LTT-RAG: An uncertainty-aware framework for retrieval in RAG

Anonymous ACL submission

Abstract

Retrieval augmented generation (RAG) is a widely adopted solution for enhancing large language models (LLM) with external knowledge. It provides a way to modify the prompt query with context from external sources, utilising the understanding of pre-trained LLMs, without a costly retraining. Although RAG provides more reliability, the relevance of the retrieved context to a query is not guaranteed. Retrieving incorrect information can both mislead the LLM and result in an incorrect response and substantially increase the input tokens, leading to increased answer-generation costs.

To mitigate these problems, we propose LTT-RAG. Leveraging conformal risk control, we apply the learn-then-test framework to RAG systems with two-stage retrieval pipelines (retriever and re-ranker), that: (1) controls the risk of incorrect retrieval and (2) introduces a controlled filtering that reduces the context length without removing relevant information. In this way, LTT-RAG aims to control the probability of incorrect answer from a RAG due to incorrect retrieval. In this system the user defines risk tolerance levels for both the retrieval and filtering steps, which allows for flexible and adjustable trade-off between speed, cost and reliability.

Experiments on multiple datasets demonstrate that LTT-RAG effectively enforces the desired risk guarantees and achieves competitive or superior performance compared to heuristic baselines.

1 Introduction

Large language models (LLM) are powerful tools, widely used in numerous fields (Minaee et al., 2025). Due to their size, design and large training corpora, LLMs' ability to answer questions, summarise texts, and generate ideas, among many other applications, have made them an everyday tool for many common users and experts alike (Brachman et al., 2025). However, it is well known (Lewis et al., 2020) that these models rely only on their internal knowledge and cannot answer correctly queries that are outside their training data. Instead, when presented with such a query, they *hallucinate i.e.*, they provide a well-formed and confident answer that can easily deceive users, not familiar with the correct response (Ji et al., 2023).

There are several approaches to mitigate hallucinations arising from insufficient knowledge (Ji et al., 2023). While retraining with new data is the most reliable solution, it is often impractical due to high computational cost and limited access to proprietary models. As an alternative, retrieval-augmented generation (RAG) (Lewis et al., 2020) incorporates external knowledge by retrieving relevant documents and augmenting the input prompt. The retrieval step is typically performed by smaller, efficient models, enabling fast access to large knowledge sources.

Despite its effectiveness, RAG introduces two key challenges. Firstly, most retrievers are data-driven neural models (Karpukhin et al., 2020; Khattab and Zaharia, 2020; Xiong et al., 2021), commonly based on BERT architectures (Devlin et al., 2019) or, more recently, larger LLM-based retrievers (Xu et al., 2024; Lin et al., 2024). However, they do not guarantee that all relevant information is retrieved (Karpukhin et al., 2020; Xu et al., 2025a), and such missed retrievals (also referred to as *silent failures*) can directly lead to incorrect answers from the LLM.

Secondly, retrieved documents are often substantially larger than the query, increas-

ing input token costs and potentially exceeding context window limits. This creates a trade-off between providing sufficient information and maintaining computational feasibility (Liu et al., 2024).

Re-rankers (Nogueira and Cho, 2020) aim to address the second issue by selecting a smaller, more relevant subset of documents, thus reducing context size. However, they introduce additional computational overhead and may discard useful information, creating a trade-off between efficiency and recall.

To mitigate both problems, we propose LTT-RAG, a methodology which applies the learn-then-test (LTT) framework (Angelopoulos et al., 2022a) to RAG systems. LTT-RAG operates on both the retrieval and filtering models in the system to control the risk of first extracting invalid context and then of filtering out relevant content from that context.

LTT-RAG allows for great flexibility, as the method is independent of the underlying models used for retrieval and filtering and the freedom to define task-specific risk tolerances allows for adjustable balance between speed, reliability and cost. Our key contributions are:

- A risk-control method for any retriever model, which provides finite-sample guarantees on the probability of correct context retrieval.
- A risk-control method for any re-ranker model, which provides finite-sample guarantees on the probability of incorrectly discarding relevant documents during filtering.
- Empirical results from experiments with LTT-RAG on multiple datasets in comparison with baseline methodologies, focusing on upholding the guarantees from LTT, and providing practical evaluations of the system's efficiency and predictive accuracy.

The remainder of the paper is structured as follows: Section 2 presents the related work. Section 3 introduces the necessary methodological background. Section 4 provides the details of the proposed methodology - the formal framework and theoretical formulation. Section 5 briefly describes the experimental setup, including the datasets, benchmark methods and evaluation metrics used for empirical comparison. Section 6 reports and discusses the experimental results. Finally, Section 7 concludes the paper and outlines directions for future work.

2 Related work

The widespread adoption of RAG systems has led to a range of methods aimed at improving efficiency, retrieval recall, and context selection (Cheng et al., 2025; Gao et al., 2024). A prominent direction is adaptive retrieval, which replaces fixed-k strategies by dynamically selecting the number of retrieved documents based on the complexity of the query (Asai et al., 2023; Taguchi et al., 2025; Rezaei and Dieng, 2025).

In parallel, prior work has addressed retrieval set denoising, focusing on removing irrelevant or redundant information to reduce context size while preserving answer-relevant content (Chang et al., 2025; Yan et al., 2024). Some researchers have also proposed metrics for estimating document-level information gain, with the goal of quantifying each document's contribution to the final answer (Wang et al., 2025).

In addition, considerable effort has been devoted to incorporating conformal prediction (CP) (Vovk et al., 2005; Angelopoulos and Bates, 2022) into RAG systems. Due to its model-agnostic nature, CP has enabled a range of methods that provide statistical guarantees for both retrieval and generation.

Most existing work focuses on the generation stage, where CP is applied to LLMs to control uncertainty in the produced answers (Quach et al., 2024; Mohri and Hashimoto, 2024; Cherian et al., 2024). Fewer approaches address retrieval, where the goal is to ensure that the retrieved set contains the relevant evidence with high probability (Ni et al., 2025).

Early work by Rouzrokh et al. (2024) introduces a multi-step framework for constructing retrieval sets with guaranteed confidence. Building on this, Li et al. (2024) combine CP with Bayesian optimization to obtain compact retrieval sets that still ensure coverage of relevant documents.

More recent research has focused on improv-

ing efficiency by reducing conformal set sizes. For example, Intrator et al. (2024) propose score refinement techniques that tighten conformal sets while preserving guarantees. Applications of CP have also focused on context filtering, aiming to remove irrelevant information without violating coverage guarantees. Notably, Chakraborty et al. (2026) introduce a CP-based context engineering framework that filters retrieved documents while maintaining high recall of supporting evidence.

The conformal risk control (CRC) (Angelopoulos et al., 2025) and learn-then-test (LTT) (Angelopoulos et al., 2022a) frameworks extend CP by allowing statistical guarantees beyond standard miscoverage. In particular, they enable control over more general types of risk, making them applicable in a wider range of settings. Angelopoulos et al. (2022b) use LTT to develop a procedure on top of a pre-trained re-ranker with rigorous finite-sample control of the false discovery rate. Xu et al. (2025b) expand the idea of risk-controlled recommendation systems by developing a two-stage risk control that addresses the risk in the retrieval and re-ranking of documents simultaneously. Xia et al. (2024) introduce multimodal RAG for medical vision language models. Using CRC, their method provides a strategy to control retrieval risks and a way to reduce an excessive reliance on retrieved knowledge.

3 Preliminaries

3.1 Retrieval-augmented generation

Retrieval-augmented generation integrates a retriever system with an LLM to leverage external context (Lewis et al., 2020). Given a query $q \in \mathcal{Q}$ and a corpus $\mathcal{C}$, the retriever $R : \mathcal{Q} \times \mathbb{N} \to \mathcal{P}(\mathcal{C})$ identifies a subset of k relevant documents $R(q, k)$ via embedding-based similarity (e.g., cosine similarity). This context is prepended to q as an augmented prompt for the generator.

To mitigate unhelpful or ambiguous retrieval, many systems employ a cross-encoder (re-ranker) $F : \mathcal{P}(\mathcal{C}) \times \mathcal{Q} \to \mathcal{P}(\mathbb{R})$. Unlike the retriever, the re-ranker processes every query-document pair jointly to compute finer relevance scores, filtering or reordering the context at a higher computational cost.

3.2 Conformal prediction and learn-then-test

Conformal prediction provides distribution-free, finite-sample guarantees for model outputs (Angelopoulos and Bates, 2022). Given a calibration set, D_{cal}, and an error rate, α, CP constructs sets, $C(X_{test})$, where $\mathbb{P}(Y_{test} \in C(X_{test})) \geq 1 - \alpha$. Our work utilizes the LTT framework, which generalizes CP by reframing risk control as multiple hypothesis testing problem.

The goal of LTT is to identify a set of parameters $\hat{\Lambda} \subseteq \Lambda$ such that the associated risk, $R(\lambda) \leq \alpha$ for all $\lambda \in \hat{\Lambda}$ with probability at least $1 - \delta$. The procedure involves: (i) defining null hypotheses $\mathcal{H}_0^{(i)} : R(\lambda_i) > \alpha$ for candidate parameters; (ii) computing valid p-values via concentration inequalities; and (iii) applying a family-wise error rate control algorithm to reject a subset of $\mathcal{H}_0^{(i)}$. We construct $\hat{\Lambda}$ and limit false rejection at level δ, by using the fixed-sequence testing method (Angelopoulos et al., 2022a), which is specifically designed for monotone risk settings. See Appendix D for more details about LTT.

4 Risk control for retrieval in RAG

In this section, we apply the LTT framework to RAG systems. Our objective is to control two sources of error: (i) the risk of invalid retrieval, where relevant documents are not retrieved, and (ii) the risk of invalid filtering, where relevant documents are incorrectly discarded from the retrieved set. By jointly controlling these risks, we obtain a formulation that enables the construction of a filtered context set supplied to the LLM and backed by the statistical guarantees of LTT. An overview of the procedure is shown in Figure 1.

4.1 Problem setup

Consider a scenario where an LLM must answer a query using a private catalogue of financial documents not present in its training data. To generate a factual response, the model requires the specific document containing the answer. A typical system first retrieves a set of potentially relevant documents to provide as context. However, providing too much information can lead to high API costs, exceed the model's limits, or confuse the generator

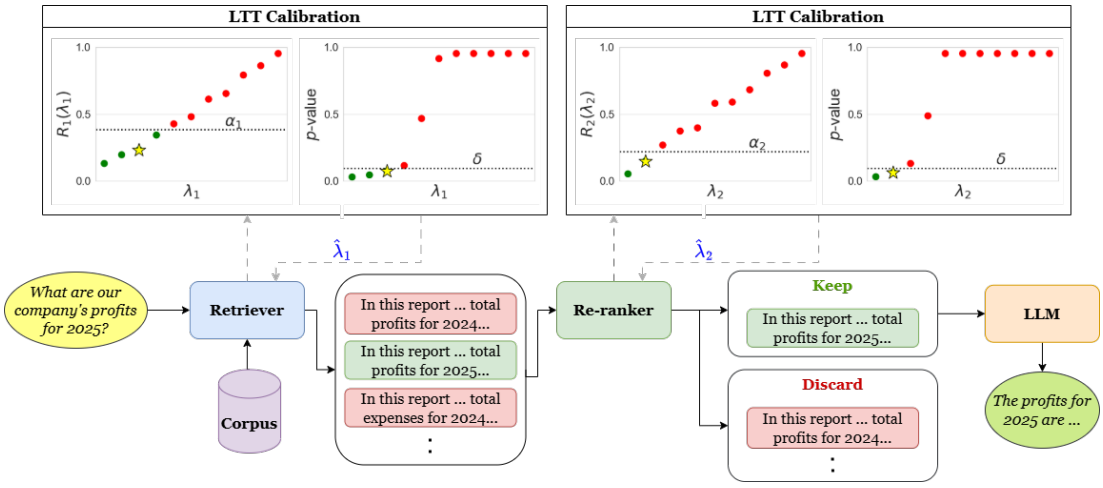

Figure 1: Overview of LTT-RAG. A retriever first selects a candidate document set for the input query, which is subsequently filtered by a re-ranker before being provided to the LLM. During calibration, the LTT procedure identifies optimal parameters $(\hat{\lambda}_1, \hat{\lambda}_2)$ that control the retrieval and filtering risks with user-defined tolerances (α_1, α_2) and confidence level at least $1 - \delta$. The resulting filtered context is then passed to the LLM to generate the final answer.

with irrelevant "noise" (Liu et al., 2024). To mitigate these issues, a filtering step is often employed to prune the context and reduce its length. However, this introduces the danger of inadvertently discarding some documents required for an accurate response.

Following the example, this two-stage process introduces two primary risks:

- Risk 1 (retrieval risk) - *Are any relevant documents not retrieved during the retrieval stage?*
- Risk 2 (filter risk) - *Are any relevant documents removed during filtering?*

To address these challenges, LTT-RAG implements a dual-stage risk control mechanism:

- **Retriever control:** A method to bound the risk within the retriever, providing finite-sample guarantees that the retrieved document sets contain the relevant documents with probability $1 - \alpha_1$.
- **Reranker control:** A method to control the risk for re-rankers, aiming to filter irrelevant documents while retaining essential information with probability $1 - \alpha_2$.

In this context, α_1 and α_2 represent user-specified risk tolerances. To construct valid statistical guarantees, we leverage the LTT framework to identify threshold pairs (λ_1, λ_2), to ensure that both risks are simultaneously controlled with a confidence level of at least $1 - \delta$. LTT-RAG outputs a set of documents that, with probability at least $1 - \delta$, simultaneously (1) contains all answer-bearing documents with probability at least $1 - \alpha_1$ and (2) excludes irrelevant documents while preserving the relevant ones with probability at least $1 - \alpha_2$.

Let $\mathcal{S} : \mathcal{Q} \to \mathcal{P}(\mathcal{C})$ be the function which returns the set of all documents from $\mathcal{C}$, required by an LLM to answer q. In our formulation we allow $|\mathcal{S}(q)| \geq 1$. To construct candidate sets we use thresholds $\lambda_1, \lambda_2 \in \mathbb{R}$, such that $C^{(1)}_{\lambda_1}(q) = \{d \mid d \in \mathcal{C}, s_1(q, d) \leq \lambda_1\}$ and $C^{(2)}_{\lambda_2}(q) = \{d \mid d \in C^{(1)}_{\lambda_1}(q) \wedge s_2(q, d) \leq \lambda_2\}$, where $s_1, s_2 : \mathcal{Q} \times \mathcal{C} \to \mathbb{R}$ are the score functions that measure the non-conformity between a query q and a document d, with higher values representing higher non-conformity. For convenience, we restrict the range of the score functions to $[0, 1)$. The resulting $C_{\lambda_2}(q)$ is the final document set, passed to the LLM. Formally, $\forall q \in \mathcal{Q}$, we define the risks above as:

$$R_1(\lambda_1) = \mathbb{P}(\exists d^* \mid d^* \in \mathcal{S}(q) \wedge d^* \notin C^{(1)}_{\lambda_1}(q))$$

$$R_2(\lambda_1, \lambda_2) = \mathbb{P}(\exists d^* \mid d^* \in \mathcal{S}(q) \wedge d^* \in C^{(1)}_{\lambda_1}(q)) \wedge d^* \notin C^{(2)}_{\lambda_2}(q))$$

We say that R_1 and R_2 are controlled if there exists a pair $(\hat{\lambda_1}, \hat{\lambda_2})$ such that $\mathbb{P}(R_1(\hat{\lambda_1}) \leq \alpha_1) \geq 1 - \delta$ and $\mathbb{P}(R_2(\hat{\lambda_1}, \hat{\lambda_2}) \leq \alpha_2) \geq 1 - \delta$.

To express the risks in a form that can be evaluated, we define two identically parametrised loss functions, $L(\lambda_i, q)$, for $i = 1, 2$ - one for the retriever and one for the reranker - which jointly capture the miscoverage of the conformal sets:

$$L(\lambda_i, q) = \mathbb{I}\big(\mathcal{S}(q) \not\subseteq C^{(i)}_{\lambda_i}(q)\big), \quad i = 1, 2.$$

We use split conformal prediction (Oliveira et al., 2024) and introduce a calibration set $D_{cal} = \{(q_i, \mathcal{S}(q_i))\}_{i=1}^m$ on which we evaluate the risks. The empirical risks on this set are defined as:

$$\hat{R}_i(\lambda_i) = \frac{1}{m} \sum_{j=1}^{m} L(\lambda_i, q_j)$$

Both risks are monotone in their respective thresholds.[1]

4.2 Data and model requirements

LTT-RAG works with datasets in the form of i.i.d. triples $\{(q_i, D_i, a_i)\}_{i=1}^n$ where q_i, a_i represent the query and ground truth answer, respectively and $D_i = \{d_1, d_2, \ldots, d_k\}, k \geq 1$ are the necessary documents for obtaining a_i.The system combines three separate models - a retriever, M_{ret} which extracts relevant documents, a reranker, M_{rer} which re-evaluates the retrieved documents and a generator, M_{gen}) that receives D_i and q_i to produce a_i. We note that LTT-RAG provides the guarantees inherent to CP, regardless of M_{ret}, M_{rer}, and M_{gen}).However, since we do not apply risk control on M_{gen}, using better models can produce smaller retrieved sets and, therefore, lead to the generation of correct answers more often.

[1]One may ask why we enforce risk control on the retrieval, given that filtering already depends on the retrieved set. If we set $\lambda_1 = 1$, we retrieve all documents and therefore ensure that all relevant ones are included. However, this is not practical, since re-ranking is significantly slower than retrieval, and applying filtering over a large corpus would be computationally expensive and slow.

5 Experimental setup

Numerical experiments were conducted on three data sets - Science QA, Biography QA, and RAG QA. Each dataset comprises of unique questions, with one or more supporting documents and one correct answer, found in the documents. All datasets were cleaned and standardised into common format.

To evaluate LTT-RAG, we compare it against four benchmark methods: LLM-only, consisting solely of the LLM without any retrieval mechanism; Oracle-RAG, where the LLM is always provided with the ground-truth relevant documents; Simple-RAG@k, a standard RAG pipeline using heuristic top-k retrieval; and Compound-RAG@k,l, a two-stage RAG system with heuristic top-k retrieval followed by top-l re-ranking and selection.

As LLM, we use the Gemma3-27B for all RAG models. In this study we compare the benchmark RAG systems with LTT-RAG by evaluating the generated answer accuracy, average token length of the context, passed to the LLM and the total inference time. To calculate the answer accuracy we compare the generated answers with the ground truth ones using a combination of exact match and F1 score.

Appendix A provides full details about the experiments.

6 Results

In this section, we discuss and summarize our findings from the conducted experiments.

We emphasize that all models are explicitly instructed to abstain from answering when no context is provided. Under this constraint, the LLM-only baseline should always abstain, resulting in a theoretical accuracy of 0; its role is therefore to establish a lower bound on inference time rather than an answer quality benchmark. However, in some experiments the LLM failed to follow the given instructions, resulting in a non-zero accuracy, which also provides an empirical lower bound for the lat-

ter. In contrast, Oracle-RAG defines the maximum achievable accuracy, since the model is given only ground-truth relevant documents, free of retrieval noise. Consequently, any remaining errors can be attributed solely to limitations of the LLM, rather than failures in the retrieval stage.

6.1 Empirical risk control on validation

In Table 1, we report $\hat{\lambda}_1$ and $\hat{\lambda}_2$ obtained via fixed-sequence testing for each α_1 and α_2, along with the corresponding empirical risks measured on the validation set. We also include the search spaces Λ_1 and Λ_2 used for the retrieval and filtering components, respectively.

Overall, the results demonstrate that the proposed procedure reliably enforces the desired risk constraints across all datasets, that is $\hat{R}_1 \leq \alpha_1$ and $\hat{R}_2 \leq \alpha_2$. In the Biography QA experiments, it can be seen that tightening the α_2 from 0.05 to 0.01 leads to a corresponding decrease in $\hat{R}_2$, while $\hat{R}_1$ remains unchanged, due to the fixed Λ_1 selection. This illustrates the modularity of the two-risk formulation, where stricter control in one component can be achieved without adversely affecting the other.

For the RAG QA dataset tighter Λ_2 search space was required due to the complexity of the dataset and the multiple documents required to correctly respond to the queries. For more details see Appendix E

6.2 Science QA

Table 2 presents the experimentation results on the Science QA test dataset. We observe that LTT-RAG achieves the second-highest accuracy among all non-oracle methods, closely matching the performance of the best-performing Compound-RAG@100,5 configuration.

Compared to the Compound-RAG variants, LTT-RAG produces significantly smaller context sets, which translates into lower inference time. Both LTT-RAG and Compound-RAG outperform the Simple-RAG baselines, highlighting the importance of combining broader retrieval with subsequent filtering to remove irrelevant information.

From a risk perspective, all Compound-RAG models satisfy both target tolerances, achieving empirical risks below the desired thresholds. However, this comes at the cost of a substantially longer context lengths. In contrast, LTT-RAG also upholds both retrieval and filtering risks within the specified tolerances, while operating with a much smaller contexts.

These results demonstrate that LTT-RAG achieves a favourable trade-off between accuracy, efficiency, and statistical reliability.

6.3 Biography QA

Table 3 further supports the observations from the previous experiment. In this experiment, retrieval emerges as the primary bottleneck: none of the heuristic baselines are able to achieve a retrieval risk below the target tolerance, $\alpha_1 = 0.1$. In addition, for both Simple-RAG and Compound-RAG, increasing the number of retrieved documents has only marginal effect on the accuracy, while still failing to meet the desired retrieval risk constraints.

In contrast, LTT-RAG, which explicitly enforces the retrieval risk constraint, is able to outperform all heuristic model variants. The average context length for LTT-RAG is substantially smaller than that of the second-best model, Compound-RAG@100,5, and even slightly smaller than the Oracle-RAG. The latter is a direct consequence of the filtering mechanism: when LTT-RAG assigns low confidence to all retrieved documents, the filtering stage may discard the entire candidate set, resulting in an empty context, which reduces the average reported length. While, in some cases, this logic can lead to an abstention on part of the model to respond to the query, it provides reliable and interpretable statistical control by explicitly taking model uncertainty into account when assessing which documents to retain.

6.4 RAG QA

The RAG QA dataset poses a significantly harder RAG challenge. As shown in Table 4, the overall answer accuracy is substantially lower than that over previous experiments, across all methods. Notably, even the Oracle-RAG configuration achieves only 0.620 accu-

Table 1: Empirical validation of risk control using fixed-sequence testing. Table shows target risk tolerances, (α_1, α_2) and the achieved empirical risks, $(\hat{R}_1, \hat{R}_2)$. The Λ_1 and Λ_2 search spaces and final parameter values $(\hat{\lambda}_1, \hat{\lambda}_2)$ are also shown.

Dataset	(α_1, α_2)	$(\hat{R}_1, \hat{R}_2)$	$\Lambda_1;\ \Lambda_2$	$(\hat{\lambda}_1, \hat{\lambda}_2)$
Science QA	(0.05, 0.10)	(0.038, 0.049)	$\text{linspace}(1, 1000, 1000)$ $\text{linspace}(0, 1, 50)$	(73, 0.79)
Biography QA	(0.10, 0.05)	(0.093, 0.038)	$\text{linspace}(1, 2500, 2500)$ $\text{linspace}(0, 1, 50)$	(246, 0.14)
Biography QA	(0.10, 0.01)	(0.093, 0.007)	$\text{linspace}(1, 2500, 2500)$ $\text{linspace}(0, 1, 50)$	(246, 0.37)
RAG QA	(0.15, 0.35)	(0.125, 0.313)	$\text{linspace}(2, 2000, 200)$ $\text{linspace}(0.99, 0.99999, 400)$	(253, 0.99511)

Table 2: Results on the Science QA dataset with $\alpha_1 = 0.05$ and $\alpha_2 = 0.10$.

Method	**Accuracy**	**Ctx. Length**	**Time (s)**	$(\hat{R}_1, \hat{R}_2)$
Simple-RAG@5	0.810	<u>400.53</u>	<u>885</u>	$(0.104, \infty)$
Simple-RAG@10	0.820	799.60	1372	$(0.066, \infty)$
Simple-RAG@20	0.823	1614.30	2440	$(0.044, \infty)$
Compound-RAG@25,5	0.836	417.34	994	$(0.040, 0.006)$
Compound-RAG@50,5	0.840	419.47	1009	$(0.026, 0.008)$
Compound-RAG@100,5	**0.848**	421.26	1093	$(0.018, 0.009)$
LTT-RAG (73, 0.79)	<u>0.844</u>	**90.27**	**715**	$(0.020, 0.037)$
LLM-only	0.005	0.00	250	–
Oracle-RAG	0.871	79.80	540	–

Table 3: Results on the Biography QA dataset with $\alpha_1 = 0.10$ and $\alpha_2 = 0.01$.

Method	**Accuracy**	**Ctx. Length**	**Time (s)**	$(\hat{R}_1, \hat{R}_2)$
Simple-RAG@5	0.545	<u>647.56</u>	<u>3168</u>	$(0.354, \infty)$
Simple-RAG@10	0.572	1293.88	5198	$(0.300, \infty)$
Simple-RAG@20	0.571	2590.27	9902	$(0.243, \infty)$
Compound-RAG@25,5	0.672	653.31	3356	$(0.23, 0.000)$
Compound-RAG@50,5	0.713	656.97	3496	$(0.18, 0.000)$
Compound-RAG@100,5	<u>0.749</u>	659.46	3684	$(0.14, 0.000)$
LTT-RAG (246, 0.37)	**0.777**	**135.76**	**2518**	$(0.085, 0.007)$
LLM-only	0.0	0.00	581	–
Oracle-RAG	0.866	137.05	1855	–

racy, indicating that performance is not solely limited by retrieval quality. Instead, this suggests that the task itself requires more complex reasoning or multi-hop aggregation, which the LLM struggles to perform even when provided with relevant context.

The Simple-RAG results further expose the limitations of heuristic retrieval. Although increasing k reduces retrieval risk, it consistently decreases answer accuracy due to the rapidly growing context length, which introduces large amounts of irrelevant information.

The Compound-RAG variants achieve the highest accuracies among the non-oracle methods. All satisfy the filtering risk constraint with empirical filtering risk below $\alpha_2 = 0.35$,

Table 4: Results on the RAG QA dataset with $\alpha_1 = 0.15$ and $\alpha_2 = 0.35$.

Method	**Accuracy**	**Ctx. Length**	**Time (s)**	$(\hat{R}_1, \hat{R}_2)$
Simple-RAG@5	0.431	<u>860.78</u>	**1108**	$(0.371, \infty)$
Simple-RAG@10	0.415	1781.30	<u>2010</u>	$(0.319, \infty)$
Simple-RAG@20	0.357	3617.48	4293	$(0.282, \infty)$
Compound-RAG@25,5	<u>0.451</u>	1079.37	2086	$(0.270, 0.081)$
Compound-RAG@50,5	0.446	1131.14	2826	$(0.234, 0.123)$
Compound-RAG@100,5	**0.457**	1161.37	4425	$(0.202, 0.160)$
LTT-RAG (253, 0.99511)	0.431	**325.48**	8257	$(0.150, 0.314)$
LLM-only	0.0	0.00	169.35	–
Oracle-RAG	0.620	298.13	589	–

but none control the retrieval risk at the desired tolerance $\alpha_1 = 0.15$.

LTT-RAG produces the smallest context across all methods, demonstrating the effectiveness of the filtering stage. However, this reduction does not improve accuracy, which remains comparable to Simple-RAG@5 and below the Compound-RAG variants. One likely reason is that many questions require multiple supporting documents; aggressive filtering may remove some of them, preventing the LLM from generating the correct answer and causing abstention due to insufficient evidence.

From a computational perspective, LTT-RAG incurs the highest inference cost, primarily due to the large retrieval set ($k = 253$). Its runtime is nearly twice that of the second slowest method, Compound-RAG@100,5.

Despite these limitations, the empirical risk control remains valid. The retrieval risk $\hat{R}_1 = 0.150$ exactly matches the target tolerance $\alpha_1 = 0.150$, while the filtering risk $\hat{R}_2 = 0.314$ remains below $\alpha_2 = 0.350$. Overall, the results show that while risk control provides reliable statistical guarantees, it does not necessarily ensure improved downstream performance in tasks requiring multi-document reasoning.

7 Conclusion

Overall, LTT-RAG demonstrates a principled approach to integrating statistical risk control into retrieval-augmented generation. Across multiple datasets, the framework consistently enforces the desired risk tolerances, providing reliable guarantees on both retrieval and filtering stages.

LTT-RAG was tested on three datasets - Science QA, Biography QA, and RAG QA. In the Science QA and Biography QA experiments, the risk control translates into clear empirical benefits: improved answer accuracy, substantially reduced context length, and competitive or improved inference time compared to heuristic baselines. These results highlight the effectiveness of combining broad candidate retrieval with selective, risk-controlled filtering.

However, the RAG QA experiments reveal important limitations. In more challenging settings requiring multi-document reasoning, risk control alone is insufficient to ensure improved downstream performance. While LTT-RAG successfully reduces irrelevant context and maintains valid risk bounds,the answer accuracy remains low. Additionally, the computational overhead can offset the efficiency benefits of context reduction.

An important direction for future work is extending risk control to the LLM itself, since the RAG QA experiments show that generation errors persist even when retrieval risks are controlled. Additionally, more expressive score functions and Λ search spaces should be explored to obtain smaller conformal sets while preserving the desired statistical guarantees.

Nevertheless, the framework provides a robust foundation for uncertainty-aware RAG systems, offering both practical improvements and formal statistical guarantees that are absent in existing heuristic approaches.

8 Limitations

Despite its advantages, LTT-RAG exhibits several limitations that affect both its practical applicability and computational efficiency.

As a conformal framework, LTT-RAG requires a separate calibration dataset. Although split conformal prediction reduces the risk of overfitting, the need for additional data can be restrictive in practice, particularly in domains where high-quality labelled data are scarce. Furthermore, the statistical guarantees rely on the assumption that the calibration data are i.i.d., which may not hold in conversational or temporally evolving settings. When calibration data are limited or distributionally mismatched, the resulting risk control may become unreliable.

The calibration procedure is also computationally expensive. The complexity of the LTT framework depends on the number of calibration examples and the size of the search spaces Λ_i, During calibration, the re-ranker must process up to $\max\{\lambda \mid \lambda \in \Lambda_1\}$ candidate documents per query, which can lead to substantial computational overhead. In our experiments, calibration required approximately 15–20$\times$ more time than inference (without including the LLM itself in the calibration phase), potentially limiting scalability in large-scale applications.

Another limitation concerns the design of the search spaces and scoring functions. If the search spaces are poorly specified, the framework may fail to identify parameters $(\hat{\lambda}_1, \hat{\lambda}_2)$ that satisfy the desired risk tolerances. This is one of the reasons for why the risk tolerances in the RAG QA experiment are high. Consequently, the method requires prior intuition regarding the behaviour of the retrieval and re-ranking components or more advanced approaches for constructing the Λ spaces. Similarly, the choice of score functions is highly important. While the desired guarantees will always be upheld, the relatively simple scoring mechanisms we employ in this work may lead to wider conformal sets and, therefore, slower performance, compared to more expressive alternatives.

Although LTT-RAG demonstrates strong overall performance while providing effective risk control, the experimental evaluation is not exhaustive. The top-k retrieval and top-l re-ranking parameters for the baseline methods were selected manually rather than systematically optimized using validation performance or recall-based tuning. In practice, such parameters are commonly chosen through grid search or validation procedures, which could improve baseline performance, although these methods would still lack the explicit statistical guarantees provided by LTT-RAG.

Finally, the scope of experimentation was constrained by available computational resources. All experiments were conducted on a single workstation GPU with 32 GB of VRAM, limiting the range of models that could be evaluated. Many larger or more advanced models were either infeasible to run or required prohibitively long execution times. Consequently, the study was restricted to a smaller set of practically manageable models and datasets, preventing broader evaluation on larger-scale benchmarks and more capable architectures.

References

Anastasios N. Angelopoulos and Stephen Bates. 2022. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *Preprint*, arXiv:2107.07511.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. 2022a. Learn then test: Calibrating predictive algorithms to achieve risk control. *Preprint*, arXiv:2110.01052.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. 2025. Conformal risk control. *Preprint*, arXiv:2208.02814.

Anastasios N. Angelopoulos, Karl Krauth, Stephen Bates, Yixin Wang, and Michael I. Jordan. 2022b. Recommendation systems with distribution-free reliability guarantees. *Preprint*, arXiv:2207.01609.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *Preprint*, arXiv:2310.11511.

Michelle Brachman, Amina El-Ashry, Casey Dugan, and Werner Geyer. 2025. Current and future use of large language models for knowledge work. *Proc. ACM Hum.-Comput. Interact.*, 9(7).

Debashish Chakraborty, Eugene Yang, Daniel Khashabi, Dawn Lawrie, and Kevin Duh. 2026. Principled context engineering for rag: Statistical guarantees via conformal prediction. *Preprint*, arXiv:2511.17908.

Chia-Yuan Chang, Zhimeng Jiang, Vineeth Rakesh, Menghai Pan, Chin-Chia Michael Yeh, Guanchu Wang, Mingzhi Hu, Zhichao Xu, Yan Zheng, Mahashweta Das, and Na Zou. 2025. MAIN-RAG: Multi-agent filtering retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2607–2622, Vienna, Austria. Association for Computational Linguistics.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Mingyue Cheng, Yucong Luo, Jie Ouyang, Qi Liu, Huijie Liu, Li Li, Shuo Yu, Bohou Zhang, Jiawei Cao, Jie Ma, Daoyu Wang, and Enhong Chen. 2025. A survey on knowledge-oriented retrieval-augmented generation. *Preprint*, arXiv:2503.10677.

John Cherian, Isaac Gibbs, and Emmanuel Candes. 2024. Large language model validity via enhanced conformal prediction methods. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Yotam Intrator, Regev Cohen, Ori Kelner, Roman Goldenberg, Ehud Rivlin, and Daniel Freedman. 2024. Streamlining conformal information retrieval via score refinement. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 186–191, Miami, Florida, USA. Association for Computational Linguistics.

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

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 39–48, New York, NY, USA. Association for Computing Machinery.

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. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023a. Making large language models a better foundation for dense retrieval. *Preprint*, arXiv:2312.15503.

Shuo Li, Sangdon Park, Insup Lee, and Osbert Bastani. 2024. Traq: Trustworthy retrieval augmented question answering via conformal prediction. *Preprint*, arXiv:2307.04642.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Sheng-Chieh Lin, Chankyu Lee, Mohammad Shoeybi, Jimmy Lin, Bryan Catanzaro, and Wei Ping. 2024. Mm-embed: Universal multimodal retrieval with multimodal llms. *Preprint*, arXiv:2411.02571.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2025. Large language models: A survey. *Preprint*, arXiv:2402.06196.

Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Bo Ni, Zheyuan Liu, Leyao Wang, Yongjia Lei, Yuying Zhao, Xueqi Cheng, Qingkai Zeng, Luna Dong, Yinglong Xia, Krishnaram Kenthapadi, Ryan Rossi, Franck Dernoncourt, Md Mehrab Tanjim, Nesreen Ahmed, Xiaorui Liu, Wenqi Fan, Erik Blasch, Yu Wang, Meng Jiang, and Tyler Derr. 2025. Towards trustworthy retrieval augmented generation for large language models: A survey. *Preprint*, arXiv:2502.06872.

Rodrigo Nogueira and Kyunghyun Cho. 2020. Passage re-ranking with bert. *Preprint*, arXiv:1901.04085.

Roberto I. Oliveira, Paulo Orenstein, Thiago Ramos, and João Vitor Romano. 2024. Split conformal prediction and non-exchangeable data. *Journal of Machine Learning Research*, 25(225):1–38.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. 2024. Conformal language modeling. In *The Twelfth International Conference on Learning Representations*.

Mohammad Reza Rezaei and Adji Bousso Dieng. 2025. Vendi-rag: Adaptively trading-off diversity and quality significantly improves retrieval augmented generation with llms. *Preprint*, arXiv:2502.11228.

Pouria Rouzrokh, Shahriar Faghani, Cooper U. Gamble, Moein Shariatnia, and Bradley J. Erickson. 2024. Conflare: Conformal large language model retrieval. *Preprint*, arXiv:2404.04287.

Chihiro Taguchi, Seiji Maekawa, and Nikita Bhutani. 2025. Efficient context selection for long-context QA: No tuning, no iteration, just adaptive-k. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20105–20130, Suzhou, China. Association for Computational Linguistics.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*.

Zihan Wang, Zihan Liang, Zhou Shao, Yufei Ma, Huangyu Dai, Ben Chen, Lingtao Mao, Chenyi Lei, Yuqing Ding, and Han Li. 2025. Infogain-rag: Boosting retrieval-augmented generation via document information gain-based reranking and filtering. *Preprint*, arXiv:2509.12765.

Peng Xia, Kangyu Zhu, Haoran Li, Hongtu Zhu, Yun Li, Gang Li, Linjun Zhang, and Huaxiu Yao. 2024. RULE: Reliable multimodal RAG for factuality in medical vision language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1081–1093, Miami, Florida, USA. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*.

Ran Xu, Wenqi Shi, Yue Yu, Yuchen Zhuang, Yanqiao Zhu, May Dongmei Wang, Joyce C. Ho, Chao Zhang, and Carl Yang. 2024. BMRetriever: Tuning large language models as better biomedical text retrievers. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22234–22254, Miami, Florida, USA. Association for Computational Linguistics.

Ran Xu, Wenqi Shi, Yuchen Zhuang, Yue Yu, Joyce C. Ho, Haoyu Wang, and Carl

Yang. 2025a. Collab-rag: Boosting retrieval-augmented generation for complex question answering via white-box and black-box llm collaboration. *Preprint*, arXiv:2504.04915.

Yunpeng Xu, Mufang Ying, Wenge Guo, and Zhi Wei. 2025b. Two-stage risk control with application to ranked retrieval. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, IJCAI '25.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *Preprint*, arXiv:2401.15884.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. *Preprint*, arXiv:2407.19669.

A Experimental setup

A.1 Dataset selection

We conducted experiments on three distinct datasets - Science QA[2], Biography QA[3] and RAG QA[4].

Science QA comprises 13,679 crowdsourced science exam questions. We use a subset of Biography QA containing 18,002 questions across 3,000 sampled biographies. RAG QA is a collection of six RAG datasets: 2WikiMultihopQA, HotpotQA, Musique, PopQA, TrivialQA, and PubMedQA. From it, we select 3,000 multi-hop queries with 6,080 supporting documents.

All datasets were cleaned of duplicates and null values, standardized into a JSON format (question, context, and ground truth answer), and partitioned into calibration, validation, and test sets using a 5:2:3 ratio.

A.2 Model selection

For the retrieval and reranking stages, we utilized the `SentenceTransformers` library via the Hugging Face ecosystem. For the Science QA and Biography QA datasets, we employed the `all-mpnet-base-v2` retriever[5] in conjunction with the `ms-marco-MiniLM-L6-v2` reranker[6]. For the RAG QA dataset, we transitioned to a larger-scale architecture, utilizing Alibaba's `gte-large-en-v1.5` retriever (Li et al., 2023b; Zhang et al., 2024) and the `bge-reranker-v2-m3` (Li et al., 2023a; Chen et al., 2024). All models were deployed using their provided off-the-shelf snapshots without further fine-tuning to evaluate their baseline performance within our framework.

A.3 Language model setup

The generative component of our pipeline utilized Gemma-3 27B (Team et al., 2025). To balance computational efficiency with model performance, we employed 4-bit quantization via the `llama-cpp` package. Specifically, we used the `google/gemma-3-27b-it-qat-q4-0-gguf` snapshot, which was executed locally to ensure data privacy and reproducible inference latency.

A.4 Computational environment

All experiments were conducted on a workstation running Ubuntu 22.04 LTS, equipped with an Intel(R) Xeon(R) w5-3425 CPU and 128 GB of RAM. Hardware acceleration was provided by a single NVIDIA RTX A5000 GPU with 32 GB of VRAM. The software stack consisted of Python 3.12 and PyTorch 2.8.0 with CUDA 12.9 support.

A.5 LTT Implementation and calibration

Our implementation of the risk control framework is based on the LTT methodology and code provided by Angelopoulos et al. (2022a)[7]. We employ the fixed sequence testing algorithm for the selection of the optimal thresholds $(\hat{\lambda}_1, \hat{\lambda}_2)$. To integrate the retrieval and reranking subsystems into the LTT framework, we define the following score functions:

- **Retriever score function (s_1):** We define a positioning score function $s_1(q, c)$, representing the rank-order position of

[2] https://www.kaggle.com/datasets/thedevastator/sciq-a-dataset-for-science-question-answering

[3] https://huggingface.co/datasets/ragrawal36/biography-qa

[4] https://huggingface.co/datasets/AQ-MedAI/RAG-QA-Leaderboard

[5] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[6] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L6-v2

[7] https://github.com/aangelopoulos/ltt

Table 5: Risk tolerances for each dataset experiment. For comparison purposes, two experiments conducted on the Biography QA dataset are shown.

Dataset	(α_1, α_2)
Science QA	(0.05, 0.10)
Biography QA	(0.10, 0.05) (0.10, 0.01)
RAG QA	(0.15, 0.35)

document c within the candidate set $\mathcal{C}$, determined by the sorted embedding distances between the query q and all documents.

- **Re-ranker score function (s_2):** For the re-ranking stage, we utilize a softmax-based relevance score function s_2, defined as:

$$s_2(q, c) = 1 - \frac{e^{r(c)}}{\sum_{i=1}^{|\mathcal{C}|} e^{r(c_i)}} \quad (1)$$

  where $r(c)$ is the relevance score assigned to document c by the re-ranker.

We explore the candidate threshold spaces Λ_1 and Λ_2 using uniformly spaced grids. For all experiments, the global error level was set to $\delta = 0.1$, while the specific risk tolerances, α_1 and α_2, utilized for each dataset are summarized in Table 5. Calibration was performed using 50% of the total data in each case and the validation sets were used to empirically confirm the performance of the risk control guarantees. Fixed random number generator seeds were used in all experiments.

A.6 LLM answer generation

To evaluate the downstream generative performance and verify the theoretical guarantees of LTT-RAG, we conducted an experimental evaluation using the held-out test dataset as follows:

1. **Retrieval and filtering:** We obtain C_{λ_2}.

2. **Generative prompting:** The query and the filtered context $\mathcal{C}_{\hat{\lambda}_2}$ are concatenated with a task-specific system instruction. The model is instructed to follow a *grounding constraint*, whereby it must abstain from answering if the provided context does not contain sufficient information to satisfy the query. See Appendix B for the full prompt template.

3. **Metrics:** We assess the generated outputs using exact match and F1 token-level scores relative to the ground truth. Additionally, we evaluate system efficiency by measuring the cumulative token length of the context, passed to the LLM and the total inference latency.

To compute accuracy, we use a hybrid matching procedure: a prediction is considered correct if the ground-truth answer appears as a substring in the generated output. If this condition is not met, we additionally compute token-level F1 between the predicted and reference answers and count the prediction as correct if the F1 exceeds 0.5. For numerical answers, all values are normalized into their word form prior to evaluation to ensure consistent matching.

To contextualize the performance of LTT-RAG, we compare it against four distinct benchmarks:

- **LLM-only:** A zero-context baseline that relies solely on the model's internal parameters, providing a lower bound for accuracy and an upper bound for inference speed.

- **Simple-RAG@k:** A standard retrieval-augmented setup utilizing a heuristic top-k strategy without statistical risk control.

- **Compound-RAG@k,l:** A retrieval + re-ranking RAG setup that applies a heuristic top-k retrieval stage followed by top-l selection.

- **Oracle-RAG:** An idealized setting where the LLM is provided only with the ground-truth relevant documents, serving as the performance upper bound.

While the LLM-only and Oracle-RAG benchmarks define the performance envelope of the system, our primary objective is to comapre LTT-RAG with the Simple-RAG and Compound-RAG baselines in terms of balancing rigorous risk control with practical utility.

B Prompt template

The instructions prompt. used in all experiments is shown in Table 6. In the prompt we replace "$$$" with the context received from the retrieval component, while the "###" is replaced with the question body.

C Extensive risk control

In Figure 2 and Figure 3 we show more extensive evaluation of the risk control on the validation sets for Science QA and Biography QA. In particular, we evaluate on multiple (α_1, α_2) pairs to show that the risk is successfully controlled in various scenarios.

D Learn-then-test framework

Our work uses the learn-then-test (LTT) framework, a calibration method for machine learning models. It adjusts model predictions so they come with clear statistical guarantees that hold even with a finite amount of data. LTT reframes the risk-control problem as multiple hypothesis testing, allowing the use of well-studied mathematical theory in UQ. The goal of LTT is to find at least one parameter $\lambda \in \Lambda$, such that the associated with the system risk $R(\lambda)$ does not exceed the user-defined risk tolerance α. To do that the procedure does the following:

1. For a finite number of candidates λ_i from Λ, assign the null hypothesis $\mathcal{H}_\rangle : R(\lambda_i) > \alpha$

2. For each null hypothesis, compute a finite-sample valid p-value using a concentration inequality.

3. Return a set $\hat{\Lambda} \subseteq \Lambda$, with the help of an algorithm, that controls the family-wise error rate (FWER).

Every element in $\hat{\Lambda}$ controls the risk with at least $(1 - \delta)$ probability.

To construct a valid p-value p_i for each null hypothesis H_i is done by comparing empirical risk estimates to concentration bounds. For each H_i, the risk is evaluated using a calibration set. The resulting p-value is obtained via a concentration inequality, which controls the deviation between empirical and true risk, ensuring that small p-values indicate violation of H_i.

To construct $\hat{\Lambda}$, an FWER control algorithm is required. An FWER-controlling algorithm is any procedure that returns a subset $P \subseteq P^*$ of rejected hypotheses, where P^* denotes the set of all null hypotheses rejected by the corresponding hypothesis tests. The goal of the algorithm is to to use the p-values to reject a subset of $\mathcal{H}_i$ while limiting the probability of making any false rejections at level δ.

Learn-then-test naturally extends to the simultaneous control of multiple risks. By allowing the parameter space Λ to be multidimensional, the framework can jointly control risks $R_1, \ldots, R_k$, each associated with a target tolerance $\alpha_1, \ldots, \alpha_k$. In this setting, the null hypothesis for a parameter configuration $\lambda \in \Lambda$ is defined as $\mathcal{H}_i : \exists j \in \{1, \ldots, k\}$ such that $R_j(\lambda) > \alpha_j$. That is, the null hypothesis is violated whenever at least one risk exceeds its prescribed tolerance. Standard FWER-controlling procedures remain applicable in this multi-risk setting.

E Lambda search space for RAG QA

The RAG QA dataset is substantially more challenging than Science QA and Biography QA due to its multi-hop nature. The relatively small retrieval and re-ranking models used in our experiments often fail to fully capture the semantic complexity of the queries, either missing relevant documents during retrieval or assigning low scores to relevant documents during re-ranking. Empirically, we observed that no re-ranking threshold below 0.99 was able to satisfy the desired filtering risk tolerance $\alpha_2 = 0.35$. Consequently, the search space for Λ_2 was restricted to a narrow interval between 0.99 and 1.

This narrow range is also influenced by the softmax-based scoring function. In many cases, the re-ranker assigned significantly lower scores to all but the highest-ranked documents. After applying the score function s_2, these values became highly concentrated near 1, requiring extremely high thresholds to achieve meaningful filtering behaviour and valid risk control.

Prompt template for the LLM

```
You are solving a reading comprehension task.

You will be given:
- a question
- a context passage

Rules:
1. Answer using ONLY the provided context.
2. If the answer appears in the context, return the exact word or short phrase from the text.
3. If the answer is not present in the context, return: NA
4. Output ONLY the answer text. Do not include explanations.
5. Return only the answer string with no additional characters.

Format:

<question>: ...
<context>: ...
<answer>:

Examples:

Example 1
<question>: What is the capital of Bulgaria?
<context>: Bulgaria is a country in Southeast Europe. Sofia is the nation's capital and largest city.
<answer>: Sofia

Example 2
<question>: What event caused the end of the Byzantine Empire?
<context>: The fall of Constantinople marked the effective end of the Byzantine Empire.
<answer>: The fall of Constantinople

Example 3
<question>: What is the measurement unit of electric current?
<context>: The volt is the unit of measurement of electric potential.
<answer>: NA

Now answer:

<question>: ###
<context>: $$$
<answer>:
```

Table 6: Prompt template used for answer generation.

F AI use

During the study, LLM agents were used for polishing the paper's writing, related work search and help with code generation. All research is done by the authors and no creative work was done by a non-human.

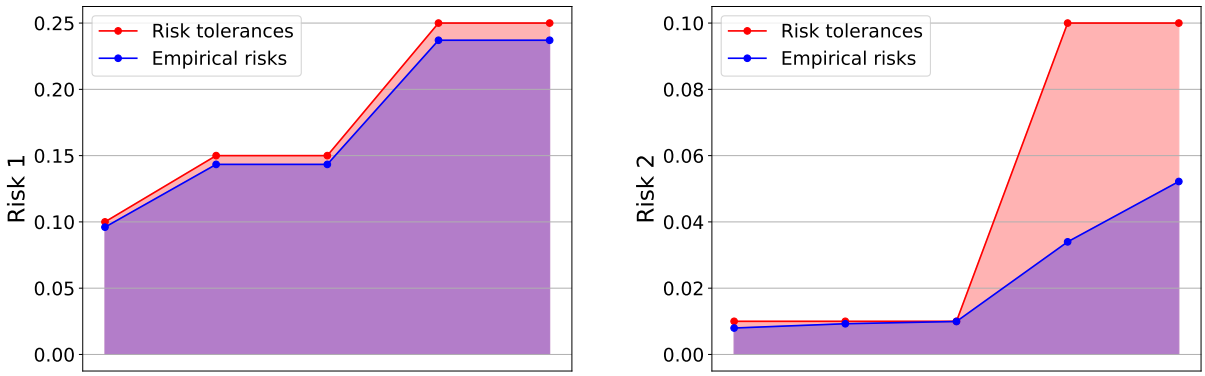

Figure 2: Comparison between risk tolerances and empirical risks for the two risks on Biography QA.

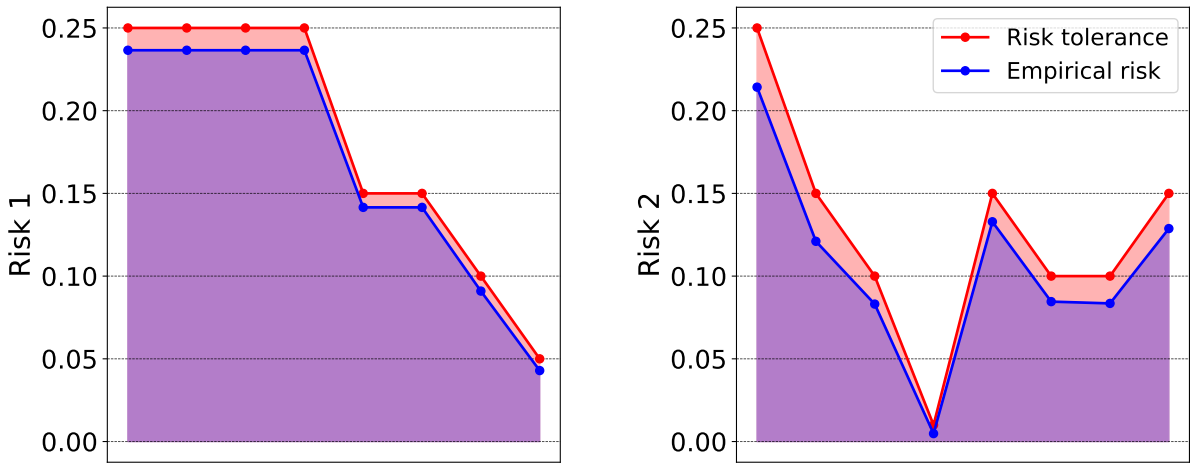

Figure 3: Comparison between risk tolerances and empirical risks for the two risks on Science QA.