



IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY



VOLUME : 11 ISSUE : 05 Print / Issue Publication Date: September 2026



ISSN : 2455-2143



Indexed In



WWW.IJEAST.COM

editor@ijeast.com



A SURVEY OF HALLUCINATION MITIGATION STRATEGIES IN LARGE LANGUAGE MODELS: FROM DECODING-TIME FIXES TO AGENTIC RETRIEVAL AND VERIFICATION

Shweta Narayan Bamne, Prof. M.S. Shegokar
Department of Computer Science and Engineering
College of Engineering and Technology, Akola, India

Abstract—Hallucination – the generation of fluent but factually unsupported or fabricated content – remains one of the most persistent obstacles to deploying Large Language Models (LLMs) in high-stakes domains such as healthcare, finance, and law. A wide range of mitigation strategies has been proposed, spanning several distinct points of intervention: constraining or calibrating the decoding process itself, fine-tuning or aligning models to prefer grounded outputs, augmenting generation with external retrieval, and verifying or correcting outputs after they are produced. This paper surveys hallucination mitigation across these four intervention points rather than focusing narrowly on any single pipeline. We discuss the underlying causes of hallucination identified in prior work, review representative techniques at each intervention point – including confidence based decoding constraints, instruction-tuning for faithfulness, hybrid sparse-dense retrieval with rank fusion, cross-encoder re-ranking, and Natural Language Inference (NLI)-based claim verification within agentic architectures – and compare the costs and guarantees each category offers. We further examine evaluation practices and benchmarks used to measure hallucination and identify a set of open problems, including the absence of standardized cross-category benchmarks, the tension between validation strictness and unnecessary refusal, and the limited maturity of hallucination mitigation for multi-hop and cross document reasoning. We conclude with directions for future work that combines lightweight decoding-time safeguards with heavier, verification-based agentic pipelines depending on application risk.

Index Terms—Hallucination Mitigation, Large Language Models, Retrieval-Augmented Generation, Agentic Systems, Response Validation, Natural Language Inference, Decoding Strategies, Survey.

I. INTRODUCTION

Large Language Models (LLMs) have achieved strong performance across question answering, summarization, and decision-support tasks, and are increasingly being deployed in domains such as healthcare, finance, education, and legal support [1]. A persistent obstacle to this deployment is hallucination: the tendency of a model to produce output that is fluent and confident in tone but factually incorrect, unsupported by any source, or entirely fabricated. Unlike a simple parsing error, a hallucinated claim is often indistinguishable from a correct one at the surface level, which makes it especially dangerous in settings where a user may reasonably trust the model's tone as a proxy for its accuracy.

Research on mitigating hallucination has grown rapidly, but it has grown in several largely separate directions rather than a single unified line of work. Some approaches intervene at decoding time, constraining or re-weighting the token generation process itself. Others intervene at training time, fine-tuning or aligning the model to prefer grounded, faithful completions. A third and heavily studied direction augments generation with external retrieval, grounding the model in documents rather than relying solely on parametric memory. A fourth, more recent direction adds post-hoc verification, checking a model's output against evidence after it has been generated and triggering correction or refusal when support is lacking.

Much existing survey work, including work closely tied to specific agentic Retrieval-Augmented Generation (RAG) pipelines, focuses primarily on the retrieval-and-verification end of this spectrum [7]. This is a reasonable focus given how actively that area is being developed, but it leaves the broader picture – how retrieval-and-verification approaches compare to decoding-time and training-time interventions in terms of cost, coverage, and applicability – less thoroughly examined. This paper aims to fill that gap by surveying hallucination mitigation across all four intervention points, situating agentic RAG and validation architectures as one important category within a larger design space rather than as the sole solution.



A. Scope and Contributions

This paper does not propose a new system. It synthesizes existing hallucination-mitigation literature into a taxonomy organized by where mitigation is applied in the generation pipeline, and makes the following contributions:

- A four-category taxonomy of hallucination mitigation – decoding-time, training-time, retrieval-based, and verification-based interventions – that spans a broader design space than pipeline-specific surveys.
- A discussion of the underlying causes of hallucination that motivate each category of intervention.
- A comparative review of representative techniques within the retrieval-based and verification-based categories, including hybrid sparse-dense retrieval, Reciprocal Rank Fusion, cross-encoder re-ranking, and NLI-based claim checking, situated alongside lighter-weight decoding- and training-time alternatives.
- A discussion of evaluation metrics and benchmarks used across categories, and the difficulty of comparing techniques that intervene at different pipeline stages.
- An analysis of open challenges and future directions, including how categories might be combined depending on application risk tolerance.

B. Organization

Section II discusses the causes of hallucination identified in prior work. Section III surveys decoding-time and training time mitigation strategies. Section IV reviews retrieval-based grounding, including multi-stage and agentic retrieval architectures. Section V discusses post-hoc verification and validation techniques. Section VI compares evaluation practices across categories. Section VII discusses open challenges and future directions, and Section VIII concludes the paper.

II. CAUSES OF HALLUCINATION IN LARGE LANGUAGE MODELS

Before surveying mitigation strategies, it is useful to distinguish the causes of hallucination that different strategies target, since this explains why no single category of intervention has proven sufficient on its own [15].

A. Parametric Knowledge Gaps

LLMs encode knowledge implicitly in their parameters at training time. When a query concerns facts absent from, underrepresented in, or contradicted by the training data – including anything after the model’s knowledge cutoff – the model may still generate a confident-sounding answer rather than signaling uncertainty. This cause primarily motivates retrieval-based mitigation, which supplies missing knowledge externally rather than relying on parametric memory alone [8].

B. Exposure Bias and Decoding Dynamics

During generation, small early errors can compound: once a model commits to an incorrect token, subsequent tokens are often generated to remain consistent with that error rather than to correct it. This is a property of autoregressive decoding itself, independent of what the model knows, and motivates decoding-time interventions that monitor and constrain the generation process.

C. Misaligned Training Objectives

Standard language-model training objectives reward fluent, plausible continuations, not calibrated uncertainty. Reinforcement-learning-based alignment (e.g., RLHF) can further reward confident, satisfying-sounding answers over hedged but more honest ones if reward models are not specifically designed to penalize unsupported claims [13]. This motivates training-time interventions that explicitly optimize for faithfulness.

D. Lack of Post-Generation Verification

Even when the causes above are individually mitigated, no generation process is perfectly reliable; some hallucinated content will still be produced. Absent an explicit verification step, such content passes to the user unchecked. This motivates the verification-based approaches discussed in Section V, several of which are increasingly embedded in agentic RAG architectures [3], [4].

III. DECODING-TIME AND TRAINING-TIME MITIGATION

A. Decoding-Time Interventions

Decoding-time methods intervene without modifying model weights or requiring external retrieval, which makes them comparatively cheap to deploy. Common strategies include contrastive decoding, which compares outputs from a stronger and weaker version of a model (or from different layers) to suppress tokens associated with hallucinated continuations [11], and self-consistency methods, which sample multiple generations for the same query and treat disagreement between samples as a proxy for uncertainty [12]. Confidence-scoring approaches, discussed further in Section V in their post-hoc form, can also be applied at the token level during decoding to flag low-confidence spans for the user [10].

The main advantage of decoding-time methods is low overhead: they require no additional infrastructure such as a retrieval index. Their main limitation is that they cannot supply knowledge the model never had access to; they can suppress confident wrong answers but cannot turn a wrong answer into a correct one when the correct fact is simply outside the model’s parametric knowledge.



B. Training-Time and Alignment-Based Interventions

Training-time methods fine-tune a base model to prefer faithful, well-grounded completions. This includes supervised fine-tuning on datasets that pair questions with citations or evidence, and reinforcement-learning approaches such as reinforcement learning from human feedback (RLHF), which reward answers verified against a knowledge source and penalize unsupported claims [13]. Some approaches further train models to explicitly abstain when they lack sufficient confidence, rather than defaulting to producing an answer.

Training-time methods can generalize better than decoding time heuristics, since faithfulness becomes part of what the model has learned rather than an external constraint applied at inference time. However, they are costly to apply (requiring curated faithfulness-labeled data and retraining), and a model's static training data still cannot cover facts that change or are introduced after training – the same underlying gap that motivates retrieval-based grounding.

IV. RETRIEVAL-BASED GROUNDING

A. Standard Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) addresses the parametric-knowledge-gap cause of hallucination directly, by retrieving relevant passages from an external knowledge base and conditioning generation on that evidence rather than on parametric memory alone [8], [14]. This allows the knowledge available to the model to be updated by editing the retrieval corpus rather than retraining, and it can supply facts about entities or events entirely absent from training data.

Single-pass RAG, however, has well-documented limitations: a single retrieval step may fail to surface all evidence relevant to a complex query, retrieved passages are not always filtered for relevance or mutual consistency, and there is typically no mechanism to check whether the eventually generated answer is actually supported by what was retrieved [8].

B. Multi-Stage and Agentic Retrieval

To address these limitations, more recent work reframes retrieval as an iterative, agent-driven process rather than a single lookup. Kumar et al. show that agents capable of re-querying or re-planning mid-task reduce hallucinated content compared to static single-pass pipelines [1]. Nguyen et al. propose multi-agent retrieval frameworks in which separate agents dynamically adjust search behavior based on intermediate confidence signals, improving factual grounding on multi hop questions [2].

A common pattern in this literature combines sparse keyword-based retrieval (e.g., BM25) with dense semantic retrieval, merged using Reciprocal Rank Fusion (RRF), which combines ranked lists from multiple retrievers

without requiring their scores to be on comparable scales [8]:

$$\text{RRF}_{\text{score}}(d) = \sum_{i=1}^n \frac{1}{k + \text{rank}_i(d)} \quad (1)$$

where $\text{rank}_i(d)$ is the rank assigned to document d by retriever i , and k (commonly 60) discounts the influence of very low ranks. Query reformulation is often layered on top of this: if initial retrieval yields sparse or low-confidence results, the query is decomposed or rephrased and retrieval repeats, turning a one-shot lookup into a bounded loop.

C. Re-Ranking as a Filtering Stage

Hybrid retrieval typically over-retrieves relative to what a generator can usefully condition on. Arya et al. survey cross-encoder re-ranking, in which a query and candidate document are jointly encoded so the model can capture finegrained lexical and semantic interactions between them, unlike the independent encoding used in dense retrieval [6]. This improves precision at the top of the ranked list at the cost of additional per-document computation, which is why re-ranking is usually applied only to the top- N candidates from a cheaper first-stage retriever rather than to the full candidate pool.

V. POST-HOC VERIFICATION AND VALIDATION

A. Why Retrieval Alone Is Insufficient

Even with strong retrieval and re-ranking, a generator can still produce claims not actually entailed by the retrieved evidence, either by over-generalizing from a passage or by blending retrieved facts with parametric knowledge. Pisano et al. and Le et al. argue that this class of error requires explicit, post-hoc verification, since it cannot be reliably prevented by improving retrieval quality alone [3], [4].

B. Claim-Level NLI Verification

A recurring verification strategy decomposes a generated answer into smaller, independently checkable factual claims and evaluates each against retrieved evidence using a Natural Language Inference (NLI) model, classifying each claim evidence pair as entailed, contradicted, or neutral [9]. A response is treated as reliable only if all, or a sufficient proportion, of its claims are entailed; contradicted or neutral claims trigger correction or refusal.

C. Confidence Scoring and Feedback-Driven Correction

Williams et al. describe confidence-scoring methods that combine validation outcomes, retrieval quality, and model internal signals into a single score used to accept, revise, or reject a response, allowing downstream applications to set domain-specific acceptance thresholds



[10]. Sarpatwar et al. propose feedback loops in which a failed validation is passed back to the generator, which attempts a bounded number of corrections using only the originally retrieved evidence before the system falls back to a safe refusal [5].

D. Agentic Integration of Verification

What distinguishes the most recent systems from earlier RAG-plus-verification pipelines is that verification is integrated as one agent among several – alongside query planning, retrieval, and re-ranking agents – rather than bolted on as a separate final step. This allows verification failures to trigger targeted re-retrieval or re-planning rather than only regeneration from the same evidence. Le et al. additionally emphasize that transparent validation – surfacing which claims were checked and why they passed or failed – matters for building user trust in such systems, independent of raw accuracy gains [4].

VI. EVALUATION PRACTICES ACROSS CATEGORIES

TABLE I: Mitigation Categories: Typical Cost and Primary Limitation

Category	Typical Cost	Primary Limitation
Decoding time	Low (inference only)	Cannot supply missing knowledge
Training-time	High (data + retraining)	Static; needs retraining to update
Retrieval based	Medium (index + retrieval)	No guarantee output matches evidence
Verification based	Medium-High (extra model calls)	Added latency; risk of over-refusal

Because these categories intervene at different points, they are rarely evaluated on shared benchmarks, which makes cross-category comparison difficult. Retrieval- and verification-based systems are typically evaluated using hallucination rate, factual accuracy, answer correctness, contextual relevance, response latency, and refusal rate [7], [9]. Decoding-and training-time methods are more often evaluated using calibration metrics (how well a model’s confidence matches its actual correctness) and benchmark-specific accuracy, which are not always reported alongside the metrics used for retrieval based systems. Dhamange et al. note that, within retrieval-and verification systems specifically, hallucination rate and refusal rate should be reported together, since a system can trivially minimize hallucination by refusing difficult questions [7].

VII. OPEN CHALLENGES AND FUTURE DIRECTIONS

A. No Shared Benchmark Across Categories

The lack of a common evaluation framework across decoding-time, training-time, retrieval-based, and verification based methods makes it difficult to answer a practical question many practitioners face: for a given application and latency budget, which category of intervention offers the best hallucination reduction per unit of added cost.

B. Latency and Computational Overhead

Retrieval- and verification-based methods, particularly agentic pipelines with multi-stage retrieval and claim-level NLI checking, add substantial computation relative to decoding time methods. Caching, batching, and smaller re-ranking or verification models mitigate but do not eliminate this overhead.

C. Over-Refusal in Verification-Based Systems

Strict validation thresholds reduce hallucination but can cause a system to refuse questions it could answer correctly, particularly when evidence is phrased differently from the claim or requires light inference beyond strict entailment.

D. Multi-Hop and Cross-Document Reasoning

Claim decomposition and NLI-based checking work well when a claim can be verified against a single passage, but are less mature for claims requiring synthesis across multiple documents or multi-step inference – a limitation noted across several of the surveyed systems.

E. Toward Combined, Risk-Adaptive Pipelines

A promising direction is combining categories according to application risk rather than treating them as competitors: lightweight decoding-time or training-time safeguards for low stakes use cases, escalating to full agentic retrieval-and verification pipelines only for high-stakes queries where the cost of a hallucination is high enough to justify the added latency. Adaptive threshold-setting methods that require less domain-specific tuning data would make this kind of escalation more practical to deploy.

VIII. CONCLUSION

This paper surveyed hallucination mitigation in large language models across four intervention points – decoding-time, training-time, retrieval-based, and verification-based – rather than focusing narrowly on any single pipeline. We discussed the distinct causes of hallucination each category targets, reviewed representative techniques within each, and compared their typical costs and limitations. Within the retrieval-and verification space specifically, we reviewed hybrid sparse dense retrieval with rank fusion, cross-encoder re-ranking, and NLI-based claim verification embedded in agentic architectures. Open challenges remain



around cross-category benchmarking, latency, over-refusal, and multi-hop reasoning; addressing these is likely to require combining lightweight and heavyweight mitigation strategies according to the risk tolerance of the application, rather than relying on any single category alone.

REFERENCES

- [1]. Kumar A., Finley S., and Thompson R. (2026). Agentic Retrieval Augmented Generation for Hallucination Mitigation in Large Language Models, arXiv preprint, DOI: arXiv:2603.07379.
- [2]. Nguyen T., Chen M., and Zhang L. (2025). Multi-Agent Frameworks for Factual Accuracy in LLM-Based Question Answering, arXiv preprint, DOI: arXiv:2501.09136.
- [3]. Pisano S., Patterson J., and Williams K. (2025). Hallucination Detection and Correction in Retrieval-Augmented Language Models, arXiv preprint, DOI: arXiv:2510.25518.
- [4]. Le H., Truong T., and Nguyen D. (2025). Explainable AI in Agentic Retrieval-Augmented Generation: A Comprehensive Framework, arXiv preprint, DOI: arXiv:2506.21931.
- [5]. Sarpatwar K., Mehta R., and Sharma A. (2024). Self-Correcting RAG Pipelines with Iterative Feedback Loops, Hugging Face Papers, DOI: arXiv:2404.08189.
- [6]. Arya V., Dave P., and Marathe M. (2024). Cross-Encoder Re-Ranking for Improved Document Selection in RAG Systems, arXiv preprint, DOI: arXiv:2402.10612.
- [7]. Dhamange N., Somy S., and Deshmukh R. (2025). Towards Explainable AI in Agentic Retrieval-Augmented Generation: A Systematic Review, ResearchGate, Publication No. 397518736.
- [8]. Johnson M., Lee P., and Kumar S. (2025). Multi-Stage Retrieval Architectures for Knowledge-Intensive NLP Tasks, in IEEE Transactions on Knowledge and Data Engineering, vol. 36, no. 8, (pp.4120-4135).
- [9]. Patel R., Singh A., and Gupta N. (2024). Natural Language Inference for Fact Verification in High-Stakes RAG Applications, in IEEE Access, vol. 12, (pp.107692-107708).
- [10]. Williams T., Brown S., and Davis L. (2025). Confidence Scoring and Uncertainty Estimation for Hallucination Detection in LLM-Generated Content, in IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 11, (pp.111830-111845).
- [11]. Li X.L., Holtzman A., Fried D., Liang P., Eisner J., Hashimoto T., Zettlemoyer L., and Lewis M. (2022). Contrastive Decoding: Open-Ended Text Generation as Optimization, arXiv preprint, DOI: arXiv:2210.15097.
- [12]. Wang X., Wei J., Schuurmans D., Le Q., Chi E., Narang S., Chowdhery A., and Zhou D. (2022). Self-Consistency Improves Chain of Thought Reasoning in Language Models, arXiv preprint, DOI: arXiv:2203.11171.
- [13]. Ouyang L., Wu J., Jiang X., Almeida D., Wainwright C., Mishkin P., Zhang C., Agarwal S., Slama K., Ray A., Schulman J., Hilton J., Kelton F., Miller L., Simens M., Askell A., Welinder P., Christiano P., Leike J., and Lowe R. (2022). Training Language Models to Follow Instructions with Human Feedback, in Advances in Neural Information Processing Systems 35 (NeurIPS 2022).
- [14]. Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N., Kuttler H., Lewis M., Yih W., Rocktaschel T., Riedel S., and Kiela D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks, in Advances in Neural Information Processing Systems 33 (NeurIPS 2020).
- [15]. Ji Z., Lee N., Frieske R., Yu T., Su D., Xu Y., Ishii E., Bang Y., Madotto A., and Fung P. (2023). Survey of Hallucination in Natural Language Generation, in ACM Computing Surveys.

IJEAST

INTERNATIONAL JOURNAL OF ENGINEERING APPLIED SCIENCE AND TECHNOLOGY



AUTHORS

 Shweta Narayan Bamne

 Prof. M.S. Shegokar



ABOUT IJEAST

International journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high - quality research paper in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

JOURNAL METRICS



H-INDEX

33



i10-INDEX

154



IJEAST
CITATIONS

9,647

(IJEAST CITATIONS)



PEER-REVIEWED
& OPEN ACCESS

PUBLISHED
MONTHLY



PEER REVIEWED

All submission are rigorously peer reviewed to ensure quality.



AUTHORS ARE OUR PRIORITY

We value our authors and recognize their contribution to the advancement of research and knowledge.



OPEN ACCESS

Free and unrestricted access to research for all.



GLOBAL REACH

Connecting researchers and professionals worldwide.



TIMELY PUBLICATION

We ensure a swift and efficient publication process.



For more information, visit our website

www.ijeast.com



editor@ijeast.com



www.ijeast.com



India



2455-2143