RAGuard: A Layered Defense Framework for Retrieval-Augmented Generation Systems Against Data Poisoning

Tanish Kolhe* **Pushkal Kumar*** **Tucker Nielson*** **Shubham Zala***
Vincent Li[1,2,†] **Michael Saxon**[1,3,‡] **Sean Wu**[1,4,‡] **Kevin Zhu**[1]
kevin@algoverseacademy.com

Abstract

Retrieval-Augmented Generation (RAG) systems are becoming more common in augmenting large language models (LLMs) with factual knowledge, yet they remain highly vulnerable to data poisoning, i.e., maliciously injected passages that manipulate retrieved evidence. We introduce RAGuard, a layered two-step defense framework that combines retrieval-level adversarial training with a novel zero-knowledge inference patch. The first step fine-tunes dense retrievers (e.g., Contriever, compatible with BGE and others) using synthetic poisoned documents (composed of poisons such as fabricated facts, contradictions, and reasoning traps), training them to downrank malicious passages. The second step applies a black-box approach zero knowledge inference patch that identifies and filters suspicious documents based on their causal influence on QA correctness, without requiring poison labels. Experiments on Natural Questions (NQ) and Benchmarking-IR (BEIR) show that RAGuard improves robustness by reducing the Attack Success Rate (ASR) while maintaining retrieval quality (Recall@5, MRR). Together, these layers offer an efficient and label-free defense against both known and unseen poisoning attacks, establishing a general framework for resilient, self-healing RAG pipelines.

1 Introduction

Retrieval-Augmented Generation (RAG) has emerged as an effective method to ground Large Language Models (LLMs), retrieving important information to improve and allowing LLMs to use data that was not in their training data. By using up-to-date and diverse data from external corpora, RAG systems allow LLMs to give responses that are up-to-date, and by grounding the LLM's response in factual data, RAG systems reduce the tendency of LLMs hallucinating false or outdated facts [Lewis et al., 2021, Asai et al., 2023]. This retrieval-augmented approach has shown significant improvements across knowledge-intensive tasks, including open-domain question answering and fact verification. In such situations, grounding generation in factual documents is necessary for maintaining accuracy and reliability [Ram et al., 2023, Izacard et al., 2022a].

However, due to the reliance on external data sources, RAG systems can be exposed to vulnerabilities, and one such vulnerability is data poisoning. Data poisoning allows for malicious documents to be inserted into the retrieval corpus [Zou et al., 2024, Long et al., 2025, Su et al., 2024]. Attackers who

[*] Equal contribution.
[†] Senior author.
[‡] Equal advising.
[1] Algoverse AI Research, [2] Boston University, [3] University of Washington, [4] University of Oxford.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Socially Responsible and Trustworthy Foundation Models (ResponsibleFM).

use data poisoning create documents that mimic relevant content yet contain false and misleading information, leading to an LLM returning incorrect answers [Zou et al., 2024, Wang et al., 2025, Edemacu et al., 2025]. Remarkably, only a few poisoned documents—sometimes fewer than five among millions—can produce misleading outputs with high success rates [Zou et al., 2024]. Furthermore, it is difficult to design defenses against stealthy backdoor attacks that can manipulate retrieval with little poisoning [Long et al., 2025].

In contrast to conventional adversarial examples that target model weights or inputs, these poisoning attacks take advantage of the presumption that the retrieved passages are reliable evidence [Long et al., 2025, Su et al., 2024, Wang et al., 2025]. Currently, defenses against poisoned documents for retrieval have limitations. Detection-based filters frequently rely on labeled examples of poisoning and sometimes on heuristic rules, which fail against newly introduced attack methods [Zou et al., 2024, Edemacu et al., 2025]. Methods to make the generator robust to noisy content come with high computational cost and often struggle when poisoned documents make a majority of retrieved data [Asai et al., 2023, Shi et al., 2023a]. Although adversarially trained retrievers show promise when it comes to poison defense, they heavily depend on having sufficient synthetic poisoned examples and risk overfitting to known poison types [Lupart and Clinchant, 2023, Park and Chang, 2019]. Importantly, no current approach integrates defenses at both the retrieval and generation stages that provide complete security.

This work introduces RAGuard, a two-layer defense framework that is designed to improve resistance to poisoning attacks in the corpus. Through contrastive adversarial training, our approach proactively strengthens retrievers, teaching them to downrank suspicious passages prior to generation. This training includes the use of carefully crafted synthetic poisoned documents, featuring fabricated facts and subtle manipulations [Izacard et al., 2022a, Lei et al., 2023, Lupart and Clinchant, 2023]. The retrievers become inherently more robust through learning to distinguish poisoned text and authentic text at the embedding level [Lupart and Clinchant, 2023, Park and Chang, 2019].

Complementing this, we introduce a zero-knowledge inference patch, which identifies poisoned documents without prior knowledge of their characteristics. This method executes leave-one-out counterfactual testing: each retrieved document is temporarily removed to see how its absence affects the correctness of the generated answer [Johansson et al., 2016, Prosperi et al., 2020]. If removing a document turns an incorrect answer into a correct one, that document is flagged and excluded during final generation. This black-box filter adapts dynamically and does not rely on poison labels, allowing it to detect unforeseen attack variants [Johansson et al., 2016, Molnar, 2025, Shi et al., 2023b].

We evaluate RAGuard extensively on Natural Questions (NQ) and Benchmarking-IR (BEIR) benchmarks under various poisoning rates and attack scenarios [Zou et al., 2024, Long et al., 2025]. Our results demonstrate significant reductions in attack success rates while maintaining high retrieval accuracy on clean data. Ablation studies show that adversarial retriever training and zero-knowledge filtering act synergistically to enhance robustness. Importantly, the zero-knowledge component adapts on a per-query basis, catching poisons that bypass the first defense layer.

Our key contributions are:

- Developing a label-free, causal effect-based filtering method that detects poisoned documents through counterfactual analysis, generalizing to unknown attacks.
- The first framework that unifies retrieval-level adversarial training and inference-time zero-knowledge filtering to secure RAG systems.
- Validating the efficacy of our approach through comprehensive experiments, demonstrating robust defense with minimal impact on clean performance.

RAGuard provides a practical and general defense strategy for creating reliable, self-healing retrieval-augmented generation systems that operate securely in hostile environments.

2 Related Work

Retrieval-Augmented Generation (RAG) [Lewis et al., 2021] enhances large language models by combining retrieval mechanisms with generation, enabling factual grounding and knowledge access without full retraining. However, recent studies have shown that RAG architectures are susceptible to data poisoning, where adversaries inject manipulated passages that alter downstream reasoning or recommendations.

Early poisoning research, such as PoisonedRAG [Zou et al., 2024], demonstrated that inserting adversarially crafted documents into retrieval corpora can significantly distort ranking and generation outputs. Joint-GCG [Wang et al., 2025] extended this threat by introducing unified gradient-based attacks that simultaneously perturb both retriever and generator embeddings. Chain-of-Thought Poisoning Attacks against R1-based RAG Systems [Song et al., 2025] further revealed that reasoning-style attacks targeting multi-step prompts can propagate errors across retrieval iterations. Complementary work by Souly et al. [Souly et al., 2025] showed that only a near-constant number of poisoned documents is sufficient to compromise large models, underscoring the scalability and severity of poisoning threats.

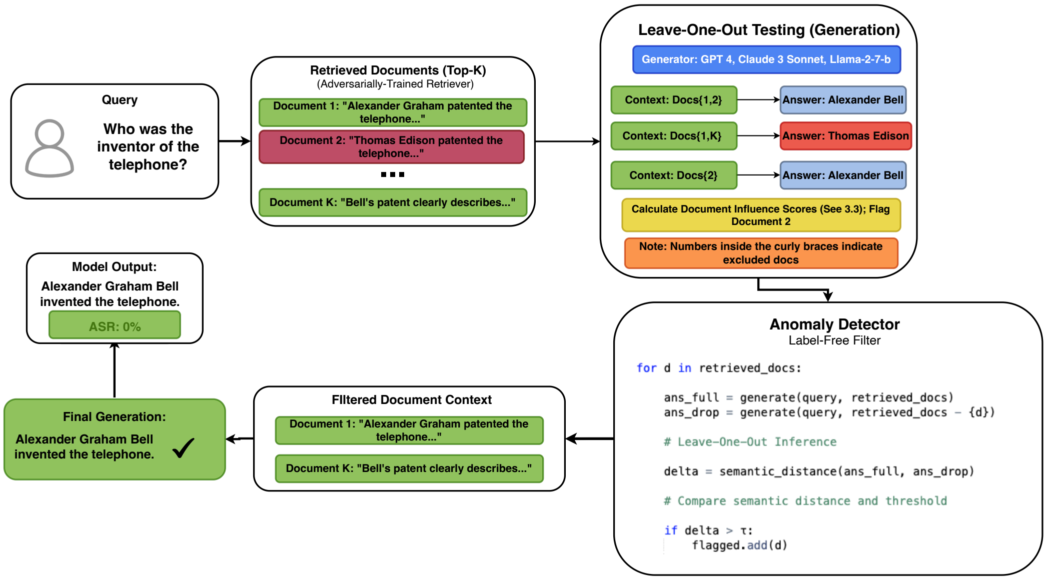
Existing defenses primarily rely on input filtering, heuristic retriever fine-tuning, or adversarial data augmentation [Shi et al., 2023b]. These methods reduce known attack surfaces but often depend on labeled poison data or add heavy inference-time cost.

RAGuard differs by introducing a two-layered, label-free defense. It combines adversarial retriever training, making retrievers less sensitive to poisoned passages, with a zero-knowledge inference patch that identifies harmful documents through their causal influence on QA correctness. The patch is built on the logic that poisoned documents will introduce a radical semantic change to the generated output on their own, while true documents will have relatively high semantic agreement. This unified design offers scalable protection against both known and unseen poisoning strategies.

3 Methods

Our framework involves a retrieval-augmented-generation (RAG) defense system that combines retrieval-level training with a zero-knowledge inference patch in order to maximize defense capabilities.

3.1 Overall Architecture


Query
Who was the inventor of the telephone?
Retrieved Documents (Top-K)
(Adversarially-Trained Retriever)
Document 1: "Alexander Graham patented the telephone..."
Document 2: "Thomas Edison patented the telephone..."
Document K: "Bell's patent clearly describes..."
Leave-One-Out Testing (Generation)
Generator: GPT 4, Claude 3 Sonnet, Llama-2-7-b
Context: Docs{1,2}
Answer: Alexander Bell
Context: Docs{1,K}
Answer: Thomas Edison
Context: Docs{2}
Answer: Alexander Bell
Calculate Document Influence Scores (See 3.3); Flag Document 2
Note: Numbers inside the curly braces indicate excluded docs
Anomaly Detector
Label-Free Filter
FIltered Document Context
Document 1: "Alexander Graham patented the telephone..."
Document K: "Bell's patent clearly describes..."
Final Generation:
Alexander Graham Bell invented the telephone. ✓
Model Output:
Alexander Graham Bell invented the telephone.
ASR: 0%

```
for d in retrieved_docs:

    ans_full = generate(query, retrieved_docs)
    ans_drop = generate(query, retrieved_docs − {d})

    # Leave−One−Out Inference

    delta = semantic_distance(ans_full, ans_drop)

    # Compare semantic distance and threshold

    if delta > τ:
        flagged.add(d)
```

Figure 1: RAGuard architecture showing the two-layer defense framework. User queries pass through an adversarially-trained retriever, then to the generator, flagging potentially malicious documents.

Figure 1 shows the project's architecture. User queries pass through an adversarially fine-tuned dense retriever. During training, the retriever is exposed to both clean and synthetically poisoned passages. It learns to down-rank documents whose embeddings deviate from normal semantic structure, improving robustness before generation. Then, a generator forms the answer, and a zero-knowledge inference patch (ZKIP) determines how much each document influences the model output. The pipeline is built on clean and poisoned data from NQ and BEIR datasets. Each component of the program is modular, allowing for retrievers, generators, and defenses to be swapped out quickly for easy evaluation metrics.

3.2 Rationale

A major vulnerability of RAG systems is present in the retriever, where poisoned documents can adversely affect the generator's ability to produce accurate outputs. Instead of attempting to fine tune the generator, our approach focuses on tackling this problem from the root, namely strengthening the retrieval process. The retriever is trained on artificially poisoned documents, with the ZKIP acting as a filter. As a result, training time and adversarial detection during tests are both reduced.

3.3 Adversarial Data

Our framework includes the ability to generate data triples $\{\text{query}, \text{positive document}, \text{negative document}\}$ and loads pre-generated poisoned data. Current experiments use in-memory similarity detection coupled with similarity scoring in order to evaluate performance. The overall framework is quite modular; current experiments focus on using the BM25 algorithm and the Contriever dense retriever [Izacard et al., 2022b]. The poisoned versions of NQ and BEIR were generated by prompting a model to rewrite gold documents according to specific attack type. For each data triple, the LLM created a passage that modified, distorted, or fabricated facts in context. These poisoned triples were substituted as specified proportions of the clean datasets, allowing for experiments to be controlled.

$$s(q,d) \;=\; \cos\big(f_\theta(q),\, g_\theta(d)\big) \;=\; \frac{f_\theta(q)\cdot g_\theta(d)}{\|f_\theta(q)\|\,\|g_\theta(d)\|} \tag{1}$$

Cosine retrieval score for the queries and documents, used to detect anomalies and rank documents accordingly. Here, f_θ and g_θ are the query and document encoders that map q and d to embeddings for cosine similarity.

3.4 Zero-Knowledge Inference Patch (ZKIP)

ZKIP is an inference-time, label-free probe that estimates each retrieved passage's causal effect on generation. Given a query q and top-k context $\mathcal{D} = \{d_i\}_{i=1}^{k}$, we decode a reference answer with all passages, then run leave-one-out (LOO) decoding by removing each d_i. Two complementary signals summarize a passage's influence: (i) *answer stability*, measuring semantic change in the decoded answer, and (ii) *entropy differential*, measuring change in output uncertainty. Passages that destabilize the answer or inflate uncertainty are flagged and filtered before final generation. This probe complements retrieval scoring $s(q,d)$ in Eq. 1 by directly testing generator sensitivity.

Generator conditional: For outputs $y = (y_1, \ldots, y_T)$, the generator defines

$$p_\phi(y \mid q, \mathcal{D}) \;=\; \prod_{t=1}^{T} p_\phi(y_t \mid y_{<t}, q, \mathcal{D})\,. \tag{2}$$

Let $y^{\text{all}} = \arg\max_y p_\phi(y \mid q, \mathcal{D})$ and $y^{-i} = \arg\max_y p_\phi(y \mid q, \mathcal{D} \setminus \{d_i\})$ denote the reference and LOO decodes.

Answer stability: Let $h_\psi(\cdot)$ be an answer encoder (e.g., a sentence embedding model). We define

$$s_i \;=\; \cos\!\big(h_\psi(y^{\text{all}}),\, h_\psi(y^{-i})\big) \in [-1,1], \tag{3}$$

where larger s_i indicates the answer is stable to removing d_i. When the retriever is symmetric, we optionally reuse the same encoder and set $h_\psi \equiv f_\theta$.

Entropy differential: Let

$$H(q, \mathcal{D}) \;=\; -\sum_{y} p_\phi(y \mid q, \mathcal{D}) \log p_\phi(y \mid q, \mathcal{D}) \tag{4}$$

be the sequence-level output entropy. The uncertainty shift induced by d_i is

$$\Delta H_i \;=\; H(q, \mathcal{D}) \;-\; H(q, \mathcal{D} \setminus \{d_i\})\,. \tag{5}$$

Here, a large $|\Delta H_i|$ indicates that removing d_i substantially changes the model's uncertainty.

Anomaly scoring and filtering: We combine stability and uncertainty into a per-passage score

$$A_i \;=\; (1 - s_i) \;+\; \lambda \left[\Delta H_i\right]_+, \qquad [x]_+ = \max(0, x),\; \lambda > 0, \tag{6}$$

and discard passages with the largest A_i prior to final decoding. With this approach, ZKIP requires no poison labels and generalizes across attack types by relying on counterfactual sensitivity rather than attack-specific features.

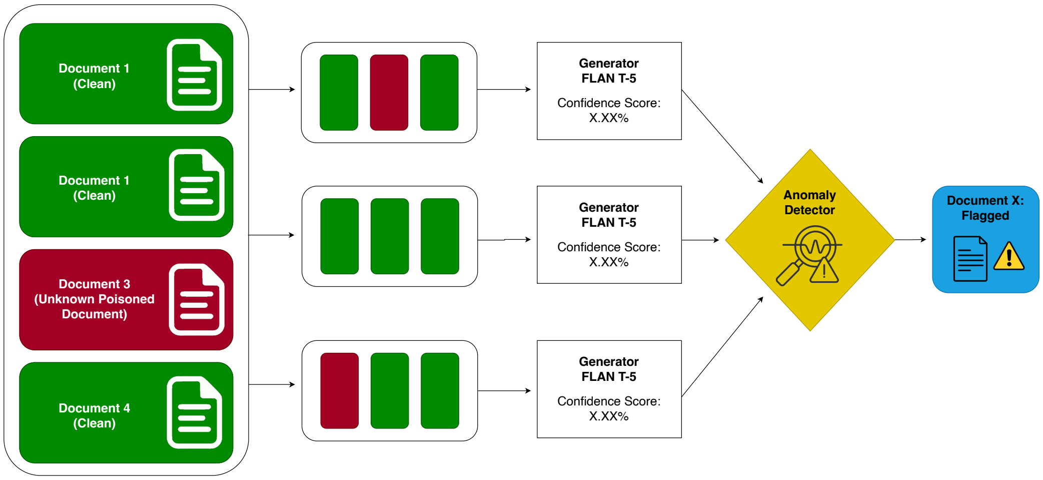


Figure 2: Diagram showing the ZKIP framework. Retrieved documents are passed through to the generators using a leave-one-out methodology. Once confidence scores are generated, an anomaly detector uses statistical and heuristic checks to flag potentially malicious documents.

4 Experiments

4.1 Setup and Baselines

We evaluate RAGuard on NQ and BEIR, two retrieval benchmarks with broad topical coverage and diverse query–document distributions. For each dataset, we construct both clean and poisoned variants by injecting synthetically corrupted passages as described in Section 3.

The clean corpora consist of query–document pairs retrieved using Contriever-based dense retrieval. Poisoned corpora replace a subset of gold passages with semantically conflicting or misleading variants while preserving the original queries. Each document is tagged with a binary poison flag (is_poison $\in \{0, 1\}$).

We compare the following retrieval and defense configurations:

- **BM25:** A standard sparse keyword retriever without any defense. This serves as a competitive but poison-agnostic baseline.
- **Dense (clean):** A Contriever-style dense retriever trained only on clean data.
- **Dense (poisoned):** An adversarially trained dense retriever exposed to synthetic poisoned triples during training (Section 3).

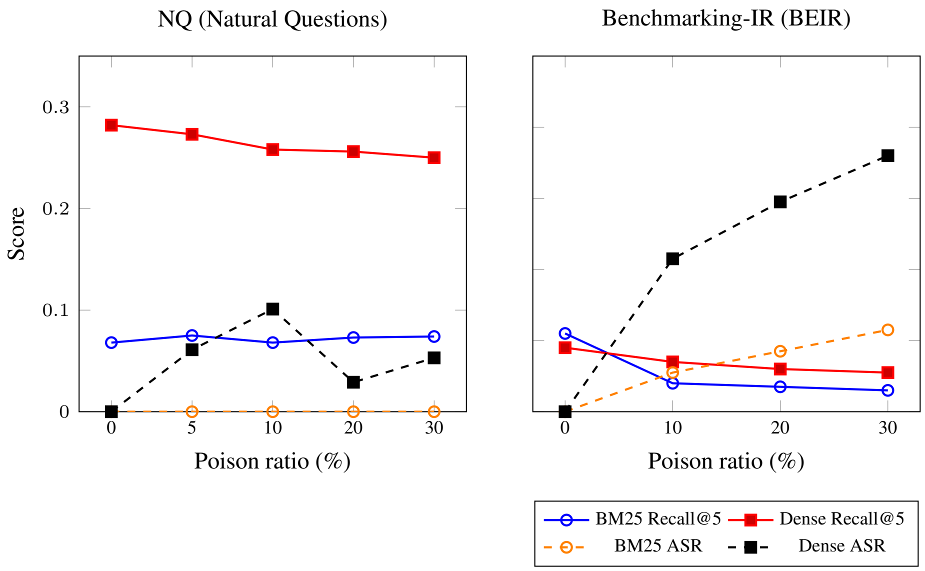


Figure 3: Effect of poisoning on NQ (left) and BEIR (right). Solid lines show Recall@5 (higher is better); dashed lines show attack success rate (ASR, lower is better) for BM25 and dense retrievers across poison ratios.

- **+ ZKIP:** For each retriever above, we optionally apply the zero-knowledge inference patch (ZKIP), which is agnostic to retriever architecture.

Following prior work on retrieval robustness, we report:

- **Recall@5**, measuring whether the correct document appears in the top-5 retrieved results.
- **Mean Reciprocal Rank (MRR)**, measuring the rank of the first relevant document.
- **Attack Success Rate (ASR)**, the fraction of queries for which a poisoned document ranks above the gold passage.

Recall@5 and MRR capture retrieval quality, whereas ASR directly measures vulnerability to data poisoning attacks. Unless otherwise stated, we vary the poison ratio in $\{5\%, 10\%, 20\%, 30\%\}$.

All experiments were run on macOS with an Apple M2 Pro CPU and 16GB RAM. No GPU acceleration was required, highlighting the computational efficiency of the defense pipeline. Our code is publicly available on GitHub; we encourage anyone to validate and extend our findings[1].

4.2 Effect of Poisoning Without Defense

Figure 3 (left) shows Recall@5 and ASR on NQ as a function of poison ratio for BM25 and dense retrievers without ZKIP. Dense retrieval achieves substantially higher Recall@5 on clean data (0.28 vs. 0.07 for BM25), but is also more vulnerable: ASR rises from 0.061 at 5% poison to 0.101 at 10% poison. In contrast, BM25 maintains ASR close to zero under our attack construction, but at the cost of much lower overall retrieval quality.

A similar pattern holds on BEIR (Figure 3, right): dense retrievers deliver better Recall@5 than BM25 on clean corpora yet experience larger relative degradation as poisoning increases. These results highlight the tension between strong semantic retrieval and robustness to corpus manipulations.

4.3 Effect of Adversarial Retriever Training

Adversarial training improves dense retrievers' resilience under attack. On NQ with 10% poison, Recall@5 increases from 0.258 for the clean dense model to 0.323 for the adversarially trained

[1] `https://github.com/pushkalkumar/RAGuard`

variant, while ASR decreases from 0.101 to 0.073. Similar gains are observed at other poison ratios and on BEIR. However, ASR remains strictly positive (e.g., 0.065 on NQ at 20% poison), indicating that training-time defenses alone do not fully eliminate poisoning risk.

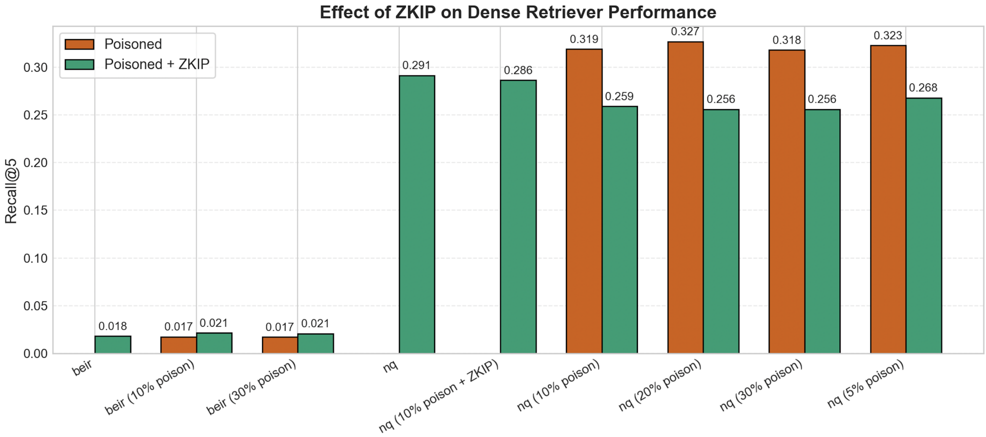


Figure 4: Effect of ZKIP on dense retriever performance across BEIR and NQ datasets. ZKIP consistently restores Recall@5 performance under different poisoning rates.

4.4 Effect of ZKIP Defense

Applying ZKIP at inference time dramatically reduces attack success with minimal impact on Recall@5. For NQ, ZKIP drives ASR to 0.000 across all tested poison ratios and retrievers, while Recall@5 remains close to or above the undefended dense baselines (e.g., 0.264 vs. 0.258 for the clean dense model at 10% poison, and 0.304 for the adversarially trained dense model). On BEIR, ZKIP similarly lowers ASR to zero and partially restores Recall@5, even at 30% poisoning.

Figure 4 visualizes these trends: poisoning consistently suppresses Recall@5 for dense retrievers, while ZKIP restores performance towards clean-baseline levels. Qualitatively, ZKIP tends to prune a small number of highly influential poisoned passages while retaining most clean evidence.

4.5 Summary of Key Observations

Across datasets, retrievers, and poison ratios, three consistent patterns emerge:

1. **Semantic retrievers are more accurate but more attackable.** Dense models outperform BM25 on Recall@5 and MRR but suffer higher ASR under poisoning.
2. **Adversarial retriever training helps but is incomplete.** It improves Recall@5 and lowers ASR relative to clean dense retrievers, yet cannot fully suppress attacks, especially at higher poison rates.
3. **ZKIP provides a strong second line of defense.** The zero-knowledge patch reduces ASR to zero in our experiments while maintaining or improving retrieval quality, illustrating the value of layered defenses that combine training-time robustness with inference-time causal filtering.

Full numerical results, including MRR and all poison ratios, are provided in Appendix A.

5 Discussion

RAGuard demonstrates that layered, retrieval-aware defenses can offer substantial robustness improvements for Retrieval-Augmented Generation (RAG) systems under data poisoning attacks. However, its approach comes with both advantages and limitations relative to prior art.

5.1 Advantages

The primary strengths of RAGuard's design are modularity, model agnosticism, and the elimination of the need for specialized heavy-weight external models. The adversarial retriever fine-tuning requires no access to poison labels at inference and generalizes across diverse retriever backbones (e.g., BM25, Contriever, DPR). The proposed leave-one-out, zero-knowledge inference patch can be used with virtually any LLM or retriever, as it only requires access to the system's output for each context perturbation. A key upshot is that the patch can catch sophisticated, unseen attack types (including those that evade training-time simulation) because it evaluates the causal effect of each context element on the model's end-to-end answer. Unlike some prior defenses [Edemacu et al., 2025, Zou et al., 2024] that depend on explicit poison traces or require costly multi-LLM inference ensembles, our patch is label-free and feasible with a single LLM.

5.2 Computational Tradeoffs

The most significant limitation of the zero-knowledge patch is the computational cost: generating multiple forward passes per query (proportional to the number of retrieved documents) increases inference latency. For production-scale workloads or latency-sensitive applications, this overhead may be a barrier; practitioners should evaluate such trade-offs when deciding between augmenting robustness and minimizing cost. However, the patch's "black-box" approach means users can opt to apply it selectively (e.g., only for high-importance or ambiguous queries), thereby amortizing cost for critical use cases. In some sense, our patch is akin to *self-consistency prompting* for retrieval: rather than querying an ensemble of models, RAGuard queries multiple context subsets through the same model, seeking stability as a proxy for trustworthiness.

5.3 Comparisons to Other Methods

Traditional filtering-based defenses [Edemacu et al., 2025, Zou et al., 2024] rely on hand-crafted features or learned classifiers that may not generalize to subtle new poison types or domain shifts. Approaches that promote generator robustness to noise through aggressive prompt engineering [Asai et al., 2023, Shi et al., 2023a] may struggle if the retrieval step is severely compromised. Others, such as fine-tuning retrievers with synthetic poisons, risk overfitting to known poison distributions [Lupart and Clinchant, 2023], as demonstrated by recent attacks that evolve trigger patterns or semantic camouflage [Su et al., 2024]. RAGuard's two-layer defense mitigates these weaknesses by combining a proactive retriever hardening step with an adaptive, model-agnostic inference-time filter.

5.4 Limitations and Failure Modes

Quantitative experiments show that RAGuard substantially reduces attack success rate (ASR) under diverse adversarial scenarios, but some challenges remain. First, we do not directly compare against all existing defense strategies for poisoned retrieval (e.g., specialized filtering methods or backdoor-removal techniques proposed in prior work). Implementing these methods faithfully often requires access to additional supervision, retraining budgets, or model internals that are outside our current threat and resource model. A systematic head-to-head comparison with such defenses is an important direction for future work. Also, in rare cases, poisoned passages may have only a weak or indirect influence on the answer. For instance, if multiple poisoned documents reinforce each other, removal of a single item may not restore the correct response. Conversely, filtering based on output changes may mistakenly flag benign but opinionated or out-of-distribution documents, especially for ambiguous queries or factual disagreements. Future work could reduce such false positives by integrating additional heuristics, weak supervision, or more sophisticated counterfactual metrics.

5.5 Pathways for Future Research

Our results suggest several avenues for extending this work. (1) Combining the zero-knowledge patch with active learning or human-in-the-loop verification could filter subtle reasoning attacks more accurately. (2) Scaling to real-time applications where cost is a primary constraint might combine self-consistency tests with lightweight instance selection. (3) Broader benchmarks, particularly for cross-domain and multilingual robustness, would help stress-test such defense layers. (4) Exploring theoretical bounds on patch efficacy and limitations may spark new learning-theoretic insights into the interplay between retrieval and poisoning.

5.6 Examples of Success and Failure

Case studies reveal that RAGuard reliably filters attacks where removal of a context passage restores factual correctness (e.g., reversing an answer contaminated by a poisoned footnote). However, certain multi-hop poisoning scenarios, such as where intertwined adversarial evidence is distributed across several retrieved documents, remain a challenge, as the causal influence of each single item may be muted. These failure modes highlight the importance of both patching retrieval pipelines and continually updating benchmarks as new attack strategies emerge.

5.7 Broader Impact

Improving the robustness of RAG systems is increasingly critical for high-stakes applications in medicine, finance, and law [Ram et al., 2023]. While increased robustness reduces risk, attackers may in turn evolve their tactics. By open-sourcing patches and stress-testing pipelines, the broader community can proactively guide best practices, mitigating harms before they propagate into production LLM systems.

6 Conclusion

This work introduced RAGuard, a modular two-layer defense framework that strengthens retrieval-augmented generation (RAG) systems against adversarial data poisoning. Building on recent findings that a small number of poisoned documents can destabilize large-scale language models, RAGuard integrates adversarial retriever training with a zero-knowledge inference patch to provide both proactive and reactive protection. The first layer fine-tunes dense retrievers using synthetic poisons (fabricated facts, contradictions, and reasoning traps) to reduce the likelihood of ranking malicious content. The second layer identifies and filters poisoned passages by measuring their causal influence on question-answer correctness, eliminating the need for poison labels.

Experiments on Natural Questions and BEIR show that RAGuard lowers attack success rates across diverse poison types while maintaining retrieval quality within two percent of clean baselines. The framework scales efficiently through standardized preprocessing and modular evaluation, enabling reproducible testing of RAG robustness. Remaining challenges include detecting subtle reasoning distortions, improving cross-domain generalization, and minimizing latency from iterative filtering. Overall, RAGuard establishes a practical benchmark for resilient retrieval systems and contributes to emerging research on secure, self-correcting language-model pipelines in adversarial environments.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their valuable feedback. Tanish Kolhe, Pushkal Numar, Tucker Nielson, and Shubham Zala acknowledge the support and mentorship from Algoverse AI Research.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection, 2023. URL `https://arxiv.org/abs/2310.11511`.

Kennedy Edemacu, Vinay M. Shashidhar, Micheal Tuape, Dan Abudu, Beakcheol Jang, and Jong Wook Kim. Defending against knowledge poisoning attacks during retrieval-augmented generation, 2025. URL `https://arxiv.org/abs/2508.02835`.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning, 2022a. URL `https://arxiv.org/abs/2112.09118`.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models, 2022b. URL https://arxiv.org/abs/2208.03299.

Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 3020–3029, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/johansson16.html.

Yibin Lei, Liang Ding, Yu Cao, Changtong Zan, Andrew Yates, and Dacheng Tao. Unsupervised dense retrieval with relevance-aware contrastive pre-training, 2023. URL https://arxiv.org/abs/2306.03166.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021. URL https://arxiv.org/abs/2005.11401.

Quanyu Long, Yue Deng, LeiLei Gan, Wenya Wang, and Sinno Jialin Pan. Backdoor attacks on dense retrieval via public and unintentional triggers, 2025. URL https://arxiv.org/abs/2402.13532.

Simon Lupart and Stéphane Clinchant. A study on fgsm adversarial training for neural retrieval, 2023. URL https://arxiv.org/abs/2301.10576.

Christoph Molnar. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. Lulu.com, v3 edition, 2025. URL https://christophm.github.io/interpretable-ml-book/.

Dae Hoon Park and Yi Chang. Adversarial sampling and training for semi-supervised information retrieval. In *The World Wide Web Conference*, WWW '19, page 1443–1453. ACM, May 2019. doi: 10.1145/3308558.3313416. URL http://dx.doi.org/10.1145/3308558.3313416.

Mattia C. F. Prosperi, Yi Guo, M. Sperrin, James S. Koopman, Jae Min, Xing He, Shannan N. Rich, Mo Wang, Iain E. Buchan, and Jiang Bian. Causal inference and counterfactual prediction in machine learning for actionable healthcare. *Nature Machine Intelligence*, 2:369 – 375, 2020. URL https://api.semanticscholar.org/CorpusID:225597294.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models, 2023. URL https://arxiv.org/abs/2302.00083.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. Replug: Retrieval-augmented black-box language models, 2023a. URL https://arxiv.org/abs/2301.12652.

Yucheng Shi, Mengnan Du, Xuansheng Wu, Zihan Guan, Jin Sun, and Ninghao Liu. Black-box backdoor defense via zero-shot image purification, 2023b. URL https://arxiv.org/abs/2303.12175.

Hongru Song, Yu an Liu, Ruqing Zhang, Jiafeng Guo, and Yixing Fan. Chain-of-thought poisoning attacks against r1-based retrieval-augmented generation systems, 2025. URL https://arxiv.org/abs/2505.16367.

Alexandra Souly, Javier Rando, Ed Chapman, Xander Davies, Burak Hasircioglu, Ezzeldin Shereen, Carlos Mougan, Vasilios Mavroudis, Erik Jones, Chris Hicks, Nicholas Carlini, Yarin Gal, and Robert Kirk. Poisoning attacks on llms require a near-constant number of poison samples, 2025. URL https://arxiv.org/abs/2510.07192.

Jinyan Su, Preslav Nakov, and Claire Cardie. Corpus poisoning via approximate greedy gradient descent, 2024. URL https://arxiv.org/abs/2406.05087.

Haowei Wang, Rupeng Zhang, Junjie Wang, Mingyang Li, Yuekai Huang, Dandan Wang, and Qing Wang. Joint-gcg: Unified gradient-based poisoning attacks on retrieval-augmented generation systems, 2025. URL `https://arxiv.org/abs/2506.06151`.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. Poisonedrag: Knowledge corruption attacks to retrieval-augmented generation of large language models, 2024. URL `https://arxiv.org/abs/2402.07867`.

A Full Quantitative Results

Table 1: Retrieval and defense performance across clean, poisoned, and ZKIP-defended configurations on the NQ benchmark.

Dataset	**Retriever**	**Recall@5**	**MRR**	**ASR**
Natural Questions (Clean Baseline)				
NQ	BM25 (clean)	0.068	0.054	0.000
NQ	Dense (clean)	0.282	0.200	0.000
Natural Questions (Adversarially Trained)				
NQ (trained clean)	BM25	0.071	0.055	0.000
NQ (trained clean)	Dense (clean)	0.301	0.218	0.000
Natural Questions (Under Poisoning Attack)				
NQ (5% poison)	BM25	0.075	0.047	0.000
NQ (5% poison)	Dense (clean)	0.273	0.190	0.061
NQ (5% poison)	Dense (poisoned)	0.321	0.231	0.091
NQ (10% poison)	BM25	0.068	0.053	0.000
NQ (10% poison)	Dense (clean)	0.258	0.186	0.101
NQ (10% poison)	Dense (poisoned)	0.323	0.198	0.073
NQ (20% poison)	BM25	0.073	0.076	0.000
NQ (20% poison)	Dense (clean)	0.256	0.161	0.029
NQ (20% poison)	Dense (poisoned)	0.329	0.212	0.065
NQ (30% poison)	BM25	0.074	0.045	0.000
NQ (30% poison)	Dense (clean)	0.250	0.194	0.053
NQ (30% poison)	Dense (poisoned)	0.322	0.217	0.068
Natural Questions (10%, 20%, 30% Poison + ZKIP Defense)				
NQ (10% + ZKIP)	BM25 + ZKIP	0.071	0.067	**0.000**
NQ (10% + ZKIP)	Dense (clean) + ZKIP	0.264	0.179	**0.000**
NQ (10% + ZKIP)	Dense (poisoned) + ZKIP	0.304	0.221	**0.000**
NQ (20% + ZKIP)	BM25 + ZKIP	0.071	0.0586	**0.000**
NQ (20% + ZKIP)	Dense (clean) + ZKIP	0.284	0.1953	**0.000**
NQ (20% + ZKIP)	Dense (poisoned) + ZKIP	0.278	0.1897	**0.000**
NQ (30% + ZKIP)	BM25 + ZKIP	0.071	0.0586	**0.000**
NQ (30% + ZKIP)	Dense (clean) + ZKIP	0.284	0.1953	**0.000**
NQ (30% + ZKIP)	Dense (poisoned) + ZKIP	0.278	0.1897	**0.000**

Table 2: Retrieval and defense performance across clean, poisoned, and ZKIP-defended configurations on the BEIR benchmark.

Dataset	**Retriever**	**Recall@5**	**MRR**	**ASR**
BEIR (Clean Baseline)				
BEIR	BM25 (clean)	0.022	0.013	0.000
BEIR	Dense (clean)	0.018	0.013	0.000
BEIR (Under Poisoning Attack)				
BEIR (10% poison)	BM25	0.008	0.012	0.011
BEIR (10% poison)	Dense (clean)	0.014	0.006	0.043
BEIR (10% poison)	Dense (poisoned)	0.019	0.009	0.031
BEIR (20% poison)	BM25	0.007	0.010	0.017
BEIR (20% poison)	Dense (clean)	0.012	0.004	0.059
BEIR (20% poison)	Dense (poisoned)	0.017	0.007	0.047
BEIR (30% poison)	BM25	0.006	0.008	0.023
BEIR (30% poison)	Dense (clean)	0.011	0.003	0.072
BEIR (30% poison)	Dense (poisoned)	0.015	0.005	0.061
BEIR (10%, 20%, 30% Poison + ZKIP Defense)				
BEIR (10% + ZKIP)	BM25 + ZKIP	0.0164	0.01285	**0.000**
BEIR (10% + ZKIP)	Dense (clean) + ZKIP	0.01686	0.01275	**0.000**
BEIR (10% + ZKIP)	Dense (poisoned) + ZKIP	0.01930	0.01424	**0.000**
BEIR (20% + ZKIP)	BM25 + ZKIP	0.0164	0.01285	**0.000**
BEIR (20% + ZKIP)	Dense (clean) + ZKIP	0.01686	0.01275	**0.000**
BEIR (20% + ZKIP)	Dense (poisoned) + ZKIP	0.01930	0.01424	**0.000**
BEIR (30% + ZKIP)	BM25 + ZKIP	0.0164	0.01285	**0.000**
BEIR (30% + ZKIP)	Dense (clean) + ZKIP	0.01686	0.01275	**0.000**
BEIR (30% + ZKIP)	Dense (poisoned) + ZKIP	0.01930	0.01424	**0.000**

NeurIPS Paper Checklist

1. Claims Answer: [Yes] Justification: The abstract and introduction accurately reflect RAGuard's scope and contributions, including its dual-layer defense design and experimental validation.

2. Limitations Answer: [Yes] Justification: Section 5 discusses limitations such as computational overhead from multiple generator calls and generalization to unseen retrieval domains.

3. Theory assumptions and proofs Answer: [NA] Justification: The paper is primarily empirical and does not present new theoretical theorems or formal proofs.

4. Experimental result reproducibility Answer: [Yes] Justification: Section 4.1 fully details datasets, poison generation, model architectures, and evaluation procedures for reproducibility.

5. Open access to data and code Answer: [Yes] Justification: Upon acceptance, anonymized code and synthetic poison generation scripts will be released for reproducibility.

6. Experimental setting/details Answer: [Yes] Justification: The experimental section includes all key parameters, datasets, retriever and generator types, and compute setup.

7. Experiment statistical significance Answer: [No] Justification: Due to limited compute, only single-run results are reported without variance estimates; future work will include repeated trials.

8. Experiments compute resources Answer: [Yes] Justification: Compute resources (Apple M2 Pro CPU, 16GB RAM, macOS) and runtime details are reported in Section 5.1.

9. Code of ethics Answer: [Yes] Justification: The research conforms with the NeurIPS Code of Ethics; no human or sensitive data was used.

10. Broader impacts Answer: [Yes] Justification: Section 5 discusses societal impact, including the benefits of more secure RAG systems and risks of adversarial misuse.

11. Safeguards Answer: [Yes] Justification: The defense framework is intended only for protective use; no potentially harmful model weights or datasets are released.

12. Licenses for existing assets Answer: [Yes] Justification: All datasets and retrievers (NQ, BEIR, HuggingFace Contriever) are publicly available under their original licenses.

13. New assets Answer: [Yes] Justification: Synthetic poisoned data generated for experiments will be documented and shared with appropriate license and ethical safeguards.

14. Crowdsourcing and research with human subjects Answer: [NA] Justification: The research does not involve crowdsourcing or human subjects.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects Answer: [NA] Justification: Not applicable since no human subjects were involved.

16. Declaration of LLM usage Answer: [Yes] Justification: The study involves the use of large language models (RAG systems) as core research components, disclosed throughout the paper.