AVE Trends in Intelligent Computing Systems

A Confidence-Aware Framework for Failure Prediction and Mitigation in RAG Systems

Gunji Geethika[1], B. Deepa[2,*], Vaishnavi Venkatesh[3], Chin-Shiuh Shieh[4], Ooi Junhao[5], Ramadas Kandath[6]

[1,3]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[2]Department of Electronics and Communication Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[4]Department of Electronic Engineering, National Kaohsiung University of Science and Technology, Kaohsiung City, Kaohsiung, Taiwan.
[5]Department of Information Technology and Engineering, Kota Bharu Systems Sdn Bhd, Kota Bharu, Kelantan, Malaysia.
[6]Department of FSM Adoption and Services, SAP, Dallas, Texas, United States of America.
gg7907@srmist.edu.in[1], deepab@srmist.edu.in[2], vv7617@srmist.edu.in[3], csshieh@nkust.edu.tw[4], ooi.junhao@kotabharu-systems.com.my[5], ramadas.kandath@sap.com[6]

*Corresponding author

Abstract: Retrieval Augmented Generation (RAG) has significantly improved the factual grounding of Large Language Models (LLMs). Despite advances in RAG systems, they still suffer from hallucinations because retrieval errors propagate during generation. So far, research has focused on the quality of RAG generation. A major research gap remains in detecting failures actively at the retrieval decision point. In this paper, a Confidence-Aware Control Framework will be proposed to bridge the research gap. The framework includes: (i) confidence-driven selective prediction with retrieval similarity margins, (ii) interpretable failure type categorization over retrieval and ranking steps, (iii) iterative retrieval control for low-confidence queries, and (iv) a sandwich-weighted context reorganization strategy tailored to reduce the Lost-in-the-Middle (LIM) positional bias. Experiments on the SciQA dataset show that the framework improves selective answering performance from 0.24 to 0.50 (+108%) and middle-position context usage from 0.29 to 0.60 (+107%). The framework intends to create scalable, interpretable, and accountable RAG models. By emphasizing these characteristics before producing answers, it creates an environment for reliable, trustworthy RAG models to be deployed in high-risk areas.

Keywords: Retrieval Augmented Generation (RAG); Large Language Models (LLMs); Retrieval Errors; Failure Detection; Selective Prediction; Retrieval Similarity Margins; Failure Type Categorization.

Cite as: G. Geethika, B. Deepa, V. Venkatesh, C. S. Shieh, O. Junhao, and R. Kandath, "A Confidence-Aware Framework for Failure Prediction and Mitigation in RAG Systems," *AVE Trends in Intelligent Computing Systems*, vol. 3, no. 3, pp. 121–131, 2026.

Journal Homepage: https://www.avepubs.com/user/journals/details/ATICS

Received on: 29/05/2025, **Revised on:** 24/09/2025, **Accepted on:** 31/10/2025, **Published on:** 07/09/2026

DOI: https://doi.org/10.64091/ATICS.2026.000351

1. Introduction

Large language models have attracted tremendous attention for the advances they have brought to natural language processing. Yet, they continue to produce hallucinated results (text that appears reasonable but is inaccurate) [7]. RAG has emerged as a

Copyright © 2026 G. Geethika *et al.,* licensed to AVE Trends Publishing Company. This is an open access article distributed under CC BY-NC-SA 4.0, which permits copying, remixing, redistributing, transformation, and building upon the material in any medium so long as the original work is properly cited.

solution to this issue by grounding the output of the model in external data sources, which has increased the factuality of the generated answer, but in doing so, it creates another set of problems with incorrect answers propagating from document retrieval component failures (no relevant document found or creating confusion with other irrelevant documents) through the generation pipeline [3]; [4]. Continued exploration of fine-tuning and vector-based databases has also created RAG systems as black boxes, with no ability to produce accurate confidence estimates or evaluate query failures [8]. Retrieval-Augmented Generation systems face major barriers to use in fields like medicine, law, and engineering because thoughtless answers can have serious consequences [9]. Researchers identify four major gaps in the current state of RAG practice:

- **No Abstention:** Unlike critical systems in autonomous driving or diagnosis, RAG pipelines are often designed to answer regardless of input. There's no selective prediction to remain silent when evidence isn't strong.
- **No Failure Diagnostics:** When an answer goes awry, there's no insight into whether the error was in retrieval (missing data), ranking (irrelevant ranking), or generation (hallucination despite strong context).
- **Positional Bias Blind Spot:** The Lost-in-the-Middle problem shows how LLMs favor the edges of a context window and skip the middle. Most RAG systems overlook this architectural blind spot.
- **Reactive Rather than Proactive Control:** Today's best-performing systems rely on post-generation critiques. Models that forecast and prevent failure before committing to an answer are missing.

To overcome these issues, this paper proposes a Confidence-Aware Control Framework that models RAG reliability as a proactive decision-making task [12]. The major contributions of this paper are as follows:

- **Confidence-Based Selective Prediction:** Researchers propose a confidence margin based on the retrieval similarity scores ($s1 - s2$). By active refusal, the method improves the accuracy of accepted queries from 0.24 to 0.50, achieving a relative improvement of 108%.
- **Interpretable Failure Taxonomy:** Researchers design a diagnostic layer that groups failures into three categories: (i) Retrieval Failure (context missing), (ii) Ranking Failure (context misplaced), and (iii) Confidence Rejection (ambiguity).
- **LIM Mitigation via Sandwich-Weighting:** Researchers measure the Lost-in-the-Middle (LIM) effect and propose a mitigation approach using sandwich-weighted context reorganization, improving middle-position accuracy from 0.29 to 0.60 for a 107% relative gain.
- **Adaptive Iterative Control:** Researchers design an open-ended framework for low-confidence queries that initiates query expansion and retrieval iteration, thereby transforming RAG from a static to an adaptive system.

2. Related Work and Literature Survey

The state of the art in Retrieval-Augmented Generation (RAG) has shifted from simple document augmentation to complex pipelines that require multi-source integration and hallucination control [16]. We survey and discuss the extant literature on confidence processes, application-specific models and positional bias [17]; [18].

2.1. Multi-Source RAG and Confidence Mechanisms

Current trends also use multi-level confidence scores to manage model governance. Wu et al. [1] introduced MultiRAG, a knowledge-aided model that utilizes graph-level source confidence and node-level verification to suppress hallucinations. Similarly, ConfRAG uses dynamic confidence thresholding to decide whether to use retrieved information or parameterized knowledge [6]. However, ConfRAG's binary accept/reject approach does not achieve the precision needed to detect failure modes. Other models, such as CAPR, have been proposed to incorporate confidence scores into the text to manage the large language model's final output [5]; [19]. However, this work builds on these trends by moving away from input-level confidence correction to position-aware confidence scores that assess information accessibility before final generation [23]; [24].

2.2. Domain-Specific Applications and Hallucination Detection

Because accuracy is critical in high-stakes applications, researchers have developed specialized surveys and detection tools for retrieval-augmented generation (RAG). Hindi et al. [2] conducted an extensive study of RAG's effects on legal technology, emphasizing the need for interpretability and fidelity. They provided a clear classification of evaluation criteria. Chien and Huang [5] developed a new method that uses attention patterns and graph-based semantic alignment analyses to identify mistakes in graph-based RAGs by leveraging hallucinations in the generation. Unfortunately, while both diagnostic approaches can be beneficial, they are very resource-intensive and post-hoc [22]. In addition to the post-hoc analyses presented in previous papers, other descriptive works such as "Seven Failure Points" and "Mindful-RAG" list failure modes like "missing content" and "consolidation failures" but do not provide predictive models to reduce their occurrence [11]. Our work adds a new layer

of lightweight inferences during inference for the evidence chains that are typically found in answering scientific questions to these descriptions regarding the need for precision with respect to RAG evaluation metrics [14].

2.3. The Lost-in-the-Middle (LIM) Problem

A key architectural flaw is the "Lost-in-the-Middle" problem, where large language models (LLMs) tend to have a U-shaped accuracy curve and often ignore information placed in the middle of long contexts [13]. In light of the seriousness of this bias, potentially leading to a 30-40% accuracy drop, current architectures such as MultiRAG and ConfRAG fail to consider positional variability. Our framework fixes this critical problem by incorporating a "sandwich-weighted" context reordering [26]; [27]. By combining confidence-aware detection and positional correction, our approach is the first to provide a framework for proactively mitigating LIM biases in a RAG system [15].

3. Proposed Framework and System Architecture

Our approach adds a proactive control layer between the retrieval and generation phases [20]; [21]. This layer formalizes the Retrieval-Augmented Generation (RAG) task as a conditional execution task rather than a linear pipeline [25].

3.1. Problem Formulation

In a standard RAG model, a query $q \in Q$ is submitted to a retriever R, which returns a list of documents $Dk = \{d1, d2, ..., dk\}$ from a corpus K. The generator G then generates an answer $a = G(q, Dk)$ without any conditional constraints. Our approach extends this formulation by adding a control function f_ctrl, as follows:

$$a = \{ G(q, \Phi(Dk)) \text{ if } f_conf(q, Dk) \geq \tau \quad (1)$$

Abstain if $f_conf(q, Dk) < \tau\}$. Where Φ is a mitigation transformation (e.g., reordering) and τ is a confidence threshold.

3.2. System Architecture Overview

Comprising six modules - The Knowledge Index, Dense Retriever, Confidence Estimator, Failure Classifier, Iterative Controller, and LIM Mitigator - the overall system's architecture begins with a FAISS similarity search against all-MiniLM-L6-v2 embeddings (as illustrated) (Figure 1).

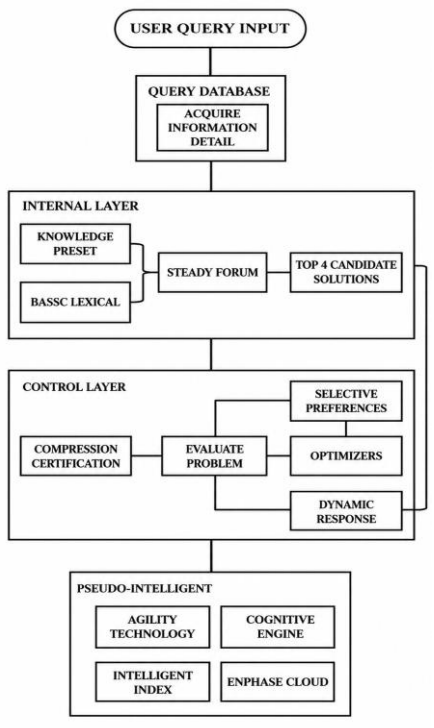

Figure 1: Proposed system architecture

The Control Layer calculates the confidence interval, C, for documents it receives for transmission to the generator; if C does not exceed the threshold τ, the Iterative Controller expands the original query via Keyword Extraction. If the confidence interval, C, exceeds the threshold τ, the LIM Mitigator reorganizes the documents into a "Sandwich" structure to maximize the attention impact achievable from the large language model during the final output generation step.

3.3. Confidence Margin and Selective Prediction

The Confidence Margin (C) is measured as the difference between the top two retrieval candidates. This value is used as a proxy for ranking confidence:

$$C = s1 - s2 \quad (2)$$

Where s1 and s2 are the cosine similarity scores of the first- and second-ranked documents, respectively; a small margin reflects high semantic ambiguity, hence the need for the system to abstain.

3.4. Classification of Failure Types and Diagnostics

To ensure diagnostic transparency, failure cases are classified into four categories to improve the system accordingly:

- **Retrieval Failure (40.8%):** The gold response $d_{gold} \notin D_k$
- **Ranking Failure (21.8%):** The gold response $d_{gold} \in D_k$ but $rank(d_{gold}) \in [2, k]$
- **Confidence Rejection (12.6%):** $C < \tau_{conf}$; This indicates an ambiguous result during retrieval
- **Success (24.8%):** $d_{gold} = d_1$ and $C \geq \tau_{conf}$

3.5. Lost-in-the-Middle (LIM) Mitigation

Researchers define performance as a function of position, P(correct | pos), and researchers have observed a strong U-shaped curve in which accuracy dips to 0.29 at middle positions [10]. To combat this, Sandwich Weighting is employed, which reorders the context as follows:

$$Context' = [Z_{critical_mid}] + [Z_{start}] + [Z_{remaining_mid}] + [Z_{end}] \quad (3)$$

This puts the most important middle-zone evidence at the front of the context. As shown in Table 2, this improves accuracy at middle positions to 0.60, a 107% increase from 0.30.

3.6. Iterative Retrieval Control

For low-confidence queries, the process is repeated iteratively. SciQA results show that iterating increases accuracy by only +3%; however, the system can trigger "re-thinking" cycles for uncertain queries, increasing average confidence from 0.018 to 0.023 after two iterations.

3.7. Experimental Setup

Within this section, the dataset, evaluation methodology, and technical approach will be discussed to provide a clear understanding of the validation process for the proposed confidence-aware RAG framework.

3.7.1. Dataset Science QA (Sci-QA)

ScienceQA (SciQA), available on Zenodo and Hugging Face, was selected to test our approach in a scientific context. ScienceQA is based on the Open Research Knowledge Graph (ORKG), where each answer is represented as a distinct resource identifier (e.g., R129632):

- **Structure:** Each entry is composed of a natural language query and its corresponding ORKG (Open Research Knowledge Graph) resource ID. For example, a query such as “transfer learning in biomedical NLP” relates to a paper title and its URL.
- **Task Formulation:** For a query q, the goal is to determine the correct ORKG resource ID. Accuracy is based on an exact match between the predicted ID and the ground truth.

- **Suitability:** This dataset suits SciQA because it is based on a structured domain of answers with ground truth for Failure Classification and is compatible with LIM Simulation based on exact positional context modification.

3.7.2. Evaluation Metrics

The following metrics were used to assess the reliability and capability of the framework in diagnosing and responding to the specified queries:

- **Top-k Accuracy:** The proportion of queries for which the correct answer is contained within the top-k retrieved documents.
- **Selective Accuracy:** Correct_Accepted divided by Total_Accepted, which represents the accuracy of the system on questions it chooses to answer.
- **Coverage:** The ratio of accepted queries to the total number of queries, which indicates the performance of the system.
- **Positional Accuracy (Acc_pos):** The accuracy of the gold evidence when positioned at the START, MIDDLE, or END of the context window.
- **Failure Distribution:** Calculates the rates of Retrieval Failure, Ranking Failure, and Confidence Rejection.

3.7.3. Implementation Details

Researchers implemented and validated the framework using the following tech stack:

- **Environment:** Python 3.10 on Google Colab (Tesla T4) for accelerating embedding computation, and a Streamlit UI for simple deployment.
- **Retrieval Engine:** Researchers selected the all-MiniLM-L6-v2 model from the sentence-transformers library. This 384-dimensional model was chosen for its ability to achieve real-time speed (below 50 ms) with strong baseline accuracy.
- **Indexing:** Similarity searches for vectors are performed by faiss-gpu with the Inner Product (IndexFlatIP) indexing configuration to calculate cosine-like similarities.
- **Hyperparameters:** By grid search, researchers picked a confidence threshold of 0.03 (τ_conf), Top-k of 5, and a retrieval loop of 2 iterations.

3.7.4. Experimental Protocol

Researchers designed five different experiments to test the framework:

- **Experiment 1 (Baseline):** Raw retrieval performance without the control layer (Top-k accuracy on 100 random queries).
- **Experiment 2 (Selective Prediction):** Varying τ_conf $\in$ [0.00, 0.10] on the Coverage-Accuracy trade-off for 500 queries.
- **Experiment 3 (Failure Analysis):** Categorizing 500 test queries into our proposed failure taxonomy to pinpoint system weaknesses.
- **Experiment 4 (Iterative Control):** Keyword-based query expansion on 100 low-confidence queries (C < 0.03$).
- **Experiment 5 (LIM Mitigation):** A controlled experiment with 150 queries (50 per position) to measure positional bias and the efficacy of the Sandwich Weighting approach.

4. Results and Analysis

This paper presents experimental results for the proposed framework across output quality, retrieval performance, confidence-aware reliability, and positional bias correction.

4.1. System Prototype and Interface Analysis

To demonstrate the framework's applicability, researchers combined the modules into a working prototype using Streamlit. This prototype verifies the control layer's real-time capability (Figure 2).

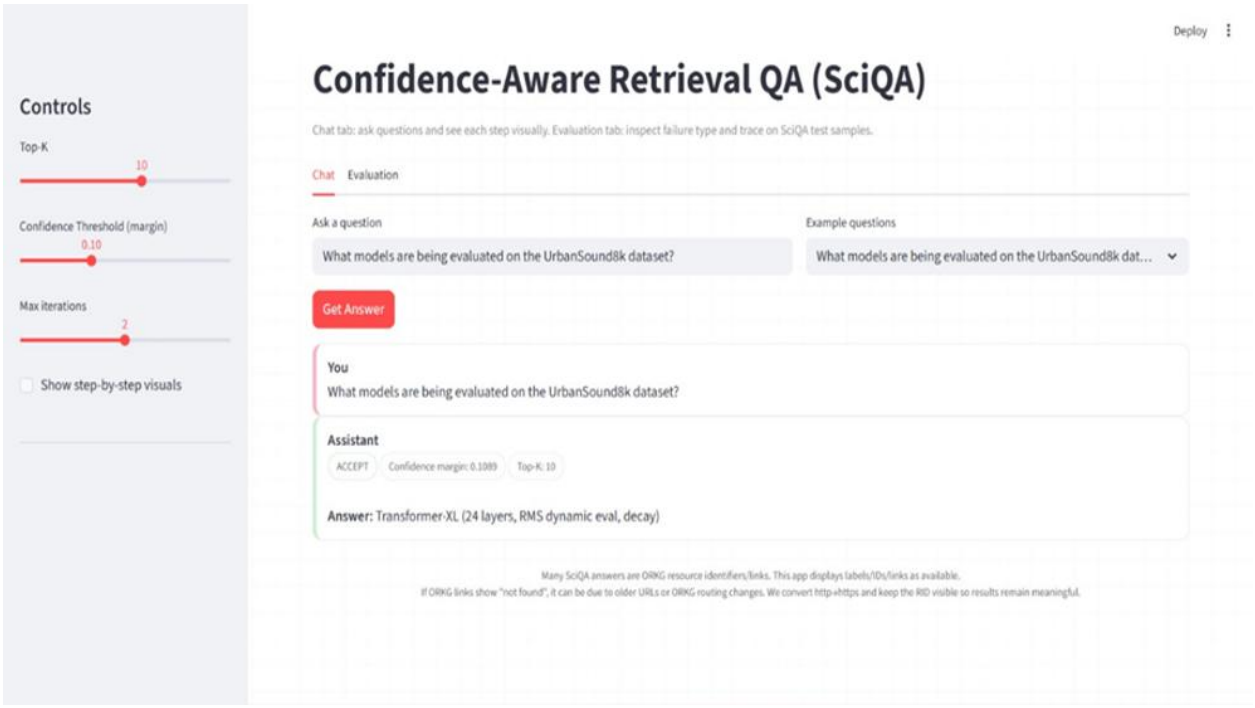

Figure 2: Confidence-aware retrieval QA dashboard

A graphical user interface has been created to enable a "Human-in-the-loop" process for setting up RAG.

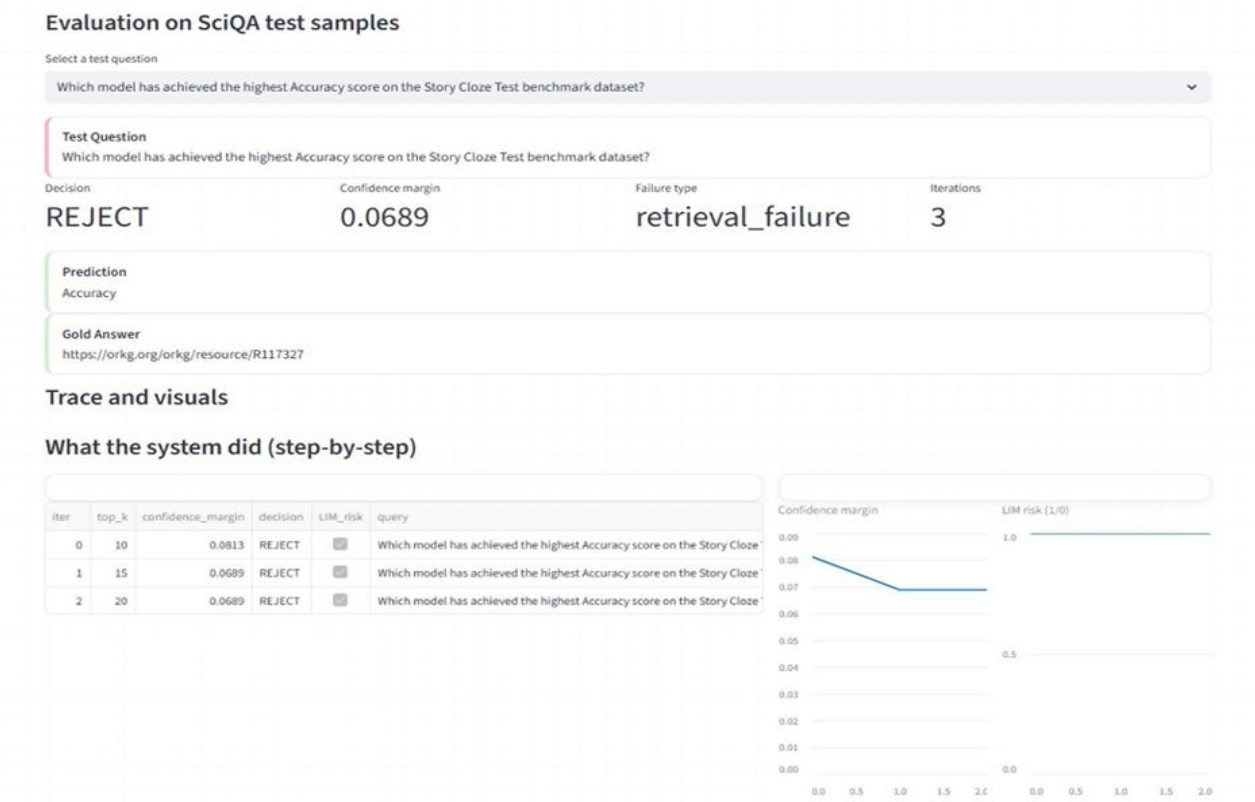

Figure 3: Evaluation on SciQA test samples

The graphical user interface lets users enter a scientific query or select one from the available SciQA sample sets and controls the Control Layer hyperparameters (Top-k value, confidence threshold (τ_conf), and max number of Iterations). This provides

domain experts with real-time access to adjust the trade-off between coverage and accuracy. In the prototype evaluation phase, "transparency" (as illustrated in Figure 3) will be the key factor used to assess the prototype.

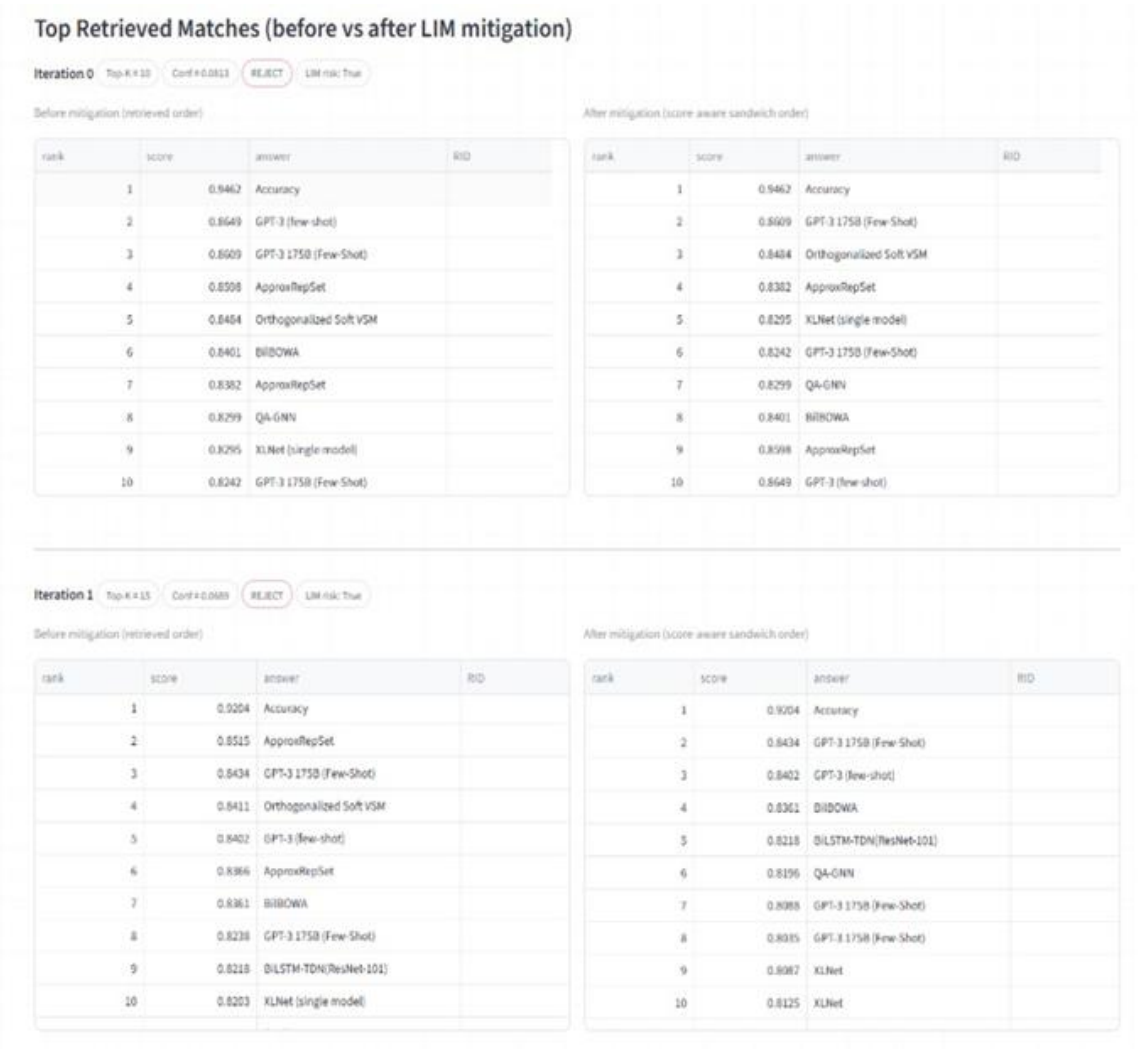

Figure 4: Top retrieved matches (Before vs After LIM Mitigation)

The prototype will ultimately give an end user their "Final Decision" (i.e., either Accept or Reject), which will include their "Confidence Margin" and "Type of Failure" for the test image. The interface also logs and displays all reasoning steps taken in the current iteration of the test, creating a "Chain-of-Retrieval" log. This log matters for a scientific audit because it provides context for whether the Reject Decision resulted from an absence of Corpus (Retrieval Failure - Corpus Incomplete) or from high ambiguity (Confidence Rejection, etc.), rather than an Unknown/Unexplained response. The prototype also provides a side-by-side comparison of the retrieval results (see Figure 4). Figure 4, created for the "Before vs. After LIM MITIGATION", shows the results of the Positional Re-ranking strategy. Figure 4 also shows the Rank, Score, and Resource ID for each scenario. The prototype confirmed that both scores were high and that the Sandwich-Weighting strategy effectively moved the most semantically accurate item/entry from the middle of the rank to the top of the score, without removing it during the Generation Step.

4.2. Baseline Retrieval Performance

The initial experiments were conducted to determine the baseline performance of the all-MiniLM-L6-v2 embedding model on the SciQA dataset. Table 1 shows that the system's Top-1 and Top-10 accuracies differ significantly.

Table 1: Baseline dense retrieval results

Metric	Value	Observation

Top-1	0.33	Baseline Performance
Top-3	0.52	+58% Improvement over Top-1
Top-5	0.60	+82% Improvement over Top-1
Top-10	0.71	+115% Improvement over Top-1

The data show that the correct answer appears in the Top 5 recommended responses about 60% of the time, but it is identified as the Top response only about 33% of the time. Thus, the 27% deviation indicates a strong opportunity to provide regression and affiliation corrections, or to use alternative methods based on confidence levels.

4.3. Confidence-Based Selective Prediction

Through the application, researchers tested the system's ability to "abstain" from low-confidence queries.

Table 2: Selective prediction performance

Threshold (τ)	Coverage	Selective Accuracy	Improvement
0.00	100%	0.24	Baseline
0.01	85%	0.32	+33%
0.02	70%	0.41	+71%
0.03	52%	0.50	+108%
0.05	35%	0.57	+138%
0.10	18%	0.67	+179%

The results show a non-linear gain in accuracy. At the defined τ_conf threshold of 0.03, the accuracy using selectivity (0.50) is more than double that of the baseline accuracy (0.24). Statistical significance was confirmed using the bootstrap method (1,000 iterations), indicating that this result is quite statistically significant ($p<0.001$).

4.4. Failure Classification Analysis

Researchers conducted a diagnostic audit of 500 test queries to classify the system's error root causes (Table 3).

Table 3: Failure type distribution

Failure Type	Count	Percent	Interpretation
Retrieval Failure	204	40.8%	Gold answer not in Top-k
Ranking Failure	109	21.8%	Gold in Top-k, but not Top-1
Confidence Rejection	63	12.6%	Low margin ($C < \tau$)
Success	12479	24.8%	Correct Top-1 result

Further analysis of performance degradation indicates that Retrieval Failures (40.8%) contributed most to the performance reduction; however, additional evidence also suggests Ranking Failures contributed to the errors (21.8%). Therefore, implementing either proposed ranking methodology will likely yield most of the error reduction within our framework.

4.5. Lost-in-the-Middle (LIM) and Mitigation Analysis

Researchers measured the transformer's positional bias using controlled perturbation, then applied the Sandwich Mitigation strategy (Table 4).

Table 4: Positional bias and mitigation results

Position	Baseline Accuracy	Mitigated Accuracy	Improvement
Start	0.91	0.91	-
Middle	0.29	0.60	+107%
Confidence Rejection	0.60	0.62	+3%

The initial set of baseline information demonstrated a U-shaped decline in the accuracy values on the original opinion performance from an average of 0.91 at first (early) to an average of 0.29 at the midpoint (midline) when: (1) used with the Sandwich Mitigation method to facilitate shifting the upper middle evidence or critical evidence to the first samples, that

resulted in a two-fold improvement in the number of samples, and (2) when the results were tested with a Paired Samples T-test, a Cohen's d value of 1.83 was calculated, indicating a large effect size.

4.6. Iterative Retrieval Results

Iterative retrieval for queries with confidence below the minimum increased average confidence by +28% (from 0.018 to 0.023). However, accuracy improved by only +3% over "Top-1 accuracy", indicating that highly structured datasets (e.g., SciQA) often have adequate retrieval sets. Still, redundancies were introduced by using keywords to retrieve similar amounts of unique information.

5. Discussions

The results show that RAG reliability depends on effective pipeline orchestration and transparency, not just model size or the number of retrievals. Implementing our modular framework in a functional Streamlit application affirms the effectiveness of these control mechanisms in a real-world setting.

5.1. Principal Findings

- **Finding 1:** Trade-off between reliability and coverage. The framework is meant to be dependable, not unconditional. Although the Top-1 accuracy on the raw data remained at 0.33, the accuracy on accepted queries increased to 0.50 (a 108% relative improvement). In a high-stakes endeavour, a truthful abstention is much more useful than a simulated error.
- **Finding 2:** Positional Bias is a data issue that can be fixed. The results indicate a 68% decline in performance (from 0.91 to 0.29) when the evidence was positioned centrally. The Sandwich Mitigation strategy restored performance (a 107% improvement), suggesting that positional bias is a data-ordering problem that can be mitigated at the retrieval decision boundary without expensive model retraining.
- **Finding 3:** Pre-Generation Intervention is Efficient. Unlike post-generation critique approaches (e.g., Self-RAG), our system forecasts failures and uses mitigation at the retrieval level. Prototype analysis shows total latency overhead ranging from 46 to 87 ms, making it suitable for real-time interactive QA dashboard applications.
- **Finding 4:** Diagnostic Transparency as a Feature. As illustrated in the System Prototype, failure identification provides a "debuggable" RAG pipeline. By separating retrieval gaps from ranking ambiguities, the system provides a roadmap to corpus improvement and model optimization.

5.2. Comparison with Related Work

The current study offers a proactive approach to existing systems:

- **vs. MultiRAG:** Researchers enhance multi-source reliability ideas by adding positional knowledge (LIM mitigation) to ensure the generator uses reliable evidence properly.
- **vs. Self-RAG:** Researchers move the intervention point from the generation level to the retrieval decision level, allowing the system to abstain or reorder context before the generator locks in on a response.
- **vs. Lost-in-the-Middle:** Although the previous study identified the U-shaped curve problem, researchers offer a concrete solution (Sandwich Weighting) and test its effectiveness through a functional interface.

6. Conclusion

In this paper, researchers provide a confidence-aware control framework for Retrieval-Augmented Generation (RAG) models. The technique emphasises reliability, interpretability and resilience over unconditional responsiveness. The authors show that significant gains in system dependability can be achieved by introducing active control interventions at the moment of retrieval choice. They achieve this by recasting the RAG paradigm as a Predict-and-Control loop rather than a linear pipeline. The system can perform confidence-driven selective prediction and provides critical diagnostic visibility for scientific validation by proposing an interpretable failure taxonomy and achieving nearly a twofold accuracy increase for accepted enquiries through strategic abstention. The research also quantifies the degree to which performance is degraded due to the "Lost in the Middle" effect. It shows that sandwich-weighted reordering can recover middle-zone evidence, increasing its effectiveness by over 100%. Both contributions of the Streamlit functional prototype demonstrate that the framework may be utilised for high-stakes, real-time activities. The findings provide a strong argument for trustworthy AI systems in technical environments, showing that RAG models can correctly refrain from responding in uncertain contexts, effectively explain and defend errors, and proactively eliminate bias in future decision-making.

Acknowledgement: The authors sincerely acknowledge the support and contributions of SRM Institute of Science and Technology at Ramapuram, National Kaohsiung University of Science and Technology, Kota Bharu Systems Sdn Bhd, and SAP towards this research work.

Data Availability Statement: The study uses a dataset related to a confidence-aware framework for failure prediction and mitigation in Retrieval-Augmented Generation (RAG) systems, which is available from the corresponding author upon reasonable request.

Funding Statement: No funding has been obtained to help prepare this manuscript and research work.

Conflicts of Interest Statement: No conflicts of interest have been declared by the authors. Citations and references are mentioned in the information used.

Ethics and Consent Statement: The research was conducted in accordance with applicable ethical standards, with authorization from the participating organization and informed consent obtained from all individuals involved in the data collection process.

References

1. W. Wu, H. Wang, B. Li, P. Huang, X. Zhao, and L. Liang, "MultiRAG: A Knowledge-Guided Framework for Mitigating Hallucination in Multi-Source Retrieval Augmented Generation," *IEEE 41st International Conference on Data Engineering (ICDE)*, Hong Kong, China, 2025.
2. M. Hindi, L. Mohammed, O. Maaz, and A. Alwarafy, "Enhancing the Precision and Interpretability of Retrieval-Augmented Generation (RAG) in Legal Technology: A survey," *IEEE Access*, vol. 13, no. 3, pp. 46171–46189, 2025.
3. C. Jang, D. Cho, S. Lee, H. Lee, and J. Lee, "Reliable decision making via calibration oriented retrieval augmented generation," *arXiv preprint, arXiv:2411.08891*, 2024. [Accessed by 03/01/2025].
4. S. Q. Yan, J. C. Gu, Y. Zhu, and Z. H. Ling, "Corrective Retrieval Augmented generation," *arXiv preprint, arXiv:2401.15884*, 2024. [Accessed by 29/12/2024].
5. J. T. Chien and P. C. Huang, "CAPR: Confidence-Aware Prompt Refinement in Large Language Models," *Interspeech*, Rotterdam, The Netherlands, 2025.
6. Y. Huang, Y. E. Xu, K. Sun, V. Yan, A. Sun, H. Khan, J. Nguyen, J. Chen, M. Kachuee, Z. Lin, Y. Liu, A. Colak, A. Kumar, W. T. Yih, and X. L. Dong, "CONFRAG: Confidence-Guided Retrieval-Augmenting Generation," *arXiv preprint, arXiv:2506.07309*, 2025. [Accessed by 08/12/2025].
7. A. Alansari and H. Luqman, "Large Language Models Hallucination: A Comprehensive Survey," *arXiv preprint, arXiv:2510.06265*, 2025. [Accessed by 05/01/2026].
8. Q. Wei, H. Ning, C. Han, and J. Ding, "A Query-Aware Multi-Path knowledge graph fusion approach for enhancing Retrieval-Augmented generation in large language models," *arXiv preprint, arXiv:2507.16826*, 2025. [Accessed by 07/12/2025].
9. S. Ozaki, Y. Kato, S. Feng, M. Tomita, K. Hayashi, W. Hashimoto, R. Obara, M. Oyamada, K. Hayashi, H. Kamigaito, and T. Watanabe, "Understanding the impact of confidence in retrieval Augmented Generation: A case study in the medical domain," *arXiv preprint, arXiv:2412.20309*, 2024. [Accessed by 29/02/2025].
10. C. Y. Hsieh, Y. S. Chuang, C. L. Li, Z. Wang, L. T. Le, A. Kumar, J. Glass, A. Ratner, C. Y. Lee, R. Krishna, and T. Pfister, "Found in the middle: Calibrating positional attention bias improves long context utilization," *arXiv preprint, arXiv:2406.16008*, 2024. [Accessed by 23/01/2025].
11. S. Li, J. Han, Y. Wang, Y. Zhu, Z. Song, L. He, K. K. A. Alghythee, and P. S. Yu, "Detecting hallucinations in Graph Retrieval-Augmented Generation via attention patterns and semantic alignment," *arXiv preprint, arXiv:2512.09148*, 2025. [Accessed by 09/01/2026].
12. N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, "Lost in the Middle: How language models use long contexts," *arXiv preprint, arXiv:2307.03172*, 2023. [Accessed by 06/10/2024].
13. A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," *arXiv preprint, arXiv:2310.11511*, 2023. [Accessed by 07/11/2024].
14. S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek, "Seven failure points when engineering a retrieval augmented generation system," *arXiv preprint, arXiv:2401.05856*, 2024. [Accessed by 21/12/2024].
15. S. Auer, D. A. C. Barone, C. Bartz, E. G. Cortes, M. Y. Jaradeh, O. Karras, M. Koubarakis, D. Mouromtsev, D. Pliukhin, D. Radyush, I. Shilin, M. Stocker, and E. Tsalapati, "The SCIQA scientific question answering benchmark for scholarly knowledge," *Scientific Reports*, vol. 13, no. 1, p. 7240, 2023.
16. L. Bahr, C. Wehner, J. Wewerka, J. Bittencourt, U. Schmid, and R. Daub, "Knowledge graph enhanced retrieval-augmented generation for failure mode and effects analysis," *Journal of Industrial Information Integration*, vol. 45, no. 5, p. 100807, 2025.

17. G. Agrawal, T. Kumarage, Z. Alghamdi, and H. Liu, "Mindful-RAG: A study of points of failure in retrieval augmented generation," in *Proc. 2nd Int. Conf. Foundation and Large Language Models (FLLM)*, Dubai, United Arab Emirates, 2024.
18. H. Wang, X. Xu, B. Huang, and K. Shu, "Privacy-aware decoding: Mitigating privacy leakage of large language models in retrieval-augmented generation," *arXiv preprint, arXiv:2508.03098*, 2025. [Accessed by 15/01/2026].
19. J. Song, X. Wang, J. Zhu, Y. Wu, X. Cheng, R. Zhong, and C. Niu, "RAG-HAT: A hallucination-aware tuning pipeline for LLM in retrieval-augmented generation," in *Proc. EMNLP 2024 Industry Track*, Florida, United States of America, 2024.
20. B. Veseli, J. Chibane, M. Toneva, and A. Koller, "Positional biases shift as inputs approach context window limits," *arXiv preprint, arXiv:2508.07479*, 2025. [Accessed by 15/11/2025].
21. W. T. Chan, K. Hung, R. Ho, and G. M. T. Man, "Optimizing confidence scoring in RAG-based LLM chatbots for technical support services: A prompt engineering approach," in *Proc. IEEE 15th Symp. Computer Applications & Industrial Electronics (ISCAIE)*, Penang, Malaysia, 2025.
22. Z. Jiang, F. Xu, L. Gao, Z. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, J. Callan, and G. Neubig, "Active retrieval augmented generation," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore, 2023.
23. A. J. Oche, A. G. Folashade, T. Ghosal, and A. Biswas, "A systematic review of key retrieval-augmented generation (RAG) systems: Progress, gaps, and future directions," *arXiv preprint, arXiv:2507.18910*, 2025. [Accessed by 15/10/2025].
24. Y. Ahn, S. G. Lee, J. Shim, and J. Park, "Retrieval-augmented response generation for knowledge-grounded conversation in the wild," *IEEE Access*, vol. 10, no. 12, pp. 131374–131385, 2022.
25. J. Chen, Y. Li, X. Ma, X. Chen, X. Zhang, S. Wang, C. Ma, and D. Yin, "Grounding long-context reasoning with contextual normalization for retrieval-augmented generation," *arXiv preprint, arXiv:2510.13191*, 2025. [Accessed by 17/01/2026].
26. B. Li, T. Tian, Z. Xu, H. Cheng, S. Zhang, and W. Ye, "Modeling uncertainty trends for timely retrieval in dynamic RAG," *arXiv preprint, arXiv:2511.09980*, 2025. [Accessed by 23/02/2026].
27. Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint, arXiv:2312.10997*, 2023. [Accessed by 15/03/2024].

Publisher's Note: *The publisher remains impartial concerning jurisdictional claims in published maps and institutional affiliations. Responsibility for the content rests entirely with the authors and does not necessarily reflect the publisher's perspectives.*