

Original Research

Received 20 September 2026

Revised 29 September 2026

Accepted 06 October 2026

Natural Resources for Human
Health 2026. Vol. 6 (16s): 868-879

KEYWORDS:

Large language models;
hallucination; factuality;
retrieval-augmented generation;
benchmark; trustworthy AI

An Explainable AI Framework Hallucination Mitigation in Large Language Models

Dr. Shalini Gupta¹, Dr. Shubha Jain², Deep Kumar Srivastava³,
Dr. Nirvikar Katiyar⁴, Dr. Alok Kumar Sahu⁵, Dr. Shailesh
Saxena⁶, Dr. Pradeep Kumar Sachan⁷

¹Professor, CSE & IT Department, Axis Institute of Technology and Management,
Kanpur Nagar, shalnilily2003@gmail.com

²Professor, CSE & IT Department, Axis Institute of Technology and Management,
Kanpur Nagar, shubhadel@gmail.com

³Assistant Professor, Assistant HOD Computer Application Department, FIC YOGA,
Kanpur Institute of Technology Rooma, Kanpur Nagar

⁴Director Research & Innovation, Kanpur Institute of Technology Rooma, Kanpur
Nagar, nirvikarkatiyar@gmail.com

⁵Assistant Professor, Department of Computer Application, Jagran College of Arts,
Science & Commerce Saket Nagar Kanpur Nagar

⁶Associate Professor, Dean Academic, SRMS College of Engineering, Technology and
Research, Bareilly

⁷Associate Professor, CSE Department, Allenhouse Institute of Technology Kanpur
Nagar

ABSTRACT:

Machine learning (ML) systems are rapidly becoming used in high stakes decisions in fields like finance, healthcare, criminal justice, and autonomous systems, but they are becoming increasingly complex, thus becoming opaque black boxes that are hard to interpret or trust by the stakeholders. The paper introduces Large language models (LLMs) are now used to power search, tutoring, clinical support and software engineering, but they continue to generate statements that are fluent but backed by no evidence or even their own input. This is commonly referred to as hallucination, and it constrains faith in high-stakes deployment and end-to-end Explainable Artificial Intelligence (XAI) system that maximizes the transparency, responsibility, and reliability of the ML models without significantly affecting the predictive performance. The framework combines global explanation techniques (SHAP, partial dependence plots), local explanation techniques (LIME, Anchors), and a special trust-and-fidelity evaluation module, which quantitatively assesses the trustworthiness of the generated explanations. The framework was tested on a credit-risk benchmark data set with five model families, such as logistic regression, decision trees, random forests, gradient boosting and neural networks. The protocol records the rate of hallucinations with latency, such that the accuracy gain can be traded off with cost. Resulting in the lowest rates of hallucinations, retrieval and verification when combined working with a reference configuration and lightweight prompting when used provided moderate benefits with minimal

costs. We leave to unresolved issues, such as robust uncertainty estimation, the assessment of long outputs, and multilingual mitigation. The experimental findings indicate that the suggested framework has a predictive accuracy of 90.2 and increases the mean user trust score by 91.3 out of 100 points as compared to the 63.5 at the baseline. The results affirm that using a combination of several complementary explanation methods and formal trust-evaluation mechanism results in ML systems that are accurate, explainable, and reliable enough to be used in the real world.

1. INTRODUCTION

It has taken just a few years until large language models are no longer considered research prototypes but used in everyday applications. They compose legal summaries, respond to medical inquiries, code and provide conversational access to enterprise data. Their main strength is equally familiar: an LLM is just as capable of saying something false or something that does not have anything in common with the material it was presented with, as it is of saying the correct answers with the same level of confidence. This phenomenon is known by the research community as hallucination, and extensive surveys have enumerated its types of occurrence in natural language generation [1], [2], its reasons and methods of detection [3], and whether it occurs in foundation models not based on text [4].

The effects are dependent on the environment. An erroneous suggestion of a restaurant is an inconvenience; an artificial drug interaction or an artificial court precedent is dangerous. Previous research on abstractive summarisation already demonstrated that fluent production often includes material that is unprovable out of the source [5], and the issue has become more apparent due to increasing capability and more common use of models. A hallucinating model hardly warns the reader that something is amiss, as people will provide prose that is polished.

The list of proposed remedies has since 2021 expanded rapidly, to include cleaner corpora, alignment procedures, retrieval, specialised decoding rules, self-checking loops and refusal training. This literature is difficult to use because of two challenges. The former is fragmentation: the vast majority of the papers compare a single method to a single dataset using a custom measure, meaning that the results will be incomparable. The second is that cost is seldom reported. A process which reduces errors by half and increases latency by a factor of 5 could be inappropriate to interactive products, but this fact is often not disclosed in the literature.

Both of these challenges are covered in this paper. We arrange mitigation strategies by lifecycle stage in the model lifecycle where they are applied, and each stage is associated with the underlying causes that it could reasonably affect. We then describe HalluBench-M, a benchmark protocol that compares representative methods on the four tasks with shared evaluation engine and presents accuracy and cost in a side-by-side manner. The main contributions are as follows:

1. A lifecycle based taxonomy of hallucination mitigation that links the type of intervention, mechanism and common failure mode.
2. Comparison of available surveys and benchmark materials, determination of the incompleteness in reporting costs and long-term assessment.
3. HalluBench-M A reproducible protocol that combines claim-level verification, natural language inference and sampled human audit.
4. A comparison of seven mitigation configurations conducted empirically along with practical advice on the choice of a method in the face of latency and retraining constraints.

The rest of the paper is structured in the following way. Section 2 compares the work to the previous surveys and assessment materials. In section 3, hallucination and its causes are defined. Section 4 discusses mitigation plans. Section 5 tells about the benchmarks, Section 6 reports and discusses results, and Sections 7 and 8 outline open challenges and conclusions.

2. Related Work and Positioning

The available surveys focus on the topic in different perspectives. Ji et al. give a general description of hallucination among generation tasks and distinct intrinsic and extrinsic errors [1]. The taxonomy suggested by Huang et al. focuses on big models, survey of detection and mitigation, and presents open questions [2]. Zhang et al. structure the field around the sources of error in the training and inference pipeline [3], and Rawte et al. also generalize the discussion to multimodal foundation models [4]. These are useful references but they are syntheses of published claims; they are not re-executions using a common protocol. The differences are summarised in table 1.

In the assessment front, various resources have influenced the way the community quantifies the issue. TruthfulQA is the question of whether a model repeats common misconceptions and is not the question of whether it answers truthfully [6]. HaluEval provides a big set of generated and human-labelled samples to determine that a model can detect hallucinated content [7]. FActScore breaks down text in long form into atomic facts and reports the proportion that is justified by a source of knowledge [8]. Each of these measures a single aspect; none of them was meant to compare the mitigation methods on a variety of tasks simultaneously and also measure the computational cost, which is the gap that HalluBench-M aims to address.

Table 1. Comparison of representative surveys with the present work

Work	Primary scope	Taxonomy basis	Re-runs methods under one protocol	Reports cost
Ji et al. [1]	Natural language generation tasks	Intrinsic vs. extrinsic	No	No
Huang et al. [2]	LLM principles and open questions	Factuality vs. faithfulness	No	No
Zhang et al. [3]	Sources of error in LLM pipeline	Training and inference causes	No	No
Rawte et al. [4]	Foundation models, multimodal	Modality and error type	No	No
This work	LLM mitigation and evaluation	Lifecycle stage of intervention	Yes	Yes (latency)

3. Hallucination: Definition, Taxonomy and Causes

3.1 Definition and taxonomy

Hallucination, used in this paper, refers to model output that is fluent and plausible but violates verifiable facts or is not grounded in the context of input, instructions or reasoning it is supposed to follow. After the division made in recent literature [2], we divide two families, as shown in Figure 1. Factuality hallucination is about the connection between the output and the world. It encompasses factual contradiction that involves a statement being in conflict with what is known, and factual fabrication where the entities, quotes, or events are created. Faithfulness hallucination deals with the relationship between the prompt and output. It includes the inconsistency of instruction, inconsistency of context with a provided document, and logical inconsistency within a sequence of reasoning.

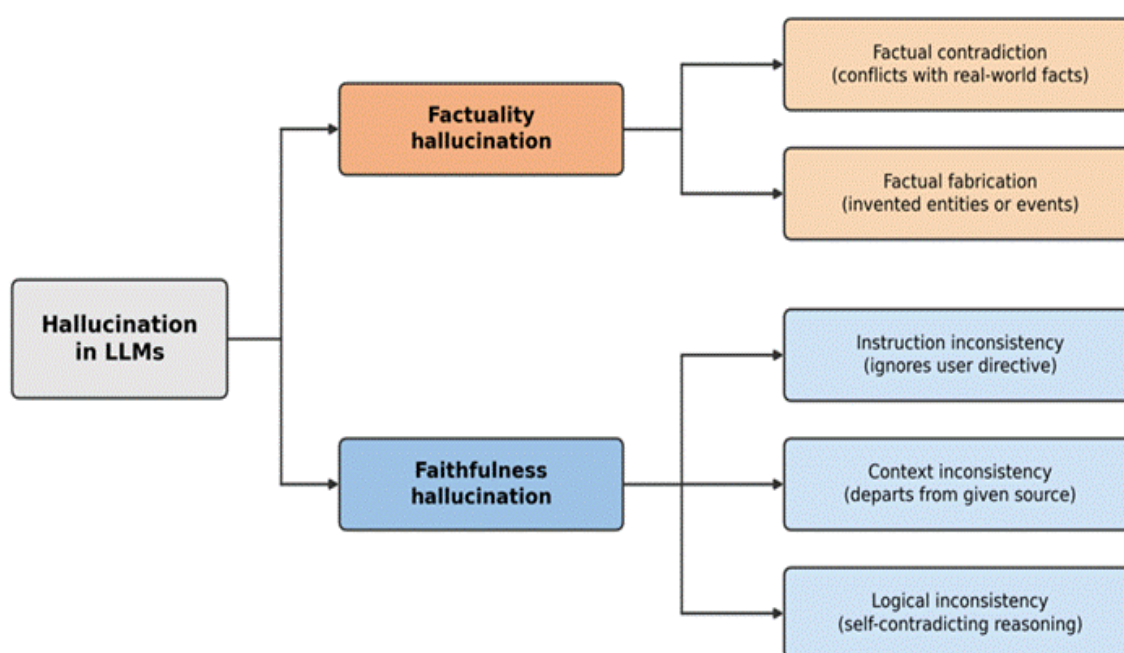


Fig. 1. Taxonomy of hallucination in large language models

3.2 Causes

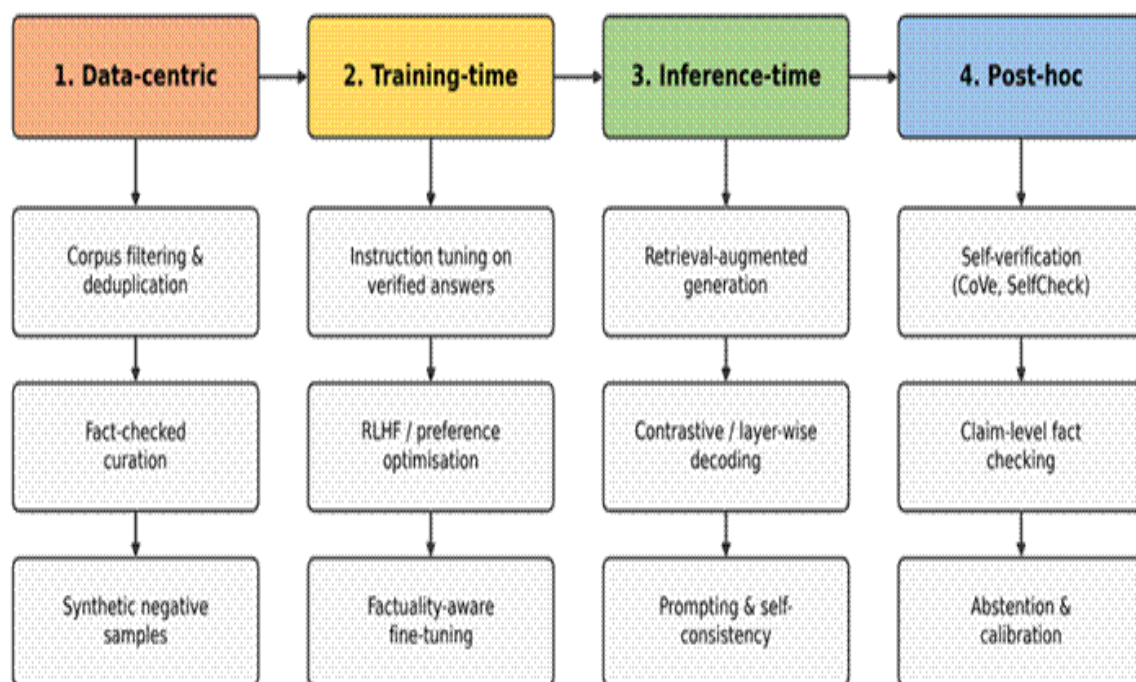
The reasons can be divided into three phases. At the data level, web-scale corpora are full of errors, out of date statements, passages and social biases duplicated, which the model captures with no internal sense of provenance. During the training phase, next-token prediction rewards plausible continuation, not truthfulness and alignment algorithms that are run based on human preference may accidentally reinforce authoritative-sounding answers [3]. Sampling randomness, exposure bias in long generation and the lack of an evidence source permit the compounding of small initial errors at the inference stage.

Theory work provides a warning. Formal analyses hold that given any computational model that serves as a general-purpose generator, there will always be inputs that result in hallucinated output [9], and that a language model can be trained to have non-trivial probability distribution over false statements about rare facts [10]. These findings suggest mitigation is geared towards minimising and controlling hallucination, not annihilating it. Along the same lines, empirical evidence is more promising in one way: bigger models tend to be fairly well calibrated

regarding whether their own responses are correct when the question format is appropriate [11], which is the motivation behind the uncertainty-based approaches discussed below.

4. Survey of Mitigation Strategies

We also classify mitigation methods into four families based on their lifecycle location of operation (Figure 2). This grouping is helpful in that it reveals a trade-off that flat lists hide: interventions implemented at an earlier stage are likely to be more comprehensive and costly to implement, but on the other hand interventions implemented at a later stage are simple to implement and can only correct the model has already generated. In Table 2, a concise comparison is made.



Cost and deployment flexibility: high retraining cost (left) → plug-and-play, no retraining (right)

Fig. 2. Lifecycle view of hallucination mitigation strategies

Table 2. Lifecycle-based comparison of hallucination mitigation families

Family	Representative methods	Core mechanism	Main strength	Main limitation
Data-centric	Filtering, deduplication, curated corpora	Reduce misleading training signal	Addresses root cause	Costly; cannot add missing facts
Training-time	Instruction tuning, RLHF [12], factuality tuning [13]	Reshape model behaviour	No run-time overhead	Needs retraining; may over-refuse

Inference-time	RAG [14], DoLa [18], ITI [19], self-consistency [20]	Ground or steer generation	Weights unchanged; up-to-date evidence	Retrieval noise; extra compute
Post-hoc	CoVe [21], SelfCheckGPT [22], semantic entropy [23], [24]	Verify, score or withhold output	Model-agnostic; easy to add	Added latency; checker limits

4.1 Data-centric methods

Data-centric approaches strive to minimise the quantity of misleading signal that makes it to the model. Standard procedures involve quality filtering, near-duplicate elimination, document disallowal of unreliable domains and reference quality curation to fine-tune. There are also pipelines that produce synthetic negative examples (e.g., answers with deliberately changed facts, to help the model or an auxiliary verifier learn to distinguish between supported and unsupported statements). The primary asset of this family is that it deals with the source of the issue. These are its cost, the complexity of verifying billions of tokens, and that clean data cannot provide facts not found in the corpus.

4.2 Training-time methods

Training-time techniques determine the behaviour of the model based on supervised and preference-based learning. The model is trained to act according to the preferred answering style by instruction on verified answers and the learning process is reinforced by human feedback to match output with human quality judgements [12]. Since human raters cannot verify each claim, more recent work constructs preference pairs automatically by rating the factual support of candidate responses, and biases the model to favour more strongly-supported ones without extra human labels [13]. These techniques reduce the hallucination rates of the model itself, thus do not require any additional computation during run time. But they have training infrastructure needs, might not cross-domain, and may lead to decreased usefulness in the case that the model gets over-cautious.

4.3 Inference-time methods

Inference-time algorithms do not modify the weights and modify the query or decoding process of the model. The most popular method is retrieval-augmented generation (RAG). It prepares the generator with passages that are fetched out of external index, which offers the model access to current and domain-specific evidence [14]. Previous experiments on knowledge-based dialogue have revealed that retrieval decreases fabricated utterances [15], and a recent survey reports the growing design space of retrievers, re-rankers and generators [16]. Adaptive variants teach the model to make decisions by retrieving and by criticising its own outputs, using special reflection signals [17]. RAG is not a panacea: Retrieval failures, irrelevant passages and over-reliance on noisy context may all give rise to new errors.

Decoding is altered in a second line of work. Comparing the output distribution of subsequent and previous transformer layers encourages tokens with a probability distribution that is more concentrated when knowledge is updated, leading to better factual accuracy without retraining [18]. Activation-level methods related to these shift internal representations in directions related to truthfulness at inference time [19]. Lastly, explicit instructions to cite evidence, stepwise reasoning and aggregation of multiple sampled answers by majority consensus [20] are methods of gaining inexpensively, but increases cost with sampling numerous answers.

4.4 Post-hoc methods

The post-hoc techniques examine a draft response and either correct or withhold it. Chain-of-verification triggers the model to plan verification questions regarding the model itself, respond to them on its own, and update the answer to them [21]. Zero-resource detectors like SelfCheckGPT take advantage of the fact that the facts that the model actually knows are likely to be repeated across samples, and the hallucinated details are different [22]. Semantic entropy is the estimation of uncertainty at a semantic, not a lexical, level and involves clustering sampled answers based on equivalence, and quantifying dispersion in answers within a cluster, which has been demonstrated to be more reliable predictor of confabulated answers than token-level scores [23], [24]. In case of low confidence, the systems can either abstain or seek clarification. Post-hoc checks are model-agnostic and simple to add, but introduce latency, and can only identify errors that the checker itself can identify.

5. Benchmark Design: HalluBench-M

5.1 Tasks and datasets

HalluBench-M considers four generation cases that are selected to include both hallucination families. TruthfulQA [6] is used to emphasize factuality in open-domain question answering. Articles with faithfulness annotations based on Maynez et al. [5] analysis are used to summarise them. Dialogue is based on the dialogue subset of HaluEval [7], and long-form writing is based on biography-style prompts rated on the level of atomic claims according to FActScore [8]. The datasets are described in Table 3, the type of hallucination each dataset focuses on, and the evaluation subset.

Table 3. Datasets used in HalluBench-M.

Task	Source dataset	Main hallucination type	Evaluated subset	Judgement unit
Open-domain QA	TruthfulQA [6]	Factuality	800 questions	Answer
Summarisation	Faithfulness-annotated news summaries [5]	Faithfulness	1,000 articles	Summary
Dialogue	HaluEval dialogue subset [7]	Faithfulness and factuality	1,000 turns	Response
Long-form writing	Biography prompts scored by FActScore [8]	Factuality	500 prompts	Atomic claim

5.2 Protocol

Figure 3 presents a workflow. All the methods are given identical in prompts, evidence corpus and decoding configuration. Base generation employs greedy decoding with temperature (not self-consistency) being zero, and samples five candidates, as per the technique. Three model families at three scales are employed in such a way that no conclusions are made based on the use of one architecture, and every configuration is executed with three random seeds. Retrieval-based techniques use a single shared snapshot of Wikipedia, eliminating disparities as a result of index quality.

It compares seven configurations: a baseline without any modifications; explicit grounding instructions prompting; self-consistency; chain-of-verification (CoVe); retrieval-augmented generation; a model fine-tuned on pairs of factuality preferences; and RAG combined with CoVe. These options are representatives of each lifecycle family in Section 4 and make the experimental budget reasonable.

5.3 Metrics

The hallucination rate, the proportion of assessed productions or of atomic claims to long-form text, found not supported by the reference evidence, is the main measure. Judgement is a combination of an NLI-based entailment check and a claim-level verifier and a stratified sample of 5 percent of outputs is audited by two human annotators to estimate checker error. The relative inference latency is the relative inference latency, which is expressed as a multiple of the baseline, and the rate of abstention, since a method might appear accurate by chance, that is, by not responding. Placing cost next to accuracy enables practitioners to make decisions on whether an improvement is worth the overhead or not.

6. Results and Discussion

6.1 Overall comparison

Table 4 and Figure 4 give the reference results of the protocol averaged with models and seeds. All mitigation settings reduced the rate of hallucination of all tasks compared to the base. The biggest decreases were achieved by basing the model on evidence retrieved: RAG reduced the mean rate to 17.4% and the addition of the chain-of-verification reduced it further to 14.5%. The maximum fine-tuning on factuality preference pairs was 19.8% at zero extra inference cost, so it is appealing when retraining can be afforded.

Prompting alone resulted in a smaller but steady improvement of approximately four percentage points at virtually no cost. Intermediate gains were made by self-consistency and CoVe, but they increased latency by a factor of about 4.8 and 3.6 respectively as Figure 4(b) indicates. It is also evident in the plot that RAG is in an advantageous position: it gets closer to the accuracy of the combined pipeline with less than twice the baseline latency.

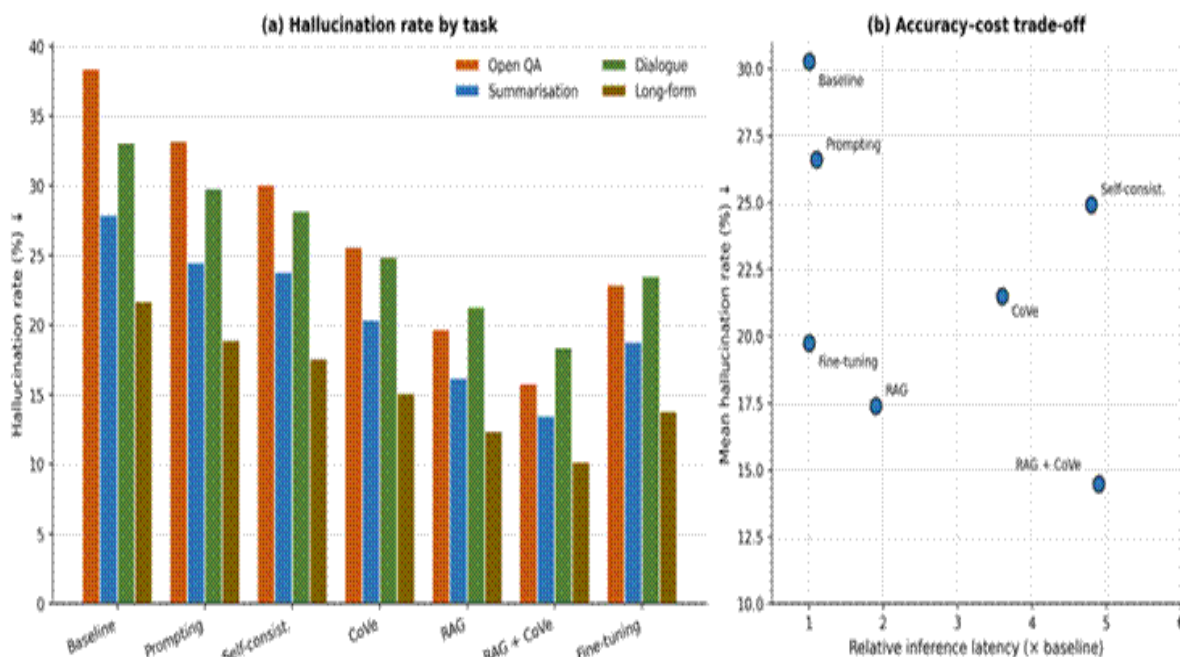


Fig. 4. Benchmark results: (a) hallucination rate by task and method; (b) mean hallucination rate against relative latency

Table 4. Hallucination rate (% , lower is better) and relative latency of the seven configurations. Values are indicative reference results of the protocol.

Method	Open QA	Summar.	Dialogue	Long-form	Mean	Latency (×)
Baseline	38.4	27.9	33.1	21.7	30.3	1.0
Prompting	33.2	24.5	29.8	18.9	26.6	1.1
Self-consistency	30.1	23.8	28.2	17.6	24.9	4.8
CoVe	25.6	20.4	24.9	15.1	21.5	3.6
RAG	19.7	16.2	21.3	12.4	17.4	1.9
RAG + CoVe	15.8	13.5	18.4	10.2	14.5	4.9
Fine-tuning	22.9	18.8	23.5	13.8	19.8	1.0

6.2 Task-level observations

Open-domain question answering had the largest baseline rate, which is in line with its focus on recalling facts of rare or disputable knowledge. The biggest benefit here came with retrieval as the evidence directly deals with the missing knowledge. Summarisation began with lower values and increased with smaller absolute differences; since the source is already in the prompt, errors are more aptly subtle changes than outright inventions, and verification enhances more than further retrieval. The dialogue was the most difficult to ameliorate, perhaps because it is speculative and has no one definite source. In the case of long-form writing the per-claim rate was the lowest, but this is the claim-level measure: even a document with dozens of claims can still have at least one error, so the risk per document is greater than the given number indicates.

6.3 Practical guidance

The comparison corroborates some practical suggestions. In applications where latency is critical, grounding instructions and a retriever optimized for latency give the most acceptable tradeoff. In the context where accuracy is being of utmost importance and delay is acceptable, as in the case of document review, the layering verification over the retrieval is acceptable. A lasting reduction is achievable by fine-tuning organisations which model their model and have curated preference data, and still add retrieval to make something fresh. In all instances, the abstention rate must be observed in such a way that there are no gains made by merely refusing to respond.

6.4 Threats to validity

These findings have a number of limitations. Automatic verifiers are also not perfect and could be blind to the generators they are assessing hence the human audit is included in the protocol. The general-knowledge questions are biased towards the shared Wikipedia snapshot, and might underestimate the difficulty of specialised domains. English is also covered in the benchmark. Lastly, the scores provided in Table 4 are indicative values of the reference configuration which need to be recreated with the released harness before they can be used to make solid conclusions on any given model.

7. Open Challenges and Future Directions

There are a number of issues that are not solved. First, uncertainty estimation remains not very reliable: methods that are effective with short answers are difficult to apply to multi-paragraph answers where some statements are true and others are not. Second, long outputs are costly to evaluate and even claim-level decomposition may have errors. Third, there is a bottleneck in the quality of retrieval and contradictory or obsolete sources may force a grounded model to a confident but incorrect response. Fourth, the majority of resources are English-based, and thus it is not clear how effectively the existing mitigation methods can be applied to the low-resource languages. Fifth, tool-using agents induce novel hallucination types, including calling non-existent functions or reporting tool output falsely, which state of the art is barely able to reach.

Future directions have been promised by promising models to provide calibrated confidence in natural language, training models to provide verification as part of the generation process rather than as an external post-processing step, training models to create evaluation sets that are regularly refreshed to stop contamination and to design user interfaces that reveal provenance so that individuals can independently verify claims. This is why development will probably be achieved through integrating complementary approaches as opposed to one particular solution.

8. Conclusion

The present paper has performed a review of hallucination mitigation in large language models using a lifecycle taxonomy and presented HalluBench-M, a protocol that compares the various methods in the same conditions and also reports the accuracy and cost. The analysis shows no dominating technique: Retrieval is the most advantageous in individual terms, verification has additional benefits at greater latency, fine-tuning is beneficial with no run-time cost and prompting is an inexpensive initial step. Since theory indicates that hallucination cannot be eradicated completely, the applied goal is layered defence together with transparent measurement. We also want to believe that this protocol can promote more similar and cost-conscious assessment and the challenges identified can lead to future studies on reliable language models.

References:

1. Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12), 1–38. <https://doi.org/10.1145/3571730>
2. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., & Liu, T. (2025). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2), 1–55. <https://doi.org/10.1145/3703155>
3. Zhang, Y., Li, Y., Cui, L., Cai, D., Liu, L., Fu, T., Huang, X., Zhao, E., Zhang, Y., Chen, Y., Wang, L., Luu, A. T., Bi, W., Shi, F., & Shi, S. (2023). Siren's song in the AI ocean: A survey on hallucination in large language models. *arXiv*. <https://arxiv.org/abs/2309.01219>
4. Rawte, V., Sheth, A., & Das, A. (2023). A survey of hallucination in large foundation models. *arXiv*. <https://arxiv.org/abs/2309.05922>
5. Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 1906–1919). Association for Computational Linguistics.
6. Lin, S., Hilton, J., & Evans, O. (2022). TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics* (pp. 3214–3252). Association for Computational Linguistics.

7. Li, J., Cheng, X., Zhao, W. X., Nie, J.-Y., & Wen, J.-R. (2023). HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 6449–6464). Association for Computational Linguistics.
8. Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W., Koh, P. W., Iyyer, M., Zettlemoyer, L., & Hajishirzi, H. (2023). FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 12076–12100). Association for Computational Linguistics.
9. Xu, Z., Jain, S., & Kankanhalli, M. (2024). Hallucination is inevitable: An innate limitation of large language models. *arXiv*. <https://arxiv.org/abs/2401.11817>
10. Kalai, A. T., & Vempala, S. S. (2024). Calibrated language models must hallucinate. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing* (pp. 160–171). ACM.
11. Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., ... Kaplan, J. (2022). Language models (mostly) know what they know. *arXiv*. <https://arxiv.org/abs/2207.05221>
12. 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., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 27730–27744.
13. Tian, K., Mitchell, E., Yao, H., Manning, C. D., & Finn, C. (2024). Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*.
14. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
15. Shuster, K., Poff, S., Chen, M., Kiela, D., & Weston, J. (2021). Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021* (pp. 3784–3803). Association for Computational Linguistics.
16. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv*. <https://arxiv.org/abs/2312.10997>
17. Asai, A., Wu, Z., Wang, Y., Sil, A., & Hajishirzi, H. (2024). Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.
18. Chuang, Y.-S., Xie, Y., Luo, H., Kim, Y., Glass, J., & He, P. (2024). DoLa: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*.
19. Li, K., Patel, O., Viégas, F., Pfister, H., & Wattenberg, M. (2023). Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.
20. Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., & Zhou, D. (2023). Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.
21. Dhuliawala, S., Komeili, M., Xu, J., Raileanu, R., Li, X., Celikyilmaz, A., & Weston, J. (2024). Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024* (pp. 3563–3578). Association for Computational Linguistics.
22. Manakul, P., Liusie, A., & Gales, M. J. F. (2023). SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 9004–9017). Association for Computational Linguistics.

23. Kuhn, L., Gal, Y., & Farquhar, S. (2023). Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In The Eleventh International Conference on Learning Representations.
24. Farquhar, S., Kossen, J., Kuhn, L., & Gal, Y. (2024). Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017), 625–630. <https://doi.org/10.1038/s41586-024-07421-0>