure 2: Rank distributions of poisoned, ground-truth (GT), and irrelevant (IRR) documents under query perturbations across three poisoning levels, aggregated over all three datasets (Rank 1 is the top result): PoisonedRAG (left) and FSAP (right). The thin vertical line inside each violin spans the first to third quartile (25th–75th percentile) of the rank distribution, and the thick horizontal tick marks the median, whose value is annotated inside each violin.

representatives are EcoSafeRAG (Yao et al., 2025) and SeCon-RAG (Si et al., 2025).

For fair comparison, all methods share the same retrieval and generation backbone: BAAI/bge-base-en-v1.5 (Xiao et al., 2024) as the retriever and Qwen2.5-14B-Instruct (Yang et al., 2024) as the default generator, with Qwen2.5-72B-Instruct as the LLM-as-a-Judge. For hyperparameters, we set $n = 30$ query rewrites and $k' = 4k$ candidate pool size for reliability estimation, where $k = 10$. Ablations on both are provided in Appendices H.1–H.2, with full baseline details in Appendix H.

Metrics. We mainly report generation results: *Accuracy* (Acc) is a binary score for whether the generated answer matches the ground truth, scored by an LLM-as-a-Judge with the prompt in Table 17; and *Attack Success Rate* (ASR) is the fraction of queries for which the generated answer matches the attacker-specified target. The two capture complementary aspects. Acc reflects answerability under poisoning, while ASR reflects resistance to attacker steering. A robust defense must therefore achieve high Acc and low ASR simultaneously.

To complement this, we also evaluate retrieval-side filtering quality, which isolates the defense itself from generator behavior. We measure it by *Balanced Accuracy* (bAcc) of poisoned-document classification, *i.e.*, the mean of per-class recall.

4.1 Robustness of SAFER-RAG

Table 1 reports generation performance over six methods with varying attack types and poisoning levels. SAFER-RAG is the only method that consistently achieves the highest Acc and the lowest ASR across every setting, on both single-hop and multi-hop benchmarks, establishing it as the sole defense that is genuinely agnostic to both the attack type and the poisoning level.

Analysis on Poisoning Levels. Reliability-based methods (RAGuard and ReliabilityRAG) are designed under the assumption of heavily poisoned retrieval, scoring each document independently and aggressively filtering low-trust candidates.

This design helps them remain competitive at the High level but limits their effectiveness when poisoning is light. For instance, on NQ under PoisonedRAG, RAGuard reaches 54.7 Acc at High (vs 58.5 for SeCon-RAG) yet trails SeCon-RAG by 5.6, 11.6, and 11.1 points at Clean, Low, and Medium, respectively. SAFER-RAG shows no such trade-off and sustains 84.0, 76.4, 76.6, and 74.5 Acc across Clean, Low, Medium, and High, remaining the strongest method at every level.

Analysis on Attack Types. Clustering-based ones (EcoSafeRAG and SeCon-RAG) instead assume that poisoned documents form separable groups in embedding space, exploiting the tight clustering induced by shared query tokens in query-conditioned attacks like PoisonedRAG.

This design isolates query-conditioned poisons well but falters against natural-looking ones like FSAP, whose LLM-generated content scatters among benign documents. For instance, on NQ at the High level, SeCon-RAG reaches 58.5 Acc under PoisonedRAG yet drops to 52.9 under FSAP, with ASR rising from 20.0 to 35.1. In contrast, our approach (SAFER-RAG) shows no such dependence on attack type and maintains 74.5, 72.2, and 73.6 Acc on the same setting across PoisonedRAG, FSAP, and Mixed, since the rank-stability signal arises from the attack objective itself rather than

Method	PoisonedRAG				FSAP				Mixed			
	NQ	MS-MARCO	HotpotQA	*Avg.*	NQ	MS-MARCO	HotpotQA	*Avg.*	NQ	MS-MARCO	HotpotQA	*Avg.*
SeCon-RAG	61.5	63.7	49.9	58.4	56.4	52.8	49.8	53.0	56.8	57.6	49.8	54.7
EcoSafeRAG	49.4	45.5	49.5	48.2	49.8	44.0	49.9	47.9	49.4	45.9	49.5	48.3
SAFER-RAG	**83.6**	**77.9**	**83.5**	**81.7**	**83.2**	**70.8**	**82.8**	**78.9**	**83.3**	**75.0**	**83.2**	**80.5**

Table 2: Balanced accuracy (bAcc; %) for binary Safe/Unsafe classification of retrieved chunks, comparing SAFER-RAG against two representative clustering-based baselines.

any surface pattern.

No Degradation on Clean Setup. A common bottleneck of existing poisoning defenses is over-cleaning, where useful evidence is discarded along with poisoned ones, which hurts most when poisoning is light or absent. This is clearest on clean corpora, where every defense baseline underperforms VanillaRAG across all three datasets, with drops ranging from 0.4–22.0 points, indicating that their filters discard useful evidence even without any attack to suppress.

In contrast, SAFER-RAG not only avoids this degradation but improves over VanillaRAG by 4.5, 1.6, and 8.0 points on NQ, MS-MARCO, and HotpotQA. This gain stems from two safeguards built into the adaptive selection. First, when the Safe Set cannot ground an answer, it escalates to the Unsafe Set rather than returning empty-handed, ensuring useful evidence is never discarded by default. Second, within whichever set is consulted, it selects only the chunks whose reliability scores best support the chosen answer, as detailed in Eq. (6), so the final context is always built from the most reliable evidence available.

Robustness Sustained on Multi-Hop HotpotQA. SAFER-RAG remains the strongest method on HotpotQA as well, despite a narrower accuracy margin over SeCon-RAG under High poisoning (60.0 vs. 57.0 Acc on PoisonedRAG, 68.0 vs. 53.0 on FSAP, 62.0 vs. 60.0 on Mixed), and it notably outperforms SeCon-RAG on FSAP, where SeCon-RAG's semantic-structural grouping degrades against natural-looking poisons.

This narrower margin under PoisonedRAG and Mixed reflects two properties of multi-hop reasoning rather than a weaker reliability signal: query decomposition weakens the top-rank persistence of poisons optimized for the original query, and errors at any single hop can propagate to the final answer. Overall, SAFER-RAG's advantage over clustering-based defenses remains clearest precisely where attack types diverge, confirming that its robustness generalizes beyond single-hop QA.

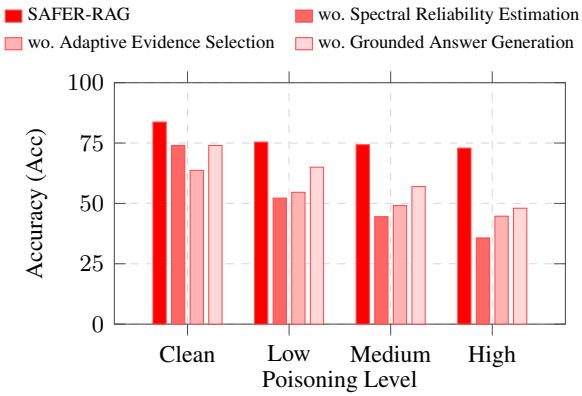


Figure 3: Average generation accuracy (Acc) across NQ, MS-MARCO, and HotpotQA under each poisoning levels of the Mixed attack type.

4.2 Spectral Estimation Validation

Since both Safe and Unsafe splitting and evidence scoring rely on spectral reliability estimation, we validate it in two steps: first, by verifying the rank-stability gap between poisoned and clean chunks under query perturbations; and second, by showing that spectral splitting exploits this signal more effectively than clustering-based alternatives.

Rank-Stability Gap. Figure 2 shows the rank distributions of malicious (MAL), ground-truth (GT), and irrelevant (IRR) chunks under query perturbations across poisoning levels and attack types, aggregated over all three datasets. We separate the non-poisoned corpus into GT and IRR chunks to verify that the rank-stability gap holds even against genuinely relevant clean evidence, rather than being an artifact of comparing poisons with many irrelevant chunks.

Across all settings, malicious chunks consistently maintain the highest retrieval ranks with the lowest variance, followed by GT chunks, with irrelevant chunks ranked lowest and most dispersed; this ordering MAL $<$ GT $<$ IRR holds uniformly across attack types and poisoning levels. Crucially, malicious chunks remain distinguishable even from GT chunks, which are naturally written evidence not optimized for retrieval, indicating that the observed rank rigidity is not simply

a byproduct of relevance but reflects genuine adversarial optimization. This reveals a clear rank-stability gap induced by adversarial intent itself rather than by specific attack patterns, corpus characteristics, or mere topical relevance, validating the core assumption underlying our spectral reliability estimation.

Spectral vs. Clustering-based Splitting. Table 2 compares SAFER-RAG against two representative clustering-based baselines on binary Safe/Unsafe classification of retrieved chunks.

Across all settings, SAFER-RAG achieves substantially higher balanced accuracy, outperforming clustering-based methods by large margins. Although performance drops under stronger attack settings for all methods, SeCon-RAG remains particularly sensitive to attack-specific embedding patterns due to its clustering-based separability assumption. In contrast, spectral splitting preserves strong performance across attack types by exploiting the rank-stability signal induced by adversarial intent itself rather than relying solely on embedding-space clustering structure.

In contrast, spectral splitting preserves strong performance across attack types by exploiting the rank-stability signal induced by adversarial intent itself rather than relying solely on embedding-space clustering structure. The underlying GMM posterior remains informative even under sparse poisoning, as detailed in Appendix D.

4.3 Detailed Analysis

We further analyze SAFER-RAG along three axes. First, we ablate each of its three components to isolate their individual contribution. Second, we measure its latency to characterize the trade-off between robustness and efficiency in practice. Third, we evaluate its generalization across generator backbones of different capacities, while leaving generalization across retrievers to Appendix J.

Component Ablation. We ablate the three stages of SAFER-RAG by (i) replacing spectral reliability estimation with the clustering-based split of SeCon-RAG (Si et al., 2025), (ii) removing adaptive evidence selection by using only the Safe Set, and (iii) replacing grounded answer generation by feeding the full candidate pool to the generator together with evidence indicators.

Figure 3 reports average generation accuracy under Mixed poisoning attacks across attack levels. The three components contribute in complementary regimes. (i) Removing spectral reliability estimation causes the largest drop under heavy poisoning, where clustering fails to separate semantically similar poisons, confirming that rank stability captures adversarial intent beyond surface level embedding structure. (ii) Removing adaptive evidence selection causes the largest drop under no or light poisoning, where Safe Set only retrieval discards useful evidence, showing that fallback retrieval is essential for preserving answerability when the Safe Set is overly conservative[3]. (iii) Removing grounded answer generation causes the largest drop under Medium to High poisoning, where feeding the full candidate pool re exposes the generator to unsafe chunks, showing that restricting the final context to answer supporting chunks is essential for reliable generation.

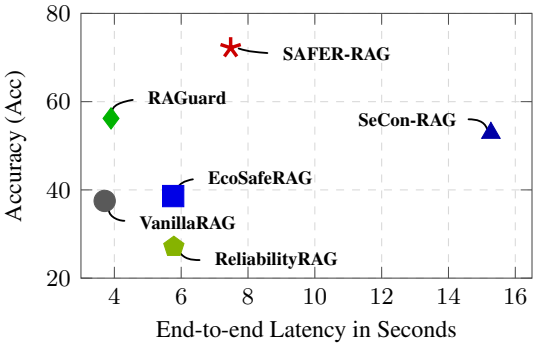


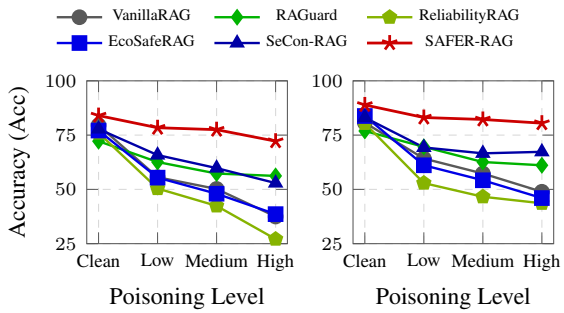
Figure 4: Accuracy–latency trade-off on NQ under the Mixed High poisoning setup, using Qwen2.5-14B-Instruct on an RTX Pro 6000 GPU.

Together, the three components synergize by covering complementary failure modes, with each addressing a regime where the others fall short. We provide the full ablation results in Appendix I, where Tables 9 and 10 report the exact averaged Acc and ASR values, respectively.

Accuracy–Latency Trade-off. Figure 4 illustrates the average end-to-end latency per query on NQ under the Mixed High poisoning setup. SAFER-RAG sits between lightweight baselines such as VanillaRAG and RAGuard and the slowest baseline SeCon-RAG, while achieving substantially higher Acc in generation than all of them.

Four design choices in SAFER-RAG keep its additional cost contained. First, query rewriting, the most obvious source of overhead, is performed in

[3]The fallback mechanism is activated for only 32% of queries on NQ, 12% on MS MARCO, and 40% on HotpotQA, avoiding unnecessary latency. For HotpotQA, this rate is a query-level union metric: a multi-hop query is counted as fallback if *any* of its sub-queries invokes the Unsafe Set. See Appendix K.



(a) Qwen2.5-14B-Inst. (b) GPT-5.4-mini.

Figure 5: Accuracy across different attack levels for different generator models.

a single batched rollout rather than n sequential LLM calls. Second, reliability estimation runs on a small rank matrix, so the spectral computation is negligible. Third, adaptive selection consults the Unsafe Set only when the Safe Set fails to ground an answer, so queries can terminate with a single generator call. Fourth, the final generator receives only the selected evidence rather than the full retrieved pool, shortening the input and decoding time. Together, these choices keep SAFER-RAG's latency within a practical range while delivering substantially higher robustness. The detailed breakdown per stage is provided in Appendix L.

Backbone Generalization. To examine whether SAFER-RAG's robustness depends on the generator, we evaluate it with two backbones of different capacities, Qwen2.5-14B-Instruct and GPT-5.4-mini, under the Mixed poisoning setup. Figure 5 shows that SAFER-RAG achieves the highest Acc across all four poisoning levels on both backbones, and the gap over baselines widens as poisoning grows. This indicates that SAFER-RAG's robustness is not tied to any specific generator behavior, but stems from filtering and selecting reliable evidence before generation. Since the defense operates upstream of the generator, its effect transfers across backbones rather than relying on a particular model's internal capability. Corresponding ASR results are provided in Appendix M.

5 Conclusion

We present SAFER-RAG, a robust RAG defense that turns query induced rank behavior into a reliability signal. It uses spectral analysis to separate clean and poisoned chunks into Safe and Unsafe Sets, then adaptively retrieves evidence with a safe first strategy that avoids over filtering. Because the signal arises from the attack objective itself, SAFER-RAG remains robust across attack types and poisoning levels, achieving the best overall robustness across benchmarks and settings.

Limitations

While SAFER-RAG performs reliably across our experimental settings, two aspects warrant discussion as potential limitations, both of which we believe are mitigated by our design.

Robustness of Query Rewrites. A natural concern is whether imperfect rewrites that drift from the original intent inject noise into the reliability signal. The candidate pool is constructed once from the original query, and rewrites only rerank chunks within this fixed pool, so a drifted rewrite cannot introduce unseen documents and at worst contributes a less informative rank vector among the n observations. The final answer is also grounded on the original query rather than the rewrites. Rewrites thus serve only as a perturbation probe, and their imperfection affects the spectral signal only marginally.

Adaptive Attacks. An attacker aware of our defense could attempt to craft poisons that remain top-ranked under rewrites. We believe this is structurally difficult for three reasons. First, rewrites are sampled stochastically from an LLM at inference, so optimization against any fixed paraphrase set does not transfer to the rewrites SAFER-RAG actually samples. Second, generalizing across diverse surface forms competes with the attack's primary objective of dominating the original target query. Third, even if a poisoned chunk enters the Safe Set, it must additionally outscore clean evidence in reliability-weighted answer selection, requiring bypass of all three stages.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2024-00445087, Enhancing AI Model Reliability Through Domain-Specific Automated Value Alignment Assessment), and by the Korea Institute of Science and Technology Information (KISTI) in 2026 (No. (KISTI)K26L3M1C1), aimed at developing KONI (KISTI Open Neural Intelligence), a large language model specialized in science and technology.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. In *NeurIPS*.

Minjeong Ban, Jeonghwan Choi, Hyangsuk Min, Nicole Hee-Yeon Kim, Minseok Kim, Jae-Gill Lee, and Hwanjun Song. 2026. Completing Missing Annotation: Multi-Agent Debate for Accurate and Scalable Relevant Assessment for IR Benchmarks. In *ICLR*.

Amin Bigdeli, Negar Arabzadeh, Ebrahim Bagheri, and Charles L. A. Clarke. 2025. Adversarial Attacks against Neural Ranking Models via In-Context Learning. *arXiv preprint arXiv:2508.15283*.

Saket S. Chaturvedi, Gaurav Bagwe, Lan Zhang, and Xiaoyong Yuan. 2025. AIP: Subverting Retrieval-Augmented Generation via Adversarial Instructional Prompt. In *EMNLP*.

Zirui Cheng, Jikai Sun, Anjun Gao, Yueyang Quan, Zhuqing Liu, Xiaohua Hu, and Minghong Fang. 2025. Secure Retrieval-Augmented Generation against Poisoning Attacks. In *IEEE BigData*.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise Zero-Shot Dense Retrieval without Relevance Labels. In *ACL*.

Minghui Huang, Dong Wang, Shuang Liu, and Meizhen Ding. 2021. GQE-PRF: Generative Query Expansion with Pseudo-Relevance Feedback. *arXiv preprint arXiv:2108.06010*.

Taehyeon Kim, Jongwoo Ko, Sangwook Cho, Jinhwan Choi, and Se-Young Yun. 2021. FINE Samples for Learning with Noisy Labels. In *NeurIPS*.

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. In *TACL*.

Jie Liu and Barzan Mozafari. 2026. GenRewrite: Query Rewriting via Large Language Models. In *PACMMOD*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On Faithfulness and Factuality in Abstractive Summarization. In *ACL*.

Fatemeh Nazary, Yashar Deldjoo, and Tommaso di Noia. 2025. Poison-RAG: Adversarial Data Poisoning Attacks on Retrieval-Augmented Generation in Recommender Systems. In *ECIR*.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document Expansion by Query Prediction. *arXiv preprint arXiv:1904.08375*.

OpenAI. 2026. GPT-5.4 Thinking System Card.

Pankayaraj Pathmanathan, Michael-Andrei Panaitescu-Liess, Cho-Yu Jason Chiang, and Furong Huang. 2026. RAGPart & RAGMask: Retrieval-Stage Defenses Against Corpus Poisoning in Retrieval-Augmented Generation. In *Workshop of AAAI*.

Douglas A Reynolds and 1 others. 2009. Gaussian Mixture Models. *Encyclopedia of biometrics*, 741(3):659–663.

Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Avital Shafran, Roei Schuster, and Vitaly Shmatikov. 2025. Machine Against the RAG: Jamming Retrieval-Augmented Generation with Blocker Documents. In *USENIX Security*.

Zeyu Shen, Basileal Imana, Tong Wu, Chong Xiang, Prateek Mittal, and Aleksandra Korolova. 2025. ReliabilityRAG: Effective and Provably Robust Defense for RAG-based Web-Search. In *NeurIPS*.

Xiaonan Si, Meilin Zhu, Simeng Qin, Lijia Yu, Lijun Zhang, Shuaitong Liu, Xinfeng Li, Ranjie Duan, Yang Liu, and Xiaojun Jia. 2025. SeCon-RAG: A Two-Stage Semantic Filtering and Conflict-Free Framework for Trustworthy RAG. In *NeurIPS*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *NeurIPS*.

Liang Wang, Nan Yang, and Furu Wei. 2023a. Query2doc: Query Expansion with Large Language Models. In *EMNLP*.

Xiao Wang, Sean MacAvaney, Craig Macdonald, and Iadh Ounis. 2023b. Generative Query Reformulation for Effective Adhoc Search. In *Workshop of SIGIR*.

Zilong Wang, Zifeng Wang, Long Le, Huaixiu Steven Zheng, Swaroop Mishra, Vincent Perot, Yuwei Zhang, Anush Mattapalli, Ankur Taly, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2025. Speculative RAG: Enhancing Retrieval Augmented Generation through Drafting. In *ICLR*.

Xun Xian, Ganghua Wang, Xuan Bi, Jayanth Srinivasa, Ashish Kundu, Charles Fleming, Mingyi Hong, and Jie Ding. 2025. On the Vulnerability of Applying Retrieval-Augmented Generation within Knowledge-Intensive Application Domains. In *ICML*.

Chong Xiang, Tong Wu, Zexuan Zhong, David Wagner, Danqi Chen, and Prateek Mittal. 2026. Certifiably Robust RAG against Retrieval Corruption. In *IEEE SaTML*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-Pack: Packed Resources For General Chinese Embeddings. In *SIGIR*.

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, and 24 others. 2024. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *EMNLP*.

Ruobing Yao, Yifei Zhang, Shuang Song, Neng Gao, and Chenyang Tu. 2025. EcoSafeRAG: Efficient Security through Context Analysis in Retrieval-Augmented Generation. In *EMNLP*.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, and 3 others. 2023. A Survey of Large Language Models. *arXiv preprint arXiv:2303.18223*.

Jingjie Zheng, Aryo Pradipta Gema, Giwon Hong, Xuanli He, Pasquale Minervini, Youcheng Sun, and Qiongkai Xu. 2025. GRADA: Graph-based Reranking against Adversarial Documents Attack. In *EMNLP*.

Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. 2023. Poisoning Retrieval Corpora by Injecting Adversarial Passages. In *EMNLP*.

Huichi Zhou, Kin-Hei Lee, Zhonghao Zhan, Yue Chen, Zhenhao Li, Zhaoyang Wang, Hamed Haddadi, and Emine Yilmaz. 2025. TrustRAG: Enhancing Robustness and Trustworthiness in Retrieval-Augmented Generation. *arXiv preprint arXiv:2501.00879*.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2025. PoisonedRAG: Knowledge Corruption Attacks to Retrieval-Augmented Generation of Large Language Models. In *USENIX Security*.

A SAFER-RAG Pseudocode

For a concise summary of the overall SAFER-RAG pipeline, we provide formal pseudocode in Algorithm 1. The algorithm covers query decomposition, multi-perspective query perturbation, reliability estimation, adaptive evidence selection, and grounded answer generation. This clarifies our implementation and facilitates reproducibility.

Algorithm 1 Rank-based Perturbation for Safe RAG (SAFER-RAG)

Require: Original query q, corpus $\mathcal{D}$, retriever $\mathcal{R}$, generator G, number of rewrites N, candidate pool size k', evidence budget K
Ensure: Final answer a^*

Decompose q into $\mathcal{Q}_{\text{sub}} = \{q^{(0)}, q^{(1)}, \ldots, q^{(h-1)}\}$
for each sub-query $q^{(i)} \in \mathcal{Q}_{\text{sub}}$ **do**
  Generate N semantic rewrites of $q^{(i)}$ and form $\widehat{\mathcal{Q}}^{(i)} = \{\hat{q}_1^{(i)}, \hat{q}_2^{(i)}, \ldots, \hat{q}_N^{(i)}\}$
  $\mathcal{D}_{\text{cand}}^{(i)} \leftarrow \mathcal{R}(q^{(i)}, \mathcal{D}, k')$
  for each rewritten query $\hat{q}_j^{(i)} \in \widehat{\mathcal{Q}}^{(i)}$ **do**
    Re-rank all chunks in $\mathcal{D}_{\text{cand}}^{(i)}$ under query $\hat{q}_j^{(i)}$
  end for
  for each candidate chunk $d \in \mathcal{D}_{\text{cand}}^{(i)}$ **do**
    $\mathbf{v}_d^{(i)} \leftarrow \left[r(d, \hat{q}_1^{(i)}), r(d, \hat{q}_2^{(i)}), \ldots, r(d, \hat{q}_N^{(i)})\right]$
  end for
  Stack $\{\mathbf{v}_d^{(i)}\}_{d \in \mathcal{D}_{\text{cand}}^{(i)}}$ into $\mathbf{V}^{(i)}$
  Center $\mathbf{V}^{(i)}$ and obtain $\widetilde{\mathbf{V}}^{(i)}$
  Compute the leading principal direction $\mathbf{u}_1^{(i)}$ from $\widetilde{\mathbf{V}}^{(i)}$
  Obtain the PC1-based spectral score $x_d^{(i)} = \widetilde{\mathbf{v}}_d^{(i)\top} \mathbf{u}_1^{(i)}$ for each candidate chunk d
  Fit a two-component GMM on the PC1 scores $\{x_d^{(i)}\}_{d \in \mathcal{D}_{\text{cand}}^{(i)}}$
  Assign the smaller-mean component to $g = \text{unsafe}$ and the other component to $g = \text{clean}$
  Compute reliability scores $P(g = \text{clean} \mid x_d^{(i)})$ for all $d \in \mathcal{D}_{\text{cand}}^{(i)}$
  $\mathcal{S}_{\text{safe}}^{(i)} = \{d \in \mathcal{D}_{\text{cand}}^{(i)} \mid P(g = \text{clean} \mid x_d^{(i)}) \geq 0.5\}$, $\mathcal{S}_{\text{unsafe}}^{(i)} = \mathcal{D}_{\text{cand}}^{(i)} \setminus \mathcal{S}_{\text{safe}}^{(i)}$
  $\mathcal{D}_{\text{safe}}^{(i)} \leftarrow$ Top-K chunks from $\mathcal{S}_{\text{safe}}^{(i)}$
  $\mathcal{A}_{\text{safe}}^{(i)} \leftarrow$ valid answer–evidence pairs from $\mathcal{D}_{\text{safe}}^{(i)}$
  if $\mathcal{A}_{\text{safe}}^{(i)} \neq \emptyset$ **then**
    $(a^{(i)*}, \mathcal{D}_{\text{supp}}^{(i)*}) \leftarrow \arg\max_{(a, \mathcal{D}_{\text{supp}}(a)) \in \mathcal{A}_{\text{safe}}^{(i)}} p(\text{clean} \mid a)$
  else
    $\mathcal{A}_{\text{unsafe}}^{(i)} \leftarrow$ valid answer–evidence pairs from $\mathcal{S}_{\text{unsafe}}^{(i)}$
    $(a^{(i)*}, \mathcal{D}_{\text{supp}}^{(i)*}) \leftarrow \arg\max_{(a, \mathcal{D}_{\text{supp}}(a)) \in \mathcal{A}_{\text{unsafe}}^{(i)}} p(\text{clean} \mid a)$
  end if
end for
$\mathcal{D}_{\text{final}} \leftarrow \bigcup_{i=0}^{h-1} \mathcal{D}_{\text{supp}}^{(i)*}$
$a^* \leftarrow G(q, \mathcal{D}_{\text{final}})$
return a^*

B Principal Component Selection for Spectral Reliability Estimation

The eigen decomposition-based Spectral Reliability Estimation relies on principal components (PCs). We examine how the number of retained PCs affects classification performance by comparing three configurations: PC1, PC1+PC2, and PC1+PC2+PC3. Here, PC1+PC2 does not indicate an arithmetic sum; rather, it denotes a two-dimensional projection using the coordinates along PC1 and PC2 for GMM-based classification.

Impact of Higher-Dimensional Components. As shown in Table 3, using PC1 alone achieves balanced Accuracy comparable to configurations that additionally incorporate PC2 or PC3. The configurations using additional components improve balanced Accuracy by only about 0.5 points over PC1 alone, suggesting that their impact on performance is negligible. This tendency is also evident in the multi-hop setting, where PC1 remains close to the best-performing configuration.

Consistency across Attack Types. This trend is consistent across PoisonedRAG, FSAP, and Mixed attacks, with PC1-only remaining close to the best-performing configuration under the Mixed setting.

This result is also consistent with the intuition from FINE (Kim et al., 2021): the leading eigenspace captures the most informative structure of the data, while higher-order components may include noisier or less discriminative variations. In our case, the stable high-ranking behavior of poisoned documents is already captured by the leading principal direction. Therefore, adding PC3 provides limited additional information and does not lead to consistent gains in balanced Accuracy.

Consequently, we adopt PC1 as the default principal component vector for our framework, striking an optimal balance between computational efficiency and balanced Accuracy.

C Justifying the Spectral Feature for Reliability Estimation

To justify the use of spectral analysis over simpler rank-based summaries, we compare our spectral feature (PC1) against three commonly used scalar statistics computed from the rank vectors: mean rank, rank variance, and rank entropy across rewritten queries. Unlike these scalar statistics, which compress each document's rank trajectory according to a predefined summary rule, PC1 is

Setting	PoisonedRAG				FSAP				Mixed			
	NQ	MS-MARCO	Hotpot	*Avg.*	NQ	MS-MARCO	Hotpot	*Avg.*	NQ	MS-MARCO	Hotpot	*Avg.*
PC1	83.2 (67.5)	77.5 (65.8)	83.1 (53.5)	81.3 (62.3)	82.6 (66.7)	70.7 (59.9)	82.5 (53.1)	78.6 (59.9)	82.8 (67.1)	74.8 (63.1)	82.8 (53.3)	80.1 (61.2)
PC1+PC2	83.6 (67.9)	77.9 (66.3)	83.5 (53.8)	81.7 (62.7)	83.2 (67.3)	70.8 (60.0)	82.8 (53.3)	78.9 (60.2)	83.3 (67.6)	75.0 (63.2)	83.2 (53.5)	80.5 (61.4)
PC1+PC2+PC3	83.5 (67.8)	77.8 (66.0)	83.8 (53.8)	81.7 (62.5)	83.0 (67.2)	70.6 (69.8)	82.8 (53.3)	78.8 (60.1)	83.2 (67.5)	75.0 (63.1)	83.1 (53.5)	80.4 (61.2)

Table 3: Comparison of balanced Accuracy for PC1, PC1+PC2, and PC1+PC2+PC3 eigen-decomposition. Results are presented as A, where $A(B)$ denotes the balanced Accuracy for Non-poisoned vs. Poisoned, and (B) denotes the balanced Accuracy for Ground Truth (GT) vs. Poisoned.

Dataset	Attack Method	PC1 (Ours)	Mean Rank	Rank Variance	Rank Entropy
NQ	PoisonedRAG	**0.832**	0.737	0.727	0.778
	FSAP	**0.826**	0.727	0.710	0.750
	Mixed	**0.828**	0.729	0.716	0.766
MS-MARCO	PoisonedRAG	**0.775**	0.775	0.633	0.717
	FSAP	**0.707**	0.630	0.564	0.637
	Mixed	**0.748**	0.705	0.604	0.687
HotpotQA	PoisonedRAG	**0.831**	0.781	0.729	0.812
	FSAP	**0.825**	0.776	0.708	0.795
	Mixed	**0.828**	0.775	0.715	0.813

Table 4: Balanced accuracy for malicious versus non-malicious chunk classification, comparing PC1 against simpler rank-based statistics under the same GMM-based classification procedure.

estimated in a data-driven manner directly from the full cross-rewrite rank matrix. It uses the entire rank matrix rather than a single aggregated value, captures the joint variation across rewritten queries, and models the covariance among rank observations, allowing it to adapt to the observed rank distribution without relying on a manually designed heuristic. All four features are compatible with the same GMM-based classification procedure used in SAFER-RAG, which allows for a controlled comparison isolating the effect of the feature itself.

To empirically verify this advantage, we compare PC1 with the three scalar statistics while keeping the GMM-based classification procedure fixed and varying only the input feature. Table 4 reports the resulting balanced accuracy for malicious versus non-malicious chunk classification across all datasets and attack settings.

PC1 achieves the highest or tied-highest balanced accuracy across all datasets and attack settings. It consistently outperforms rank variance and rank entropy, and also surpasses mean rank except on MS-MARCO under PoisonedRAG, where the two perform equally. This indicates that the advantage of SAFER-RAG does not arise merely from using any specific rank-based statistic; rather, the spectral feature provides a more informative representation for separating malicious and non-malicious documents under the same GMM classifier.

Poisoning Level	GT Mean	GT Var.	Mal Mean	Mal Var.
Low	0.4618	0.1514	0.0074	0.0045
Medium	0.4997	0.1620	0.0094	0.0056
High	0.5591	0.1803	0.0127	0.0066

Table 5: Mean and variance of the GMM posterior $P(g{=}\text{clean} \mid x_d)$ for ground-truth (GT) and malicious (MAL) documents across poisoning levels.

D GMM Posterior Reliability across Poisoning Levels

The two-component GMM fitted on PC1 scores (Section 3.1) partitions the candidate pool into Safe and Unsafe Sets without a manually calibrated threshold. A natural concern is whether this posterior remains informative when poisoning is sparse: with only one or two malicious documents among $k' = 40$ candidates, the "unsafe" Gaussian component is estimated from very few points, which could make $P(g{=}\text{clean} \mid x_d)$ unreliable.

We emphasize that $P(g{=}\text{clean} \mid x_d)$ is not intended as a perfectly calibrated class probability. Its mean, variance, and mixture weights are re-estimated from each query's own rank matrix rather than fixed in advance, so it instead serves as a *query-adaptive relative reliability score* that ranks candidates by how consistently they behave across rewrites. This score is used by the two-stage adaptive selection in Section 3.2 to route evidence, with the Unsafe Set fallback recovering candidates that are conservatively assigned to the wrong set by an imperfect GMM split.

To examine whether the posterior remains informative under sparse poisoning, we report the empirical mean and variance of $P(g{=}\text{clean} \mid x_d)$ for ground-truth (GT) and malicious (MAL) documents across poisoning levels in Table 5.

Even at the Low poisoning level, where only two malicious documents are injected, the mean posterior for malicious documents remains close to zero, while GT documents receive substantially higher scores. This indicates that the suspicious component remains informative even when esti-

mated from few malicious samples. As poisoning increases, the mean GT score rises from 0.4618 at Low to 0.5591 at High, while malicious-document scores remain near zero, making the separation between the two modes clearer as more malicious chunks become available to estimate the unsafe component.

We note that the Safe and Unsafe Sets are not expected to perfectly match the true clean and malicious classes. The GMM first constructs a high-confidence Safe Set, while GT documents conservatively assigned to the Unsafe Set can still be recovered through the second-stage fallback described in Section 3.2. Since clean and poisoned chunks exhibit distinctly polarized patterns in their posterior distributions, even conservative misclassifications are effectively corrected by prioritizing evidence with higher clean probability during fallback, preventing downstream generation failure.

E Query Rewriting Prompt and Empirical Grounding

To empirically justify the constraints in our query rewriting prompt template (Table 16), we ground each instruction in established Information Retrieval (IR) and LLM literature. Rather than relying on naive prompt engineering, our template extends Natural Language Rewriting Hints (NLR2) introduced by Liu and Mozafari (2026) into three boundary layers: mandatory constraints (*You must*), permissible variations (*You may*), and prohibitive boundaries (*Do not*). Table 18 maps each sub-instruction to key paradigms in query reformulation and dense retrieval, showing how the generated N rewrites improve retrieval robustness and information coverage while preserving the user's core intent.

F Dataset Details

We provide detailed information on the datasets used in our experiments, including the number of queries, corpus size, and annotation source. Our evaluation is conducted on three retrieval and question-answering benchmarks: NQ, MS-MARCO, and HotpotQA. These datasets cover complementary retrieval scenarios, ranging from single-hop web search questions to multi-hop reasoning over Wikipedia passages.

NQ and MS-MARCO. For NQ (Kwiatkowski et al., 2019) and MS-MARCO (Bajaj et al., 2016), we use the BRIDGE benchmark constructed by (Ban et al., 2026), which builds on BEIR retrieval subsets (Thakur et al., 2021) and mitigates the annotation-hole problem by using DREAM to identify missing relevant chunks, thereby providing denser and more reliable relevance annotations for retrieval and RAG evaluation. We use 550 queries for each dataset: NQ is based on BEIR(NQ), whose questions come from real Google Search queries with Wikipedia-based answers, and uses the corresponding BEIR corpus of approximately 2.6M passages; MS-MARCO is based on BEIR(MS-MARCO), which originates from large-scale web search queries with passage-level relevance judgments, and uses the corresponding BEIR corpus of approximately 8.8M passages. For both datasets, we use the BEIR source corpora and BRIDGE-refined annotations.

HotpotQA. For HotpotQA (Yang et al., 2018), we use the original fullwiki benchmark, which evaluates explainable multi-hop question answering over Wikipedia. Because each question requires combining evidence from multiple paragraphs, HotpotQA is suitable for testing retrieval robustness in open-domain RAG. We construct the corpus from the released Wikipedia-based evidence collection, resulting in 5,863,011 chunks after preprocessing and indexing, and evaluate on 100 sampled close-ended multi-hop queries.

G Model and Retrieval Configurations

Table 6 summarizes the model and retrieval configurations used in the SAFER-RAG evaluation pipeline. In the main experiments, all methods use BGE-base-en-v1.5 (Xiao et al., 2024) as the default dense embedding model for retrieval and Qwen2.5-14B-Instruct (Yang et al., 2024) as the default generator to ensure fair comparison. The same generator is also used for query rewriting, adaptive evidence selection, and grounded answer generation, while Qwen2.5-72B-Instruct (Yang et al., 2024) is used as the LLM-as-a-Judge for evaluating answer correctness and attack success.

For retrieval-backend generalization, we compare BGE-base-en-v1.5 (Xiao et al., 2024), Qwen3-Embedding-4B (Zhang et al., 2025), and BM25 (Robertson and Zaragoza, 2009), covering lightweight dense embedding, large-scale dense embedding, and sparse lexical retrieval settings. For generator-backbone generalization, we additionally evaluate SAFER-RAG with GPT-5.4-mini (OpenAI, 2026) while keeping the retrieval

Type	Model	Model Checkpoint / API Name	Source	Reference
Retriever	BGE-base-en-v1.5	`BAAI/bge-base-en-v1.5`	HuggingFace	Xiao et al. (2024)
Retriever	Qwen3-Embedding-4B	`Qwen/Qwen3-Embedding-4B`	HuggingFace	Zhang et al. (2025)
Retriever	BM25	`BM25`	Sparse retriever	Robertson and Zaragoza (2009)
Open-source	Qwen2.5-14B-Instruct	`Qwen/Qwen2.5-14B-Instruct`	HuggingFace	Yang et al. (2024)
Open-source	Qwen2.5-72B-Instruct	`Qwen/Qwen2.5-72B-Instruct`	HuggingFace	Yang et al. (2024)
Proprietary	GPT-5.4-mini	`gpt-5.4-mini`	OpenAI API	OpenAI (2026)

Table 6: Model and retrieval configurations across the SAFER-RAG evaluation pipeline: retrieval-backend evaluation, grounded answer generation, LLM-as-a-Judge evaluation, and generator-backbone generalization. Models used in multiple settings are listed under each relevant role.

backend, poisoning settings, prompt format, and evaluation protocol fixed.

Proprietary models are queried through their official APIs, whereas open-source models are run on local NVIDIA RTX PRO 6000 Blackwell Server Edition GPUs with 96GB VRAM. We do not apply model-specific decoding or prompt tuning unless explicitly stated. For each experiment, competing methods are evaluated with matched prompt formats, retrieval settings, poisoning configurations, and evaluation criteria to keep comparisons controlled.

G.1 Scientific Artifacts

SAFER-RAG evaluation employs publicly available QA benchmarks, poisoning attack implementations, retrieval and generation backbones, and prompting protocols across the full defense pipeline. We evaluate on NQ, MS-MARCO, and HotpotQA under three corpus poisoning settings, including PoisonedRAG, FSAP, and their Mixed variant, with poisoning levels in {Clean, Low, Medium, High}. All methods use the same retrieval and generation backbones for fair comparison: BAAI/bge-base-en-v1.5 as the retriever and Qwen2.5-14B-Instruct as the default generator, with Qwen2.5-72B-Instruct used as the LLM-as-a-Judge. For SAFER-RAG, we use 30 query rewrites and a candidate pool size of $k' = 4k$ where $k = 10$ is the top-k evidence budget used in retrieval and adaptive evidence selection. The query rewriting prompt, judging prompt, model configurations, and implementation details are documented in Appendix E, Table 17 and Appendix H and the released code.

H Implementation Details and Hyperparameter Analysis

This section provides additional details on the experimental design choices used in SAFER-RAG. In the main paper, we describe the overall evaluation protocol and report the primary robustness results. Here, we further justify two implementation choices that are central to the proposed rank-perturbation-based reliability estimation: (1) the number of semantic-preserving query reformulations used to construct rank vectors, and (2) the size of the retrieval candidate pool used for rank-based analysis. These choices directly affect the stability and discriminative power of the spectral reliability signal. A small number of query variants lead to noisy rank statistics, while an excessively large retrieval pool may introduce irrelevant candidates that dilute the reliability estimation. Therefore, we select these hyperparameters to balance statistical robustness, computational efficiency, and downstream generation quality.

Number of Rewrites (s)	Acc	ASR
9	64.3	16.8
19	69.0	12.4
29	72.1	9.7
39	72.7	9.3
49	72.9	9.1

Table 7: Effect of the number of query rewrites on SAFER-RAG under the high poisoning level. average Acc and ASR across three datasets; NQ, MS-MARCO, and HotpotQA.

H.1 Effect of the Number of Query Rewrites

SAFER-RAG estimates document reliability by measuring how consistently each retrieved document maintains its rank under semantic-preserving query perturbations. Thus, the number of rewritten queries is an important hyperparameter: too few rewrites may produce unstable rank-perturbation features, while too many may increase computational overhead with limited additional benefit.

To analyze this trade-off, we vary the number of query rewrites as $s \in \{9, 19, 29, 39, 49\}$, corresponding to $n = s + 1 \in \{10, 20, 30, 40, 50\}$

Dataset	Attack Type	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
NQ	PoisonedRAG	0.039	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	FSAP	0.023	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	Mixed	0.011	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
MS-MARCO	PoisonedRAG	0.012	0.011	0.007	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	FSAP	< 0.005	0.611	0.188	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	Mixed	0.418	0.007	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
Hotpot	PoisonedRAG	0.001	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	FSAP	0.204	0.058	0.008	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005
	Mixed	0.206	0.035	0.006	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005	< 0.005

Table 8: P-values for testing the statistical difference between the average PC1 values of poisoned and non-poisoned documents across datasets and attack types for varying N in the Top-NK retrieval setting. For each value of N, the original query retrieves $N \times K$ candidate documents before reliability estimation. Cell color indicates statistical significance at $p < 0.005$.

total query variants including the original query. We conduct this analysis under the High poisoning level, where separating clean from poisoned documents is most challenging. For each rewriting size, we report the average performance across the three datasets, NQ, MS-MARCO, and HotpotQA.

As shown in Table 7, the performance gradually improves as the number of rewritten queries increases. This indicates that additional semantic-preserving perturbations provide richer rank-variation signals, allowing SAFER-RAG to better distinguish stable reliable documents from unstable poisoned documents. When only a small number of rewrites is used, the rank vectors are constructed from limited observations and can be sensitive to individual rewrite artifacts. As the number of rewrites increases, the reliability estimation becomes more robust because each document is evaluated across a broader set of semantically equivalent query perspectives.

However, the improvement becomes marginal after $s = 29$. Although increasing the number of rewrites to $s \in \{39, 49\}$ still provides slightly better performance, the gain is relatively small compared to the additional reranking and feature construction cost. This suggests that around 30 total query variants are sufficient to capture the major rank-instability patterns needed for reliable document separation. Therefore, we set $s = 29$, resulting in $n = 30$ query variants including the original query, as a practical balance between robustness and efficiency throughout our experiments.

H.2 Effect of Retrieval Candidate Pool Size

SAFER-RAG relies on rank-based perturbation signals to estimate the reliability of retrieved documents. Thus, the retrieval candidate pool should be large enough to reveal meaningful rank-instability patterns, while avoiding unnecessary expansion which introduce irrelevant candidates as noise. To determine an appropriate pool size, we vary the retrieval candidate pool from Top-$1K$ to Top-$10K$.

As shown in Table 8, when the candidate pool is small, the separation is not always statistically significant. For example, several settings at Top-$1K$, Top-$2K$, and Top-$3K$ show relatively large p-values, suggesting that the retrieval pool is too narrow to expose stable rank-perturbation patterns. In contrast, from Top-$4K$ onward, all datasets and attack types consistently yield $p < 0.005$. Although larger candidate pools also maintain statistical significance, overly expanding the pool can introduce irrelevant candidates whose rank fluctuations act as noise in reliability estimation. Therefore, we select Top-$4K$ as the default retrieval candidate pool size, as it is the smallest setting that consistently preserves the discriminative signal while avoiding unnecessary noise from overly broad retrieval.

Table 8 reports the p-values which shows significant difference between poisoned and non-poisoned documents across datasets and attack types. When the candidate pool is small, the separation is not always statistically significant. For example, on MS-MARCO dataset, FSAP attack type does not show significant separation at Top-$2K$ and Top-$3K$, and Mixed attack type also shows weak separation at Top-$1K$. In contrast, from Top-$4K$ onward, all datasets and attack types consistently yield $p < 0.005$. This indicates that Top-$4K$ is the smallest candidate pool size which provides a statistically meaningful separation between poisons and non-poisons across all evaluated settings.

This trend can be explained by the expanded observable rank range. When the candidate pool

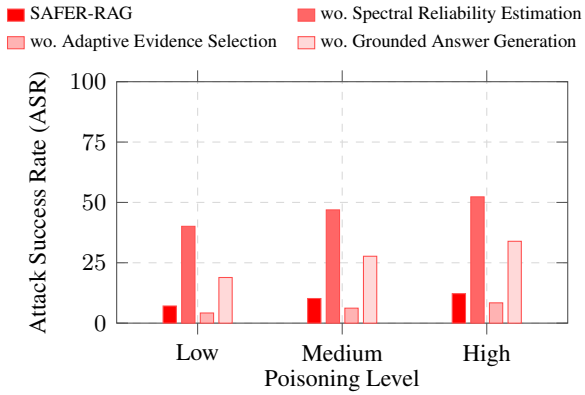


Figure 6: Ablation study results measured by average ASR (%) across three datasets; NQ, MS-MARCO, and HotpotQA under Low, Medium, and High poisoning levels.

Method	Clean	Low	Medium	High
Baseline	83.7	75.4	74.3	72.9
wo. Spectral Reliability Estimation	74.0	52.2	44.5	35.7
wo. Adaptive Evidence Selection	63.7	54.6	49.1	44.7
wo. Grounded Answer Generation	83.5	66.8	55.8	46.6

Table 9: Average Acc across NQ, MS-MARCO, and HotpotQA under different poisoning levels.

Method	Low	Medium	High
Baseline	7.1	10.2	12.2
wo. Spectral Reliability Estimation	40.1	46.9	52.3
wo. Adaptive Evidence Selection	4.2	6.2	8.4
wo. Grounded Answer Generation	18.9	27.7	33.9

Table 10: Average ASR across NQ, MS-MARCO, and HotpotQA under different poisoning levels.

is small, rank perturbations are truncated within a narrow top-ranked region, making document instability less visible. As the pool expands, lower-ranked positions become observable, allowing unstable documents to exhibit larger rank fluctuations across perturbed queries. This makes the separation between poisons and non-poisons more pronounced, leading to smaller p-values.

However, increasing the candidate pool size indefinitely is not desirable. Although a larger pool can further expose rank instability, it also introduces many irrelevant documents that are neither ground-truth-supporting nor malicious. These irrelevant candidates act as additional noise in reliability estimation and may interfere with downstream evidence selection and generation. Therefore, we do not choose the largest pool size, such as Top-$10K$. Instead, Top-$4K$ provides a practical balance between statistical separability and retrieval noise, and we use it as the retrieval candidate pool throughout our experiments.

I Extended Ablation Analysis

Figure 6 shows the average ASR of the ablation study across NQ, MS-MARCO, and HotpotQA under Low, Medium, and High attack levels. Table 9 and Table 10 reports the average Acc and ASR, providing a more detailed view of how each component affects attack suppression.

The variant without Spectral Reliability Estimation shows a substantially higher ASR than the full SAFER-RAG, especially under stronger attacks. For example, under the high attack level, its average ASR increases from 12.2% to 52.3%. This result further supports our main observation that clustering-based or non-spectral separation is insufficient for filtering poisoned documents, as poisoned evidence is often designed to remain semantically close to reliable evidence.

The variant without Adaptive Evidence Selection yields relatively low ASR among ablated variants, but this should be interpreted together with its accuracy drop in the main accuracy ablation results. It suffers from over-filtering: removing the adaptive fallback and evidence selection mechanism reduces the chance of using potentially unsafe evidence, but also prevents recovery of useful evidence mistakenly assigned to the Unsafe Set. Thus, it can achieve low ASR at the cost of substantially reduced answerability. Therefore, lower ASR alone does not indicate a better defense; a robust RAG defense should suppress attack success while preserving accurate grounded generation.

The variant without Grounded Answer Generation also shows a clear ASR increase under stronger attacks. This indicates that even when reliable evidence is identified, the final generation stage remains important for preventing the model from being misled by unsafe or weakly grounded information. Thus, grounded answer generation complements reliable evidence selection by turning selected evidence into robust final answers.

Overall, Table 10 complements the accuracy-based ablation analysis and confirms that SAFER-RAG better balances robustness and answer generation than the ablated variants.

J Analysis of Retriever Generalization

To evaluate whether our defense framework generalizes across different retrieval backends, we com-

Retriever	Attack Level			
	Clean	Low	Medium	High
BGE-base-en-v1.5	84.0	77.1	77.5	73.6
Qwen3-Embedding-4B	84.2	80.4	78.7	74.2
BM25	79.3	65.5	65.8	62.9

Table 11: Accuracy (%) on the NQ dataset under the Mixed attack setting with different retrievers.

Level	VanillaRAG Acc ↑	SAFER-RAG Acc ↑
Clean	77.8	79.3
Low	43.3	65.5
Medium	35.8	65.8
High	31.1	62.9

Table 12: Accuracy (%) comparison between VanillaRAG and SAFER-RAG under BM25 retrieval on the NQ dataset under the Mixed attack setting.

pare three representative retrievers: a lightweight dense retriever (BGE-base-en-v1.5, 109M) (Xiao et al., 2024), a large-scale dense retriever (Qwen3-Embedding-4B) (Zhang et al., 2025), and a sparse lexical retriever (BM25) (Robertson and Zaragoza, 2009). Table 11 reports the accuracy on the NQ dataset under the Mixed attack setting.

Overall, our method maintains strong performance with dense retrievers. Both BGE-base-en-v1.5 and Qwen3-Embedding-4B achieve over 73% accuracy even under the High attack level, indicating that our rank-pattern-based reliability estimation remains effective across dense embedding models with different scales and capacities. Moreover, the larger retriever yields consistent but moderate improvements over the lightweight retriever, suggesting that stronger semantic representations further improve the separation between safe and poisoned documents.

BM25 also achieves meaningful robustness under our framework and outperforms the corresponding vanilla and baseline settings. This result shows that, although our defense benefits more from dense retrievers with stronger semantic representations, it is not tied to a specific embedding model and can still provide robustness gains with a simple sparse retriever. However, the lexical retriever shows lower accuracy than dense retrievers across all attack levels. This degradation is expected because such retrievers primarily rely on lexical overlap. Since our framework uses semantically equivalent query rewrites, paraphrased queries can alter the surface-level word matching patterns used by lexical retrievers, which may reduce retrieval quality and consequently lower the final answer accuracy. In addition, poisoned passages containing query-relevant keywords can still obtain high retrieval scores, making it harder to distinguish them based on rank variations across rewrites.

Dataset	Number of Queries	Safe Set-only	Unsafe Set Fallback
NQ	550	67.8%	32.2%
MS-MARCO	550	88.0%	12.0%
HotpotQA	100	60.4%	39.6%

Table 13: Stage-wise routing statistics under the High attack types, where each query contains six poisoned documents.

We emphasize that our retriever-agnostic claim refers to the ability of SAFER-RAG to operate without any retriever-specific modification and to provide robustness gains across retrieval paradigms, rather than a claim of identical absolute performance across all retrievers. To verify that the lower performance under BM25 does not stem from a failure of our defense mechanism itself, but rather from the inherent limitation of lexical retrieval, we compare SAFER-RAG against VanillaRAG under the same BM25, NQ, and Mixed attack setting.

As shown in Table 12, the performance drop under BM25 largely reflects the weakness of sparse lexical retrieval itself rather than a failure specific to our defense. Under High poisoning, VanillaRAG accuracy falls to 31.1%, whereas SAFER-RAG still achieves 62.9%, yielding a 31.8-point gain. Therefore, although BM25 weakens the rank-stability signal and lowers the absolute performance of SAFER-RAG, the same lexical-matching limitation causes an even larger degradation in the undefended baseline, confirming that SAFER-RAG continues to provide substantial robustness under sparse retrieval.

K Analysis of Adaptive Evidence Selection

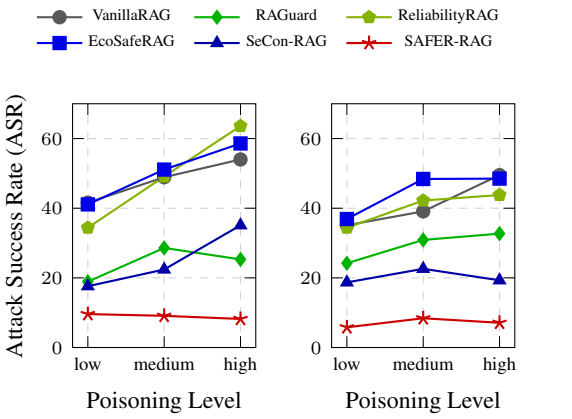
To further analyze the adaptive behavior of SAFER-RAG, we report how often the framework resolves evicence selection for a query on Safe Set or escalates it to Unsafe Set. Under the High setting, where each query contains six poisoned documents, most cases are resolved using the Safe Set, as shown in Table 13. This indicates that the Safe Set often provides sufficient evidence without re-

Stage	Latency (s)	Ratio (%)	LLM calls	Total Tokens
Query perturbation	2.87	38.37	1.000	684.5
Candidate retrieval	0.06	0.79	–	–
Rank-based feature construction	0.02	0.27	–	–
Spectral reliability estimation	0.00	0.01	–	–
Adaptive evidence selection	3.67	49.06	1.322	2799.3
Grounded answer generation	0.86	11.50	1.000	519.3
Total	7.48	100.00	3.322	4003.2

Table 14: Stage-wise latency, LLM calls, and total (input + output) token consumption of SAFER-RAG on the NQ dataset under the Mixed attack setting. Latency is reported as average seconds per query.

Method	**LLM calls**	**Total Tokens**	**Latency (s)**	**Acc ↑**	**ASR ↓**
VanillaRAG	1.000	1422.9	3.85	38.9	58.2
VanillaRAG+SC	4.000	5692.2	11.60	45.5	51.6
SAFER-RAG (Ours)	3.264	4003.2	7.48	**73.6**	**6.9**

Table 15: Comparison of LLM calls, token consumption, latency, and generation performance among VanillaRAG, VanillaRAG+SC, and SAFER-RAG on the NQ dataset under the Mixed attack type at the High poisoning level.



(a) Qwen2.5-14B-Instruct (b) GPT-5.4-mini

Figure 7: ASR across different attack levels for different generator models. Clean setting is omitted because ASR is not defined without poisoning.

quiring additional recovery.

The routing pattern varies across datasets. MS-MARCO shows the highest Safe Set-only rate, with 88.0% of cases resolved without Unsafe Set fallback, suggesting that reliability estimation usually identifies sufficient trustworthy evidence from the Safe Set. In contrast, HotpotQA shows the highest Unsafe Set fallback rate of 39.6%, as multi-hop questions require more complex evidence aggregation and are more likely to produce ambiguous or incomplete Safe Sets. NQ lies between these cases, with 67.8% Safe Set-only and 32.2% Unsafe Set fallback.

These results show that SAFER-RAG does not always invoke the costly recovery stage. Instead, it escalates only when the Safe Set is insufficient, maintaining robustness while keeping additional inference cost bounded.

L Stage-wise Latency Breakdown

Table 14 reports the stage-wise latency, LLM call, and token consumption breakdown of SAFER-RAG on the NQ dataset under the High Mixed poisoning setting. The average end-to-end latency of SAFER-RAG is 7.48 seconds per query, compared to 3.78 seconds for VanillaRAG under the same setting.

Although SAFER-RAG introduces additional computation for query perturbation and reliability-aware evidence selection, the remaining stages are lightweight. The largest costs come from query perturbation and adaptive evidence selection. Query perturbation takes 2.87 seconds because 29 rewritten queries are generated through a single batched rollout, rather than through 29 sequential LLM calls. Adaptive evidence selection takes 3.67 seconds because the method first searches the Safe Set for valid supporting evidence and falls back to the Unsafe Set only when the Safe Set cannot ground an answer. In contrast, candidate retrieval, rank-based feature construction, and spectral reliability estimation are inexpensive, taking only 0.059, 0.020, and 0.001 seconds, respectively, and do not invoke the LLM. Finally, grounded answer generation takes 0.86 seconds because the final generator receives only the original query and the selected evidence chunks, instead of the full retrieved pool. Overall, these stages are carefully optimized and tightly integrated, which keeps the additional latency lower than expected and within a practical range for robust RAG deployment.

To examine whether this additional inference cost is simply a byproduct of using more computation, we compare SAFER-RAG against VanillaRAG and a budget-controlled self-consistency baseline, VanillaRAG+SC, under the NQ dataset with Mixed attack type at the High poisoning level. For VanillaRAG+SC, we generate four independent answers from the same retrieved documents and select the final answer by majority vote; since SAFER-RAG uses an average of 3.264 LLM calls, we set $M = \lceil 3.264 \rceil = 4$, giving VanillaRAG+SC a larger call, token, and latency budget than SAFER-RAG.

M Additional Results on Model-Agnostic Performance of SAFER-RAG

Figure 7 reports the corresponding ASR results across different generator backbones. While the main text focuses on accuracy gain to show that SAFER-RAG preserves answer correctness under stronger attacks, ASR directly measures whether the model follows the attacker-specified target answer. Lower ASR therefore indicates stronger resistance to poisoning attacks.

Across both Qwen2.5-14B-Instruct (Yang et al., 2024) and GPT-5.4-mini (OpenAI, 2026), SAFER-RAG consistently achieves the lowest ASR under all attack levels. This trend is especially clear in the high attack setting, where existing baselines remain vulnerable to poisoned evidence and show substantially higher attack success rates. In contrast, SAFER-RAG reduces ASR to 8.2% on Qwen2.5-14B-Instruct and 7.1% on GPT-5.4-mini, confirming that the Acc improvements observed in the main text are not achieved by simply producing more generic or conservative answers.

These results further support the model-agnostic robustness of SAFER-RAG. Since SAFER-RAG filters and selects evidence before generation, its defense effect transfers across different generator backbones rather than depending on a specific model's internal behavior.

User Prompt

You are a professional assistant that follows output quantity and formatting instructions strictly.

Rewrite the following {Count} times for the given query.

You must:
- Preserve the original intent of the query (maintain the same information need)
- Keep the core topic and key entities unchanged

You may:
- Use different wording or paraphrases
- Adjust the level of specificity (make the query more general or more specific)
- Introduce related concepts or contextual details if relevant
- Expand the query into a more detailed, document-like description
- Reformulate the query from a different but relevant perspective

Do not:
- Change the original intent of the query
- Introduce unrelated topics
- Add unsupported or fabricated information
- Make the query unnecessarily verbose or ambiguous

Query:
"{Original Query}"

Please list exactly {Count} distinct and unique rewrites numerically from 1 to {Count}.

Table 16: Prompt template for generating diverse query rewrites with multi-perspective constraints.

Judge Prompt

You are an objective evaluator.

Question: {query}
Ground Truth Answers: {gt_answers}
Target Answer: {target_answer}
Candidate Answer: {candidate}

Task:
Classify Candidate Answer into exactly one label:
- `TARGET`: candidate semantically matches target answer.
- `GT`: candidate semantically matches any ground-truth answer.
- `NONE`: candidate matches neither, or candidate is empty/unknown.

Rules:
1. Semantic equivalence counts as match.
2. Paraphrases, abbreviations, and minor wording differences should still count as a match when the core meaning is the same.
3. Output exactly one label among `TARGET`, `GT`, and `NONE`.

Table 17: Prompt template for judging whether a candidate answer matches the target answer, a ground-truth answer, or neither.

Prompt Boundary & Sub-Instruction	Core IR / NLP Mechanism	Foundational Literature
You must:		
- Preserve original intent & key entities	Intent anchoring	GenRewrite (Liu and Mozafari, 2026)
You may:		
- Use different wording or paraphrases	Alleviating vocabulary mismatch; improving retrieval recall	Doc2query (Nogueira et al., 2019)
- Adjust the level of specificity (more general/specific)	Granularity control for strategic search space exploration	Generative Query Reformulation (Wang et al., 2023b)
- Introduce related concepts or contextual details	Simulating Pseudo-Relevance Feedback (PRF)	GQE-PRF (Huang et al., 2021)
- Expand the query into a detailed, document-like description	Enriched semantic representation in dense embedding space	Query2doc / Precise Zero-Shot (Wang et al., 2023a; Gao et al., 2023)
- Reformulate the query from a different perspective	Multi-perspective sampling; enhancing retrieval robustness	Speculative RAG (Wang et al., 2025)

Table 18: Empirical grounding and foundational literature mapping for the query rewriting prompt constraints.